



HAL
open science

Modélisation probabiliste et exploration visuelle autonome pour la reconstruction de scènes inconnues

Grégory Flandin

► **To cite this version:**

Grégory Flandin. Modélisation probabiliste et exploration visuelle autonome pour la reconstruction de scènes inconnues. Robotique [cs.RO]. Université Rennes 1, 2001. Français. NNT : . tel-00843884

HAL Id: tel-00843884

<https://theses.hal.science/tel-00843884>

Submitted on 12 Jul 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

N° d'ordre : 2585

THÈSE

présentée

DEVANT L'UNIVERSITÉ DE RENNES I

pour obtenir

le grade de : *DOCTEUR DE L'UNIVERSITÉ DE RENNES I*

Mention : Informatique

par

Grégory Flandin

Équipe d'accueil : VISTA (IRISA/INRIA, RENNES)

École doctorale : Mathématiques, Informatique, Signal, Electronique et
Télécommunications

Composante Universitaire : Institut de formation supérieure
en informatique et communication

Titre de la thèse :

**Modélisation probabiliste et exploration visuelle autonome
pour la reconstruction de scènes inconnues**

Soutenue le 29 Novembre 2001, devant la commission d'examen

COMPOSITION DU JURY :

MM.	Raja Patrick	CHATILA RIVES	Rapporteurs
-----	-----------------	------------------	-------------

MM.	Dominique Gaëtan Claude François	MEIZEL GARCIA LABIT CHAUMETTE	Examineurs
-----	---	--	------------

Table des matières

Notations	5
Introduction	7
1 Contexte et contributions	9
1.1 Représentation de l'environnement	10
1.1.1 Information d'encombrement et information structurelle	10
1.1.2 Décomposition d'une scène en objets	11
1.2 Décomposition hiérarchique de l'exploration	12
1.3 Exploration de scènes : un problème de recherche d'objets	15
1.3.1 Contexte	15
1.3.2 Approche proposée	16
1.4 Exploration des objets : un problème de reconstruction	18
1.4.1 Reconstruction d'objets	19
1.4.1.1 L'enveloppe visuelle	19
1.4.1.2 Les modèles approchés	19
1.4.1.3 Approche proposée	21
1.4.2 Modèles mixtes	21
1.4.2.1 Géométrie probabiliste et transformation des lois	22
1.4.2.2 Modèles à erreur bornée	23
1.4.2.3 Approche proposée	24
1.4.3 Exploration pour la reconstruction des objets	25
1.4.3.1 Contexte	25
1.4.3.2 Approche proposée	27
1.5 Asservissement visuel	27
1.6 Détection d'objets au sens du mouvement	29
1.7 Plan du rapport	31
I Recherche des objets dans une scène encombrée	33
Rappels concernant l'approche proposée	35
1 Modélisation de l'environnement et intégration de contraintes	37
1.1 Modélisation	37
1.2 Principe d'intégration	42

1.3	Méthode itérative	43
1.4	Modification d'une contrainte	46
1.4.1	Principe	46
1.4.2	Permutation d'une contrainte	48
1.4.3	Calcul de $\widetilde{F}_{k/k}$	49
1.4.4	Résumé des équations de modification	49
1.5	Suppression d'une contrainte	50
1.6	Contraintes inégalité	54
1.7	Redondance et incompatibilité	56
1.8	Conclusion	57
2	Algorithme de fusion de l'observation et stratégies de recherche	59
2.1	Intégration des observations	59
2.1.1	Algorithme d'intégration	59
2.1.2	Initialisation	60
2.1.3	Modèle d'observation	63
2.2	Stratégies de recherche	66
2.2.1	Configurations atteignables et t-chemins	66
2.2.2	Espérance du gain d'information (EGI)	67
2.2.3	Fonction gain d'information (FGI)	68
2.3	Conclusion	70
3	Simulations	73
3.1	Choix de la fonction gain d'information	73
3.1.1	Somme des différences	74
3.1.2	Norme 2	75
3.1.3	Norme ∞	75
3.1.4	Variation de l'entropie	75
3.2	Influence de l' <i>a priori</i>	76
3.3	Stratégie gloutonne	76
3.4	Conclusion	78
	Conclusion	89
II	Reconstruction approchée et exploration des objets complexes	91
	Rappels concernant l'approche proposée	93
1	Modélisation	95
1.1	Modélisation des objets : la loi d'ensemble	95
1.2	Modélisation des observations	98
2	Propagation	101
2.1	Transformations générales	101
2.2	Propagation des données visuelles	105
2.2.1	Extraction des données visuelles	105

2.2.2	Rétro-projection	106
2.2.2.1	Configuration monoculaire	106
2.2.2.2	Configuration binoculaire	107
2.2.3	Projection perspective	109
2.2.4	Changement de repère	110
2.3	Conclusion	111
3	Estimation	113
3.1	Initialisation	113
3.2	Fusion de nouvelles observations	114
3.2.1	Estimation de l'incertitude	115
3.2.2	Estimation du support de l'erreur	116
3.2.2.1	Approximation externe E_{k+1}^+	116
3.2.2.2	Approximation interne E_{k+1}^-	119
3.2.2.3	Approximation intermédiaire	120
3.3	Simulations	121
3.3.1	Estimation de la position de l'objet	122
3.3.2	Estimation des axes de l'enveloppe	126
3.3.3	Cas d'un objet non sphérique	127
3.4	Conclusion	130
4	Exploration	131
4.1	Information prédite <i>a posteriori</i> (IPP)	132
4.2	Loi de commande exploratoire	134
4.2.1	Exploration localement optimale (LO)	135
4.2.2	Loi de commande basée sur le meilleur point de vue (BVP)	138
4.3	Simulations	139
4.3.1	Loi de commande LO	140
4.3.2	Loi de commande BVP	141
4.3.3	Critère d'arrêt	142
4.3.4	Comparaison des performances	143
4.4	Conclusion	144
5	Expérimentations	147
5.1	Cas d'un seul objet	148
5.2	Cas de deux objets	148
	Conclusion	157
	Conclusion et perspectives	159
	Annexes	167
A	Asservissement visuel et coopération	169
A.1	Introduction	169

A.2	Description et modélisation de la tâche	170
A.2.1	Commande de translation	171
A.2.2	Commande en rotation	174
A.2.3	Interaction	175
A.3	Coopération	175
A.3.1	Estimation de la perturbation	177
A.3.2	Redondance de tâches	178
A.4	Résultats expérimentaux	179
A.4.1	Estimation et compensation de l'erreur	179
A.4.2	Formalisme de la redondance	179
A.5	Conclusion	183
B	Changement de repère et incertitudes d'un robot cartésien 6 axes	185
	Bibliographie	187
	Publications	201

Notations

Abréviations

FGI.....	Fonction Gain d'Information.
EGI.....	Espérance du Gain d'Information.
NIM.....	Nombre d'Itérations Moyen.
MAP.....	Maximum A Posteriori.
LO.....	Loi de commande Localement Optimale.
BVP.....	Loi de commande basée sur le meilleur point de vue (Best View Point).

Notations

\mathbb{I}	Matrice identité.
$\llbracket i, j \rrbracket$	Intervalle des valeurs discrètes entre i et j .
$\mathcal{P}(e)$	Probabilité de l'événement e .
$\mathcal{P}(e_1 e_2)$...	Probabilité de l'événement e_1 sachant l'événement e_2 .

Partie I

\mathcal{A}	Axe global définissant une zone de recherche.
\mathcal{C}_i	Cellule issue d'une discrétisation de \mathcal{A} .
X_i	Variable aléatoire d'occupation de \mathcal{C}_i .
p_k	Vecteur des probabilités conjointes des X_i à l'instant k .
(h, z)	Contrainte de valeur z dans la direction h .
F	Matrice de variabilité.
\mathcal{C}_p	Cellule définissant la position de la caméra.
\mathcal{C}_d	Cellule visée par la caméra.
\mathcal{C}	Configuration de la caméra représentée par le couple $(\mathcal{C}_p, \mathcal{C}_d)$.
\mathcal{C}_t	Chemin d'ordre t : succession de t configurations atteignables.

Partie II

\mathcal{O}	Objet à reconstruire.
c	Point quelconque de \mathcal{O} .
$\mathcal{E}(,,,).....$	Distribution d'ensemble.
\bar{c}	Position de \mathcal{O} représentée par le centre de son ellipsoïde englobant.
P	Matrice de covariance représentant l'incertitude sur \bar{c} .
Σ	Inverse de P .
E	Matrice symétrique positive représentant l'ellipsoïde englobant \mathcal{O} .
\mathcal{E}^i	Distribution d'ensemble 2D calculée dans l'image.
\mathcal{E}^c	Distribution d'ensemble 3D exprimée dans le repère de la caméra.
\mathcal{E}^o	Distribution d'ensemble 3D exprimée dans le repère de référence.
R	Matrice de rotation.
t	Vecteur de translation.
u	Axe de rotation.

Introduction

Vers la fin des années 70, la vision par ordinateur a connu une première évolution importante au vu de son impact sur la recherche moderne. Alors que les systèmes de vision artificielle s'apparentent à du "bricolage", David Marr [Marr 82] et ses collègues montrent la nécessité de s'appuyer sur des modules fondamentaux correspondant à des fonctionnalités identifiables du système visuel humain. Ces fonctionnalités peuvent se résumer de la façon suivante : extraction de primitives, représentation spatiale et reconnaissance. Cette méthodologie a eu le mérite d'initier un lent processus de formalisation : dans le début des années 80, la vision par ordinateur devient progressivement un domaine de recherche à part entière. Mais il faut bien reconnaître que les théories de Marr atteignent rapidement leur limite dans la mesure où elles s'appuient sur une acquisition passive des données. Cette passivité est à l'origine de nombreuses difficultés dont, notamment, le manque de robustesse et de stabilité des algorithmes associés et l'inobservabilité partielle due, d'une part, à la perte de l'information de profondeur dans le système visuel et, d'autre part, aux occultations engendrées par les objets en présence. La fin des années 80 marque le début des recherches sur l'influence du comportement de l'observateur et montre l'intérêt d'un contrôle intelligent des paramètres de la caméra (sa position, son mouvement, sa focale, ...). Les pionniers de l'ère de la vision active sont : Aloimonos [Aloimonos 88], Bajcsy [Bajcsy 88], Ballard [Ballard 91] et Tsotsos [Tsotsos 92].

Problème posé

Le problème auquel nous nous intéressons est, au cœur de la vision active, celui de l'exploration. Si le concept d'exploration semble relativement bien défini pour un être humain : "action de parcourir un lieu totalement ou partiellement inconnu en l'étudiant avec soin", celui-ci apparaît bien moins explicite lorsqu'il s'agit de demander à un système d'étudier un lieu avec soin ! L'exploration est un terme vaste qui regroupe divers aspects ; aussi devrions-nous parler, plus justement, d'exploration pour la reconstruction. Étant donnée une scène dans laquelle se trouve un système d'acquisition visuelle, nous souhaitons déterminer la suite des actions permettant de reconstruire une carte plus ou moins détaillée de l'environnement. Cette vaste problématique mérite quelques précisions :

- Nous entendons par scène un domaine restreint de l'espace composé d'objets. Cette scène est supposée initialement totalement inconnue sans toutefois écarter la possibilité de prendre en compte certaines connaissances *a priori* susceptibles de guider le système.
- La suite des actions à mener est complètement autonome, c'est-à-dire que le système

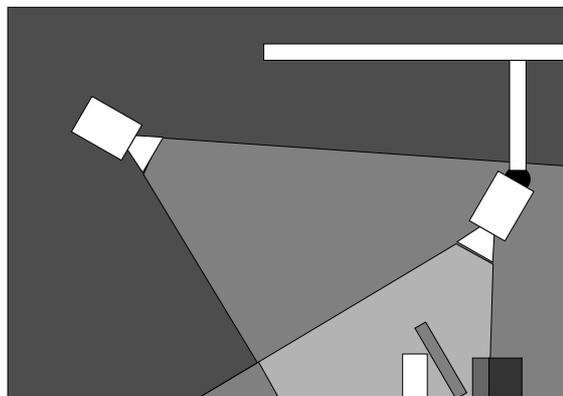
doit parvenir, sans intervention extérieure, à un état de connaissance satisfaisant de la scène.

- Le système visuel considéré est, de façon générale, un ensemble de caméras dont la configuration sera explicitée ultérieurement.

En résumé, nous nous intéressons à l'**exploration visuelle autonome pour la reconstruction de scènes inconnues**.

Coopération globale/locale

Le problème, tel que nous l'avons posé, ne dit pas si l'observateur, en l'occurrence le système visuel, se trouve immergé dans la scène ou, au contraire, extérieur à celle-ci. Ce détail, volontairement omis dans le paragraphe précédent, mérite notre attention. En effet, dans un contexte très général d'observation, nous distinguons deux types d'acquisitions correspondant à des points de vue différents. Une **image globale** est une image acquise d'un point de vue tel que son champ de vision englobe la totalité de la scène d'intérêt ; le point de vue est alors extérieur à la scène. Une **image locale** se focalise, au contraire, sur une partie réduite de l'environnement ; le point de vue est intérieur à la scène. En pratique ces deux types d'images peuvent faire l'objet de systèmes d'acquisition distincts, nous parlons alors de **caméra déportée** pour l'image globale et de **caméra embarquée** pour l'image locale, ou bien d'une caméra unique dont les capacités de manœuvre permettent d'accéder aux deux types de points de vue. Pour des tâches d'observation en environnement encombré, ces deux types d'images apparaissent très complémentaires : l'image locale dispose d'une vue précise mais restreinte contrairement à l'image globale qui dispose d'une vue large mais imprécise et comportant généralement des occultations importantes. Des applications susceptibles d'utiliser la complémentarité de ces deux configurations sont, par exemple, la surveillance de zones sensibles où des robots mobiles équipés de caméras permettent à une caméra fixe de surveillance d'acquérir des informations supplémentaires, la robotique de manipulation où la caméra déportée joue le rôle d'aide à la planification de tâches réalisées par la caméra embarquée... Dans l'ensemble de nos travaux nous cherchons à tirer partie de cette complémentarité dans le cadre des problèmes d'exploration. C'est pourquoi toutes nos études sont teintées de la coopération entre des observations globales et des observations locales.



Coopération entre une observation globale et une observation locale.

Chapitre 1

Contexte et contributions

Dans ce chapitre, nous décrivons l'ensemble des problématiques liées à l'exploration autonome pour la reconstruction. Nous replaçons chacune d'elles dans son contexte bibliographique, ce qui nous permet de justifier l'approche que nous adoptons et les différentes contributions apportées.

- La section 1.1 est consacrée à la représentation de l'environnement. Nous posons les principes d'une modélisation efficace afin de justifier les choix que nous ferons, ultérieurement, en termes de représentation mathématique de la scène. Ces choix dépendent à la fois de la tâche que le système doit effectuer et de l'environnement dans lequel il évolue.
- En section 1.2, nous décomposons le problème de l'exploration en fonctionnalités indépendantes et de niveaux hiérarchiques décroissants : l'exploration de scènes, l'exploration d'objets, l'asservissement visuel et le traitement d'images. Dans les sections qui suivent, nous détaillons chacune de ces fonctionnalités qui appartiennent à des domaines de recherche relativement différents.
- Le plus haut niveau d'exploration est défini comme étant un problème de recherche de tous les objets (ou groupes d'objets) de la scène. Cette problématique est définie en section 1.3.
- À un niveau inférieur et pour chaque objet de la scène, l'exploration est posée comme un problème de reconstruction optimale d'objet. La section 1.4 détaille et justifie notre approche concernant ce point.
- Au delà des aspects exploratoires, nous nous sommes intéressés au problème de l'asservissement visuel qui est essentiel au suivi des consignes. L'asservissement visuel permet, notamment, d'assurer la visibilité de l'objet au cours des déplacements. Ce domaine et l'approche que nous adoptons sont présentés en section 1.5.
- Enfin, au plus bas niveau de la hiérarchie, se situent les aspects de traitement d'images. Nous sommes plus spécifiquement concernés par les algorithmes de détection des objets. Ceux-ci ne constituent pas le cœur de nos travaux mais demeurent essentiels à l'implémentation des algorithmes d'exploration. Nous nous intéresserons plus particulièrement à la détection du mouvement pour des raisons que nous justifierons en section 1.6.

1.1 Représentation de l'environnement

La définition d'un modèle pour l'environnement est la question centrale de tout problème d'exploration et de reconstruction. Aussi nous jugeons important de préciser les qualités que requiert une modélisation efficace de l'environnement. Le modèle doit être :

1. **adapté et concis** : S'il est impératif que le modèle prenne en compte les spécificités de l'environnement et de la tâche assignée, il est, en revanche, important de restreindre sa complexité au strict nécessaire. Cette mise en garde est essentielle lorsque le système évolue dans un milieu complexe et de façon réactive. La représentation doit, de plus, être adaptée aux données. Les capteurs visuels fournissent une connaissance en essence incertaine et partielle de la scène observée. Le modèle doit donc permettre la comparaison, la fusion et l'intégration de données incertaines et partiellement observées.
2. **général et unifié** : La décision de construire un modèle concis donc simplifié ne doit rien enlever à la généralité du formalisme. Nous attachons une grande importance à l'aspect unifié de la représentation. L'intérêt étant de pouvoir manipuler aisément, et pour le plus large éventail de situations possibles, aussi bien les données réelles que les équations mathématiques sous-jacentes.
3. **constructif** : Enfin l'aspect constructif d'un modèle répond de la pertinence et de la variété des informations que l'on peut en extraire. En ce qui concerne les problèmes de reconstruction, nous pensons que l'information à extraire d'un modèle de représentation de scène contient deux aspects essentiels : l'**information d'encombrement** et l'**information structurelle**.

1.1.1 Information d'encombrement et information structurelle

L'information d'encombrement est un aspect purement géographique qui détermine si telle ou telle zone est occupée ou non. En guise d'exemple, les modèles échantillonnés contiennent l'information d'encombrement dans la mesure où ils fournissent l'occupation de chaque voxel (voir [Slabaugh 01] pour une bibliographie récente). Cette classe de techniques, pour lesquelles une discrétisation volumique exhaustive de la scène est nécessaire, est souvent associée à des techniques hiérarchiques telles que les octrees [Garcia 98, Potmesil 87, Szeliski 93, Zell 95] dans le but de réduire la taille de la représentation. L'information d'encombrement est nécessaire pour les problèmes de navigation dans des zones encombrées mais reste insuffisante si l'on s'intéresse à la structure de la scène comme c'est le cas, par exemple, pour la manipulation d'objets. L'information structurelle est, quant à elle, donnée en termes de constituants ou d'objets et concerne l'aspect géométrique des entités présentes dans la scène. Classiquement on accède à cette information soit en considérant des objets de forme géométrique, auquel cas les techniques sont basées sur la reconstruction de diverses primitives [Faugeras 87, Chaumette 96], soit en représentant les objets par des maillages [Boissonnat 84].

La notion d'information structurelle, purement géométrique, n'est pas suffisante en présence d'incertitudes. Pour prendre en compte les incertitudes inhérentes aux capteurs visuels, les modèles que nous dérivons sont essentiellement probabilistes. Nous verrons en section 1.4 que, pour la construction d'un modèle d'objet, le choix d'un modèle probabiliste judicieusement paramétré permet d'unifier à la fois des aspects géométriques et des aspects

probabilistes. Le choix d'une représentation probabiliste est d'autant plus intéressant qu'il permet de modéliser efficacement l'inobservabilité partielle des mesures. Il suffit pour cela de s'accorder la possibilité de modéliser des variances infinies.

La définition de l'information structurelle nécessite donc l'introduction du concept d'objet. Celui-ci n'est pas clairement défini puisque la notion d'entité ou de constituant d'une scène n'est pas unique et dépend de la tâche assignée.

1.1.2 Décomposition d'une scène en objets

Un grand nombre d'études ont porté sur la décomposition en parties en tant que recherche d'une partition qui se conforme à une décomposition naturelle de la scène ou d'un objet [Hoffman 84, Pentland 89]. Beaucoup de ces travaux ont été motivés par les problèmes de reconnaissance, considérant que le processus humain de reconnaissance d'objets est basé sur une pré-segmentation de celui-ci en composantes, suivie d'une classification de chacune de ces composantes. Hoffman [Hoffman 84] a, en quelque sorte, posé les bases de la recherche d'une partition dans l'image. Son analyse concerne les points singuliers du contour. En basant son étude sur l'explication de certaines illusions visuelles, il décrit des règles concernant la régularité du contour. Pentland [Pentland 89] décrit une méthode de partitionnement de l'image en deux étapes : une étape de mise en correspondance avec une collection de silhouettes prédéfinies permettant d'extraire un certain nombre d'hypothèses sur la structure du partitionnement et une étape d'optimisation globale sur l'ensemble de ces hypothèses permet alors d'exhiber la décomposition la plus probable de l'image. Les travaux que nous citons ne constituent pas une liste exhaustive puisque la littérature est très riche dans ce domaine.

Les études menées par Ferrie [Ferrie 88, Ferrie 92] concernent le partitionnement d'objets dont la surface est représentée par un ensemble de points 3D. Dans [Ferrie 88], le processus de reconstruction à partir d'un capteur laser est décrit en trois étapes : une étape de décomposition en parties basée sur la courbure, l'estimation des primitives correspondant à chaque partie et enfin l'élimination des primitives redondantes. Dans [Ferrie 92], le partitionnement du contour des objets utilise une stratégie similaire à celle proposée par Zucker en 2D [Zucker 88]. Celle-ci est basée sur l'ajustement de splines par minimisation d'une fonction d'énergie bien choisie. Citons enfin d'autres travaux [Leonardis 94, Gupta 93] utilisant des modèles superquadriques à des fins de décomposition en parties. Dans la mesure où, pour ces études, la reconstruction du modèle est fortement corrélée à sa décomposition, nous préférons les détailler ultérieurement (section 1.4.1.2).

L'ensemble des travaux que nous venons de citer montre l'intérêt porté au problème de la décomposition en parties et la possibilité de le résoudre en se basant sur des considérations liées à l'apparence. En effet toutes les études utilisant soit le contour soit les points singuliers de la représentation visent une décomposition la plus naturelle possible. Le reproche que nous adressons à cette approche est qu'elle ne considère pas l'aspect fonctionnel de la décomposition. Gupta et Bajcsy [Gupta 93] sont les seuls à proposer une procédure itérative capable de s'adapter à la complexité de la description. Suivant la tâche considérée, la décomposition souhaitée peut être différente.

Sans apporter de réponse à ce problème qui n'est pas au cœur de nos préoccupations, nous préférons donner une définition générale d'un objet permettant de s'affranchir des problèmes d'interprétation subjective. Il n'est pas restrictif de considérer que toute scène

puisse être partitionnée en groupes ou parties d'objets *cohérents*. Le terme cohérent doit être compris dans le sens où le groupe, ou la partie, en question peuvent être décrits de façon significative, vis-à-vis de la tâche assignée, par un simple volume englobant. Cette notion est à rapprocher de la notion appelée *significant part* par Gupta et Bajcsy [Gupta 93]. Typiquement, un ensemble d'objets, proches les uns des autres, est cohérent s'il est observé de suffisamment loin mais il est incohérent lorsqu'on l'observe de près auquel cas chaque objet doit être traité comme une partie cohérente. Même un objet isolé peut apparaître incohérent lorsqu'il est constitué de plusieurs parties géométriquement distinctes. Lorsque ce sera nécessaire chaque partie devra être traitée individuellement. Par abus de langage, nous appellerons "objet" tout groupe ou partie d'objets physiques qui soit cohérent. Ainsi, pour une tâche donnée, la scène d'intérêt est formée d'objets clairement identifiés. Dans la suite de nos travaux, nous considérons acquise cette décomposition en objets.

1.2 Décomposition hiérarchique de l'exploration

Dans cette section, nous présentons quelques travaux très généraux sur l'exploration et tentons de décomposer le problème en deux niveaux hiérarchiques distincts.

Dans le contexte de l'exploration autonome, Connolly apparaît comme un pionnier avec ce qu'il appelle *l'algorithme du planétarium*. Dans [Connolly 85], il s'intéresse à la détermination des points de vue d'une image télémétrique issue d'un capteur laser permettant la reconstruction d'une scène inconnue. Cette scène est modélisée par des voxels structurés en octrees. Chacun de ces voxels peut prendre la valeur "vide", "occupé" ou "non-observé". Les algorithmes d'exploration proposés sont basés sur la maximisation du volume découvert et donc la réduction du nombre de cellules non-observées. Par l'intermédiaire des cellules occupées, ils tiennent compte des occultations éventuelles. Le schéma de base est très intuitif puisqu'il consiste à calculer l'ensemble des cellules découvertes pour tous les points de vue possibles. Les points de vue possibles appartiennent à une sphère centrée sur la scène. Même si l'auteur propose une version allégée de cette recherche exhaustive, l'algorithme apparaît très coûteux en temps de calcul. Si d'autres reproches peuvent lui être adressés concernant la gestion des incertitudes et des *a priori*, l'intérêt de l'algorithme est indéniable.

Toutefois, la faiblesse principale de l'algorithme de Connolly réside sans doute dans le fait qu'il confond, de par la modélisation, l'exploration d'une scène et l'exploration des objets qui la constituent. En fait la notion d'objet n'a pas besoin d'être clairement identifiée puisque la scène est modélisée par des voxels qui sont soit occupés soit vides. Marchand et Chaumette [Marchand 96, Marchand 99] ont introduit une distinction entre exploration locale et exploration globale.

En effet, ils utilisent la vision monoculaire active pour explorer une scène composée d'objets géométriques tels que cylindres ou polygones. À un niveau local, les travaux de Chaumette et Boukir [Chaumette 96] permettent de contraindre les mouvements de la caméra de manière à optimiser la qualité des résultats de reconstruction des primitives constituant les objets (points, droites, cylindres, ...). À un niveau supérieur, une stratégie de perception est développée qui commande les mouvements de la caméra de manière à observer l'ensemble de la scène tout en focalisant celle-ci sur des zones particulières, résolvant ainsi le problème soulevé par les occultations. Cette stratégie de perception est divisée en

deux niveaux distincts. La première phase est locale dans le sens où seules les informations disponibles sont utilisées afin de déterminer sur quelle primitive la caméra doit se focaliser. Elle repose sur une approche de prédiction/vérification d'hypothèses gérées à l'aide de réseaux Bayésiens. Le second niveau vise à assurer la complétude de la reconstruction de la scène grâce à un calcul explicite de nouveaux points de vue provenant de la minimisation d'une fonction de coût et permettant d'obtenir un maximum d'informations sur les zones encore inconnues de la scène.

Beaucoup d'autres travaux se sont appliqués à définir une décomposition hiérarchique exhaustive des tâches. Citons [Moutarlier 91, Grandjean 91, Chatila 93, Chatila 95, Lacroix 95, Lacroix 97], dont les travaux sont axés plus particulièrement vers des tâches de robotique mobile. La décomposition peut concerner la modélisation de la scène, des tâches de navigation ou d'exploration. Grandjean [Grandjean 91] décrit le processus d'interprétation de scènes comme un processus de structuration hiérarchique. À chaque niveau d'interprétation correspond un niveau de conceptualisation de l'environnement plus ou moins explicite. Cette structuration permet au système de se focaliser sur les primitives pertinentes. Moutarlier [Moutarlier 91] présente un modèle géométrique intégrant une représentation surfacique probabiliste, un modèle incrémental de l'espace libre et des interprétations topologiques de l'environnement. À chacun de ces modèles est associé un processus de construction spécifique. En termes de modélisation, le choix de la représentation utilisée est particulièrement important en environnement naturel en raison de l'absence de structure sous-jacente. Dans [Chatila 93] les auteurs expliquent comment regrouper les différentes représentations d'une même entité au sein d'une structure multi-niveaux dont le niveau supérieur est une représentation discrétisée. Un processus de classification permet d'associer, à chaque cellule de cette discrétisation, un qualificatif du type : "obstacle", "plat", "pente", "accidenté" ou "vide". Dans [Chatila 95], est présentée une approche de la navigation autonome en environnement extérieur impliquant différents niveaux de raisonnement et plusieurs modes de déplacement en fonction de la représentation précitée. Les trois niveaux de raisonnement sont, dans un ordre hiérarchique : la planification de routes basée sur la sélection d'objectifs intermédiaires, la planification de chemins et de tâches de perception pour atteindre ces objectifs et, enfin, la planification de la trajectoire à exécuter. Enfin Lacroix et Chatila [Lacroix 97] se sont plus particulièrement intéressés au développement de stratégies de perception et d'action. Un algorithme de recherche fournit un chemin optimal au sein d'un graphe puis ce chemin est analysé pour définir la stratégie de perception associée.

Tous ces travaux montrent très clairement l'intérêt de structurer à la fois la représentation de la scène et les tâches de perception ou d'action considérées. Dans son approche, Marchand décompose une scène en un ensemble de primitives géométriques, ce qui sous-entend que la scène n'est composée que de ces entités interdisant toute application de l'algorithme à des scènes complexes. En ce qui nous concerne, nous préférons situer la frontière hiérarchique au niveau des objets tels que nous les avons introduits précédemment. Ceci se traduit par une approche de l'exploration également en deux niveaux : la découverte des objets puis la reconstruction de chaque objet. Plus précisément, nous considérons que le système doit satisfaire quatre fonctionnalités (voir figure 1.1) :

1. La tâche de plus haut niveau consiste à déterminer la suite des actions aboutissant à un inventaire de tous les objets de la scène. Dans ce processus, nous souhaitons pouvoir intégrer des connaissances *a priori* sur le type de scène considéré et notamment sur son encombrement (dense ou épars, ...); connaissance sans laquelle une

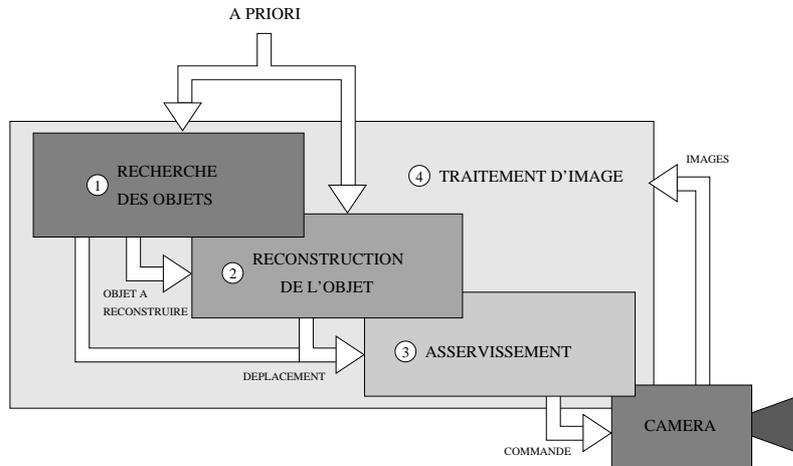


FIG. 1.1 – Nous identifions quatre fonctionnalités nécessaires au développement d'un système d'exploration autonome à des fins de reconstruction : 1- déterminer la suite des actions aboutissant à un inventaire de tous les objets de la scène, 2- pour chacun des objets, déterminer l'ensemble des consignes aboutissant à sa reconstruction, 3- transformer ces consignes en commandes directement applicables au robot et 4- extraire les observations utiles de l'image.

stratégie “gloutonne” consistant à observer toutes les zones de la scène serait vraisemblablement la plus adaptée. À ce stade, l'objectif est de disposer d'une carte des zones à l'intérieur desquelles se trouve un objet, dans l'idée d'appliquer à chacune de ces zones un traitement spécifique. Les observations requises à ce niveau s'énoncent en termes de détection des objets dans le champ de vue de la caméra.

2. Pour chacune des zones identifiées à l'étape 1, un second objectif est de déterminer l'ensemble des consignes aboutissant à la reconstruction de l'objet qu'elle contient. À ce stade, les observations seront les masques de projection des objets dans l'image. Les consignes se traduisent en termes de déplacements permettant d'optimiser la reconstruction.
3. Le problème consistant à déterminer l'ensemble des commandes directement applicables au système visuel, assurant le suivi des consignes de déplacement citées précédemment tout en conservant l'objet dans le champ de vue de la caméra est un problème d'asservissement. La consigne de suivi des objets nécessite, comme observation, les coordonnées du centre de la projection des objets dans l'image.
4. Enfin, en interaction avec chacune des fonctionnalités précédentes, se trouvent les algorithmes de traitement d'images qui fournissent les observations utiles au bon déroulement des étapes 1, 2 et 3. Étant donné les besoins que nous avons décrits précédemment (détection des objets et identification du masque des objets dans l'image), nous nous intéresserons plus particulièrement aux algorithmes de détection d'objets.

Chacune des fonctionnalités que nous venons d'énoncer s'insère dans une communauté et des domaines scientifiques distincts. Dans la suite de ce chapitre, nous replaçons chacune d'elles dans son contexte historique et scientifique puis justifions l'approche choisie.

1.3 Exploration de scènes : un problème de recherche d'objets

La première étape de l'exploration d'une scène est celle consistant à faire l'inventaire de tous les objets la constituant. C'est un problème de recherche d'objets.

1.3.1 Contexte

Beaucoup d'études ont été menées sur la recherche d'objets avec des objectifs et des points de vue très variés. Rimey et Brown [Rimey 92,Rimey 93,Rimey 94] introduisent des relations géométriques dans un réseau bayésien de telle sorte que le système puisse, en fonction de connaissances relationnelles, restreindre la recherche à des zones particulières de la scène, soit en déplaçant la caméra soit en restreignant le traitement à une partie de l'image. Les réseaux bayésiens permettent d'intégrer à la fois une connaissance *a priori* et les observations visuelles. La règle de décision est basée sur la maximisation de l'espérance d'une fonction d'utilité qui est le rapport d'une mesure de gain sur le coût engendré par le déplacement ou le traitement qu'il a fallu appliquer. Le système semble toutefois plus adapté à répondre à une question spécifique avec une certaine confiance (comme c'est le propre des réseaux bayésiens) qu'à des problèmes de recherche d'objets à proprement parler.

Wixson et Ballard [Wixson 94a,Wixson 94b] ont proposé une modélisation de l'efficacité de recherche et ont montré comment améliorer cette efficacité en contraignant la zone de recherche dans le cas particulier d'une recherche indirecte. La notion de recherche indirecte a été introduite par Garvey [Garvey 76]. L'idée est de diriger en priorité le capteur vers des objets intermédiaires qui sont *a priori* en relation spatiale avec l'objet d'intérêt puis d'effectuer la recherche dans la zone spécifiée par cette même relation. L'inconvénient de la recherche indirecte d'un objet est, d'une part, que la relation spatiale présumée entre l'objet recherché et un objet intermédiaire n'existe pas forcément et, d'autre part, que rien ne prouve que la recherche de l'objet intermédiaire, s'il existe, soit plus simple que celle de l'objet recherché.

Ye et Tsotsos [Ye 95,Ye 96b,Ye 96c,Ye 97] ont proposé un formalisme élégant des problèmes de recherche, considérant pour toute partie de l'environnement sa probabilité de contenir l'objet recherché. Une règle de Bayes permet de prendre en compte les observations. Ils ont également proposé une stratégie de maximisation de la reconnaissance répondant successivement à deux problèmes: "Où effectuer la prochaine recherche?" et "Où déplacer le capteur de sorte que la zone à explorer apparaisse non occultée pour le capteur?" La modélisation utilisée fait intervenir le temps de traitement en chaque point de vue en fonction de l'algorithme de reconnaissance utilisé ainsi que la qualité de reconnaissance en fonction de la distance par rapport à la zone observée. Dans [Ye 96a], le formalisme de [Ye 97] est adapté à la coopération multi-agents. La formalisation est élégante mais sa mise en application semble peu évidente et nécessiter des simplifications qui limitent son intérêt. Enfin Ye et Tsotsos ont prolongé des travaux concernant l'analyse de la complexité de la recherche, travaux initiés dans [Tsotsos 90,Tsotsos 92]. Dans [Tsotsos 90], Tsotsos s'intéresse à la complexité du problème très général de la recherche d'une cible dans l'image. Dans [Tsotsos 92], il montre que la recherche visuelle active dans un espace non borné est NP-complet. Soulignons que tous ces travaux considèrent la recherche d'objets comme un problème de reconnaissance.

La communauté en recherche opérationnelle a, d'un point de vue différent, très largement contribué aux problèmes de recherche optimale. Ces études furent, en majorité, initiées par Koopman [Koopman 57] dont les travaux ont été étendus par De Guenin [Guenin 61] et plus récemment par Dambreville [Dambreville 99]. Le problème posé est de trouver une allocation optimale de l'effort de recherche global. La théorie sous-jacente est connue sous le nom de *Search Theory*. Sa résolution est basée sur la détermination d'un modèle de détection adéquat. Ce dernier dépend de la distribution de l'effort de recherche et d'un paramètre de visibilité lié à l'environnement dans lequel la recherche est effectuée ainsi qu'au type de capteur utilisé. La résolution des problèmes de *Search Theory* fait souvent intervenir un nombre considérable de variables (pouvant atteindre plusieurs milliers). Des extensions de la théorie ont permis de dériver des stratégies de recherche en temps limité. En particulier Brown [Brown 80] propose un algorithme itératif basé sur la discrétisation du temps en périodes. Pour chaque période, les ressources allouées sont bornées par une valeur prédéfinie. Or, dans un schéma d'optimisation globale, la probabilité finale de détection est fortement dépendante de la succession des ressources allouées. Aussi Dambreville [Dambreville 99] propose-t-il une stratégie optimale de recherche lorsque les ressources de détection disponibles se renouvellent suivant des contraintes linéaires généralisées.

De façon générale, les travaux concernant la *search theory* furent initialement motivés par des considérations militaires puis appliqués plus tard aux opérations de sauvetage en mer. Dans ce contexte, citons également la théorie des jeux différentiels et ses applications aux problèmes de recherche [Gal 80, Basar 89]. Grâce à leur aspect non coopératif, les jeux permettent de formaliser la recherche d'objets mobiles ou en présence d'incertitudes lorsque l'étude du pire cas est envisagée. L'ensemble des travaux en recherche opérationnelle présente un formalisme très intéressant mais coûteux en calculs. Leur application dans un contexte de réactivité en temps réel n'est pas encore d'actualité mais mériterait véritablement d'être étudiée ; d'autant qu'ils présentent l'avantage de pouvoir s'appliquer à des cibles en mouvement markovien.

Enfin, des travaux plus récents, motivés par la complexité des algorithmes de recherche en ligne, adressent des problèmes de recherche dans des cadres géométriques spécifiques. Le domaine de recherche est soit borné soit infini mais appartient à une classe géométrique particulière. Citons les travaux de Baeza-Yates [BaezaYates 95] concernant la recherche dans un plan, ceux de López-Ortiz pour la recherche le long de rayons [LópezOrtiz 98] ou pour la recherche à l'intérieur de polygones [LópezOrtiz 96]. Tous ces travaux visent à calculer soit un encadrement (bornes supérieures ou inférieures) soit la moyenne d'une quantité appelée "competitive ratio". Pour une stratégie donnée, cette dernière correspond au rapport entre la distance parcourue pour achever la recherche et la distance correspondant au plus court chemin possible. Ces travaux sont particulièrement bien adaptés lorsque l'on dispose d'un *a priori* fort sur la géométrie de la scène et fournissent alors un cadre d'étude très complet. En revanche ils n'offrent pas un cadre probabiliste permettant de gérer les incertitudes sur la détection des objets.

1.3.2 Approche proposée

Le cadre de notre étude est relativement différent des travaux précités dans le sens où notre problématique est de construire une carte d'encombrement de la scène. Il s'agit donc d'effectuer la recherche de tous les objets la constituant. D'autre part, aucune idée

de reconnaissance n'est sous-jacente à ce que nous entendons par "recherche". Toutefois, des travaux que nous citons se dégage une composante commune qu'il nous semble intéressant de conserver. Que ce soit par l'intermédiaire de relations géométriques, d'objets intermédiaires, de distributions de probabilité ou dans la définition de cadres géométriques spécifiques, toutes les recherches s'effectuent avec l'aide d'un *a priori* sur la localisation de l'objet. Nous pensons que sans ce type de connaissance *a priori*, des algorithmes de recherche exhaustive (parcours de toutes les parties de la scène) seraient, dans la plupart des cas, satisfaisants. D'autre part, les techniques basées sur une modélisation probabiliste de la scène considèrent des probabilités de présence évidemment mieux adaptées à la gestion de données incertaines comme peuvent l'être les mesures de détection (voir section 1.6) ou même les connaissances *a priori* qui sont rarement certaines.

Le modèle que nous utilisons s'insère dans un cadre probabiliste suffisamment général pour prendre en compte, en plus des observations successives, d'éventuels *a priori* structurels ou relationnels sur la distribution des objets dans la scène. Notre algorithme de recherche repose essentiellement sur trois fonctionnalités (voir figure 1.2) :

La modélisation de la connaissance que le système a de l'environnement dans lequel il effectue la recherche. Nous avons choisi d'exprimer cette connaissance sous forme de probabilités d'occupation, celles-ci étant définies sur des zones (ou cellules) issues d'un échantillonnage préalable de la scène. Cette approche peut être comparée à la notion de grille d'occupation introduite par Elfes [Elfes 96]. L'échantillonnage que nous proposons est construit sur la base des observations de l'image globale : les zones de recherche sont les troncs de cône engendrés par la projection des objets dans l'image globale. À l'instant initial, l'état de la connaissance se résume à un certain *a priori* sur le type de scène (encombrée ou non, dense ou éparse, ...). Mais, aux instants suivants et tout au long de la recherche, le modèle doit intégrer les observations successives.

Le modèle d'observation se charge de transformer les observations issues du traitement d'image en données homogènes et compatibles avec le modèle de connaissance ; en l'occurrence des contraintes sur les probabilités d'occupation des cellules observées.

Le choix du point de vue qui, en fonction du modèle de connaissance courant, optimise la recherche selon un certain critère.

Puisque nous nous intéressons plus particulièrement au problème de la recherche visuelle, nous devons identifier, dans un premier temps, le type d'observation que peut fournir une caméra en termes de détection d'objets. En fait nous considérons que le traitement d'image nous fournit une observation sous forme de probabilité d'occupation le long de chaque ligne de vue. Nous justifions ce choix en section 1.6 au vu des possibilités offertes actuellement par les algorithmes de traitement d'images.

La recherche s'effectue dans des zones appelées cellules issues d'un échantillonnage préalable de la scène. Nous définissons l'échantillonnage de la façon suivante : nous pouvons identifier, dans l'image globale, un certain nombre de lignes de vue dont la probabilité de détection est suffisamment grande (c'est-à-dire supérieure à un certain seuil). Ces lignes de vue sont des axes privilégiés pour la recherche dans le sens où elles contiennent presque sûrement au moins un objet. Chacun de ces axes peut alors être discrétisé en cellules. Explorer la scène revient à vérifier si chacune d'entre elles est occupée ou non. En ce qui concerne la modélisation de la zone de recherche, notre approche s'apparente à la recherche

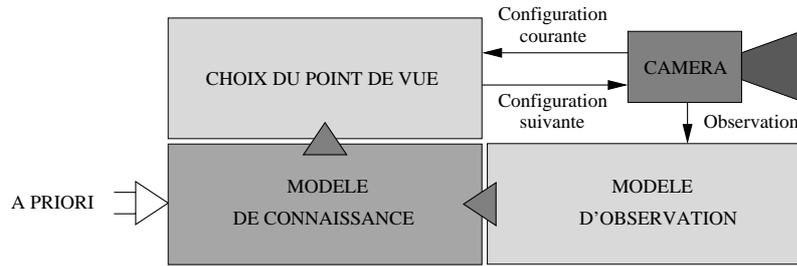


FIG. 1.2 – Décomposition fonctionnelle de l'algorithme de recherche en trois étapes : 1-modélisation de l'observation 2-modélisation de la connaissance et 3-choix du point de vue.

le long de rayons de [LópezOrtiz 98] mais adopte un point de vue radicalement différent visant la réduction de l'incertitude liée à la probabilité d'occupation de chaque cellule.

En effet, chaque cellule est dotée d'une probabilité d'occupation de sorte que l'exploration d'une cellule revient à faire tendre cette probabilité vers 0 si elle est vide ou bien vers 1 si elle contient un objet. Pour tenir compte d'éventuelles dépendances entre les cellules (ces dépendances peuvent se traduire en termes de proximité des objets ou du nombre d'objets présents), les probabilités de chaque cellule (ou lois marginales) prise individuellement ne suffisent pas. Pour cela il faut plutôt s'intéresser à la loi conjointe entre toutes les cellules. De cette loi conjointe il est alors facile d'extraire les différentes marginales.

Nous verrons alors que les *a priori* sur la scène ainsi que les observations se traduisent par des contraintes linéaires sur la loi conjointe. L'intégration de contraintes linéaires s'exprime comme la résolution d'un système sous-contraint ; résolution que nous construisons de manière itérative pour s'affranchir des problèmes d'inversion et surtout pour pouvoir intégrer les observations à mesure qu'elles se présentent. Les observations sont données par les images locales fournissant, pour chaque ligne de vue, une probabilité de détection. Dans un souci de clarté de la présentation, nous ne tenons compte que de la détection de long de l'axe optique de la caméra. Le capteur répond donc à la question : *avec quelle probabilité peut-on affirmer qu'un objet se trouve sur l'axe optique?*

Enfin nous nous intéressons à la construction d'une stratégie de recherche optimale définissant la configuration de la caméra à atteindre pour optimiser la recherche. Il s'agit donc de prédire le gain d'information apporté par de nouvelles observations. Un choix déterminant au sein de l'algorithme de recherche concerne la façon de quantifier le gain d'information. Cette quantification n'est pas triviale dans la mesure où elle doit valoriser les détections et les non détections. Nous proposons quatre quantifications possibles dont trois sont basées sur la norme de la variation des marginales (ou probabilités d'occupation) et diffèrent par le choix de la norme. La dernière quantification proposée porte sur la variation d'entropie de ces marginales.

1.4 Exploration des objets : un problème de reconstruction

L'étape que nous venons de décrire doit aboutir à l'identification de zones susceptibles de contenir un objet. Au niveau plus local, le système est amené à se focaliser sur chacune de ces zones afin de reconstruire chaque objet. Cette section est consacrée à la reconstruction automatique des objets depuis la modélisation jusqu'à la génération des commandes

d'exploration.

1.4.1 Reconstruction d'objets

1.4.1.1 *L'enveloppe visuelle*

La reconstruction d'objets à partir d'une image ou d'une séquence d'images joue un rôle important en vision par ordinateur. De nombreuses techniques ont vu le jour. Elles peuvent être répertoriées en fonction de l'information qui est utilisée dans l'image : le flot optique, l'ombre, des points en correspondance (reconstruction stéréoscopique), les contours. L'utilisation du contour de l'objet (appelé également silhouette) ne permet pas de reconstruire les parties creuses de celui-ci. En revanche les contours présentent l'intérêt d'être l'une des informations les plus simples à extraire dans l'image ; en tout cas plus simple et plus rapide que le flot optique ou qu'un ensemble de points mis en correspondance. Pour ces raisons, l'utilisation des contours a fait l'objet de nombreuses études.

Les études concernant la reconstruction des objets à partir des contours apparents se classent essentiellement en deux catégories. Les approches volumiques, initiées par Baumgart [Baumgart 74], utilisent le fait que le cône engendré par le contour contient l'objet, ainsi une succession de points de vue permet, par intersection, de calculer un volume contenant l'objet. Le concept d'enveloppe visuelle a été introduit par Laurentini [Laurentini 94]. L'enveloppe visuelle d'un objet se définit comme le plus petit volume pouvant être déterminé par cette approche. Des variantes de cette méthode utilisent des polyèdres ou des octrees pour représenter les cônes et leur intersection [Connolly 89, Sullivan 98, Srivastava 90]. Une autre classe d'approches, dites surfaciques, utilisent le fait que le cône visuel est tangent à l'objet observé pour reconstruire l'enveloppe des plans tangents à partir d'une séquence continue d'images [Vaillant 92, Cipolla 92, Boyer 97]. Zell et Garcia [Zell 94, Zell 95] présentent une méthode de construction d'octrees pour modéliser des objets 3D à partir de plusieurs vues. Ils proposent une méthode permettant de prendre en compte les erreurs dues au système d'acquisition par simple dilatation des silhouettes extraites de sorte que l'enveloppe estimée soit sur-englobante. Sullivan et Ponce [Sullivan 98] combinent les approches volumiques et surfaciques. Dans un premier temps, une approximation polyédrale de l'objet est générée par intersection des cônes visuels. Puis une surface spline dont les points de contrôle sont les sommets du polyèdre est calculée et déformée jusqu'à ce qu'elle soit tangente, en tout point, aux cônes visuels issus des différentes positions de la caméra.

Les méthodes visant la reconstruction de l'enveloppe visuelle sont très réalistes dans le sens où elles utilisent une information de l'image pertinente et relativement aisée à extraire : le contour des objets. Celles-ci souffrent néanmoins de leurs prétentions dans la mesure où elles cherchent à extraire l'enveloppe visuelle exacte et que les modèles correspondants sont souvent coûteux.

1.4.1.2 *Les modèles approchés*

La plupart des études que nous venons de citer se concentrent sur la modélisation 3D en termes de précision et d'exactitude de la reconstruction. Mais, pour une large classe d'applications telles que la planification de trajectoires, l'évitement d'obstacles ou l'exploration, une carte préliminaire et simplifiée des constituants de la scène et de l'encombrement est suffisante. Comme nous l'avons indiqué précédemment, une représentation concise et

donc simplifiée devient même nécessaire lorsque le système est soumis à des exigences de réactivité en temps réel. Ces constatations ont motivé nos travaux sur l'estimation ou la reconstruction de modèles approchés (appelés *coarse models* dans la littérature anglo-saxonne).

La notion de modèle approché n'est pas nouvelle et a fait l'objet de précédentes études par Agin et Binford [Agin 76], Marr et Nishihara [Marr 77] et plus récemment Ferrie [Ferrie 88]. Agin et Binford [Agin 76] sont, sans doute, les premiers à s'être intéressés à la représentation de surfaces 3D par des cylindres généralisés. Les cylindres généralisés sont complètement décrits par une courbe appelée axe du cylindre et une fonction décrivant la section perpendiculaire à l'axe (*cross section*) en tout point de celui-ci. Shani et Ballard [Shani 84] ont, à leur tour, contribué à la description des cylindres généralisés en utilisant des splines 3D et des B-splines.

Ferrie [Ferrie 88] examine le problème suivant : comment déduire une structure géométrique approchée d'un objet à partir d'une description de sa surface. Le modèle approché utilisé est basé sur des cylindres et des ellipsoïdes. L'article porte sur la décomposition de l'objet en parties : par l'étude de sa courbure, la surface est partitionnée en régions approximativement convexes. Il porte également sur l'estimation des paramètres décrivant chaque partie par des ellipsoïdes ou des cylindres et sur la suppression des modèles redondants de sorte à ne conserver que les modèles les plus pertinents.

Beaucoup d'auteurs se sont également intéressés à l'usage des superquadriques à des fins de modélisation approchée des objets [Whaite 94, Leonardis 94, Solina 90, Gupta 93]. Les superquadriques permettent de modéliser seulement un sous-ensemble des formes que peuvent décrire les cylindres généralisés. Toutefois elles constituent un compromis très intéressant entre simplicité de modélisation et complexité des formes qu'elles peuvent appréhender. La puissance de cette représentation réside dans la compacité de sa paramétrisation. Elle constitue une extension des ellipsoïdes au cas où les exposants sont rationnels. Une superquadrique centrée est définie par une formulation implicite du type :

$$\left(\left(\frac{x}{a_1} \right)^{\frac{2}{\epsilon_2}} + \left(\frac{y}{a_2} \right)^{\frac{2}{\epsilon_2}} \right)^{\frac{\epsilon_2}{\epsilon_1}} + \left(\frac{z}{a_3} \right)^{\frac{2}{\epsilon_1}} = 1$$

a_1 , a_2 et a_3 définissent la taille de la superquadrique alors que ϵ_1 et ϵ_2 déterminent sa forme. Une formulation explicite identique des superquadriques est donnée par :

$$x(\eta, \omega) = \begin{cases} a_1 \cos^{\epsilon_1}(\eta) \cos^{\epsilon_2}(\omega) \\ a_2 \cos^{\epsilon_1}(\eta) \sin^{\epsilon_2}(\omega) \\ a_3 \sin^{\epsilon_1}(\eta) \end{cases} \quad -\frac{\pi}{2} \leq \eta \leq \frac{\pi}{2} \quad -\pi \leq \omega \leq \pi$$

Une limite de cette description est qu'elle ne procure pas une paramétrisation régulière de la surface ce qui conduit à un comportement numérique peu stable aux asymptotes. En réponse à ce problème, Montiel [Montiel 98] modélise les superquadriques par déformation d'un ellipsoïde ce qui a pour effet de régulariser la distribution des paramètres et de simplifier leur estimation.

[Solina 90] développe une technique de détermination des paramètres d'une superquadrique décrivant au mieux un ensemble de points 3D. Les paramètres optimaux correspondent au minimum d'une fonction d'énergie calculée en fonction de la distance des

points au modèle de surface. L'utilisation d'une technique stochastique d'optimisation permet de s'affranchir des minima locaux de cette fonctionnelle. D'autre part, une restriction de l'espace des paramètres permet de choisir parmi l'ensemble des minima globaux correspondants aux solutions acceptables. En effet il peut exister plusieurs paramétrisations possibles de superquadriques pour représenter une même forme.

[Gupta 93] propose une procédure de segmentation récursive d'une carte de points 3D utilisant des modèles superquadriques pour représenter les régions segmentées. Celle-ci est capable de s'adapter à la complexité des données et permet de choisir le degré de finesse de la reconstruction. Dans [Leonardis 94], les travaux sont également orientés vers la décomposition en parties. La méthode proposée consiste en deux étapes : la recherche de modèles superquadriques, éventuellement redondants, par une combinaison de classification des données et d'estimation de paramètres puis la sélection des modèles pertinents par élimination des modèles redondants.

1.4.1.3 Approche proposée

La majorité des travaux que nous venons de citer sont en général peu adaptés à la reconstruction visuelle car ils nécessitent une représentation 3D de la surface ou des points la constituant. En revanche, les méthodes concernant la reconstruction des enveloppes visuelles exactes nous semblent coûteuses en termes de représentation et surtout mal adaptées à la gestion des incertitudes. Notre approche se situe à l'intersection de ces deux classes de méthodes et concerne la reconstruction des enveloppes visuelles approchées.

Comme nous l'avons dit précédemment, pour les techniques basées sur la reconstruction de l'enveloppe visuelle, le traitement d'image est réduit à l'extraction des contours (ou d'une approximation des contours dans notre cas) du masque des objets dans l'image. Par conséquent notre étude peut, en théorie, s'appliquer à n'importe quel type d'objet dans la mesure où il peut être segmenté et mis en correspondance tout au long de la séquence d'acquisitions. En pratique, pour n'imposer aucune contrainte concernant l'aspect des objets, nous faisons l'hypothèse que la frontière des objets est marquée par une discontinuité de profondeur. Ainsi un algorithme de détection du mouvement nous donnera le masque des objets (section 1.6).

Puisque nous décidons de baser notre étude sur la reconstruction d'un volume englobant approché de l'objet, se pose, à présent, la question du modèle que nous allons utiliser pour sa représentation.

1.4.2 Modèles mixtes

Un point rarement abordé en modélisation des objets concerne l'unification des aspects géométriques et incertitudes de reconstruction. L'aspect évident de la modélisation d'un objet, se situe au niveau géométrique : il est entièrement représenté par sa position et sa forme. Mais pouvoir extraire une position et une forme d'un modèle dont on souhaite qu'il prenne en compte les incertitudes, n'est pas suffisant. En effet, en présence d'incertitudes, comment affirmer qu'un point de la scène est occupé par un objet ou non ? Il apparaît plus raisonnable d'attribuer à ce point une probabilité d'appartenance à un objet. La construction du modèle d'objet que nous présentons vise à unifier ces deux aspects.

Les méthodes que nous développons s'insèrent dans le cadre général des techniques d'estimation. Celles-ci supposent généralement une description probabiliste des observa-

tions. La géométrie probabiliste permet alors de rendre les observations compatibles et comparables afin de les fusionner.

1.4.2.1 Géométrie probabiliste et transformation des lois

La géométrie probabiliste [Ayache 91, DurrantWhyte 87, DurrantWhyte 88a, Collins 93] permet de résoudre le paradoxe lié à l'unification des aspects géométriques (information structurelle de haut niveau) et des aspects probabilistes plus faciles à modéliser dans le cadre de représentations locales (information d'encombrement). Le compromis réside dans la nécessité d'une représentation qui puisse prendre en compte les interactions géométriques de haut niveau entre le robot et son environnement tout en offrant un formalisme adapté aux prises de décision sachant que les mesures et les actions ont un degré d'incertitude intrinsèque non négligeable. La clef d'une représentation efficace d'un environnement géométrique incertain est la possibilité de fusionner des données de natures différentes. La géométrie probabiliste diffère de la théorie classique des probabilités en cela qu'elle requière une interprétation physique des variables aléatoires et consiste, en quelque sorte, en la transformation et la manipulation de ces variables. Ces manipulations peuvent concerner le changement de repère ou de paramétrisation, la fusion de données, l'agrégation de caractéristiques dans une description de plus haut niveau, ...

Les variables géométriques aléatoires manipulées sont en général les paramètres d'une caractéristique observée. L'incertitude d'un objet est alors représentée par une densité de probabilité sur l'espace des paramètres de l'objet. En effet tout objet géométrique peut être décrit par une fonction vectorielle g et un vecteur de paramètres p tels que pour tout point x appartenant à l'objet,

$$g(x,p) = 0$$

g détermine le type de primitive géométrique observée (point, droite, cylindre,...) et la valeur de p définit ses paramètres. L'incertitude d'observer une valeur particulière de p peut alors être représentée par une densité de probabilité $f(p)$.

Toute transformation de l'objet géométrique peut alors s'écrire $p' = T(p)$ où p' est un nouveau vecteur aléatoire. Si T est bijective et continûment différentiable, f' s'obtient par application du changement de variable dans les intégrales :

$$f'(p') = \frac{1}{|\det(\frac{\partial T}{\partial p})|} f(T^{-1}(p'))$$

Étudier la transformation de vecteurs de lois quelconques devient vite très difficile puisqu'une transformation quelconque n'a aucune raison de conserver le type de loi et les modèles de lois peuvent devenir très complexes au bout d'une ou deux transformations seulement. Pour pallier cette complexification, l'idée est de transformer les moments successifs de la fonction de probabilité jusqu'à l'ordre de précision souhaité. Ainsi pour les moments d'ordre 0 et 1, les transformations sont :

– moyenne

$$\hat{p}' = T(\hat{p})$$

– matrice de covariance

$$\Lambda' = J\Lambda J^T$$

Si, de plus, on fonctionne sous l'hypothèse de lois gaussiennes alors la description nécessaire et suffisante d'un vecteur aléatoire se limite à son espérance et à sa matrice de covariance. Enfin, lorsque p est gaussien mais T non linéaire, $T(p)$ n'a aucune raison d'être lui-même gaussien. L'idée est alors d'approximer linéairement T autour de la valeur moyenne de p .

Les travaux menés en robotique et en vision dans ce domaine [Ayache 91, Durrant-Whyte 87] concernent principalement l'intégration d'informations multi-sensorielles. La géométrie probabiliste permet de résoudre le problème posé par la combinaison de mesures prises à différents instants, en différents lieux et avec différents capteurs. Les applications se situent principalement au niveau de la localisation des robots, la mise à jour de cartes de l'environnement représenté par un ensemble de primitives géométriques et la reconnaissance d'objets.

1.4.2.2 Modèles à erreur bornée

Lorsque l'incertitude n'est pas intrinsèquement stochastique ou que le modèle probabiliste n'est pas connu, une alternative intéressante est de considérer un modèle à erreur bornée. Ceux-ci ont été introduits par Witsenhausen et Schweppe [Witsenhausen 68, Schweppe 68]. L'idée est de modéliser l'erreur par un bruit additif uniforme dont les bornes du support sont connues. De nombreux auteurs ont depuis repris ces travaux et largement contribué au domaine. Citons Chernousko [Chernousko 82] dont les travaux concernent l'étude de la contrôlabilité des systèmes pour lesquels il définit des domaines d'atteignabilité. Milanese, Norton, Walter et Pronzato se sont plus particulièrement intéressés à l'identification et à l'estimation de paramètres. Les contributions et les connaissances actuelles dans ce domaine sont décrites dans [Milanese 91b, Milanese 96, Norton 87]. Kieffer, Jaulin, Walter et Meizel [Kieffer 00] présentent une solution au problème de localisation et de suivi dans un contexte où les erreurs de mesure et d'incertitude sur l'état sont bornées. À chaque itération, le résultat de la localisation est un ensemble contenant la configuration du robot.

Dans le cadre des modèles à erreur bornée, la connaissance et les observations sont modélisées par le support d'une loi uniforme. Se pose alors le problème de décrire ces supports. D'autre part, fusionner les observations nécessite le calcul d'intersections de supports parfois complexes. Une méthode efficace d'un point de vue numérique consiste à approcher tous les supports par un volume englobant simplifié et appartenant à une classe donnée. Les types d'approximations étudiées jusqu'à ce jour sont principalement les polytopes, les orthotopes et les ellipsoïdes.

Lorsque le système et les contraintes d'observation sont linéaires, les supports sont décrits, de façon exacte par des polytopes [Broman 86]. Le principal inconvénient de cette description est que le nombre de faces et d'arêtes du polytope augmente rapidement avec le nombre d'observations. Pour pallier cette augmentation de la taille du modèle, une solution est d'approcher le polytope par une forme plus simple et de paramétrisation invariante vis-à-vis de l'intégration de nouvelles observations.

Les orthotopes sont des hyper-rectangles dont les arêtes sont parallèles aux axes du repère dans lequel est décrit le support. Plusieurs algorithmes ont été proposés dans le but d'estimer ces hyper-rectangles. Milanese et Vicino [Milanese 91a] ont proposé un algorithme réutilisant toutes les contraintes déjà introduites aux itérations précédentes. Il devient donc rapidement coûteux en temps de calculs. Messaoud et Favier [Messaoud 94]

proposent un algorithme récursif permettant de contourner ce problème.

Pour ce qui concerne l'approximation ellipsoïdale, citons les travaux de Evans [Saleem 96, Evans 94], Maksarov et Norton [Maksarov 96], Walter, Pronzato et Durieu [Pronzato 94a, Pronzato 94b, Durieu 96]. Maksarov [Maksarov 96] présente un état de l'art détaillé et des comparaisons avec des approches statistiques (telles que le filtre de Kalman) en ce qui concerne des problèmes d'identification. La plupart des auteurs s'intéressent au développement d'algorithmes de sommation et d'intersection d'ellipsoïdes. Or l'intersection de deux ellipsoïdes n'est pas un ellipsoïde, il faut donc en faire une approximation. Classiquement l'intersection est approchée par le plus petit ellipsoïde englobant. Apparaît donc la notion de minimisation d'un ellipsoïde [Pronzato 94a, Saleem 96] qui est souvent traduite en termes de trace ou de déterminant de la matrice symétrique correspondante [Durieu 96].

1.4.2.3 Approche proposée

Notre étude tire parti des modèles probabilistes et à erreur bornée au sein d'une représentation mixte des objets (voir [Hanebeck 96] pour d'autres travaux concernant la combinaison de théories statistiques et ensemblistes) pour laquelle la position de l'objet et l'incertitude associée sont les paramètres d'une loi normale et l'enveloppe de l'objet est représentée par un modèle à erreur bornée dont le support est une approximation de son volume englobant. Comme nous l'avons vu précédemment, la méthode classiquement utilisée en géométrie probabiliste pour modéliser une enveloppe dans un cadre incertain est de décrire cette enveloppe sous une forme paramétrique puis d'ajouter un bruit sur les paramètres. L'approche que nous proposons diffère de ce type de méthode en cela qu'elle décorrèle la représentation de l'incertitude de celle de l'enveloppe.

En ce qui concerne la description de l'enveloppe, notre choix se porte sur l'approximation ellipsoïdale pour son analogie forte avec les matrices de covariances ce qui permet de définir des règles de propagation semblables et donc relativement compactes dans leur formulation. En effet, les ellipsoïdes se définissent entièrement par une forme quadratique donc un vecteur x_0 représentant la position du centre et une matrice symétrique E telle que pour tout point x :

$$\begin{cases} (x - x_0)^T E (x - x_0) > 1 & \text{si } x \text{ est extérieur à l'ellipsoïde} \\ (x - x_0)^T E (x - x_0) = 1 & \text{si } x \text{ appartient à la frontière de l'ellipsoïde} \\ (x - x_0)^T E (x - x_0) < 1 & \text{si } x \text{ est intérieur à l'ellipsoïde} \end{cases}$$

ils présentent donc une forte analogie avec les distributions gaussiennes. Le centre de l'ellipsoïde devient la moyenne de la distribution et la matrice symétrique de forme quadratique est, à une inversion près, la matrice de covariance. Un autre intérêt des ellipsoïdes réside dans la simplicité d'appréhender l'orientation des objets qu'ils modélisent, l'orientation étant déterminée par les vecteurs propres de la forme quadratique associée.

Le modèle ainsi défini est d'autant plus intéressant qu'il permet à la fois, pour chaque point de la scène, le calcul de sa probabilité d'appartenance à un objet et une extraction simple de sa géométrie (position et enveloppe). Il permet donc l'unification de l'incertitude à des aspects géométriques. Le modèle est établi sous une forme paramétrique appelée *loi d'ensemble* dans laquelle chaque paramètre correspond à une interprétation physique soit en terme de position, d'incertitude ou encore d'enveloppe.

Une fois le modèle établi, il s'agit de définir des règles de propagation permettant d'obtenir les paramètres du modèle après transformation. Nous développons des règles très générales concernant les difféomorphismes et les projections. Ces règles permettent de modéliser plus particulièrement la chaîne des observations visuelles pour différentes configurations : monoculaire ou binoculaire. Ces règles sont établies sur la base d'une linéarisation des transformations. Cette approximation au premier ordre n'est pas gênante dans le cadre d'une reconstruction approchée comme celle que nous visons. Elle apparaît, qui plus est, nécessaire pour assurer la consistance du modèle. En effet, sans cette linéarisation, la transformée d'une loi d'ensemble ne serait pas une loi d'ensemble au quel cas, en une ou deux itérations, le modèle deviendrait inutilisable en perdant la concision de sa paramétrisation.

Nous dérivons également un schéma d'estimation de la loi d'ensemble. Étant donnée la connaissance à l'instant k de la loi *a priori* et étant donnée la loi représentant l'observation à l'instant $k + 1$, il s'agit de calculer une estimation de la loi *a posteriori*. Nous avons choisi d'estimer le paramètre de position et l'incertitude associée au sens du maximum *a posteriori*. Le paramètre définissant l'enveloppe est, quant à lui, une approximation ellipsoïdale de l'intersection de deux ellipsoïdes. Ainsi, sous réserve d'avoir pu initialiser le modèle au début du processus et de disposer d'une séquence d'observations, nous sommes capables d'intégrer de nouvelles observations et notamment des informations visuelles.

Décrire le volume englobant d'un objet par un ellipsoïde peut paraître, dans certains cas, très approximatif. Mais nous rappelons que lorsque la description d'un objet physique fortement concave par un ellipsoïde apparaît incohérent vis-à-vis de la tâche assignée, cet objet devra être partitionné en parties cohérentes. Ceci dépasse, néanmoins, le cadre de notre étude.

1.4.3 Exploration pour la reconstruction des objets

1.4.3.1 Contexte

Un autre problème concernant la reconstruction d'un objet est la génération autonome des mouvements exploratoires permettant une estimation optimale de ce modèle. De façon générale, ce problème est appelé dans la littérature anglo-saxonne "optimal sensor planning" [Tarabani 95]. Lorsque le but est la reconstruction, fine ou approchée de scènes ou d'objets, nous parlons d'exploration. Celle-ci est dite autonome lorsque la scène est initialement totalement ou partiellement inconnue.

Kutulakos, Dyer et Lumelsky exploitent les variations de la frontière d'occultation qui regroupe l'ensemble des points séparant la partie visible de la partie occultée de l'objet à des fins de reconstruction [Kutulakos 92a, Kutulakos 93] et d'exploration [Kutulakos 94, Kutulakos 92b]. En ce qui concerne les travaux sur l'exploration, leur approche consiste à contrôler l'apparition de zones non encore visibles en analysant le comportement de la frontière d'occultation. Celle-ci évolue de façon continue en fonction de la position du robot excepté en des points de vue spécifiques pour lesquels la topologie de la courbe change. L'exploration est formulée comme une tâche permettant de faire "glisser" la frontière d'occultation sur des points de la surface de l'objet non visibles initialement. La principale difficulté de cette approche est que la forme de la frontière d'occultation ne dépend pas seulement de la position du robot mais également de la forme de l'objet qui est inconnue. Le contrôle de son évolution est donc incomplet. L'algorithme proposé garantit

néanmoins qu'un nombre fini de points de vue permet l'exploration complète de l'objet. Cette étude concerne aussi bien les capteurs visuels que les capteurs laser. En revanche la méthode ne permet pas de gérer l'incertitude associée à la mesure.

Whaite et Ferrie se sont intéressés à l'estimation de modèles superquadriques [Ferrie 92] à partir de données 3D (plus appropriées aux capteurs lasers). Ce type de modèle présente l'avantage de décrire un grand nombre de formes avec un nombre réduit de paramètres (seulement 2 de plus qu'il n'en faut pour décrire un ellipsoïde). Si a est le vecteur de paramètres à estimer, l'algorithme consiste à minimiser, par une méthode type Levenburg-Marquardt, la quantité

$$\chi^2(a) = \sum \frac{D^2(x_i, a)}{\sigma_i^2}$$

où $D(x_i, a)$ mesure la distance entre le point observé x_i et le modèle. La somme est pondérée par l'incertitude associée à chaque observation. Le principe de l'algorithme est simple mais présente une ambiguïté vis-à-vis de la solution obtenue. En effet, la surface $\chi^2(a)$ ne présente pas, en général, un minimum global unique mais une vallée correspondant à une infinité de solutions. L'idée astucieuse proposée par les auteurs [Whaite 91, Whaite 92, Whaite 94] est de tirer partie de cette ambiguïté à des fins exploratoires. La vallée correspondant à cette ambiguïté est modélisée par un ellipsoïde et traduite dans l'espace 3D de sorte à en déduire un schéma de minimisation de l'ambiguïté par des choix de points de vue adéquats.

Durrant-Whyte [DurrantWhyte 88b] propose, comme application directe de la géométrie probabiliste en robotique, le développement de stratégies minimisant l'incertitude associée à l'observation. Du fait de contraintes topologiques sur la position de la caméra par rapport à la primitive observée, l'incertitude d'observation dépend de la position de la caméra. D'autre part, il modélise l'erreur d'estimation comme étant inversement proportionnelle à la distance de la caméra à l'objet. Le critère choisi pour déplacer la caméra est la trace de la matrice d'information associée à l'observation qu'il minimise en utilisant les multiplicateurs de Lagrange. À notre connaissance, cette étude n'a malheureusement pas fait l'objet d'expérimentations.

Maver et Bajcsy [Maver 93] ont proposé une stratégie d'acquisition de données 3D obtenues par triangulation au sein d'un couple laser/caméra. La stratégie est basée sur la découverte de zones occultées par d'autres parties de la scène. Celles-ci sont classées en deux catégories : les zones non visibles par la caméra et les zones non atteintes par le faisceau laser. Après acquisition d'une image, les données manquantes dues au premier type d'occultation sont acquises en effectuant des mouvements de rotations adéquats. Ensuite, les zones correspondant au deuxième type d'occultation sont localisées afin de déterminer la direction du prochain point de vue.

Mentionnons également les récents travaux de Arbel et Ferrie [Arbel 99] concernant la sélection de points de vue. Même s'ils traitent du problème, relativement différent, de la reconnaissance d'objets, l'approche est intéressante. Elle est basée sur le calcul hors ligne d'une carte d'entropie associée à l'ambiguïté de reconnaissance. Caglioti [Caglioti 01a, Caglioti 01b] propose également un critère basé sur l'entropie. Il utilise, plus précisément, l'espérance de la variation d'entropie associée à l'observation à des fins de reconnaissance et de localisation d'objets. La reconnaissance est caractérisée par un processus discret de classification de l'objet. La localisation est quant à elle modélisée par une paramétrisation continue. Un critère classique en estimation de paramètres discrets est

l'entropie du système. En continu, il est plus naturel d'utiliser une fonction de la variance de la distribution des paramètres. Dans [Caglioti 01a] Caglioti propose une solution au problème hybride combinant paramétrisations discrètes et continues.

1.4.3.2 Approche proposée

Pour résoudre l'ambiguïté de positionnement d'un objet causé par son auto-occultation, un système visuel de reconstruction quel qu'il soit doit acquérir des données à partir de différents points de vue. L'exploration autonome d'un objet peut être vue comme un moyen de déterminer automatiquement les mouvements du système permettant de lever cette ambiguïté. Dans le cas d'un objet, le terme d'exploration peut être comparé à celui de vision active pour la reconstruction.

Dans notre cas où la position est modélisée par une distribution gaussienne et où la forme est représentée par un ellipsoïde, le concept d'exploration doit être vu comme un moyen d'améliorer la localisation de l'objet. L'exploration est dite optimale si le taux de convergence de l'estimation est le plus rapide possible. La stratégie que nous développons consiste à réduire l'incertitude de la distribution associée à l'objet observé en intégrant des données visuelles. La modélisation gaussienne de l'incertitude ainsi qu'une linéarisation adéquate du processus d'acquisition visuel nous permettent de construire une solution analytique au problème d'exploration optimale.

Dans la continuité de Durrant-Whyte [DurrantWhyte 88b] qui propose d'utiliser comme critère la trace de la matrice d'information liée à l'observation, nous choisissons un critère lié à l'information prédite *a posteriori* c'est-à-dire une prédiction sur la connaissance du modèle après fusion d'une nouvelle observation elle-même prédite. Ce critère présente l'avantage d'optimiser la reconstruction en termes de diminution de l'incertitude et de maximiser, indirectement, le volume découvert puisque l'incertitude est liée à l'inobservabilité, c'est-à-dire que les directions les plus incertaines sont *a priori* celles qui ont été les moins observées. L'optimisation de ce critère nous permet de décrire deux types de lois de commande exploratoires : une commande localement optimale et une commande basée sur le meilleur point de vue.

1.5 Asservissement visuel

L'opération consistant à transformer des consignes en commandes, directement utilisables par le système est appelée asservissement. L'asservissement est dit visuel lorsqu'il utilise les données d'une ou plusieurs caméras. Il existe plusieurs architectures d'asservissement visuel. Les deux types principaux se distinguent par le domaine de définition de la consigne :

- soit la consigne est définie dans l'espace de la tâche (en général l'espace cartésien) ; dans ce cas le système de vision doit permettre de retrouver des informations 3D afin de s'asservir sur ces données ;
- soit la consigne est définie dans l'image ; c'est à dire que les déplacements sont générés directement par comparaison de certaines caractéristiques de l'image courante avec la valeur de ces caractéristiques à l'instant final.

Hutchinson [Hutchinson 96] fournit une comparaison détaillée de ces deux méthodes. L'asservissement pour lequel les consignes sont des caractéristiques de l'image est dit "référéncé

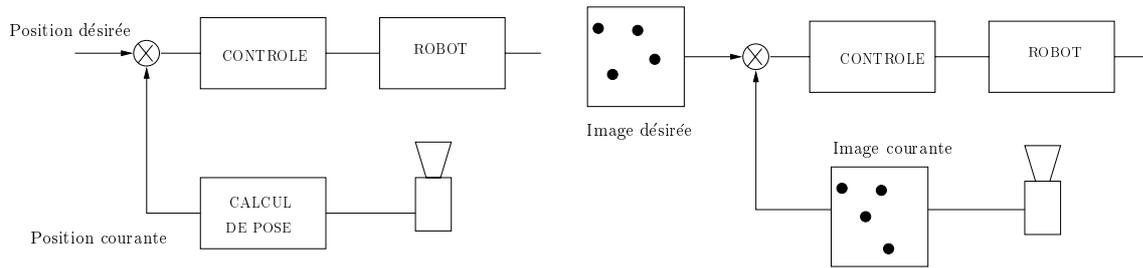


FIG. 1.3 – Asservissement 3D / Asservissement 2D

capteur” ou “2D”. Il présente l’avantage de s’affranchir des connaissances 3D qui, par nature, dépendent fortement de la calibration de la caméra et qui, surtout, nécessitent des connaissances supplémentaires (vitesse de déplacement, informations issues de deux capteurs, structure de l’objet observé, ...) puisqu’une image ne contient pas l’information de profondeur.

Pour toutes ces raisons, depuis quelques années, beaucoup d’auteurs se sont consacrés à l’asservissement visuel 2D embarqué : Chaumette [Chaumette 98a, Chaumette 90, Chaumette 91, Chaumette 93, Chaumette 98b], Espiau [Espiau 92], Weiss [Weiss 84], Hashimoto [Hashimoto 93]. Au cœur de cette approche, réside le jacobien image (ou matrice d’interaction). Il relie les variations de certaines caractéristiques de l’image à la commande du robot. Plusieurs types de caractéristiques de l’image peuvent être choisis. Citons parmi d’autres :

- les paramètres de primitives bien choisies (points, droites, cercles, sphères, ...) [Chaumette 93]
- le mouvement dans l’image [Crétual 98]
- la signature polaire ou les coefficients de Fourier de contours fermés [Collewet 98, Collewet 99].

L’idéal étant de trouver une formulation analytique du jacobien suivant le type de primitive choisi, lorsque ce n’est pas possible, certains se sont penchés sur le problème de la réactualisation et de l’estimation numérique du jacobien [Hosoda 94, Hosoda 95, Jägersand 94, Jägersand 97]. Citons enfin des travaux qui s’intéressent à la sensibilité et au conditionnement du jacobien [Hashimoto 93, Hashimoto 98].

Souvent, la loi de commande qui en découle peut être exprimée comme régulant une fonction de tâche [Samson 91]. Tout le problème est alors de déterminer les conditions et donc le domaine de stabilité de la loi de commande. Malis [Malis 98c, Malis 98b, Malis 98a, Malis 99] étudie la stabilité d’un asservissement visuel hybride tirant partie des asservissements 2D et 3D appelé asservissement visuel 2D 1/2. L’étude de la stabilité est, en général, dépendante de la calibration c’est pourquoi certains ont cherché des asservissements exempts de calibration [Yoshimi 94]. Si l’asservissement visuel est souvent basé sur l’approche fonction de tâche, d’autres [Papanikolopoulos 93] ont proposé une formulation différente de l’asservissement visuel rendant possible un contrôle proportionnel intégral, par placement de pôles ou encore LQG.

En ce qui concerne l’asservissement visuel embarqué, citons enfin les travaux de Pissard-Gibollet et Rives [PissardGibollet 91, PissardGibollet 93] qui se sont intéressés à l’asservissement visuel dans le cadre de la robotique mobile. Marchand [Marchand 98] se sert de

l’asservissement visuel pour l’évitement d’occultations. Mezouar [Mezouar 00, Mezouar 01b, Mezouar 01a] planifie, dans l’image la trajectoire des consignes ce qui permet de conserver les points d’intérêt dans l’image, d’éviter les auto-occultations ou les butées articulaires.

Dans [Horaud 98, Dornaika 92], les résultats de [Espiau 92] sont étendus au cas où la caméra est déportée. Dans ce cas de figure, le schéma de commande diffère du cas embarqué par le simple fait que le jacobien image doit prendre en compte le passage du repère caméra au repère de commande du robot. En matière d’asservissement visuel déporté citons également les travaux de Reyes et Kelly [Reyes 98] et Wijesoma [Wijesoma 93]. Plus récemment, Ruf [Ruf 00] utilise le cadre de la géométrie projective pour effectuer l’asservissement.

Beaucoup d’études ont abordé l’utilisation de plusieurs caméras en vision par ordinateur : en général à des fins de reconstruction 3D par stéréovision, parfois pour de l’asservissement visuel basé sur une paire d’images [Hager 97, Horaud 98] mais rarement dans le cadre d’une coopération caméra globale / caméra locale [Marchand 98]. Le travail que nous décrivons en annexe A constitue une première étape vers une telle coopération. Nous définissons et montrons la faisabilité d’une tâche d’asservissement visuel fusionnant une vue locale et une vue globale de la scène. Dans un souci de robustesse, nous démontrons la stabilité de la loi de commande et comparons différentes techniques de coopération d’un point de vue expérimental.

1.6 Détection d’objets au sens du mouvement

Comme toujours en vision par ordinateur, nos travaux reposent sur des algorithmes “bas niveaux” liés au traitement d’images. Nous sommes plus particulièrement concernés par la détection des objets dans une séquence d’images. Cette section décrit très brièvement les algorithmes de détection dont l’approche probabiliste semble plus adaptée à notre problème.

Un certain nombre de méthodes visant à détecter des objets dans une scène considèrent des images fixes et sont basées sur des critères photométriques tels que la couleur [Pateux 00, Leclerc 89], la texture [Paragios 00, Kervrann 95], les contours [Abe 00]. De ce fait elles dépendent fortement du type de scène considéré et sont souvent spécialisées à une classe d’objets particuliers [Papageorgiou 99, Meng 00]. Pour s’affranchir d’une quelconque restriction sur l’apparence des objets, les méthodes basées sur l’analyse de séquences d’images et notamment sur des critères de mouvement nous paraissent plus adaptées. En effet, si on fait l’hypothèse, peu restrictive, que les objets présentent des discontinuités de profondeur le long de leur frontière, ceux-ci auront, pour des mouvements de translation de la caméra, un mouvement apparent distinct et donc identifiable par des algorithmes de détection au sens du mouvement. C’est pourquoi nous consacrons cette section à la détection du mouvement dans une séquence d’images [Mitiche 96, Odoñez 94, Fablet 99, Irani 92, Torr 93, Thompson 93, François 91].

Les techniques proposées se classent en deux catégories, celles dont les séquences analysées correspondent à une caméra fixe et celles pour lesquelles la caméra est en mouvement. Dans notre situation, puisque les objets sont fixes, la caméra doit obligatoirement être en mouvement afin que les objets correspondent à des zones en mouvement dans l’image. Mais cette situation peut se ramener au cas d’une caméra fixe par compensation du mou-

vement dominant dans l'image [Stiller 99, Ayer 96, Odobez 95]. La détection des objets est alors basée sur l'hypothèse que le mouvement dominant est induit par le mouvement de la caméra.

Parmi les techniques de détection au sens du mouvement, nous examinons plus particulièrement celles, de plus en plus nombreuses, offrant un cadre statistique prometteur puisqu'elles fournissent, plus qu'une carte binaire (détection ou non détection), la probabilité de l'hypothèse avancée.

L'utilisation des tests d'hypothèses semble bien adaptée à la détection des changements temporels [Torr 93, Mitiche 96]. Si l'on modélise l'intensité dans une fenêtre autour d'un pixel (ou site) donné, par une certaine distribution de probabilité, et que l'on note H_0 : l'hypothèse qu'il n'y a pas de changement temporel entre les instants t et $t + dt$ (c'est-à-dire que les paramètres du modèle sont identiques dans les images t et $t + dt$) et H_1 : l'hypothèse qu'il y a changement temporel entre les instants t et $t + dt$ (c'est-à-dire que les paramètres du modèle sont différents dans les images t et $t + dt$) alors le choix de l'une ou l'autre des hypothèses en fonction d'une observation notée z peut se faire de la façon suivante :

$$\begin{cases} H_0 & \text{si } \ln \frac{\widehat{L}_1}{\widehat{L}_0} < \lambda \\ H_1 & \text{sinon} \end{cases}$$

où $\widehat{L}_0 = \max p(z|H_0)$ et $\widehat{L}_1 = \max p(z|H_1)$ sont les maximums de vraisemblance obtenus en faisant varier les paramètres du modèle d'intensité et λ est un seuil de comparaison. Si le choix retenu est H_1 , c'est-à-dire qu'un objet est détecté, nous pouvons lui associer la probabilité d'avoir raison :

$$p(H_1|H_1) = 1 - p(H_0|H_1)$$

où $p(H_0|H_1)$ est connu sous le nom de probabilité de non détection.

D'autres méthodes [Odobez 95, François 91, Xiong 94] adoptent également un cadre statistique en se basant sur une modélisation par champs de Markov [Pérez 98, Geman 84]. L'idée est de modéliser la distribution jointe des étiquettes et des observations sous la forme d'une distribution de Gibbs

$$p(e,o) = \frac{1}{Z} \exp - U(e,o)$$

où e à valeur dans $\{0,1\}$ est l'étiquette déterminant s'il y a mouvement ou non, o est une mesure de l'activité de mouvement définissant l'observation, $U(e,o)$ est une fonction d'énergie et Z une constante de normalisation. L'estimation \widehat{e} de e au sens du critère MAP (celle qui maximise la distribution *a posteriori* de l'étiquetage $p(e|o)$) est donc celle qui minimise $U(e,o)$:

$$\widehat{e} = \arg \min_e U(e,o)$$

Le calcul, pour chaque zone, de la probabilité de l'étiquetage \widehat{e} est donc possible (même s'il est souvent coûteux en temps).

Pour les deux types de détection d'objet que nous venons de présenter, nous mettons en avant la possibilité d'associer à un étiquetage binaire (détection ou non détection) une probabilité sur cette étiquetage. Il nous semble particulièrement intéressant de pouvoir tenir compte de cette probabilité d'autant que les méthodes classiques utilisant un seuillage binaire [Irani 92, Thompson 93] s'intègrent également dans un cadre probabiliste plus général pour lequel les modèles de vraisemblance sont uniformément distribués.

En résumé des techniques que nous venons de décrire, nous considérons que le traitement d'images nous fournit une observation soit sous forme d'un masque binaire décrivant la projection de l'objet dans l'image, soit d'une probabilité d'occupation le long de chaque ligne de vue. Ces deux fonctionnalités sont tout à fait adaptées aux exigences des algorithmes que nous avons développés.

1.7 Plan du rapport

La partie I est consacrée à la recherche des objets dans la scène. Plus précisément :

- au chapitre 1, nous établissons le modèle de connaissance basé sur un échantillonnage de la scène dont chaque cellule est décrite par une probabilité d'occupation. La scène est alors modélisée par un ensemble de lois conjointes permettant de définir des dépendances entre les probabilités d'occupation. Nous verrons que la mise à jour de ce modèle en fonction des observations ou, éventuellement, d'*a priori* nécessite la mise en place d'un processus d'intégration des contraintes linéaires. Ce processus et les outils s'y rapportant sont également décrits au chapitre 1.
- le chapitre 2 concerne l'application des outils développés au chapitre 1 au modèle que nous avons spécifiquement introduit pour la recherche des objets. Dans un premier temps nous détaillons les algorithmes permettant d'intégrer des contraintes sur une loi conjointe et de traduire des observations en contraintes. Dans un deuxième temps, nous décrivons le principe du calcul des configurations de la caméra permettant d'optimiser la recherche.
- le chapitre 3 regroupe des résultats de simulation permettant de valider le principe des algorithmes d'intégration des observations et d'étudier en détail les performances des stratégies de recherche proposées.

La partie II est consacrée à l'exploration des objets. Elle est organisée de la façon suivante :

- au chapitre 1, nous décrivons précisément le modèle utilisé pour représenter les objets. Ce modèle est constitué d'un mélange de modèle stochastique et de modèle à erreur bornée. Ce modèle est vu comme une densité de probabilité appelée distribution d'ensemble. Les paramètres de cette distribution ont une interprétation géométrique simple. Ils correspondent à la position de l'objet, l'incertitude associée et une approximation ellipsoïdale de son enveloppe.
- au chapitre 2 nous développons, dans un premier temps, des règles de transformation générales des lois d'ensembles. Ces règles concernent les difféomorphismes, les difféomorphismes en présence d'incertitude et la projection sur un sous-espace. Dans un deuxième temps, nous appliquons ces règles à la propagation de données visuelles. Nous développons des règles de rétro-projection, de projection perspective et de changement de repère. Ces règles doivent nous permettre d'exprimer n'importe quelle observation décrite par une loi d'ensemble sous une forme homogène et comparable avec d'autres observations ou avec la distribution de l'objet.
- au chapitre 3, nous décrivons un processus d'estimation de la loi d'ensemble représentant un objet. Il est basé sur une estimation au sens du maximum *a posteriori* de la position de l'objet et de l'incertitude associée et d'un calcul d'intersection ellipsoïdale pour l'enveloppe. Ainsi, à partir du mouvement de la caméra nous pouvons

reconstruire de façon approchée les objets. Nous présentons également des résultats de simulations pour lesquelles nous reconstruisons le modèle approché d'un objet de synthèse en appliquant une trajectoire circulaire à la caméra. Ces simulations nous permettent de valider le principe de l'estimation et d'étudier la stabilité de la reconstruction.

- au chapitre 4, dans le contexte de l'exploration autonome, nous définissons un critère d'optimalité de la reconstruction basé sur l'incertitude de localisation de l'objet. L'optimisation de ce critère nous permet d'exprimer deux lois de commande exploratoires définissant les mouvements optimaux de la caméra. Chacune des lois de commande est testée en simulation afin d'étudier ses performances.
- enfin l'ensemble de ces travaux a fait l'objet d'une implémentation sur plate-forme expérimentale dont nous présentons les résultats au chapitre 5. Ces expérimentations valident, en situation réelle, toute la chaîne algorithmique que nous avons développée.

L'annexe A reprend les travaux que nous avons menés dans le cadre de la coopération caméra globale/ caméra locale pour l'asservissement visuel. La tâche d'asservissement que nous présentons consiste, d'une part, à positionner la caméra embarquée par asservissement visuel déporté et, d'autre part à suivre un objet d'intérêt par asservissement visuel embarqué. Nous montrons les problèmes posés par la réalisation simultanée de ces deux tâches et proposons différentes solutions à leur couplage. Ces solutions ont fait l'objet d'expérimentations dont les résultats permettent d'étudier la robustesse de chacune.

Première partie

Recherche des objets dans une scène
encombrée

Rappels concernant l'approche proposée

Nous avons établi que la première étape de l'exploration d'une scène est celle consistant à faire l'inventaire de tous les objets la constituant. C'est un problème de recherche d'objets. Nous avons choisi d'exprimer la connaissance de la scène sous forme de probabilités d'occupation, celles-ci étant définies sur des zones (ou cellules) issues d'un échantillonnage préalable de la scène, construit en fonction des observations de l'image globale. À l'instant initial, les probabilités d'occupation définies sur cet échantillonnage sont initialisées en tenant compte d'un certain *a priori* sur le type de scène (encombrée ou non, dense ou éparse, ...). Mais, aux instants suivants et tout au long de la recherche, le modèle doit intégrer les observations successives qui sont données par les algorithmes de traitement d'image (voir section 1.6) sous forme de probabilités d'occupation le long de chaque ligne de vue.

L'échantillonnage de la scène est défini de la façon suivante : nous identifions, dans l'image globale, les lignes de vue dont la probabilité de détection est supérieure à un certain seuil. Ces lignes de vue sont des axes privilégiés pour la recherche puisqu'elles contiennent, presque sûrement, au moins un objet. Chacun de ces axes est ensuite discrétisé en cellules. Explorer la scène revient alors à vérifier si chacune d'entre elles est occupée ou non, c'est-à-dire à faire tendre la probabilité d'occupation correspondante vers 0 si elle est vide, ou bien vers 1 si elle contient un objet. Pour tenir compte d'éventuelles dépendances entre les cellules, les probabilités d'occupation de chaque cellule ne suffisent pas. Pour cela nous modélisons la connaissance de la scène à un instant donné par une loi conjointe entre toutes les cellules. De cette loi conjointe il est alors facile d'extraire les différentes marginales qui correspondent aux probabilités d'occupation des cellules.

Nous verrons alors que les *a priori* sur la scène ainsi que les observations se traduisent par des contraintes linéaires sur la loi conjointe. C'est pourquoi nous présenterons des outils se rapportant à l'intégration de contraintes linéaires. Celle-ci s'exprime comme la résolution d'un système sous-contraint que l'on résout de manière itérative pour s'affranchir des problèmes d'inversion et qui permet également d'intégrer les observations à mesure qu'elles se présentent. Les observations à intégrer sont, comme pour l'image globale, des probabilités de détection selon chaque ligne de vue mais, dans un souci de clarté de la présentation, nous ne tiendrons compte que de la détection le long de l'axe optique. En d'autres termes, le capteur répond à la question : *avec quelle probabilité peut-on affirmer qu'un objet se trouve sur l'axe optique pour la position courante de la caméra locale?*

Nous nous intéressons également à la construction d'une stratégie de recherche optimale. À un instant donné, la stratégie consiste à déterminer la configuration de la caméra à atteindre pour optimiser la recherche. Il s'agit donc de prédire le gain d'information apporté

par de nouvelles observations. Cette prédiction peut, en théorie, se faire sur un horizon de taille quelconque mais l'utilité de prédire le gain d'information pour un nombre d'itérations supérieur à 1 semble faible tant les prédictions deviennent hasardeuses au delà de deux itérations et d'autant que le coût de calcul devient rapidement considérable voire prohibitif. Nous nous contenterons donc de prévisions à une itération.

Un choix déterminant au sein de l'algorithme de recherche concerne la façon de quantifier le gain d'information induit par une observation donnée. Cette quantification n'est pas triviale dans la mesure où elle doit valoriser les détections et les non détections. Nous proposons quatre quantifications possibles dont trois sont basées sur la norme de la variation des marginales (ou probabilités d'occupation) et diffèrent par le choix de la norme. La dernière quantification proposée porte sur la variation d'entropie de ces marginales. Nous espérons ainsi copier la propriété informative de l'entropie en théorie de l'information et quantifier l'incertitude sur les détections et les non détections.

Cette partie est organisée de la façon suivante :

- au chapitre 1, nous établissons le modèle de connaissance qui est au cœur de notre approche probabiliste et définissons les outils relatifs à l'intégration de contraintes linéaires.
- l'intégration de ces outils au modèle de connaissance ainsi que le modèle d'observation sont détaillés au chapitre 2. Dans ce chapitre, nous décrivons également le principe d'une recherche optimale des objets.
- le chapitre 3 conclut et valide cette étude par le biais de simulations.

Chapitre 1

Modélisation de l'environnement et intégration de contraintes

Dans ce chapitre, nous proposons une modélisation probabiliste de l'environnement pour laquelle l'état de la connaissance à un instant donné est représenté par une loi conjointe sur les probabilités d'occupation ; ces probabilités étant définies sur différentes zones issues d'une discrétisation préalable de la scène. La section 1.1 détaille le modèle et montre que la prise en compte des observations et des *a priori* se traduit sous la forme de contraintes linéaires sur la loi conjointe. Dans les sections suivantes, nous décrivons un processus d'intégration des contraintes linéaires. Plus précisément :

- en section 1.2, nous décrivons le principe de l'algorithme qui se traduit sous forme de résolution d'un système linéaire sous-contraint. L'algorithme vise à exhiber une solution particulière : celle de norme minimale ;
- en section 1.3, nous détaillons l'algorithme itératif permettant de résoudre le système en s'affranchissant des problèmes calculatoires liés à son inversion ;
- les sections 1.4 et 1.5 concernent la modification et la suppression d'une contrainte. Ces problèmes sont également résolus sans recourir à une inversion coûteuse du système ;
- les contraintes dont nous avons parlé jusqu'à présent sont des contraintes égalité. Or sur un vecteur de probabilités jointes pèsent également des contraintes de type inégalité. La résolution de ce problème est décrite en section 1.6 ;
- enfin toute contrainte ne peut être introduite sans vérifier, au préalable, sa redondance et sa compatibilité avec celles déjà introduites. Ces notions sont explicitées en section 1.7.

1.1 Modélisation

Le concept d'image globale que nous avons décrit en introduction joue un rôle important en ce qui concerne la modélisation de l'environnement puisque nous basons la discrétisation de la scène sur les observations de l'image globale. Un échantillonnage préalable de l'image globale détermine un ensemble de lignes de vue. Ces lignes de vue sont les droites passant par le centre optique de la caméra et le centre de chaque zone définie par l'échantillonnage (voir figure 1.1). Nous les appellerons axes globaux (car définis dans l'image globale) ou axes

lorsqu'il n'y a pas d'ambiguïté. Les algorithmes de détection d'objets que nous envisageons d'utiliser (voir section 1.6) permettent de calculer, pour chacun de ces axes, une probabilité de détection. Celle-ci représente la probabilité qu'au moins un objet se trouve le long de cet axe. Nous pouvons réduire la complexité calculatoire de la recherche en restreignant l'ensemble des axes globaux à ceux dont la probabilité de détection est supérieure à un certain seuil λ . Les axes globaux restants, notés \mathcal{A}_j , définissent les axes de recherche des objets. À chacun est associé une probabilité de détection $p_{g_j} > \lambda$.

Intéressons nous à l'un des axes globaux noté \mathcal{A} lorsqu'il n'y a pas d'ambiguïté. \mathcal{A} est discrétisé en n cellules, \mathcal{C}_i $i \in \{1..n\}$ et pour chaque cellule, nous définissons une variable aléatoire d'occupation X_i à valeur dans $\{0,1\}$ de la façon suivante (voir figure 1.2) :

$$X_i = \begin{cases} 1 & \text{si } \mathcal{C}_i \text{ est occupée par un objet} \\ 0 & \text{sinon} \end{cases}$$

La probabilité associée à X_i est notée \mathcal{P}_i . Elle définit la probabilité d'occupation de la cellule \mathcal{C}_i :

$$\mathcal{P}_i(a) = \mathcal{P}(X_i = a) \quad a \in \{0,1\}$$

$\mathcal{P}_i(1)$ est la probabilité que la cellule \mathcal{C}_i soit occupée et $\mathcal{P}_i(0)$ est la probabilité que \mathcal{C}_i soit vide. L'équation de normalisation associée à la probabilité \mathcal{P}_i est :

$$\mathcal{P}_i(1) + \mathcal{P}_i(0) = 1$$

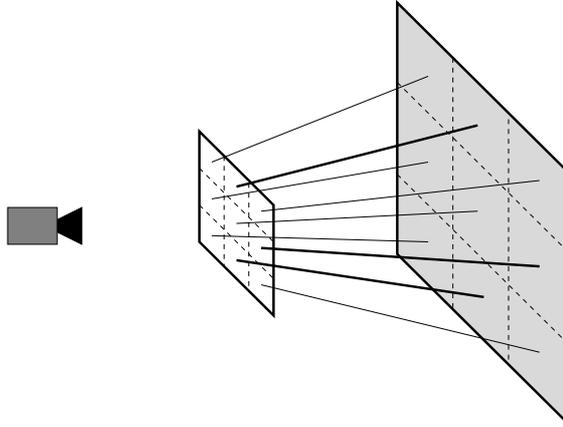


FIG. 1.1 – Un échantillonnage de l'image globale détermine un ensemble de lignes de vue qui sont les droites passant par le centre optique de la caméra et le centre de chaque zone définie par l'échantillonnage.

Comme nous l'avons indiqué en introduction, nous souhaitons garder la possibilité d'introduire des connaissances *a priori* sur la scène (par exemple en fixant le nombre de cellules occupées) pour guider la recherche des objets. Nous pensons gérer ce type de connaissance par l'intermédiaire de dépendances entre les différentes probabilités d'occupation des cellules appartenant à un même axe global. Or, pour prendre en compte d'éventuelles dépendances entre les variables aléatoires X_i de \mathcal{A} , les lois de probabilités \mathcal{P}_i prises individuellement ne suffisent pas. C'est pourquoi nous nous intéressons à la loi conjointe \mathcal{P} du

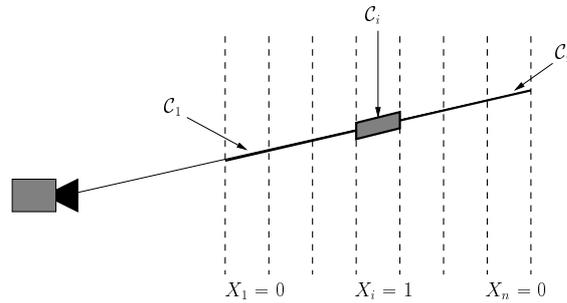


FIG. 1.2 – Les axes de recherche sont les discrétisations des zones de recherche en cellules.

vecteur $X = (X_1, \dots, X_n)$. Une réalisation de X est assimilable à un “mot” de n bits valant 0 ou 1. Il est donc défini sur les nœuds m_l ($l = 0..2^n - 1$) d'un hypercube \mathcal{C} de dimension 2^n . En guise d'exemple, l'hypercube de dimension trois est donné en figure 1.3. Le tableau 1.1 détaille les nœuds ou mots correspondants. Insistons sur le fait que la donnée de la loi

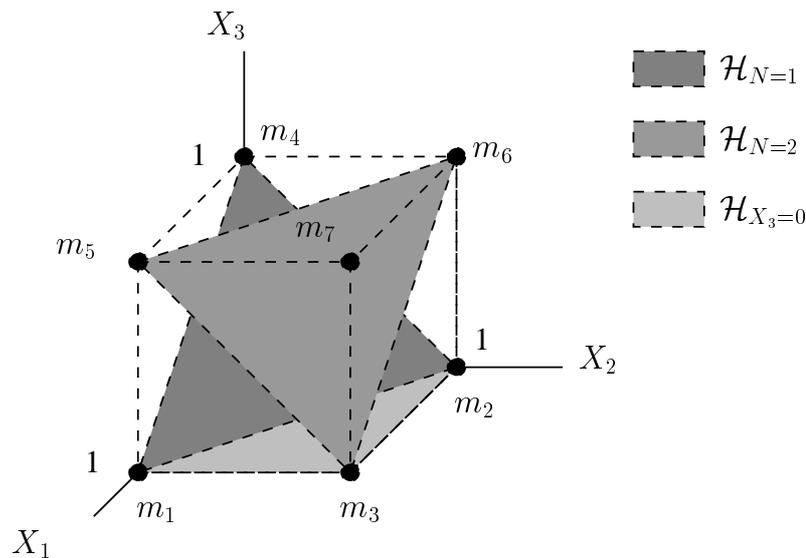


FIG. 1.3 – Hypercube des probabilités conjointes. Chaque niveau de gris représente un hyperplan particulier. N représente le nombre de cellules occupées le long de l'axe \mathcal{A} considéré.

conjointe \mathcal{P} sur l'ensemble des nœuds m_l de l'hypercube \mathcal{C} définit complètement l'axe dans la mesure où \mathcal{C} contient tous les états possibles de \mathcal{A} . Ainsi, nous choisissons comme vecteur d'état représentant la connaissance, le vecteur

$$p = \begin{pmatrix} \mathcal{P}(m_0) \\ \mathcal{P}(m_1) \\ \dots \\ \mathcal{P}(m_{2^n-1}) \end{pmatrix}$$

Nous appelons événement une réunion disjointe de nœuds de \mathcal{C} . Quel que soit l'événement

	X_1	X_2	X_3
m_0	0	0	0
m_1	1	0	0
m_2	0	1	0
m_3	1	1	0
m_4	0	0	1
m_5	1	0	1
m_6	0	1	1
m_7	1	1	1

TAB. 1.1 – En dimension 3, l'hypercube des probabilités jointes contient 8 nœuds. Chaque nœud correspond à un évènement unitaire. Par exemple le nœud m_3 correspond à l'évènement $\{ \mathcal{C}_1 \text{ est occupé} \} \cap \{ \mathcal{C}_2 \text{ est occupé} \} \cap \{ \mathcal{C}_3 \text{ est vide} \}$.

e , sa probabilité se décompose de la façon suivante :

$$\mathcal{P}(e) = \mathcal{P}(X \in e) = \mathcal{P}\left(\bigcup_{m \in e} m\right) = \sum_{m \in e} \mathcal{P}(m)$$

Par conséquent, $\mathcal{P}(e)$ s'écrit comme le produit d'un vecteur ligne h_e avec p :

$$\mathcal{P}(e) = h_e p$$

En guise d'illustration, l'évènement $\{ \mathcal{C}_3 \text{ est vide} \}$ de l'exemple précédent correspond au vecteur de contrainte :

$$h_{\{\mathcal{C}_3 \text{ est vide}\}} = 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0$$

Inversement, quel que soit l'évènement e , fixer $\mathcal{P}(e) = z_e$ revient à imposer la contrainte

$$h_e p = z_e$$

sur p . Nous pourrions donc traduire toute observation ou tout *a priori* z_e sur un évènement e par une contrainte linéaire sur p . Cette contrainte est représentée par le couple (h_e, z_e) . Dans la suite nous donnons la liste de quelques contraintes associées à des évènements particuliers :

- Si l'on définit (conformément à la figure 1.3) pour tout $i \in \llbracket 1, n \rrbracket$ l'hyperplan $\mathcal{H}_{X_i=1}$ des nœuds dont la $i^{\text{ème}}$ composante vaut 1, alors la loi des X_i s'obtient par un calcul de marginales :

$$\begin{cases} \mathcal{P}_i(1) = \mathcal{P}(X \in \mathcal{H}_{X_i=1}) = \sum_{m \in \mathcal{H}_{X_i=1}} \mathcal{P}(m) \\ \mathcal{P}_i(0) = 1 - \mathcal{P}_i(1) \end{cases}$$

De plus, si l'observation de la cellule \mathcal{C}_i est donnée par sa probabilité d'être occupée $z_{m_i} \in [0,1]$ alors l'intégration de cette observation se traduit par une contrainte linéaire sur p du type :

$$h_{m_i} p = z_{m_i} \tag{1.1}$$

où h_{m_i} est le vecteur ligne de taille 2^n

$$\forall j \in \llbracket 0, 2^n - 1 \rrbracket \quad h_{m_i}[j] = \begin{cases} 0 & \text{si } m_j \text{ est associé à l'événement } \{X_i = 0\} \\ 1 & \text{si } m_j \text{ est associé à l'événement } \{X_i = 1\} \end{cases}$$

Le couple (h_{m_i}, z_{m_i}) est appelé $i^{\text{ème}}$ contrainte marginale. À titre d'exemple, en dimension trois, les trois vecteurs de contraintes marginales possibles sont :

$$\begin{aligned} h_{m_0} &= 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ h_{m_1} &= 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ h_{m_2} &= 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{aligned}$$

- Une contrainte très particulière est celle traduisant que p est bien la représentation d'une distribution de probabilité, c'est-à-dire que p est normalisé :

$$\sum_{m \in \mathcal{C}} \mathcal{P}(m) = 1$$

Cette contrainte est incontournable. Elle servira systématiquement à l'initialisation du processus. Le couple qui lui est associé est noté (h_s, z_s) et appelé contrainte de normalisation :

$$\begin{aligned} h_s &= 1 & 1 & \dots & 1 \\ z_s &= 1 \end{aligned}$$

- Considérons N la variable aléatoire à valeur dans $\{0, \dots, n\}$ définie par

$$N = \sum_{i=1}^n X_i$$

N représente le nombre de cellules occupées le long de l'axe \mathcal{A} considéré. Définissons pour tout $i = 0, \dots, n$ l'hyperplan $\mathcal{H}_{N=i}$ contenant les nœuds dont la somme des composantes vaut i (voir figure 1.3). La loi de N se décompose sur la partition formée par les $(\mathcal{H}_{N=i})_{i=0..n}$ en :

$$\forall i \in 0, \dots, n \quad \mathcal{P}_N(i) = \mathcal{P}(N = i) = \sum_{m \in \mathcal{H}_{N=i}} \mathcal{P}(m)$$

Ainsi les $n + 1$ vecteurs de contraintes correspondants $h_{N=i}$, appelés vecteurs de contraintes numériques, permettent d'introduire des *a priori* $z_{N=i}$ sur la distribution du nombre d'objets présents sur l'axe \mathcal{A} . En dimension trois, les 4 vecteurs de contraintes numériques possibles sont :

$$\begin{aligned} h_{N=0} &= 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ h_{N=1} &= 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ h_{N=2} &= 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ h_{N=3} &= 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{aligned}$$

- Il est également possible d'introduire, sous forme d'une contrainte, la dépendance de X_j par rapport à X_i ($j \neq i$), c'est-à-dire la connaissance de $\mathcal{P}(X_j = 1 | X_i = 1)$ et de

$\mathcal{P}(X_j = 1|X_i = 0)$. En effet si on décompose l'événement $\{X_j = 1\}$ sur la partition $\{X_i = 1\} \cup \{X_i = 0\}$ on peut écrire :

$$\begin{aligned}\mathcal{P}_j(1) &= \mathcal{P}(X_j = 1|X_i = 1)\mathcal{P}_i(1) + \mathcal{P}(X_j = 1|X_i = 0)\mathcal{P}_i(0) \\ &= \mathcal{P}(X_j = 1|X_i = 1)\mathcal{P}_i(1) + \mathcal{P}(X_j = 1|X_i = 0)(1 - \mathcal{P}_i(1)) \\ &= \mathcal{P}(X_j = 1|X_i = 0) + [\mathcal{P}(X_j = 1|X_i = 1) - \mathcal{P}(X_j = 1|X_i = 0)]\mathcal{P}_i(1) \\ &= \beta_{ij} + [\alpha_{ij} - \beta_{ij}]\mathcal{P}_i(1)\end{aligned}$$

où $\alpha_{ij} = \mathcal{P}(X_j = 1|X_i = 1)$ et $\beta_{ij} = \mathcal{P}(X_j = 1|X_i = 0)$. L'expression précédente s'écrit donc :

$$\mathcal{P}_j(1) + [\beta_{ij} - \alpha_{ij}]\mathcal{P}_i(1) = \beta_{ij}$$

soit :

$$\sum_{m \in \mathcal{H}_{X_j=1}} \mathcal{P}(m) + [\beta_{ij} - \alpha_{ij}] \sum_{m \in \mathcal{H}_{X_i=1}} \mathcal{P}(m) = \beta_{ij}$$

Il s'agit également d'une contrainte linéaire du type :

$$[h_{m_j} + (\beta_{ij} - \alpha_{ij})h_{m_i}]p = \beta_{ij}$$

Tous les exemples précédents montrent l'intérêt de disposer d'un processus d'intégration des contraintes linéaires. Les sections qui suivent ont pour but de décrire ce processus ainsi que les outils s'y rapportant.

1.2 Principe d'intégration

Notons p_k l'état de connaissance du système à l'instant k . Nous rappelons que ses composantes sont les probabilités conjointes $\mathcal{P}(m_l)$. L'indice k signifie que p_k vérifie k contraintes linéaires $(h_i, z_i)_{i \in [1, k]}$ en plus de l'initialisation. En d'autres termes p_k vérifie :

$$\forall i \in [1, k] \quad h_i p_k = z_i \quad \text{et} \quad h_s p_k = z_s$$

soit le système global suivant :

$$H_k p_k = Z_k \quad \text{où} \quad H_k = \begin{pmatrix} h_s \\ h_1 \\ \cdot \\ \cdot \\ h_k \end{pmatrix} \quad \text{et} \quad Z_k = \begin{pmatrix} z_s \\ z_1 \\ \cdot \\ \cdot \\ z_k \end{pmatrix}$$

Z_k est abusivement appelé vecteur des observations et H_k matrice de contraintes. Le système précédent est un système sous-contraint puisque le nombre de contraintes sera, en général, faible par rapport à la taille du système (2^n lorsque n est le nombre de marginales). La résolution d'un système sous-contraint fournit, en toute rigueur, un hyperplan de solutions parmi lesquelles on peut choisir une solution particulière. En général, pour les systèmes sous-contraints, on utilise la pseudo-inverse à droite :

$$H_k p_k = Z_k \implies p_k = H_k^+ Z_k \quad \text{où} \quad H_k^+ = H_k^T (H_k H_k^T)^{-1} \quad (1.2)$$

Cette solution particulière est celle de norme minimale. Celle-ci est particulièrement intéressante car nous verrons qu'elle garantit, à chaque étape de la méthode itérative, la plus faible variation (en norme) de la solution. Cette propriété est intéressante vis-à-vis de la stabilité des résultats dans le calcul itératif des solutions. Nous choisirons donc également la solution de norme minimale. Une résolution directe de l'équation 1.2 n'est pas souhaitable dans la mesure où, les observations arrivant successivement, elle exigerait une inversion de matrice à chaque itération. C'est pourquoi nous détaillons à présent une méthode de résolution itérative.

1.3 Méthode itérative

Comme nous l'avons vu précédemment, à l'instant k , p_k est donné par

$$p_k = H_k^+ Z_k \quad (1.3)$$

À l'instant $k + 1$, nous disposons d'une contrainte supplémentaire (h_{k+1}, z_{k+1}) . La solution correspondante est alors donnée par

$$\begin{aligned} p_{k+1} &= H_{k+1}^T (H_{k+1} H_{k+1}^T)^{-1} Z_{k+1} \\ &= (H_k^T, h_{k+1}^T) \begin{pmatrix} H_k H_k^T & H_k h_{k+1}^T \\ h_{k+1} H_k^T & h_{k+1} h_{k+1}^T \end{pmatrix}^{-1} \begin{pmatrix} Z_k \\ z_{k+1} \end{pmatrix} \end{aligned} \quad (1.4)$$

En utilisant la formule d'inversion par bloc, on montre que $(H_{k+1} H_{k+1}^T)^{-1}$ s'écrit :

$$G_{k+1} = \begin{pmatrix} G_k + G_k H_k h_{k+1}^T h_{k+1} H_k^T G_k d_k & -G_k H_k h_{k+1}^T d_k \\ -h_{k+1} H_k^T G_k d_k & d_k \end{pmatrix} \quad (1.5)$$

où

$$d_k = \frac{1}{h_{k+1} (\mathbb{I} - F_k) h_{k+1}^T}$$

et en notant pour tout k

$$F_k = H_k^+ H_k \quad \text{et} \quad G_k = (H_k H_k^T)^{-1} \quad (1.6)$$

La matrice F_k joue un rôle primordial dans le processus d'intégration puisqu'elle contient implicitement l'ensemble des contraintes déjà introduites. $(\mathbb{I} - F_k)$ étant l'opérateur de projection sur le noyau¹ de H_k , F_k traduit l'étendue de l'espace des solutions possibles. À ce titre nous l'appellerons **matrice de variabilité**. Nous verrons en section 1.4 et 1.5 que G_k et H_k doivent impérativement être stockés en mémoire lorsque l'on souhaite modifier ou supprimer une contrainte. L'état du processus d'intégration est donc, à tout instant k , caractérisé par les paramètres

$$p_k, F_k, G_k \text{ et } H_k$$

1. On montre aisément que :

$$H_k (\mathbb{I} - F_k) = H_k - H_k H_k^T (H_k H_k^T)^{-1} H_k = 0$$

En remplaçant (1.5) dans (1.4), on montre que

$$\begin{aligned} p_{k+1} &= H_k^+ Z_k + H_k^+ H_k h_{k+1}^T h_{k+1} H_k^+ d_k Z_k \\ &\quad - h_{k+1}^T h_{k+1} H_k^+ d_k Z_k - H_k^+ H_k h_{k+1}^T d_k z_{k+1} + h_{k+1}^T d_k z_{k+1} \\ &= H_k^+ Z_k - (\mathbb{I} - F_k) h_{k+1}^T d_k h_{k+1} H_k^+ Z_k + (\mathbb{I} - F_k) h_{k+1}^T d_k z_{k+1} \end{aligned}$$

Si l'on note :

$$K_{k+1} = (\mathbb{I} - F_k) h_{k+1}^T d_k$$

et en remplaçant (1.3) dans l'expression de p_{k+1} , on montre :

$$p_{k+1} = p_k + K_{k+1}(z_{k+1} - h_{k+1} p_k)$$

D'autre part, on calcule F_{k+1} en utilisant l'équation (1.5) :

$$\begin{aligned} F_{k+1} &= H_{k+1}^+ H_{k+1} \\ &= H_k^+ H_k + H_k^+ H_k h_{k+1}^T h_{k+1} H_k^+ d_k H_k \\ &\quad - h_{k+1}^T h_{k+1} H_k^+ d_k H_k - H_k^+ H_k h_{k+1}^T d_k h_{k+1} + h_{k+1}^T d_k h_{k+1} \\ &= F_k + K_{k+1} h_{k+1} (\mathbb{I} - F_k) \end{aligned}$$

Pour résumer, les formules de réactualisation de l'état du système sont les suivantes :

$$\left\{ \begin{array}{l} p_{k+1} = p_k + K_{k+1}(z_{k+1} - h_{k+1} p_k) \\ F_{k+1} = F_k + K_{k+1} h_{k+1} (\mathbb{I} - F_k) \\ K_{k+1} = \frac{(\mathbb{I} - F_k) h_{k+1}^T}{h_{k+1} (\mathbb{I} - F_k) h_{k+1}^T} \\ G_{k+1} = \left(\begin{array}{cc} G_k + \frac{G_k H_k h_{k+1}^T h_{k+1} H_k^T G_k}{h_{k+1} (\mathbb{I} - F_k) h_{k+1}^T} & - \frac{G_k H_k h_{k+1}^T}{h_{k+1} (\mathbb{I} - F_k) h_{k+1}^T} \\ - \frac{h_{k+1} H_k^T G_k}{h_{k+1} (\mathbb{I} - F_k) h_{k+1}^T} & \frac{1}{h_{k+1} (\mathbb{I} - F_k) h_{k+1}^T} \end{array} \right) \\ H_{k+1} = \left(\begin{array}{c} H_k \\ h_{k+1} \end{array} \right) \end{array} \right. \quad (1.7)$$

Exemple :

Pour guider le lecteur, nous présentons un exemple numérique simple d'intégration d'une contrainte. Cet exemple servira également à la compréhension des outils que nous présenterons dans la suite. Considérons le vecteur p de dimension 3 et dont les composantes sont notées (p_x, p_y, p_z) . Dans un premier temps, nous initialisons le processus avec une contrainte de normalisation du type :

$$p_x + p_y + p_z = 1 \quad (C0)$$

qui correspond au couple :

$$h_0 = [1 \quad 1 \quad 1], \quad z_0 = 1$$

Puisque $H_0 = h_0$, l'initialisation des paramètres p_0 , F_0 et G_0 s'obtient à partir des définitions (1.3) et (1.6) :

$$p_0 = h_0^T (h_0 h_0^T)^{-1} z_0 = \begin{pmatrix} 1/3 \\ 1/3 \\ 1/3 \end{pmatrix}$$

$$F_0 = h_0^T (h_0 h_0^T)^{-1} h_0 = \begin{pmatrix} 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \end{pmatrix}$$

$$G_0 = (h_0 h_0^T)^{-1} = 1/3$$

Dans un deuxième temps, nous intégrons la contrainte

$$p_x = 1/2 \tag{C1}$$

qui est représentée par le couple :

$$h_1 = [1 \ 0 \ 0], \quad z_1 = 1/2$$

Le calcul de p_1 et F_1 s'obtient à partir des équations (1.7), en calculant successivement :

$$K_1 = \frac{(\mathbb{I} - F_0)h_1^T}{h_1(\mathbb{I} - F_0)h_1^T} = \begin{pmatrix} 1 \\ -1/2 \\ -1/2 \end{pmatrix}$$

$$F_1 = F_0 + K_1 h_1 (\mathbb{I} - F_0) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1/2 & 1/2 \\ 0 & 1/2 & 1/2 \end{pmatrix}$$

$$p_1 = p_0 + K_1 (z_1 - h_1 p_0) = \begin{pmatrix} 1/2 \\ 1/4 \\ 1/4 \end{pmatrix}$$

La solution ainsi trouvée vérifie bien les contraintes (C0) et (C1). Reste à exprimer :

$$H_1 = \begin{pmatrix} h_0 \\ h_1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

$$G_1 = \begin{pmatrix} G_0 + \frac{G_0 H_0 h_1^T h_1 H_0^T G_0}{h_1 (\mathbb{I} - F_0) h_1^T} & -\frac{G_0 H_0 h_1^T}{h_1 (\mathbb{I} - F_0) h_1^T} \\ -\frac{h_1 H_0^T G_0}{h_1 (\mathbb{I} - F_0) h_1^T} & \frac{1}{h_1 (\mathbb{I} - F_0) h_1^T} \end{pmatrix} = \begin{pmatrix} 1/2 & -1/2 \\ -1/2 & 3/2 \end{pmatrix}$$

□

Remarques

- Une partie des équations (1.7) rappelle les problèmes de programmation non linéaire sous contrainte dont une formulation peut être trouvée dans [Avriel 76].
- On peut trouver une certaine ressemblance avec les équations d'estimation de Kalman, et faire un parallèle entre K_{k+1} et le gain de Kalman mais également entre la matrice de variabilité et la matrice de covariance. Ces équations sont néanmoins différentes.
- Enfin, comme nous l'avons signalé précédemment, la réactualisation de p_k telle qu'elle est donnée par les équations précédentes est celle qui minimise sa variation en norme $\|p_{k+1} - p_k\|$. En effet p_{k+1} est de la forme $p_k + \delta$ où :

$$\begin{aligned}
 & p_k = H_k^T (H_k H_k^T)^{-1} Z_k \quad \text{et} \quad \delta \in \text{Ker}(H_k) \\
 \implies & p_k^T \delta = 0 \\
 \implies & p_k \perp \delta \\
 \implies & \|p_{k+1}\|^2 = \|p_k\|^2 + \|\delta\|^2
 \end{aligned}$$

p_k étant fixé, comme p_{k+1} est de norme minimale, δ l'est également.

Pour disposer d'un algorithme complet d'intégration de contraintes, nous devons définir un certain nombre d'outils se rapportant à la modification d'une contrainte, à la suppression d'une contrainte, à la gestion des contraintes inégalité et enfin aux tests de redondance et de compatibilité. Ces outils sont définis dans les sections suivantes.

1.4 Modification d'une contrainte

1.4.1 Principe

Supposons qu'à l'itération q une contrainte h_q ait été fixée à la valeur z_q et qu'après l'itération $k \geq q$ nous disposons d'une information plus précise de sa valeur z'_q car obtenue dans de meilleures conditions. Ceci se produit typiquement lorsqu'une détection plus récente bénéficie d'une connaissance plus fine de la scène. Pour ce type de situation, il est nécessaire de construire une méthode permettant de modifier la valeur z_q associée à h_q . Rappelons que l'état de la connaissance à l'instant courant k est déterminé par les paramètres (p_k, F_k, G_k, H_k) . En d'autres termes, nous souhaitons répercuter la modification $z_q \rightarrow z'_q$ sur le quadruplet (p_k, F_k, G_k, H_k) . Notons que F_k , G_k et H_k ne sont pas modifiés puisque l'espace contraint est inchangé. Seul p_k doit être remis à jour. L'idée est de rechercher sa nouvelle valeur sous la forme :

$$p'_k = p_k + \delta$$

où δ détermine la direction de recherche de la nouvelle solution (voir figure 1.4). Si l'on note $H_{k/q}$ la matrice H_k privée de la ligne q , la direction δ doit impérativement appartenir au noyau de $H_{k/q}$ pour que les contraintes autres que h_q ne soient pas modifiées. En conséquence, puisque l'on modifie la valeur de l'observation dans la direction h_q , il faut chercher la nouvelle solution dans la direction de h_q en projection sur le noyau de $H_{k/q}$. En d'autres termes, δ est du type :

$$\delta = \lambda (\mathbb{I} - F_{k/q}) h_q^T$$

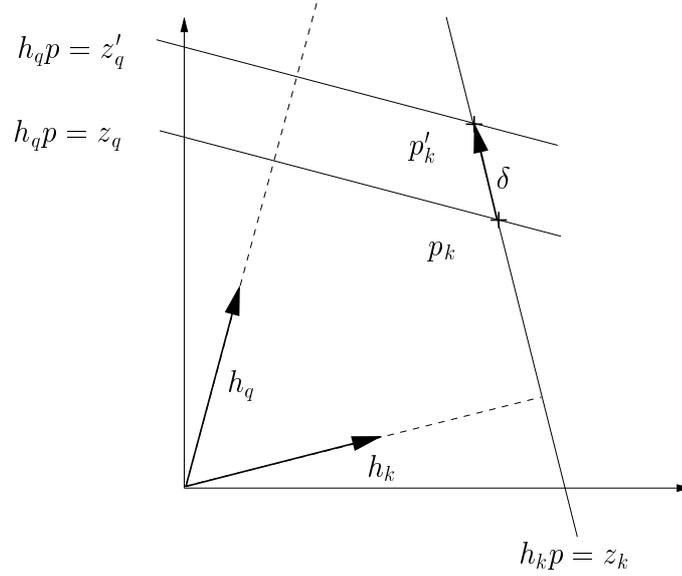


FIG. 1.4 – Représentation schématique en dimension 2 du processus de modification de la contrainte h_q .

Si l'on connaît $F_{k/q}$, on peut calculer λ en remarquant que p'_k vérifie la contrainte :

$$h_q p'_k = z'_q$$

Ce qui se traduit par :

$$z'_q = h_q p_k + \lambda h_q (\mathbb{I} - F_{k/q}) h_q^T \Rightarrow \lambda = \frac{z'_q - h_q p_k}{h_q (\mathbb{I} - F_{k/q}) h_q^T}$$

Et donc

$$p'_k = p_k + \frac{(\mathbb{I} - F_{k/q}) h_q^T}{h_q (\mathbb{I} - F_{k/q}) h_q^T} (z'_q - h_q p_k)$$

Sans trop de surprise, on retrouve une formule semblable à (1.7). Mais il reste évidemment à calculer $F_{k/q}$. Pour cela, nous procédons en deux étapes :

1. Dans un premier temps, il faut se ramener, par permutation, au cas où $q = k$ comme si (h_q, z_q) était la dernière contrainte introduite. Nous noterons $(\tilde{p}_k, \tilde{F}_k, \tilde{G}_k, \tilde{H}_k)$ les paramètres issus de cette permutation.
2. Puis nous devons calculer $F_{k/q}$ qui est la matrice de variabilité correspondant à l'introduction des $k - 1$ contraintes autres que (h_q, z_q) elle vaut donc $\tilde{F}_{k/k}$.

Nous détaillons à présent ces deux étapes.

1.4.3 Calcul de $\widetilde{F}_{k/k}$

Rappelons que par définition (voir équation (1.6)) :

$$\widetilde{F}_{k/k} = \widetilde{H}_{k/k}^T \widetilde{G}_{k/k} \widetilde{H}_{k/k}$$

Nous devons donc calculer $\widetilde{G}_{k/k}$ qui s'obtient à partir de \widetilde{G}_k au vue de l'équation (1.5) et si l'on décompose \widetilde{G}_k sous la forme :

$$\widetilde{G}_k = \begin{pmatrix} A & B \\ B^T & c \end{pmatrix}$$

où A est une matrice carrée, B un vecteur colonne et c un scalaire. Il est alors facile de montrer que

$$\widetilde{G}_{k/k} = A - \frac{BB^T}{c}$$

D'autre part $\widetilde{H}_{k/k}$ est obtenue en supprimant simplement la dernière ligne de \widetilde{H}_k . Comme nous l'avions indiqué précédemment, cette étape nous contraint à stocker H et G à chaque itération.

1.4.4 Résumé des équations de modification

Les étapes permettant de modifier la valeur z_q associée au vecteur de contrainte h_q en la valeur z'_q sont :

1. Décomposer :

$$E_q G_k E_q = \begin{pmatrix} A & B \\ B^T & c \end{pmatrix}$$

auquel cas $G_{k/q} = A - BB^T/c$.

2. $\widetilde{H}_{k/k} = H_{k/q}$ est obtenue en supprimant la $(q+1)^{\text{ème}}$ ligne de H_k
3. Calculer :

$$F_{k/q} = \widetilde{F}_{k/k} = H_{k/q}^T G_{k/q} H_{k/q}$$

4. Finalement l'état du système est donné par :

$$\begin{cases} p'_k = p_k + \frac{(\mathbb{I} - F_{k/q})h_q^T}{h_q(\mathbb{I} - F_{k/q})h_q^T} (z'_q - h_q p_k) \\ F'_k = F_k \\ H'_k = H_k \\ G'_k = G_k \end{cases}$$

Suite de l'exemple :

Dans l'exemple que nous avons présenté, deux contraintes ont été introduites. Nous souhaitons modifier la valeur de la première contrainte (de normalisation) en imposant non plus :

$$p_x + p_y + p_z = 1$$

mais :

$$p_x + p_y + p_z = 2$$

c'est-à-dire que $z_0 = 1$ devient $z'_0 = 2$. En suivant les étapes résumées dans la section précédente pour $q = 0$, nous obtenons :

1.

$$E_0 G_1 E_0 = \begin{pmatrix} 3/2 & -1/2 \\ -1/2 & 1/2 \end{pmatrix}$$

soit $A = 3/2$, $B = -1/2$ et $c = 1/2$ et donc $G_{1/0} = 1$.

2. $H_{1/0} = h_1 = [1 \ 0 \ 0]$

3.

$$F_{1/0} = H_{1/0}^T \left(A - \frac{BB^T}{c} \right) H_{1/0} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

4. Finalement l'état du système est donné par :

$$\begin{cases} p'_1 = p_1 + \frac{(\mathbb{I} - F_{1/0})h_0^T}{h_0(\mathbb{I} - F_{1/0})h_0^T} (z'_0 - h_0 p_1) = \begin{pmatrix} 1/2 \\ 3/4 \\ 3/4 \end{pmatrix} \\ F'_1 = F_1, \quad H'_1 = H_1, \quad G'_1 = G_1 \end{cases}$$

Nous pouvons vérifier que p'_1 satisfait les contraintes (h_0, z'_0) et (h_1, z_1) .

□

1.5 Suppression d'une contrainte

Lorsqu'une contrainte a été introduite et que nous ne souhaitons plus imposer sa satisfaction, typiquement lorsqu'un *a priori* n'apparaît plus correct, nous devons pouvoir procéder à la suppression de la contrainte correspondante. D'après la section précédente, la suppression d'une contrainte, s'effectue en deux étapes :

1. Permutation de la contrainte à supprimer avec la dernière contrainte introduite. Les paramètres correspondants sont $(\widetilde{p}_k, \widetilde{F}_k, \widetilde{G}_k, \widetilde{H}_k)$.
2. Calcul des paramètres après suppression de la contrainte : $(p_{k/q} = \widetilde{p}_{k/k}, F_{k/q} = \widetilde{F}_{k/k}, G_{k/q} = \widetilde{G}_{k/k}, H_{k/q} = \widetilde{H}_{k/k})$.

La première étape est strictement identique à celle que nous décrivons en section 1.4.2. Pour le reste, le calcul de $\widetilde{F}_{k/k}$, $\widetilde{G}_{k/k}$ et $\widetilde{H}_{k/k}$ est identique à ce qui est présenté en section 1.4.3. Seul le calcul de $p_{k/q} = \widetilde{p}_{k/k}$ est différent (voir figure 1.5). Puisque p_k est solution du système entier il est, *a fortiori*, solution du système privé d'une contrainte. Nous pourrions donc, en théorie, conserver $p_{k/q} = p_k$ comme solution mais nous perdriions alors la propriété de norme minimale de la solution et donc celle de variation minimale. Or nous avons vu que cette propriété peut s'avérer intéressante pour favoriser la stabilité des résultats. Pour la conserver nous devons modifier la solution en conséquence :

$$p_{k/q} = p_k + \delta$$

où, comme précédemment, l'axe de modification δ doit être colinéaire à la projection de h_q sur le noyau de $H_{k/q}$ (voir figure 1.5). C'est-à-dire que δ est de la forme :

$$\delta = \lambda(\mathbb{I} - F_{k/q})h_q^T$$

Puisque $p_{k/q}$ est la solution de norme minimale du système privé de la contrainte q , λ peut être obtenu en remarquant que la projection de $p_{k/q}$ sur le noyau de $H_{k/q}$ est nulle, ce qui se traduit par :

$$(\mathbb{I} - F_{k/q})p_{k/q} = 0$$

soit :

$$(\mathbb{I} - F_{k/q})p_k + \lambda(\mathbb{I} - F_{k/q})^2 h_q^T = 0$$

Puisque l'opérateur de projection sur le noyau $(\mathbb{I} - F_{k/q})$ est idempotent³, la contrainte s'écrit :

$$(\mathbb{I} - F_{k/q})p_k + \lambda(\mathbb{I} - F_{k/q})h_q^T = 0 \quad (1.8)$$

L'équation vectorielle précédente représente un ensemble de k contraintes ($F_{k/q}$ est de rang k) dont une seule n'est pas triviale. En effet, pour tout $i \in \llbracket 0, k \rrbracket$ différent de q , la projection du premier terme de l'équation (1.8) sur h_i est nulle par définition du noyau. Reste donc une contrainte possible, celle correspondant à la projection de l'équation (1.8) sur h_q :

$$h_q(\mathbb{I} - F_{k/q})p_k + \lambda h_q(\mathbb{I} - F_{k/q})h_q^T = 0$$

Celle-ci conduit à :

$$\lambda = -\frac{h_q(\mathbb{I} - F_{k/q})p_k}{h_q(\mathbb{I} - F_{k/q})h_q^T}$$

et finalement :

$$p_{k/q} = p_k - \frac{h_q(\mathbb{I} - F_{k/q})p_k}{h_q(\mathbb{I} - F_{k/q})h_q^T}(\mathbb{I} - F_{k/q})h_q^T$$

ce qui est la solution recherchée. En résumé, les étapes permettant de supprimer la contrainte (h_q, z_q) sont :

1. Décomposer :

$$E_q G_k E_q = \begin{pmatrix} A & B \\ B^T & c \end{pmatrix}$$

auquel cas $G_{k/q} = A - BB^T/c$.

2. $\widetilde{H}_{k/k} = H_{k/q}$ est obtenu en supprimant la $(q+1)^{\text{ème}}$ ligne de H_k
3. Calculer :

$$F_{k/q} = \widetilde{F}_{k/k} = H_{k/q}^T G_{k/q} H_{k/q}$$

4. Calculer la solution correspondante :

$$p_{k/q} = p_k - \frac{h_q(\mathbb{I} - F_{k/q})p_k}{h_q(\mathbb{I} - F_{k/q})h_q^T}(\mathbb{I} - F_{k/q})h_q^T$$

3. Une matrice carrée A est idempotente si $A^2 = A$. Il est facile de montrer qu'un opérateur de projection sur un noyau du type $(\mathbb{I} - F)$ est idempotent :

$$\begin{aligned} (\mathbb{I} - H^T(HH^T)^{-1}H)^2 &= \mathbb{I} - 2H^T(HH^T)^{-1}H + H^T(HH^T)^{-1}HH^T(HH^T)^{-1}H \\ &= \mathbb{I} - 2H^T(HH^T)^{-1}H + H^T(HH^T)^{-1}H \\ &= \mathbb{I} - H^T(HH^T)^{-1}H \end{aligned}$$

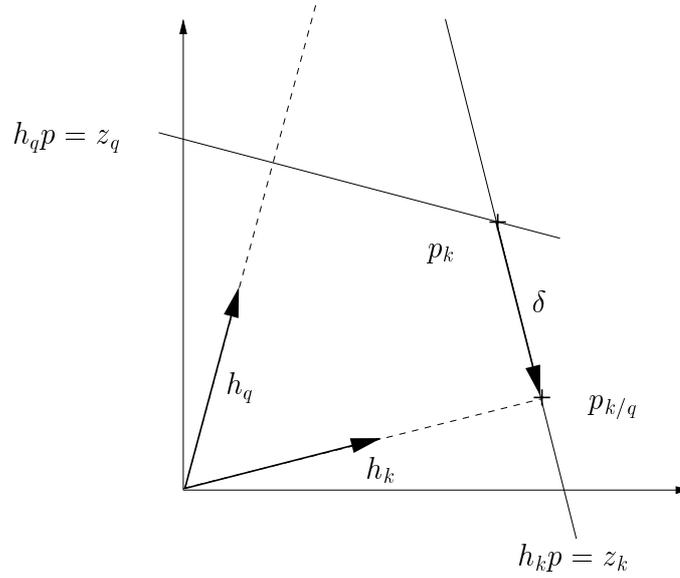


FIG. 1.5 – Représentation schématique en dimension 2 du processus de suppression de la contrainte h_q .

Suite de l'exemple :

Nous illustrons cette section en supprimant la contrainte h_0 dans l'exemple précédent. Les paramètres $F_{1/0}$, $G_{1/0}$ et $H_{1/0}$ ont déjà été calculés. Ils valent respectivement :

$$F_{1/0} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$H_{1/0} = [1 \ 0 \ 0] \quad \text{et} \quad G_{1/0} = 1$$

Reste donc à calculer :

$$\begin{aligned} p_{1/0} &= p_1 - \frac{h_0(\mathbb{I} - F_{1/0})p_1}{h_0(\mathbb{I} - F_{1/0})h_0^T} (\mathbb{I} - F_{1/0})h_0^T \\ &= \begin{pmatrix} 1/2 \\ 0 \\ 0 \end{pmatrix} \end{aligned}$$

Il s'agit bien de la solution de norme minimale de la contrainte $p_x = 1/2$. Dans la suite de l'exemple et pour alléger la notation, nous accolerons abusivement l'indice 0 aux paramètres plutôt que l'indice 1/0 comme si la contrainte h_0 n'avait jamais été introduite.

□

Remarques :

- Pour modifier la valeur z_q de la contrainte h_q en une valeur z'_q , nous pourrions envisager de supprimer la contrainte (h_q, z_q) puis d'intégrer la contrainte (h_q, z'_q) plutôt que d'utiliser les équations de la section 1.4. Cependant un tel schéma conduirait au calcul de $p_{k/q}$ puis au calcul de K'_k et p'_k en utilisant les équations (1.7) alors que le schéma présenté en section 1.4 permet un calcul direct de p'_k .
- Le terme **relaxation** d'une contrainte introduit un concept différent de celui concernant la suppression d'une contrainte. En effet relaxer une contrainte h_q consiste à conserver la solution p_k actuelle tout en posant $F_k = F_k/q$, $G_k = G_k/q$ et $H_k = H_k/q$ ce qui est possible car, p_k étant solution du système global, il est *a fortiori* solution du système relaxé (voir figure 1.5). Cette manipulation permet de prendre en compte l'influence d'une contrainte à l'instant q et de ne plus imposer sa satisfaction à partir de l'instant k . Mais, à cet instant k , la solution obtenue n'est plus celle de norme minimale. Nous verrons, dans la démonstration qui suit, que la perte de cette propriété ne perturbe pas le processus cependant, comme nous l'avons déjà dit, la stabilité des résultats peut apparaître moins bonne.

Preuve :

Les équations (1.7) ont été calculées sous l'hypothèse qu'à chaque itération p_k est la solution de norme minimale. Nous devons vérifier que si, pour une raison quelconque (par exemple après la relaxation d'une contrainte), la solution obtenue à l'instant k vaut $p'_k = p_k + \delta$ où p_k est bien la solution de norme minimale et δ une perturbation appartenant à $Ker(H_k)$, la solution p'_{k+1} obtenue, après intégration d'une contrainte (h_{k+1}, z_{k+1}) à partir des équations (1.7) :

$$p'_{k+1} = p'_k + K_{k+1}(z_{k+1} - h_{k+1}p'_k)$$

avec

$$K_{k+1} = \frac{(\mathbb{I} - F_k)h_{k+1}^T}{h_{k+1}(\mathbb{I} - F_k)h_{k+1}^T}$$

vérifie tout de même l'ensemble de contraintes (H_{k+1}, Z_{k+1}) . En effet :

$$\begin{aligned} H_{k+1}p'_{k+1} &= H_{k+1}p_k + H_{k+1}\delta + H_{k+1}K_{k+1}(z_{k+1} - h_{k+1}p_k - h_{k+1}\delta) \\ &= H_{k+1}[p_k + K_{k+1}(z_{k+1} - h_{k+1}p_k)] + H_{k+1}\delta - H_{k+1}K_{k+1}h_{k+1}\delta \\ &= Z_{k+1} + \begin{pmatrix} H_k\delta - H_kK_{k+1}h_{k+1}\delta \\ h_{k+1}\delta - h_{k+1}K_{k+1}h_{k+1}\delta \end{pmatrix} \end{aligned}$$

Pour montrer que $H_{k+1}p'_{k+1} = Z_{k+1}$, remarquons simplement que

$$H_k\delta = 0$$

car $\delta \in Ker(H_k)$,

$$H_kK_{k+1}h_{k+1}\delta = 0$$

car K_{k+1} appartient au noyau de H_k et enfin

$$h_{k+1}\delta - h_{k+1}K_{k+1}h_{k+1}\delta = (1 - h_{k+1}K_{k+1})h_{k+1}\delta = 0$$

car $h_{k+1}K_{k+1} = 1$ d'après les équations (1.7). □

- Introduire une observation ou une connaissance sous forme de contrainte réelle comme nous l'avons présenté jusqu'à présent (c'est-à-dire en imposant qu'elle soit vérifiée jusqu'à la fin du processus ou bien jusqu'à une éventuelle suppression ou modification de celle-ci) signifie que nous avons, en l'instant, une confiance forte dans l'observation ou la connaissance en question. Ceci n'est évidemment pas toujours le cas. Il est des situations où l'on souhaite modifier la solution en introduisant une contrainte sans en imposer la satisfaction pour les itérations suivantes. Nous appelons ces contraintes des **contraintes virtuelles** (voir section 2.1.3 pour un exemple de l'utilisation des contraintes virtuelles). Elles reviennent à introduire une contrainte réelle puis à la relaxer aussitôt ce qui correspond au cas $q = k$ dans la remarque précédente. Ceci ne change rien à la solution par rapport à une intégration réelle. En revanche la contrainte sera rapidement caduque puisque l'on n'impose pas sa satisfaction pour la suite du processus.

1.6 Contraintes inégalité

Le schéma de réactualisation donné par les équations (1.7) ne gère pas les contraintes de type inégalité. Or chaque composante du vecteur p_k est implicitement contrainte par le fait qu'elle définit, dans le cadre de notre modèle, une probabilité, c'est-à-dire :

$$\forall i \in \llbracket 1, 2^n \rrbracket \quad 0 \leq p_k(i) \leq 1$$

En utilisant les équations (1.7), il est tout à fait possible qu'à une itération k donnée, il existe un ensemble d'indices I tel que pour tout $i \in I$ les $p_k(i)$ correspondants ne vérifient pas cette contrainte. Dans ce cas, il faut modifier la solution tout en conservant la satisfaction des contraintes égalité. Un moyen simple d'y parvenir est de rajouter pour chaque $i \in I$ une contrainte du type

$$p'_k(i) = \begin{cases} 0 & \text{si } p_k(i) < 0 \\ 1 & \text{si } p_k(i) > 1 \end{cases}$$

et de procéder ainsi de manière itérative jusqu'à ce que la contrainte soit vérifiée pour tout i . Il se peut tout à fait que cette procédure ne converge pas vers une solution satisfaisante avant que le nombre de contraintes dépasse le nombre d'inconnues. Lorsque l'on détecte que le système est sur-contraint et que la solution ne satisfait toujours pas les contraintes inégalités, le système n'a pas de solution et il faut supprimer une contrainte d'*a priori* comme nous le ferons en cas d'incompatibilité (voir section 1.7). À la fin du processus, on relâche les contraintes qui ont permis de satisfaire les contraintes inégalités en imposant à F , G et H de reprendre leur valeur antérieure au processus d'intégration des contraintes inégalités. Le relâchement de ces contraintes est nécessaire puisque, dès l'itération suivante, nous n'avons plus de raison d'imposer leur satisfaction.

Remarquons toutefois que, lorsque la contrainte de normalisation est assurée, les contraintes de positivité suffisent à assurer qu'aucune composante du vecteur solution ne sera supérieure à 1. En effet imposer, pour tout i :

$$p_k(i) \geq 0 \quad \text{et} \quad \sum_i p_k(i) = 1$$

implique que :

$$\forall i, p_k(i) \leq 1$$

Suite de l'exemple :

Rajoutons, à présent, une contrainte égalité :

$$p_y = p_z - 1 \quad (C2)$$

correspondant au couple

$$h_1 = [0 \ 1 \ -1], \quad z_1 = -1$$

et trois contraintes inégalité :

$$p_x \geq 0, \quad p_y \geq 0, \quad p_z \geq 0$$

En utilisant les équations 1.7, l'ajout de la contrainte égalité conduit aux paramètres suivants :

$$p_1 = \begin{pmatrix} 1/2 \\ -1/2 \\ 1/2 \end{pmatrix} \quad F_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1/2 & -1/2 \\ 0 & -1/2 & 1/2 \end{pmatrix}$$

$$G_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1/2 \end{pmatrix} \quad H_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \end{pmatrix}$$

La contrainte $p_y \geq 0$ n'est pas vérifiée. Puisqu'il nous reste un degré de liberté (2 contraintes introduites pour 3 variables) nous allons recalculer une solution en imposant la contrainte $p_y = 0$ qui correspond au couple :

$$h_2 = [0 \ 1 \ 0], \quad z_2 = 0$$

En utilisant les équations 1.7, nous obtenons la solution suivante :

$$p'_1 = \begin{pmatrix} 1/2 \\ 0 \\ 1 \end{pmatrix}$$

qui vérifie bien $p_x = 1/2$ et $p_y = p_z - 1$ tout en satisfaisant les inégalités imposées. Une seule itération a suffi dans cet exemple mais il aurait pu se produire qu'une autre composante devienne négative auquel cas le processus aurait dû être réitéré. Ayant trouvé une solution et puisque la contrainte artificielle $p_y = 0$ que nous avons introduite n'a pas lieu d'être pour les itérations à venir, nous devons imposer à F'_1 , G'_1 et H'_1 de reprendre respectivement leur valeur initiale F_1 , G_1 et H_1 .

□

1.7 Redondance et incompatibilité

Avant d'introduire une contrainte, il faut vérifier sa redondance et sa compatibilité avec celles déjà introduites. Cette section décrit ces deux notions ainsi que leur expression mathématique. Nous nous plaçons donc dans la situation où nous devons intégrer une contrainte (h_{k+1}, z_{k+1}) non encore introduite, c'est-à-dire que h_{k+1} est différente de toutes les lignes de H_k . Il se peut toutefois que cette contrainte soit redondante en tant que combinaison linéaire des lignes de H_k . En d'autres termes :

$$\begin{aligned} h_{k+1} \in \text{Im}(H_k) &\Leftrightarrow h_{k+1} \perp \text{Ker}(H_k) \\ &\Leftrightarrow (\mathbb{I} - F_k)h_{k+1}^T = 0 \end{aligned} \quad (1.9)$$

Dans ce cas, K_{k+1} est indéfini. La technique développée en section 1.3 est donc inutilisable. Deux cas peuvent alors se produire :

- Il y a tout de même **compatibilité** de la contrainte c'est-à-dire que

$$h_{k+1}p_k = z_{k+1}$$

auquel cas la nouvelle contrainte ne modifie en rien la solution déjà acquise.

- La nouvelle contrainte est **incompatible** :

$$h_{k+1}p_k \neq z_{k+1}$$

alors le problème est insoluble étant donné l'*a priori* fixé. Nous verrons en section 2.1.1 que la seule solution est de supprimer une à une les contraintes liées à l'*a priori* initial qui est vraisemblablement faux.

En pratique, à cause des approximations numériques, les tests de redondance et de compatibilité ne seront pas des tests d'égalité parfaite mais des comparaisons avec un seuil proche de zéro.

Suite et fin de l'exemple :

Rappelons que la solution actuelle satisfait deux contraintes :

$$(C1) p_x = 1/2 \quad \text{et} \quad (C2) p_y = p_z - 1$$

La contrainte $2p_x + p_y = p_z$ qui correspond au couple :

$$h_2 = [2 \ 1 \ -1], \quad z_2 = 0$$

est redondante puisqu'elle correspond à $2x(C1) + (C2)$; ce que nous vérifions en utilisant l'équation (1.9) :

$$(\mathbb{I} - F_1)h_2^T = 0$$

La contrainte est toutefois compatible car :

$$h_2p_1 = 0 = z_2$$

□

1.8 Conclusion

Dans ce chapitre, nous avons proposé une modélisation probabiliste de l'environnement et décrit un schéma d'intégration itératif des contraintes linéaires. Ce schéma permet, à tout instant, de modifier la solution d'un problème linéaire en fonction d'une nouvelle contrainte égalité. L'état du système est déterminé par sa solution p , une matrice F appelée matrice de variabilité et qui contient l'historique des contraintes déjà introduites ainsi que des matrices G et H dont le stockage est nécessaire pour supprimer ou modifier une contrainte déjà introduite. En effet, outre l'intégration proprement dite, nous avons décrit un certain nombre d'outils s'y rapportant. Un de ces outils concerne la modification d'une contrainte déjà introduite permettant ainsi de modifier la solution du problème en fonction d'une nouvelle valeur de cette contrainte. Nous pouvons également supprimer une contrainte, c'est-à-dire recalculer les paramètres correspondant au même problème privé d'une de ses contraintes. D'autre part, nous avons décrit une méthode d'intégration des contraintes inégalité. Cet outil s'avère nécessaire pour s'assurer que la solution correspond bien à une loi de probabilité. Enfin, nous introduisons les notions de redondance et de compatibilité des contraintes. L'ensemble de ces outils nous permet de construire des algorithmes propres à l'intégration des observations et des *a priori*. Ces algorithmes sont décrits dans le chapitre suivant.

Chapitre 2

Algorithme de fusion de l'observation et stratégies de recherche

Ce chapitre est consacré à l'intégration des observations dans le modèle de connaissance et à la définition d'une stratégie de recherche des objets dans la scène. En section 2.1, nous décrivons les algorithmes permettant de prendre en compte une contrainte dans la mise à jour du modèle et de traduire les observations sous forme de contraintes. Enfin, en section 2.2, nous exposons le principe du calcul des configurations de la caméra permettant d'optimiser la recherche des objets.

2.1 Intégration des observations

Nous rappelons que tout axe global \mathcal{A} est représenté par sa loi conjointe dont l'état est déterminé par les paramètres :

$$(p, F, G, H)$$

Tout d'abord, nous décrivons l'algorithme général d'intégration d'une contrainte sur la loi conjointe d'un axe global quelconque \mathcal{A} . Il utilise les outils du chapitre 1 et décrit leur enchaînement de sorte que l'algorithme puisse gérer un maximum de situations : redondance, incompatibilité, prise en compte de contraintes inégalités... Cet algorithme prend, en entrée, les paramètres de l'état à l'instant courant ainsi que la contrainte à introduire et rend la valeur des paramètres à l'instant $k + 1$ soit

$$(p_{k+1}, F_{k+1}, G_{k+1}, H_{k+1})$$

Il nécessite évidemment une étape d'initialisation des paramètres avec l'intégration d'éventuels *a priori*. Il nous faut ensuite préciser comment une observation déterminée par la *probabilité de détection le long de l'axe optique de la caméra* se traduit en termes de contraintes sur un ou plusieurs axes globaux.

2.1.1 Algorithme d'intégration

Tous les outils décrits au chapitre 1 sont les briques de l'algorithme que nous proposons à présent. Nous nous plaçons dans la situation suivante :

- notre connaissance du modèle est déterminée *a priori*, et pour un axe global \mathcal{A} donné,

par les paramètres

$$(p_k, F_k, G_k, H_k)$$

- à $k + 1$ nous devons intégrer une contrainte (h_{k+1}, z_{k+1}) issue soit d'une observation soit d'un *a priori* sur la scène.

L'algorithme d'intégration est donné figure 2.1 et correspond aux étapes suivantes :

1 Le premier test à effectuer est de savoir si la contrainte a déjà été introduite ou non. Ceci revient à tester l'égalité entre h_{k+1} et toutes les lignes de H_k .

1.1 Si la contrainte a déjà été introduite, nous sommes en présence de deux observations d'une même entité. Si les observations sont égales, la solution reste inchangée. Si elles sont différentes, dans la mesure où l'observation s'affine en même temps que la connaissance du modèle augmente, nous choisissons d'accorder une plus grande confiance à la dernière observation. Par conséquent, il faut modifier la contrainte en fonction de la dernière observation comme indiqué en section 1.4 puis satisfaire les contraintes inégalités selon la méthode proposée en section 1.6.

1.2 Si la contrainte n'a pas été introduite, il faut tester si elle est redondante avec celles déjà introduites (section 1.7).

1.2.1 Si c'est le cas, il faut tester sa compatibilité avec la solution précédente (section 1.7).

1.2.1.1 Si elle est compatible alors la solution reste inchangée.

1.2.1.2 Si elle est incompatible, cela signifie que soit l'*a priori* introduit lors de l'initialisation est faux, soit l'observation courante est erronée. Si nous choisissons de faire confiance à l'observation courante plutôt qu'à l'*a priori*¹ (qui est plus un indicateur qu'une véritable connaissance) alors il faut supprimer une contrainte d'*a priori* (section 1.5) puis renouveler le processus d'intégration (**1.2**). Le choix de l'ordre de suppression des contraintes d'*a priori* peut se faire en fonction de leur dépendance linéaire vis-à-vis de la contrainte courante, c'est-à-dire en commençant par les contraintes d'*a priori* dont le produit scalaire avec h_{k+1} est le plus grand.

1.2.2 Enfin lorsque la contrainte n'est pas redondante avec celles déjà introduites, nous pouvons procéder à son intégration soit réelle (section 1.3) soit virtuelle, suivant le cas, (section 1.5) puis satisfaire les contraintes inégalité.

2.1.2 Initialisation

À l'instant initial, avant même d'introduire le moindre *a priori* sur la scène, nous donnons à p_0 , F_0 , G_0 et H_0 et pour chaque axe \mathcal{A} , les valeurs correspondant à la seule contrainte de normalisation² :

$$\begin{cases} h_0 = h_s \\ z_0 = z_s \end{cases}$$

1. Il est également possible d'accorder une plus grande confiance à l'*a priori* lorsqu'on dispose initialement de connaissances certaines.

2. Ceci correspond au choix d'une loi uniforme pour laquelle les X_i sont indépendantes.

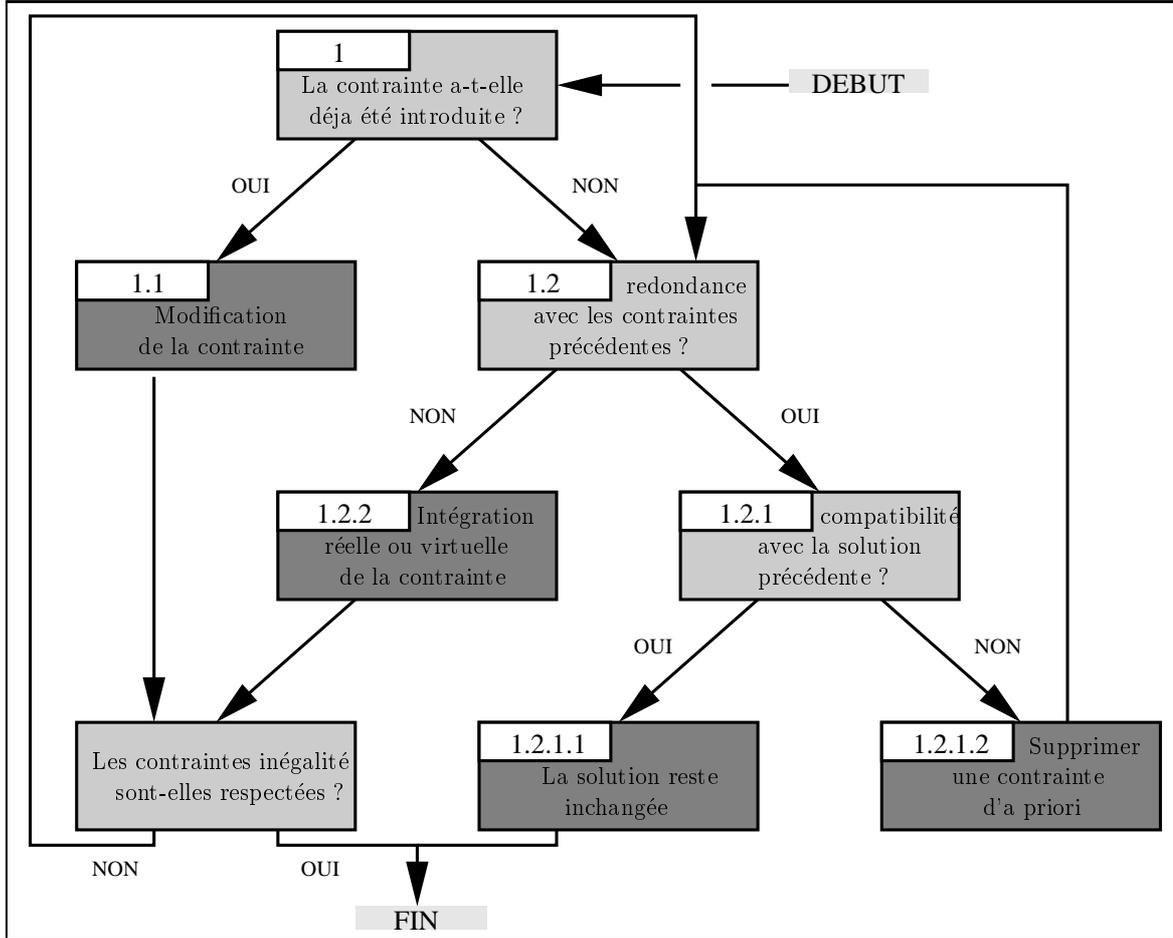


FIG. 2.1 – L'algorithme d'intégration des contraintes prend, en entrée, les paramètres d'état à l'instant courant ainsi que la contrainte et rend la valeur des paramètres à l'instant $k+1$.

ce qui correspond à :

$$\begin{cases} p_0 = h_s^T (h_s h_s^T)^{-1} z_s = \frac{1}{2^n} h_s^T z_s \\ F_0 = h_s^T (h_s h_s^T)^{-1} h_s = \frac{1}{2^n} h_s^T h_s \\ G_0 = \frac{1}{2^n} \\ H_0 = h_s \end{cases}$$

où n est le nombre de cellules de \mathcal{A} . Cette initialisation étant faite, nous devons introduire la contrainte liée à la probabilité p_g de détection³ sur \mathcal{A} . Soit $\{X_i\}_i$ l'ensemble des variables aléatoires d'occupation définies sur les cellules de \mathcal{A} , nous rappelons que p_g définit la

3. Nous rappelons que p_g est la probabilité de détection dans l'image globale des axes susceptibles de comporter au moins un objet, c'est-à-dire que p_g est supérieure à un certain seuil. Lors des simulations, nous considérerons que la probabilité de détection est parfaite c'est à dire que $p_g = 1$ mais tous les intermédiaires sont possibles. En situation réelle, la valeur de p_g est fixée par l'algorithme de détection des objets utilisé.

probabilité que l'une ou l'autre des X_i vaille 1, c'est-à-dire :

$$\begin{aligned}
 p_g &= \mathcal{P}(\bigcup_i \{X_i = 1\}) \\
 &= 1 - \mathcal{P}(\overline{\bigcup_i \{X_i = 1\}}) \\
 &= 1 - \mathcal{P}(\bigcap_i \{X_i = 0\}) \\
 &= 1 - \mathcal{P}(N = 0)
 \end{aligned} \tag{2.1}$$

où N est la variable aléatoire définissant le nombre de cellules occupées le long de \mathcal{A} . La contrainte à introduire est donc la suivante :

$$\mathcal{P}(N = 0) = 1 - p_g$$

Nous pouvons alors introduire des contraintes liées aux *a priori* sur le type de scène. Comme nous l'avons indiqué au chapitre 1, un premier type d'*a priori* à considérer est la loi du nombre N d'objets le long de l'axe \mathcal{A} excepté en $N = 0$. Dans la figure 2.2, nous exprimons deux exemples de lois de N correspondant à deux types de scènes donnés : les scènes éparse et les scènes denses. Toutes les distributions de N sont possibles à condition de s'assurer que

$$\forall s > n, \mathcal{P}(N = s) = 0$$

puisque'il ne peut pas y avoir plus de cellules occupées que de cellules et que

$$p_g = \sum_{s=1}^n \mathcal{P}(N = s)$$

pour une raison évidente de normalisation. Un autre type d'*a priori* lié à des considérations plus fonctionnelles que structurelles consisterait à imposer une dépendance entre cellules. Par exemple lorsque nous souhaitons spécifier qu'un objet dans telle zone (ou cellule) a de fortes chances d'être isolé ou au contraire entouré d'autres objets.

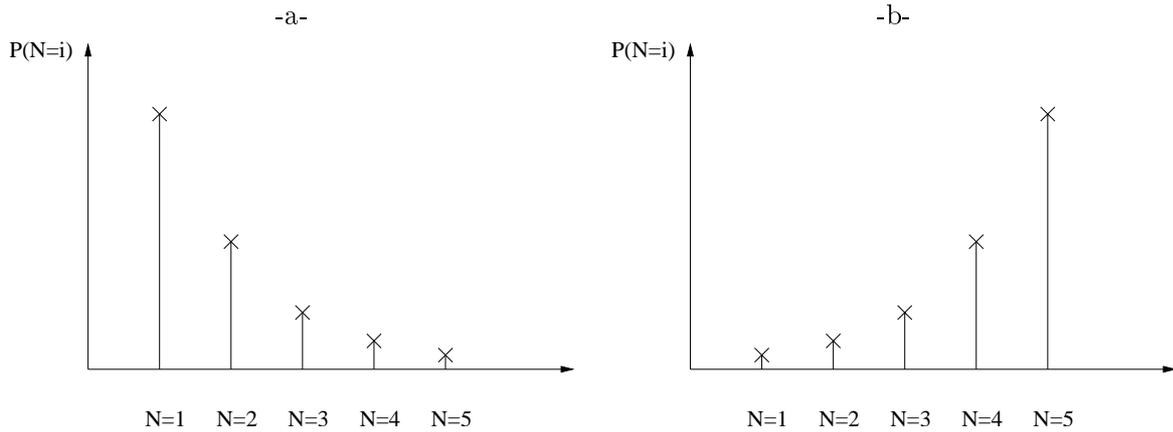


FIG. 2.2 – Exemples de lois de N pour a- une scène éparse b- une scène dense.

Quel que soit le type d'*a priori* et d'après les outils que nous avons présentés, il existe deux façons de les intégrer. En procédant à une intégration virtuelle des *a priori*, il s'agit simplement d'initialiser la distribution à une certaine valeur plus plausible sans imposer la

satisfaction de l'*a priori* au court du processus. La distribution évoluera donc librement et s'écartera progressivement de l'*a priori*. Mais, ayant relaxé les contraintes d'*a priori*, nous perdons la propriété de variation minimale de la norme de la distribution. En conséquence, l'*a priori* sera rapidement obsolète. Une solution qui semble plus appropriée est de procéder à une intégration réelle des *a priori* de sorte qu'ils restent satisfaits jusqu'à la détection d'une éventuelle redondance incompatible (voir section 1.7). Cependant nous verrons en simulation que l'intégration réelle d'un faux *a priori* induit l'algorithme de recherche en erreur si bien que la recherche peut devenir fortement sous-optimale. L'intégration réelle est donc à réserver à des *a priori* certains.

2.1.3 Modèle d'observation

Enfin il nous reste à traduire, en termes de contraintes, l'observation fournie par une caméra. Notons $\{X_{obs_i}\}_i$ l'ensemble des variables aléatoires d'occupation des cellules définissant l'axe observé \mathcal{A}_{obs} à un instant donné. Celui-ci est déterminé par les cellules appartenant à un axe global et croisant l'axe optique de la caméra. La figure 2.3 montre une situation pour laquelle l'axe d'observation contient deux cellules. Nous considérons que

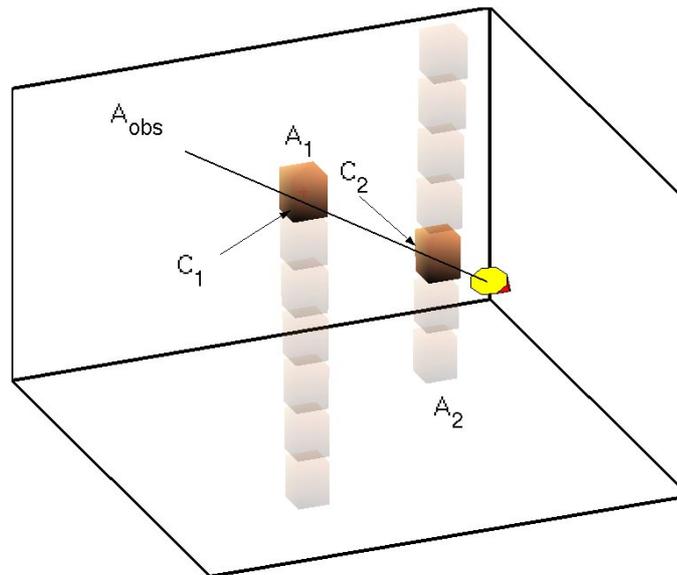


FIG. 2.3 – Ensemble des cellules observées. L'axe observé \mathcal{A}_{obs} contient deux cellules.

le capteur et l'algorithme de traitement associé fournissent à chaque instant la probabilité p_{obs} que l'une ou l'autre des X_{obs_i} soit occupée. De façon équivalente à (2.1), nous pouvons montrer que :

$$p_{obs} = 1 - \mathcal{P}(N_{obs} = 0)$$

où N_{obs} est la variable aléatoire définissant le nombre de cellules occupées parmi celles observées. À l'axe \mathcal{A}_{obs} nous associons une loi conjointe de taille $2^{n_{obs}}$ où n_{obs} est le nombre de cellules de \mathcal{A}_{obs} . Après initialisation de cette loi, comme indiqué en section 2.1.2, nous disposons d'un autre *a priori*: celui lié à l'estimation précédente de l'état du système. En d'autres termes, pour chaque cellule de \mathcal{A}_{obs} , il faut intégrer la contrainte marginale liée à

sa probabilité d'occupation estimée à l'instant précédent. Puis en introduisant la contrainte numérique suivante :

$$\mathcal{P}(N_{obs} = 0) = 1 - p_{obs} \quad (2.2)$$

nous obtenons une nouvelle estimation de la loi conjointe dont les marginales constituent les contraintes à introduire sur les axes globaux auxquels les cellules appartiennent. Ainsi l'algorithme d'intégration d'une observation est le suivant :

- 1 Dans un premier temps, il faut recenser l'ensemble des cellules observées à l'instant courant. Étant donnée une configuration de la caméra, il est facile de déterminer les lignes de vues correspondantes. Nous rappelons que, pour simplifier, nous considérons, comme ligne de vue, uniquement l'axe optique de la caméra. Ayant déterminé cet axe, les cellules observées sont les intersections entre cet axe et les différents axes globaux de la scène.
- 2 Ensuite nous initialisons la loi conjointe associée aux cellules observées. L'initialisation de \mathcal{A}_{obs} consiste en 2 étapes :
 - 2.1 Intégration de la contrainte de normalisation.
 - 2.2 Intégration d'éventuels *a priori* sur la densité de la scène ou autre.
- 3 Puis il faut intégrer la connaissance que l'on a sur les probabilités d'occupation de chaque cellule observée et qui est fournie par l'estimé précédent. Il est à ce stade possible de distinguer deux types d'intégrations suivant le type de cellule dont il est question. Nous dirons qu'une cellule est de type 1 si elle a été observée **seule** au cours du processus, c'est-à-dire que, lors de l'observation, toutes les autres cellules croisées par l'axe optique n'appartenaient à aucun axe global ; sinon la cellule sera dite de type 0. Ainsi nous procéderons :
 - 3.1 à une intégration virtuelle si la cellule correspondante est de type 0 ;
 - 3.2 ou à une intégration réelle si elle est de type 1. Ce choix permet d'augmenter la confiance accordée à l'estimation précédente si la cellule a été observée seule au cours des observations précédentes.
- 4 Nous pouvons alors intégrer la contrainte liée à la probabilité de détection notée p_{obs} (équation (2.2)).
- 5 Enfin, les marginales de la loi conjointe associée à \mathcal{A}_{obs} peuvent être calculées. Chacune d'elles correspond à une contrainte marginale qui doit être introduite à l'axe global auquel la cellule appartient.

Cet algorithme est récapitulé figure 2.4.

Exemple :

Pour illustrer l'algorithme d'intégration d'une observation, nous proposons un exemple concret. Supposons que la scène contienne deux axes globaux \mathcal{A}_1 et \mathcal{A}_2 (voir figure 2.3).

- 1 Étant donnée la configuration de la caméra à l'instant courant, son axe optique croise deux cellules : \mathcal{A}_{obs} est constitué de \mathcal{C}_1 et \mathcal{C}_2 , appartenant respectivement aux axes \mathcal{A}_1 et \mathcal{A}_2 . Si l'on note respectivement X_1 et X_2 les variables aléatoires

décrivant l'occupation de \mathcal{C}_1 et \mathcal{C}_2 , nous supposons que l'itération précédente nous a conduit à estimer les probabilités d'occupation suivantes :

$$\mathcal{P}(X_1 = 1) = \frac{3}{4} \quad \text{et} \quad \mathcal{P}(X_2 = 1) = \frac{1}{2}$$

- 2 Le vecteur représentant la loi conjointe de \mathcal{A}_{obs} est noté p . Il est composé de 4 probabilités élémentaires :

$$p = \begin{pmatrix} \mathcal{P}(\{X_1 = 0\} \text{ et } \{X_2 = 0\}) \\ \mathcal{P}(\{X_1 = 1\} \text{ et } \{X_2 = 0\}) \\ \mathcal{P}(\{X_1 = 0\} \text{ et } \{X_2 = 1\}) \\ \mathcal{P}(\{X_1 = 1\} \text{ et } \{X_2 = 1\}) \end{pmatrix}$$

- 2.1 L'intégration de la contrainte de normalisation conduit à une première estimation :

$$p_0 = \begin{pmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{pmatrix}$$

- 2.2 Dans cet exemple, nous ne prenons pas en compte d'*a priori*.

- 3 Pour intégrer l'estimation précédente des probabilités d'occupation, nous considérons qu'aucune des deux cellules \mathcal{C}_1 et \mathcal{C}_2 n'a été observée individuellement : \mathcal{C}_1 et \mathcal{C}_2 sont de type 0. Nous procédons donc à une intégration virtuelle de leur estimé :

- 3.1 Nous devons intégrer deux contraintes :

$$[0 \ 1 \ 0 \ 1]p = \frac{3}{4} \quad \text{et} \quad [0 \ 0 \ 1 \ 1]p = \frac{1}{2}$$

ce qui conduit à la solution suivante :

$$p'_0 = \begin{pmatrix} 1/8 \\ 3/8 \\ 1/8 \\ 3/8 \end{pmatrix}$$

La notation p'_0 rappelle que l'intégration est virtuelle c'est-à-dire que les paramètres $F'0$, $G'0$ et $H'0$ sont ceux obtenus à l'initialisation.

- 4 Nous allons tester l'algorithme pour deux observations différentes. Dans un premier temps, supposons que l'observation nous indique une probabilité de détection $p_{obs} = 0.95$, c'est-à-dire qu'il y a presque sûrement un objet. Il faut donc intégrer la contrainte :

$$[1 \ 0 \ 0 \ 0]p = 0.05$$

ce qui conduit à une première solution :

$$p_1^1 = \begin{pmatrix} 0.05 \\ 0.4 \\ 0.15 \\ 0.4 \end{pmatrix}$$

Supposons maintenant que l'observation nous indique une probabilité de détection $p_{obs} = 0.1$, c'est-à-dire qu'il y a presque sûrement aucun objet. Il faut donc intégrer la contrainte :

$$[1 \ 0 \ 0 \ 0]p = 0.9$$

ce qui conduit à une autre solution :

$$p_1^2 = \begin{pmatrix} 0.9 \\ 0.05 \\ 0 \\ 0.05 \end{pmatrix}$$

- 5 Les marginales de la loi conjointe associée à \mathcal{A}_{obs} nous donnent alors de nouvelles contraintes qui sont, pour $p_{obs} = 0.95$:

$$\mathcal{P}(X_1 = 1) = 0.8 \quad \text{et} \quad \mathcal{P}(X_2 = 1) = 0.55$$

et pour $p_{obs} = 0.1$:

$$\mathcal{P}(X_1 = 1) = 0.1 \quad \text{et} \quad \mathcal{P}(X_2 = 1) = 0.05$$

□

Les algorithmes que nous venons de présenter permettent pour une configuration donnée de la caméra, d'intégrer l'observation correspondante dans le modèle de connaissance. Le calcul de la prochaine configuration permettant d'optimiser la recherche constitue l'étape ultime du processus. Nous décrivons à présent cette étape.

2.2 Stratégies de recherche

Cette section concerne les choix successifs de points de vue (ou configurations) permettant d'optimiser la recherche des objets dans la scène. Parmi les configurations possibles, seules certaines sont immédiatement accessibles à partir d'une configuration donnée. Celles-ci sont communément appelées *configurations atteignables*. Cette notion se généralise en une autre notion : les t-chemins. Le problème revient alors à choisir parmi cet ensemble de configurations celle qui optimise la recherche. Nous choisissons comme critère l'*espérance du gain d'information (EGI)*. Le calcul de l'EGI pour un t-chemin étant d'une grande complexité calculatoire, nous nous limiterons à des chemins d'ordre 1 c'est-à-dire les configurations atteignables. Pour chaque configuration testée, ce critère fait intervenir la *probabilité de détection a priori* et une *fonction gain d'information (FGI)* qui quantifie le gain apporté par une éventuelle détection ou non détection d'un objet le long de l'axe optique.

2.2.1 Configurations atteignables et t-chemins

Une configuration de la caméra (notée \mathcal{C}) est entièrement déterminée par la cellule occupée par la caméra (\mathcal{C}_p) qui détermine sa position, et une des cellules situées sur son axe optique (\mathcal{C}_d) qui détermine sa direction, soit le couple :

$$\mathcal{C} = (\mathcal{C}_p, \mathcal{C}_d)$$

De façon très générale, à un instant donné, notre problème est de trouver le couple $(\widehat{\mathcal{C}}_p, \widehat{\mathcal{C}}_d)$ optimisant la recherche selon un certain critère. Dans cette optique, le couple $(\widehat{\mathcal{C}}_p, \widehat{\mathcal{C}}_d)$ doit être choisi dans un ensemble de configurations dites atteignables :

- Les déplacements en translation de la caméra ayant un coût (en temps) non négligeable, il est naturel de choisir $\widehat{\mathcal{C}}_p$ parmi toutes les cellules voisines de la cellule courante (voir figure 2.5). Dans cet ensemble de cellules, nous ne conservons que les cellules non occupées. Nous définissons une cellule non occupée comme étant soit une cellule n'appartenant à aucun axe global soit une cellule de type 1 (ayant été observée seule au cours des observations précédentes) et dont la probabilité d'occupation est inférieure à un certain seuil. Nous noterons \mathcal{V} le voisinage ainsi défini.
- $\widehat{\mathcal{C}}_d$ peut être choisie parmi toutes les cellules appartenant à un axe global \mathcal{A}_i de la scène puisqu'il est inutile d'observer des cellules dont on est sûr qu'elles ne sont pas occupées. Contrairement au choix de $\widehat{\mathcal{C}}_p$, nous ne restreignons pas l'ensemble des directions atteignables à un certain voisinage de la direction courante, considérant que les mouvements de rotation ont un coût négligeable.

Pour généraliser cette notion de configuration atteignable, nous appellerons chemin d'ordre t ou t -chemin une succession de t configurations successivement atteignables.

2.2.2 Espérance du gain d'information (EGI)

Le point de vue que nous adoptons est celui qui maximise à chaque itération l'EGI en t étapes. t détermine l'horizon de prédiction que l'on se fixe *a priori*. Étant donné un chemin d'ordre t noté \mathcal{C}_t , la FGI est une fonction $g_{\mathcal{C}_t}$ de la suite des observations, qui prend ses valeurs dans \mathbb{R} . $g_{\mathcal{C}_t}$ doit quantifier le gain apporté par les observations successives le long de \mathcal{C}_t (voir section 2.2.3 pour plus de détails sur le choix de la FGI). L'espérance de $g_{\mathcal{C}_t}$ est donc une prédiction du gain à "espérer" si l'on choisit le chemin \mathcal{C}_t . Pour calculer cette espérance, appelons O^i la variable aléatoire définissant l'observation dans i itérations :

$$O^i = \begin{cases} 1 & \text{si la présence d'un objet est détectée à l'itération } i \\ 0 & \text{sinon} \end{cases}$$

Soit O la suite des observations $(O^i)_{i \in [1, t]}$ au cours du chemin. L'EGI le long de \mathcal{C}_t vaut par définition :

$$E(g_{\mathcal{C}_t}(O)) = \sum g_{\mathcal{C}_t}(O)P(O) = \sum g_{\mathcal{C}_t}(O)P(O^t|O^{t-1}, \dots, O^1) \dots P(O^2|O^1)P(O^1)$$

La somme est prise sur l'ensemble des suites de détections (1 ou 0) possibles, c'est-à-dire l'ensemble des chemins du graphe de la figure 2.6. Le chemin optimal est alors donné par

$$\widehat{\mathcal{C}}_t = \arg \max_{\mathcal{C}_t} E(g_{\mathcal{C}_t}(O))$$

dont il est facile d'extraire la configuration à atteindre à l'instant suivant. Le calcul de $E(g_{\mathcal{C}_t}(O))$ nécessite celui de $g_{\mathcal{C}_t}(O)$ et des probabilités de transition $P(O^i|O^{i-1}, \dots, O^1)$ pour toute suite d'observations possible. Ces quantités s'obtiennent aisément par simulation. Cependant pour chaque chemin d'ordre t (il y en a x^n où x est le nombre de configurations), il faudrait simuler $\sum_1^t 2^n$ fois le système (correspondant au nombre de branches dans le

graphe de la figure 2.6). Même si l'on peut imaginer réduire le nombre de simulations, c'est-à-dire le nombre de branches du graphe, en supprimant les branches dont la probabilité de transition est trop faible, l'optimisation exhaustive de la recherche sur tous les chemins d'ordre t est impensable. D'autre part, la prévision des observations à plus d'une itération semble bien hasardeuse et peu informative. En tenir compte ne relève pas d'un grand intérêt et ne justifie pas la lourdeur des calculs. C'est pourquoi nous limiterons la recherche à des chemins d'ordre 1, c'est-à-dire l'ensemble des configurations atteignables. Si l'on note $O = O^1$ et g la FGI pour la configuration (C_p, C_d) alors l'EGI dans la configuration (C_p, C_d) se simplifie en :

$$E(g(O)) = g(O = 1)P(O = 1) + g(O = 0)P(O = 0) \quad (2.3)$$

D'après les notations de la section 2.2.1, le problème se formalise de la façon suivante :

$$(\hat{C}_p, \hat{C}_d) = \arg \max_{C_p \in \mathcal{V}, C_d \in (\mathcal{A}_i)_i} E(g(O)) \quad (2.4)$$

Dans l'équation (2.3), $P(O = 1)$ est la probabilité de détecter l'objet dans la configuration donnée et $P(O = 0) = 1 - P(O = 1)$. En section 2.1.3, $P(O = 1)$ était une observation réelle notée p_{obs} et issue des algorithmes de traitement d'image et de détection. Dans le cas présent, $P(O = 1)$ n'est qu'une estimation *a priori* de la probabilité de détection dans l'hypothèse où la caméra se déplacerait dans la configuration (C_p, C_d) .

$P(O = 1)$ peut se calculer, sur la base de la connaissance courante, de la façon suivante : \mathcal{A}_{obs} étant l'axe d'observation dans la configuration (C_p, C_d) , nous lui associons une loi conjointe que l'on initialise en intégrant les contraintes liées aux marginales et d'éventuels *a priori* sur le type de scène. Il est ensuite facile de calculer :

$$P(O = 1) = 1 - P(N_{obs} = 0)$$

où N_{obs} est la variable aléatoire définissant le nombre de cellules occupées parmi celles observées.

2.2.3 Fonction gain d'information (FGI)

Reste à définir la FGI g . Elle est définie sur les événements $\{O = 1\}$ et $\{O = 0\}$. En d'autres termes, elle quantifie le gain qu'apporterait une détection et le gain qu'apporterait une non détection. Au vu des algorithmes décrits précédemment, nous sommes capables de simuler l'influence de l'un ou l'autre de ces événements sur le modèle de connaissance puis d'en extraire une nouvelle distribution des probabilités d'occupation sur chaque axe global. Le calcul du gain est naturellement basé sur la variation de cette distribution. Notons p_m le vecteur obtenu en concaténant les distributions marginales selon chaque axe global avant simulation et p_m^+ le même vecteur obtenu après simulation de l'un ou l'autre des événements cités précédemment. Nous considérons des FGI qui dépendent de p_m^+ et p_m et que nous noterons $g(p_m^+, p_m)$. Il existe évidemment une infinité de possibilités de caractériser le gain de p_m^+ par rapport à p_m . Nous en étudierons principalement quatre : la somme des différences des marginales qui quantifie la variation positive des marginales, la norme 2 de

la variation des marginales qui quantifie les variations en norme donc positives et négatives, la norme ∞ de la variation des marginales qui favorise localement les grandes variations de probabilité et enfin la variation de l'entropie qui quantifie le degré d'incertitude sur les probabilités d'occupation.

1. La somme des différences des marginales s'écrit :

$$g_s = \sum_i (p_m^+(i) - p_m(i))$$

On peut noter que g_s est dissymétrique ce qui favorise les variations positives de probabilité au détriment des variations négatives. Nous verrons, lors des simulations, que cette stratégie n'est pas forcément judicieuse dans la mesure où les variations négatives de probabilité apportent également une information intéressante.

2. La norme 2 de la variation des marginales s'écrit :

$$g_2 = \|p_m^+ - p_m\|_2 = \sqrt{\sum_i (p_m^+(i) - p_m(i))^2}$$

Cette FGI favorise la variation globale des probabilités de détection sans tenir compte du sens de variation des probabilités. Contrairement à g_s , une variation positive apporte autant qu'une variation négative des probabilités.

3. La norme ∞ de la variation des marginales s'écrit :

$$g_\infty = \|p_m^+ - p_m\|_\infty = \max_i |p_m^+(i) - p_m(i)|$$

g_∞ favorise localement les grandes variations de probabilité. Cette stratégie vise à améliorer localement le gain d'information au détriment d'une information plus globale. Nous verrons en simulation que ce choix n'est pas pertinent.

4. En théorie de l'information, l'entropie est définie par :

$$S = -\sum_i p(i) \ln p(i)$$

où p est une loi de probabilité discrète. En ce qui nous concerne, p_m n'est pas une loi de probabilité puisqu'elle n'est pas normalisée. Malgré tout nous pouvons tirer parti de ses propriétés. p_m est défini sur un domaine borné :

$$\forall i, 0 \leq p_m(i) \leq 1$$

Sur ce domaine, nous définissons :

$$S_m = -\sum_i p_m(i) \ln p_m(i)$$

S_m est négative et convexe⁴. Elle est maximum en tous les points tels que :

$$\forall i, \text{ soit } p_m(i) = 0 \text{ soit } p_m(i) = 1$$

4. Pour s'assurer de la convexité de l'entropie, il suffit de calculer son hessien et de montrer qu'il est positif.

Cette propriété est très intéressante puisqu'elle quantifie l'incertitude sur les détections et les non détections. Nous définissons alors une nouvelle FGI par la variation d'entropie estimée :

$$g_S = \sum_i [p_m^+(i) \ln p_m^+(i) - p_m(i) \ln p_m(i)]$$

Du dernier point concernant l'entropie apparaît un critère d'arrêt possible : l'entropie du vecteur des lois marginales à l'instant courant soit :

$$Cr = \sum_i p_m(i) \ln p_m(i)$$

Lorsque Cr est suffisamment proche de 0, la recherche peut être stoppée dans la mesure où cela signifie que les différentes probabilités sont soit proches de 0 soit proches de 1. Nous verrons lors des simulations comment se comporte ce critère.

2.3 Conclusion

Dans ce chapitre, nous avons déduit, des outils proposés au chapitre 1, un algorithme général permettant d'intégrer des contraintes. L'enchaînement des outils que nous proposons est de nature à gérer un maximum de situations : redondance, incompatibilité, satisfaction des contraintes inégalité ... L'algorithme permet ainsi d'intégrer tout type de contrainte sous réserve d'avoir initialisé le système de façon adéquat. L'étape d'initialisation consiste d'une part à traduire le fait que la solution du problème est une loi de probabilité (contrainte de normalisation), d'autre part à intégrer la probabilité de détection globale sur l'axe concerné et enfin à intégrer des *a priori* sur la scène. Puis nous avons décrit le processus permettant de transformer une observation en contraintes sur les lois conjointes associées aux axes globaux. Enfin nous avons présenté une méthode permettant d'optimiser la recherche des objets dans la scène. Étant donnée la connaissance à l'instant courant, l'algorithme que nous proposons permet de déterminer la configuration suivante qui maximisera l'espérance du gain d'information. Cette fonction gain d'information doit quantifier le gain apporté par une modification des lois de probabilités marginales. Nous en avons proposés 4 : la somme des différences, la norme 2, la norme ∞ et la variation de l'entropie. Les simulations que nous proposons dans le chapitre suivant doivent permettre de valider l'une ou l'autre de ces FGI ainsi que l'algorithme dans sa globalité.

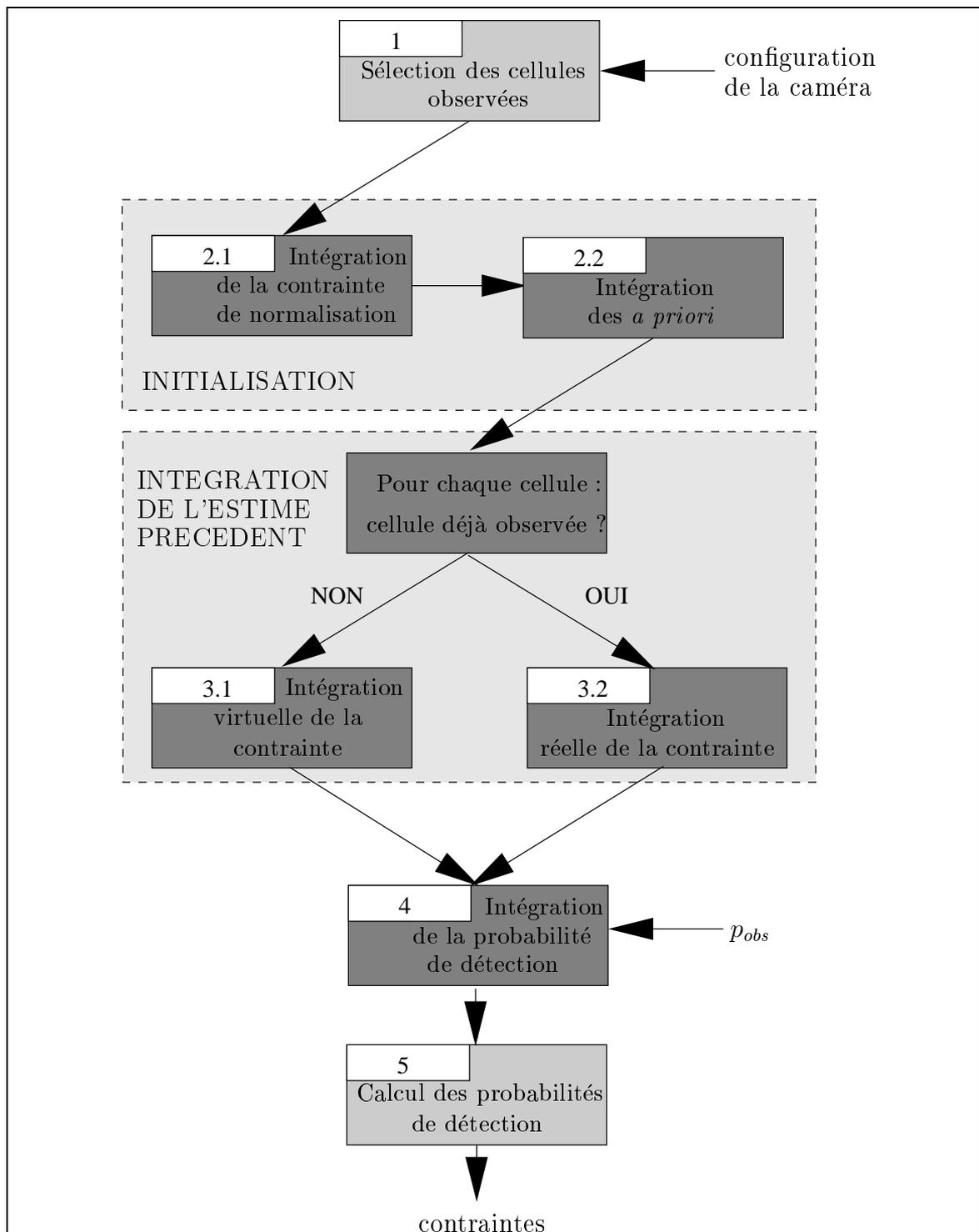


FIG. 2.4 – L'algorithme lié à l'observation traduit l'observation sous forme d'une ou plusieurs contraintes sur le modèle de connaissance.

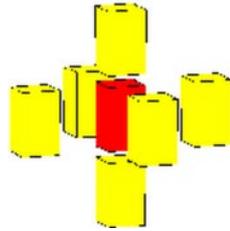


FIG. 2.5 – Exemple de voisinage à 6 voisins.

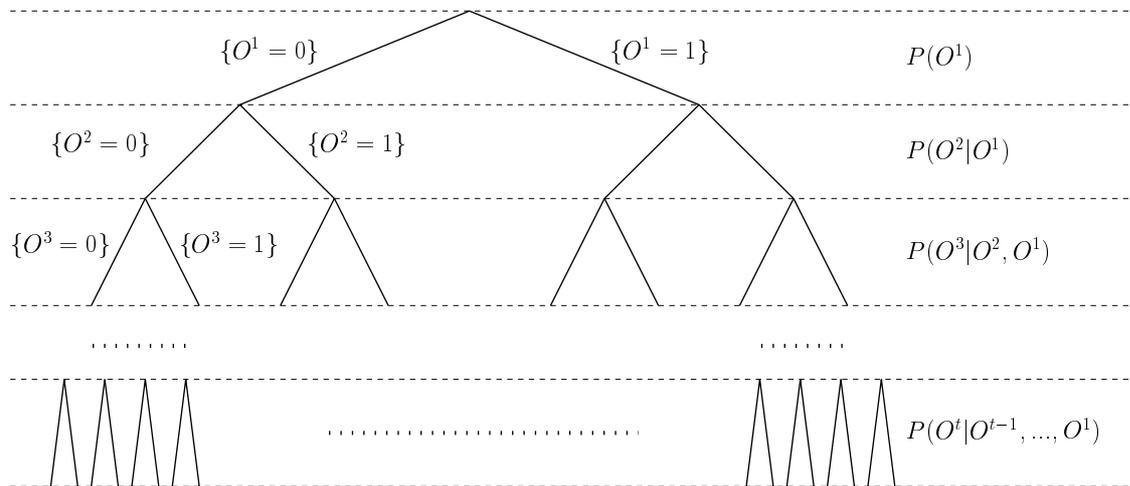


FIG. 2.6 – Graphe des observations.

Chapitre 3

Simulations

Dans cette partie, nous nous proposons de tester différents aspects de l'algorithme de recherche que nous venons de décrire. Pour cela nous définissons un domaine de recherche cubique discrétisé en $6 \times 6 \times 6$ cellules. Après avoir positionné un certain **nombre d'objets** dans la scène, l'initialisation, conformément à l'image globale, consiste à définir les axes globaux du domaine le long desquels nous devons effectuer la recherche. Nous supposons que la probabilité de détection est certaine et vaut 0 ou 1. En fonction de **l'a priori sur la loi du nombre d'objets**, la loi conjointe relative à chaque axe peut être initialisée, ce qui fournit l'estimation initiale de la connaissance et donc les probabilités d'occupation dans chaque cellule. La caméra est alors positionnée arbitrairement dans un des coins du domaine. Nous supposons qu'elle fournit également des observations sous forme de probabilités de détection valant 0 ou 1. Le processus de recherche peut alors commencer dès que l'on a choisi une **FGI** déterminant la stratégie de recherche. Au vu de la description précédente, trois choix conditionnent le processus : le nombre d'objets dans la scène, l'*a priori* sur la loi du nombre d'objets et la FGI. Nous proposons donc trois types de simulations :

1. La première série de simulations consiste à tester, pour une scène donnée, l'influence du choix de la FGI. Nous comparons l'efficacité de chaque stratégie (section 3.1).
2. Ensuite nous choisissons comme FGI la fonction g_2 et augmentons le nombre d'objets dans la scène (nous plaçons 4 objets sur chaque axe) tout en fixant un *a priori* erroné (probabilité d'avoir 3 objets égale à 1). Enfin, pour la même scène que précédemment et la même fonction g_2 , nous changeons l'*a priori* de sorte qu'il corresponde à la réalité (probabilité de 1 d'avoir 4 objets sur chaque axe). Cette simulation permet de valider l'intérêt de l'*a priori* sur la scène (section 3.2).

La section 3.3 présente une étude de la stratégie gloutonne. Cette stratégie consiste à observer tour à tour chaque cellule en les choisissant de façon aléatoire.

3.1 Choix de la fonction gain d'information

La première série de simulations consiste à tester l'influence du choix de la FGI. Pour cela nous positionnons deux objets selon deux axes globaux distincts (voir figure 3.1). L'*a priori* sur le nombre d'objets est exact soit une probabilité de 1 d'avoir un objet selon chaque axe global. Dans ces conditions, nous simulons les stratégies de recherche correspondant aux FGI décrites en section 2.2.3. Afin de comparer les stratégies entre

elles, nous simulons la recherche dans différentes situations correspondant à des positions initiales distinctes de la caméra. Ces positions sont les huit sommets du cube définissant le domaine de recherche (figure 3.1). Chacun de ces sommets est indicé de 0 à 7. Pour une scène donnée, le nombre moyen d'itérations nécessaire à la recherche (NIM) est une quantité pertinente pour comparer les stratégies entre elles et vis-à-vis de la stratégie gloutonne dont l'étude théorique est donnée en section 3.3.

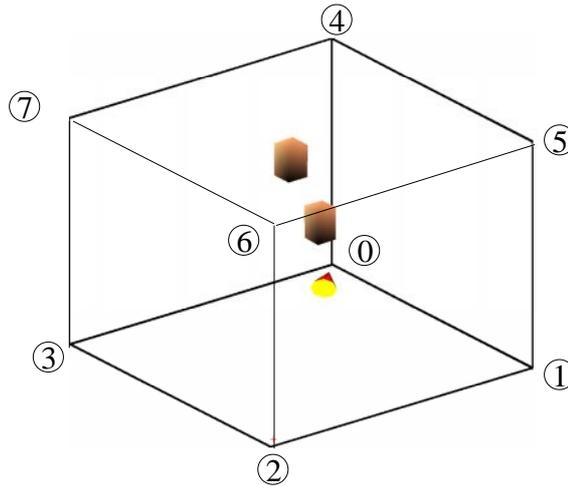


FIG. 3.1 – Scène contenant 2 objets sur des axes distincts.

3.1.1 Somme des différences

Comme nous l'avons déjà remarqué, g_s étant dissymétrique, elle favorise les variations positives de probabilité (c'est-à-dire la détection) au détriment des variations négatives. Or il existe des situations pour lesquelles une non détection doit être valorisée. Dans une scène très encombrée, il est probable qu'une non détection apporte plus qu'une détection. Un autre problème est apparu lors des simulations de la stratégie SD, celui des cycles limites. La figure 3.3 montre un exemple où la recherche aboutit à un tel cycle. Les vignettes 10 à 15 présentent une période de répétition de deux itérations. L'étude de ces cycles limites est trop complexe pour être faite de façon rigoureuse. Cependant, nous pouvons imaginer intuitivement que le fait que l'algorithme cherche, en priorité, à augmenter les probabilités de détection alors que chaque axe ne contient qu'un objet, n'est pas indifférent à l'apparition de ce phénomène.

La première colonne du tableau 3.1 donne les nombres d'itérations nécessaires pour chaque simulation. Les symboles ∞ signifient que la simulation aboutit à un cycle limite. Si l'on omet ceux-ci, le nombre moyen d'itération vaut :

$$NIM_s \approx 9.3$$

ce qui est relativement médiocre comparé à ce que donnerait, en théorie, une stratégie gloutonne : $NIM=6.7$ (section 3.3).

Position initiale	SD	N_2	N_{inf}	Δ_S
0	∞	4	8	5
1	8	4	4	5
2	12	8	9	7
3	8	4	4	5
4	∞	6	7	8
5	8	6	6	7
6	12	9	9	8
7	8	6	6	7
Moyenne		5.8	6.6	6.5

TAB. 3.1 – Nombre d’itérations nécessaire à la recherche pour chaque FGI et pour différentes positions initiales de la caméra.

3.1.2 Norme 2

Cette FGI favorise la variation globale des probabilités de détection sans tenir compte du sens de variation des probabilités. Contrairement à g_s , une variation positive apporte autant qu’une variation négative des probabilités.

La stratégie correspondante est très performante en pratique puisqu’elle conduit à un NIM de 5.8 pour la scène choisie. La figure 3.4 montre un exemple de recherche utilisant cette méthode. Hormis la première observation qui correspond à une configuration initiale prédéterminée, les autres configurations concernent des observations de type 1. La stratégie semble privilégier les configurations ne conduisant pas à des occultations d’un axe global par rapport à un autre. La figure 3.5 montre l’évolution du critère et confirme qu’au bout de 4 itérations, l’entropie du système est nulle.

3.1.3 Norme ∞

Cette norme favorise localement les grandes variations de probabilité, qu’elles soient positives ou négatives. Cette maximisation se fait donc au détriment des variations moyennes sur tout un axe. La recherche correspondante fournit un NIM de 6.6 moins performant que la norme N_2 et équivalent à la stratégie gloutonne. La figure 3.6 est un exemple de recherche utilisant ce type de stratégie. Nous nous apercevons que, contrairement à la stratégie N_2 , les observations doubles sont préférées aux observations uniques. La figure 3.7 correspond à l’évolution de l’entropie au cours de la recherche. Elle s’annule au bout de 8 itérations.

3.1.4 Variation de l’entropie

L’entropie quantifie l’incertitude sur les détections et les non détections. En maximisant à chaque itération la variation d’entropie Δ_S , nous espérons optimiser la recherche. Les simulations effectuées en utilisant cette FGI fournissent un NIM de 6.5 ce qui améliore la stratégie gloutonne sans toutefois atteindre les performances de la norme N_2 . La figure 3.8 est un exemple de recherche utilisant cette stratégie. La figure 3.9 correspond à l’évolution du critère choisi.

3.2 Influence de l'a priori

Les trois simulations suivantes correspondent à une scène différente : 8 objets sont répartis sur 2 axes (4 objets par axe). Dans un premier temps, l'a priori sur la scène est fixé à 3 objets par axe (la probabilité initiale qu'il y ait 3 objets par axe vaut 1). L'a priori est donc erroné. Une recherche basée sur la norme N_2 s'effectue en 17 étapes (voir figure 3.10). Lorsqu'il est faux, l'a priori conduit donc à une recherche largement sous-optimale. L'évolution du critère (voir figure 3.11) présente des décroissances aux étapes 9 et 13 ce qui correspond à des accroissements d'incertitude. En effet, aux instants 9 et 13, certaines cellules passent d'une probabilité 1 ou 0 à une probabilité intermédiaire. Cette perte de certitude sur la probabilité d'occupation d'une cellule provient du fait que l'a priori est mauvais. Nous constatons également (voir figure 3.11) que le critère peut être nul sans que la recherche soit achevée (itérations 15 et 16). Nous avons donc volontairement inhibé le critère d'arrêt et stoppé manuellement la recherche. Les étapes 15 et 16 correspondent à des situations pour lesquelles la connaissance est certaine mais fausse ce qui est typiquement le genre d'information véhiculée par un mauvais a priori. Il faut noter que dans la même situation, une stratégie gloutonne se serait arrêtée dès que le nombre d'objets rencontrés aurait atteint la valeur fixée par le mauvais a priori donc prématurément.

Dans un deuxième temps, l'a priori est correct (4 objets par axe). La recherche s'effectue alors sur 9 itérations (voir figures 3.12 et 3.13). Le NIM d'une stratégie gloutonne pour le même type de scène vaut 8.5. La figure 3.14 montre la recherche que fournit une stratégie basée sur la variation d'entropie. Celle-ci semble plus efficace dans une scène encombrée puisqu'elle aboutit en 7 étapes pour les mêmes conditions de recherche, donc bien plus rapidement que le NIM calculé pour une stratégie gloutonne.

3.3 Stratégie gloutonne

Dans cette section, nous faisons l'étude théorique de la stratégie gloutonne afin d'en comparer les performances aux stratégies que nous proposons. La stratégie gloutonne consiste à parcourir toutes les cellules dans un ordre aléatoire jusqu'à ce que tous les objets soient repérés. Cela suppose donc une détection binaire et exacte des objets : "Il y a un objet" ou "il n'y a pas d'objet". Cette stratégie est donc très pauvre mais son NIM servira de référence pour les simulations que nous proposons.

Considérons un axe global comportant n cellules. Sans a priori sur le nombre d'objets présents sur l'axe, la stratégie gloutonne doit systématiquement observer une à une les n cellules. Dans ce cas $NIM = n$. Mais si p est le nombre d'objets et que p est connu de façon exacte (cette exigence est une nouvelle limitation de la stratégie gloutonne), alors le nombre moyen d'itérations nécessaire à la recherche vaut :

$$NIM = p \left[\sum_{i=p}^{n-1} \frac{(i-1)!}{(i-p)!} i + b(1) + a(1)(b(2) + a(2)(b(3) + a(3)(\dots))) \right]$$

$$\text{où } a(i) = p - i \quad \text{et} \quad b(i) = \frac{(n-i-1)!}{(n-p-1)!} (n - i)$$

Preuve :

Une stratégie de recherche gloutonne correspond au placement équiprobable de p objets distincts parmi n cellules soit un nombre total de possibilité valant $n!/(n-p)!$. Les cellules étant ordonnées du début à la fin de la recherche, nous appellerons I_i la variable aléatoire correspondant au numéro de l'itération associée à la rencontre du $i^{\text{ème}}$ objet (voir figure 3.2). Pour tout i , I_i est à valeurs dans $\llbracket 1, n \rrbracket$ et

$$1 \leq I_1 < I_2 < \dots < I_p \leq n$$

Tous les objets sont donc observés en I_p observations ($I_p \in \llbracket p, n \rrbracket$). Lorsque le nombre total d'objets p est connu, la recherche s'arrête soit lorsque p objets ont été observés soit lorsque $n - p$ cellules vides ont été observées. Notons A la variable aléatoire définissant le nombre d'itérations nécessaire à la recherche. A est définie de la façon suivante :

$$A = \begin{cases} I_p & \text{si } I_p \in \llbracket p, n-1 \rrbracket \\ n-1 & \text{sinon et si } I_{p-1} \in \llbracket p-1, n-2 \rrbracket \\ \vdots & \\ n-p+1 & \text{sinon et si } I_1 \in \llbracket 1, n-p \rrbracket \\ n-p & \text{sinon} \end{cases}$$

Nous sommes en présence d'une relation de récurrence dont il faut compter les occurrences. Ayant choisi l'objet occupant la dernière place (il y a p possibilités) :

- le nombre d'occurrences pour lesquelles $I_p = i$ avec $i \in \llbracket p, n-1 \rrbracket$ vaut le nombre de façons de placer $p-1$ objets parmi $i-1$ cellules :

$$\frac{(i-1)!}{(i-p)!}$$

Dans chacune de ces situations, $A = i$.

- si l'on fixe $I_p = n$
 - le nombre d'occurrences pour lesquelles $I_{p-1} \in \llbracket p-1, n-2 \rrbracket$ vaut le nombre de façons de placer $p-1$ objets parmi $n-2$ cellules :

$$\frac{(n-2)!}{(n-p-1)!}$$

Dans ce cas $A = n-1$.

- fixons $I_{p-1} = n-1$, il y a $p-1$ possibilités
 - le nombre d'occurrences pour lesquelles $I_{p-2} \in \llbracket p-2, n-3 \rrbracket$ vaut le nombre de façons de placer $p-2$ objets parmi $n-3$ cellules :

$$\frac{(n-3)!}{(n-p-1)!}$$

Dans ce cas $A = n-2$.

- etc.

p	NIM
0	0
1	10/3
2	64/15
3	9/2
4	64/15
5	10/3
6	0

TAB. 3.2 – Valeur du nombre d’itérations moyen en fonction de p dans le cas d’une stratégie gloutonne et pour $n = 6$.

En résumé, si on note

$$a(i) = p - i \quad \text{et} \quad b(i) = \frac{(n - i - 1)!}{(n - p - 1)!} (n - i)$$

alors l’espérance de A se calcule par la formule suivante :

$$E(A) = p \left[\sum_{i=p}^{n-1} \frac{(i-1)!}{(i-p)!} i + b(1) + a(1)(b(2) + a(2)(b(3) + a(3)(\dots))) \right]$$

□

Dans le cas $n = 6$, $E(A)$ prend les valeurs indiquées dans le tableau 3.2. Par conséquent, la scène contenant deux objets sur deux axes différents (figure 3.1) utilisée pour les simulations précédentes correspond à un NIM d’environ 6.7.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$I_1 = 3$			$I_2 = 5$			$I_3 = 9$			$I_4 = 10$		$I_5 = 13$			

FIG. 3.2 – Exemple de stratégie de recherche gloutonne pour $n = 15$ et $p = 5$.

3.4 Conclusion

Les recherches que nous avons simulées ont montré l’importance du choix de la FGI. Les stratégies basées sur la norme N_2 et sur la variation de l’entropie apportent un gain significatif par rapport à une stratégie gloutonne. La méthode utilisant la variation d’entropie semble plus adaptée aux scènes encombrées. Les figures 3.15 et 3.16 montrent une dernière simulation, utilisant la variation d’entropie, dans une scène plus encombrée puisqu’elle contient 10 objets. Elle aboutit en 18 étapes au repérage de tous les objets alors que le NIM en stratégie gloutonne vaut 21.3. Le critère d’arrêt basé sur l’entropie des lois marginales fonctionne parfaitement lorsque l’*a priori* sur la scène est correct. Lorsqu’il est erroné, il existe des situations pour lesquelles l’entropie est nulle alors que les probabilités d’occupation sont fausses. Introduire un *a priori* faux est très pénalisant à la fois en ce qui concerne le critère mais également au niveau du nombre d’itérations nécessaire à la découverte de tous les objets. Mieux vaut donc ne pas fixer d’*a priori* que d’en fixer un qui soit faux. Une solution, lorsque l’*a priori* n’est pas certain, serait sans doute de l’intégrer de façon virtuelle (voir section 1.5).

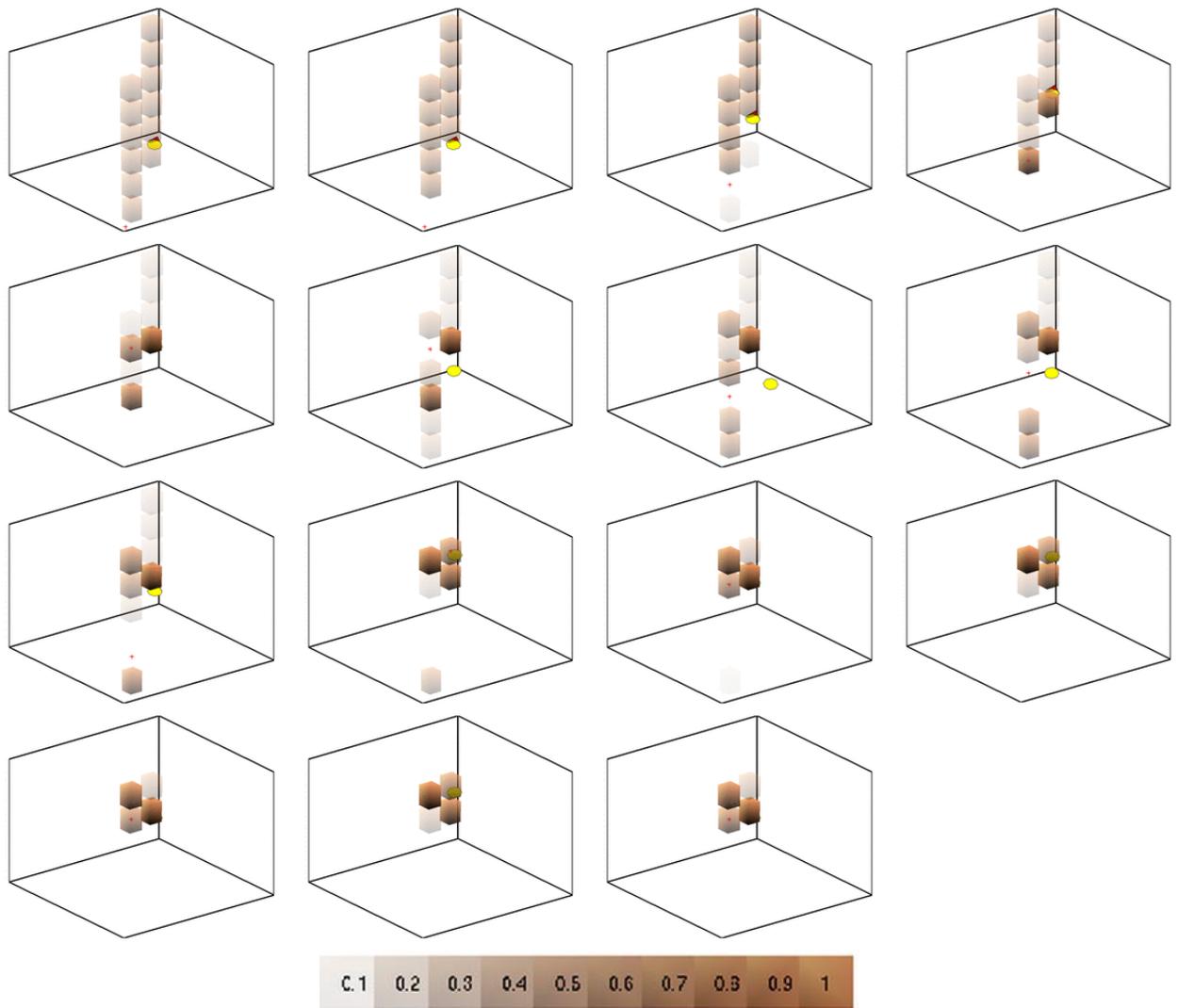


FIG. 3.3 – Stratégie basée sur la somme des différences. La caméra est initialement placée en position 0. La stratégie conduit à un cycle limite.

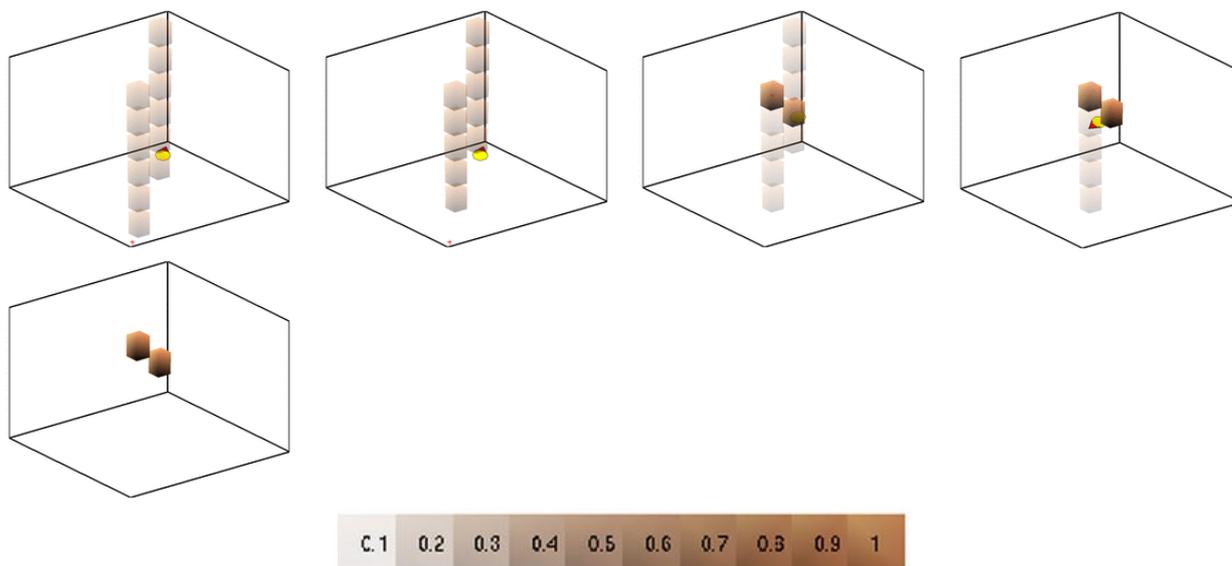


FIG. 3.4 – Stratégie basée sur la norme N_2 . La caméra est initialement placée en position 0.

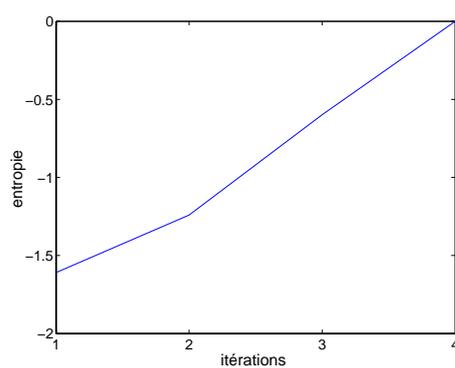


FIG. 3.5 – Evolution du critère lorsque la stratégie est basée sur la norme N_2 et que la caméra est initialement placée en position 0.

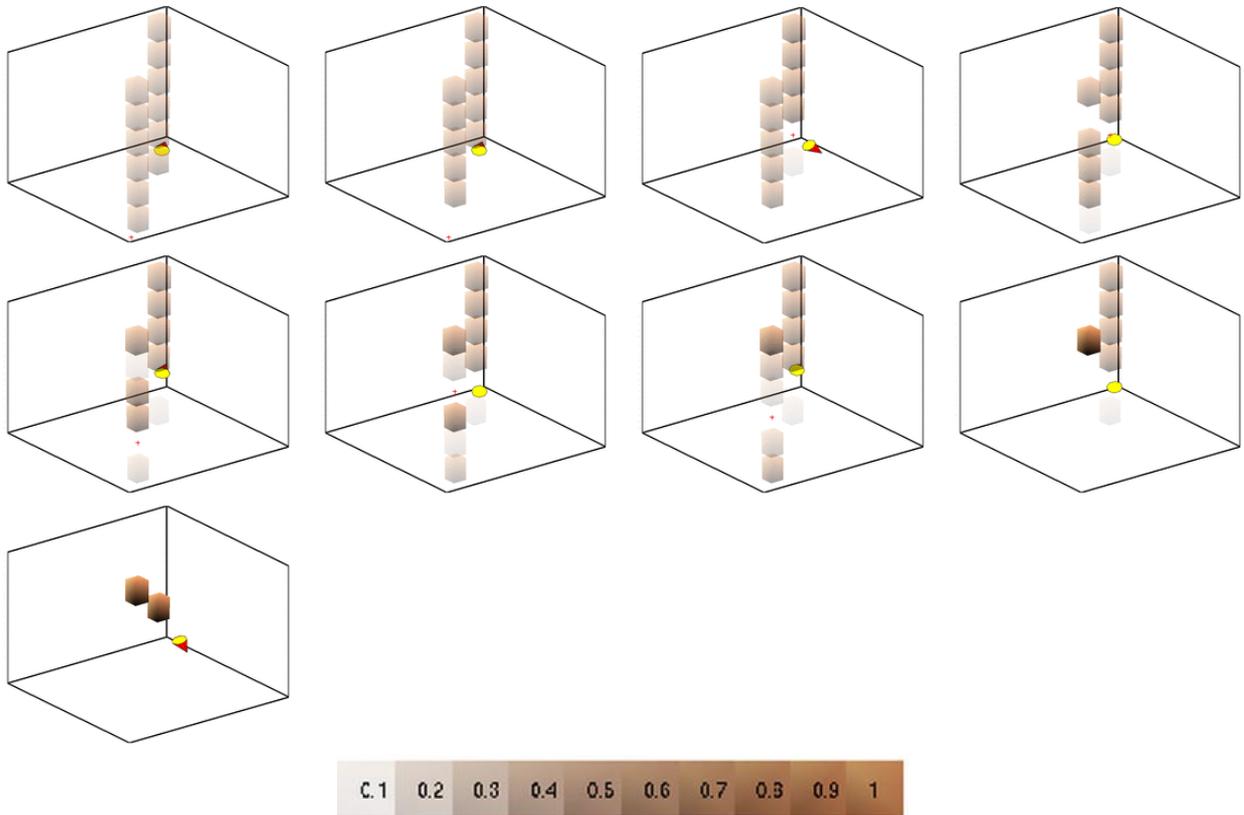


FIG. 3.6 – Stratégie basée sur la norme N_∞ . La caméra est initialement placée en position 0.

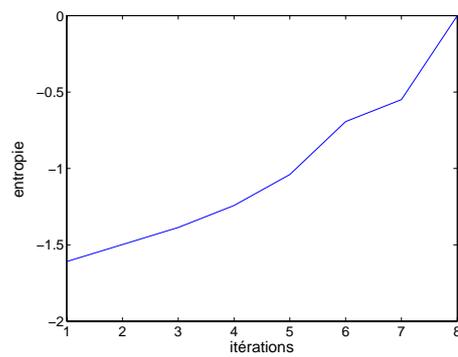


FIG. 3.7 – Evolution du critère lorsque la stratégie est basée sur la norme N_∞ et que la caméra est initialement placée en position 0.

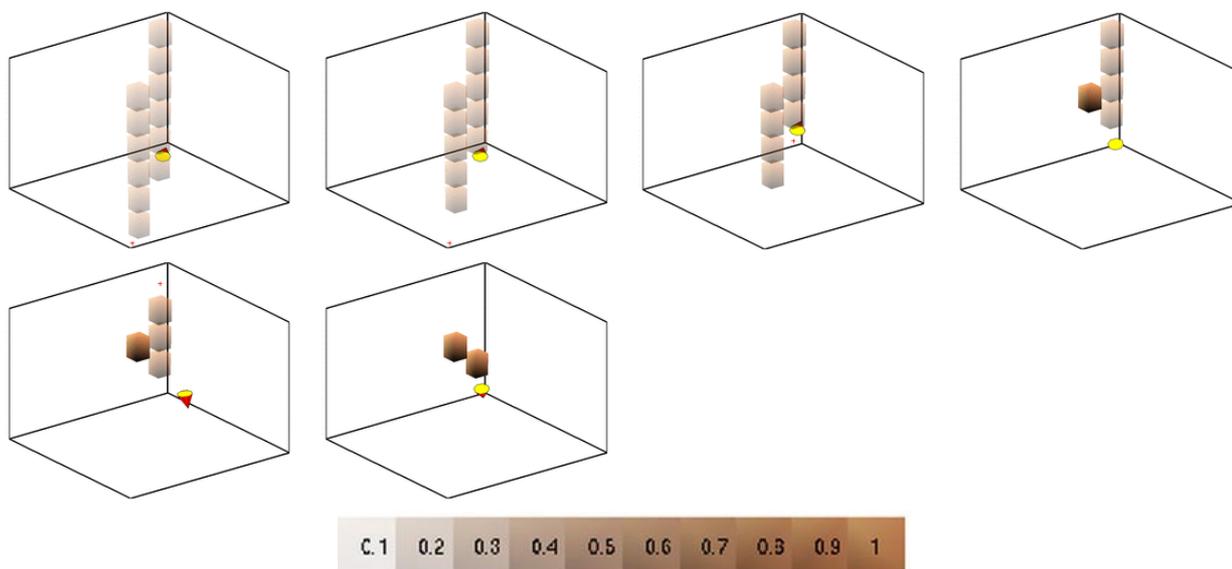


FIG. 3.8 – Stratégie basée sur la variation d'entropie. La caméra est initialement placée en position 0.

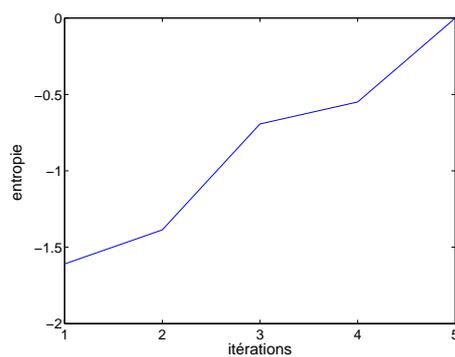


FIG. 3.9 – Evolution du critère lorsque la stratégie est basée sur la variation d'entropie et que la caméra est initialement placée en position 0.

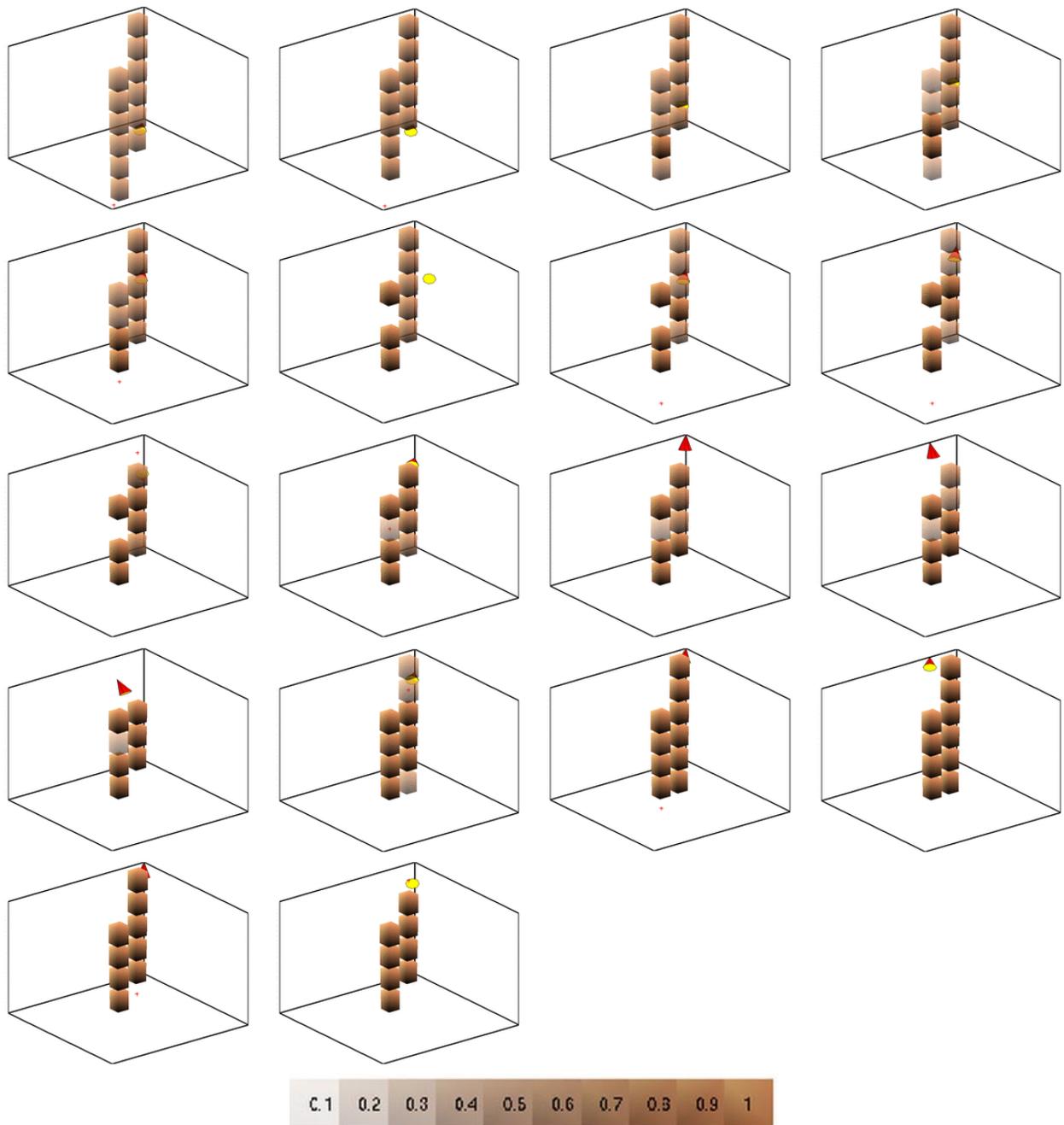


FIG. 3.10 – Stratégie basée sur la norme N_2 . L'a priori sur la scène est faux.

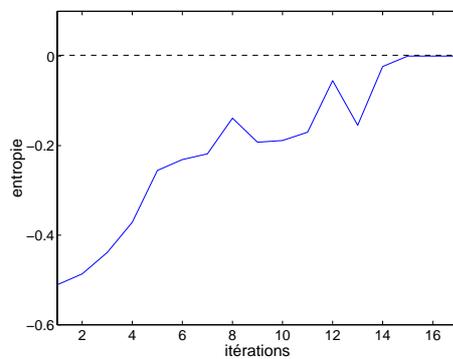


FIG. 3.11 – Evolution du critère lorsque la stratégie est basée sur la norme N_2 et que l'a priori sur la scène est faux.

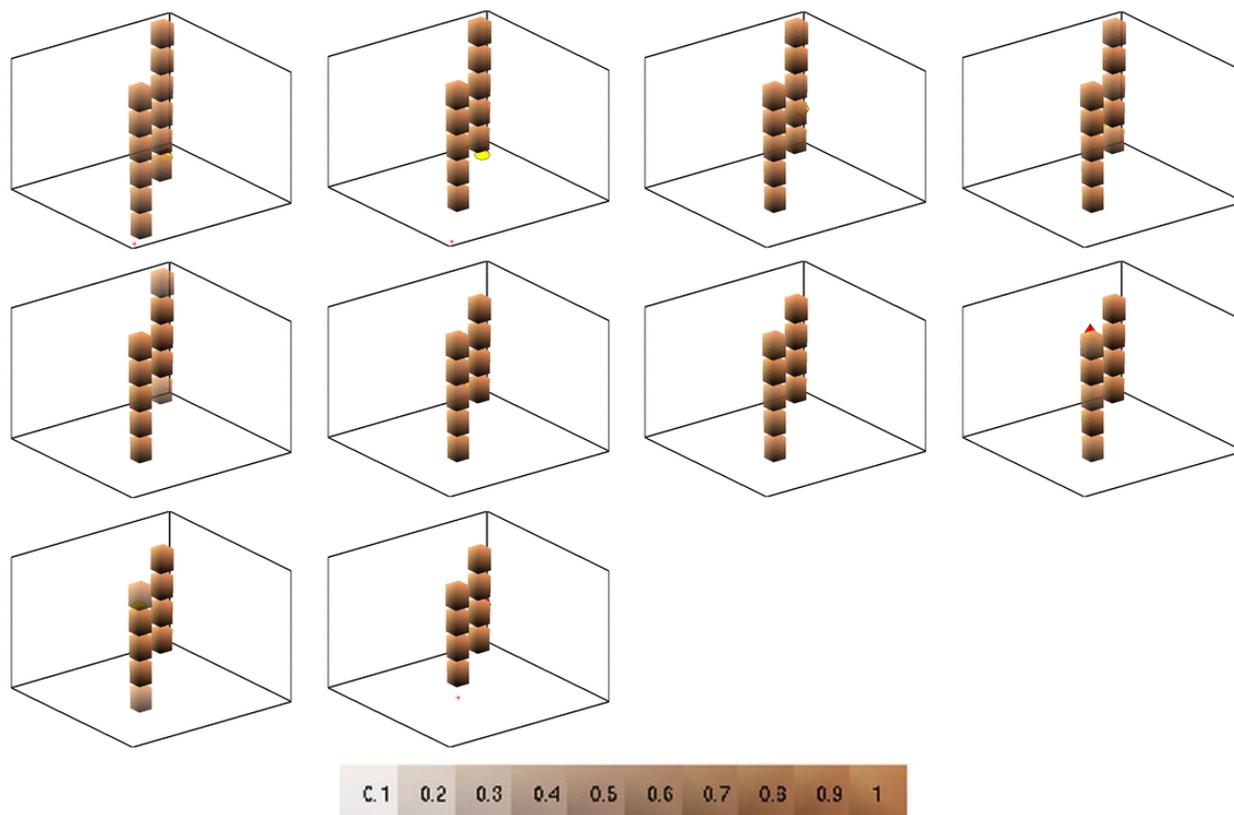


FIG. 3.12 – Stratégie basée sur la norme N_2 . L'a priori sur la scène est correct.

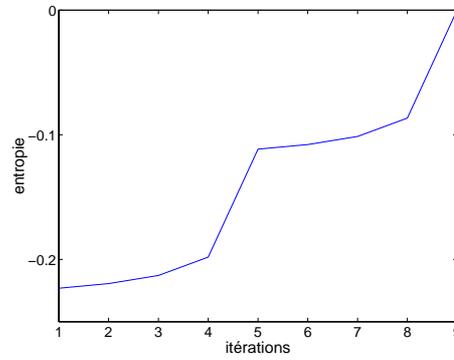


FIG. 3.13 – Evolution du critère lorsque la stratégie est basée sur la norme N_2 et que l'a priori sur la scène est correct.

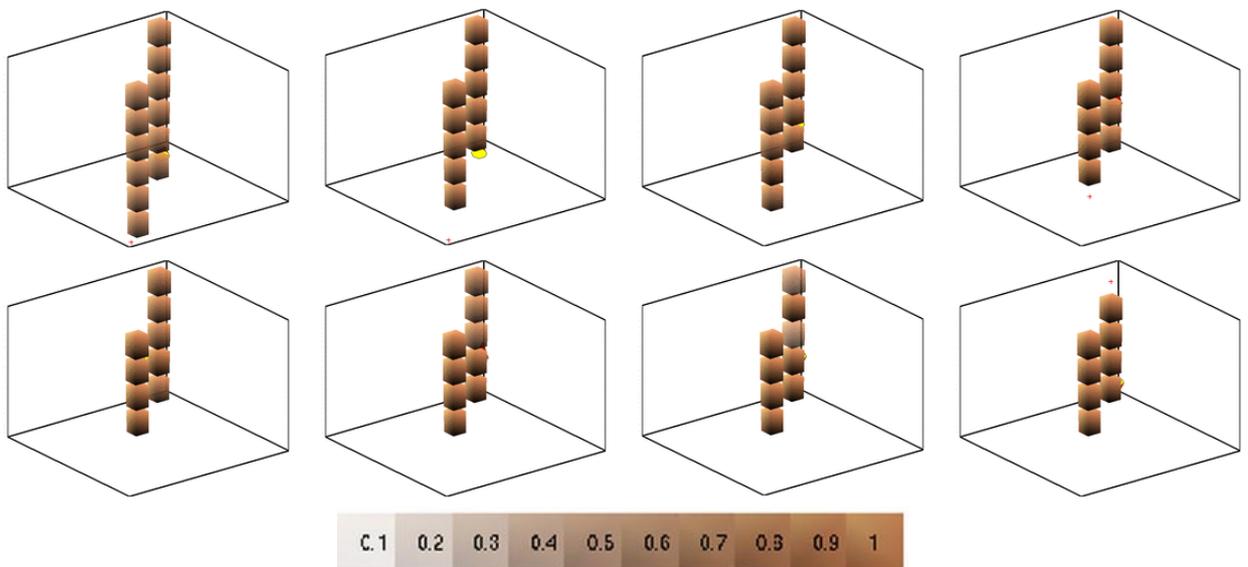


FIG. 3.14 – Stratégie basée sur la variation de l'entropie. L'a priori sur la scène est correct.

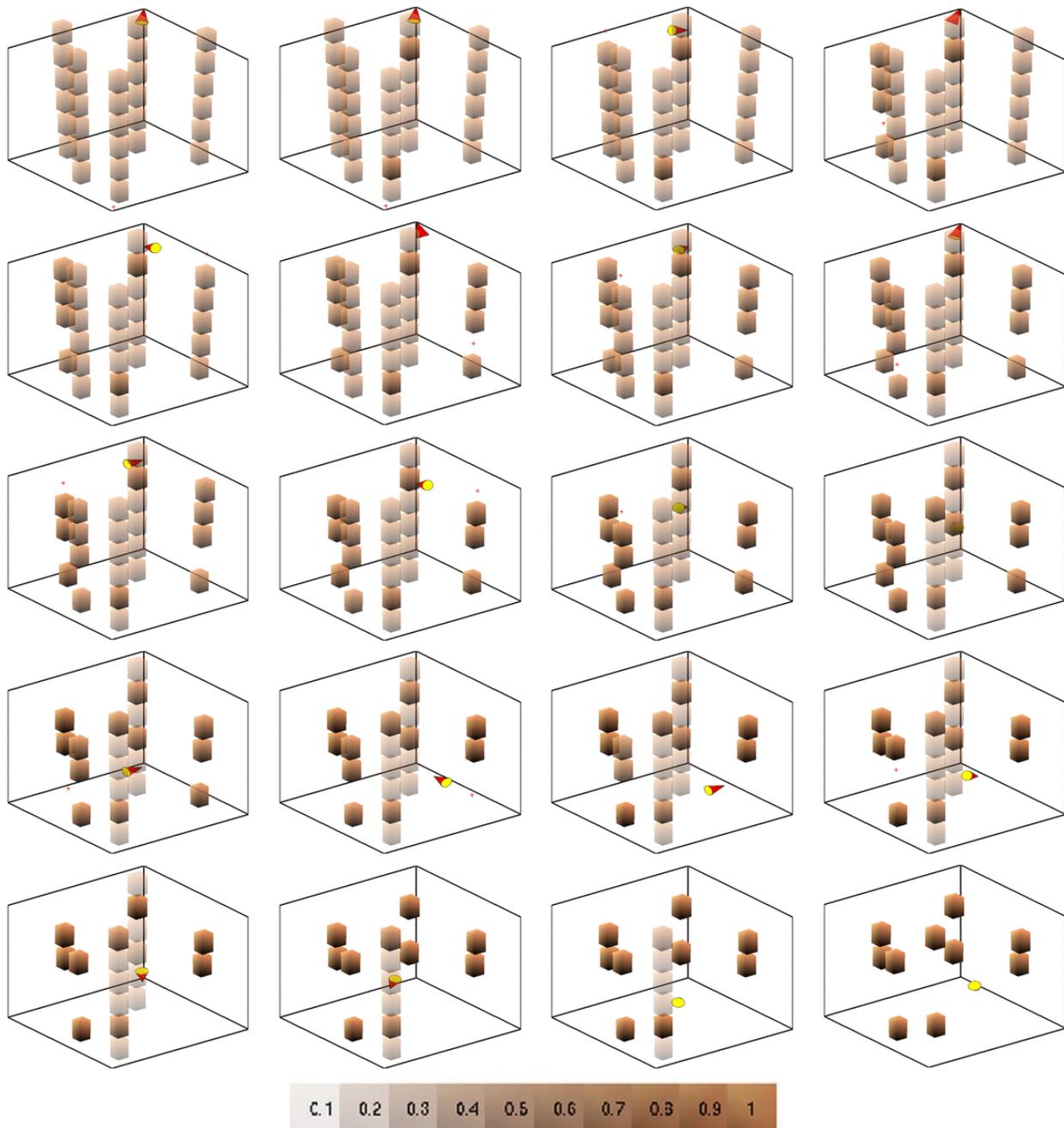


FIG. 3.15 – *Stratégie basée sur la variation de l'entropie.*

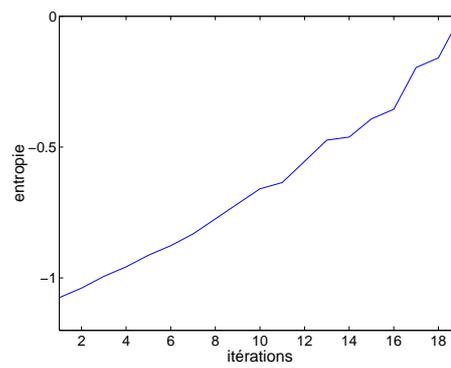


FIG. 3.16 – *Evolution du critère lorsque la stratégie est basée sur la variation de l'entropie. La scène contient 10 objets.*

Conclusion

Cette partie était consacrée à la recherche de tous les objets d'une scène. Nous avons défini les zones de recherche des objets comme étant des lignes de vue de l'image globale dont la probabilité de détection est supérieure à un certain seuil. Chacun de ces axes ayant été discrétisé en cellules, nous avons défini, pour chaque cellule, une variable aléatoire d'occupation X_i à valeur dans $\{0,1\}$. La probabilité associée définit ainsi la probabilité d'occupation d'une cellule. Afin de prendre en compte d'éventuelles dépendances entre les variables aléatoires X_i , nous avons plutôt considéré la loi conjointe du vecteur $X = (X_1, \dots, X_n)$. Ayant montré comment tout *a priori* et toute observation d'un événement se traduit en une contrainte linéaire sur cette loi conjointe, nous avons décrit un schéma d'intégration itératif des contraintes linéaires. Outre l'intégration proprement dite, nous avons décrit un certain nombre d'outils s'y rapportant. Ceux-ci concernent : la modification et la suppression d'une contrainte, l'intégration des contraintes inégalité et des tests de redondance et de compatibilité des contraintes. L'enchaînement de ces outils nous a permis de construire un algorithme général permettant d'intégrer des contraintes et de nature à gérer un maximum de situations. L'étape d'initialisation de cet algorithme consiste d'une part à traduire le fait que la solution du problème est une loi de probabilité (contrainte de normalisation), d'autre part à intégrer la probabilité de détection globale sur l'axe concerné et enfin à intégrer des *a priori* sur la scène. Pour intégrer les observations, il restait à développer un algorithme permettant de les traduire sous forme de contraintes sur les probabilités d'occupation. Enfin, étant données la connaissance du modèle et la configuration de la caméra à l'instant courant, nous avons proposé un algorithme permettant de déterminer la configuration suivante qui optimise la recherche. Le critère retenu est basé sur le calcul de l'espérance du gain d'information qui quantifie le gain apporté par une modification des probabilités d'occupation. Nous avons proposé 4 quantifications possibles : la somme des différences, la norme 2, la norme ∞ et la variation de l'entropie.

L'algorithme de recherche dans sa globalité a fait l'objet de nombreuses simulations. Celles-ci ont montré l'importance du choix de la fonction gain d'information. Les différentes stratégies ont été évaluées vis-à-vis du nombre moyen d'itérations (NIM) nécessaire à l'aboutissement de la recherche. Les stratégies basées sur la norme N_2 et sur la variation de l'entropie apportent un gain significatif par rapport à une stratégie gloutonne dont nous avons calculé le NIM théorique. Nous avons proposé un critère d'arrêt de la recherche basé sur l'entropie des lois marginales. Au cours de ces simulations, il apparaît qu'introduire un *a priori* faux est très pénalisant à la fois en ce qui concerne le critère mais également au niveau du nombre d'itérations nécessaires à la découverte de tous les objets. Mieux vaut donc ne pas fixer d'*a priori* que d'en fixer un qui soit faux.

Cette partie nous a permis de valider l'approche proposée pour la recherche des objets

dans une scène. Dans la partie suivante nous étudions le problème de l'exploration et la reconstruction individuelle de ces objets.

Deuxième partie

Reconstruction approchée et exploration des objets complexes

Rappels concernant l'approche proposée

Dans la partie précédente, nous nous sommes intéressés à la recherche des objets dans la scène. L'objectif étant, à présent, de caractériser chacun d'entre eux, nous consacrons cette partie à la reconstruction de chaque objet pris individuellement.

Le chapitre 1 concerne la modélisation d'un objet. Comme nous l'avons vu en introduction, la plupart des études passées se sont concentrées sur la modélisation 3D en termes de précision de reconstruction et considèrent donc des modèles les plus réalistes possibles. Mais, pour une large classe d'applications telles que la planification de trajectoires, l'évitement d'obstacles ou l'exploration, une carte préliminaire et simplifiée des constituants de la scène et de leur encombrement est suffisante. Une représentation concise et donc simplifiée devient même nécessaire lorsque le système est soumis à des exigences de réactivité en temps réel. Ces constatations ont motivé nos travaux sur l'estimation et la reconstruction de modèles approchés. Aussi avons nous choisi de décrire un objet par sa position et son volume englobant (la position de l'objet étant définie comme le centre de l'enveloppe englobante).

La modélisation que nous avons développée est basée sur l'utilisation d'un mélange de modèles stochastiques et à erreur bornée. Elle permet de représenter à la fois des aspects d'incertitude liée à la position de l'objet et des aspects géométriques liés à sa forme tout en conservant une relative indépendance entre les deux. D'une part, la position de l'objet et l'incertitude associée sont les paramètres d'une loi normale et, d'autre part, la forme de l'objet est associée à un modèle à erreur bornée dont le support ellipsoïdal coïncide avec son volume englobant. Ce modèle est d'autant plus intéressant qu'il permet à la fois, pour chaque point de la scène, le calcul de sa probabilité d'appartenance à un objet et une extraction simple de la géométrie de cet objet (position et enveloppe). Le modèle ainsi défini est établi sous une forme paramétrique appelée *loi d'ensemble* dans laquelle chaque paramètre correspond à une interprétation physique.

Une fois le modèle établi, il s'agit de définir des règles de propagation permettant d'obtenir les paramètres du modèle après transformation. Nous développons des règles très générales au chapitre 2. Celles-ci permettent de modéliser plus particulièrement la chaîne des observations visuelles pour différentes configurations : monoculaire ou binoculaire. Ces règles sont établies sur la base d'une approximation au premier ordre nécessaire pour assurer la consistance du modèle. En effet, sans cette linéarisation, la transformée d'une loi d'ensemble ne serait pas une loi d'ensemble. D'autre part, soulignons que cette approximation n'est absolument pas gênante dans le cadre d'une reconstruction approchée comme celle que nous visons.

Le schéma d'estimation d'une loi d'ensemble que nous décrivons au chapitre 3 adresse le problème suivant : étant donnée la connaissance à l'instant k de la loi *a priori* et étant donnée la loi décrivant l'observation à l'instant $k + 1$, il s'agit de calculer une estimation de la loi *a posteriori*. Les paramètres de position et d'incertitude sont estimés au sens du maximum *a posteriori* alors que le paramètre définissant l'enveloppe est une approximation ellipsoïdale de l'intersection de deux ellipsoïdes. Fort de ce schéma d'estimation et sous réserve d'avoir initialisé le modèle de l'objet, nous pouvons affiner ce modèle en fonction d'une suite d'observations issues, par exemple, d'un système visuel.

Le dernier problème, considéré au chapitre 4, est celui de la reconstruction autonome d'un objet, c'est-à-dire la génération des mouvements de la caméra permettant d'optimiser l'estimation des paramètres du modèle correspondant à l'objet. Dans notre cas, le concept d'exploration est vu comme un moyen d'améliorer la localisation de l'objet. La stratégie que nous développons consiste à réduire l'incertitude de la distribution associée à l'objet observé en intégrant des données visuelles. Nous présentons deux solutions analytiques au problème du calcul de la loi de commande d'exploration optimale. L'une est basée sur une optimisation localement optimale ; l'autre concerne le calcul du meilleur point de vue à atteindre.

Enfin nous validons l'ensemble de cette étude à travers des simulations et des expérimentations sur une plate-forme robotique dont les résultats sont présentés au chapitre 5.

Chapitre 1

Modélisation

La modélisation d'un objet (noté \mathcal{O}) introduit évidemment des aspects géométriques puisque \mathcal{O} est entièrement représenté par sa position et sa forme. Mais pouvoir extraire la position et la forme d'un modèle dont on souhaite qu'il prenne en compte des incertitudes, n'est pas suffisant. En effet, en présence d'incertitudes, comment affirmer qu'un point de la scène (notée \mathcal{S}) est occupé par \mathcal{O} ? Il apparaît plus raisonnable d'attribuer à chaque point de \mathcal{S} une probabilité d'appartenance à \mathcal{O} . La construction du modèle d'objet que nous présentons dans ce chapitre vise à unifier deux aspects : un aspect géométrique "confortable" spécifiant la position et la forme de l'objet et un aspect probabiliste "inévitabile" permettant de calculer, pour tout point, sa probabilité d'appartenance à \mathcal{O} .

- La section 1.1 est consacrée à la construction d'un modèle probabiliste appelé *loi d'ensemble* intégrant les aspects que nous venons d'énoncer.
- La section 1.2 vise à décrire les observations visuelles au moyen des lois d'ensemble.

1.1 Modélisation des objets : la loi d'ensemble

Pour tout objet \mathcal{O} d'une scène \mathcal{S} et pour tout point x de \mathcal{S} , nous souhaitons calculer la probabilité que x appartienne à \mathcal{O} notée $\mathcal{P}(x \in \mathcal{O})$. Introduisons le vecteur aléatoire des coordonnées d'un point c de \mathcal{O} dont la distribution est $\mathcal{P}(c = x) = \mathcal{P}_c(x)$ pour tout $x \in \mathcal{S}$ et montrons que, de cette distribution, nous pouvons déduire $\mathcal{P}(x \in \mathcal{O})$. En effet, en décomposant l'événement $\{c = x\}$ sur la partition constituée des événements $\{x \in \mathcal{O}\}$ et $\{x \notin \mathcal{O}\}$, nous pouvons écrire :

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

Dans cette relation :

- $\mathcal{P}_c(x|x \notin \mathcal{O})$ est la probabilité que le point c de \mathcal{O} soit en x sachant que $x \notin \mathcal{O}$. $\mathcal{P}_c(x|x \notin \mathcal{O})$ vaut donc 0, par définition de c .
- $\mathcal{P}_c(x|x \in \mathcal{O})$ est la probabilité que le point c de \mathcal{O} soit en x sachant que $x \in \mathcal{O}$. Nous modélisons naturellement $\mathcal{P}_c(x|x \in \mathcal{O})$ par une loi uniforme dont la valeur se calcule, après normalisation et pour tout \mathcal{O} , de la façon suivante :

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

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

En général, $Volume(\mathcal{O})$ n'est pas connu avant la fin de la reconstruction mais l'important est qu'il soit constant.

En résumé des équations précédentes :

$$\mathcal{P}_c(x) = \frac{\mathcal{P}(x \in \mathcal{O})}{Volume(\mathcal{O})} \quad (1.1)$$

Modéliser \mathcal{S} revient donc à chercher, pour tout \mathcal{O} , une distribution modélisant efficacement la densité de $\mathcal{P}_c(x)$. Dans ce but et afin d'intégrer au mieux les aspects géométriques de l'objet, nous décomposons c en la somme d'une constante \bar{c} et de deux vecteurs aléatoires centrés et indépendants (voir figure 1.1) :

$$c = \bar{c} + p + e$$

Ce qui signifie que \bar{c} est la moyenne des points c de \mathcal{O} . \bar{c} est donc naturellement choisi pour représenter la **position** de l'objet. D'autre part, p représente l'**incertitude** de localisation de l'objet et le support de l'**erreur** e définit son volume. Cette modélisation nous conduit à faire la distinction suivante : une incertitude appartient à la classe des modèles probabilistes alors qu'une erreur appartient à la classe des modèles à erreur bornée. L'erreur est donc, par définition, uniformément distribuée sur un domaine borné. Cette décomposition additive de c est à la base de l'unification des aspects géométriques représentés par \bar{c} (position de l'objet) et e (dont le support représente la forme approchée de l'objet) et de l'aspect probabiliste représenté par p (incertitude de localisation). Le choix de cette décomposition additive correspond au désir de découpler la description de l'enveloppe englobante des incertitudes de modélisation. L'inconvénient de ce découplage est que l'incertitude sur l'enveloppe n'est pas explicitement modélisée, même s'il est possible d'en tenir compte par l'intermédiaire de l'incertitude de localisation p . En revanche cette décomposition s'avère très intéressante pour propager le modèle qui apparaît comme l'intégration d'un modèle probabiliste et d'un modèle à erreur bornée.

Hypothèses d'étude : Nous décrivons à présent les hypothèses sous-jacentes à l'étude :

1. en premier lieu, nous supposons que p suit une loi normale centrée de covariance P . C'est une approximation satisfaisante pour de nombreuses sources d'incertitudes telles que la mesure dans l'image ou la localisation de la caméra. L'hypothèse gaussienne présente également l'avantage de rendre la propagation des incertitudes calculable analytiquement. Afin de pouvoir tenir compte de l'observabilité partielle des capteurs visuels, nous préférons, par la suite, utiliser la **matrice d'information**¹ $\Sigma = P^{-1}$ plutôt que la matrice de covariance P . Une variance infinie le long des axes d'observabilité est ainsi remplacée par une information nulle. La distribution

1. Le terme matrice d'information provient de l'analogie avec la matrice d'information de Fisher I_F en théorie de l'estimation :

$$I_F = E\left[\left(\frac{\partial}{\partial x} \ln \mathcal{P}(z, x)\right)^T \left(\frac{\partial}{\partial x} \ln \mathcal{P}(z, x)\right)\right]$$

où x est l'état à estimer, z l'observation et $\mathcal{P}(z, x)$ la probabilité jointe associée. Il se trouve que, lorsque \mathcal{P} suit une distribution gaussienne, I_F est l'inverse de la matrice de covariance associée à \mathcal{P} .

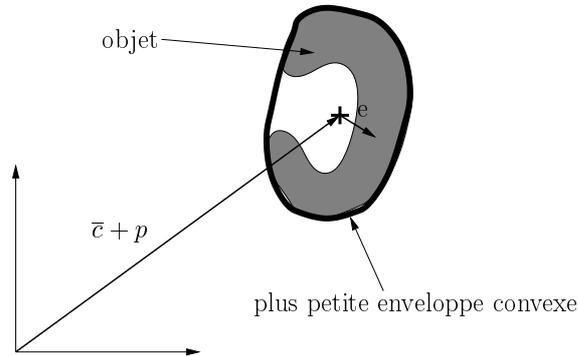


FIG. 1.1 – Le modèle additif unifie l’aspect géométrique représenté par \bar{c} (position de l’objet) et e (dont le support représente la forme de l’objet) et l’aspect probabiliste représenté par p (incertitude de localisation).

associée à p est finalement² :

$$\mathcal{N}(0, \Sigma^{-1}) \rightsquigarrow \frac{\sqrt{\det \Sigma}}{(2\pi)^{3/2}} e^{-\frac{1}{2}x^T \Sigma x}$$

- En accord avec notre approche *reconstruction approchée*, et dans la mesure où la représentation mathématique du support de l’erreur e doit être simple, sans quoi sa manipulation devient prohibitive, nous approchons ce support par un ellipsoïde. Nous rappelons qu’un ellipsoïde est complètement déterminé par sa forme quadratique : son centre et une matrice symétrique définie positive E . Dans la suite et par abus de langage, nous confondrons l’ellipsoïde et sa matrice E , son centre étant représenté par \bar{c} (aux incertitudes p près).

La distribution associée à un objet est donc entièrement déterminée par trois paramètres : \bar{c} , Σ et E . Plus précisément, il s’agit de la distribution de la somme de deux variables aléatoires indépendantes, c’est-à-dire le produit de convolution entre la loi uniforme \mathcal{U}_E et la loi normale $\mathcal{N}(\bar{c}, \Sigma^{-1})$:

$$\mathcal{P}_{\bar{c}+p+e}(x) = \int_{\mathbf{e}} \mathcal{P}_{\bar{c}+p}(x - \mathbf{e}) \mathcal{P}_e(\mathbf{e}) = (\mathcal{P}_{\bar{c}+p} * \mathcal{P}_e)(x)$$

Nous appelons cette distribution loi d’ensemble et la notons :

$$\mathcal{E}(\bar{c}, \Sigma, E) = \mathcal{N}(\bar{c}, \Sigma^{-1}) * \mathcal{U}_E \quad (1.2)$$

La forme d’une distribution d’ensemble est donnée à titre indicatif figure 1.2 dans le cas 2D. Elle correspond à un étalement du support de e selon la loi de p .

En résumé, d’après l’équation (1.1) et pour tout objet \mathcal{O} , $\mathcal{P}(x \in \mathcal{O})$ est proportionnel à une loi d’ensemble $\mathcal{E}(\bar{c}, \Sigma, E)$. Les travaux que nous présentons dans la suite concernent l’estimation des paramètres (\bar{c}, Σ, E) la constituant.

². Nous utiliserons abusivement la notation Σ^{-1} pour parler de la covariance P même si Σ n’est pas inversible.

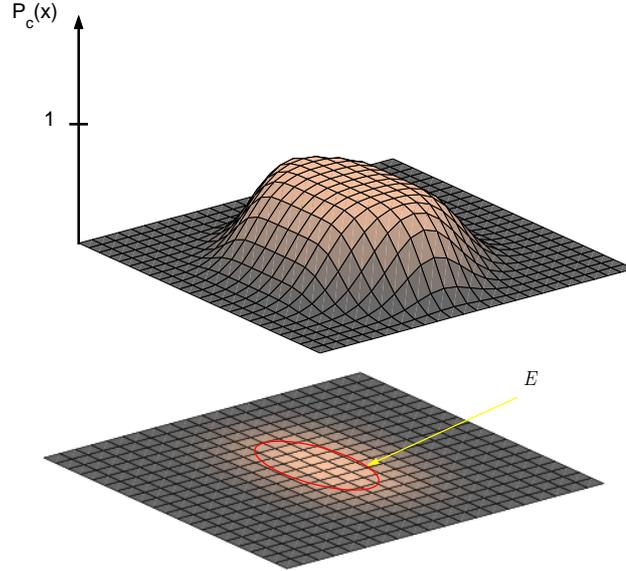


FIG. 1.2 – La loi d'ensemble (représentée ici en dimension deux) correspond à un étalement du support de e selon la loi de p .

1.2 Modélisation des observations

Dans le but d'appliquer le modèle *loi d'ensemble* au cas particulier des données visuelles, nous consacrons cette section à la description de la chaîne des observations visuelles. La figure 1.3 représente trois repères dans lesquels nous pouvons exprimer les observations. Le repère \mathcal{R}^i de dimension 2 est lié à l'image et repère la projection d'un point 3D sur le plan image. Le repère \mathcal{R}^c est le repère de la caméra, d'origine le centre de la caméra et dont deux de ses axes sont parallèles à ceux de \mathcal{R}^i . Il dispose d'une dimension supplémentaire parallèle à l'axe optique et déterminant la distance de la caméra à un point 3D. \mathcal{R}^o est un repère 3D fixe, indépendant de la position de la caméra. Il est essentiel pour comparer et fusionner les informations visuelles. Dans la chaîne des observations visuelles, nous distinguons trois étapes (voir figure 1.4) :

1. Dans l'image, la mesure est une loi d'ensemble 2D exprimée dans \mathcal{R}^i : $\mathcal{E}^i(\bar{c}^i, \Sigma^i, E^i)$ où \bar{c}^i et E^i représentent le centre et la matrice de la plus petite ellipse englobant la projection de l'objet. Pour pouvoir les déterminer, nous considérons implicitement que la projection de l'objet dans l'image peut être identifiée et extraite. Si cette phase est la source de nombreux travaux, elle n'est pas au cœur de cette étude. Néanmoins, elle s'avère nécessaire lors de la phase expérimentale et joue un rôle important dans l'efficacité de la méthode. Nous décrivons au chapitre 5 comment extraire le masque des objets sans imposer de contraintes sur leur apparence. Σ^i doit, pour sa part, prendre en compte toutes les sources d'incertitudes inhérentes au calcul de l'ellipse : erreur sur les paramètres intrinsèques de la caméra et imprécision du traitement d'image (voir section 2.2.1).
2. L'opération permettant de passer des mesures 2D aux observations 3D est appelée rétro-projection. La loi d'ensemble 3D associée est notée $\mathcal{E}^c(\bar{c}^c, \Sigma^c, E^c)$. L'exposant

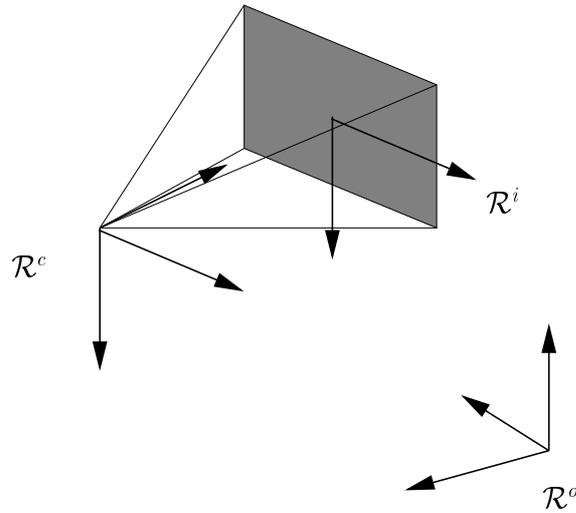


FIG. 1.3 – Le repère \mathcal{R}^i de dimension 2 est lié à l'image et repère la projection d'un point 3D sur le plan image. Le repère \mathcal{R}^c est le repère 3D de la caméra, d'origine le centre de la caméra et dont deux de ses axes sont parallèles à ceux de \mathcal{R}^i . \mathcal{R}^o est un repère 3D fixe, indépendant de la position de la caméra.

c signifie que \mathcal{E}^c est exprimée dans le repère \mathcal{R}^c . Cette étape nous amène à faire la distinction entre une **mesure** que nous définissons comme étant 2D (extraite de l'image) et une **observation** qui est définie dans l'espace 3D. Contrairement à la mesure, l'observation est homogène au modèle de l'objet. Bien entendu, \mathcal{E}^c est une loi d'ensemble dégénérée puisqu'elle est issue d'une mesure 2D donc partiellement observable. En d'autres termes, Σ^c et E^c ont des valeurs propres nulles indiquant que l'information dans la direction du vecteur propre correspondant (parallèle à la droite passant par l'axe optique de la caméra et le centre de l'objet) est nulle (voir section 2.2.2).

3. À ce stade, les observations sont homogènes au modèle de l'objet car exprimées en dimension trois. Pour comparer et fusionner des observations relatives à des instants différents (après déplacement de la caméra), il nous reste à les exprimer dans un repère commun. Nous choisissons naturellement le repère \mathcal{R}^o appelé repère de référence. La loi d'ensemble associée est notée $\mathcal{E}^o(\overline{c^o}, \Sigma^o, E^o)$.

Ces trois étapes sont indispensables pour comparer et fusionner les informations visuelles extraites des images. Avant de décrire plus précisément chacune de ces étapes, nous devons définir des règles de propagation des lois d'ensemble, ce qui est fait dans le chapitre suivant.

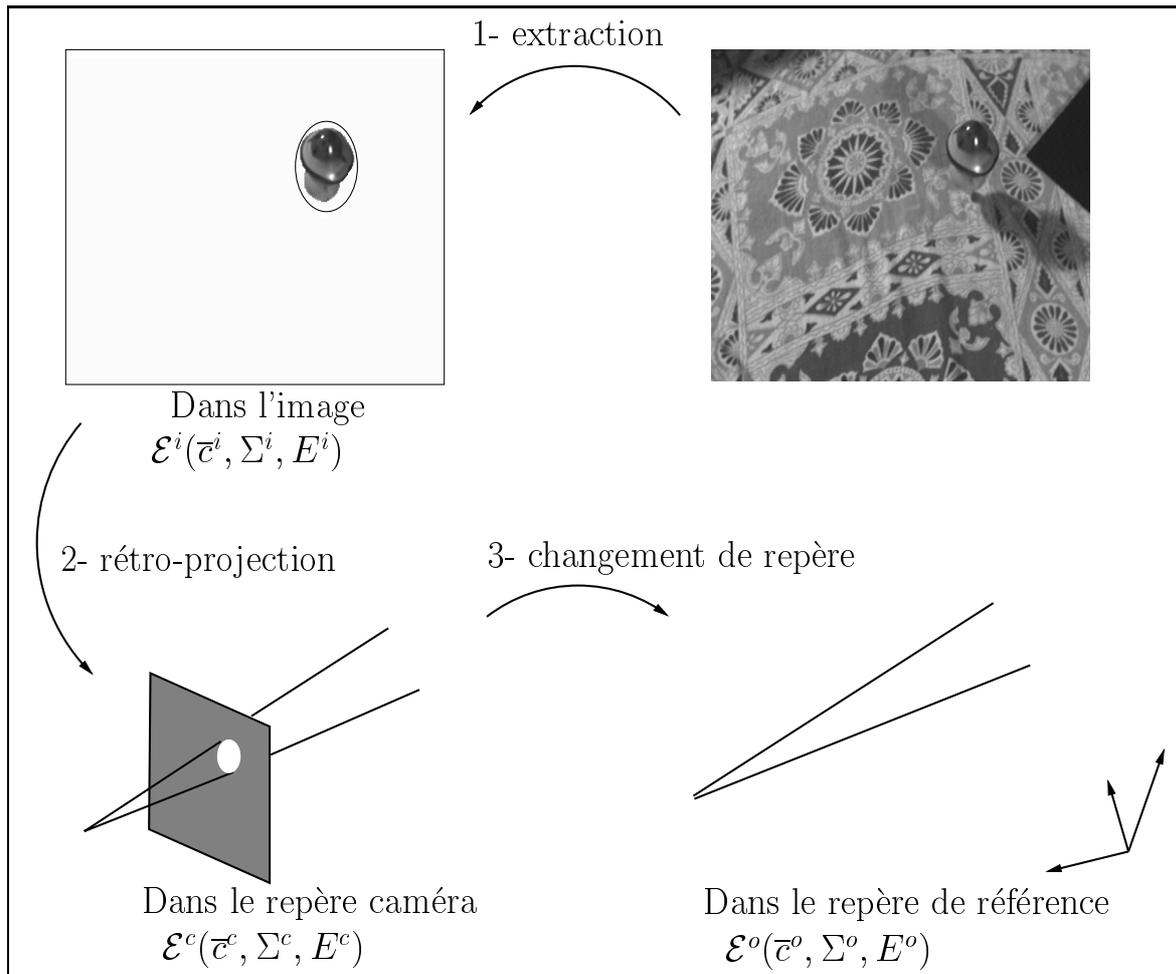


FIG. 1.4 – Transformations successives des données visuelles 1-Extraction de la projection de l'objet dans l'image et calcul de la loi 2D associée. 2-Rétro-projection de la loi 2D dans l'espace 3D. 3-Expression de l'observation dans le repère de référence.

Chapitre 2

Propagation

Dans le chapitre précédent, nous avons défini un modèle paramétrique sous forme d'une distribution de probabilité appelée loi d'ensemble. Le problème qui se pose naturellement est alors de déterminer l'influence d'une transformation sur cette distribution afin de pouvoir la propager. Dans ce chapitre, nous définissons un certain nombre de règles se rapportant à la propagation d'une loi d'ensemble :

- la section 2.1 est consacrée à la démonstration de règles générales applicables à divers types de transformations : les difféomorphismes en présence d'incertitudes ou non, et les projections sur un sous-espace ;
- en section 2.2, nous spécialisons ces règles à la propagation des observations visuelles. D'après la section 1.2, nous devons dériver une règle de rétro-projection qui, nous le verrons, dépend de la configuration utilisée (monoculaire ou binoculaire) puis une règle de changement de repère. De plus, en prévision du calcul de la commande exploratoire, nous démontrons une règle de projection perspective.

2.1 Transformations générales

La première règle est consacrée aux transformations qui conservent la dimension de l'espace des paramètres et la complétude de leur représentation : *les difféomorphismes*. Nous rappelons qu'un difféomorphisme est une application bijective continûment différentiable et d'inverse également continûment différentiable.

Règle 1 (Difféomorphisme d'une loi d'ensemble)

Soit T un difféomorphisme et c un vecteur aléatoire de loi $\mathcal{E}(\bar{c}, \Sigma, E)$ sur l'espace de définition de T . Si l'on note $J = \left. \frac{\partial T^{-1}}{\partial c} \right|_{\bar{c}}$ le jacobien de la transformation inverse, en première approximation, le vecteur aléatoire $c' = T(c)$ suit une loi $\mathcal{E}'(\bar{c}', \Sigma', E')$ sur l'espace image de T où :

$$\begin{cases} \bar{c}' &= T(\bar{c}) \\ \Sigma' &= J^T \Sigma J \\ E' &= J^T E J \end{cases}$$

Preuve :

La démonstration est basée sur une approximation de T au premier ordre :

$$c' = T(c) \approx T(\bar{c}) + \left. \frac{\partial T}{\partial c} \right|_{\bar{c}} (c - \bar{c}) = \underbrace{T(\bar{c})}_{\text{constante } \bar{c}} + \underbrace{\left. \frac{\partial T}{\partial c} \right|_{\bar{c}} p}_{\text{incertitude } p'} + \underbrace{\left. \frac{\partial T}{\partial c} \right|_{\bar{c}} e}_{\text{erreur } e'}$$

Puisque $T(\bar{c})$ est un terme constant, $\left. \frac{\partial T}{\partial c} \right|_{\bar{c}} p$ un vecteur aléatoire gaussien centré et $\left. \frac{\partial T}{\partial c} \right|_{\bar{c}} e$ défini sur un ellipsoïde également centré, nous identifions terme à terme :

$$\bar{c}' = T(\bar{c}), \quad p' = \left. \frac{\partial T}{\partial c} \right|_{\bar{c}} p, \quad e' = \left. \frac{\partial T}{\partial c} \right|_{\bar{c}} e$$

D'autre part, $p' = \left. \frac{\partial T}{\partial c} \right|_{\bar{c}} p$ suit une loi normale de covariance P' :

$$\begin{aligned} P' &= E[p'p'^T] \\ &= E\left[\left. \frac{\partial T}{\partial c} \right|_{\bar{c}} pp^T \left. \frac{\partial T}{\partial c} \right|_{\bar{c}}^T\right] \\ &= \left. \frac{\partial T}{\partial c} \right|_{\bar{c}} E[pp^T] \left. \frac{\partial T}{\partial c} \right|_{\bar{c}}^T \\ &= \left. \frac{\partial T}{\partial c} \right|_{\bar{c}} P \left. \frac{\partial T}{\partial c} \right|_{\bar{c}}^T \end{aligned}$$

T étant un difféomorphisme, cette dernière relation s'inverse en :

$$\Sigma' = \left. \frac{\partial T^{-1}}{\partial c} \right|_{\bar{c}}^T \Sigma \left. \frac{\partial T^{-1}}{\partial c} \right|_{\bar{c}}$$

Enfin $e' = \left. \frac{\partial T}{\partial c} \right|_{\bar{c}} e$ suit une loi uniforme sur un ellipsoïde E' . Le support de e est défini par :

$$\begin{aligned} e^T E e = 1 &\Leftrightarrow \left[\left. \frac{\partial T^{-1}}{\partial c} \right|_{\bar{c}} e' \right]^T E \left[\left. \frac{\partial T^{-1}}{\partial c} \right|_{\bar{c}} e' \right] = 1 \\ &\Leftrightarrow e'^T \left. \frac{\partial T^{-1}}{\partial c} \right|_{\bar{c}}^T E \left. \frac{\partial T^{-1}}{\partial c} \right|_{\bar{c}} e' = 1 \end{aligned}$$

En conséquence :

$$E' = \left. \frac{\partial T^{-1}}{\partial c} \right|_{\bar{c}}^T E \left. \frac{\partial T^{-1}}{\partial c} \right|_{\bar{c}}$$

ce qui termine la démonstration. □

Lorsque T dépend également d'autres paramètres tels que la position de la caméra : $c' = T(c, \gamma)$ (où γ suit une loi $\mathcal{N}(\bar{\gamma}, \Gamma)$), il est possible de linéariser T autour de \bar{c} et $\bar{\gamma}$ afin de prendre en compte de possibles incertitudes sur ces paramètres externes (telles qu'une incertitude sur la position ou la calibration de la caméra). L'approximation de T au premier ordre devient :

$$\begin{aligned} c' = T(c, \gamma) &\approx T(\bar{c}, \bar{\gamma}) + \left. \frac{\partial T}{\partial c} \right|_{\bar{c}, \bar{\gamma}} (c - \bar{c}) + \left. \frac{\partial T}{\partial \gamma} \right|_{\bar{c}, \bar{\gamma}} (\gamma - \bar{\gamma}) \\ &= T(\bar{c}, \bar{\gamma}) + \left. \frac{\partial T}{\partial c} \right|_{\bar{c}, \bar{\gamma}} p + \left. \frac{\partial T}{\partial \gamma} \right|_{\bar{c}, \bar{\gamma}} (\gamma - \bar{\gamma}) + \left. \frac{\partial T}{\partial c} \right|_{\bar{c}, \bar{\gamma}} e \end{aligned}$$

Comme précédemment, nous identifions terme à terme :

$$\bar{c}' = T(\bar{c}, \bar{\gamma}), \quad p' = \left. \frac{\partial T}{\partial c} \right|_{\bar{c}, \bar{\gamma}} p + \left. \frac{\partial T}{\partial \gamma} \right|_{\bar{c}, \bar{\gamma}} (\gamma - \bar{\gamma}), \quad e' = \left. \frac{\partial T}{\partial c} \right|_{\bar{c}, \bar{\gamma}} e$$

Dans ce cas, l'erreur e' demeure inchangée par rapport au cas précédent mais l'incertitude

$$p' = \left. \frac{\partial T}{\partial c} \right|_{\bar{c}, \bar{\gamma}} p + \left. \frac{\partial T}{\partial \gamma} \right|_{\bar{c}, \bar{\gamma}} (\gamma - \bar{\gamma})$$

suit une loi normale de covariance :

$$P' = \left. \frac{\partial T}{\partial c} \right|_{\bar{c}, \bar{\gamma}} P \left. \frac{\partial T}{\partial c} \right|_{\bar{c}, \bar{\gamma}}^T + \left. \frac{\partial T}{\partial \gamma} \right|_{\bar{c}, \bar{\gamma}} \Gamma \left. \frac{\partial T}{\partial \gamma} \right|_{\bar{c}, \bar{\gamma}}^T \quad (2.1)$$

en considérant que p et γ sont indépendants. De plus, en utilisant le fait que :

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

la relation (2.1) s'inverse en :

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

$$\text{où } J_c = \left. \frac{\partial T^{-1}}{\partial c} \right|_{\bar{c}, \bar{\gamma}} \quad \text{et} \quad J_\gamma = \left. \frac{\partial T}{\partial \gamma} \right|_{\bar{c}, \bar{\gamma}}$$

$\left. \frac{\partial T}{\partial \gamma} \right|_{\bar{c}, \bar{\gamma}}$ n'est pas inversé dans l'expression de J_γ car, si T est bien un difféomorphisme vis-à-vis de c , il n'a aucune raison de l'être vis-à-vis de γ . En conséquence, en présence d'incertitudes, la règle 1 devient :

Règle 2 (Difféomorphisme d'une loi d'ensemble en présence d'incertitudes)

Soit c un vecteur aléatoire suivant une loi $\mathcal{E}(\bar{c}, \Sigma, E)$. En première approximation, le vecteur aléatoire $c' = T(c, \gamma)$, où γ est $\mathcal{N}(\bar{\gamma}, \Gamma)$, suit une loi d'ensemble $\mathcal{E}'(\bar{c}', \Sigma', E')$ où :

$$\begin{cases} \bar{c}' &= T(\bar{c}, \bar{\gamma}) \\ \Sigma' &= (J_\gamma \Gamma J_\gamma^T)^{-1} ((J_\gamma \Gamma J_\gamma^T)^{-1} + J_c^T \Sigma^c J_c)^{-1} J_c^T \Sigma^c J_c \\ E' &= J_c^T E J_c \end{cases}$$

Les deux règles précédentes requièrent que T soit un difféomorphisme par rapport à c . Lorsque nous avons à faire à une transformation qui ne conserve pas la complétude de la représentation, ce qui est typiquement le cas des observations visuelles du fait de leur nature projective, une règle de projection sur un sous-espace est nécessaire :

Règle 3 (Projection sur un sous-espace)

Soit c un vecteur aléatoire se décomposant en $c = (c_1, c_2)^T$. Si c suit une loi

$$\mathcal{E}\left(\begin{pmatrix} \bar{c}_1 \\ \bar{c}_2 \end{pmatrix}, \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{12}^T & \Sigma_{22} \end{pmatrix}, \begin{pmatrix} E_{11} & E_{12} \\ E_{12}^T & E_{22} \end{pmatrix}\right)$$

le vecteur aléatoire c_1 , projection de c sur un sous-espace, suit une loi d'ensemble $\mathcal{E}'(\bar{c}', \Sigma', E')$ où :

$$\begin{cases} \bar{c}' &= \bar{c}_1 \\ \Sigma' &= \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^T \\ E' &= E_{11} - E_{12}E_{22}^{-1}E_{12}^T \end{cases}$$

Preuve :

La démonstration de l'expression de Σ' provient directement de la formule d'inversion par blocs :

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

qui est la covariance du vecteur gaussien (p_1, p_2) . Par conséquent, la covariance de p_1 est le bloc du haut à gauche :

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

dont l'inverse est

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

D'autre part, le support de l'erreur est défini par la relation :

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

D'autre part, tout point $e' = e_1$ appartenant à la frontière de la projection du support de l'erreur est tel que le gradient de f au point (e', e_2) est parallèle au sous-espace de e_1 . Le gradient étant défini par :

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

les points e' satisfont :

$$E_{12}^T e' + E_{22} e_2 = 0 \tag{2.2}$$

De plus

$$\begin{aligned} (e', e_2)E(e', e_2)^T = 1 &\Leftrightarrow e'^T E_{11} e' + e_2^T (E_{22} e_2 + E_{12}^T e') + e'^T E_{12} e_2 = 1 \\ &\Leftrightarrow e'^T E_{11} e' - e'^T E_{12} E_{22}^{-1} E_{12}^T e' = 1 \text{ grâce à l'équation (2.2)} \\ &\Leftrightarrow e'^T (E_{11} e' - E_{12} E_{22}^{-1} E_{12}^T e') = 1 \end{aligned}$$

ce qui clôt la démonstration de la règle 3. □

2.2 Propagation des données visuelles

Par application directe des règles précédentes, nous pouvons déduire d'autres règles dédiées à la propagation d'informations visuelles. Au vu de la chaîne des données visuelles décrite en section 1.2, nous dérivons des règles spécialisées pour le **rétro-projection** et le **changement de repère**. Mais nous présentons avant tout l'étape indispensable consistant à extraire les données dans l'image.

2.2.1 Extraction des données visuelles

La première étape consistant à extraire la projection des objets dans l'image sera étudiée au cours de la section 5 consacrée aux expérimentations. Disposant du masque des objets dans l'image, nous pouvons envisager l'extraction des données qui nous intéressent, c'est-à-dire les paramètres constituant la loi $\mathcal{E}^i(\bar{c}^i, \Sigma^i, E^i)$.

Pour chaque objet, \bar{c}^i et E^i représentent le centre et la matrice définissant le plus petit ellipsoïde englobant sa projection. Ils peuvent s'obtenir au moyen d'algorithmes tels que celui proposé dans [Welzl 91]. D'autre part, Σ^i doit tenir compte de toutes les sources d'incertitude pouvant dégrader le calcul de \bar{c}^i et E^i : incertitudes sur les paramètres intrinsèques et sur le traitement d'image.

Nous rappelons tout d'abord la transformation mètres/pixels nécessaire au calcul de Σ^i . Si l'on note $\bar{c}^i = (X^i, Y^i)^T$ les coordonnées en mètres d'un point dans l'image, elles sont reliées aux coordonnées en pixels $c^p = (X^p, Y^p)^T$ par la relation :

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

où $l_x = d_x/f$ et $l_y = d_y/f$ (en notant d_x et d_y la largeur et la hauteur d'un pixel et f la focale) sont exprimés en mètres et (q_x, q_y) sont les coordonnées (en pixels) du point principal. Si l'on note $p_{int} = (q_x, q_y, l_x, l_y)^T$ le vecteur des paramètres intrinsèques, la relation précédente s'écrit :

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

en supposant que c^p et p_{int} suivent respectivement les lois $\mathcal{N}(\bar{c}^p, P^p)$ et $\mathcal{N}(\bar{p}_{int}, P_{int})$. Alors en utilisant le résultat de l'équation (2.1), nous pouvons écrire :

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

où

$$\begin{cases} J_{c^p} = \frac{\partial T}{\partial c^p} \Big|_{\bar{c}^p, \bar{p}_{int}} = \begin{pmatrix} \bar{l}_x & 0 \\ 0 & \bar{l}_y \end{pmatrix} \\ J_{p_{int}} = \frac{\partial T}{\partial p_{int}} \Big|_{\bar{c}^p, \bar{p}_{int}} = \begin{pmatrix} -\bar{l}_x & 0 & \bar{X}^p - \bar{q}_x & 0 \\ 0 & -\bar{l}_y & 0 & \bar{Y}^p - \bar{q}_y \end{pmatrix} \end{cases}$$

Il reste à calculer Σ^i en prenant l'inverse de P^i . L'hypothèse gaussienne sur les paramètres intrinsèques peut sembler peu réaliste dans la mesure où des erreurs de calibration agissent plutôt sous forme de biais. Nous nous efforcerons donc de choisir P_{int} suffisamment grand de sorte que le modèle gaussien puisse prendre en compte des biais de faible importance. Nous modéliserons P_{int} et P^p par des matrices diagonales, considérant que les incertitudes sont décorréelées ce qui est une hypothèse pertinente en pratique. P^p étant fonction de

l'algorithme d'extraction de (\bar{c}^i, E^i) , il semble difficile de l'estimer objectivement. Nous lui assignerons donc une valeur relativement grande (voir section 3.3).

Enfin, notons que si E^p est l'ellipse mesurée dans l'image et exprimée en pixels alors E^i se déduit des équations de la règle 1 :

$$E^i = J_{c^p}^{-T} E^p J_{c^p}^{-1}$$

2.2.2 Rétro-projection

L'opération de rétro-projection dépend de la configuration utilisée. Dans les sections suivantes, nous étudions les configurations monoculaires et binoculaires.

2.2.2.1 Configuration monoculaire

Dans cette configuration, à cause de l'inobservabilité partielle de la caméra, \mathcal{E}^c est dégénérée. Notons (x^c, y^c, z^c) les coordonnées de c^c . Pour tenir compte de la méconnaissance de z^c , nous augmentons le vecteur aléatoire c^i en y ajoutant artificiellement la mesure indépendante z^c à laquelle nous associons la distribution d'information nulle : $\mathcal{E}(\bar{z}^c, 0, 0)$. En d'autres termes, nous allouons à z^c une valeur sans confiance. D'autre part, c^c est relié à (c^i, z^c) par l'équation :

$$(x^c, y^c, z^c)^T = (z^c X^i, z^c Y^i, z^c)^T \quad (2.3)$$

Dans cette partie, nous considérons implicitement que la projection du centre d'un ellipsoïde est le centre de l'ellipse projetée. Cela est rigoureusement faux. Néanmoins, la différence est très faible, et ce d'autant plus que l'ellipse est centrée dans l'image, ce qui sera en pratique le cas pour assurer le maintien de l'objet dans le champ de vue de la caméra (voir chapitre 4 dédié à l'exploration des objets). Puisque la transformation (2.3) est un difféomorphisme, la règle 1 nous permet d'écrire :

Règle 4 (Rétro-projection d'une loi d'ensemble)

La rétro-projection c^c de c^i suit, en première approximation, une loi d'ensemble $\mathcal{E}^c(\bar{c}^c, \Sigma^c, E^c)$:

$$\begin{cases} \bar{c}^c &= (\bar{z}^c \bar{X}^i, \bar{z}^c \bar{Y}^i, \bar{z}^c)^T \\ \Sigma^c &= J^T \begin{pmatrix} \Sigma^i & 0 \\ 0 & 0 \end{pmatrix} J \\ E^c &= J^T \begin{pmatrix} E^i & 0 \\ 0 & 0 \end{pmatrix} J \end{cases}$$

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

Dans cette règle, \bar{z}^c est *a priori* inconnu mais nous verrons en section 3 que sa valeur peut être fixée par l'estimation à l'instant précédent.

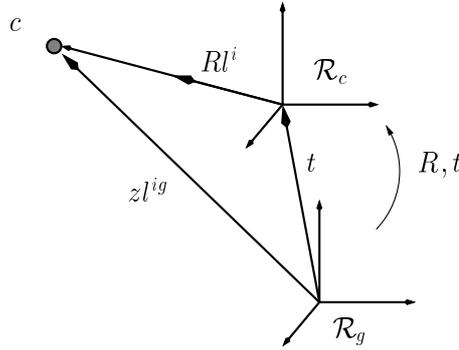


FIG. 2.1 – Initialisation de la distribution avec deux images. \mathcal{R}^c est le repère attaché à l'image locale et \mathcal{R}^g est le repère attaché à l'image globale.

2.2.2.2 Configuration binoculaire

Cette configuration peut être réalisée soit par un système de stéréovision, soit en utilisant simultanément une caméra déportée et une caméra embarquée. Quel que soit le système utilisé, deux images du même objet sont disponibles simultanément. Nous les nommons, en référence au système déporté/embarqué, **image locale** et **image globale**. Un point c se projette en $c^{ig} = (X^{ig}, Y^{ig})$ dans l'image globale et en $c^i = (X^i, Y^i)$ dans l'image locale. Si l'on note $l^{ig} = (c^{ig}, 1)$, $l^i = (c^i, 1)$ et (R, t) le déplacement entre les deux caméras (voir figure 2.1), alors :

$$\begin{cases} (c - t) \wedge Rl^i & = 0 \\ c \wedge l^{ig} & = 0 \end{cases} \quad (2.4)$$

Nous obtenons ainsi un système linéaire d'inconnue c , composé de six équations dont quatre sont indépendantes lorsque les caméras ne sont pas alignées. Nous pouvons trouver une solution aux moindres carrés $c = T(R, t, c^{ig}, c^i)$ puis, grâce à la règle 1, déterminer les paramètres de la loi d'ensemble \mathcal{E}^c .

Preuve :

Le système 2.4 peut s'écrire :

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

où $A = S(Rl^i)$ et $B = S(l^{ig})$, $S(a)$ étant la matrice antisymétrique associée au vecteur a . La solution de ce système aux moindres carrés fournit la solution optimale suivante :

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

Cette solution peut être calculée numériquement. Mais nous devons aussi déterminer son jacobien $J = \frac{\partial c}{\partial (c^i, c^{ig})}$. Puisque l'expression formelle de $(A^T A + B^T B)^{-1} A^T A t$ est plutôt complexe, nous développons une méthode pour calculer J numériquement : Notons $Y = (A^T A + B^T B)$, $z = A^T A t$ et $x = (c^i, c^{ig})$. Nous devons calculer :

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

où $(x_i)_{i \in \{1,2,3,4\}}$ sont les composantes de x . Dans cette expression, Y^{-1} et $\frac{\partial z}{\partial x}$ se calculent numériquement. Plus précisément, nous avons :

$$\frac{\partial z}{\partial x} = \begin{bmatrix} 2(r^i R_{3,1} + q^i R_{2,1})tx & 2(r^i R_{3,2} + q^i R_{2,2})tx & & \\ -(p^i R_{2,1} + q^i R_{1,1})ty & -(p^i R_{2,2} + q^i R_{1,2})ty & 0 & 0 \\ -(p^i R_{3,1} + r^i R_{1,1})tz & -(p^i R_{3,2} + r^i R_{1,2})tz & & \\ 2(r^i R_{3,1} + p^i R_{1,1})ty & 2(r^i R_{3,2} + p^i R_{1,2})ty & & \\ -(p^i R_{2,1} + q^i R_{1,1})tx & -(p^i R_{2,2} + q^i R_{1,2})tx & 0 & 0 \\ -(q^i R_{3,1} + r^i R_{2,1})tz & -(q^i R_{3,2} + r^i R_{2,2})tz & & \\ 2(q^i R_{2,1} + p^i R_{1,1})tz & 2(q^i R_{2,2} + p^i R_{1,2})tz & & \\ -(p^i R_{3,1} + r^i R_{1,1})tx & -(p^i R_{3,2} + r^i R_{1,2})tx & 0 & 0 \\ -(q^i R_{3,1} + r^i R_{2,1})ty & -(q^i R_{3,2} + r^i R_{2,2})ty & & \end{bmatrix}$$

où

$$\begin{cases} r^i &= R_{3,1}X^i + R_{3,2}Y^i + R_{3,3} \\ q^i &= R_{2,1}X^i + R_{2,2}Y^i + R_{2,3} \\ p^i &= R_{1,1}X^i + R_{1,2}Y^i + R_{1,3} \end{cases}$$

Les matrices $\frac{\partial Y^{-1}}{\partial x_i}$ sont, après une réorganisation matricielle, composées des colonnes de $\frac{\partial \text{vect}(Y^{-1})}{\partial x}$ où $\text{vect}(A)$ empile les colonnes de A en un vecteur colonne. Il se calcule en utilisant la relation suivante [Lütkepohl 96] :

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

où \otimes symbolise le produit de Kronecker¹ et

$$\frac{\partial \text{vect}(Y)}{\partial x} = \begin{bmatrix} 2(r^i R_{3,1} + q^i R_{2,1}) & 2(r^i R_{3,2} + q^i R_{2,2}) & 0 & 2Y^{ig} \\ -(p^i R_{2,1} + q^i R_{1,1}) & -(p^i R_{2,2} + q^i R_{1,2}) & -Y^{ig} & -X^{ig} \\ -(p^i R_{3,1} + r^i R_{1,1}) & -(p^i R_{3,2} + r^i R_{1,2}) & -1 & 0 \\ -(p^i R_{2,1} + q^i R_{1,1}) & -(p^i R_{2,2} + q^i R_{1,2}) & -Y^{ig} & -X^{ig} \\ 2(r^i R_{3,1} + p^i R_{1,1}) & 2(r^i R_{3,2} + p^i R_{1,2}) & 2X^{ig} & 0 \\ -(q^i R_{3,1} + r^i R_{2,1}) & -(q^i R_{3,2} + r^i R_{2,2}) & 0 & -1 \\ -(p^i R_{3,1} + r^i R_{1,1}) & -(p^i R_{3,2} + r^i R_{1,2}) & -1 & 0 \\ -(q^i R_{3,1} + r^i R_{2,1}) & -(q^i R_{3,2} + r^i R_{2,2}) & 0 & -1 \\ 2(q^i R_{2,1} + p^i R_{1,1}) & 2(q^i R_{2,2} + p^i R_{1,2}) & 2X^{ig} & 2Y^{ig} \end{bmatrix}$$

□

1. Si $A = [a_{ij}]$ est une matrice de dimensions $m \times n$ et B une matrice de dimensions $p \times q$ alors le produit de Kronecker de A par B est une matrice de dimension $mp \times nq$ formée des blocs $a_{ij}B$:

$$A \otimes B = \begin{pmatrix} a_{11}B & \dots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \dots & a_{mn}B \end{pmatrix}$$

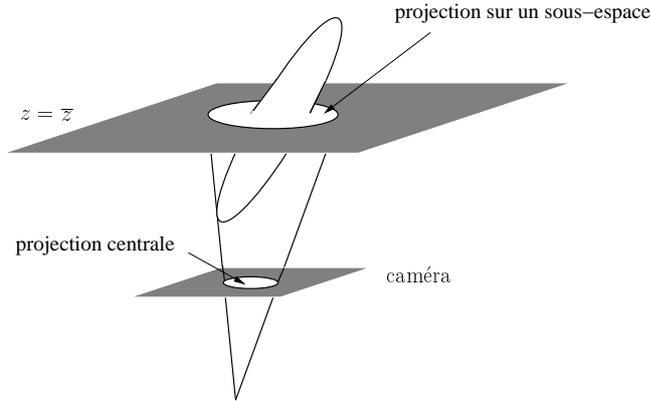


FIG. 2.2 – Décomposition de la projection perspective : 1-projection sur un sous-espace 2-projection centrale de centre le centre de la caméra.

2.2.3 Projection perspective

Réciproquement à la rétro-projection, se pose le problème de la projection perspective. Pour cette transformation, $c^i = (X^i, Y^i)$ est relié à $c^c = (x^c, y^c, z^c)$ par :

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

qui n'est pas un difféomorphisme. Néanmoins, comme le montre la figure 2.2, nous pouvons l'approximer par la composition d'une projection sur le plan $z = \bar{z}$ et d'une homothétie (ou projection centrale) de centre le centre optique de la caméra. Cette approximation est parfois appelée projection para-perspective. Elle ne coïncide pas avec la projection exacte de la surface des limbes mais c'est une très bonne approximation dont la précision dépend de la profondeur relative de l'objet par rapport à sa distance au centre de la caméra. En utilisant la composition d'une projection et d'un difféomorphisme, nous montrons la règle suivante :

Règle 5 (Projection perspective d'une loi d'ensemble)
 Si l'on note

$$E^c = \begin{pmatrix} E_{11}^c & E_{12}^c \\ E_{12}^{cT} & E_{22}^c \end{pmatrix} \text{ et } \Sigma^c = \begin{pmatrix} \Sigma_{11}^c & \Sigma_{12}^c \\ \Sigma_{12}^{cT} & \Sigma_{22}^c \end{pmatrix}$$

où E_{11}^c et Σ_{11}^c sont $(2,2)$, E_{12}^c et Σ_{12}^c sont $(2,1)$ et E_{22}^c et Σ_{22}^c scalaires alors la projection perspective c^i de c^c suit, en première approximation, une loi d'ensemble $\mathcal{E}(\bar{c}^i, \Sigma^i, E^i)$ où :

$$\begin{cases} \bar{c}^i &= (x^c/z^c, y^c/z^c)^T \\ \Sigma^i &= J^T [\Sigma_{11}^c - \frac{\Sigma_{12}^c \Sigma_{12}^{cT}}{\Sigma_{22}^c}] J \\ E^i &= J^T [E_{11}^c - \frac{E_{12}^c E_{12}^{cT}}{E_{22}^c}] J \end{cases} \text{ où } J = \begin{pmatrix} \bar{z}^c & 0 \\ 0 & z^c \end{pmatrix}$$

Preuve :

Dans un premier temps, grâce à la règle 3, la projection c' de c^c sur le sous-espace $z^c = \bar{z}^c$ est un vecteur aléatoire suivant une loi d'ensemble $\mathcal{E}'(\bar{c}', \Sigma', E')$ où :

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

Enfin, l'homothétie se traduit par $c^i = T(c') = (x'/z^c, y'/z^c)$ qui est un difféomorphisme. En appliquant la règle 1, nous montrons que c^i suit une loi d'ensemble $\mathcal{E}(\bar{c}^i, \Sigma^i, E^i)$ dont les paramètres s'expriment de la façon suivante :

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

La preuve est achevée en remplaçant Σ' et E' par leur expression. □

2.2.4 Changement de repère

La loi d'ensemble \mathcal{E}^c est exprimée dans le repère de la caméra \mathcal{R}^c . Or, pour pouvoir comparer des lois relatives à des positions différentes de la caméra, toutes les observations doivent être exprimées dans le même repère de référence \mathcal{R}^o . Signalons que, dans le cas d'un système déporté/embarqué, nous pouvons assimiler \mathcal{R}^o et \mathcal{R}^g de sorte que toutes les observations sont exprimées dans le repère fixe de la caméra déportée. Le déplacement entre \mathcal{R}^o et \mathcal{R}^c est représenté par le vecteur γ que l'on modélise par une distribution gaussienne : $\gamma = \bar{\gamma} + \mathcal{N}(0, \Gamma)$. Pour un robot à six degrés de liberté tel que celui que nous utilisons lors des expérimentations, γ tient compte de la position des articulations et du déplacement entre la pince et l'effecteur (voir annexe B). Si c^o est le vecteur des coordonnées de c^c exprimées dans le repère \mathcal{R}^o , nous pouvons écrire :

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

où D est la matrice de déplacement associée à γ . En utilisant la règle 2, nous déduisons la règle suivante :

Règle 6 (Changement de repère)

Si c^o est l'expression de c^c dans \mathcal{R}^o alors c^o suit une loi d'ensemble $\mathcal{E}(\bar{c}^o, \Sigma^o, E^o)$ dont les paramètres sont :

$$\begin{cases} \bar{c}^o &= R\bar{c}^c + t \\ \Sigma^o &= (J_\gamma \Gamma J_\gamma^T)^{-1} ((J_\gamma \Gamma J_\gamma^T)^{-1} + J_c^T \Sigma^c J_c)^{-1} J_c^T \Sigma^c J_c \\ E^o &= J_c^T E^c J_c \end{cases}$$

où $J_c = \left. \frac{\partial T^{-1}}{\partial c} \right|_{\bar{c}, \bar{\gamma}} = R^T$ et $J_\gamma = \left. \frac{\partial T}{\partial \gamma} \right|_{\bar{c}, \bar{\gamma}}$.

Dans un premier temps, puisque la transformation T est exprimée par un déplacement (R,t) , γ est la concaténation des composantes de la matrice de rotation et du vecteur de translation.

$$\gamma = (r_{11}, r_{12}, r_{13}, r_{21}, r_{22}, r_{23}, r_{31}, r_{32}, r_{33}, t_1, t_2, t_3)^T$$

Ainsi il est facile de calculer :

$$J_\gamma = \left. \frac{\partial T}{\partial \gamma} \right|_{\bar{c}, \bar{\gamma}} = \begin{pmatrix} \bar{x}_c & \bar{y}_c & \bar{z}_c & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \bar{x}_c & \bar{y}_c & \bar{z}_c & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \bar{x}_c & \bar{y}_c & \bar{z}_c & 0 & 0 & 1 \end{pmatrix}$$

Mais pour calculer Γ , nous devons exprimer γ en fonction des sources d'incertitudes du robot. La définition de ces sources d'incertitudes et le calcul de Γ pour un robot cartésien à six degrés de liberté sont donnés en annexe B.

2.3 Conclusion

Dans ce chapitre, nous avons déterminé un ensemble de règles servant à propager une loi d'ensemble. Nous avons tout d'abord défini des règles très générales pouvant s'appliquer à des difféomorphismes (en présence ou non d'une source d'incertitudes externe) et aux projections sur un sous-espace. Cette dernière règle s'avère nécessaire lorsque les dimensions du domaine de définition et de l'espace image ne sont pas identiques. Ces règles sont issues d'une linéarisation de la transformation permettant de conserver le modèle sous forme d'une loi d'ensemble après transformation. Nous avons ensuite appliqué ces règles au cas particulier de la propagation des données visuelles. Les règles ainsi obtenues concernent la rétro-projection (dans le cas monoculaire comme dans le cas binoculaire), la projection perspective et le changement de repère. L'ensemble de ces règles doit nous permettre d'exprimer les observations visuelles sous une forme homogène et compatible rendant possible la fusion de ces observations entre elles. Dans le chapitre suivant, nous décrivons un schéma de fusion basé sur l'estimation *a posteriori* des paramètres \bar{c} et Σ et une approximation ellipsoïdale de l'intersection de deux ellipsoïdes en ce qui concerne le paramètre E .

Chapitre 3

Estimation

Comme le montre la figure 3.1, deux images ne suffisent à donner une bonne estimation ni de la forme, ni de la position d'un objet. Lorsque la caméra est en mouvement, les images acquises au cours de ce mouvement permettent d'affiner l'estimation. Ce chapitre est consacré au processus d'estimation des paramètres d'une loi d'ensemble à partir d'une séquence d'images :

- en section 3.1, nous donnons brièvement le principe de l'initialisation ;
- en section 3.2, nous décrivons le schéma d'estimation permettant de fusionner de nouvelles observations pour affiner le modèle de connaissance ;
- enfin, en section 3.3, nous détaillons des résultats de simulation.

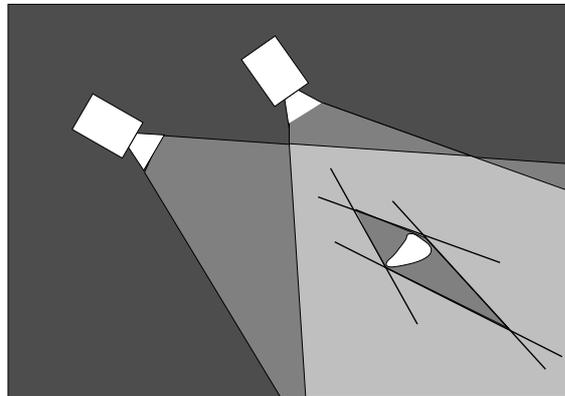


FIG. 3.1 – Estimation de la position et de l'enveloppe d'un objet à partir de deux images.

3.1 Initialisation

À l'initialisation, deux images du même objet sont nécessaires pour obtenir une première estimation non dégénérée du modèle de l'objet. Dans le cas monoculaire, elles correspondent à deux positions successives de la caméra décrites par les repères \mathcal{R}_0^c et \mathcal{R}_1^c (figure 3.2). La première position \mathcal{R}_0^c servira de repère \mathcal{R}^o de référence pour décrire le modèle. En utilisant

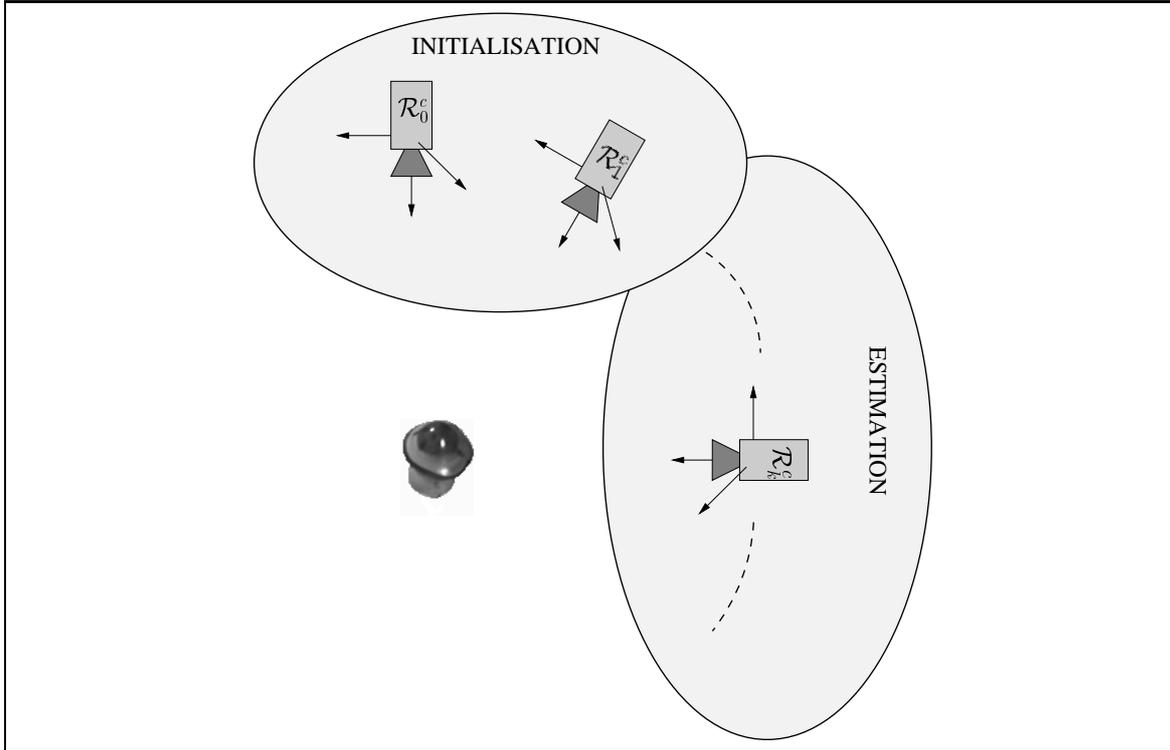


FIG. 3.2 – Dans le cas monoculaire, l’initialisation correspond à deux positions successives de la caméra décrites par les repères \mathcal{R}_0^c et \mathcal{R}_1^c . Les positions suivantes permettent d’intégrer de nouvelles observations au processus d’estimation.

la règle de rétro-projection dans le cas binoculaire (section 2.2.2.2) et en faisant l’analogie :

$$\begin{aligned} \mathcal{R}^g &\rightarrow \mathcal{R}_0^c \\ \mathcal{R}^c &\rightarrow \mathcal{R}_1^c \end{aligned}$$

nous pouvons calculer les paramètres de la distribution qui initialise le modèle de connaissance. Dans le cas binoculaire, deux images sont acquises en même temps. En conséquence, une seule prise de vue est nécessaire à cette initialisation.

En conclusion, quel que soit le système utilisé, nous considérons que l’initialisation du modèle de connaissance fournit une loi d’ensemble initiale que l’on note \mathcal{E}_0 .

3.2 Fusion de nouvelles observations

Considérons un instant k quelconque et postérieur à l’initialisation ($k > 1$). Depuis l’initialisation, $k - 1$ observations ont été intégrées au modèle de connaissance et ont abouti à la distribution $\mathcal{E}_k(\overline{c}_k, \Sigma_k, E_k)$. Cette distribution est appelée distribution *a priori*. À l’instant $k + 1$, nous disposons d’une nouvelle observation issue de la chaîne des observations visuelles (section 1.2). Nous notons $\mathcal{E}_{k+1}^o(\overline{c}_{k+1}^o, \Sigma_{k+1}^o, E_{k+1}^o)$ la loi d’ensemble déduite de cette nouvelle observation. L’objectif de cette partie est d’estimer les paramètres de la distribution *a posteriori* $\mathcal{E}_{k+1}(\overline{c}_{k+1}, \Sigma_{k+1}, E_{k+1})$. Puisque nous avons supposé l’indépendance entre l’in-

certitude et l'erreur dans le modèle additif proposé au début de cette partie (section 1.1), nous estimons séparément les paramètres liés à l'incertitude et ceux liés à l'erreur.

3.2.1 Estimation de l'incertitude

Dans cette partie, nous estimons les paramètres $\overline{c_{k+1}}$ et Σ_{k+1} de la loi d'ensemble \mathcal{E}_{k+1} , c'est-à-dire la position de l'objet et l'incertitude associée. Notons \bar{c} la position réelle du centre de l'objet, à l'instant k , nous savons *a priori* que $\mathcal{P}(\bar{c})$ suit une distribution normale $\mathcal{N}(\bar{c}_k, \Sigma_k^{-1})$. À l'instant $k+1$, nous observons :

$$\overline{c_{k+1}} = \bar{c} + p_{k+1}^o \quad \text{où } p_{k+1}^o \text{ est } \mathcal{N}(0, \Sigma_{k+1}^o)^{-1}$$

Puisque nous voulons fusionner un *a priori* et une observation, nous cherchons à estimer \bar{c} au sens du maximum *a posteriori* (MAP), c'est-à-dire :

$$\overline{c_{k+1}} = \arg \max_{\bar{c}} \mathcal{P}(\bar{c} | \overline{c_{k+1}}^o)$$

Le résultat de cette maximisation dans le cas gaussien est un résultat classique :

$$\overline{c_{k+1}} = (\Sigma_k + \Sigma_{k+1}^o)^{-1} (\Sigma_k \bar{c}_k + \Sigma_{k+1}^o \overline{c_{k+1}}^o)$$

Toutefois nous rappelons les grandes lignes de la démonstration.

Preuve :

La distribution de $\mathcal{P}(\bar{c} | \overline{c_{k+1}}^o)$ se calcule grâce à la règle de Bayes :

$$\begin{aligned} \mathcal{P}(\bar{c} | \overline{c_{k+1}}^o) &= \frac{\mathcal{P}(\overline{c_{k+1}}^o | \bar{c}) \mathcal{P}(\bar{c})}{\mathcal{P}(\overline{c_{k+1}}^o)} = \frac{\mathcal{P}(p_{k+1}^o = \overline{c_{k+1}}^o - \bar{c}) \mathcal{P}(\bar{c})}{\mathcal{P}(\overline{c_{k+1}}^o)} \\ &= \frac{\sqrt{\det \Sigma_{k+1}^o} \sqrt{\det \Sigma_k}}{\mathcal{P}(\overline{c_{k+1}}^o)} \exp\left[-\frac{1}{2} (\overline{c_{k+1}}^o - \bar{c})^T \Sigma_{k+1}^o (\overline{c_{k+1}}^o - \bar{c})\right] \\ &\quad \exp\left[-\frac{1}{2} (\bar{c} - \bar{c}_k)^T \Sigma_k (\bar{c} - \bar{c}_k)\right] \end{aligned}$$

Cette expression est de la forme $\mathcal{P}(\bar{c} | \overline{c_{k+1}}^o) = C e^{f(\bar{c})}$ où C est une constante de normalisation et

$$f(\bar{c}) = e^{-\frac{1}{2} ((\overline{c_{k+1}}^o - \bar{c})^T \Sigma_{k+1}^o (\overline{c_{k+1}}^o - \bar{c}) + (\bar{c} - \bar{c}_k)^T \Sigma_k (\bar{c} - \bar{c}_k))}$$

Donc $\mathcal{P}(\bar{c} | \overline{c_{k+1}}^o)$ est maximal en $\overline{c_{k+1}}$ si

$$\begin{aligned} \left. \frac{\partial C e^{f(\bar{c})}}{\partial \bar{c}} \right|_{\overline{c_{k+1}}} = 0 &\Rightarrow \left. \frac{\partial f(\bar{c})}{\partial \bar{c}} \right|_{\overline{c_{k+1}}} = 0 \\ &\Rightarrow -\Sigma_{k+1}^o (\overline{c_{k+1}}^o - \overline{c_{k+1}}) + \Sigma_k (\overline{c_{k+1}} - \bar{c}_k) = 0 \\ &\Rightarrow (\Sigma_{k+1}^o + \Sigma_k) \overline{c_{k+1}} = \Sigma_{k+1}^o \overline{c_{k+1}}^o + \Sigma_k \bar{c}_k \end{aligned}$$

Ce qui prouve le résultat. □

L'expression de $\overline{c_{k+1}}$ correspond à une moyenne entre l'estimé à l'instant k et la nouvelle observation c_{k+1}^o , pondérés respectivement par la confiance (inverse de la covariance) qu'on leur accorde. De plus, nous pouvons analyser la variance de l'erreur correspondante ($\overline{c_{k+1}} - \bar{c}$) qui vaut :

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

car $Var[\overline{c_k} - \bar{c}] = \Sigma_k^{-1}$, $Var[c_{k+1}^o - \bar{c}] = \Sigma_{k+1}^{o-1}$ et les deux vecteurs aléatoires sont indépendants. Finalement on montre le résultat suivant (figure 3.3-a) :

$$\Sigma_{k+1} = \Sigma_k + \Sigma_{k+1}^o$$

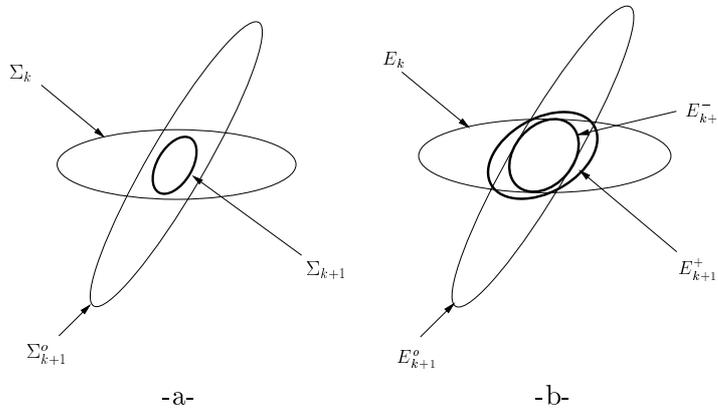


FIG. 3.3 – -a- Estimation de l'incertitude. -b- Estimation du support de l'erreur. L'approximation ellipsoïdale de l'intersection de deux ellipsoïdes peut être externe ou interne.

3.2.2 Estimation du support de l'erreur

Nous estimons à présent le paramètre E_{k+1} de \mathcal{E}_{k+1} , représentant le nouveau support de l'erreur qui est donné par l'intersection entre deux ellipsoïdes (E_k et E_{k+1}^o) supposés centrés à l'origine. Dans le modèle additif, le décentrage d'un ellipsoïde correspond au terme d'incertitude p qui peut être négligé par rapport à la taille des ellipsoïdes dans le calcul de leur intersection. Cette intersection n'est pas elle-même un ellipsoïde. Si nous souhaitons conserver une loi d'ensemble de la forme (1.2) après fusion d'une observation, nous devons faire une approximation ellipsoïdale de l'intersection des deux ellipsoïdes. Essentiellement deux types d'approximations peuvent être calculées : une approximation externe et une approximation interne (voir figure 3.3-b).

3.2.2.1 Approximation externe E_{k+1}^+

Cette approximation correspond au "plus petit" ellipsoïde contenant l'intersection $E_k \cap E_{k+1}^o$. Si l'on considère la famille d'ellipsoïdes engendrée par l'expression

$$(1 - \lambda)E_k + \lambda E_{k+1}^o \quad \lambda \in [0,1]$$

il est possible de montrer que c'est une famille d'ellipsoïdes englobants.

Preuve :

Soit x les coordonnées d'un point de $E_k \cap E_{k+1}^o$. x vérifie :

$$\begin{cases} x^T E_k x \leq 1 \\ x^T E_{k+1}^o x \leq 1 \end{cases}$$

x est un point de $(1 - \lambda)E_k + \lambda E_{k+1}^o$ car :

$$(1 - \lambda)x^T E_k x + \lambda x^T E_{k+1}^o x \leq (1 - \lambda) + \lambda = 1$$

pour $\lambda \in [0,1]$. Ainsi, tout point de l'intersection est un point de $(1 - \lambda)E_k + \lambda E_{k+1}^o$ donc :

$$E_k \cap E_{k+1}^o \subset (1 - \lambda)E_k + \lambda E_{k+1}^o$$

□

Nous devons choisir λ de sorte que l'ellipsoïde soit minimal selon un certain critère. Classiquement il est possible de maximiser soit le déterminant¹ soit la trace de l'ellipsoïde (voir [Maksarov 96]). Lorsque les valeurs propres de l'ellipsoïde sont semblables en ordre de grandeur, ces deux critères sont très proches². Mais dans ce cas précis, utiliser la trace comme critère conduit à maximiser :

$$tr[(1 - \lambda)E_k + \lambda E_{k+1}^o] = (1 - \lambda)tr E_k + \lambda tr E_{k+1}^o$$

qui est linéaire en λ . La maximisation de cette expression sous la contrainte $\lambda \in [0,1]$ conduit donc à $\lambda = 1$ ou $\lambda = 0$ suivant le signe de $(tr E_{k+1}^o - tr E_k)$ ce qui ne donne pas une approximation satisfaisante de l'intersection. Nous utiliserons donc un critère basé sur le déterminant :

$$\lambda = \arg \max_{\lambda \in [0,1]} \det[(1 - \lambda)E_k + \lambda E_{k+1}^o]$$

La valeur optimale de λ a une solution analytique :

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

où $A = E_k$ et $B = E_{k+1}^o - E_k$.

1. Maximiser le déterminant revient à minimiser le volume de l'ellipsoïde car :

$$volume E = \frac{4\pi}{3\sqrt{\det E}}$$

2. En effet, maximiser le déterminant d'une matrice par rapport à un paramètre λ revient à maximiser le logarithme du produit de ses valeurs propres l_i soit à annuler l'expression :

$$\sum_i \frac{1}{\lambda} \frac{\partial l_i}{\partial \lambda}$$

En revanche, maximiser la trace d'une matrice revient à annuler

$$\sum_i \frac{\partial l_i}{\partial \lambda}$$

Preuve :

Nous cherchons à résoudre le problème suivant :

$$\hat{\lambda} = \arg \max_{\lambda \in [0,1]} \det(A + \lambda B)$$

où les matrices $A = E_k$ et $B = E_{k+1}^o - E_k$ sont supposées régulières³. $\det(A + \lambda B)$ est un polynôme du troisième degré en λ :

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

Rappelons que nous cherchons le maximum de $P(\lambda)$. Il est donné par :

$$\begin{cases} P'(\lambda) &= 3a\lambda^2 + 2b\lambda + c = 0 \\ P''(\lambda) &= 6a\lambda + 2b \leq 0 \end{cases}$$

ces équations peuvent s'interpréter comme suit :

- si $a > 0$, $\lambda < \frac{-b}{3a}$ est la plus petite solution de $P'(\lambda) = 0$
- si $a < 0$, $\lambda > \frac{-b}{3a}$ est la plus grande solution de $P'(\lambda) = 0$

Il ne nous reste plus qu'à déterminer a , b , c et d :

- $\lim_{\lambda \rightarrow 0} P(\lambda) = d = \det(A)$
- $\lim_{\lambda \rightarrow +\infty} \frac{1}{\lambda^3} P(\lambda) = a = \lim_{\lambda \rightarrow +\infty} \det(\frac{1}{\lambda}A + B) = \det(B)$
- Si l'on note

$$P(\lambda) \det(-B^{-1}) = \det(-AB^{-1} - \lambda \mathbb{I})$$

nous reconnaissons le polynôme caractéristique de $-AB^{-1}$. Le coefficient de λ^2 est donc $tr(-AB^{-1})$. Ainsi

$$b \cdot \det(-B^{-1}) = tr(-AB^{-1}) \Rightarrow b = \det(B) tr(AB^{-1})$$

- De manière analogue,

$$P(\lambda) \frac{\det(-A^{-1})}{\lambda^3} = \det(-BA^{-1} - \frac{1}{\lambda} \mathbb{I})$$

est le polynôme caractéristique de $-BA^{-1}$. Le coefficient de $\frac{1}{\lambda^2}$ est donc $tr(-BA^{-1})$. Ainsi

$$c \cdot \det(-A^{-1}) = tr(-BA^{-1}) \Rightarrow c = \det(A) tr(BA^{-1})$$

Finalement nous pouvons écrire :

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

Par conséquent, la solution optimale du problème est donnée par :

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

□

3. A et B seront régulières en pratique.

Remarques :

- Lorsque $\lambda = 0$, l'ellipsoïde \mathcal{E}_{k+1}^o contient toute l'information. Ceci signifie que $E_{k+1}^o \subset E_k$.
- Si $\lambda = 1$, la mesure n'apporte plus d'information. Ce cas se présente soit lorsque la caméra revient sur ses pas, soit lorsque l'objet a été complètement reconstruit.
- Il est possible de montrer que la suite des volumes de E_k converge :

Preuve :

Puisqu'elle est minorée par le volume réel de l'objet, il suffit de montrer qu'elle est décroissante, c'est-à-dire que la suite des déterminants de E_k est croissante. Or :

$$\forall k, \det E_{k+1} = \max_{\lambda \in [0,1]} \det[(1-\lambda)E_k + \lambda E_{k+1}^o]$$

$$\implies \forall \lambda \in [0,1], \det E_{k+1} \geq \det[(1-\lambda)E_k + \lambda E_{k+1}^o]$$

ce qui implique en particulier :

$$\det E_{k+1} \geq \det E_k$$

□

Toutefois la convergence de la suite des ellipsoïdes peut être lente dans la mesure où l'approximation externe de l'intersection est très pessimiste. Aussi nous introduisons l'approximation interne qui est beaucoup plus optimiste dans le sens où elle minore le volume réel de l'objet.

3.2.2.2 Approximation interne E_{k+1}^-

Cette approximation correspond au plus grand ellipsoïde inclus dans l'intersection $E_{k+1}^o \cap E_k$ (voir figure 3.4). Pour le calculer, définissons \mathcal{A} l'affinité qui transforme E_k en une sphère. Pour alléger les notations, considérons que toutes les matrices sont exprimées dans la base propre de E_k , auquel cas, cette affinité s'exprime comme étant l'application qui à tout vecteur X associe :

$$\mathcal{A}(X) = \Delta^{-1/2} X$$

où Δ est la matrice diagonale de E_k issue d'une décomposition en valeurs propres. L'application de \mathcal{A} à une forme quadratique, déterminée par sa matrice E , produit une forme quadratique $\mathcal{A}(E)$ d'équation :

$$\mathcal{A}(E) = \Delta^{-1/2} E \Delta^{-1/2}$$

Si l'on décompose $\mathcal{A}(E_{k+1}^o)$ sous la forme :

$$\mathcal{A}(E_{k+1}^o) = \begin{pmatrix} v_1 & v_2 & v_3 \end{pmatrix} \begin{pmatrix} e_1 & 0 & 0 \\ 0 & e_2 & 0 \\ 0 & 0 & e_3 \end{pmatrix} \begin{pmatrix} v_1^T \\ v_2^T \\ v_3^T \end{pmatrix}$$

où v_i est le vecteur propre associé à la valeur propre e_i ($i = 1..3$), alors $\mathcal{A}(E_{k+1}^-)$ est construit de la façon suivante :

$$\mathcal{A}(E_{k+1}^-) = \begin{pmatrix} v_1 & v_2 & v_3 \end{pmatrix} \begin{pmatrix} \max(1, e_1) & 0 & 0 \\ 0 & \max(1, e_2) & 0 \\ 0 & 0 & \max(1, e_3) \end{pmatrix} \begin{pmatrix} v_1^T \\ v_2^T \\ v_3^T \end{pmatrix}$$

Cette construction revient à affecter, à chaque direction propre, la plus petite valeur parmi les deux quantités suivantes : le rayon de la sphère (représentant l'estimation précédente du modèle) et le demi axe de l'observation $\mathcal{A}(E_{k+1}^o)$. Il s'agit bien d'une approximation interne de l'intersection. La transformation inverse \mathcal{A}^{-1} permet de retrouver $E_{k+1}^- = \Delta^{1/2}(E_{k+1}^-)\Delta^{1/2}$.

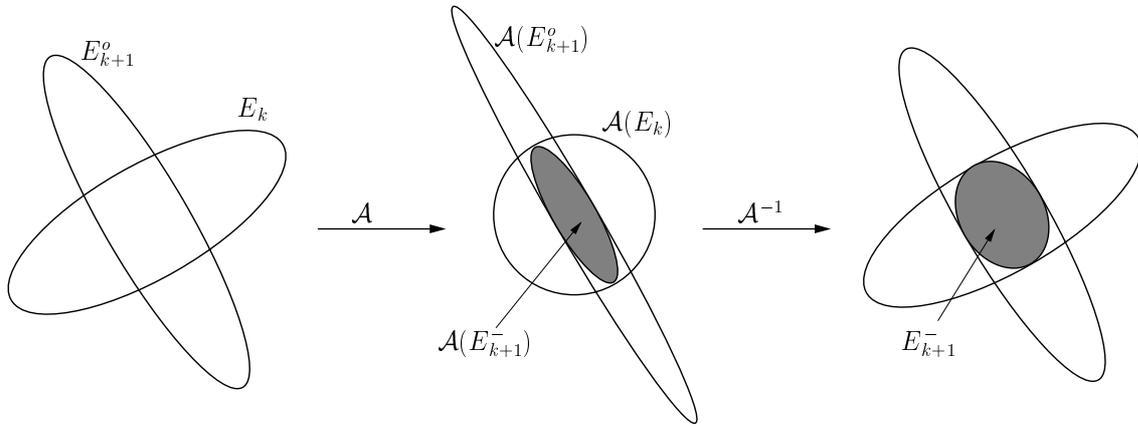


FIG. 3.4 – Approximation interne de l'intersection de deux ellipsoïdes. Soit \mathcal{A} l'affinité qui transforme E_k en une sphère, alors, pour tout vecteur propre de $\mathcal{A}(E_{k+1}^o)$ associé à la valeur propre e_i ($i = 1..3$), nous associons à $\mathcal{A}(E_{k+1}^-)$ la valeur propre $\max(1, e_i)$. Grâce à \mathcal{A}^{-1} , nous trouvons E_{k+1}^- .

3.2.2.3 Approximation intermédiaire

Parce que très pessimiste, l'utilisation de E^+ est plus robuste aux erreurs de mesure que l'utilisation de E^- . Par contre, le taux de convergence de E^+ est très lent et dépend du déplacement entre deux observations. L'utilisation d'une approximation intermédiaire $E^- \subset E \subset E^+$ est préférable. Dans nos futures simulations et expérimentations, nous choisirons une moyenne pondérée de E^+ et E^- :

$$\alpha E^+ + (1 - \alpha) E^-$$

Nous verrons qu'un choix empirique de α peut être fixé convenablement pour toutes les simulations et expérimentations effectuées.

En résumé, étant donnée la distribution *a priori* $\mathcal{E}_k(\overline{c}_k, \Sigma_k, E_k)$ et la distribution de l'observation $\mathcal{E}_{k+1}^o(\overline{c}_{k+1}^o, \Sigma_{k+1}^o, E_{k+1}^o)$, la distribution *a posteriori* $\mathcal{E}_{k+1}(\overline{c}_{k+1}, \Sigma_{k+1}, E_{k+1})$ est donnée par :

$$\begin{cases} \overline{c}_{k+1} &= (\Sigma_k + \Sigma_{k+1}^o)^{-1}(\Sigma_k \overline{c}_k + \Sigma_{k+1}^o \overline{c}_{k+1}^o) \\ \Sigma_{k+1} &= \Sigma_k + \Sigma_{k+1}^o \\ E_{k+1} &= \alpha E^+ + (1 - \alpha) E^- \end{cases} \quad (3.1)$$

3.3 Simulations

La configuration monoculaire est sans doute la plus défavorable à un processus d'estimation dans la mesure où elle fournit des données partiellement inobservables. C'est pourquoi les résultats que nous présentons sont issus de simulations en configuration monoculaire. Nous donnerons toutefois un exemple d'estimation binoculaire basée sur le couple image globale / image locale (voir figure 3.9).

À l'initialisation (voir figure 3.5-a), la caméra est placée dans deux positions voisines de sorte que l'ellipsoïde estimé E_0 est fortement allongé en direction de la caméra (axe de faible observabilité). La première position de la caméra détermine le repère de référence dans lequel le modèle est exprimé. Ensuite la caméra effectue une trajectoire circulaire à une vitesse constante de 3 cm par itération. La figure 3.5-b montre la suite des positions estimées de la caméra. Celle-ci est bruitée en raison de l'incertitude introduite (voir section 3.3.1). Au centre de cette trajectoire, se trouve l'objet : une sphère dont la position ($X = -4.7$ cm, $Y = 4.7$ cm et $Z = 63$ cm) et le rayon (4 cm) sont connus. L'estimation s'effectue sur 200 itérations sans critère d'arrêt. Dans un premier temps, nous étudions l'influence de

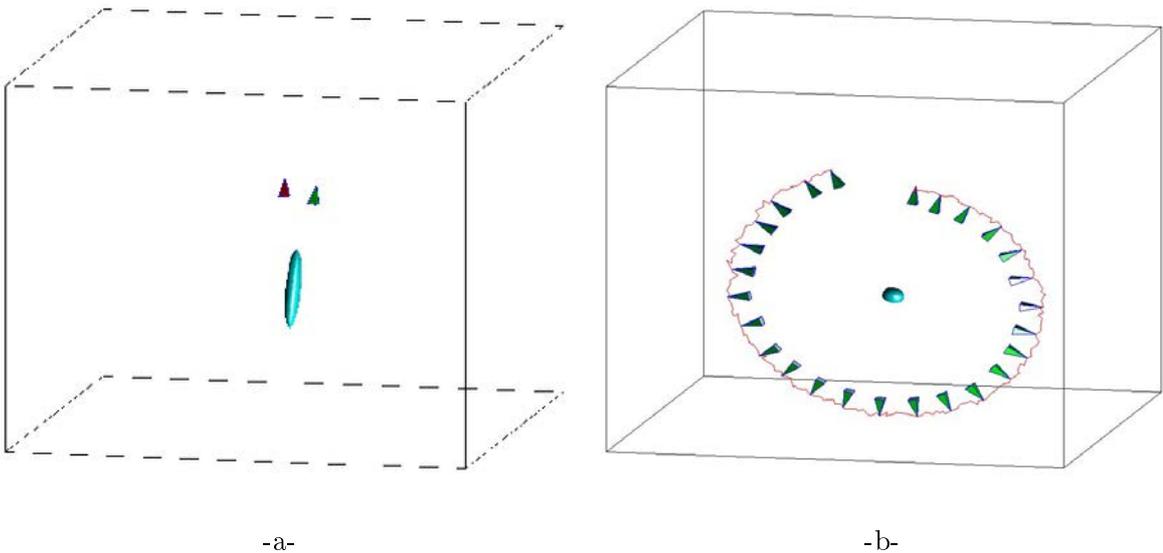


FIG. 3.5 – Conditions de simulation : a- À l'initialisation, la caméra est placée dans deux positions voisines de sorte que l'ellipsoïde estimé est fortement allongé en direction de la caméra. b- Ensuite, la caméra effectue une trajectoire circulaire à vitesse constante.

la prise en compte ou non d'un modèle d'incertitudes sur le positionnement de la caméra vis-à-vis de l'estimation de la position de l'objet. Dans un deuxième temps, nous analysons la convergence et la précision de l'estimation des axes de l'enveloppe.

3.3.1 Estimation de la position de l'objet

L'incertitude sur la position de la caméra est un bruit additif gaussien centré, de variance 10 cm en translation et 5° en rotation. Ces valeurs sont volontairement choisies grandes afin de tester la stabilité de l'algorithme. Les covariances liées au modèle d'incertitude sur les paramètres intrinsèques et à l'algorithme d'extraction (voir section 2.2.1) sont choisies de la façon suivante :

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

Comme nous le signalions en section 2.2.1, P_{int} et P^p sont, toutes deux, décorrélées puisque nous considérons les sources d'incertitudes comme étant indépendantes. Les écarts-types sur P_{int} (2 pixels et 0.3 mm) et P^p (2 pixels) sont choisis suffisamment grands de sorte que le modèle gaussien puisse prendre en compte des biais de faible importance.

Dans un premier temps, la reconstruction ne tient pas compte de l'incertitude sur le positionnement de la caméra, c'est-à-dire que la covariance Γ de la règle 6 est fixée à 0. La figure 3.6 montre la convergence de la position estimée \bar{c}_k vers sa valeur réelle. En 60 itérations, l'estimation se stabilise à une valeur très proche de la valeur réelle avec une précision de l'ordre de 1mm. L'estimation le long de z est initialement la plus bruitée car elle correspond à l'axe de moins bonne observabilité. Le tableau 3.1 indique la valeur de la position estimée en X , Y et Z qui, comparée à la position réelle de l'objet, fournit l'erreur estimée. L'écart-type estimé est issu du modèle d'incertitude. Il correspond à l'inverse de la racine carrée des valeurs propres de Σ à la fin du processus. L'erreur d'estimation est donc de l'ordre du millimètre et toujours inférieure à 3 fois l'écart-type correspondant. La modélisation de l'incertitude semble donc tout à fait adaptée. L'importance de l'erreur selon X par rapport aux autres axes semble être une coïncidence puisque ce fait ne s'est pas reproduit de façon systématique au cours des nombreuses simulations que nous avons effectuées.

		Position de l'objet		
		X (cm)	Y (cm)	Z (cm)
Position réelle		-4.66	4.67	63.40
Trajectoire circulaire	position estimée	-4.53	4.70	63.41
	erreur d'estimation	0.13	0.03	0.01
	écart-type estimé	0.062	0.043	0.069

TAB. 3.1 – Valeurs numériques de l'estimation.

Dans un deuxième temps, l'incertitude de positionnement est prise en compte par l'intermédiaire d'une covariance Γ non nulle. Γ est alors calculée comme nous l'indiquons en annexe B, sans tenir compte de la calibration caméra/effecteur ($P_{\alpha,t} = 0$) puisque

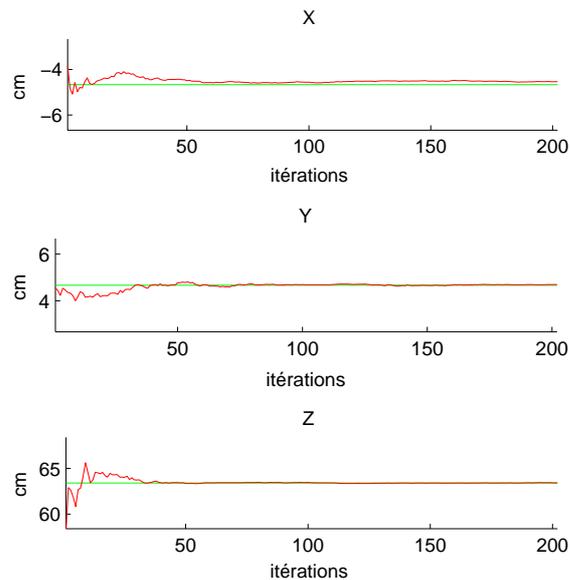


FIG. 3.6 – Estimation de la localisation lorsque l’incertitude sur le positionnement de la caméra n’est pas modélisée. L’évolution de l’estimation apparaît bruitée.

nous utilisons un robot cartésien virtuel. Cela revient à modéliser l’incertitude par un bruit additif gaussien centré de variance 10 cm en translation et 5° en rotation, soit exactement le bruit introduit artificiellement. Dans ce cas, l’estimation de la position est beaucoup plus lisse (voir figure 3.7) mais également beaucoup plus lente. Le tableau 3.2 compare les erreurs d’estimation et les écarts-types estimés au bout de 40 itérations en fonction de la prise en compte ou non de l’incertitude. Le choix d’une grande covariance pour Γ est sans doute la raison de ce faible taux de convergence et mériterait d’être ajustée pour obtenir un compromis entre stabilité et rapidité. Le tableau 3.3 récapitule la position estimée, l’erreur d’estimation et l’écart-type estimé à la fin du processus d’estimation lorsque l’incertitude est prise en compte. La précision obtenue est, à nouveau, de l’ordre du millimètre mais l’écart-type estimé est plus important que dans les simulations précédentes. Ceci est bien entendu imputable à la prise en compte de l’incertitude de position de la caméra qui rend l’estimation finale plus incertaine.

		X (cm)	Y (cm)	Z (cm)
Sans modélisation de l’incertitude	erreur d’estimation	0.19	0.09	0.26
	écart-type estimé	0.18	0.08	0.10
Avec modélisation de l’incertitude	erreur d’estimation	0.29	0.24	2.44
	écart-type estimé	0.23	0.37	1.97

TAB. 3.2 – Comparaison de l’erreur d’estimation et de l’écart-type à l’itération 40 en fonction de la prise en compte ou non de l’incertitude.

La figure 3.8 montre les variances selon chaque axe de l’incertitude associée. L’axe initialement parallèle à l’axe optique est très logiquement mal estimé au départ.

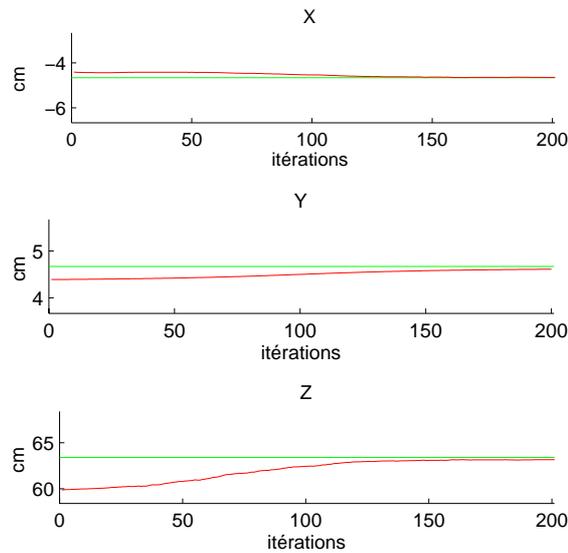


FIG. 3.7 – Estimation de la localisation avec prise en compte de l'incertitude sur le positionnement de la caméra. La convergence de l'estimation est lissée.

		Position de l'objet		
		X (cm)	Y (cm)	Z (cm)
Position réelle		-4.66	4.67	63.40
Trajectoire circulaire	position estimée	-4.65	4.56	63.31
	erreur d'estimation	0.01	0.11	0.09
	écart-type estimé	0.16	0.15	0.10

TAB. 3.3 – Valeurs numériques de l'estimation avec prise en compte de l'incertitude.

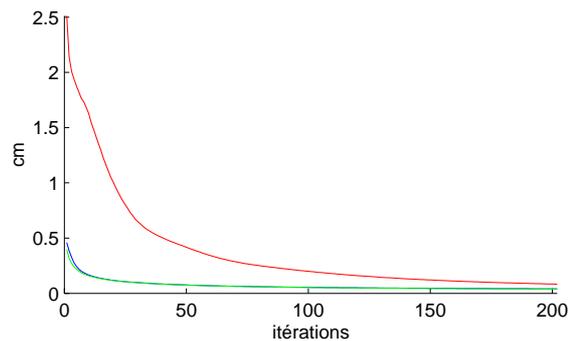


FIG. 3.8 – Incertitudes sur l'estimation de localisation (écart-type selon chaque axe). L'axe initialement parallèle à l'axe optique est logiquement mal estimé lors des premières itérations du processus.

La figure 3.9 fournit un exemple de localisation de l'objet dans un cas binoculaire correspondant à une situation du type image globale/image locale. Ces courbes sont à comparer à l'estimation monoculaire de la figure 3.6. Dans le cas binoculaire, les courbes apparaissent

beaucoup plus stables surtout au début de l'estimation. Cette simulation corrobore ce que nous précisons au début de cette section, c'est-à-dire que l'estimation monoculaire représente le cas le plus défavorable et justifie que nous focalisions les simulations sur le cas monoculaire.

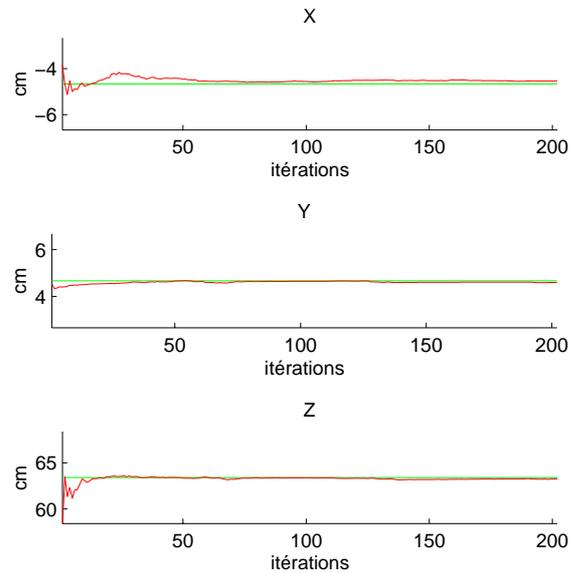


FIG. 3.9 – *Estimation binoculaire de la localisation basée sur le couple image globale/image locale.*

Dans le cas monoculaire et lorsque les déplacements en translation sont faibles, l'estimation de la localisation de l'objet est fortement instable. La figure 3.10 représente le comportement divergeant de l'estimation lorsque la caméra effectue un simple mouvement de centrage en rotation. Le problème est alors totalement inobservable. Dans ce cas, le simple fait de modéliser l'incertitude de positionnement Γ permet de ne pas diverger trop rapidement dans l'attente de déplacements en translation (voir figure 3.11).

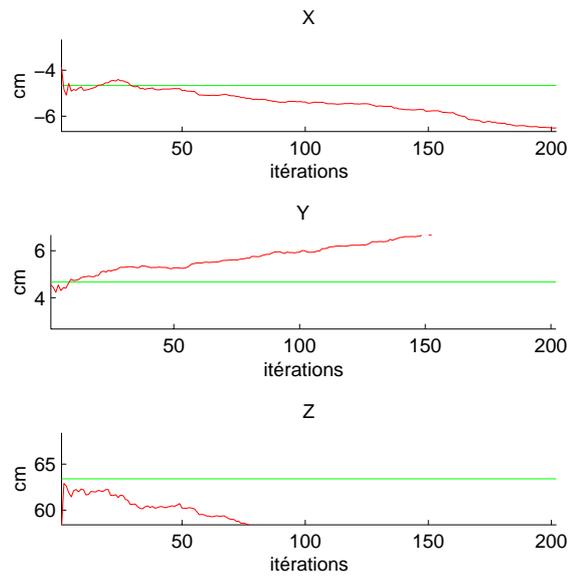


FIG. 3.10 – Estimation de la localisation lorsque l'incertitude sur le positionnement de la caméra n'est pas modélisée et que le mouvement en translation est nul.

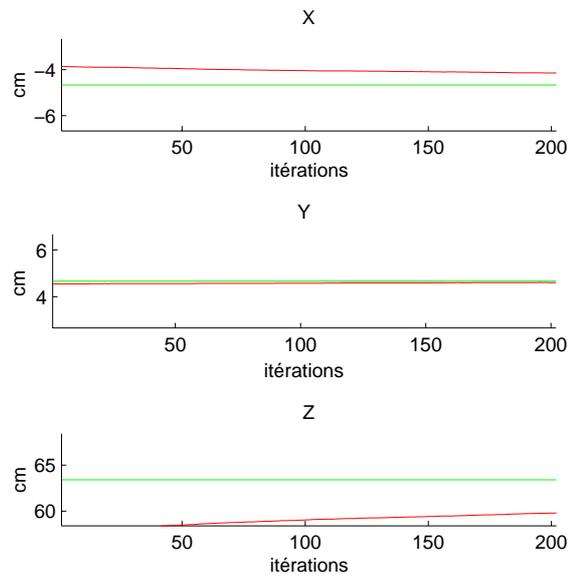


FIG. 3.11 – Estimation de la localisation lorsque l'incertitude sur le positionnement de la caméra est modélisée et que le mouvement en translation est nul.

3.3.2 Estimation des axes de l'enveloppe

Nous avons simulé l'estimation des axes de l'ellipsoïde englobant E_k en utilisant les trois approximations suivantes de l'intersection de deux ellipsoïdes: E^+ , E^- et $E = 0.98E^+ + 0.02E^-$ (voir figure 3.12). Comme nous l'avons annoncé, la précision finale et le taux de

convergence dépendent de la combinaison utilisée. L'approximation externe converge lentement alors que l'approximation interne sous-estime le volume englobant l'objet. L'approximation intermédiaire $E = 0.98E^+ + 0.02E^-$ est apparue comme étant un bon compromis entre vitesse et précision et s'est avérée très satisfaisante au cours des nombreuses simulations et expérimentations effectuées. Nous utiliserons cette approximation dans la suite de notre étude.

3.3.3 Cas d'un objet non sphérique

Les simulations précédentes concernaient la reconstruction d'une sphère. Le plus petit ellipsoïde englobant une sphère étant la sphère elle-même, ces simulations ont permis de s'affranchir des erreurs d'approximation ellipsoïdale. Il convient de vérifier que, dans le cas d'un objet quelconque, l'algorithme n'est pas déstabilisé. Pour ce faire, nous simulons la reconstruction d'une brique de taille $4\text{ cm} \times 4\text{ cm} \times 20\text{ cm}$ positionnée en ($X = -4.7\text{ cm}$, $Y = 4.7\text{ cm}$ et $Z = 63\text{ cm}$). Comme précédemment, la caméra effectue une trajectoire circulaire autour de l'objet et les covariances liées au modèle d'incertitude P_{int} et P^p sont inchangées. La reconstruction ne tient pas compte de l'incertitude sur le positionnement, c'est-à-dire que $\Gamma = 0$.

Nous constatons (voir figure 3.14) que la position converge mais plus difficilement et qu'en particulier sa composante en z est initialement très mal estimée. Ceci s'explique par les erreurs d'approximations ellipsoïdales sur chaque mesure. La figure 3.13 montre l'enveloppe estimée à intervalles réguliers de 5 itérations pour les 35 premières itérations seulement (l'enveloppe estimée varie peu au delà des 35 premières itérations). Dans chaque image, nous avons superposé la brique à reconstruire pour montrer la correspondance avec son enveloppe estimée. Nous pouvons constater que l'enveloppe ellipsoïdale estimée initialement est considérablement réduite au bout de 30 itérations et qu'elle reste toujours englobante vis-à-vis de la brique.

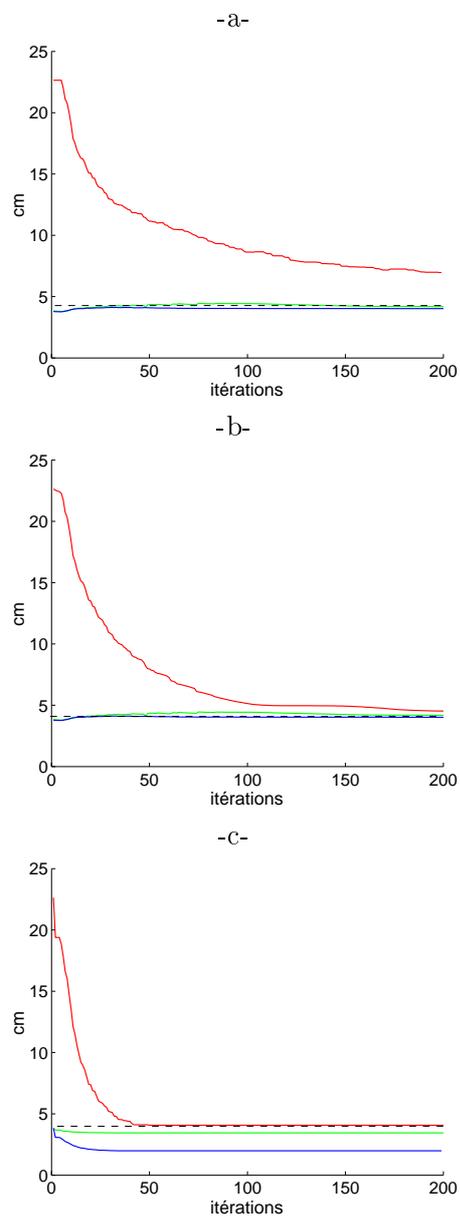


FIG. 3.12 – Estimation des axes de E en utilisant: a- $E = E^+$, b- $E = 0.98E^+ + 0.02E^-$ et c- $E = E^-$. La précision finale et la vitesse de convergence dépendent de l'approximation utilisée.

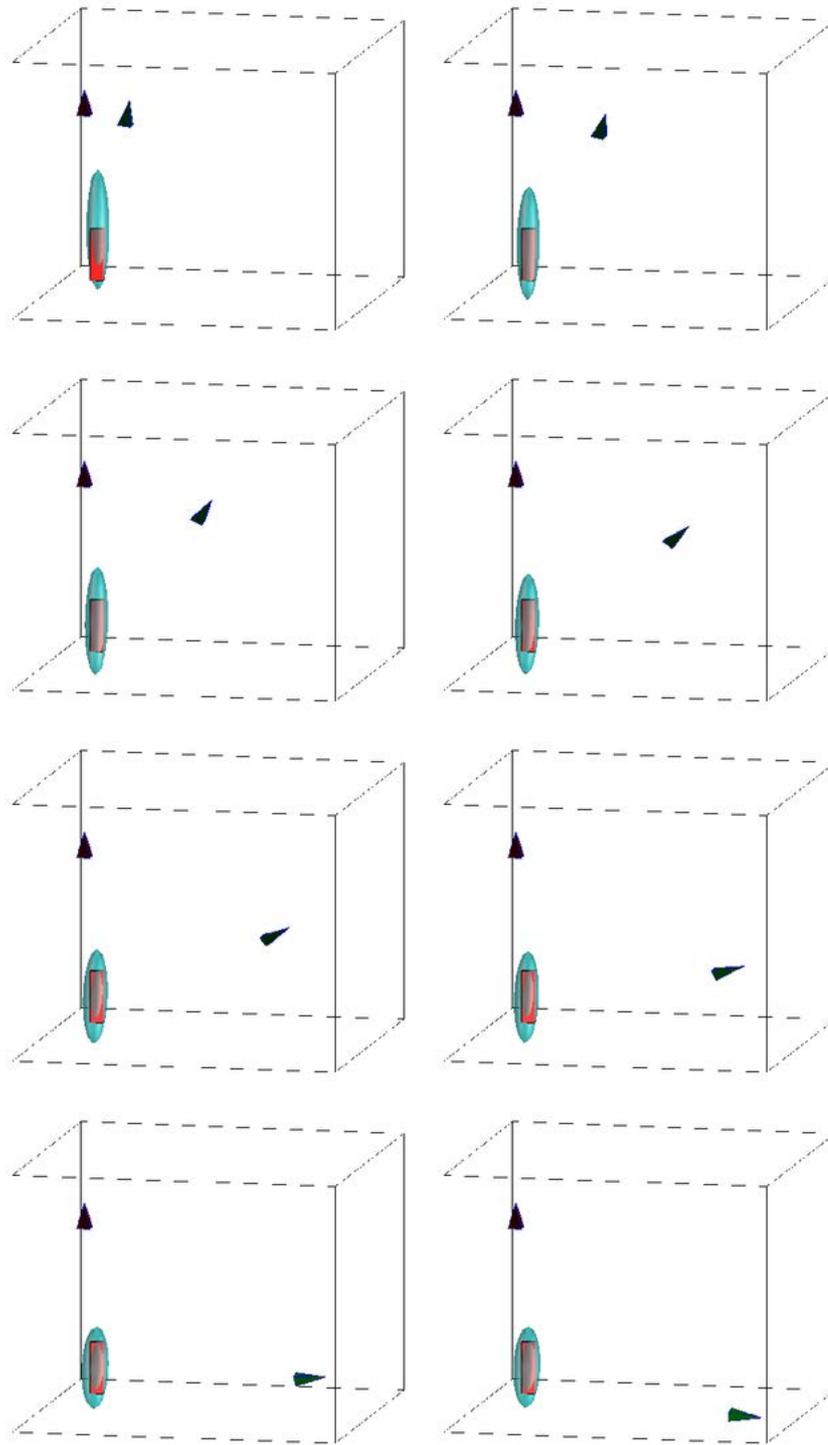


FIG. 3.13 – Estimation des axes de l'enveloppe d'une brique.

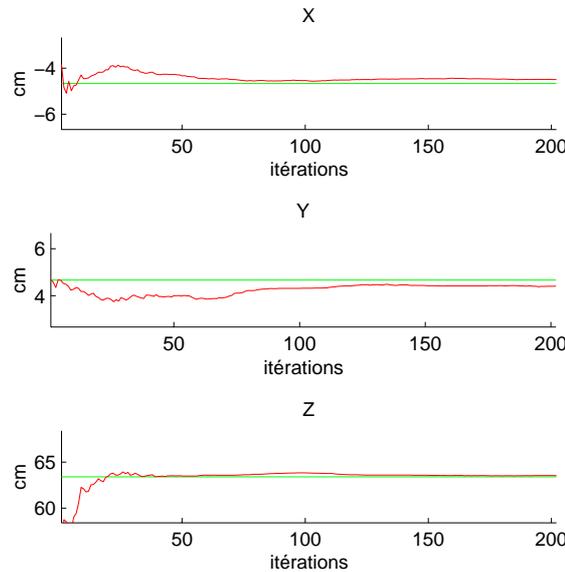


FIG. 3.14 – Estimation de la localisation d'une brique allongée.

3.4 Conclusion

Nous avons établi un schéma de fusion des observations visuelles dans le cadre de la modélisation par loi d'ensemble. Ce schéma est basé sur l'estimation *a posteriori* des paramètres \bar{c} et Σ et une approximation ellipsoïdale de l'intersection de deux ellipsoïdes en ce qui concerne le paramètre E . Nous avons défini deux types d'approximations de cette intersection : une approximation externe et une approximation interne. Les simulations ont montré que la première s'avère très pessimiste car très largement sur-englobante et présente donc un taux de convergence très faible. La seconde montre une convergence plus rapide mais tend vers une limite légèrement inférieure à l'enveloppe réelle. Cependant nous avons pu trouver un compromis en utilisant une combinaison linéaire des deux approximations permettant d'avoir une convergence rapide vers l'enveloppe réelle. Les simulations ont également permis de valider le processus dans une situation plutôt défavorable : l'observation monoculaire. Le risque principal, dans l'estimation de paramètres partiellement inobservables, est que le système soit peu stable. Nous avons montré que le système convergeait dans de nombreuses situations profitant des déplacements de la caméra pour rendre le système observable. La précision atteinte est alors très satisfaisante. Dans les simulations que nous avons présentées, la caméra n'est pas commandée dans le sens où le mouvement qu'elle exécute est prédéfini. Dans le chapitre suivant nous présentons une solution au calcul autonome d'une commande active de reconstruction.

Chapitre 4

Exploration

Dans ce chapitre, nous définissons des lois de commande qui génèrent automatiquement les mouvements exploratoires de la caméra permettant de reconstruire l'objet observé. Dans le modèle de loi d'ensemble que nous avons défini, il s'agit d'identifier un critère traduisant la qualité de la reconstruction. L'optimisation de ce critère sert alors à définir la commande exploratoire. Nous ne considérerons pas de critères basés sur le volume englobant l'objet (c'est-à-dire un critère faisant intervenir le paramètre E) dans la mesure où la forme réelle de l'objet nous est inconnue. En effet, baser la stratégie d'exploration sur la minimisation du volume englobant ne semble pas judicieux si rien ne nous indique que ce volume doit être réduit et surtout dans quelle direction. Il faudrait, pour cela, disposer d'un *a priori* sur la forme de l'objet, ce qui n'est pas le cas. En revanche, choisir un critère lié à l'incertitude de l'estimation de la position présente le double avantage :

- d'optimiser la reconstruction en termes de réduction de l'incertitude ;
- de maximiser, indirectement, le volume découvert puisque l'incertitude est liée à l'inobservabilité, c'est-à-dire que les directions les plus incertaines sont *a priori* celles qui ont été les moins observées.

Ce chapitre est organisé de la façon suivante :

- En section 4.1, nous définissons et exprimons l'information prédite *a posteriori* (ou **IPP**) qui représente une extrapolation de l'information à l'itération suivante par propagation de la connaissance à travers le modèle d'observation et d'estimation.
- En section 4.2, nous déterminons deux types de commandes exploratoires : une commande localement optimale (appelée **LO**) et une commande basée sur le meilleur point de vue (appelée **BVP** pour Best View Point). Toutes deux ont pour but de maximiser l'IPP ou plutôt une fonction de l'IPP puisque la maximisation d'une matrice n'a pas de sens. Dans notre cas, nous attacherons beaucoup d'importance à la sphéricité de l'IPP lors de la maximisation afin d'introduire une certaine isotropie concernant les directions d'observation.
- Enfin, en section 4.3, nous simulons chacune des deux commandes et comparons la convergence des reconstructions obtenues.

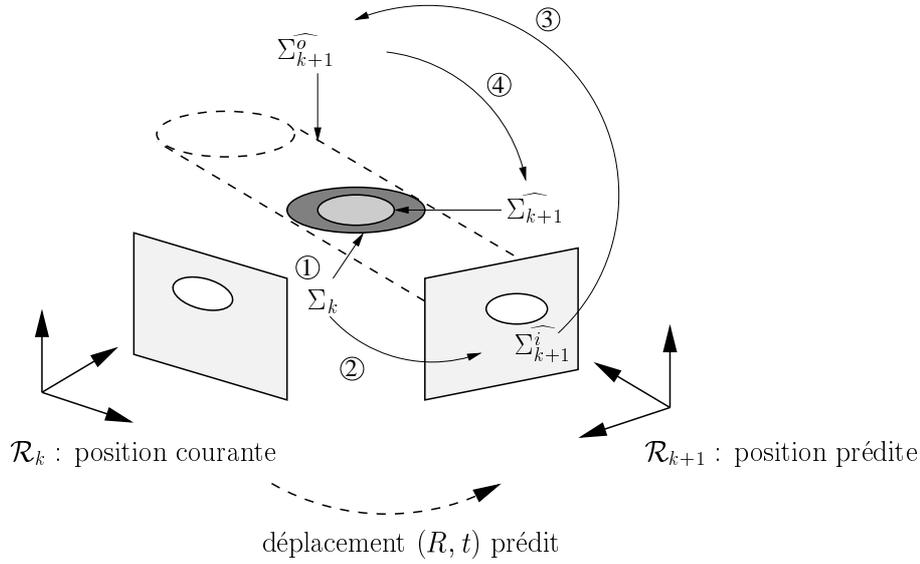


FIG. 4.1 – 1-L'information courante dans le repère de la caméra est Σ_k . À l'instant $k+1$, le déplacement prédit de la caméra vaut (R, t) . 2-La mesure prédite dans l'image est Σ_{k+1}^i . 3-L'observation prédite correspondante est Σ_{k+1}^o . 4-Enfin, en conjuguant l'observation et l'information *a priori*, nous obtenons l'IPP Σ_{k+1} .

4.1 Information prédite *a posteriori* (IPP)

À l'instant k , nous avons estimé les paramètres de la distribution $\mathcal{E}_k(\bar{c}_k, \Sigma_k, E_k)$. Pour alléger la notation, nous considérons que cette distribution est exprimée dans le repère courant de la caméra \mathcal{R}^c . L'information courante dans le repère de la caméra est donc Σ_k . Si, à l'instant $k+1$, le déplacement prédit de la caméra vaut (R, t) , nous pouvons calculer (voir figure 4.1) :

1. l'information *a priori* dans le nouveau repère ;
2. la mesure prédite correspondante ;
3. l'observation prédite ;
4. et enfin, en conjuguant l'observation et l'information *a priori*, nous obtenons l'IPP.

Détaillons à présent chacune de ces étapes. Pour cela, nous utiliserons les règles de propagation vues au chapitre 2.

Puisque l'objet est fixe, la connaissance *a priori* est simplement la transformation de la loi d'ensemble \mathcal{E}_k par le changement de repère (R, t) . Dans ce qui suit nous n'avons besoin que de l'expression de l'information Σ_k , c'est pourquoi nous omettons les autres paramètres de la loi. Si nous supposons que le mouvement est parfaitement connu, ce qui est l'hypothèse la plus logique pour faire de la prédiction, alors la règle 6 nous donne l'information associée :

$$\Sigma'_k = R^T \Sigma_k R \quad (4.1)$$

En l'absence de mesures, une estimation de la mesure prédite est donnée par la projection de \mathcal{E}_k sur le plan image. Si l'on note :

$$R^T \Sigma_k R = \begin{pmatrix} A & B \\ B^T & c \end{pmatrix} \quad (4.2)$$

où A est (2,2), B est (2,1) et c scalaire alors, grâce à la règle 5, l'information liée à la mesure prédite vaut :

$$\widehat{\Sigma_{k+1}^i} = J_0^T \left(A - \frac{BB^T}{c} \right) J_0 \quad \text{où } J_0 = \begin{pmatrix} z_{k+1} & 0 \\ 0 & z_{k+1} \end{pmatrix}$$

Suivant la règle 4, $\widehat{\Sigma_{k+1}^i}$ correspond à l'information 3D suivante :

$$\widehat{\Sigma_{k+1}^o} = J_1^T \begin{pmatrix} J_0^T \left(A - \frac{BB^T}{c} \right) J_0 & 0 \\ 0 & 0 \end{pmatrix} J_1 \quad (4.3)$$

$$\text{où } J_1 = \begin{pmatrix} 1/z_{k+1} & 0 & -X_{k+1}^i/z_{k+1} \\ 0 & 1/z_{k+1} & -Y_{k+1}^i/z_{k+1} \\ 0 & 0 & 1 \end{pmatrix}$$

X_{k+1}^i et Y_{k+1}^i sont les coordonnées prédites de \bar{c}^i à l'instant $k+1$. En pratique, nous utilisons un asservissement visuel de sorte que l'objet apparaisse au centre de l'image ($\forall k, X_k^i = Y_k^i \approx 0$) pour en garder la visibilité tout au long du processus. L'asservissement est réalisé en calculant une commande de suivi en rotation telle que celle présentée dans l'annexe A.

En combinant, grâce à l'équation (3.1), l'observation prédite (donnée par la relation (4.3)) avec la connaissance prédite *a priori* (donnée par la relation (4.1)), l'IPP se calcule de la façon suivante :

$$\begin{aligned} \widehat{\Sigma_{k+1}} &= \Sigma'_k + \widehat{\Sigma_{k+1}^o} \\ &= R^T \Sigma_k R + J_1^T \begin{pmatrix} J_0^T \left(A - \frac{BB^T}{c} \right) J_0 & 0 \\ 0 & 0 \end{pmatrix} J_1 \\ &= \begin{pmatrix} A & B \\ B^T & c \end{pmatrix} + z_{k+1} J_1^T \begin{pmatrix} A - \frac{BB^T}{c} & 0 \\ 0 & 0 \end{pmatrix} J_1 z_{k+1} \\ &= \begin{pmatrix} A & B \\ B^T & c \end{pmatrix} + \begin{pmatrix} A - \frac{BB^T}{c} & 0 \\ 0 & 0 \end{pmatrix} \end{aligned}$$

L'IPP, dans le repère de la caméra à l'instant $k+1$, prend donc la forme suivante :

$$\widehat{\Sigma_{k+1}} = \begin{pmatrix} 2A - \frac{BB^T}{c} & B \\ B^T & c \end{pmatrix} \quad (4.4)$$

4.2 Loi de commande exploratoire

À l'itération k , la loi de commande est déterminée par le déplacement (R,t) ¹ permettant de rejoindre la position voulue à l'instant $k+1$. Il peut être plus intéressant d'exprimer ce déplacement, ramené au temps δ entre deux itérations, par le torseur cinématique de la caméra noté :

$$T = \begin{pmatrix} V \\ \Omega \end{pmatrix} \text{ où } V = \frac{t}{\delta} = \begin{pmatrix} V_x \\ V_y \\ V_z \end{pmatrix} \text{ et } \Omega = \frac{\alpha}{\delta}u = \begin{pmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{pmatrix}$$

Dans l'expression précédente, α est l'angle de rotation et u l'axe unitaire de rotation représentant R . Plus précisément, nous avons :

$$R = \mathbb{I} + \sin \alpha S(u) + (1 - \cos \alpha)S(u)^2$$

où $S(u)$ est la matrice antisymétrique associée à u . Dans le cas d'une commande exploratoire optimale, le déplacement (R,t) doit être calculé de telle sorte que $\widehat{\Sigma}_{k+1}$ soit maximale pour un certain critère. Afin d'introduire l'idée d'isotropie concernant les directions de points de vue, nous attacherons de l'importance à la sphéricité de $\widehat{\Sigma}_{k+1}$. Une matrice symétrique définie positive est sphérique si elle est semblable, à un facteur près, à la matrice identité. Obtenir une IPP sphérique revient à pénaliser les directions peu informatives et tend donc à acquérir une information isotrope. Ce choix nous a paru essentiel dans la mesure où ce que l'on souhaite obtenir, dans le cadre de l'exploration d'un objet, c'est une observation uniforme de toute sa surface. Et puisque l'information est liée à l'observabilité, c'est-à-dire que les directions les plus informatives sont celles qui ont été les mieux observées, tenter d'obtenir une information sphérique semble un bon moyen d'explorer un objet.

On peut montrer dans l'équation (4.4) que la profondeur z_{k+1} de la caméra à l'objet n'influence pas $\widehat{\Sigma}_{k+1}$. Ceci est dû aux linéarisations sous-jacentes aux règles 1 et 2. En conséquence, la vitesse V_z selon l'axe optique n'intervient pas dans le calcul de la commande optimale. Ce degré de liberté peut alors servir à réguler la surface projetée dans l'image. Nous obtenons comme loi de commande :

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

où S_k est la surface apparente de l'objet à l'instant k et où S^* est la surface désirée².

Preuve :

Si l'on suppose que la courbe 3D définissant le contour d'occultation de l'objet observé est plane et parallèle au plan image, alors, d'après l'équation (A.5), la variation de la surface projetée est donnée par les équations suivantes :

$$\begin{cases} \dot{S} = -(2S/z)\dot{z} \\ \dot{z} = -V_z - zY\Omega_x + zX\Omega_y \end{cases}$$

1. Nous omettons volontairement l'indice k s'il n'y a pas d'ambiguïté.

2. S^* peut par exemple s'exprimer comme un pourcentage de la surface de l'image ou bien comme étant égale à celle observée lors de l'initialisation.

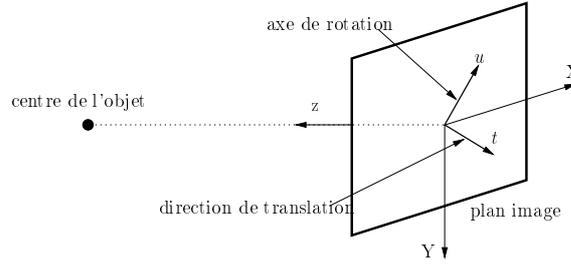


FIG. 4.2 – Lorsque l'objet est centré dans l'image, la commande peut être définie soit en translation (t), soit en rotation (u).

Puisque l'objet est centré dans l'image ($X = Y \approx 0$), la deuxième relation devient :

$$\dot{z} = -V_z$$

Si à l'instant k , on note $S = S_k$ et $z = z_k$ et si l'on souhaite imposer une décroissance exponentielle de S vers S^* , c'est-à-dire :

$$\dot{S}_k = -\lambda(S_k - S^*)$$

où λ est un gain fixant le taux de convergence, alors la commande selon z à l'instant k est :

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

□

La loi de commande ainsi définie est robuste vis-à-vis des hypothèses que nous avons introduites pour sa démonstration. Elle fonctionne même si l'objet n'est pas centré ou si la courbe 3D définissant le contour d'occultation n'est pas plane ou parallèle au plan image.

Pour que l'objet reste centré dans l'image, la direction de translation t et l'axe unitaire de rotation u doivent être liés par la relation : $t = \vec{z} \wedge u$ où \vec{z} est le vecteur unitaire normal au plan image. La commande peut donc être définie soit en translation (t) soit en rotation (u) (voir figure 4.2). Nous choisirons le type de commande en fonction des facilités de calculs offertes par l'une ou l'autre.

Dans la suite, nous proposons deux lois de commande : une commande localement optimale (LO) et une commande basée sur le meilleur point de vue (BVP).

4.2.1 Exploration localement optimale (LO)

Dans cette partie, le mouvement de la caméra optimise localement l'accroissement d'information soit $\widehat{\Sigma}_{k+1}$. Le critère choisi pour quantifier cet accroissement est la trace de $\widehat{\Sigma}_{k+1}$. En maximisant la somme des valeurs propres de la matrice d'information nous espérons capter la notion de sphéricité dont nous parlions précédemment dans les sens où une valeur propre faible pénaliserait le critère. À l'instant $k + 1$, la caméra aura pivoté d'un angle $\alpha \geq 0$ autour du vecteur unitaire $u = (u_x, u_y, 0)$. L'angle α détermine la vitesse de la caméra. Pour α fixé, le mouvement optimal de la caméra est défini par son axe de rotation. Soit :

$$(u_x, u_y) = \operatorname{argmax} \operatorname{tr}[\widehat{\Sigma}_{k+1}]$$

Si l'on note v_1, v_2 et v_3 les vecteurs propres de Σ_k exprimés dans le repère de la caméra et v_x, v_y et v_z l'expression des axes du repère de la caméra dans la base (v_1, v_2, v_3) , nous rappelons que

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

Si l'on note Δ la matrice diagonale des valeurs propres, on montre que l'axe de rotation optimal est donné par :

$$\begin{cases} u_x = \frac{v_y^T \Delta \gamma}{\|\Delta \gamma\|} \\ u_y = -\frac{v_x^T \Delta \gamma}{\|\Delta \gamma\|} \end{cases} \quad \text{où} \quad \gamma = \frac{v_x^T \Delta v_z}{v_z^T \Delta v_z} (v_x^T \Delta v_z v_z - v_x) + \frac{v_y^T \Delta v_z}{v_z^T \Delta v_z} (v_y^T \Delta v_z v_z - v_y)$$

Preuve :

Puisque nous intégrons toutes les observations le long de la trajectoire, nous considérons que les déplacements entre deux observations successives sont petits. Par conséquent, nous faisons l'approximation $\alpha \ll 1$. La matrice de rotation vaut alors :

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

Si l'on note V la matrice des vecteurs propres de Σ_k , nous pouvons décomposer Σ'_k sous la forme :

$$\Sigma'_k = R^T \Sigma_k R = R^T V \Delta V^T R$$

En appliquant la décomposition donnée par l'équation (4.2), nous trouvons :

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

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

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

En utilisant l'équation (4.4), le problème du calcul de la commande revient à déterminer :

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

Notons $\beta = u_y v_x - u_x v_y$. En appliquant l'approximation $\alpha \ll 1$, nous pouvons

écrire :

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

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

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

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

Dans les équations précédentes, K_1 et K_2 sont des termes n'intervenant pas dans le calcul de $tr(\widehat{\Sigma}_{k+1})$. Nous ne donnons donc pas explicitement leur expression.

Après simplification, nous obtenons :

$$tr(\widehat{\Sigma}_{k+1}) \approx K_5 + 2\alpha \left[\frac{v_x^T \Delta v_z}{v_z^T \Delta v_z} \left(\frac{v_x^T \Delta v_z}{v_z^T \Delta v_z} v_z^T - v_x^T \right) + \frac{v_y^T \Delta v_z}{v_z^T \Delta v_z} \left(\frac{v_y^T \Delta v_z}{v_z^T \Delta v_z} v_z^T - v_y^T \right) \right] \Delta \beta$$

où K_5 est un terme constant. L'expression précédente est de la forme

$$tr(\widehat{\Sigma}_{k+1}) \approx K_5 + 2\alpha \gamma^T \Delta \beta$$

$$\text{où } \gamma = \frac{v_x^T \Delta v_z}{v_z^T \Delta v_z} \left(\frac{v_x^T \Delta v_z}{v_z^T \Delta v_z} v_z - v_x \right) + \frac{v_y^T \Delta v_z}{v_z^T \Delta v_z} \left(\frac{v_y^T \Delta v_z}{v_z^T \Delta v_z} v_z - v_y \right)$$

Deux cas sont à considérer :

- Lorsque v_z est un vecteur propre de Δ , $\gamma = 0$ puisque Δv_z est colinéaire à v_z . Ceci implique que $v_x^T \Delta v_z = v_y^T \Delta v_z = 0$ car V est une matrice orthogonale. Dans ce cas, $tr(\widehat{\Sigma}_{k+1})$ ne dépend plus de u , ce qui signifie que n'importe quel mouvement de la caméra peut être choisi.
- Lorsque v_z n'est pas un vecteur propre de Δ , il faut maximiser $\gamma^T \Delta \beta = (\Delta \gamma)^T \beta$, soit le produit scalaire entre $\Delta \gamma$ et $(u_x v_y - u_y v_x)$. Pour cela, remarquons que $\Delta \gamma$ appartient au plan (v_x, v_y) :

$$\begin{aligned} v_z^T \Delta \gamma &= \frac{v_x^T \Delta v_z}{v_z^T \Delta v_z} \left(\frac{v_x^T \Delta v_z}{v_z^T \Delta v_z} v_z^T \Delta v_z - v_z^T \Delta v_x \right) + \frac{v_y^T \Delta v_z}{v_z^T \Delta v_z} \left(\frac{v_y^T \Delta v_z}{v_z^T \Delta v_z} v_z^T \Delta v_z - v_z^T \Delta v_y \right) \\ &= \frac{(v_x^T \Delta v_z)^2}{v_z^T \Delta v_z} \left(\frac{v_x^T \Delta v_z}{v_z^T \Delta v_z} - 1 \right) + \frac{(v_y^T \Delta v_z)^2}{v_z^T \Delta v_z} \left(\frac{v_y^T \Delta v_z}{v_z^T \Delta v_z} - 1 \right) \\ &= 0 \end{aligned}$$

Par conséquent, $(u_x v_y - u_y v_x)$ doit être colinéaire à $\Delta \gamma$, c'est-à-dire :

$$\begin{cases} u_x &= \lambda v_y^T \Delta \gamma \\ u_y &= -\lambda v_x^T \Delta \gamma \end{cases} \quad \text{où } \lambda = \frac{1}{\|\Delta \gamma\|} \text{ est un facteur de normalisation}$$

ce qui achève la démonstration. \square

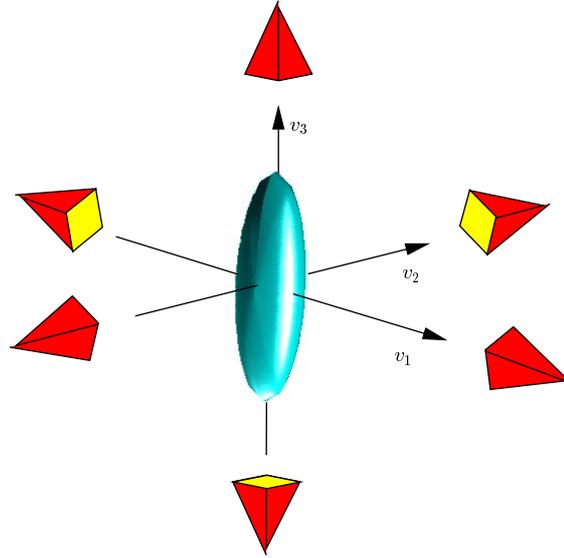


FIG. 4.3 – Les minima locaux correspondent à des positions de la caméra le long des axes principaux de Σ_k .

Remarques :

1. Pour cette loi de commande, les configurations pour lesquelles v_z est un vecteur propre de Δ induisent une commande u nulle. De telles configurations sont des minima locaux. D'autre part, v_z correspond à l'expression de l'axe optique de la caméra dans la base (v_1, v_2, v_3) . Lorsque v_z est un vecteur propre de Δ , cela signifie que l'axe optique est parallèle à l'un des axes de l'ellipsoïde Σ_k . Puisque celui-ci est centré dans l'image, la caméra est donc située sur ce même axe. En résumé, les minima locaux correspondent à des positions de la caméra le long des axes principaux de Σ_k (voir figure 4.3). En particulier lorsque Σ_k est sphérique (c'est-à-dire que toutes les valeurs propres sont égales), toutes les positions correspondent à des minima locaux puisque $tr[\widehat{\Sigma}_{k+1}]$ est constante. Dans ce cas une commande peut être choisie au hasard.
2. Seule la direction du mouvement est contrainte par la loi de commande que nous avons calculée. La norme de la vitesse de déplacement est donc un gain qui peut être choisi au préalable par l'utilisateur.

4.2.2 Loi de commande basée sur le meilleur point de vue (BVP)

Dans cette partie, plutôt que d'optimiser localement $\widehat{\Sigma}_{k+1}$, le déplacement est calculé dans la direction du meilleur point de vue : celui qui conduit à la *plus grande information sphérique* $\widehat{\Sigma}_{k+1}$, c'est-à-dire proportionnelle à la matrice identité. Nous montrons le résultat suivant :

Le meilleur point de vue est situé sur la direction propre de Σ_k de plus grande information

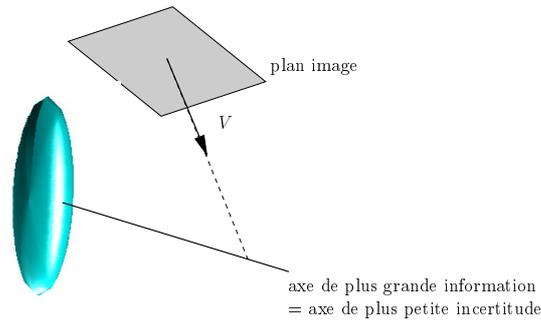


FIG. 4.4 – Le vecteur vitesse $(V_x, V_y)_k$ est dirigé vers l'intersection entre le plan image et l'axe de plus grande information.

Preuve :

Au vu de l'équation (4.4), le meilleur point de vue est celui pour lequel :

$$\begin{aligned} \widehat{\Sigma}_{k+1} = \begin{pmatrix} 2A - \frac{BB^T}{c} & B \\ B^T & c \end{pmatrix} \text{ est diagonale} &\Leftrightarrow B = 0 \text{ et } A \text{ diagonale} \\ &\Leftrightarrow R^T \Sigma_k R \text{ diagonale} \\ &\Leftrightarrow \text{d'après l'équation (4.2)} \\ &\Leftrightarrow R \text{ est composé de vecteurs} \\ &\quad \text{propres de } \Sigma_k \end{aligned}$$

D'autre part, si l'on note a_1 et a_2 les deux valeurs propres de Σ_k hormis c alors Σ_{k+1} a pour valeurs propres $2a_1$, $2a_2$ et c . Par conséquent, une condition nécessaire mais non suffisante pour que les valeurs propres de Σ_{k+1} soient égales est que c soit la plus grande valeur propre de Σ_k . En résumé, les deux conditions précédentes montrent que la troisième colonne de R est le vecteur propre de Σ_k associé à sa plus grande valeur propre. Puisque l'objet est centré dans l'image, le centre de la caméra est situé sur la direction pointée par ce vecteur. Autrement dit, le meilleur point de vue est situé sur la direction propre de Σ_k de plus grande information.

□

Le vecteur vitesse associé $(V_x, V_y)_k$ est donc dirigé vers l'intersection entre le plan image et l'axe de plus grande information (voir figure 4.4). Cette commande est recalculée à chaque itération sans attendre d'avoir atteint le meilleur point de vue. Là encore, la norme de la vitesse est indépendante du processus d'exploration et peut être fixée une fois pour toutes.

4.3 Simulations

Les deux lois de commande ont été simulées en configuration monoculaire dans des conditions similaires à celles de la section 3.3 : à l'initialisation, la caméra est placée dans deux positions voisines. L'objet à reconstruire est une sphère de rayon 4 cm positionnée en $X = -4.7 \text{ cm}$, $Y = 4.7 \text{ cm}$ et $Z = 63 \text{ cm}$. Nous introduisons artificiellement un bruit sur la mesure de la position de la caméra. Ce bruit est un bruit additif gaussien centré, de variance

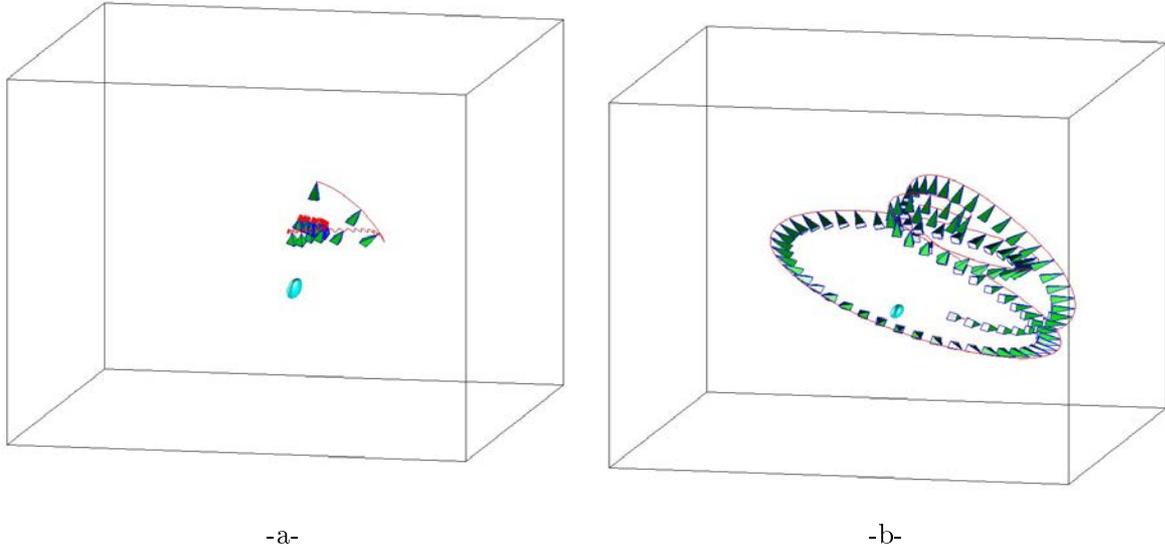


FIG. 4.5 – Trajectoire simulée pour -a- la loi de commande LO, -b- la loi de commande BVP. La loi de commande LO conduit à un minimum local alors que la loi de commande BVP génère une trajectoire très intuitive.

10 *cm* en translation et 5° en rotation. Les covariances liées au modèle d'incertitude sur les paramètres intrinsèques et à l'algorithme d'extraction (voir section 2.2.1) sont choisies de la façon suivante :

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

L'incertitude de positionnement n'est, quant à elle, pas modélisée pour cette étude. Enfin, la trajectoire n'est plus circulaire mais contrôlée par l'une des lois de commandes exploratoires. La norme de la vitesse de déplacement étant fixée à 3 *cm* par itération.

4.3.1 Loi de commande LO

La figure 4.5-a fournit la trajectoire générée par la loi de commande LO. Comme nous l'avons prévu par la théorie, celle-ci conduit à un minimum local. La figure 4.6 montre le résultat de l'estimation de la localisation de l'objet dans le repère de référence. Notons que l'estimation de z ne converge pas vers sa valeur réelle, ce qui s'explique par le fait que le minimum local correspond à une position proche des positions initiales pour lesquelles l'axe z est faiblement observable. L'évolution de l'incertitude d'estimation (figure 4.7) est cohérente avec ce résultat : nous constatons que l'incertitude sur z ne tend pas vers 0 contrairement à l'incertitude sur x et y . Une conséquence directe est que l'axe de l'enveloppe dans cette même direction est très mal estimé (figure 4.11). En appliquant le critère d'évaluation suivant :

$$Q = \frac{\text{Volume initial} - \text{Volume final}}{\text{Volume initial} - \text{Volume réel}}$$

et au terme de l'exploration, la loi de commande LO aboutit à une réduction $Q_{LO} = 67\%$ de la différence (Volume initial - Volume réel). Les valeurs correspondant à l'estimation de la position de l'objet sont données dans le tableau 4.1. L'erreur d'estimation est toujours comprise entre 0 et 3 fois l'écart-type estimé, mais semble en moyenne largement supérieure à l'écart-type estimé. Il semble donc que la modélisation de l'incertitude que nous avons introduite est plus faible que l'incertitude simulée. Ceci n'a rien d'étonnant dans la mesure où, pour ces simulations, nous ne modélisons pas les incertitudes de positionnement.

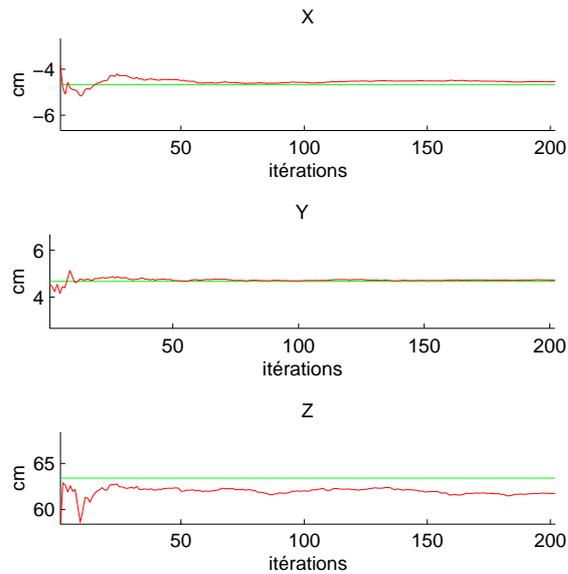


FIG. 4.6 – Estimation de la localisation lorsque le déplacement de la caméra suit une loi de commande LO.

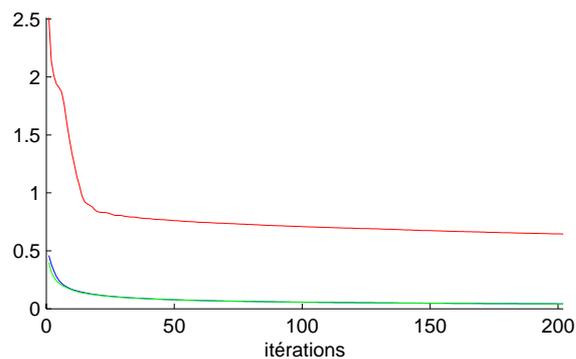


FIG. 4.7 – Incertitudes sur l'estimation de localisation lorsque le déplacement de la caméra suit une loi de commande LO.

4.3.2 Loi de commande BVP

La loi de commande BVP (figure 4.5-b) conduit à une trajectoire très intuitive : la caméra effectue une spirale du pôle à l'équateur. La position de l'objet redevient observable

ce qui se vérifie sur les courbes de convergence de l'estimation (figure 4.8) et de l'incertitude (figure 4.9). L'axe z est correctement estimé. L'erreur d'estimation est de l'ordre du millimètre. Notons que les écarts-types estimés sont sensiblement égaux, ce qui est cohérent puisque notre approche est basée sur l'isotropie de la reconstruction. Les axes de l'enveloppe sont également mieux estimés (voir figure 4.11). En appliquant le même critère que précédemment, la loi de commande BVP aboutit à une réduction $Q_{BVP} = 99.5\%$.

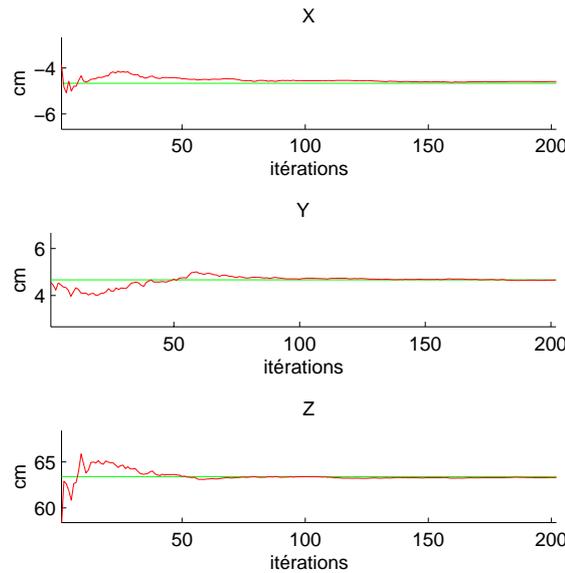


FIG. 4.8 – Estimation de la localisation lorsque le déplacement de la caméra suit une loi de commande BVP.

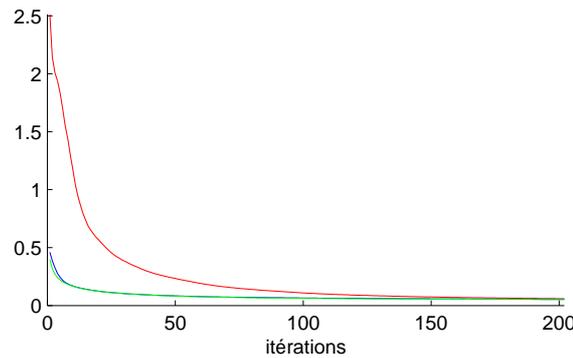


FIG. 4.9 – Incertitudes sur l'estimation de localisation lorsque le déplacement de la caméra suit une loi de commande BVP.

4.3.3 Critère d'arrêt

Le critère d'arrêt choisi pour l'exploration est basé sur la trace de la matrice de covariance Σ^{-1} qui traduit la somme des incertitudes du modèle reconstruit selon chaque axe.

Le seuil correspondant à l'arrêt de la simulation a été arbitrairement fixé à 0.002. Notons qu'en raison du minimum local, ce seuil n'est jamais atteint lors de l'exploration LO (voir figure 4.10-a). Par souci de comparaison, la simulation de la loi de commande LO a été fixée à environ 200 itérations, c'est-à-dire le même temps qu'il faut à la loi de commande BVP pour atteindre son critère d'arrêt (figure 4.10-b).

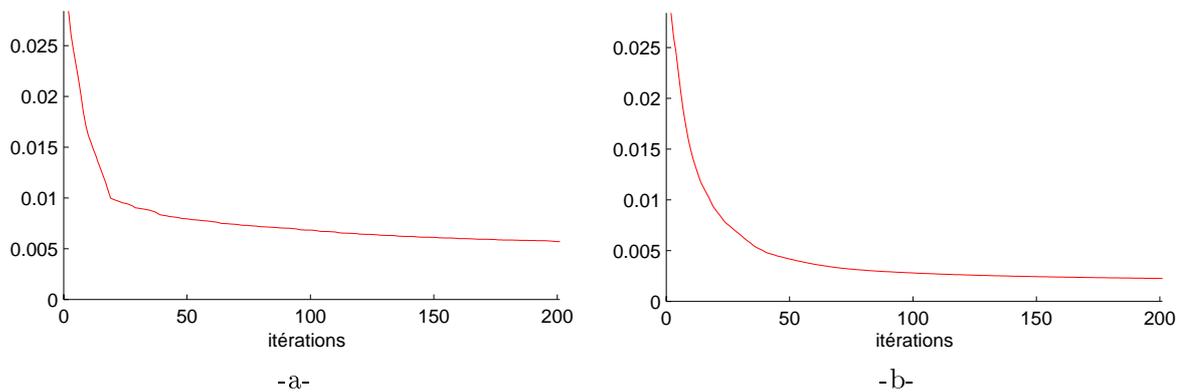


FIG. 4.10 – Convergence du critère d'arrêt pour -a- la loi de commande LO et -b- la loi de commande BVP.

4.3.4 Comparaison des performances

La figure 4.11 témoigne du gain introduit par l'exploration BVP par rapport à une trajectoire circulaire et à la loi de commande LO. La convergence des axes est à la fois plus rapide et plus précise dans le cas d'une loi de commande BVP. Rappelons que la loi LO ne converge pas puisqu'elle conduit à un minimum local. La différence de convergence entre la trajectoire circulaire et la trajectoire BVP peut paraître faible mais il est important de noter qu'une trajectoire circulaire constitue en soi une stratégie d'exploration tout à fait acceptable.

Grâce à sa simplicité et à l'évitement des minima locaux, la loi de commande BVP semble être une meilleure solution au problème de l'exploration. Mais nous verrons que pour des expérimentations réelles, la commande LO ne conduit pas à des minima locaux en raison des incertitudes inhérentes au processus (la commande exécutée n'est pas exactement la commande calculée). Cette stratégie peut donc tout aussi bien être utilisée.

Le tableau 4.1 rappelle les valeurs de l'estimation de la position pour chaque type de trajectoire (circulaire, LO et BVP).

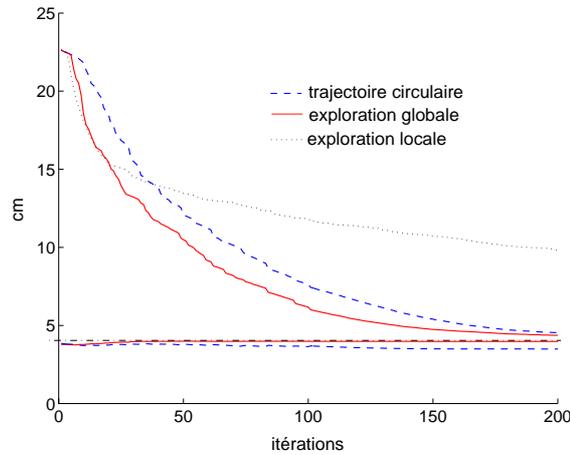


FIG. 4.11 – Comparaison entre trajectoire circulaire, loi de commande BVP et loi de commande LO. Chaque courbe représente l'évolution de la longueur des axes de l'enveloppe estimée pour les différentes trajectoires.

		Position de l'objet		
		X (cm)	Y (cm)	Z (cm)
Position réelle		-4.66	4.67	63.4
Trajectoire circulaire	position estimée	-4.53	4.70	63.41
	écart-type estimé	0.062	0.043	0.069
	erreur d'estimation	0.13	0.03	0.01
Trajectoire LO	position estimée	-4.54	4.72	61.75
	écart-type estimé	0.042	0.043	0.645
	erreur d'estimation	0.12	0.05	1.65
Trajectoire BVP	position estimée	-4.59	4.66	63.29
	écart-type estimé	0.052	0.053	0.059
	erreur d'estimation	0.07	0.01	0.11

TAB. 4.1 – Pour chaque type de trajectoire (circulaire, LO et BVP), ce tableau indique la valeur de la position estimée en X , Y et Z qui, comparée à la position réelle de l'objet, fournit l'erreur estimée. L'écart-type estimé est issu du modèle d'incertitude. Il correspond à l'inverse de la racine carrée des valeurs propres de Σ à la fin du processus.

4.4 Conclusion

Nous avons proposé deux solutions au calcul autonome d'une commande active à des fins de reconstruction : la commande LO et la commande BVP. Chacune de ces lois de commande est calculée de sorte à minimiser l'information prédite *a posteriori*. Cette information résulte de la propagation de la connaissance à travers la chaîne des informations visuelles pour une position prédite de la caméra à l'instant suivant. La commande LO est une commande localement optimale dans le sens où elle maximise à chaque itération le gain d'information. La commande BVP est une commande visant à rejoindre le meilleur point

de vue mais recalculée également à chaque itération. Les simulations correspondant à ces deux lois de commande ont montré que la commande LO conduit à des minima locaux. La commande BVP montre un comportement très intuitif, décrivant une spirale autour de l'objet reconstruit. Cette commande est très satisfaisante d'autant que les performances en termes de reconstruction s'avèrent intéressantes au niveau de la précision et de la vitesse de convergence. Ces résultats sont prometteurs mais nous devons les valider en situation réelle, ce que nous faisons dans le chapitre suivant.

Chapitre 5

Expérimentations

Pour valider notre étude en situation réelle, nous devons extraire le masque des objets d'intérêt. Afin de n'imposer aucune contrainte concernant l'aspect des objets (texture, couleur, niveau de gris,...), nous faisons l'hypothèse (peu restrictive) que la frontière des objets est marquée par une discontinuité de profondeur. Alors, pour tout mouvement de translation de la caméra, cette discontinuité de profondeur induit une discontinuité du champ de mouvement apparent dans l'image. Ainsi un algorithme de détection du mouvement nous donnera le masque des objets. Nous avons choisi l'algorithme d'estimation paramétrique du mouvement de Odobez et Bouthemy [Odobez 95] qui, conjointement à l'estimation du modèle de mouvement dominant, fournit une carte des points non conformes au mouvement dominant. Cet algorithme s'applique à des objets texturés ou non sur des fonds texturés ou non (figure 5.1). Dans notre application, la robustesse de la détection des objets est renforcée par le fait que la seule détection des points de la frontière de l'objet nous permet de retrouver l'ellipse englobante.

Le processus d'exploration a été implanté sur un robot cartésien à 6 axes. Les modèles

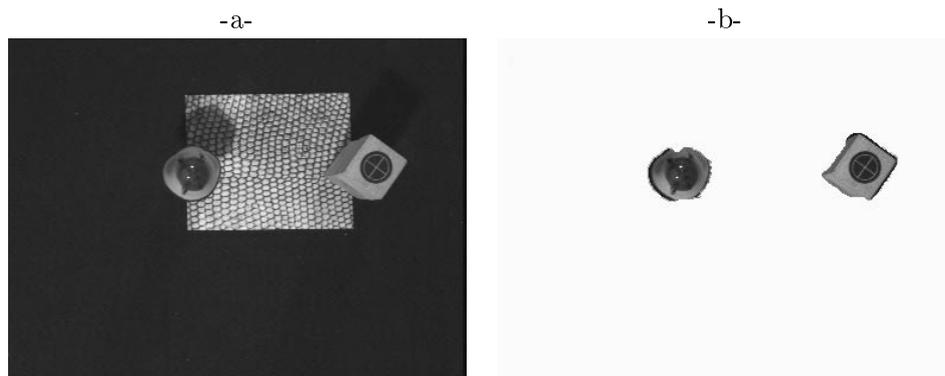


FIG. 5.1 – Résultats de la détection des objets. La figure -a- représente l'image initiale et la figure -b- le résultat de la détection des objets après un faible mouvement de translation.

de covariance P^p et P_{int} ont été choisis identiques à ceux de la section 3.3 :

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

De plus, le modèle d'incertitude sur l'odométrie du robot est un bruit additif normal et centré d'écart-type $1mm$ pour les translations et 1° pour les rotations (voir annexe B pour le détail concernant la modélisation du robot).

5.1 Cas d'un seul objet

Les figures 5.2, 5.3, 5.4 et 5.5 décrivent le processus de reconstruction d'un objet (un champignon) en utilisant la stratégie BVP. La loi de commande LO qui, grâce au bruit inhérent à l'expérimentation, ne rencontre plus de minimum local, fournit des résultats semblables à la loi de commande BVP. Nous verrons en section 5.2 que les trajectoires correspondantes sont en effet très proches dans les deux cas. Dans cette section, nous nous intéressons uniquement aux résultats de la stratégie BVP.

La figure 5.2 représente la séquence des images acquises. L'algorithme de suivi maintient l'objet au centre de l'image. Les masques de segmentation correspondants et les ellipses E^i estimées dans l'image à partir de ces masques sont donnés figure 5.3. Puisque la caméra ne bouge pas lors de l'initialisation, les premières segmentations de l'objet sont estimées grossièrement à la main. Une autre méthode envisageable serait de déplacer très faiblement la caméra lors de ces initialisations afin d'automatiser complètement le processus. Quoi qu'il en soit les résultats de reconstruction sont très stables vis-à-vis de l'initialisation. La figure 5.4 est la séquence des ellipsoïdes estimés. Pour montrer leur correspondance avec l'objet réel, ceux-ci sont reprojétés dans l'image finale en figure 5.5.

Les courbes 5.6 montrent l'évolution de la position estimée de l'objet. Cette convergence est rapide (environ 7s). Puisque nous ne connaissons pas la position réelle de l'objet, il nous est impossible d'analyser quantitativement la précision de l'algorithme. L'incertitude associée à cette estimation est donnée figure 5.7. L'incertitude dont le taux de convergence est le plus faible correspond à l'axe parallèle à l'axe optique de la première image, c'est-à-dire l'axe le moins observable. En ce qui concerne la convergence des axes de l'ellipsoïde englobant E (figure 5.8), là encore, il est impossible d'analyser la précision du résultat final. Les traits en pointillés correspondent à une évaluation grossière des demi longueurs du champignon. L'enveloppe estimée étant englobante, il est normal que ces valeurs ne soient jamais atteintes par l'estimation. L'axe qui converge le plus lentement est toujours l'axe qui est parallèle à l'axe optique de la caméra en phase d'initialisation. Notons enfin que le critère d'arrêt de l'exploration est fixé à 0.002. Son évolution au cours de la reconstruction est donné figure 5.9

5.2 Cas de deux objets

La rapidité de l'algorithme d'estimation (environ 150ms par itération segmentation comprise) nous permet d'estimer, en temps réel, des scènes comportant plusieurs objets. Les

figures 5.10 et 5.11 décrivent la reconstruction approchée de deux objets : un champignon et un marqueur. À l'initialisation, deux images de la scène sont disponibles (voir figures 5.10-a et 5.10-b) et conduisent à l'estimation des ellipsoïdes donnés figure 5.10-c (l'estimation de \bar{c} n'est pas analysée dans cette partie puisque nous ne disposons pas de la position réelle des objets). Comme précédemment, les masques initiaux sont grossièrement définis à la main. La figure 5.10-d représente la projection des ellipsoïdes dans l'image finale et témoigne de la nécessité d'affiner cette estimation trop largement englobante. Dans un deuxième temps, nous basons la stratégie de déplacement de la caméra sur l'exploration de l'un des deux objets (le champignon). Le temps d'exploration est arbitrairement fixé à 20s pour éviter que le robot ne rencontre ses butées articulaires. Comme nous l'avons dit précédemment, la loi de commande LO (figure 5.11-a) ne rencontre plus de minimum local et conduit à une trajectoire très proche de celle induite par la loi de commande BVP (figure 5.11-b). Cette trajectoire est tout à fait semblable à celle simulée mis à part que le temps expérimental est plus court que le temps de simulation. L'estimation finale des ellipsoïdes englobants (figure 5.11-c) a été projetée dans l'image finale (figure 5.11-d) pour montrer l'efficacité de l'algorithme d'exploration BVP. Nous avons remarqué que les résultats expérimentaux de reconstruction diffèrent très peu en fonction de la loi de commande utilisée.

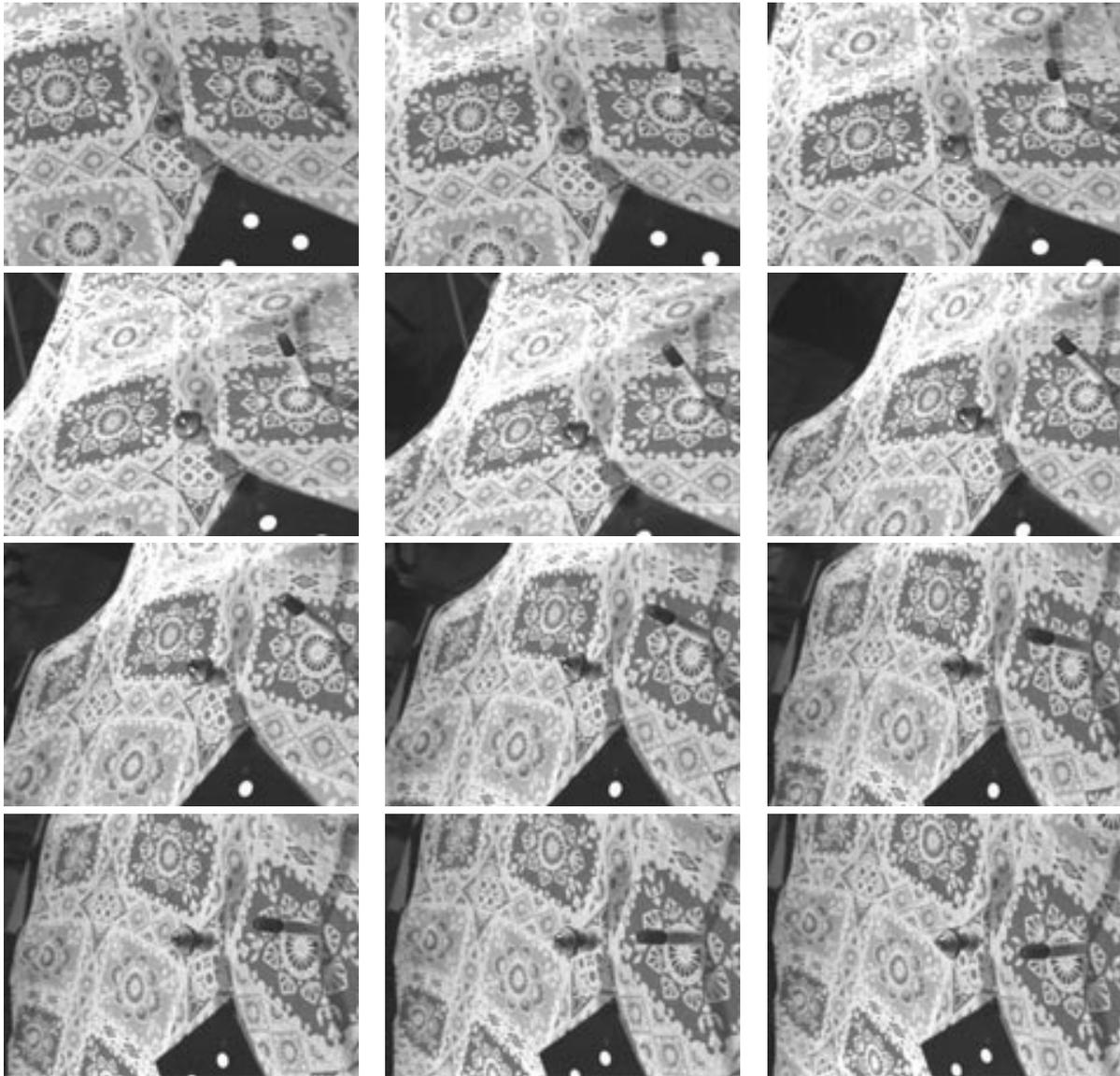


FIG. 5.2 – Séquence d'images acquises. L'algorithme de suivi maintient l'objet au centre de l'image.

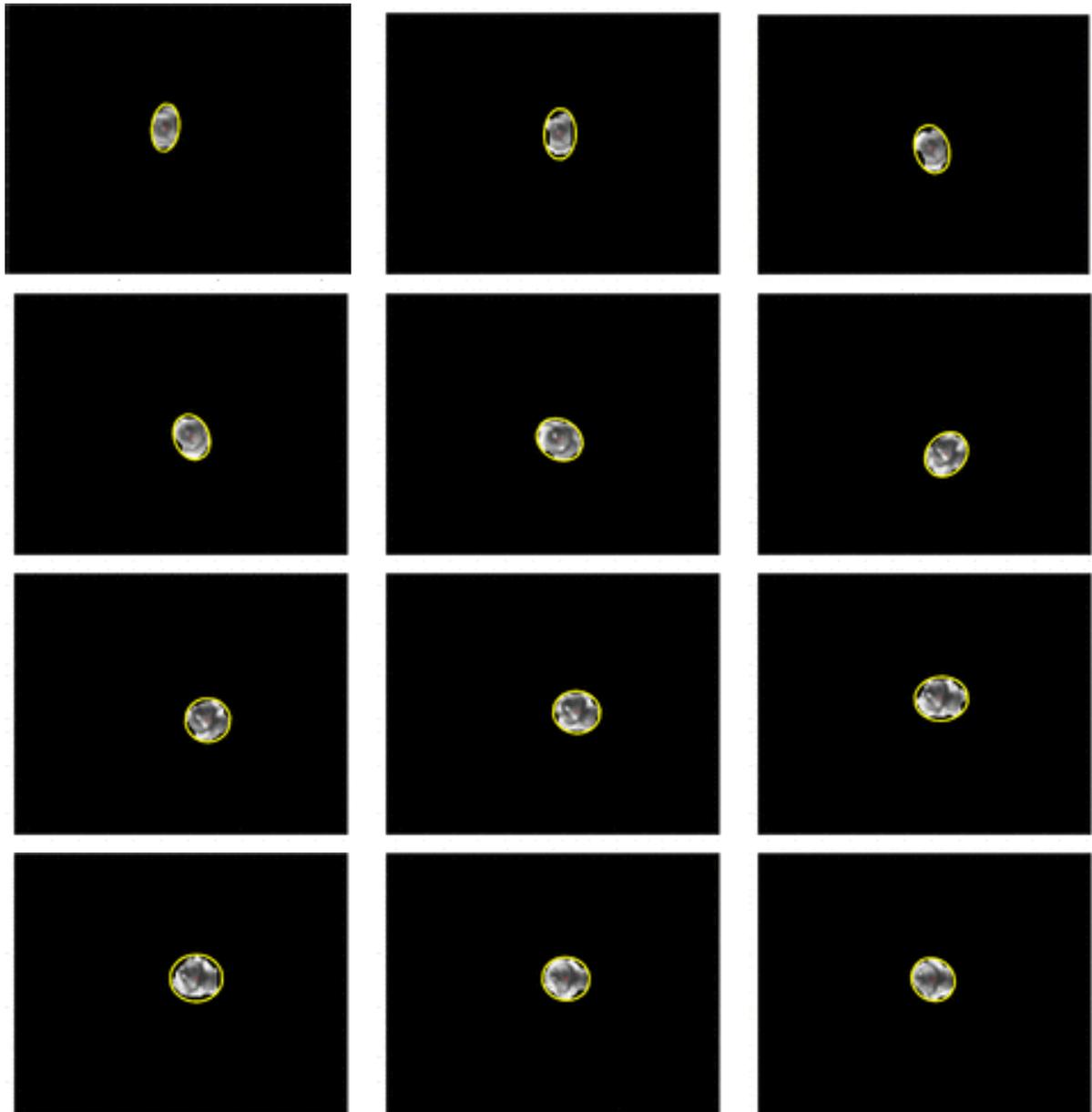


FIG. 5.3 – Masques de l'objet extraits de la séquence d'images à partir desquels sont calculées les ellipses E^i .

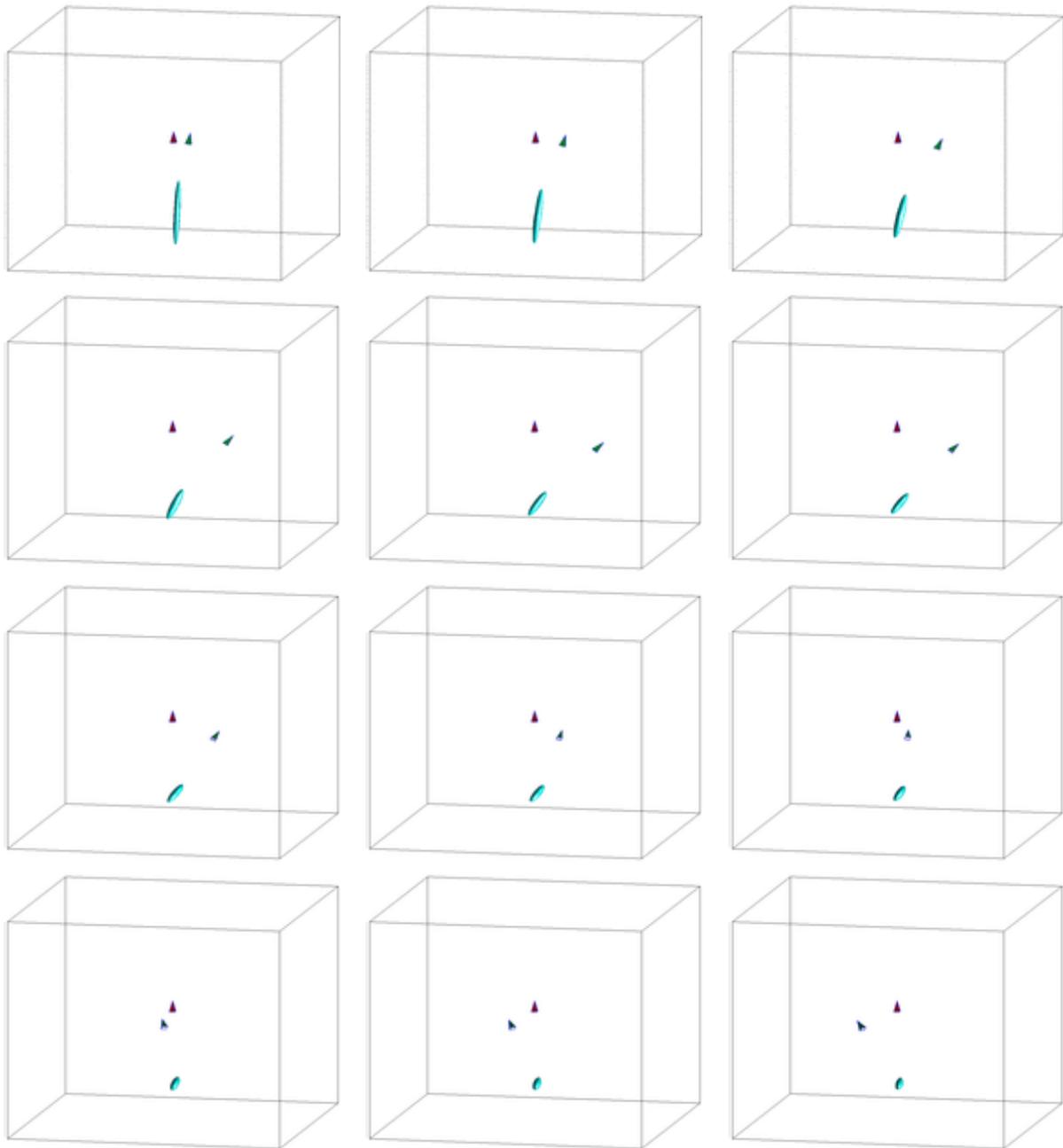


FIG. 5.4 – *Reconstruction approchée de la forme de l'objet.*

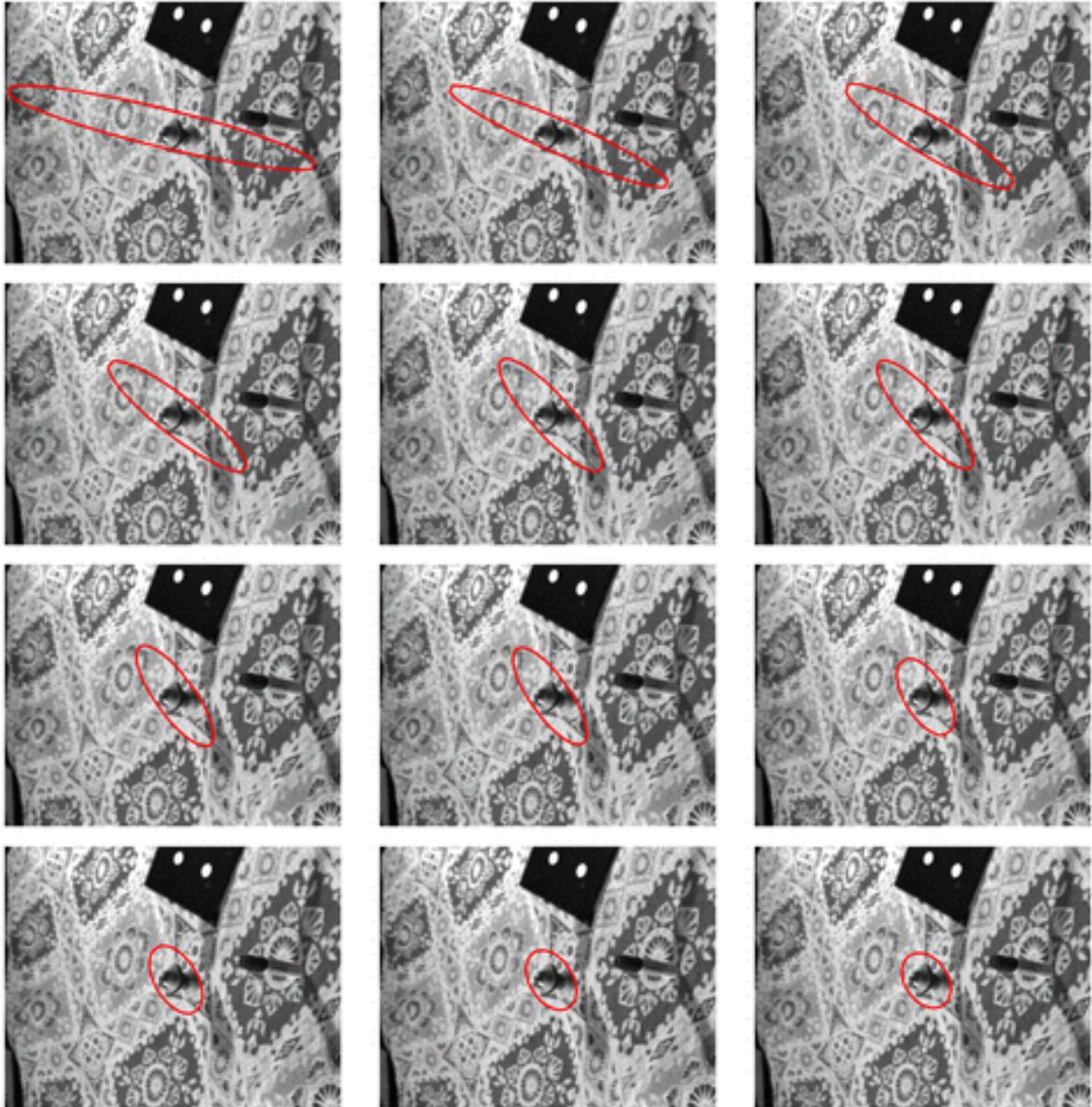
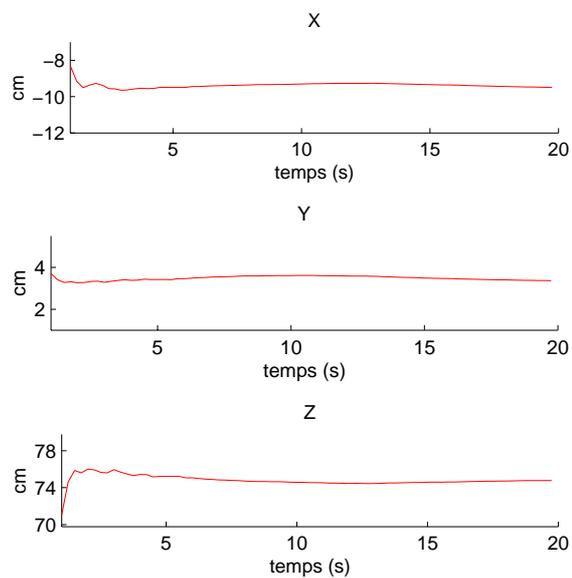
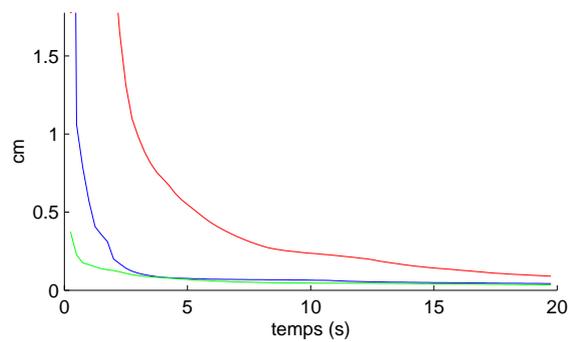
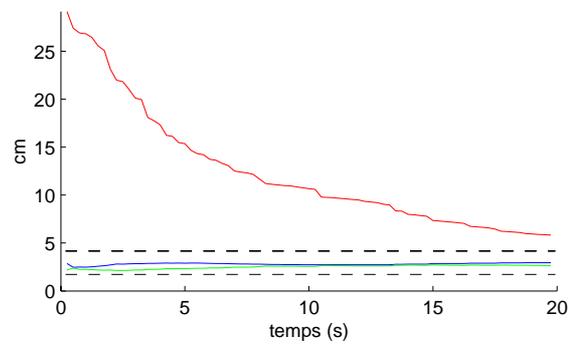


FIG. 5.5 – *Projection de l'ellipsoïde englobant dans l'image finale montrant sa convergence vers l'enveloppe réelle de l'objet.*

FIG. 5.6 – *Estimation de la localisation.*FIG. 5.7 – *Incertitude sur l'estimation de localisation.*FIG. 5.8 – *Estimation des axes de E.*

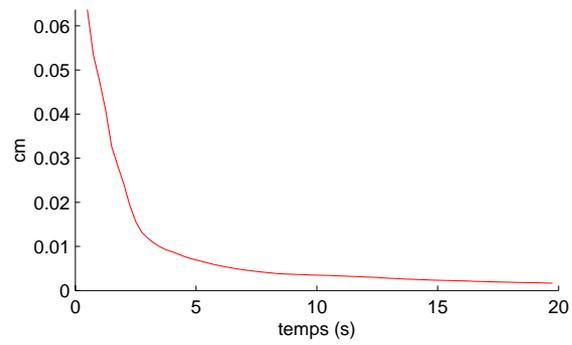


FIG. 5.9 – Evolution du critère d'arrêt de l'exploration.

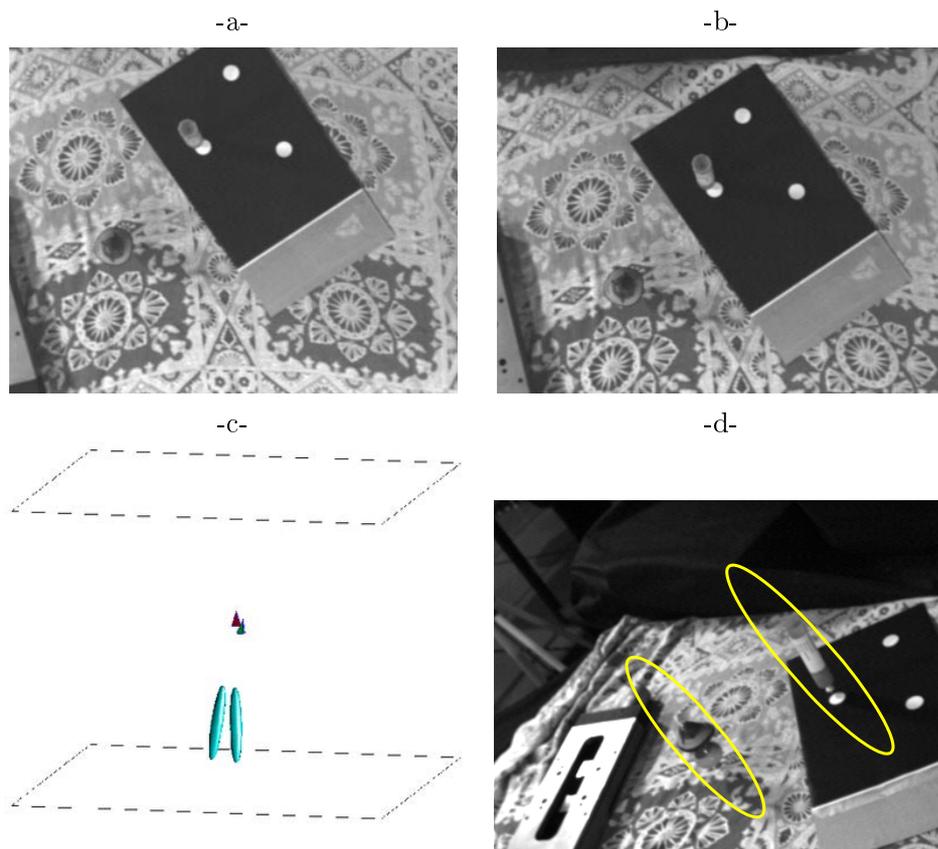


FIG. 5.10 – Initialisation avec deux objets : -a- et -b- images initialisant le processus d'exploration -c- reconstruction approximative des deux objets -d- projection associée dans l'image finale.

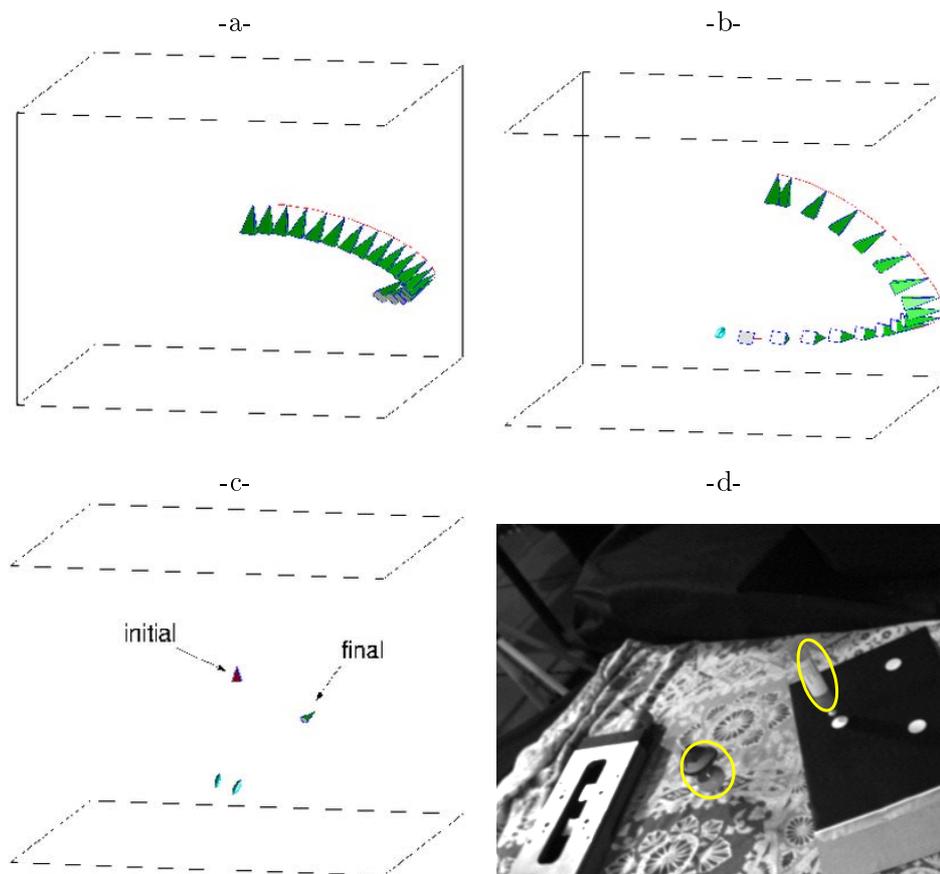


FIG. 5.11 – Reconstruction approximative des objets : -a- trajectoire LO -b- trajectoire BVP -c- reconstruction finale des deux objets et -d- projection des enveloppes englobantes dans l'image finale pour la loi de commande BVP.

Conclusion

Dans cette partie, nous avons défini un modèle représentant chaque objet par une distribution probabiliste. Celle-ci nous permet de calculer pour chaque point de la scène sa probabilité d'appartenance à un objet et d'extraire très facilement les caractéristiques géométriques des objets (position et enveloppe englobante). Ce modèle est peu coûteux puisqu'il requiert simplement l'estimation d'un vecteur 3D \bar{c} et de deux matrices 3D symétriques Σ et E pour représenter la position, l'incertitude et le volume de l'objet. Un ensemble de règles de propagation associées à ce modèle et déduites de la géométrie probabiliste ainsi que du calcul ellipsoïdal nous a permis de construire un processus d'estimation rapide et robuste. Il est basé sur l'estimation *a posteriori* des paramètres \bar{c} et Σ et sur une approximation ellipsoïdale de l'intersection de deux ellipsoïdes en ce qui concerne le paramètre E . Nous avons ensuite défini un processus d'exploration autonome permettant de générer les mouvements de la caméra assurant une reconstruction optimale en termes de réduction de l'incertitude. Nous avons proposé deux solutions au calcul autonome d'une commande active : la commande LO (localement optimale) et la commande BVP (calculée à partir du meilleur point de vue). Ces lois de commandes tendent à minimiser l'information prédite *a posteriori*. Cette approche permet, d'une part, d'optimiser la reconstruction en termes de réduction de l'incertitude et, d'autre part, de maximiser le volume découvert puisque le critère choisi quantifie l'isotropie de l'observation.

Un atout important des algorithmes que nous avons développés réside dans leur rapidité d'exécution qui autorise une implémentation en temps réel. Les nombreuses simulations ont permis d'analyser les performances de l'algorithme vis-à-vis de la reconstruction et de la convergence des paramètres. Nous avons ainsi pu montrer l'importance du choix de l'approximation ellipsoïdale de l'intersection qui est utilisée dans le calcul de l'estimation du modèle. Cette approximation influence la vitesse de convergence et la précision finale des paramètres. Néanmoins, un choix adéquat de cette approximation nous a permis de traiter avec succès et sans exception toutes les situations rencontrées y compris en environnement réel. Les nombreuses validations expérimentales de notre algorithme ont, en effet, montré un comportement tout à fait satisfaisant et prometteur des méthodes développées.

Conclusion et perspectives

Le problème que nous avons abordé est celui de l'exploration visuelle autonome de scènes inconnues. Dans l'idée, désormais largement acceptée par la communauté, que la vision doit être active pour être performante, l'exploration visuelle consiste à contrôler les mouvements d'un système visuel à des fins de reconstruction de l'environnement. Le choix des modèles utilisés pour représenter la scène ou ses constituants a reposé sur différents aspects que nous énoncions au début de ce rapport. En premier lieu, nous avons souligné l'importance de la concision du modèle et sa faculté à représenter uniquement l'information nécessaire aux tâches assignées afin de simplifier la reconstruction proprement dite et la manipulation du modèle associé. En second lieu, nous avons insisté sur le caractère probabiliste du modèle dans le double objectif de permettre la gestion des incertitudes et de prendre en compte les inobservabilités partielles qui sont, par ailleurs, inhérentes aux systèmes visuels. Enfin, est apparue l'idée que le modèle doit permettre une extraction immédiate des informations pertinentes ; informations que nous avons classées en deux catégories. D'une part, l'information peut être de nature locale ou géographique, c'est-à-dire que la scène est définie par des zones vides ou occupées. Cette information, dite d'encombrement, se prête aisément à la manipulation de probabilités. D'autre part, l'information peut être de nature structurelle ou géométrique, c'est-à-dire que la scène est définie par des primitives censées représenter les constituants de la scène. Le souhait d'accéder simultanément à une information d'encombrement et à une information structurelle nous a amené à définir la notion d'objet en tant que constituant de la scène. Cette décomposition hiérarchique de l'environnement scène/objets nous a conduit naturellement à décomposer l'exploration d'une scène en deux étapes. Dans un premier temps, le système doit retrouver tous les objets de la scène. En second lieu, chaque objet doit être reconstruit. Ces deux étapes fournissent des consignes sur lesquelles le système visuel doit s'asservir. Typiquement il est apparu que le système doit suivre les objets d'intérêt tout en effectuant des mouvements de nature à explorer la scène. En conclusion de cette décomposition, nous avons apporté des contributions dans trois domaines distincts : la recherche d'objets, l'exploration d'objets et l'asservissement visuel. Par ailleurs, nous avons souligné, en introduction, la complémentarité des rôles joués par un système visuel extérieur à la scène et par un système visuel immergé dans la scène. Nous avons souhaité faire de cette complémentarité un fil conducteur à nos travaux et avons remarqué que la coopération entre des informations globales acquises d'un point de vue extérieur et des informations locales acquises au sein même de la scène permettent souvent de simplifier le problème posé.

Recherche d'objets

En ce qui concerne la recherche d'objets, nous avons défini les zones de recherche des objets comme étant des lignes de vue de l'image globale dont la probabilité de détection est supérieure à un certain seuil. Chacun de ces axes ayant été discrétisé en cellules, nous avons défini, pour chaque cellule, une variable aléatoire d'occupation X_i à valeur dans $\{0,1\}$. La probabilité associée définit ainsi la probabilité d'occupation d'une cellule. Afin de prendre en compte d'éventuelles dépendances entre les variables aléatoires X_i , nous avons plutôt considéré la loi conjointe du vecteur $X = (X_1, \dots, X_n)$. Ayant montré comment tout *a priori* et toute observation d'un événement se traduit en une contrainte linéaire sur cette loi conjointe, nous avons décrit un schéma d'intégration itératif des contraintes linéaires et un certain nombre d'outils s'y rapportant. L'enchaînement de ces outils nous a permis de construire un algorithme général permettant d'intégrer des contraintes et de nature à gérer un maximum de situations. Ces algorithmes permettent donc, pour une configuration donnée de la caméra, d'intégrer l'observation correspondante au modèle de connaissance. Le calcul de la prochaine configuration permettant d'optimiser la recherche constitue l'étape ultime du processus. Étant donnée la connaissance à l'instant courant, nous avons proposé un algorithme permettant de déterminer la configuration suivante qui maximisera l'espérance d'une fonction appelée *fonction gain d'information (FGI)*. Cette fonction doit quantifier le gain apporté par une modification quelconque des probabilités d'occupation.

À la différence des études déjà menées à propos de la recherche d'objet, le problème, tel que nous l'avons défini, concerne la recherche de tous les objets d'une scène ce qui est un problème sensiblement différent de celui de la recherche d'un objet souvent traité, d'ailleurs, du point de vue de la reconnaissance. La discrétisation de la scène que nous proposons est basée sur les informations issues de l'image globale. Celle-ci permet de définir des zones de recherche originales et dont la simplicité laisse entrevoir la possibilité d'une modélisation très générale du problème. L'approche utilisant des probabilités d'occupation dans chaque cellule n'est pas nouvelle [Ye 97]. Ce qui l'est plus, c'est la modélisation par loi conjointe rendue possible grâce à la simplicité de la décomposition de la zone de recherche en axes globaux et qui permet de tenir compte de toutes sortes de dépendances entre les probabilités. Nous avons vu que ces dépendances sont un moyen très général d'intégrer un *a priori* sur la scène. Un autre aspect innovant réside dans la remise à jour des probabilités par intégration de contraintes dans un système linéaire. Cette remise à jour peut alors se faire de façon itérative. Enfin, nous avons abordé le calcul de la configuration suivante permettant d'optimiser la recherche. Le point de vue adopté est relativement classique, il consiste à parcourir l'ensemble des chemins possible et de choisir celui aboutissant à la plus grande espérance du gain d'information qui en découle. L'originalité du travail réside plutôt dans la façon de quantifier le gain apporté par une observation et la comparaison de différentes méthodes notamment avec les résultats théoriques d'une stratégie gloutonne.

L'algorithme de recherche dans sa globalité a fait l'objet de nombreuses simulations. Celles-ci ont montré l'importance du choix de la FGI. Les différentes stratégies ont été évaluées vis-à-vis du nombre moyen d'itérations (NIM) nécessaire à l'aboutissement de la recherche. Les stratégies basées sur la norme N_2 et sur la variation de l'entropie apportent un gain significatif par rapport à une stratégie gloutonne dont nous avons calculé le NIM théorique. Nous avons proposé un critère d'arrêt de la recherche basé sur l'entropie des

lois marginales. Au cours de ces simulations, il apparaît qu'introduire un *a priori* faux est très pénalisant à la fois en ce qui concerne le critère mais également au niveau du nombre d'itérations nécessaires à la découverte de tous les objets.

Une première limitation de la méthode concerne donc la gestion des mauvais *a priori*. Lorsque l'on intègre un *a priori* qui est faux, le système est induit en erreur de sorte que la recherche est sous-optimale. Dans ce cas il faut envisager une intégration virtuelle de l'*a priori* de sorte qu'il ait une influence moindre sur la stratégie. Un autre problème concerne le critère d'arrêt. Dans cette même situation l'entropie peut être nulle sans pour autant que la recherche ait abouti à un état de connaissance exacte. Cet inconvénient est inhérent à l'influence du mauvais *a priori* et peut être résolu de la même façon, c'est-à-dire par intégration virtuelle de l'*a priori*. Néanmoins une conclusion naturelle s'impose : mieux vaut ne pas fixer d'*a priori* que d'en fixer un qui soit faux.

Un inconvénient majeur de la méthode que nous avons proposée réside dans la taille de la loi conjointe qui vaut 2^n où n est le nombre de cellules sur chaque axe. L'algorithme tel que nous l'avons implémenté permet de traiter des échantillonnages de taille 8 en quelques secondes par itération. Mais il paraît peu envisageable d'augmenter la taille de la discrétisation, en tout cas pas indéfiniment. Une solution à cet accroissement de complexité est de proposer un schéma de recherche multirésolution. Chaque axe global peut être discrétisé de façon hiérarchique selon un arbre dont les cellules sont de plus en plus fines. La figure 5.12 est un exemple de décomposition hiérarchique binaire. En pratique, il serait vraisemblablement plus judicieux d'utiliser des décompositions de taille 4 ou 8. Une cellule à un niveau L définit un axe au niveau $L + 1$, ainsi les probabilités d'occupation à un niveau L permettent d'initialiser les lois conjointes au niveau $L + 1$:

$$\mathcal{P}(X_i^L) = 1 - \mathcal{P}(\cup\{X_j^{L+1} = 0\})$$

la somme étant prise sur l'ensemble des cellules de niveau $L + 1$ issues de X_i^L . La recherche s'effectue donc du niveau le plus grossier de l'arbre à son niveau le plus fin selon un scénario qui peut être le suivant : dans un premier temps, le système visuel effectue la recherche dans des zones très larges donc avec un fort angle de vue (par exemple en se plaçant à bonne distance des zones observées) et une précision de détection réduite. Puis, peu à peu, la recherche est localisée sur des zones précises, le capteur se rapproche et affine sa détection.

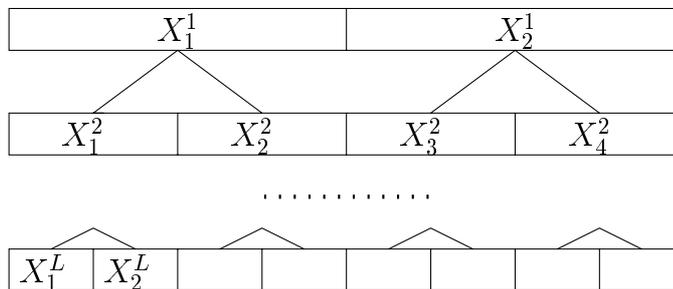


FIG. 5.12 – Chaque axe global peut être discrétisé de façon hiérarchique selon un arbre dont les cellules sont de plus en plus fines. Une cellule à un niveau L définit un axe au niveau $L + 1$.

Une autre amélioration possible se situe au niveau de la prise en compte des obser-

vations. Actuellement, lorsque des observations multiples apparaissent (c'est-à-dire que la caméra observe plusieurs cellules à la fois), aucune mémoire de leur dépendance n'est conservée. Or nous pouvons imaginer des situations pour lesquelles cette mémoire améliore la connaissance des probabilités d'occupation. Par exemple si la caméra observe deux axes en même temps et que l'observation indique qu'il se trouve au moins un objet sur les deux cellules observées, si la scène est peu dense, il est peu probable que les deux cellules soient occupées. Par conséquent, l'observation ultérieure de l'une ou l'autre de ces deux cellules fournira une indication sur la probabilité d'occupation de l'autre cellule. Cette situation est typiquement un problème de mémoire des observations précédentes. Un moyen simple d'y parvenir consiste à sauvegarder la loi conjointe de chaque observation multiple de sorte à pouvoir la modifier lorsque l'une des cellules est observée ultérieurement puis se servir du résultat pour modifier les autres cellules en conséquence.

Par ailleurs, les simulations que nous avons effectuées concernent uniquement des *a priori* sur le nombre de cellules occupées. Il serait intéressant de tester l'influence d'un *a priori* fonctionnel du type proximité des objets. Ce type de connaissance doit permettre de localiser la recherche dans certaines zones plus ou moins encombrées ou proches de certains objets déjà observés. Le gain vis-à-vis du nombre d'étapes nécessaires peut s'avérer important.

Enfin une perspective évidente de ces travaux est de tester l'algorithme en situation réelle. Le seul obstacle à sa réalisation est la robustesse de l'algorithme de détection utilisé. D'autre part l'ensemble des configurations atteignables n'est sans doute plus le voisinage cartésien que nous avons utilisé par commodité mais dépend de la conception du robot qui peut être non holonome ou bien contraint par des butées articulaires.

Reconstruction des objets

Dans le cadre de la reconstruction des objets, le choix d'estimer l'enveloppe visuelle nous a paru intéressant dans la mesure où elle utilise une information de contour des objets que l'on sait extraire dans un certain nombre de situations. Cependant, la description des enveloppes visuelles exactes est coûteuse. Le choix d'estimer un modèle approché de cette enveloppe est apparu plus judicieux et cohérent avec l'idée de concision que nous avons déjà présentée. Nous avons introduit une représentation mixte des objets qui mélange un modèle probabiliste gaussien et un modèle à erreur bornée. Le premier décrit la position de l'objet et l'incertitude associée. Le second modélise son enveloppe. Nous avons choisi d'approcher l'enveloppe visuelle par un ellipsoïde dont la représentation est simple et semblable aux loi gaussiennes de sorte que les règles de propagation associées au modèle apparaissent compactes et simples d'utilisation.

Chaque objet est ainsi représenté par une distribution probabiliste appelée loi d'ensemble. Celle-ci permet de calculer, pour chaque point de la scène, la probabilité d'appartenance à l'objet et d'extraire très facilement des caractéristiques géométriques telles que la position de l'objet et son enveloppe englobante. Ce modèle est peu coûteux puisque la distribution d'ensemble s'exprime sous une forme paramétrique simple, ne dépendant que d'un vecteur 3D et de deux matrices 3D symétriques. Un ensemble de règles de propagation associées à ce modèle et déduites de la géométrie probabiliste ainsi que du calcul ellipsoïdal nous a permis de construire un processus d'estimation rapide et robuste. Celui-ci est basé

sur l'estimation *a posteriori* des paramètres de localisation de l'objet et sur une approximation ellipsoïdale de l'intersection de deux ellipsoïdes en ce qui concerne le paramètre décrivant son enveloppe. Nous avons, ensuite, défini un processus d'exploration autonome et optimal en proposant deux solutions au calcul de la commande active : la commande LO (localement optimale) et la commande BVP (calculée à partir du meilleur point de vue). Chacune de ces lois de commande est calculée de sorte à minimiser l'information prédite *a posteriori*.

Notre approche se situe dans la classe des techniques d'estimation approchée de modèles. Elle diffère des travaux précédents sur plusieurs points. Tout d'abord, la décomposition additive que nous utilisons permet de représenter les objets sous forme d'un mélange de modèles stochastiques et à erreur bornée. Cette modélisation permet d'accéder simultanément à des informations probabilistes sur l'occupation des points de la scène et à des informations géométriques immédiates. D'autre part, la description du modèle dans un schéma probabiliste unifié le rend robuste à diverses sources d'incertitudes et fournit un cadre général à la fusion et à l'estimation approchée. Il nous a permis de développer des règles générales de propagation de la connaissance. En conséquence la méthode peut s'appliquer à n'importe quel capteur. L'idée sous-jacente à l'estimation du modèle tel que nous l'avons défini résulte d'une hypothèse d'indépendance entre l'incertitude de position et l'enveloppe de l'objet. Celle-ci nous permet de retomber sur un schéma classique d'estimation *a posteriori* de la position de l'objet et d'approximation ellipsoïdale du support de l'erreur. Ayant pu dériver une expression analytique de cette approximation, la formulation de l'estimation demeure relativement compacte et rapide d'utilisation. Par ailleurs, nous avons dérivé une solution analytique au problème de l'exploration de telle sorte que le calcul d'une trajectoire d'exploration réduisant l'incertitude du modèle devient un problème soluble en ligne. Même si la reconstruction n'est qu'approchée, répondre à la question *quel mouvement de la caméra améliorera notre connaissance de l'objet?* devient un problème bien posé.

Un intérêt pratique de nos travaux est leur applicabilité en temps réel et ce grâce à la vitesse des algorithmes rendue possible par la concision du modèle proposé. Un atout important de l'étude concerne les nombreuses simulations et expérimentations permettant de valider notre approche. L'algorithme s'est avéré très robuste vis-à-vis des incertitudes et capable, dans divers situations, de fournir une estimation très satisfaisante du modèle de l'objet. Une limitation évidente de l'algorithme réside dans le choix d'estimer un modèle approché de l'objet qui ne permet, en aucun cas, d'espérer une reconstruction fine. Toutefois nous revendiquons cette limitation dans la mesure où nous considérons que la concision du modèle est la clef d'un algorithme robuste et rapide.

Une autre limitation concerne le choix du paramètre α de pondération entre l'approximation externe et l'approximation interne de l'intersection de deux ellipsoïdes. Dans notre étude, le choix de ce paramètre est réalisé de façon empirique. Même si la valeur choisie $\alpha = 0.98$ semble convenir pour l'ensemble des simulations et expérimentations réalisées, il paraît évident que son choix dépend de l'angle entre deux points de vue successifs, de la vitesse de convergence, de la robustesse et de la précision finale souhaitées. Il serait donc intéressant d'envisager le calcul automatique de α . Par ailleurs nous avons considéré, pour le processus d'exploration, que l'intervalle de temps entre deux acquisitions était constant. Nous ne nous sommes pas préoccupés de la planification en temps des acquisitions. Ce travail mérite d'être étudié plus en détail. Nous pressentons que le choix de l'intervalle de

temps entre deux itérations dépend également de la vitesse de convergence, de la robustesse et de la précision finale souhaitées.

Un point abordé en introduction et non encore implémenté est la reconstruction d'objets fortement concaves. Au vu de nos développements, une telle reconstruction apparaît clairement réalisable si toutefois nous sommes capables de découper l'objet en parties cohérentes (non fortement concaves) et de suivre chacune de ces parties au cours du processus.

Enfin, une perspective intéressante à nos travaux se situe au niveau de l'estimation conjointe du modèle de l'objet et de la localisation du système visuel. La difficulté de cette estimation réside dans le couplage important qu'il existe entre la position de la caméra et les observations qu'il en résulte. Nous avons réalisé une pré-étude à ce sujet et avons montré sa faisabilité d'un point de vue théorique. Toutefois la validation de cette méthode par des simulations s'avère nécessaire dans la mesure où la stabilité de l'estimation reste à prouver.

Asservissement visuel

Enfin nous avons étudié la coopération caméra globale/caméra locale du point de vue de l'asservissement visuel. Cette étude est présentée dans l'annexe qui suit. Nous avons développé un schéma de coopération, intégrant une caméra fixe et une caméra mobile, en attribuant à chaque caméra une tâche particulière. Les consignes fixées s'énoncent en termes de positionnement et de suivi de l'objet d'intérêt. Le découpage naturel qui s'ensuit est le suivant : la caméra globale assure le positionnement en translation de la caméra locale tandis que celle-ci assure le suivi en rotation de l'objet qu'elle observe. Nous avons pu démontrer la stabilité de chacune des lois de commande prises individuellement. Lorsque les tâches sont réalisées simultanément, nous avons mis en évidence un phénomène prévu : la tâche de positionnement engendre une erreur de traînage pour la tâche de suivi. La coopération des deux tâches s'est donc avérée poser quelques problèmes de couplage. Pour pallier cette interaction nous avons suggéré plusieurs solutions basées soit sur une estimation de la perturbation, soit sur une technique utilisant la redondance de l'approche fonction de tâche. En ce qui concerne l'estimation de la perturbation, là encore plusieurs techniques sont utilisables. Nous avons proposé une estimation utilisant l'odométrie du robot, une estimation itérative et une méthode de comparaison entre l'observation et la prédiction. La plupart des techniques d'asservissement visuel ont donc été adaptées à la coopération et comparées d'un point de vue expérimental.

L'analyse et la réduction du couplage entre deux tâches d'asservissement visuel constitue notre principale contribution en ce qui concerne la coopération en asservissement visuel. L'originalité du travail ne provient pas des techniques employées qui sont toutes issues de l'asservissement visuel classique mais plutôt de leur utilisation dans un cadre coopératif et de leur comparaison exhaustive d'un point de vue expérimental. Un autre intérêt de l'étude est d'avoir pu montrer, en conservant l'indépendance entre les tâches, la stabilité des lois de commande que nous proposons.

Toutes les techniques envisagées pour réaliser un couplage entre les deux tâches proposées ont montré leur capacité à résoudre le problème considéré mais quelques unes telles que le filtre de Kalman ou l'odométrie, se sont avérées plus efficaces en termes de robustesse. La méthode utilisant une comparaison entre l'observation et la prédiction couplée

à un filtrage de Kalman a notre préférence dans la mesure où elle est indépendante de l'odométrie en translation du robot et qu'elle permet de suivre des objets en mouvement. Notons enfin qu'il est tout à fait possible de généraliser, à moindre coût, le formalisme de sorte que toutes les techniques employées permettent le suivi d'un objet mobile ; mais ce travail reste à faire.

Perspectives générales

Des trois études que nous avons présentées, la recherche d'objets, la reconstruction des objets et l'asservissement visuel, seules les deux dernières ont fait l'objet d'une intégration sinon totale tout au moins partielle. En effet le module de suivi des objets utilisé pour leur reconstruction est basé sur la méthode par filtrage de Kalman que nous avons expérimentée dans le cadre de la coopération en asservissement visuel. L'intégration est qualifiée de partielle car il n'est fait aucun usage de la caméra globale dans la boucle d'asservissement. Nous pourrions envisager son utilisation directement en phase de recherche des objets. Assurer le positionnement de la caméra locale permettrait de s'assurer qu'elle ne pénètre pas dans des zones inexplorées donc susceptibles de contenir des objets. Cette intégration ne semble pas poser de problèmes majeurs puisque toutes les fonctionnalités sont relativement bien découplées les unes par rapport aux autres.

Enfin, nous tenons à souligner que nos travaux ont été réalisés dans un souci constant d'applicabilité en temps réel. C'est pourquoi ils constituent une base solide à la réalisation d'autres tâches dont l'exécution sera facilitée par la reconstruction préliminaire de l'environnement que nous proposons. La possibilité de calculer, pour chaque point de la scène, une probabilité d'occupation se prête bien à des tâches de planification de trajectoires sans collision. La modélisation géométrique des objets peut également servir d'indicateur à une tâche plus fine de reconnaissance ou de manipulation des objets constituant la scène.

Annexes

Annexe A

Asservissement visuel et coopération

Dans cette partie, nous présentons un système intégrant une caméra déportée et une caméra embarquée pour la réalisation simultanée de deux tâches d'asservissement visuel. Dans un souci de robustesse, nous démontrons la stabilité de la loi de commande et comparons différentes techniques de coopération d'un point de vue expérimental.

A.1 Introduction

Durant les quinze dernières années, l'asservissement visuel 2D (c'est-à-dire basé sur des consignes définies dans l'image) embarqué a fait l'objet de nombreuses recherches [Weiss 84, Espiau 92, Hashimoto 93, Hutchinson 96]. Au cœur de cette approche, réside le jacobien image (ou matrice d'interaction). Il relie les variations de caractéristiques dans l'image, notées s , à la commande du robot, généralement exprimée par son torseur cinématique $T = [V, \Omega]$. Cette relation s'exprime très simplement lorsque s ne dépend que de la position r du robot :

$$\dot{s} = \frac{\partial s}{\partial r} \frac{dr}{dt} = LT \quad (\text{A.1})$$

Le jacobien image L apparaît alors tout naturellement. Souvent, la loi de commande peut être exprimée comme régulant une fonction de tâche [Samson 91]. Si s^* est la valeur désirée de s , l'objectif de l'asservissement est de contrôler le robot de sorte que $s - s^*$ décroisse vers 0. Une méthode simple pour y parvenir est d'appliquer la commande :

$$T = -\lambda \widehat{L}^+(s - s^*), \quad \lambda > 0 \quad (\text{A.2})$$

où \widehat{L}^+ est la pseudo-inverse du jacobien image estimé. Lorsque $\dim(s) \leq \dim(T)$, c'est-à-dire qu'il y a au plus autant de contraintes dans l'image que de degrés de liberté à commander, et dans le cas où \widehat{L} est recalculé à chaque itération, on peut montrer que (A.2) assure une décroissance exponentielle de $s - s^*$. Mais, \widehat{L} est souvent fixé à une matrice constante (calculée, en général, à la position désirée) notamment pour éviter de potentiels minima locaux et des mouvements irréalistes du robot [Chaumette 98b]. Néanmoins, il est toujours possible d'assurer, sous certaines conditions, la stabilité de la loi de commande (A.2).

De façon générale, la stabilité d'un asservissement caractérise la convergence du système vers son point d'équilibre (en l'occurrence la position pour laquelle $s = s^*$). La stabilité

asymptotique est dite locale lorsque cette propriété est vérifiée pour tout point d'un voisinage suffisamment proche du point d'équilibre ; elle est dite globale lorsque tout point de l'espace est asymptotiquement stable. Au vu des équations (A.1) et (A.2), le comportement de s est déterminé par l'équation différentielle suivante :

$$\dot{s} = -\lambda L \hat{L}^+(s - s^*)$$

Notons Q la matrice telle que l'état du système est régi par $\dot{s} = -\lambda Q(s - s^*)$. Le système est linéaire lorsque Q est indépendante de s . Dans ce cas, les notions de stabilité globale et de stabilité locale coïncident et une condition nécessaire et suffisante de stabilité asymptotique globale (et donc locale) est la positivité des parties réelles des valeurs propres de Q . Mais dans notre cas, $Q = L \hat{L}^+$ dépend de s (le système est non linéaire). La condition précédente ne fonction alors qu'au voisinage du point d'équilibre (par linéarisation) et ne caractérise, par conséquent, que la stabilité asymptotique locale. La stabilité asymptotique globale est alors, quant à elle, beaucoup plus difficile à caractériser. En pratique, nous disposons toutefois d'une condition suffisante qui est que Q doit être définie positive. Cette condition est évidemment plus forte puisqu'elle implique que $\|s - s^*\|$ décroisse à chaque itération.

Dans [Horaud 98], les résultats de [Espiau 92] sont étendus au cas où la caméra est déportée. Dans ce cas de figure, le jacobien image doit prendre en compte le passage du repère caméra au repère de commande du robot. Si l'on note $[R, t]$ le déplacement correspondant (R étant la matrice de rotation et t le vecteur de translation), le jacobien déporté L_g est relié au jacobien embarqué L par la relation :

$$L_g = -L \underbrace{\begin{pmatrix} R & -RS(-R^T t) \\ 0 & R \end{pmatrix}}_D \quad (\text{A.3})$$

où $S(a)$ est la matrice antisymétrique associée au vecteur a . D synthétise les formules bien connues liées au changement de point de réduction des torseurs et le changement de signe traduit le fait que la caméra est fixe et observe l'effecteur en mouvement. L'étude de la commande est identique au cas embarqué si bien que dans l'approche fonction de tâche, la loi de commande est similaire à (A.2).

Comme nous l'avons dit au chapitre 1, beaucoup d'études ont abordé l'utilisation de plusieurs caméras en vision par ordinateur mais rarement dans le cadre d'une coopération caméra globale / caméra locale [Marchand 98]. Le travail que nous décrivons dans cette partie constitue une première étape vers une telle coopération. Nous définissons et montrons la faisabilité d'une tâche d'asservissement visuel fusionnant une vue locale et une vue globale de la scène.

En section A.2, nous décrivons et modélisons précisément les tâches que chacune des caméras doit accomplir et montrons, en section A.3, comment les fusionner en une seule. Nous verrons que leur interaction naturelle ne peut pas être ignorée et décrivons comment la prendre en compte de manière efficace et robuste. En section A.4, nous présentons des résultats obtenus avec un robot cartésien six axes.

A.2 Description et modélisation de la tâche

La répartition des tâches, telle que nous la définissons dans la suite, découle des caractéristiques propres à chaque configuration. La caméra locale, du fait de son insertion

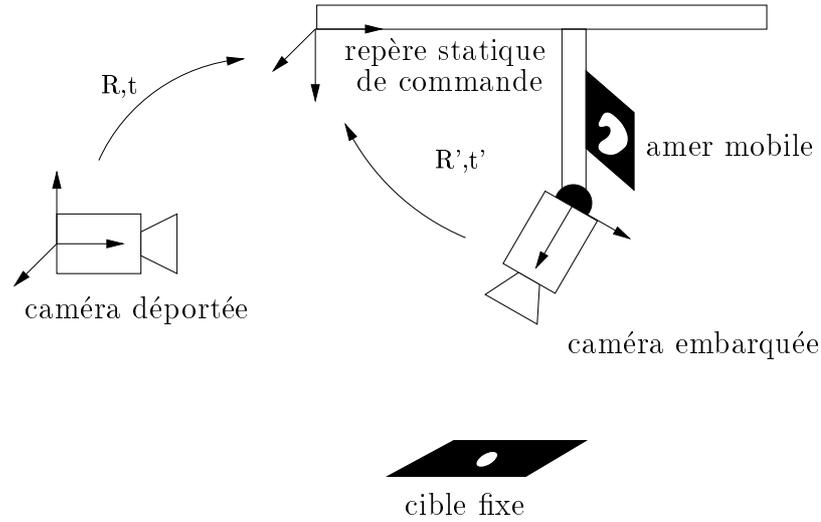


FIG. A.1 – Coopération caméra embarquée / caméra déportée

dans la scène et de sa mobilité, aura pour objectif de se focaliser sur les objets d'intérêt. La caméra déportée disposant, quant à elle, d'une vue globale de la scène est plus à même de contrôler les mouvements de translation et le positionnement de la caméra embarquée par rapport à son environnement. Ainsi, dans notre système (voir figures A.1 et A.2), la caméra globale est fixe et contrôle les degrés de translation du robot pour assurer le positionnement de l'effecteur (section A.2.1) alors que la caméra locale contrôle son orientation pour centrer une cible fixe dans son image (section A.2.2).

A.2.1 Commande de translation

L'amer mobile, d'où les caractéristiques de l'image globale sont extraites, est monté sur les organes de translation du robot. Ainsi les mouvements de rotation de l'effecteur n'ont pas d'influence sur l'image globale ce qui a pour effet de diminuer l'interaction des deux tâches. Pour assurer le positionnement de l'effecteur, l'image globale doit contrôler trois degrés de liberté. Nous devons donc extraire au moins trois caractéristiques indépendantes de l'image. Pour simplifier la démonstration de la stabilité, nous en choisissons uniquement trois : les coordonnées dans l'image du centre de gravité (c.d.g) de l'amer observé et la surface projetée. Nous supposons, d'autre part, que l'amer est plan et parallèle au plan de l'image. En conséquence, le c.d.g. peut être considéré comme un point physique et son évolution est régie par l'équation :

$$\begin{pmatrix} \dot{X}_G \\ \dot{Y}_G \end{pmatrix} = \begin{pmatrix} \frac{1}{z_G} & 0 & -\frac{X_G}{z_G} \\ 0 & \frac{1}{z_G} & -\frac{Y_G}{z_G} \end{pmatrix} \begin{pmatrix} \dot{x}_G \\ \dot{y}_G \\ \dot{z}_G \end{pmatrix} \quad (\text{A.4})$$

où (X_G, Y_G) sont les coordonnées du c.d.g. dans l'image correspondant à une focale de 1 et (x_G, y_G, z_G) sont les coordonnées du c.d.g. de l'amer exprimées dans le repère de la caméra. D'autre part, l'évolution de la surface projetée S est donnée par :

$$\dot{S} = -2 \frac{S}{z_G} \dot{z}_G \quad (\text{A.5})$$

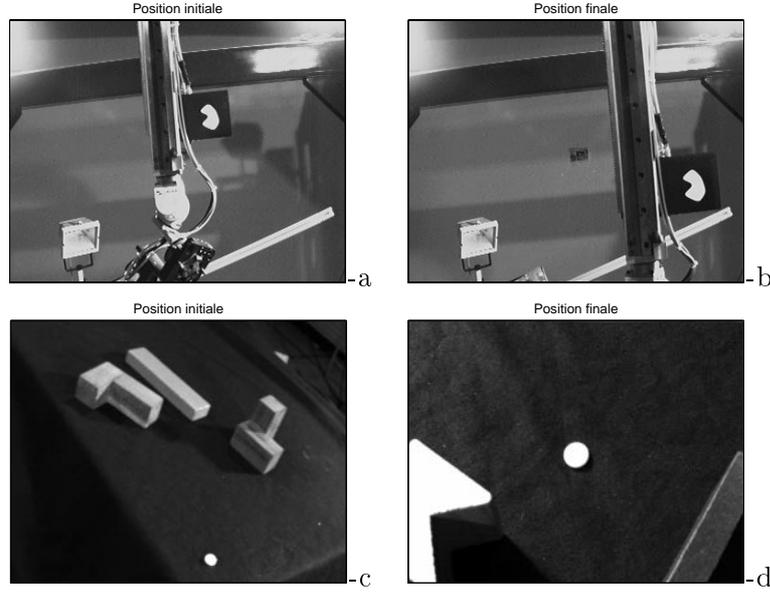


FIG. A.2 – Images initiales (à gauche) et désirées (à droite) pour le système déporté (ligne du haut) et le système embarqué (ligne du bas)

Preuve :

En effet, puisque l'amer est plan et parallèle au plan image, si la surface de l'amer vaut S_0 alors il est facile de montrer que la surface projetée sur le plan image vaut :

$$S = \frac{S_0}{z_G^2}$$

que l'on dérive en :

$$\dot{S} = -2 \frac{\dot{z}_G S_0}{z_G^3} = -2 \frac{S}{z_G} \dot{z}_G$$

□

Les relations (A.4) et (A.5) sont résumées dans l'équation suivante :

$$\begin{pmatrix} \dot{X}_G \\ \dot{Y}_G \\ \dot{S} \end{pmatrix} = \underbrace{\begin{pmatrix} \frac{1}{z_G} & 0 & -\frac{X_G}{z_G^2} \\ 0 & \frac{1}{z_G} & -\frac{Y_G}{z_G^2} \\ 0 & 0 & -\frac{2S}{z_G} \end{pmatrix}}_{L_0} \begin{pmatrix} \dot{x}_G \\ \dot{y}_G \\ \dot{z}_G \end{pmatrix} \quad (\text{A.6})$$

Nous choisissons d'exprimer la commande de translation V dans le repère statique de référence du robot. Le déplacement $[R, t]$ du repère de la caméra embarquée au repère de commande est donc constant. Si l'on note s_T le vecteur de caractéristiques, les équations (A.3) et (A.6) conduisent à :

$$\dot{s}_T = L_T V = L_0 R V \quad (\text{A.7})$$

L_T est le jacobien de translation. La loi de commande associée, visant une décroissance exponentielle de l'erreur, est donnée par :

$$V = -\lambda_T \widehat{L}_T^+ (s_T - s_T^*) \quad (\text{A.8})$$

où s_T^* est le vecteur des caractéristiques désirées et \widehat{L}_T est le jacobien estimé. Dans le but de choisir convenablement \widehat{L}_T , nous proposons de déterminer le domaine de stabilité asymptotique locale de la loi de commande (A.8) sous les hypothèses définies précédemment.

◆ Étude de la stabilité

Pour tenir compte de la calibration de la caméra, nous devons rappeler la transformation mètres/pixels. Si nous ignorons les distorsions radiales et supposons que les axes de l'image sont parfaitement orthogonaux, celle-ci s'exprime de la façon suivante :

$$\begin{pmatrix} \dot{X}_G \\ \dot{Y}_G \\ \dot{S} \end{pmatrix}_{\text{pixels}} = \underbrace{\begin{pmatrix} \frac{f}{l_x} & 0 & 0 \\ 0 & \frac{f}{l_y} & 0 \\ 0 & 0 & \frac{f^2}{l_x l_y} \end{pmatrix}}_A \begin{pmatrix} \dot{X}_G \\ \dot{Y}_G \\ \dot{S} \end{pmatrix}_m \quad (\text{A.9})$$

où f , l_x et l_y sont respectivement la focale, la largeur et la hauteur d'un pixel en mètres. D'après (A.7) et (A.9), le jacobien global de translation est du type $L = AL_0R$. Nous devons étudier la positivité des valeurs propres de $L\widehat{L}^+$ ($= L\widehat{L}^{-1}$ dans le cas présent¹). En notant \widehat{x} l'estimé de x et dR la matrice de rotation correspondant à l'erreur d'estimation de R de sorte que $R = dR.\widehat{R}$, nous obtenons la décomposition suivante :

$$L\widehat{L}^{-1} = AL_0 dR \widehat{L}_0^{-1} \widehat{A}^{-1}$$

Nous proposons d'étudier deux cas :

1. $R = \widehat{R}$ (i.e. $dR = \mathbb{I}$)
Dans ce cas ;

$$L\widehat{L}^{-1} = AL_0 \widehat{L}_0^{-1} \widehat{A}^{-1} = \begin{pmatrix} \frac{f\widehat{z}_G \widehat{l}_x}{l_x z_G f} & 0 & \frac{1}{2} \frac{f\widehat{z}_G \widehat{l}_x \widehat{l}_y (X_G - \widehat{X}_G)}{l_x z_G \widehat{S} f^2} \\ 0 & \frac{f\widehat{z}_G \widehat{l}_y}{l_y z_G f} & \frac{1}{2} \frac{f\widehat{z}_G \widehat{l}_x \widehat{l}_y (Y_G - \widehat{Y}_G)}{l_y z_G \widehat{S} f^2} \\ 0 & 0 & \frac{f^2 \widehat{l}_y \widehat{S} \widehat{z}_G \widehat{l}_x}{l_x z_G \widehat{S} f^2 l_y} \end{pmatrix}$$

et les valeurs propres de $L\widehat{L}^{-1}$ sont :

$$\left(\frac{f\widehat{z}_G \widehat{l}_x}{l_x z_G f}, \frac{f\widehat{z}_G \widehat{l}_y}{l_y z_G f}, \frac{f^2 \widehat{z}_G \widehat{l}_x \widehat{l}_y \widehat{S}}{l_x z_G \widehat{S} f^2 l_y} \right)$$

En conséquence, la stabilité asymptotique locale du système est assurée puisque la condition

$$f, \widehat{f}, l_x, l_y, \widehat{l}_x, \widehat{l}_y, z_G, \widehat{z}_G, S \text{ et } \widehat{S} > 0$$

n'est pas restrictive et ce quelles que soient les estimations \widehat{X}_G et \widehat{Y}_G .

1. \widehat{L} est inversible en tant que produit de matrices carrées régulières.

2. $A = \widehat{A}$ et $L_0 = \widehat{L}_0$

Sous ces conditions, $L\widehat{L}^{-1} = AL_0dRL_0^{-1}A^{-1}$ ce qui traduit que dR et $L\widehat{L}^{-1}$ sont semblables et ont les mêmes valeurs propres. Nous devons simplement étudier la positivité des parties réelles des valeurs propres de dR qui sont

$$(1, e^{i\theta}, e^{-i\theta})$$

où θ est l'angle de rotation entre R et \widehat{R} . Ceci signifie que, quel que soit l'axe de rotation, le système est localement asymptotiquement stable si

$$-\frac{\pi}{2} < \theta < \frac{\pi}{2}$$

□

À en juger par cette étude de stabilité, nous pouvons nous permettre de fixer \widehat{L}_T à une valeur constante tout en assurant la convergence de la commande. Dans notre application, nous avons choisi de fixer \widehat{X}_G , \widehat{Y}_G et \widehat{z}_G à leur valeur désirée, \widehat{z}_G étant grossièrement "estimé à la main". La valeur constante de R est également grossièrement estimée.

A.2.2 Commande en rotation

L'objectif de la seconde tâche est de contrôler l'orientation de la caméra embarquée de telle sorte que le centre d'une cible fixe apparaisse au centre de l'image (voir figure A.2). Cette tâche assure la visibilité de la cible durant le positionnement. Si l'objet observé est un point alors le mouvement de sa projection (X, Y) est relié à la commande de rotation $\Omega = [\Omega_X, \Omega_Y, \Omega_Z]$ par :

$$\begin{pmatrix} \dot{X} \\ \dot{Y} \end{pmatrix} = \underbrace{\begin{pmatrix} XY & -(1+X^2) & Y \\ 1+Y^2 & -XY & -X \end{pmatrix}}_{L_R} \Omega$$

L_R est le jacobien de rotation. Si $s_R = (X, Y)$ représente le vecteur des caractéristiques embarquées courantes, s_R^* celui désiré et \widehat{L}_R le jacobien estimé, la commande de suivi est donnée par :

$$\Omega = -\lambda_R \widehat{L}_R^+ (s_R - s_R^*) \quad (\text{A.10})$$

Comme précédemment, nous pouvons montrer que le domaine de stabilité de cette commande est très large.

◆ Étude de la stabilité

Si on nomme B la matrice de transformation mètres/pixels :

$$B = \begin{pmatrix} \frac{f}{l_x} & 0 \\ 0 & \frac{f}{l_x} \end{pmatrix}$$

alors nous devons étudier la positivité de $BL_R(\widehat{B}\widehat{L}_R)^+$ où $A^+ = A^T(AA^T)^{-1}$ si bien que :

$$BL_R(\widehat{B}\widehat{L}_R)^+ = \begin{pmatrix} \frac{f\widehat{l}_x(1+X^2+Y\widehat{Y})}{\widehat{f}l_x(1+\widehat{X}^2+\widehat{Y}^2)} & \frac{f\widehat{l}_y Y(X-\widehat{X})}{\widehat{f}l_x(1+\widehat{X}^2+\widehat{Y}^2)} \\ \frac{f\widehat{l}_x X(Y-\widehat{Y})}{\widehat{f}l_y(1+\widehat{X}^2+\widehat{Y}^2)} & \frac{f\widehat{l}_y(1+X\widehat{X}+Y^2)}{\widehat{f}l_y(1+\widehat{X}^2+\widehat{Y}^2)} \end{pmatrix}$$

Dans notre application, \widehat{L}_R est calculé une fois pour toutes en posant $\widehat{X} = 0$ et $\widehat{Y} = 0$ auquel cas $BL_R(\widehat{B}\widehat{L}_R)^+$ se simplifie en :

$$\begin{pmatrix} \frac{f\widehat{l}_x(1+X^2)}{\widehat{f}\widehat{l}_x} & \frac{f\widehat{l}_y Y X}{\widehat{f}\widehat{l}_x} \\ \frac{f\widehat{l}_x X Y}{\widehat{f}\widehat{l}_y} & \frac{f\widehat{l}_y(1+Y^2)}{\widehat{f}\widehat{l}_y} \end{pmatrix}$$

Pour que les valeurs propres de la matrice précédente soient positives, il faut et il suffit que sa trace et son déterminant soient positifs. Or :

$$\begin{cases} \text{trace}(BL_R(\widehat{B}\widehat{L}_R)^+) &= \frac{f\widehat{l}_x(1+X^2)}{\widehat{f}\widehat{l}_x} + \frac{f\widehat{l}_y(1+Y^2)}{\widehat{f}\widehat{l}_y} \\ \det(BL_R(\widehat{B}\widehat{L}_R)^+) &= \frac{f^2\widehat{l}_x\widehat{l}_y(1+X^2+Y^2)}{\widehat{f}^2\widehat{l}_x\widehat{l}_y} \end{cases}$$

Une condition nécessaire et suffisante de stabilité asymptotique locale est donc :

$$f, \widehat{f}, l_x, \widehat{l}_x, l_y, \widehat{l}_y > 0$$

□

Comme précédemment, cette étude montre qu'il est tout à fait possible de fixer \widehat{L}_R tout en assurant la stabilité de la boucle de commande. Notons toutefois que le calcul de \widehat{L}_R à chaque itération serait également très simple à implémenter.

A.2.3 Interaction

Alors que les mouvements de rotation n'influencent pas les caractéristiques de l'image globale, les mouvements de translation de l'effecteur induisent des mouvements dans l'image embarquée. Plus précisément, le mouvement de la projection de la cible fixe est relié à la commande en translation par :

$$\dot{s}_R = \underbrace{\begin{pmatrix} -1/z & 0 & X/z \\ 0 & -1/z & Y/z \end{pmatrix}}_{L_{RT}} R' V$$

où z est la profondeur de la cible fixe dans le repère de la caméra locale et R' est la rotation entre ce repère et le repère de commande en translation. Pour estimer \widehat{L}_{RT} , nous fixons \widehat{X} et \widehat{Y} à leur valeur désirée 0 et \widehat{z} à une valeur constante z^* . R' est calculé grâce à l'odométrie du robot.

A.3 Coopération

À présent, nous souhaitons faire coopérer les tâches de rotation et de translation dans un schéma de commande unifié. D'après la modélisation précédente, le jacobien de la tâche globale est :

$$\widehat{L} = \begin{pmatrix} \widehat{L}_T & 0 \\ \widehat{L}_{RT} & \widehat{L}_R \end{pmatrix} = \begin{pmatrix} \widehat{L}_d \\ \widehat{L}_e \end{pmatrix} \begin{cases} \text{déporté} \\ \text{embarqué} \end{cases}$$

Lorsque l'on utilise la commande (A.2) où \widehat{L} présente un terme de couplage ($\widehat{L}_{RT} \neq 0$), on observe des comportements indésirables dans l'image (voir figure A.3) avec, éventuellement,

une perte des amers si \widehat{L} n'est pas parfaitement estimé. Ces craintes ont pu être vérifiées au cours d'expérimentations ; c'est pourquoi nous souhaitons conserver le maximum d'indépendance dans la commande des tâches. De plus, si l'indépendance est préservée, la stabilité de la tâche globale sera assurée tant que chacune d'elles reste stable.

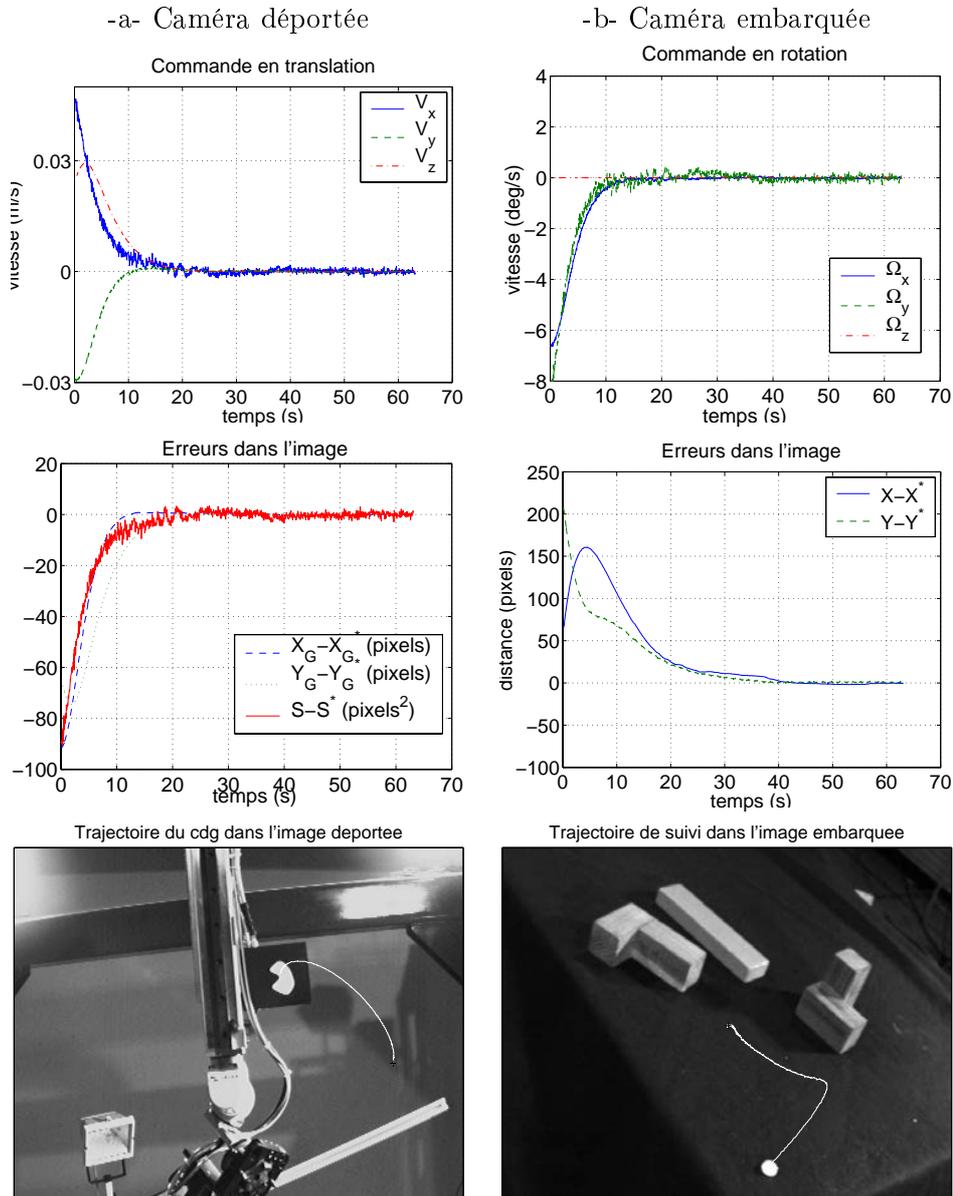


FIG. A.3 – Commande couplée

En revanche, fixer $\widehat{L}_{RT} = 0$ introduit une imprécision de modélisation qui conduit à des erreurs de suivi (figure A.4-b), parfois nommées erreurs de traînage. Le problème est d'autant plus dangereux que les amers peuvent, là encore, sortir de l'image. Afin de supprimer cette erreur de suivi tout en conservant l'indépendance, nous nous proposons de comparer deux classes de méthodes : des méthodes basées sur une estimation de la

perturbation et une méthode introduisant la notion de redondance de tâches.

A.3.1 Estimation de la perturbation

La translation de l'effecteur peut être vue comme une perturbation inconnue agissant sur les caractéristiques embarquées. À cause de cette perturbation inconnue sur s_R , l'équation (A.1) n'est plus valide puisque s_R ne dépend plus de la seule position du robot mais également du temps : $s_R(r,t)$. Ce qui se dérive en :

$$\dot{s}_R = \frac{\partial s_R}{\partial r} \frac{dr}{dt} + \frac{\partial s_R}{\partial t} = L_R \Omega + \frac{\partial s_R}{\partial t}$$

Si l'on souhaite assurer une décroissance exponentielle de $s_R - s_R^*$, la commande correspondante est donnée par [Chaumette 93] :

$$\Omega = \underbrace{-\lambda_R \widehat{L}_R^+ (s_R - s_R^*)}_{\text{loi de commande classique}} \quad \underbrace{-\widehat{L}_R + \frac{\partial s_R}{\partial t}}_{\text{compensation de l'erreur de traînage}} \quad (\text{A.11})$$

où l'on reconnaît un terme de commande classique identique à (A.2) et un terme lié à la compensation de l'erreur. Il faut donc estimer la part des variations de l'image due à la perturbation, c'est-à-dire $\widehat{\partial s_R / \partial t}$. Plusieurs techniques peuvent permettre de réaliser cette estimation :

Odométrie : Si l'on a accès à l'odométrie du robot, V peut être mesurée et l'estimation est donnée par $\widehat{(\frac{\partial s_R}{\partial t})} = L_{RT} V$. Malheureusement, peu de robots peuvent se permettre une mesure précise de leurs mouvements de translation. Les robots mobiles ont, par exemple, une odométrie en translation peu fiable.

Estimation itérative : Cette méthode nous vient directement de la théorie de la commande et consiste à insérer un intégrateur dans la boucle de commande. L'intégration est réalisée par le schéma itératif suivant :

$$\widehat{(\frac{\partial s_R}{\partial t})}_{k+1} = \widehat{(\frac{\partial s_R}{\partial t})}_k + \mu (s_{Rk} - s_R^*) \quad (\text{A.12})$$

où μ est un gain de compensation. Nous pouvons vérifier que, lorsque la stationnarité est atteinte, c'est-à-dire pour

$$\widehat{(\frac{\partial s_R}{\partial t})}_{k+1} = \widehat{(\frac{\partial s_R}{\partial t})}_k$$

alors $s_{Rk} = s_R^*$, l'erreur de traînage est compensée.

Observation/Prédiction : Une autre méthode consiste à comparer la vitesse observée dans l'image et la vitesse prédite à partir de la commande précédente. La différence de ces deux termes doit correspondre à la variation due à la perturbation. À l'instant k , l'estimation est :

$$\widehat{(\frac{\partial s_R}{\partial t})}_k = \underbrace{\frac{s_{Rk}}{\Delta t}}_{\text{observation}} - \underbrace{\frac{s_{Rk/k-1}}{\Delta t}}_{\text{prédiction}}$$

où $s_{Rk/k-1} = s_{Rk-1} + L_R \Omega_{k-1} \Delta t$. Donc :

$$\widehat{\left(\frac{\partial s_R}{\partial t}\right)}_k = \frac{s_{Rk} - s_{Rk-1}}{\Delta t} - L_R \Omega_{k-1} \quad (\text{A.13})$$

À l'instant k , la connaissance de Ω_{k-1} peut être obtenue soit en supposant que Ω_{k-1} est la commande précédente, soit en la mesurant. Cette dernière technique est celle que nous avons implémentée puisque la première suppose un temps de réponse idéal du système².

L'estimation donnée par (A.13) serait parfaite si les données n'étaient pas bruitées. Puisque ce schéma n'est pas itératif, il ne filtre pas le bruit sur les mesures de s_R et Ω . Pour le filtrer, nous pouvons utiliser soit un simple filtre avec facteur d'oubli :

$$\widehat{\left(\frac{\partial s_R}{\partial t}\right)}_k = \alpha \left(\frac{s_{Rk} - s_{Rk-1}}{\Delta t} - L_R \Omega_{k-1} \right) + (1 - \alpha) \widehat{\left(\frac{\partial s_R}{\partial t}\right)}_{(k-1)} \quad (\text{A.14})$$

où α est le facteur d'oubli, soit un filtre de Kalman comme celui proposé dans [Chauvette 93]. Évidemment, cette technique implique que l'on connaisse une approximation du modèle d'évolution de $\frac{\partial s_R}{\partial t}$ mais il est bien connu que le filtre de Kalman est assez robuste vis-à-vis de cette modélisation. Nous avons choisi un modèle à vitesse constante et bruit corrélé.

Un atout important des techniques d'observation/prédiction est qu'elles fonctionnent également pour le suivi de cibles mobiles. Dans ce cas, on ne fait pas de différence entre la perturbation due à la translation de l'effecteur et celle due au mouvement de la cible, mais ceci n'influence pas le comportement du système.

A.3.2 Redondance de tâches

Une autre méthode est de considérer la tâche de suivi en rotation comme tâche principale. Puisqu'elle ne contraint pas tous les degrés de liberté du robot, nous pouvons utiliser le formalisme de la redondance de l'approche fonction de tâche [Samson 91, Espiau 92]. Une tâche secondaire g_s peut alors être réalisée conjointement en régulant la fonction de tâche :

$$e = W^+(s_R - s_R^*) + (\mathbb{I} - W^+W)\alpha_2 g_s^T$$

où W est une matrice de rang plein telle que $\text{Ker } W = \text{Ker } \widehat{L}_e$ si on note $\widehat{L}_e = [\widehat{L}_{RT} \quad \widehat{L}_R]$. La matrice $\mathbb{I} - W^+W$ est donc un opérateur de projection sur le noyau de \widehat{L}_e . Cela signifie que la réalisation de la tâche secondaire n'aura pas d'effet sur la tâche principale tant que \widehat{L}_e est une bonne approximation de L_e .

Dans notre cas, le jacobien de la tâche principale $L_e = (L_{RT} \quad L_R)$ est de rang plein donc nous pouvons poser $W = \widehat{L}_e$. De plus, la tâche secondaire est une tâche de positionnement dans l'image déportée :

$$g_s = C(s_T - s_T^*)$$

2. Il est important de noter que les mouvements de rotation sont toujours plus faciles à mesurer précisément que les mouvements de translation.

où C est une matrice de combinaison. Par application de la commande $T = -\lambda e$, l'évolution de s_T lorsque s_R a convergé est donnée par :

$$\dot{s}_T = -\lambda \alpha_2 L_d (\mathbb{I} - \widehat{L}_e^+ \widehat{L}_e) C (s_T - s_T^*)$$

Une condition nécessaire à la décroissance exponentielle de s_T est que

$$C = (\widehat{L}_d (\mathbb{I} - \widehat{L}_e^+ \widehat{L}_e))^+$$

En conclusion, la fonction de tâche s'écrit :

$$e = \widehat{L}_e^+ (s_R - s_R^*) + (\mathbb{I} - \widehat{L}_e^+ \widehat{L}_e) (\widehat{L}_d (\mathbb{I} - \widehat{L}_e^+ \widehat{L}_e))^+ \alpha_2 (s_T - s_T^*)$$

A.4 Résultats expérimentaux

Cette section et les figures A.4 à A.5 montrent les résultats expérimentaux obtenus pour chacune des méthodes citées. Les gains de commande ont été choisis de la façon suivante : $\lambda_T = \lambda_R = 0.1$. L'amer mobile d'où l'on extrait les caractéristiques déportées s_T est la tâche de forme complexe et inconnue que l'on aperçoit figure A.2-a. s_T^* a été préalablement appris (figure A.2-b) mais aurait tout aussi bien pu être fixé par un opérateur.

Lorsque l'on contrôle chaque tâche indépendamment, sans compensation (figure A.4), la translation adopte un comportement exponentiel alors que la rotation ne peut pas s'achever tant que la translation n'a pas convergé. Ceci correspond à une erreur de traînage.

A.4.1 Estimation et compensation de l'erreur

Pour toutes ces méthodes, la commande en translation ne change pas et ses performances sont identiques à celles présentées figure A.4. Les résultats obtenus en utilisant un intégrateur sont présentés figure A.5-a, le gain étant fixé à $\mu = 0.01$. Cette méthode améliore la vitesse de convergence mais μ est assez difficile à ajuster. Les performances ne sont pas optimales du fait des variations de vitesse.

L'utilisation du filtre (A.14) pour lequel nous avons fixé $\alpha = 0.7$ laisse apparaître un bruit résiduel significatif (figure A.5-b) sur la commande même si l'effet de ce bruit sur les caractéristiques image est naturellement filtré par la dynamique du robot. Le filtre de Kalman a, quant à lui, de très bonnes performances (figure A.5-c). Enfin, l'utilisation de l'odométrie semble également très efficace (figure A.5-d) mais ses performances dépendent principalement de la précision des mesures du robot.

A.4.2 Formalisme de la redondance

Le formalisme de la redondance présente l'avantage de spécifier l'importance d'une tâche par rapport à l'autre. La figure A.6 montre que la tâche secondaire suit une convergence exponentielle à partir du moment où la tâche primaire a convergé. La réalisation de la tâche primaire semble influencée, au cours des premières itérations, par la tâche secondaire dans la mesure où son noyau n'est pas parfaitement estimé.

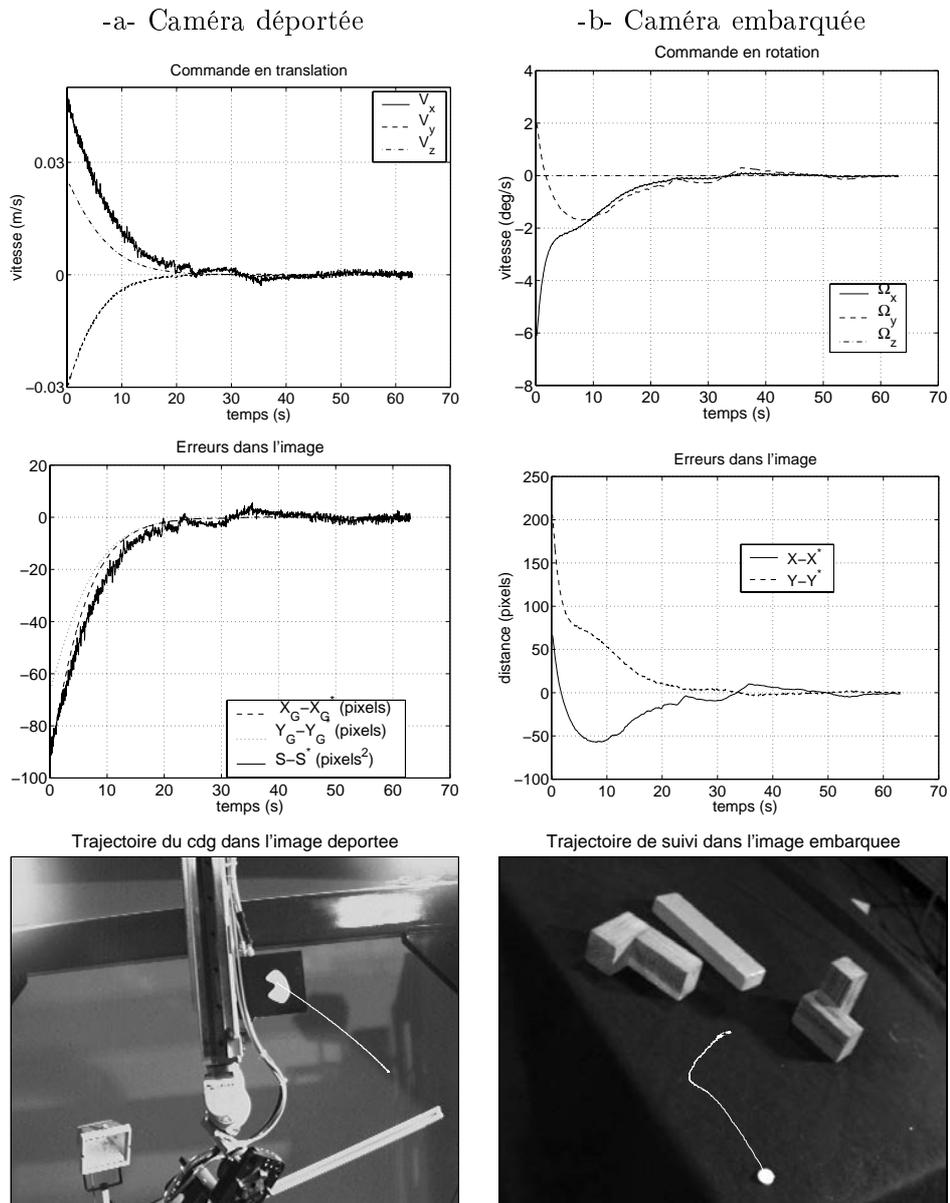
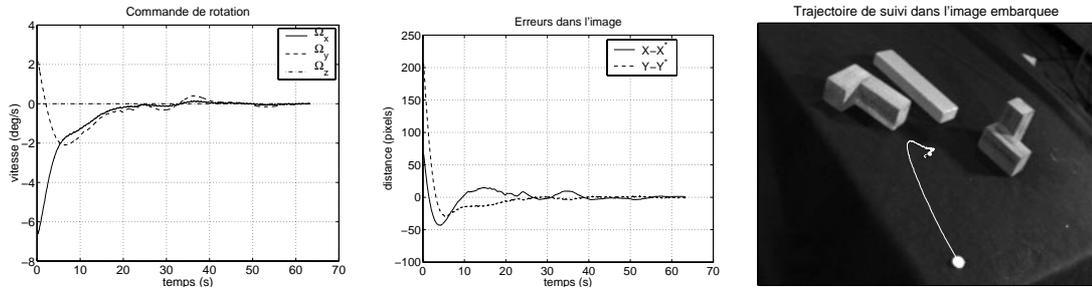
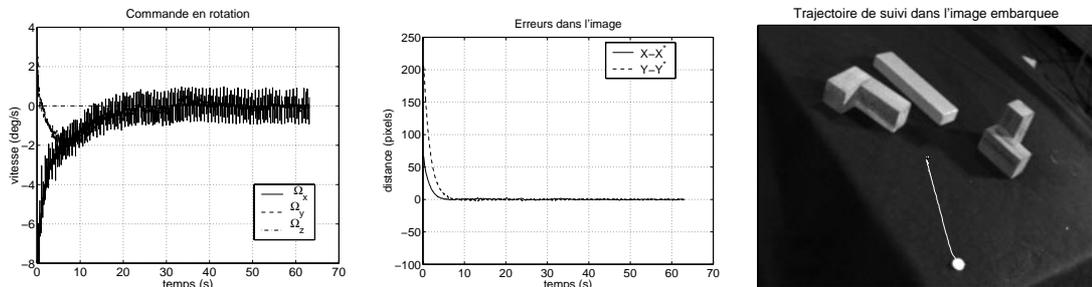


FIG. A.4 – Commande indépendante des tâches

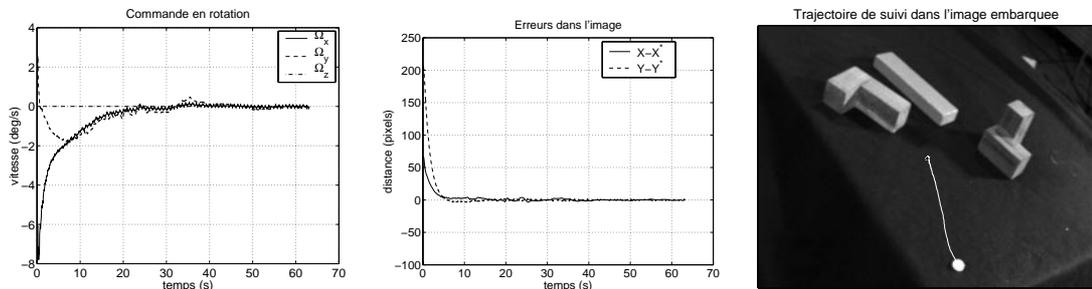
-a- Compensation par intégration



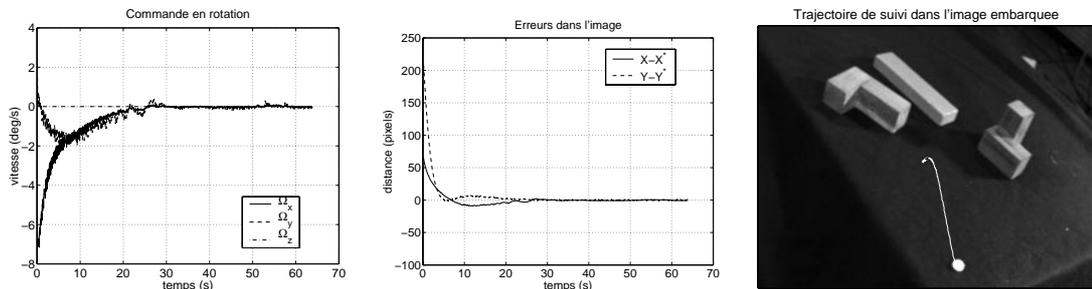
-b- Compensation par estimation + filtrage simple



-c- Compensation par estimation + filtrage de Kalman



-d- Compensation par odométrie

FIG. A.5 – Tâche de suivi utilisant une estimation de $\frac{\partial s_R}{\partial t}$

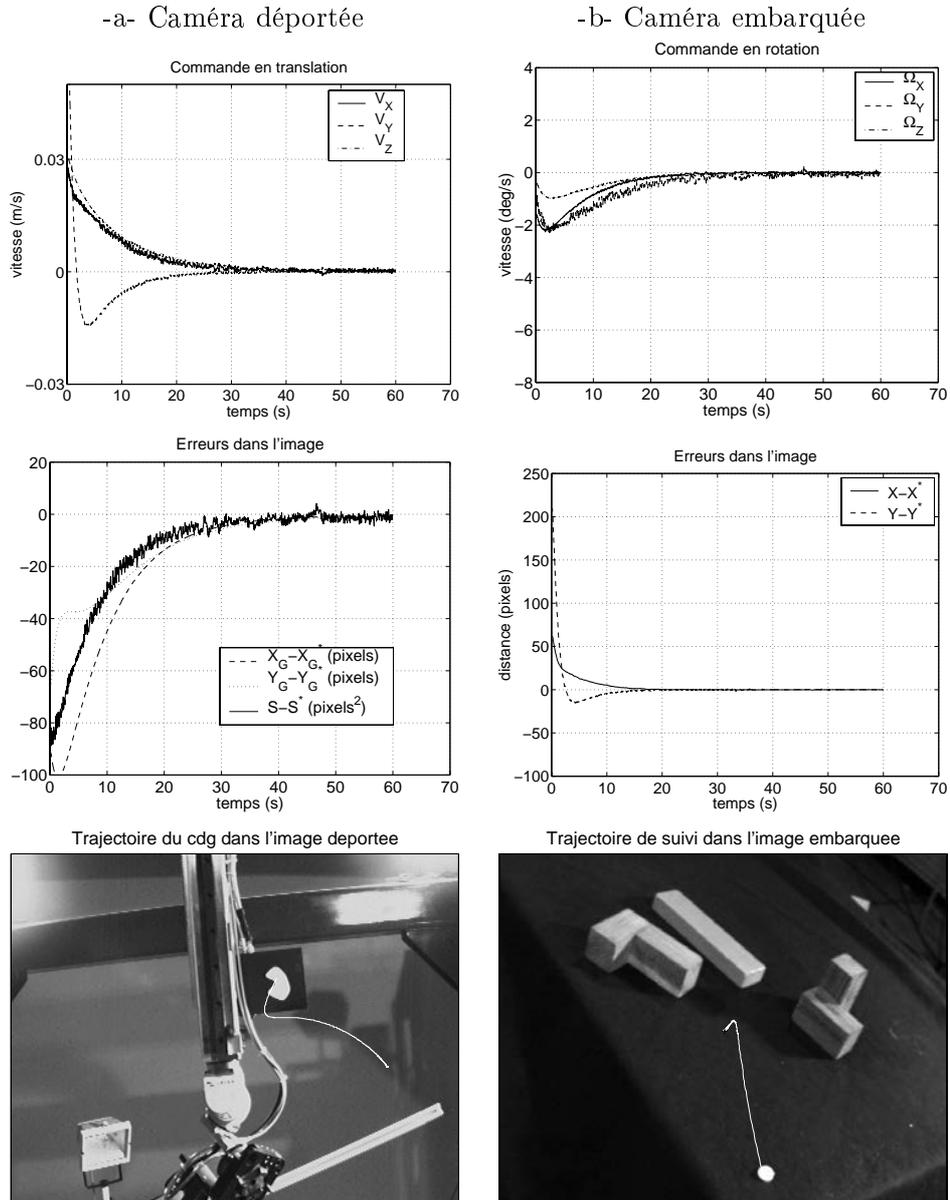


FIG. A.6 – Utilisation du formalisme de la redondance

A.5 Conclusion

La plupart des techniques d'asservissement visuel ont été adaptées à la coopération et comparées expérimentalement. Toutes ont montré leur capacité à résoudre le problème considéré mais quelques unes telles que le filtre de Kalman ou l'odométrie, se sont avérées plus efficaces en terme de robustesse. La méthode basée sur la comparaison observation/prédiction et filtrage de Kalman présente le double avantage de ne pas utiliser l'odométrie en translation du robot, qui, on le sait, est délicate à obtenir précisément (surtout en robotique mobile) et de pouvoir suivre les objets mobiles. Notre choix s'est donc porté tout naturellement sur cette méthode.

Annexe B

Changement de repère et incertitudes d'un robot cartésien 6 axes

Cette annexe concerne la détermination des incertitudes de positionnement d'un robot cartésien 6 axes tel que celui que nous utilisons lors des expérimentations. Nous présentons là une modélisation assez fine du robot, permettant de spécifier les incertitudes au niveau le plus bas de la chaîne cinématique et de la géométrie du robot : la position des articulations et la calibration pince/caméra. Une modélisation plus frustre, permettant de modéliser les incertitudes directement au niveau du positionnement de la caméra, pourrait malgré tout convenir mais ne tiendrait pas compte de la géométrie du robot.

Comme l'indique la figure B.1, le robot est constitué d'une succession de 3 articulations de translation dont les positions sont repérées par q_1 , q_2 et q_3 et de 3 articulations de rotation q_4 , q_5 et q_6 faisant passer des repères \mathcal{R}_3 à \mathcal{R}_4 , \mathcal{R}_4 à \mathcal{R}_5 , et \mathcal{R}_5 à \mathcal{R}_6 . Les repères \mathcal{R}_3 , \mathcal{R}_4 et \mathcal{R}_5 ont la même origine alors que \mathcal{R}_6 est une composition de rotation et translation (L) de \mathcal{R}_5 . Le repère \mathcal{R}_0 est le repère fixe de référence. Le repère \mathcal{R}_7 est le repère de la caméra, différent du repère de l'effecteur \mathcal{R}_6 . Les sources d'incertitudes possibles sont les articulations q , la longueur L et les paramètres du déplacement D_{67} entre \mathcal{R}_6 et \mathcal{R}_7 .

Dans la règle 6, nous avons exprimé la position de la caméra par un déplacement R, t entre le repère de référence et le repère de la caméra. L'expression c^o de l'observation dans le repère de référence est donc fonction de son expression dans le repère de la caméra par la relation :

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

où γ est le vecteur des paramètres externes, c'est-à-dire la concaténation des composantes de la matrice de rotation et du vecteur de translation.

$$\gamma = (r_{11}, r_{12}, r_{13}, r_{21}, r_{22}, r_{23}, r_{31}, r_{32}, r_{33}, t_1, t_2, t_3)^T$$

Nous devons exprimer γ en fonction des paramètres physiques du robot qui sont la valeur des articulations q , la longueur du bras L et la calibration caméra/effecteur D_{67} . Or, en utilisant les coordonnées homogènes, nous pouvons récrire le déplacement entre \mathcal{R}_0 et \mathcal{R}_7

de la façon suivante :

$$\begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix} = D_{04}D_{45}D_{56}D_{67}$$

où chaque déplacement s'écrit en fonction des articulations :

$$D_{04} = \begin{pmatrix} \cos q_4 & -\sin q_4 & 0 & q_1 \\ \sin q_4 & \cos q_4 & 0 & q_2 \\ 0 & 0 & 1 & q_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad D_{45} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ \cos q_5 & -\sin q_5 & 0 & 0 \\ \sin q_5 & \cos q_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$D_{56} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ \cos q_6 & -\sin q_6 & 0 & 0 \\ \sin q_6 & \cos q_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

excepté D_{67} que l'on note pour l'instant :

$$D_{67} = \begin{pmatrix} d_{11} & d_{12} & d_{13} & d_{14} \\ d_{21} & d_{22} & d_{23} & d_{24} \\ d_{31} & d_{32} & d_{33} & d_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Si l'on note :

$$p_{67} = (d_{11}, d_{12}, d_{13}, d_{21}, d_{22}, d_{23}, d_{31}, d_{32}, d_{33}, d_{14}, d_{24}, d_{34})^T$$

il apparaît que γ est fonction de q , L et p_{67} :

$$\gamma = f(q, L, p_{67})$$

Si l'on modélise q , L et p_{67} par des lois normales de covariances respectives P_q , P_L et $P_{p_{67}}$, la matrice de covariance de γ se calcule de la façon suivante :

$$\Gamma = \frac{\partial f}{\partial q} P_q \frac{\partial f^T}{\partial q} + \frac{\partial f}{\partial L} P_L \frac{\partial f^T}{\partial L} + \frac{\partial f}{\partial p_{67}} P_{p_{67}} \frac{\partial f^T}{\partial p_{67}}$$

p_{67} n'est toujours pas une représentation minimale du déplacement D_{67} . Nous cherchons donc à l'exprimer sous la forme :

$$p_{67} = g(\alpha_x, \alpha_y, \alpha_z, t_x, t_y, t_z)$$

où $\alpha = (\alpha_x, \alpha_y, \alpha_z)$ représente, par exemple, les angles d'Euler de la rotation et $t = (t_x, t_y, t_z)$ la translation. Auquel cas on peut écrire :

$$P_{p_{67}} = \frac{\partial g}{\partial(\alpha, t)} P_{(\alpha, t)} \frac{\partial g^T}{\partial(\alpha, t)}$$

Finalement :

$$\Gamma = \frac{\partial f}{\partial q} P_q \frac{\partial f^T}{\partial q} + \frac{\partial f}{\partial L} P_L \frac{\partial f^T}{\partial L} + \frac{\partial f}{\partial p_{67}} \frac{\partial g}{\partial(\alpha, t)} P_{(\alpha, t)} \frac{\partial g^T}{\partial(\alpha, t)} \frac{\partial f^T}{\partial p_{67}}$$

Il ne nous apparaît pas utile de donner l'expression des matrices $\frac{\partial f}{\partial q}$, $\frac{\partial f}{\partial L}$, $\frac{\partial f}{\partial p_{67}}$ et $\frac{\partial g}{\partial(\alpha,t)}$ qui sont relativement lourdes mais s'obtiennent aisément à l'aide d'un logiciel de calcul formel. En revanche la modélisation des covariances sur les différents bruits est estimée de la façon suivante :

- si l'écart-type sur les mesures des articulations vaut α en ce qui concerne les translations et β pour les rotations alors :

$$P_q = \begin{pmatrix} \alpha^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \alpha^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \beta^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \beta^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \beta^2 \end{pmatrix}$$

Le choix de α et β dépend de la situation. En simulation, lorsque nous avons testé la robustesse de l'estimation face à de grandes valeurs du bruit, nous avons choisi $\alpha = 10cm$ et $\beta = 5^\circ$. En ce qui concerne la modélisation des incertitudes de positionnement de la plate-forme robotique, nous avons choisi $\alpha = 1mm$ et $\beta = 1^\circ$.

- l'écart-type sur la longueur L est fixé à $1mm$

$$P_L = 0.000001$$

- l'écart-type sur la calibration caméra/effecteur est fixée à $1mm$ pour les translations et 1° pour les rotations.

$$P_{(\alpha,t)} = \begin{pmatrix} 0.000001 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.000001 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.000001 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.0003 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.0003 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.0003 \end{pmatrix}$$

En ce qui concerne les simulations, la calibration caméra/effecteur n'intervient pas, ce qui revient à fixer $P_L = 0$ et toutes les valeurs de $P_{(\alpha,t)}$ à 0.

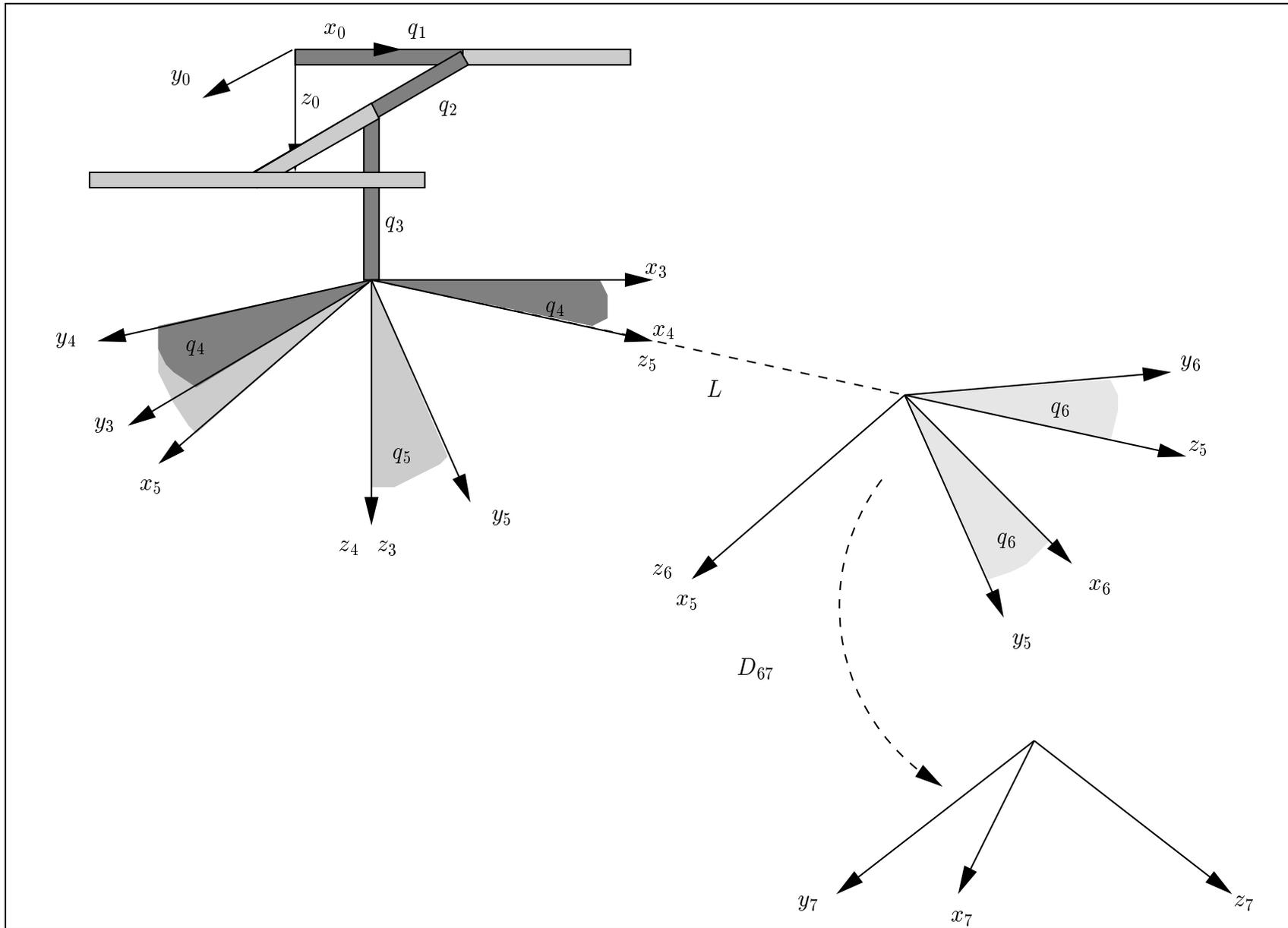


FIG. B.1 – Différents repères du robot cartésien 6 axes. Le robot est constitué d'une succession de 3 articulations de translation (q_1, q_2 et q_3) et de 3 articulations de rotation (q_4, q_5 et q_6) faisant passer des repères \mathcal{R}_3 à \mathcal{R}_4 , \mathcal{R}_4 à \mathcal{R}_5 , et \mathcal{R}_5 à \mathcal{R}_6 . Les repères $\mathcal{R}_3, \mathcal{R}_4$ et \mathcal{R}_5 ont la même origine alors que \mathcal{R}_6 est une composition de rotation et translation (L) de \mathcal{R}_5 . Le repère \mathcal{R}_0 est le repère fixe de référence. Le repère \mathcal{R}_7 est le repère de la caméra.

Bibliographie

- [Abe 00] T. Abe, Y. Matsuzawa. – A region extraction method using multiple active contour models. – *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR-00)*, pp. 64–69, Los Alamitos, juin 2000.
- [Agin 76] G. J. Agin, T. O. Binford. – Computer description of curved objects. *IEEE Transactions on Computing*, 1976, C-25(4):439–449, 1976.
- [Aloimonos 88] J. Aloimonos, I. Weiss, A. Bandyopadhyay. – Active vision. *International Journal of Computer Vision*, pp. 333–356, 1988.
- [Arbel 99] T. Arbel, F. P. Ferrie. – Viewpoint selection by navigation through entropy maps. – *Proceedings of the 7th IEEE Int. Conf. on Computer Vision*, vol. I, pp. 248–254, Los Alamitos, septembre 1999.
- [Avriel 76] M. Avriel. – *Nonlinear Programming: Analysis and Methods*. – Prentice-Hall, 1976.
- [Ayache 91] N. Ayache. – *Artificial Vision for Mobile Robots*. – The MIT Press, 342p., Cambridge, MA, 1991.
- [Ayer 96] S. Ayer, H.S. Sawhney. – Compact representations of videos through dominant and multiple motion estimation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(8):514–830, 1996.
- [BaezaYates 95] R. Baeza-Yates, R. Schott. – Parallel searching in the plane. *CGTA: Computational Geometry: Theory and Applications*, 5, 1995.
- [Bajcsy 88] R. Bajcsy. – Active perception. – *Proceedings of the IEEE*, vol. 76, p. 8, août 1988.
- [Ballard 91] D. H. Ballard. – Animate vision. *Artificial Intelligence*, 48(1):57–86, 1991.
- [Basar 89] T. Basar, P. Bernhard. – Differential games and applications. *Lecture Notes in Control and Information Sciences*, 119, 1989.
- [Baumgart 74] B. G. Baumgart. – *Geometric Modeling for Computer Vision*. – Palo Alto, CA, PhD. Thesis, Computer Science Department, Stanford University, 1974.
- [Boissonnat 84] J. D. Boissonnat. – Representing 2D and 3D shapes with the delaunay triangulation. – *Seventh International Conference on Pattern Recognition*, pp. 745–748, Montreal, Canada, 1984.
- [Boyer 97] E. Boyer, M. O. Berger. – 3D surface reconstruction using occluding contours. *International Journal of Computer Vision*, pp. 219–233, 1997.

- [Broman 86] V. Broman, M. J. Shensa. – Polytopes, a novel approach to tracking. – *Proceedings of 25th Conference on Decision and Control*, Athens, Greece, décembre 1986.
- [Brown 80] S. S. Brown. – Optimal search for a moving target in discrete time and space. *Operations Research*, 1980.
- [Caglioti 01a] V. Caglioti. – An entropic criterion for minimum uncertainty sensing in recognition and localization-part i: Theoretical and conceptual aspects. *IEEE Transactions on Systems, Man and Cybernetics*, 31(2):187–196, avril 2001.
- [Caglioti 01b] V. Caglioti. – An entropic criterion for minimum uncertainty sensing in recognition and localization-part ii: A case study on directional distance measurements. *IEEE Transactions on Systems, Man and Cybernetics*, 31(2):197–214, avril 2001.
- [Chatila 93] R. Chatila, M. Devy, M. Herb, S. Lacroix. – *Perception pour la navigation autonome en environnement naturel*. – Rapport de Recherche n93332, LAAS-CNRS, 1993.
- [Chatila 95] R. Chatila, S. Lacroix, T. Simeon, M. Herb. – *Planetary Exploration by a Mobile Robot: Mission Teleprogramming and Autonomous Navigation*. – Rapport de Recherche n93332, LAAS-CNRS, 1995.
- [Chaumette 90] F. Chaumette. – La relation vision-commande: théorie et application à des tâches robotiques. *Thèse de doctorat, Université de Rennes 1, IFSIC*, 1990.
- [Chaumette 91] F. Chaumette, P. Rives, B. Espiau. – Positioning of a robot with respect to an object, tracking it and estimating its velocity by visual servoing. – *Proc. IEEE Int. Conf. on Robotics and Automation*, Sacramento, Californie, avril 1991.
- [Chaumette 93] F. Chaumette, A. Santos. – Tracking a moving object by visual servoing. – *12th IFAC World Congress*, vol. 3, pp. 643–648, Sidney, Australie, juillet 1993.
- [Chaumette 96] F. Chaumette, S. Boukirand P. Bouthemy, D. Juvin. – Structure from controlled motion. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(5):492–504, mai 1996.
- [Chaumette 98a] F. Chaumette. – De la perception à l'action: l'asservissement visuel; de l'action à la perception: la vision active. *Habilitation à diriger les recherches, Université de Rennes 1, IFSIC*, 1998.
- [Chaumette 98b] F. Chaumette. – Potential problems of stability and convergence in image-based and position-based visual servoing. *The Confluence of Vision and Control*, éd. par D. Kriegman, G. Hager, A.S. Morse, pp. 66–78. – LNCIS Series, No 237, Springer-Verlag, 1998.
- [Chernousko 82] F. L. Chernousko. – Ellipsoidal bounds for sets of attainability and uncertainty in control problems. *Optimal Control Applications and Methods*, 3:187–202, 1982.
- [Cipolla 92] R. Cipolla, A. Blake. – Surface shape from the deformation of apparent contours. *Int. Journal on Computer Vision*, 9(2):83–112, novembre 1992.

- [Collewet 98] C. Collewet, F. Chaumette, L. Wallian, P. Marchal. – *Positionnement par rapport à un objet plan de forme inconnue par asservissement visuel 2D*. – Rapport de Recherche nRR-3419, Inria, Institut National de Recherche en Informatique et en Automatique, 1998.
- [Collewet 99] C. Collewet. – Contribution à l'élargissement du champ applicatif des asservissements visuels 2d. *Thèse de doctorat, Université de Rennes 1*, 1999.
- [Collins 93] Robert T. Collins. – *Model Acquisition Using Stochastic Projective Geometry*. – PhD. Thesis, University of Massachusetts, 1993.
- [Connolly 85] C. Connolly. – The determination of next best views. – *Proc. IEEE Int. Conf. on Robotics and Automation*, vol. 2, pp. 432–435, St Louis, Missouri, mars 1985.
- [Connolly 89] C. I. Connolly, J. R. Stenstrom. – 3d scene reconstruction from multiple intensity images. – *Proceedings, Workshop on Interpretation of 3D Scenes (Austin, TX, November 27–29, 1989)*, pp. 124–130, Washington, DC., 1989. Computer Society Press, Computer Society Press.
- [Crétual 98] A. Crétual, F. Chaumette. – Image-based visual servoing by integration of dynamic measurement. – *Proc. IEEE Int. Conf. on Robotics and Automation*, Leuven, Belgique, mai 1998.
- [Dambreville 99] F. Dambreville, J-P. Le Cadre. – *Detection of a Markovian Target with Optimization of the Search Efforts under Generalized Linear Constraints*. – Rapport de recherche, INRIA, 1999.
- [Dornaika 92] F. Dornaika. – Contribution à l'intégration vision / robotique : calibrage, localisation et asservissement. *Thèse de doctorat, Institut National Polytechnique de Grenoble*, 1992.
- [Durieu 96] C. Durieu, B. T. Polyak, E. Walter. – Trace versus determinant in ellipsoidal outer-bounding with application to state estimation. *IFAC 13th Triennial World Congress*, 1996.
- [Durrant-Whyte 87] H. F. Durrant-Whyte. – *Integration, Coordination, and Control of Multi-Sensor Robot Systems*. – Kluwer Academic Publishers, Boston, 1987.
- [DurrantWhyte 87] H. F. Durrant-Whyte. – Consistent integration and propagation of disparate sensor observations. *International Journal of Robotics Research*, 6(3):3–24, 1987.
- [DurrantWhyte 88a] H. F. Durrant-Whyte. – Uncertain geometry. *IEEE Journal of Robotics and Computer Vision*, 4, 1988.
- [DurrantWhyte 88b] H. F. Durrant-Whyte. – Uncertain geometry in robotics. *IEEE Journal of Robotics and Automation*, 4(1):23–31, 1988.
- [Elfes 96] A. Elfes. – Robot navigation: Integrating perception, environmental constraints and task execution within a probabilistic framework. *Lecture Notes in Computer Science*, 1093:93–130, 1996.
- [Espiau 92] B. Espiau, F. Chaumette, P. Rives. – A new approach to visual servoing in robotics. *IEEE Trans. on Robotics and Automation*, 8(3):313–326, juin 1992.

- [Evans 94] R. J. Evans, C. Zhang, Y. C. Soh. – Bounded-error estimation using dead zone and bounding ellipsoid. *International Journal of Adaptive Control and Signal Processing*, 8:31–42, 1994.
- [Fablet 99] R. Fablet, P. Bouthemy, M. Gelgon. – Moving object detection in color image sequences using region-level graph labeling. – *Proc. of 6th IEEE Int. Conf. on Image Processing, ICIP'99*, pp. 939–943, Kobe, octobre 1999.
- [Faugeras 87] O. Faugeras, F. Lustman, G. Toscani. – Motion and structure from motion from point and line matches. – *First International Conference on Computer Vision*, pp. 25–34, Washington, DC., 1987.
- [Ferrie 88] F. P. Ferrie, M. D. Levine. – Deriving coarse 3D models of objects. – *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pp. 345–353, Washington, DC., juin 1988.
- [Ferrie 92] F. P. Ferrie, J. Lagarde, P. Whaite. – Recovery of volumetric object descriptions from laser rangefinder images. – *Second European Conference on Computer Vision*, avril 1992. – Genoa, Italie.
- [François 91] E. François, P. Bouthemy. – *Multiframe-based identification of mobile components of a scene with moving camera*. – Rapport de Recherche nRR-1368, Inria, Institut National de Recherche en Informatique et en Automatique, 1991.
- [Gal 80] S. Gal. – *Search Games*. – Academic Press, 1980.
- [Garcia 98] B. Garcia, P. Brunet. – 3d reconstruction with projective octrees and epipolar geometry. – *IEEE Int. Conf. on Computer Vision*, pp. 1067–1072, janvier 1998.
- [Garvey 76] T. D. Garvey. – *Perceptual Strategies for Purposive Vision*. – n AI Center Technical note 117, SRI International, septembre 1976.
- [Geman 84] S. Geman, D. Geman. – Stochastic relaxation, Gibbs distribution and the Bayesian restoration of images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.
- [Grandjean 91] P. Grandjean. – Perception multisensorielle et interprétation de scènes. *Thèse de doctorat, Université Paul Sabatier de Toulouse*, 1991.
- [Guenin 61] J. De Guenin. – Optimum distribution of effort: an extension of the koopman basic theory. *Operations research*, (28), 1961.
- [Gupta 93] A. Gupta, R. Bajcsy. – Volumetric segmentation of range images of 3D objects using superquadratic models. *Computer Vision, Graphics, and Image Processing. Image Understanding*, 58(3):302–326, novembre 1993.
- [Hager 97] G.D. Hager. – A modular system for robust positioning using feedback from stereo vision. *IEEE Trans. on Robotics and Automation*, 13(4):582–595, août 1997.
- [Hanebeck 96] U. D. Hanebeck, J. Horn, G. Schmidt. – On combining set theoretic and bayesian estimation. – *IEEE Int. Conf. on Robotics and Automation*, 1996.
- [Hashimoto 93] K. Hashimoto. – *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, Singapor, 1993.
- [Hashimoto 98] K. Hashimoto, T. Noritsugu. – Performance and sensitivity in visual servoing. – *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 2321–2326, mai 1998.
- [Hoffman 84] D. D. Hoffman, W. A. Richards. – Parts of recognition. *Cognition*, 18:65–96, 1984.
- [Horaud 98] R. Horaud, F. Dornaika, B. Espiau. – Visually guided object grasping. *IEEE Trans. on Robotics and Automation*, 14(4):525–532, août 1998.
- [Hosoda 94] K. Hosoda, M. Asada. – Versatile visual servoing without knowledge of true jacobian. – *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 186–193, 1994.
- [Hosoda 95] K. Hosoda, K. Sakamoto, M. Asada. – Trajectory generation for obstacle avoidance of uncalibrated stereo visual servoing without 3d reconstruction. – *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, pp. 29–34, Pittsburgh, USA, août 1995.
- [Hutchinson 96] S. Hutchinson, G.D. Hager, P.I. Corke. – A tutorial on visual servo control. *IEEE Trans. on Robotics and Automation*, 12(5):651–670, octobre 1996.
- [Irani 92] M. Irani, B. Rousso, S. Peleg. – Detecting and tracking multiple moving objects using temporal integration. – *Proc. of 2nd Eur. Conf. on Computer Vision, ECCV'92*, LNCS Vol 588, pp. 282–287, Santa Margherita, mai 1992. Springer.
- [Jägersand 94] M. Jägersand, R. Nelson. – *Adaptative differential visual feedback for uncalibrated hand-eye coordination and motor control*. – Rapport de Recherche n579, University of Rochester, 1994.
- [Jägersand 97] M. Jägersand, O. Fuentes, R. Nelson. – Experimental evaluation of uncalibrated visual servoing for precision manipulation. – *Proc. IEEE Int. Conf. on Robotics and Automation*, 1997.
- [Kervrann 95] C. Kervrann, F. Heitz. – A Markov random field model-based approach to unsupervised texture segmentation using local and global statistics. *IEEE Trans. on Image Processing*, 4(6):856–862, 1995.
- [Kieffer 00] M. Kieffer, L. Jaulin, E. Walter, D. Meizel. – Localisation et suivi robustes d'un robot mobile grâce à l'analyse par intervalles. *Traitement du signal*, 17(3):207–219, 2000.
- [Koopman 57] B. O. Koopman. – The theory of search: part i,ii and iii. *Operations Research*, 1956-1957.
- [Kutulakos 92a] K. N. Kutulakos, C. R. Dyer. – Recovering shape by purposive viewpoint adjustment. – *Proc. Computer Vision and Pattern Recognition*, pp. 16–22, 1992.
- [Kutulakos 92b] K. N. Kutulakos, V. J. Lumelsky. – *Object Exploration by Purposive, Dynamic Viewpoint Adjustment*. – Rapport de Recherche n 1121, Computer Sciences Department, University of Wisconsin Madison, novembre 1992.

- [Kutulakos 93] K. N. Kutulakos, C. R. Dyer. – *Global Surface Reconstruction by Purposive Control of Observer Motion*. – Rapport de Recherche n1141, Computer Sciences Department, University of Wisconsin Madison, avril 1993.
- [Kutulakos 94] K. N. Kutulakos, C. R. Dyer, V. J. Lumelsky. – Provable strategies for vision-guided exploration in three dimensions. – *IEEE Int. Conf. Robotics and Automation*, pp. 1365–1372, Los Alamitos, CA, 1994.
- [Lacroix 95] S. Lacroix. – Stratégies de perception et de déplacement pour la navigation d'un robot mobile autonome en environnement naturel. *Thèse de doctorat, Université Paul Sabatier de Toulouse*, 1995.
- [Lacroix 97] S. Lacroix, R. Chatila. – *Motion and perception strategies for outdoor mobile robot navigation in unknown environments*. – Springer-Verlag, New York, 1997.
- [Laurentini 94] A. Laurentini. – The Visual Hull Concept for Silhouette-Based Image Understanding. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(2):150–162, février 1994.
- [Leclerc 89] Y.G. Leclerc. – Constructing simple stable descriptions for image partitioning. *International Journal of Computer Vision*, 3(1):73–102, 1989.
- [Leonardis 94] A. Leonardis, F. Solina, A. Macerl. – A direct recovery of superquadric models in range images using recover-and-select paradigm. *Lecture Notes in Computer Science*, 800:309–318, 1994.
- [LópezOrtiz 96] A. López-Ortiz. – *On-line Target Searching in Bounded and Unbounded Domains*. – PhD. Thesis, University of Waterloo, 1996.
- [LópezOrtiz 98] A. López-Ortiz, S. Schuierer. – The ultimate strategy to search on m rays? *Lecture Notes in Computer Science*, 1449, 1998.
- [Lütkepohl 96] H. Lütkepohl. – *Handbook of Matrices*. – John Wiley & Sons, Ltd., 1996.
- [Maksarov 96] D. G. Maksarov, J. P. Norton. – State bounding with ellipsoidal set description of the uncertainty. *Int. Journal on Control*, 65(5):847–866, 1996.
- [Malis 98a] E. Malis. – Contributions à la modélisation et à la commande en asservissement visuel. *Thèse de doctorat, Université de Rennes 1, IFSIC*, 1998.
- [Malis 98b] E. Malis, F. Chaumette, S. Boudet. – 2d 1/2 visual servoing stability analysis with respect to camera calibration errors. – *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, vol. 2, pp. 691–697, 1998.
- [Malis 98c] E. Malis, F. Chaumette, S. Boudet. – Positioning a coarse-calibrated camera with respect to an unknown object by 2d 1/2 visual servoing. – *IEEE Int. Conf. on Robotics and Automation*, pp. 1352–1359, Leuven, Belgique, mai 1998.
- [Malis 99] E. Malis, F. Chaumette, S. Boudet. – 2 1/2 d visual servoing. *IEEE Transactions on Robotics and Automation*, 15(2):238–250, avril 1999.
- [Marchand 96] E. Marchand. – Stratégies de perception par vision active pour la

- reconstruction et l'exploration de scènes statiques. *Thèse de doctorat, Université de Rennes 1, IFSIC, 1996.*
- [Marchand 98] E. Marchand, G. Hager. – Dynamic sensor planning in visual servoing. – *IEEE Int. Conf. on Robotics and Automation*, vol. 3, pp. 1988–1993, Leuven, Belgique, mai 1998.
- [Marchand 99] E. Marchand, F. Chaumette. – Active vision for complete scene reconstruction and exploration. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(1):65–72, janvier 1999.
- [Marr 77] D. Marr, K. Nishihara. – Representation and recognition of the spatial organization of three-dimensional shapes. *Proc. Royal Soc. London Bulletin*, pp. 269–294, 1977.
- [Marr 82] D. Marr. – *Vision*. – W. H. Freeman and Company, San Francisco, 1982, 1982.
- [Maver 93] J. Maver, R. Bajcsy. – Occlusions as a guide for planning the next view. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15(5):417–433, mai 1993.
- [Meng 00] L. Meng, T. Nguyen, D. Castañón. – An image-based bayesian framework for face detection. – *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR-00)*, pp. 302–307, Los Alamitos, juin 2000. IEEE.
- [Messaoud 94] H. Messaoud, G. Favier. – Recursive determination of parameter uncertainty intervals for linear models with unknown but bounded errors. – *Proceedings of SYSID 94. 10th IFAC Symposium on System Identification*, vol. 3, Copenhagen, Danemark, juillet 1994.
- [Mezouar 00] Y. Mezouar, F. Chaumette. – Path planning in image space for robust visual servoing. – *IEEE Int. Conf. on Robotics and Automation*, San Francisco, CA, avril 2000.
- [Mezouar 01a] Y. Mezouar, F. Chaumette. – Design and tracking of desirable trajectories in the image space by integrating mechanical and visibility constraints. – *IEEE Int. Conf. on Robotics and Automation*, Séoul, Corée du Sud, mai 2001.
- [Mezouar 01b] Y. Mezouar, F. Chaumette. – *Path Planning for Robust Image-based Visual Servoing*. – Rapport de Recherche n1377, IRISA, 2001.
- [Milanese 91a] M. Milanese, A. Vicino. – Estimation theory for nonlinear models and set membership uncertainty. *Automatica*, 27:403–408, 1991.
- [Milanese 91b] M. Milanese, A. Vicino. – Optimal estimation theory for dynamic systems with set membership uncertainty: An overview. *Automatica*, 27:997–1009, 1991.
- [Milanese 96] M. Milanese, J. Norton, H. Piet-Lahanier, E. Walter (Eds.). – *Bounding Approaches to System Identification*. – Plenum, New York, 1996.
- [Mitiche 96] A. Mitiche, P. Bouthemy. – Computation and analysis of image motion: A synopsis of current problems and methods. *International Journal of Computer Vision*, 19(1):29–55, 1996.
- [Montiel 98] M. E. Montiel, A. S. Aguado, E. Zaluska. – Surface subdivision for generating superquadrics. *The Visual Computer*, 14(1):1–17, 1998.

- [Moutarlier 91] P. Moutarlier. – Modélisation autonome de l’environnement par un robot mobile. *Thèse de doctorat, Université Paul Sabatier de Toulouse*, 1991.
- [Norton 87] J. P. Norton. – Identification and application of bounded-parameter models. *Automatica*, 23:497–507, 1987.
- [Odobez 94] J.-M. Odobez. – Estimation, détection et segmentation du mouvement: une approche robuste et markovienne. *Thèse, Université de Rennes 1*, 1994.
- [Odobez 95] J.M. Odobez, P. Bouthemy. – Robust multiresolution estimation of parametric motion models. *Journal of Visual Communication and Image Representation*, 6(4):348–365, 1995.
- [Papageorgiou 99] C. Papageorgiou, T. Poggio. – Trainable pedestrian detection. – *Proc. of 7th IEEE Int. Conf. on Computer Vision, ICCV’99*, pp. 1223–1228, Kerkyra, septembre 1999.
- [Papanikolopoulos 93] N. P. Papanikolopoulos, P. K. Khosla, T. Kanade. – Visual tracking of a moving target by a camera mounted on a robot: a combination of control and vision. *IEEE Transactions on Robotics and Automation*, 9(1):14–35, février 1993.
- [Paragios 00] N. Paragios, R. Deriche. – Geodesic active contours and level sets for the detection and tracking of moving objects. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(3):266–280, 2000.
- [Pateux 00] S. Pateux. – Spatial segmentation of color images according to the MDL formalism. – *Proc. of 7th IEEE Int. Conf. on Image Processing, ICIP’2000*, pp. 92–95, Vancouver, septembre 2000.
- [Pentland 89] A. Pentland. – Part segmentation for object recognition. *Neural Computation*, 1:82–91, 1989.
- [PissardGibollet 91] R. Pissard-Gibollet, P. Rives. – *Asservissement visuel appliqué a un robot mobile : état de l’art et modélisation cinématique*. – Rapport de Recherche n RR-1577, Inria, Institut National de Recherche en Informatique et en Automatique, 1991.
- [PissardGibollet 93] R. Pissard-Gibollet. – Conception et commande par asservissement visuel d’un robot mobile. *Thèse de doctorat, Ecole des Mines de Paris*, 1993.
- [Potmesil 87] M. 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, octobre 1987.
- [Pérez 98] P. Pérez. – Markov random fields and images. *CWI Quarterly*, 11(4):413–437, 1998.
- [Pronzato 94a] L. Pronzato, E. Walter. – Minimal volume ellipsoids. *International Journal of Adaptive Control and Signal Processing*, 8:15–30, 1994.
- [Pronzato 94b] L. Pronzato, E. Walter. – Minimum-volume ellipsoids containing compact sets: Application to parameter bounding. *Automatica*, 30(11):1731–1739, 1994.

- [Reyes 98] F. Reyes, R. Kelly. – Experimental evaluation of fixed-camera direct visual controllers on a direct-drive robot. – *IEEE Int. Conf. on Robotics and Automation*, Leuven, Belgique, mai 1998.
- [Rimey 92] R.D. Rimey, C.M. Brown. – Where to look next using a Bayes net: Incorporating geometric relations. – *Second European Conference on Computer Vision*, pp. 542–550, Santa Margherita Ligure, Italie, mai 1992.
- [Rimey 93] R. D. Rimey. – *Control of Selective Perception Using Bayes Nets and Decision Theory*. – PhD. Thesis, University of Rochester, Computer Science Department, 1993.
- [Rimey 94] R.D. Rimey, C.M. Brown. – Control of selective perception using Bayes nets and decision theory. *International Journal of Computer Vision*, pp. 173–207, 1994.
- [Ruf 00] A. Ruf. – Vision-based guidance and control of robots in projective space. – *Proc. of the 6th European Conference on Computer Vision (ECCV'00)*, vol. II, pp. 50–66, Dublin, Irlande, juin 2000. Springer.
- [Saleem 96] S. K. Saleem, R. J. Evans, E. Skafidas. – Minimum volume overbounding ellipsoids for set-based estimation in target tracking applications. – *Proceedings of the 35th IEEE Conference on Decision and Control*, 1996.
- [Samson 91] C. Samson, M. Le Borgne, B. Espiau. – *Robot Control: the Task Function Approach*. – Clarendon Press, Oxford, United Kingdom, 1991.
- [Schweppe 68] F. C. Schweppe. – Recursive state estimation: unknown but bounded errors and system inputs. *IEEE Trans. on Automatic Control*, AC-13:22–28, 1968.
- [Shani 84] U. Shani, D. H. Ballard. – Splines as embeddings for generalized cylinders. *Computer Vision, Graphics and Image Processing*, 27:129–156, 1984.
- [Slabaugh 01] G. Slabaugh, B. Culbertson, T. Malzbender, R. Schafer. – *A survey of methods for volumetric scene reconstruction from photographs*. – Rapport de recherche, Center for Signal and Image Processing, Georgia Institute of Technology, 2001.
- [Solina 90] F. Solina, R. Bajcsy. – Recovery of Parametric Models from Range Images: The Case for Superquadrics with Global Deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(2):131–147, février 1990.
- [Srivastava 90] S. K. Srivastava, N. Ahuja. – Octree generation from object silhouettes in perspective views. *Computer Vision, Graphics, and Image Processing*, 49(1):68–84, janvier 1990.
- [Stiller 99] C. Stiller, J. Konrad. – Estimating motion in image sequences. *IEEE Signal Processing Magazine*, 16(4):70–91, 1999.
- [Sullivan 98] S. Sullivan, J. Ponce. – Automatic model construction, pose estimation, and object recognition from photographs using triangular

- splines. – *Proceedings of the International Conference on Computer Vision*, pp. 510–516, 1998.
- [Szeliski 93] R. Szeliski, P. Golland. – Rapid octree construction from image sequences. *Computer Vision, Graphics and Image Processing: Image Understanding*, 58(1):23–32, juillet 1993.
- [Tarabanis 95] K. A. Tarabanis, P. K. Allen, R. Y. Tsai. – A survey of sensor planning in computer vision. *IEEE Trans. on Robotics and Automation*, 11(1):86–104, février 1995.
- [Thompson 93] W.B. Thompson, P. Lechleider, E.R. Stuck. – Detecting moving objects using rigidity constraint. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(2):162–165, 1993.
- [Torr 93] P. H. S. Torr, D. W. Murray. – Statistical detection of independent movement from a moving camera. *Image and Vision Computing*, 11(4):180–187, 1993.
- [Tsotsos 90] J. K. Tsotsos. – Analysing vision at the complexity level. *Behavioral and Brain Sciences*, 13(3):423–496, 1990.
- [Tsotsos 92] J. K. Tsotsos. – On the relative complexity of active vs passive visual search. *International Journal of Computer Vision*, 7(2):127–141, 1992.
- [Vaillant 92] R. Vaillant, O. Faugeras. – Using extremal boundaries for 3-D object modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):157–173, février 1992.
- [Weiss 84] L.E. Weiss. – *Dynamic Visual Servo Control of Robots. An Adaptive Image based Approach*. – Rapport de Recherche nCMU-RI-TR-84-16, Carnegie-Mellon University, avril 1984.
- [Welzl 91] E. Welzl. – Smallest enclosing disks (balls and ellipsoids). *Lecture Notes in Computer Science*, 555:359–370, 1991.
- [Whaite 91] P. Whaite, F. P. Ferrie. – From uncertainty to visual exploration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(10):1038–1049, 1991.
- [Whaite 92] P. Whaite, F. P. Ferrie. – Uncertain views. – *Proc of IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pp. 3–9, Champaign, USA, juin 1992.
- [Whaite 94] P. Whaite, F. P. Ferrie. – Autonomous exploration: Driven by uncertainty. – *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 339–346, Los Alamitos, CA, juin 1994. IEEE Computer Society Press.
- [Wijesoma 93] S.W. Wijesoma, D.F.H. Wolfe, R.J. Richards. – Eye-to-hand coordination for vision-guided robot control applications. *Int. Journal of Robotics Research*, 12(1):65–78, février 1993.
- [Witsenhausen 68] H. S. Witsenhausen. – Sets of possible states of linear systems given perturbed observations. *IEEE Transactions on Automatic Control*, AC-13:556–558, 1968.
- [Wixson 94a] L. Wixson. – Viewpoint selection for visual search. – *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pp. 800–805, Los Alamitos, CA, juin 1994. IEEE Computer Society Press.

- [Wixson 94b] L.E. Wixson, D.H. Ballard. – Using intermediate objects to improve the efficiency of visual search. *International Journal of Computer Vision*, pp. 209–230, 1994.
- [Xiong 94] W. Xiong, C. Graffigne. – A hierarchical method for detection of moving objects. – *Proc. of 1st IEEE Int. Conf. on Image Processing, ICIP'94*, pp. 795–799, Austin, novembre 1994.
- [Ye 95] Y. Ye, J. K. Tsotsos. – Where to look next in 3d object search. – *Proceedings of 1995 IEEE International Symposium for Computer Vision*, Floride, U.S.A., novembre 1995.
- [Ye 96a] Y. Ye, J. K. Tsotsos. – On the collaborative object search team: a formulation. – *ICMAS-96 workshop on Learning, Interactions and Organizations in Multiagent Environment*, Kyoto, Japon, 1996.
- [Ye 96b] Y. Ye, J. K. Tsotsos. – Sensor planning in 3d object search. – *Int. Symposium on Intelligent Robotic Systems*, Lisbonne, Portugal, juillet 1996.
- [Ye 96c] Y. Ye, J. K. Tsotsos. – Sensor planning in 3d object search: its formulation and complexity. – *The 4th International Symposium on Artificial Intelligence and Mathematics*, Floride, U.S.A., janvier 1996.
- [Ye 97] Y. Ye. – *Sensor Planning for Object Search*. – PhD. Thesis, University of Toronto, Department of Computer Science, 1997.
- [Yoshimi 94] B. H. Yoshimi, P. K. Allen. – Active, uncalibrated visual servoing. – *Proc. IEEE Int. Conf. on Robotics and Automation*, vol. 1, pp. 156–161, mai 1994.
- [Zell 94] J.-F. Zell, G. Garcia. – Octree models of 3D worlds using a camera mounted on a robot wrist. – *Proc. of Third Int. Conf. on Automation, Robotics and Computer Vision*, vol. 3, pp. 1678–1682, Singapour, novembre 1994.
- [Zell 95] J.-F. Zell. – Utilisation de silhouettes pour la perception et la modélisation de l'environnement en robotique. *Thèse de doctorat, Ecole Centrale de Nantes*, 1995.
- [Zucker 88] S. Zucker, C. David, L. Iverson. – The organization of curve detection: Coarse tangent fields and fine spline coverings. – *2ND International Conference on Computer Vision*, Tampa, Floride, décembre 1988.

Publications

Congrès internationaux

G. Flandin, F. Chaumette et É. Marchand. – Eye-in-hand / Eye-to-hand Cooperation for Visual Servoing. – *IEEE Int. Conf. on Robotics and Automation*, San Francisco, avril 2000.

G. Flandin et F. Chaumette. – Autonomous Visual Exploration of Complex Objects. – *IEEE Int. Conf. on Intelligent Robots and Systems*, Hawaii, octobre 2001.

G. Flandin et F. Chaumette. – Vision-based Control Using Probabilistic Geometry for Objects Reconstruction. – *IEEE Int. Conf. on Decision and Control*, Orlando, décembre 2001.

Rapport de recherche

G. Flandin et F. Chaumette. – Visual Data Fusion: Application to Objects Localisation and Exploration. – *Rapport de recherche*, IRISA-INRIA, février 2001.

Congrès nationaux

G. Flandin, F. Chaumette et É. Marchand. – Coopération caméra embarquée/caméra déportée pour l'asservissement visuel. – *12ème Journées des Jeunes Chercheurs en Robotique*, pages 58-63, Bourges, février 2000.

G. Flandin et F. Chaumette. – Fusion d'informations visuelles pour la localisation d'objets complexes. – *ORASIS*, Cahors, juin 2001.

Soumissions

G. Flandin et F. Chaumette. – Fusion d'informations visuelles pour la localisation d'objets complexes. – *soumis à Traitement du Signal*.

G. Flandin et F. Chaumette. – Visual Data Fusion for Objects Localisation. – *soumis à IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Résumé

Cette étude concerne l'exploration visuelle autonome pour la reconstruction de scènes inconnues. Il s'agit d'interpréter des informations visuelles et de générer les points de vue qui permettent de construire progressivement une carte de l'environnement. Le problème est décomposé, de façon hiérarchique, en trois fonctionnalités. Dans un premier temps, le système détermine la suite des actions aboutissant à un inventaire de tous les objets de la scène. Cette problématique s'inscrit dans le contexte très général de la recherche d'objets. L'approche que nous présentons est basée sur une description probabiliste de l'occupation de la scène par des objets. La recherche consiste alors à générer une suite d'observations aboutissant à des probabilités proches de 1 aux endroits où se trouve un objet et proches de 0 ailleurs. Nous développons plusieurs stratégies allant dans ce sens. Dans un second temps, l'exploration est focalisée sur chaque objet afin d'en améliorer la description. Nous présentons un modèle d'objet basé sur un mélange de modèles stochastique et à erreur bornée permettant de représenter l'enveloppe approchée de l'objet tout en tenant compte des incertitudes de localisation. Nous développons un algorithme d'estimation en ligne de ce modèle et élaborons un processus d'exploration optimale en temps réel basé sur la minimisation de l'incertitude de localisation de l'objet observé. Enfin, la dernière fonctionnalité concerne le suivi des consignes permettant de déplacer la caméra tout en suivant l'objet d'intérêt. Ce problème est résolu par asservissement visuel dont nous étudions les potentialités du point de vue de la coopération caméra globale/caméra locale.

Mots clefs : Exploration, vision active, recherche d'objets, modélisation, asservissement visuel.

Abstract

This thesis is concerned with autonomous visual exploration for unknown scene reconstruction. Based on visual informations, the system must select motions and view points in order to build a map of its own environment. An hierarchical decomposition of the problem is proposed. The first step is dedicated to the object search in order to inventory all the objects of the scene. Our approach is based on a probabilistic description of the scene occupancy. Our search strategy consists in generating a sequence of observations such that all the probabilities will reach 1 everywhere an object is present and 0 elsewhere. The next step deals with the exploration of each particular object in order to improve its description. We present an object modeling as a mixture of stochastic and set membership models allowing to coarsely approximate the objects envelope while taking localization uncertainties into account. For this particular model, we develop an estimating algorithm and elaborate an optimal exploration process based on the localization uncertainty minimization. At last, we focus on servoing aspects that make the system able to track an object while moving around it. This problem is solved thanks to visual servoing technics whose performances are studied from the eye-in-hand/eye-to-hand cooperation point of view.

Keywords : Exploration, active vision, object search, modeling, visual servoing.