



**Titre:** Représentation étendue de scènes combinant des primitives  
Title: intrinsèques géométriques et photométriques

**Auteur:** Alexandre Fortin  
Author:

**Date:** 2006

**Type:** Mémoire ou thèse / Dissertation or Thesis

**Référence:** Fortin, A. (2006). Représentation étendue de scènes combinant des primitives  
Citation: intrinsèques géométriques et photométriques [Master's thesis, École  
Polytechnique de Montréal]. PolyPublie. <https://publications.polymtl.ca/8099/>

 **Document en libre accès dans PolyPublie**  
Open Access document in PolyPublie

**URL de PolyPublie:** <https://publications.polymtl.ca/8099/>  
PolyPublie URL:

**Directeurs de  
recherche:** Paul Cohen  
Advisors:

**Programme:** Unspecified  
Program:

UNIVERSITÉ DE MONTRÉAL

REPRÉSENTATION ÉTENDUE DE SCÈNES COMBINANT DES  
PRIMITIVES INTRINSÈQUES GÉOMÉTRIQUES ET PHOTOMÉTRIQUES

ALEXANDRE FORTIN  
DÉPARTEMENT DE GÉNIE ÉLECTRIQUE  
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION  
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES  
(GÉNIE ÉLECTRIQUE)  
DÉCEMBRE 2006



Library and  
Archives Canada

Bibliothèque et  
Archives Canada

Published Heritage  
Branch

Direction du  
Patrimoine de l'édition

395 Wellington Street  
Ottawa ON K1A 0N4  
Canada

395, rue Wellington  
Ottawa ON K1A 0N4  
Canada

*Your file* *Votre référence*  
*ISBN: 978-0-494-25542-1*  
*Our file* *Notre référence*  
*ISBN: 978-0-494-25542-1*

**NOTICE:**

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

**AVIS:**

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

---

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

  
**Canada**

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

REPRÉSENTATION ÉTENDUE DE SCÈNES COMBINANT DES  
PRIMITIVES INTRINSÈQUES GÉOMÉTRIQUES ET PHOTOMÉTRIQUES

présenté par: FORTIN Alexandre

en vue de l'obtention du diplôme de: Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de:

M. LESAGE Frédéric, Ph.D., président

M. COHEN Paul, Ph.D., membre et directeur de recherche

M. BRAULT Jean-Jules, Ph.D., membre

## REMERCIEMENTS

Je voudrais en premier lieu remercier M. Paul Cohen, mon directeur de maîtrise et responsable du Groupe de Recherche en Perception et Robotique (GRPR), sans qui ce projet n'aurait pas été possible. Je remercie également tous les membres du GRPR pour leur aide et leur support tout au long de mon séjour au GRPR. Un grand merci spécialement à mes collègues MM. Vincent Zalzal et Raphaël Gava qui ont participé à d'innombrables discussions enrichissantes.

Je remercie M. François Trottier pour ses judicieux conseils en géométrie et en mathématiques et pour le grand intérêt qu'il a porté envers mon projet.

Un grand merci à ma soeur Anne-Marie Fortin et à mes parents Louise Hamel et Pierre Fortin ainsi qu'à leurs conjoints respectifs Richard Masson et Catherine Beaudin qui m'ont encouragé et supporté durant ma maîtrise. Je remercie aussi Louise et Richard pour la relecture de ce mémoire.

Finalement, je tiens à remercier les organismes suivants pour les bourses qu'ils m'ont octroyées pour la réalisation de ce projet : le Conseil de recherche en sciences naturelles et en génie du Canada (CRSNG), le Fonds québécois de la recherche sur la nature et les technologies (FQRNT), l'Agence spatiale canadienne (ASC) et l'École Polytechnique de Montréal.

## RÉSUMÉ

L'objectif de ce travail est de détecter les surfaces des objets présents dans un environnement structuré correspondant par exemple à l'intérieur d'un bâtiment et de reconstruire virtuellement cet environnement en respectant les positions, orientations et couleurs des surfaces observées.

Nous utilisons les hypothèses limitatives qui suivent. L'environnement observé est statique et la majorité des surfaces observées sont planes. L'éclairage de l'environnement est uniforme et constant pendant la durée des observations. De plus, nous supposons que le capteur est installé sur une plate-forme se déplaçant sur une surface plane.

Nous avons conçu un nouveau capteur 3D qui utilise un télémètre laser, une caméra et une plate-forme rotative afin de réaliser l'acquisition de données télémétriques et photométriques enregistrées. À chaque numérisation, ce capteur fournit une grille de 61 009 données télémétriques et photométriques superposées correspondant à des points présents dans l'environnement situé devant la plate-forme. Les résultats obtenus montrent que l'erreur télémétrique moyenne de notre capteur est de 5,9 mm et que le décalage maximal observé entre les données télémétriques et photométriques est de 1 pixel.

Nous représentons l'environnement observé par un ensemble de primitives géométriques correspondant à des portions de surfaces planes, cylindriques et sphériques. Notre technique pourrait être généralisée à l'usage d'autres types de primitives. L'extraction de ces primitives provient d'une double segmentation de la scène, l'une basée sur la télémétrie et la seconde basée sur la photométrie. Les deux segmentations sont ensuite fusionnées pour obtenir une segmentation qui tient compte des deux modalités sensorielles. Nous montrons également que la photométrie peut être utilisée dans certaines situations pour raffiner les contours de la segmentation télémétrique. Les résultats obtenus démontrent que notre technique de segmentation

détecte adéquatement plus de 90 % des surfaces présentes dans l'environnement. Les surfaces obtenues à la suite de la segmentation sont intégrées à une carte en construction. Pour y arriver, nous avons conçu un nouvel algorithme de localisation du capteur dans son environnement. Cet algorithme est une modification de l'algorithme ICP (Iterative Closest Points) qui est très utilisé en robotique mobile. Les modifications apportées permettent d'utiliser les surfaces détectées plutôt que d'utiliser les données brutes, ce qui offre un gain de performance considérable par rapport à l'algorithme ICP classique. L'intégration des surfaces à la carte en construction permet d'obtenir une représentation étendue de l'environnement dans laquelle l'erreur de dimension des surfaces est d'environ 5 cm et l'erreur d'orientation d'environ  $0,25^\circ$ .

Le choix des points de vue et le déplacement de la plate-forme entre ces points de vue sont réalisés par un opérateur. À la demande de l'opérateur, la plate-forme mobile sur laquelle est situé le capteur exécute de manière autonome les opérations suivantes : acquisition des données, segmentation des données, correction de la localisation du capteur en fonction des observations effectuées et intégration des nouvelles surfaces à la carte en construction. Le système que nous avons conçu pourrait servir de base à des développements futurs, dont le choix optimisé des points de vue et le déplacement autonome, avec évitements d'obstacles, entre ces points de vue.

## ABSTRACT

The aim of this work is to detect objects' surfaces in an structured environment, such as the interior of a building, and to create a virtual reconstruction of this environment which respects the positions, orientations and colors of the observed surfaces.

We use the following restrictive assumptions. The observed environment is static and most of the observed surfaces are planar. The environment's light is uniform and constant throughout the observations. Moreover, we suppose that the sensor is mounted on a platform moving on a planar surface.

We designed a new 3D sensor based on a laser rangefinder, a camera and a tilt platform in order to carry out registered telemetric and photometric data acquisition. For each acquisition, this sensor provides a grid of 61,009 positions and colors corresponding to points observed in the environment facing the platform. The obtained results show that the average telemetric error of our sensor is 5.9 mm and that the maximum shift observed between telemetric and photometric data is 1 pixel.

The reconstructed environment is made of many kind of geometric primitives which are portions of planes, spheres and cylinders. Our representation can be expanded to add other types of primitives as well. To extract these primitives, two segmentations are performed, one based on telemetry and the other based on photometry. These two segmentations are then merged to obtain a new segmentation which takes into account both sensory informations. We also show that photometry can be used in certain situations to refine the edges of telemetric segmentation. The obtained results show that our segmentation method detects correctly more than 90% of surfaces found in the environment.

Surfaces obtained after the segmentation process are integrated into a map. In order to do so, we designed a new localization algorithm to locate the sensor in its

environment. This algorithm is a modified version of the ICP algorithm (Iterative Closest Points) which is often used in mobile robotics. Our modifications make it possible to use the detected surfaces to carry out the localization rather than use the raw data directly. Hence we get a considerable gain of performance compared to traditional ICP. With the integration of surfaces in a map we obtain an extended representation of the environment in which the error of the surfaces' dimension is approximately 5 cm and the error of orientation is around  $0.25^\circ$ .

The choice of the best viewpoints and the navigation from one viewpoint to another are carried out by an operator. Upon the operator's request, the mobile platform on whom is mounted the sensor carries out autonomously data acquisition, data segmentation, sensor localization based on the observations and integration of new surfaces to the map being built. The system we designed could be used as building block for future developments. These developments may consist of the optimized choice of viewpoints and the autonomous navigation, with obstacle avoidance, from one viewpoint to the other.

## TABLE DES MATIÈRES

REMERCIEMENTS . . . . .	iv
RÉSUMÉ . . . . .	v
ABSTRACT . . . . .	vii
TABLE DES MATIÈRES . . . . .	ix
LISTE DES TABLEAUX . . . . .	xiv
LISTE DES FIGURES . . . . .	xv
LISTE DES NOTATIONS ET DES SYMBOLES . . . . .	xix
LISTE DES ANNEXES . . . . .	xxii
INTRODUCTION . . . . .	1
CHAPITRE 1 REVUE DE LA LITTÉRATURE . . . . .	7
1.1 Acquisition des données . . . . .	8
1.1.1 Créer un capteur 3D à partir d'un capteur 2D . . . . .	8
1.1.1.1 Translation d'un télémètre laser . . . . .	8
1.1.1.2 Rotation d'un télémètre laser . . . . .	10
1.1.1.3 Combinaison télémétrie/photométrie . . . . .	10
1.1.2 Autres approches . . . . .	12
1.1.3 Capteur utilisé . . . . .	12
1.2 Différentes représentations de cartes 3D . . . . .	13
1.2.1 Maillage . . . . .	14
1.2.2 Octree . . . . .	15

1.2.3	Subdivision de la scène en objets . . . . .	16
1.2.4	Cartes hiérarchiques . . . . .	17
1.2.5	Représentation adoptée dans ce mémoire . . . . .	19
1.3	Traitement des données . . . . .	19
1.3.1	Segmentation . . . . .	19
1.3.1.1	Segmentation basée sur la télémétrie . . . . .	20
1.3.1.2	Segmentation basée sur la photométrie . . . . .	22
1.3.1.3	Segmentation combinant télémétrie et photométrie . . . . .	25
1.3.1.4	Méthode de segmentation utilisée dans ce mémoire . . . . .	26
1.3.2	Intégration de vues multiples . . . . .	27
1.3.2.1	Localisation . . . . .	28
1.3.2.2	Cartes topologiques . . . . .	29
1.3.2.3	Méthode de localisation utilisée dans ce mémoire . . . . .	30
1.4	Conclusion de la revue de la littérature . . . . .	30
CHAPITRE 2 CONCEPTION D'UN CAPTEUR . . . . .		33
2.1	Revue des capteurs commerciaux . . . . .	34
2.1.1	Technologie temps de vol par mesure directe . . . . .	35
2.1.2	Technologie temps de vol par mesure de phase . . . . .	36
2.1.3	Technologie par triangulation avec motif lumineux . . . . .	37
2.1.4	Technologie stéréoscopie . . . . .	39
2.1.5	Conclusion de la revue des capteurs commerciaux . . . . .	40
2.2	Fonctionnement du capteur proposé . . . . .	41
2.2.1	Télémètre laser . . . . .	42
2.2.1.1	Principe de fonctionnement général . . . . .	42
2.2.1.2	Repère lié au télémètre laser . . . . .	43
2.2.1.3	Fonctionnement détaillé . . . . .	45
2.2.2	Caméra . . . . .	46

2.2.2.1	Repère lié à la caméra . . . . .	47
2.2.2.2	Changement du repère lié au télémètre vers le re- père lié à la caméra . . . . .	48
2.2.2.3	Projection dans le plan image . . . . .	48
2.2.3	Plate-forme rotative . . . . .	50
2.2.3.1	Principe de fonctionnement . . . . .	51
2.2.3.2	Repère lié au capteur 3D . . . . .	52
2.2.3.3	Mode d'acquisition ligne par ligne . . . . .	52
2.2.3.4	Mode d'acquisition continu . . . . .	53
2.2.4	Procédure d'acquisition complète . . . . .	55
2.3	Exemple de résultat . . . . .	57
2.4	Intégration du capteur à la plate-forme mobile . . . . .	57
2.5	Problèmes rencontrés . . . . .	60
2.5.1	Occlusions . . . . .	60
2.5.2	Points fantômes . . . . .	62
2.6	Simulateur . . . . .	63
2.7	Conclusion de la conception du capteur . . . . .	66
CHAPITRE 3 SEGMENTATION . . . . .		67
3.1	Segmentation par la télémétrie . . . . .	70
3.1.1	Extraction de surfaces planes . . . . .	71
3.1.1.1	Description de la technique retenue . . . . .	71
3.1.1.2	Détermination des paramètres . . . . .	74
3.1.1.3	Modifications apportées à l'algorithme de base . . . . .	75
3.1.1.4	Plans aux moindres carrés . . . . .	78
3.1.1.5	Exemple de résultat . . . . .	79
3.1.1.6	Résultat sous forme de graphe . . . . .	80
3.1.2	Extraction de surfaces courbes . . . . .	81

3.1.2.1	Algorithme de Levenberg-Marquardt . . . . .	84
3.1.2.2	Modèles utilisés . . . . .	86
3.1.2.3	Processus de fusion des régions . . . . .	86
3.1.2.4	Exemple de résultat . . . . .	87
3.2	Segmentation par la photométrie . . . . .	87
3.2.1	Détails d'implantation . . . . .	90
3.2.2	Exemple de résultat . . . . .	92
3.3	Fusion des segmentations télémétriques et photométriques . . . . .	92
3.3.1	Utilisation de la photométrie pour désambiguïser la télémétrie	95
3.3.2	Schéma de segmentation détaillé . . . . .	98
3.3.3	Exemple de résultat . . . . .	100
3.3.4	Cas des données non enregistrées . . . . .	100
3.3.4.1	Autocalibration du capteur . . . . .	103
3.4	Régions finales . . . . .	103
3.5	Conclusion du chapitre sur la segmentation . . . . .	105
CHAPITRE 4 INTÉGRATION DE VUES MULTIPLES . . . . .		107
4.1	Localisation . . . . .	108
4.1.1	Description de l'algorithme utilisé . . . . .	109
4.1.1.1	Appariement des surfaces . . . . .	110
4.1.1.2	Calcul de la rotation . . . . .	112
4.1.1.3	Calcul de la translation . . . . .	113
4.2	Fusion des surfaces . . . . .	115
4.2.1	Fusion des surfaces planes . . . . .	117
4.2.2	Fusion des surfaces cylindriques . . . . .	117
4.2.3	Fusion des surfaces sphériques . . . . .	118
4.2.4	Fusion d'une surface courbe et d'une surface plane . . . . .	120
4.2.5	Fusion de deux éléments surfaciques courbes mal détectés . . . . .	120

4.2.6	Cas général . . . . .	121
4.3	Exemple de résultat . . . . .	121
4.4	Conclusion du chapitre de l'intégration de vues multiples . . . . .	123
CHAPITRE 5 RÉSULTATS ET ANALYSE . . . . .		126
5.1	Caractéristiques du capteur . . . . .	126
5.1.1	Précision télémétrique . . . . .	127
5.1.2	Précision de la registration . . . . .	128
5.2	Résultats de segmentations . . . . .	131
5.2.1	Segmentation d'images provenant d'une scène simulée . . . . .	133
5.2.1.1	Métriques utilisées pour l'analyse . . . . .	133
5.2.1.2	Caractéristiques des simulations effectuées . . . . .	135
5.2.1.3	Résumé des résultats obtenus . . . . .	136
5.2.1.4	Exemples visuels . . . . .	138
5.2.2	Simulations spécifiques . . . . .	142
5.2.3	Résultats provenant d'images réelles . . . . .	144
5.2.4	Conclusion des résultats de notre technique de segmentation . . . . .	151
5.3	Résultats d'intégrations de vues multiples . . . . .	153
5.3.1	Résultats de simulations . . . . .	154
5.3.2	Résultats provenant d'images réelles . . . . .	157
5.3.3	Conclusion des résultats de la fusion des vues multiples . . . . .	167
CONCLUSION . . . . .		169
RÉFÉRENCES . . . . .		173
ANNEXES . . . . .		182

## LISTE DES TABLEAUX

Tableau 2.1	Modes d'opération du télémètre laser . . . . .	45
Tableau 3.1	Taille des régions de l'exemple de la figure 3.11 . . . . .	96
Tableau 3.2	Choix des régions télémétriques et justifications accompa- gnant l'exemple de la figure 3.11 . . . . .	97
Tableau 3.3	Légende de la figure 3.13 . . . . .	99
Tableau 5.1	Taux de réussite pour les éléments surfaciques de plus de 100 points . . . . .	137
Tableau 5.2	Erreurs moyennes et écarts-types obtenus à la suite des seg- mentations . . . . .	137
Tableau 5.3	Valeurs des paramètres utilisés pour la localisation et la fu- sion des surfaces . . . . .	154
Tableau 5.4	Erreurs de positionnement avec et sans correction . . . . .	157
Tableau 5.5	Comparaison entre les surfaces attendues et obtenues . . . . .	158
Tableau IV.1	Caractéristiques des simulations effectuées . . . . .	203
Tableau IV.2	Répartition des éléments surfaciques mal segmentés et en trop en fonction de leur taille . . . . .	208
Tableau IV.3	Taux de réussite pour les éléments surfaciques de plus de 100 points pour $\tau=0,8$ . . . . .	209
Tableau IV.4	Erreurs moyennes et écarts-types . . . . .	210
Tableau IV.5	Comparaison des techniques de segmentation . . . . .	218
Tableau V.1	Seuils photométriques nécessaires pour discerner les éléments surfaciques . . . . .	226

## LISTE DES FIGURES

Figure 2.1	Triangulation avec un motif lumineux . . . . .	38
Figure 2.2	Stéréoscopie . . . . .	39
Figure 2.3	Vue d'ensemble du capteur . . . . .	42
Figure 2.4	Coupe 2D parallèle au plancher . . . . .	43
Figure 2.5	Repère lié au télémètre laser . . . . .	44
Figure 2.6	Ordre interne d'acquisition plane en mode $0,25^\circ$ . . . . .	46
Figure 2.7	Repères liés au télémètre laser et à la caméra . . . . .	47
Figure 2.8	Image prise par la caméra pendant l'acquisition du télémètre laser de la figure 2.4 . . . . .	50
Figure 2.9	Acquisition télémétrique et photométrique 2D . . . . .	51
Figure 2.10	Motif de points pour le mode ligne par ligne . . . . .	52
Figure 2.11	Motif de points pour le mode continu . . . . .	54
Figure 2.12	Exemple d'une numérisation 3D . . . . .	58
Figure 2.13	Dissociation des données télémétriques et photométriques de l'exemple de la figure 2.12 . . . . .	59
Figure 2.14	Exemple d'une situation d'occlusion . . . . .	61
Figure 2.15	Méthode de détection des occlusions . . . . .	62
Figure 2.16	Exemple réel de détection des occlusions . . . . .	63
Figure 2.17	Points fantômes . . . . .	64
Figure 2.18	Image obtenue au moyen du simulateur . . . . .	65
Figure 3.1	Vue d'ensemble du système de segmentation . . . . .	69
Figure 3.2	Nécessité de détecter les zones 4-connectées . . . . .	73
Figure 3.3	Exemple de l'utilité de vérifier les zones 4-connectées à chaque itération . . . . .	76
Figure 3.4	Répartition spatiale des vecteurs normaux obtenus à la suite de la numérisation d'un plan. . . . .	78

Figure 3.5	Résultat d'une extraction de surfaces planes . . . . .	80
Figure 3.6	Exemple d'une scène et de son graphe. . . . .	81
Figure 3.7	Procédé d'extraction des surfaces courbes . . . . .	83
Figure 3.8	Exemple d'extraction de surfaces courbes . . . . .	88
Figure 3.9	Résultat d'une segmentation photométrique . . . . .	93
Figure 3.10	Fusion des segmentations . . . . .	94
Figure 3.11	Exemple de désambiguïsation de la télémétrie par la photométrie . . . . .	96
Figure 3.12	Effet de la désambiguïsation sur la qualité de la fusion . . . . .	98
Figure 3.13	Schéma détaillé du processus complet de segmentation . . . . .	100
Figure 3.14	Exemple d'une fusion des segmentations télémétriques et photométriques . . . . .	101
Figure 3.15	Graphe correspondant au résultat de segmentation montré à la figure 3.14 . . . . .	104
Figure 4.1	Problèmes pouvant survenir lors de la correction de la position de la plate-forme . . . . .	114
Figure 4.2	Fusion de deux surfaces planes . . . . .	116
Figure 4.3	Exemple de surfaces courbes mal détectées . . . . .	121
Figure 4.4	Exemple d'intégration de vues multiples . . . . .	122
Figure 5.1	Distribution de l'erreur de distance . . . . .	127
Figure 5.2	Répartition des erreurs de registration . . . . .	129
Figure 5.3	Image complexe bien enregistrée . . . . .	130
Figure 5.4	Exemple d'image simulée bien segmentée . . . . .	139
Figure 5.5	Autre exemple d'image simulée bien segmentée . . . . .	140
Figure 5.6	Exemple d'image simulée mal segmentée . . . . .	141
Figure 5.7	Segmentation de l'image réelle « Grande salle sur deux étages »	145
Figure 5.8	Segmentation de l'image réelle « Colonne et salle de cours »	147
Figure 5.9	Segmentation de l'image réelle « Poste de travail 1 » . . . . .	148

Figure 5.10	Segmentation de l'image réelle « Poste de travail 2 » . . . . .	150
Figure 5.11	Intégration de vues multiples provenant d'une scène simulée	156
Figure 5.12	Intégration de vues multiples provenant d'une scène réelle .	159
Figure 5.13	Intégration de vues multiples d'un local de travail . . . . .	161
Figure 5.14	Vues multiples d'une portion d'un corridor, premier essai . .	163
Figure 5.15	Intégration des vues multiples d'une portion d'un corridor, premier essai . . . . .	164
Figure 5.16	Vues multiples d'une portion d'un corridor, deuxième essai .	165
Figure 5.17	Intégration des vues multiples d'une portion d'un corridor, deuxième essai . . . . .	166
Figure II.1	Fonctionnement de Player . . . . .	186
Figure II.2	Exemple d'un circuit Acropolis et de l'environnement Acro- builder . . . . .	187
Figure III.1	Exemple de convergence vers un minimum local . . . . .	193
Figure III.2	Paramétrisation d'un cylindre . . . . .	195
Figure IV.1	Scène simulée pour valider la méthode de segmentation . . .	204
Figure IV.2	Classification des éléments surfaciques en fonction de $\tau$ . . .	206
Figure IV.3	Classification des éléments surfaciques pour $\tau=0,8$ . . . . .	207
Figure IV.4	Distribution de l'erreur spatiale pour la série 2 avec $\tau=0,8$ .	211
Figure IV.5	Distribution de l'erreur télémétrique obtenue pour $\tau=0,8$ . .	212
Figure IV.6	Comparaison entre l'erreur télémétrique obtenue et le bruit ajouté . . . . .	213
Figure IV.7	Distribution de l'erreur en couleur pour $\tau=0,8$ . . . . .	213
Figure IV.8	Distribution de l'erreur d'orientation pour $\tau=0,8$ . . . . .	215
Figure V.1	Montage expérimental simulé pour la détection des surfaces courbes . . . . .	220
Figure V.2	Rayon obtenu en fonction du rayon réel des surfaces courbes numérisées . . . . .	221

Figure V.3	Détection des surfaces cylindriques et sphériques en fonction de la distance au capteur . . . . .	223
Figure V.4	Détection des surfaces cylindriques en fonction de l'inclinaison du cylindre . . . . .	224
Figure V.5	Montage expérimental simulé pour obtenir les résultats du tableau V.1. . . . .	225
Figure V.6	Surface grise éclairée par une lumière blanche avec un niveau de lumière de 100 % . . . . .	228
Figure V.7	Influence de l'éclairage non uniforme sur la détection d'un élément surfacique . . . . .	228
Figure V.8	Effet de la lumière sur la distinction entre les surfaces . . . . .	230

## LISTE DES NOTATIONS ET DES SYMBOLES

## Repères et coordonnées

$tl$	Indice dénotant le repère lié au télémètre laser
$cam$	Indice dénotant le repère lié à la caméra
$img$	Indice dénotant le repère lié à l'image de la caméra
$cap$	Indice dénotant le repère lié au capteur 3D
$\vec{i}, \vec{j}, \vec{k}$	Indices dénotant respectivement les composantes d'un vecteur le long des axes $x$ , $y$ et $z$
$(x, y, z)$	Coordonnées cartésiennes d'un point
$(r, \theta, \varphi)$	Coordonnées sphériques d'un point

## Autres indices

$i, j$	Indices dénotant le numéro d'un point
$k$	Indice dénotant une itération
$S$	Indice dénotant une sphère
$C$	Indice dénotant un cylindre

## Paramètres de segmentation

$N_{iter}$	Nombre d'itérations effectuées pour détecter un plan
$N_{min}$	Nombre de points minimum que doit contenir une surface
$d_{max}$	Distance maximale tolérée d'un point à un plan
$S_D$	Seuil appliqué à la distance entre les points
$S_P$	Seuil appliqué lors de la segmentation photométrique

## Quantités

$n$	Nombre de points d'une surface
$n_{tl}$	Nombre de points dans une acquisition du télémètre laser
$n_{cap}$	Nombre de points dans une acquisition 3D complète
$n_{int}$	Nombre d'acquisitions internes réalisées lors d'une acquisition du télémètre laser
$q_1$	Proportion des points d'un scan appartenant au plan principal
$q_2$	Probabilité de détecter correctement le plan principal dans un scan

## Paramètres de localisation et de fusion

$\tau_{coul}$	Différence de couleur maximale tolérée pour l'appariement des éléments surfaciques
$\tau_{ang}$	Différence d'angle maximale tolérée pour l'appariement des éléments surfaciques
$\tau_{dist}$	Distance maximale tolérée pour l'appariement des éléments surfaciques

## Localisation et fusion

$\vec{n}$	Vecteur normal à une surface plane
$\gamma$	Différence d'orientation entre deux surfaces
$\Delta x, \Delta y, \Delta \gamma$	Correction apportée à la position de la plate-forme
$a, b, c, d$	Paramètres du plan d'équation $ax + by + cz + d = 0$
$\rho$	Poids d'une surface. Correspond au nombre de points que représente cette surface.

## Détection des surfaces courbes

$\vec{P}$	Paramétrisation d'une surface
$e_i(\vec{P}, p_i)$	Erreur en distance du point $p_i$ à la surface paramétrée par $\vec{P}$
$\vec{e}(\vec{P})$	Vecteur formé des $e_i(\vec{P}, p_i)$ pour la surface paramétrée par $\vec{P}$
$f(\vec{P})$	Somme des moindres carrés devant être minimisée par l'algorithme de Levenberg-Marquardt
$\Delta\vec{P}$	Modification apportée à la paramétrisation $\vec{P}$
$J$	Jacobien de $\vec{e}$ autour de $\vec{P}$
$c = (h, k, l)$	Centre d'une sphère ou « centre » d'un cylindre
$r$	Rayon d'une sphère ou d'un cylindre
$\theta_C, \phi_C$	Orientation spatiale d'un cylindre
$\vec{v}_C$	Vecteur directeur unitaire de l'axe d'un cylindre
$(\cdot)$	Premier estimateur d'un paramètre
$(\ddot{\cdot})$	Second estimateur d'un paramètre
$(\bar{\cdot})$	Moyenne d'un ensemble de valeurs

## Autres symboles

$p$	Un point dans l'espace 3D
$h_e$	Hauteur d'eau pour la ligne de partage des eaux
$\tau$	Degré de similarité requis entre deux surfaces

## LISTE DES ANNEXES

ANNEXE I	RÉSUMÉ DE CERTAINES TECHNIQUES DE SEGMENTATION EN SURFACES PLANES . . . . .	182
ANNEXE II	INTÉGRATION DU CAPTEUR 3D À L'ARCHITECTURE LOGICIELLE DE LA PLATE-FORME MOBILE . . . . .	185
II.1	Architecture logicielle . . . . .	185
II.1.1	Couche d'abstraction matérielle . . . . .	185
II.1.2	Couche d'abstraction logicielle . . . . .	187
II.2	Modules existants . . . . .	188
II.3	Modules développés . . . . .	188
ANNEXE III	MODÈLES DE SURFACES COURBES UTILISÉS DANS CE MÉMOIRE . . . . .	191
III.1	Modèle de sphère . . . . .	191
III.2	Modèle de cylindre . . . . .	194
ANNEXE IV	SEGMENTATION D'IMAGES PROVENANT D'UNE SCÈNE SIMULÉE . . . . .	199
IV.1	Métriques utilisées pour l'analyse . . . . .	199
IV.2	Caractéristiques des simulations effectuées . . . . .	201
IV.3	Détection des éléments surfaciques . . . . .	205
IV.4	Erreurs spatiale, photométrique et angulaire . . . . .	209
IV.4.1	Erreur spatiale . . . . .	210
IV.4.2	Erreur photométrique . . . . .	212
IV.4.3	Erreur angulaire . . . . .	214
IV.5	Comparaison avec d'autres travaux . . . . .	216
IV.6	Conclusion . . . . .	217

ANNEXE V	SIMULATIONS SPÉCIFIQUES . . . . .	219
V.1	Détection des surfaces cylindriques et sphériques en fonction de leur rayon . . . . .	220
V.2	Détection des surfaces cylindriques et sphériques en fonction de la distance . . . . .	222
V.3	Détection des surfaces cylindriques en fonction de leur orientation .	223
V.4	Détection des éléments surfaciques en fonction de la différence de leur couleur . . . . .	225
V.5	Détection d'un élément surfacique éclairé de manière non uniforme	227
V.6	Seuil photométrique de détection des éléments surfaciques en présence de lumière . . . . .	227
V.7	Conclusion . . . . .	231

## INTRODUCTION

Le présent projet concerne la production de représentations d'espaces structurés, correspondant par exemple à l'intérieur de bâtiments, possédant les caractéristiques suivantes :

- les représentations doivent combiner, de manière enregistrée, c'est-à-dire en superposition tridimensionnelle exacte, une description de la géométrie (télémétrie) ainsi qu'une description de l'aspect visuel (photométrie) de la scène capturée ;
- les éléments de base de ces représentations doivent être constitués de primitives géométriques ou photométriques intrinsèques à la scène, telles que surfaces ou zones de caractéristiques homogènes, par opposition aux primitives extrinsèques (pixels, voxels, etc.) qui correspondent aux représentations élémentaires courantes ;
- les représentations doivent être extensibles, c'est-à-dire être capables de décrire de façon unifiée des scènes excédant les limites géométriques du champ de vision du capteur utilisé, à partir d'un déplacement du capteur le long d'une série discrète de points de vue.

Nous nommons «cartographie 3D» le processus d'acquisition et de traitement des données télémétriques et photométriques pour obtenir la représentation désirée, ou la «carte», de l'environnement.

Ce type de représentation présente un intérêt dans plusieurs domaines d'application. La cartographie 3D d'un lieu permet d'incorporer ce lieu dans les jeux vidéos et de mener à la création d'effets spéciaux au cinéma. Les représentations de bâtiments en 3D avec couleur sont utiles aussi aux architectes, urbanistes et ingénieurs. De plus, l'usage de ces représentations peut permettre la planification de tâches de

navigation par des plates-formes robotiques mobiles en milieux préalablement inconnus, sinistrés ou hostiles.

La construction de telles représentations impose la résolution de plusieurs sous-problèmes : choix ou conception du capteur approprié, localisation du capteur dans la scène, sélection de la séquence de points de vue d'acquisition, segmentation des données et extraction des primitives descriptives et intégration des différentes vues de façon à créer une représentation étendue de la scène.

Plusieurs capteurs existent pour réaliser l'acquisition de données télémétriques et photométriques enregistrées. Différentes technologies sont employées tels que télémètres laser par mesure directe de temps de vol, par mesure de phase, par balayage auto-synchronisé, systèmes à lumière structurée, et systèmes stéréoscopiques. Certaines de ces technologies sont inappropriées pour la cartographie d'un environnement intérieur, en raison d'une faible précision télémétrique ou d'une portée minimale trop grande, alors que les autres ont un coût trop élevé. Pour ces raisons, plusieurs équipes de recherches ont construit leur propre capteur en combinant un télémètre laser linéique (c'est-à-dire fournissant des données télémétrique dans un plan unique) avec une caméra et en compensant l'aspect linéique du télémètre par un déplacement du capteur en translation (Weingarten et al., 2004) ou en rotation (Surmann et al., 2001; Nüchter et al., 2005). L'approche la plus courante pour combiner les données sensorielles ainsi acquises consiste à convertir les données télémétriques sous forme d'un maillage tridimensionnel et à y projeter les données prises par la caméra (Ramalingam et Lodha, 2003; Pervözl et al., 2004) dans le but d'obtenir une représentation visuelle combinant les deux modalités. Une autre approche, moins utilisée, consiste à associer une donnée photométrique à chaque donnée télémétrique dans le but de segmenter la scène en tenant compte des deux modalités sensorielles (Pujas et Aldon, 1997).

La segmentation est un processus qui consiste à regrouper les données télémétriques et/ou photométriques capturées selon des primitives présentant une homogénéité de propriétés géométriques ou visuelles. Au niveau des données télémétriques, il a été proposé de détecter différents types de surfaces — planes (Hoover et al., 1996; Jiang et al., 2000), quadriques (Gotardo et al., 2004) et polynomiales bivariables (Sabata et al., 1990) — ou de classifier des nuages de points pré-segmentés en surfaces de type connu (plan, cône, sphère et cylindre) (Newman et al., 1993). La segmentation d'images photométrique a également fait l'objet de nombreux travaux comme en témoigne la revue de Lucchese et Mitra (Lucchese et Mitra, 2001). Quelques travaux s'intéressent à la segmentation de données combinées télémétriques-photométriques. Les techniques utilisées (Yu et al., 2001; Pujas et Aldon, 1997) définissent un espace à plusieurs dimensions en combinant position et couleur des données acquises et utilisent des critères de proximité pour diviser les images en nuages de points dont la forme n'est pas connue avec exactitude avant la segmentation.

Le choix des points de vue optimaux pour la production de descriptions étendues est un problème complexe pour lequel différentes stratégies ont été proposées. Par exemple, Nüchter et al. (Nüchter et al., 2003a) utilisent une méthode aléatoire pour déterminer le point de vue menant à l'observation de la zone inconnue la plus grande et Sequeira et al. (Sequeira et al., 1998) détectent les occlusions dans les données acquises et cherchent le point de vue qui élimine le plus grand nombre de ces occlusions. Ce problème n'est pas abordé dans le cadre du présent mémoire, étant donné l'ampleur des travaux réalisés sur les autres aspects.

Afin d'obtenir une représentation étendue, il est nécessaire de pouvoir intégrer dans une même représentation les données provenant des différents points de vue. Cela requiert de connaître avec précision la position du capteur. Une approche privilégiée pour cela consiste à utiliser l'algorithme itératif des points les plus proches (ICP)

(Besl et McKay, 1992). Les éléments détectés plus d'une fois doivent également être fusionnés en un seul élément. Les types de méthodes utilisées — fusion de maillage (Sequeira et al., 1998; Ramalingam et Lodha, 2003), utilisation de structures de données en octree (Nüchter et al., 2003b), etc. — dépendent des représentations choisies. Lorsque la carte est une représentation en primitives géométriques, ces primitives sont extraites à partir des données brutes uniquement lorsque que toutes les acquisitions sont terminées.

Notre projet s'intéresse à tous les aspects du problème de création de représentations étendues, à l'exception du sous-problème de sélection de la séquence de points de vue. Nous supposons donc que les différents points de vue d'acquisition sont spécifiés d'avance. Toutefois leurs positions relatives précises ne sont pas connues et leur détermination fait partie des problèmes abordés. Les solutions développées dans ce mémoire comportent plusieurs contributions innovatrices :

Nous avons conçu un nouveau capteur 3D fournissant des données télémétrique et photométrique registrées. Un télémètre laser fournit la position 3D de points de l'environnement et une caméra permet d'associer à chacun de ces points une donnée photométrique. Le télémètre laser et la caméra sont situés sur une plateforme rotative pour réaliser l'acquisition de données non-coplanaires à partir du même centre optique.

Nous avons choisi un type de représentation dans laquelle les primitives sont des surfaces planes, cylindriques et sphériques. Toutefois, la représentation proposée et sa construction peuvent être généralisées à d'autres types de surfaces. Ces primitives sont extraites à partir d'une segmentation télémétrique dont les contours sont raffinés au moyen de la photométrie. Une segmentation entièrement photométrique est également superposée à la segmentation télémétrique pour ajouter les éléments indétectables par la télémétrie en raison du manque de relief. Nous obtenons ainsi

des segmentations plus détaillées que chaque modalité utilisée individuellement.

Nous proposons un nouvel algorithme de localisation du capteur dérivé de l'algorithme ICP conventionnel. Les modifications permettent d'utiliser les surfaces obtenues à la suite de la segmentation plutôt que les données brutes, ce qui offre un gain de performance appréciable.

La carte complète est mise à jour à chaque nouvelle acquisition ; les surfaces détectées plus d'une fois sont alors fusionnées pour ne former qu'une seule surface de plus grandes dimensions. Aucun post-traitement n'est nécessaire après que la dernière acquisition ait été réalisée.

Dans le cadre de ce projet, nous avons utilisé quelques hypothèses limitatives. Premièrement, l'environnement est statique. Deuxièmement, le capteur est installé sur une plate-forme motrice qui se déplace sur une surface plane. Troisièmement, l'éclairage est constant et uniforme. D'autre part, le problème du déplacement autonome de la plate-forme motrice et celui du choix optimal des points de vue ne sont pas traités dans le cadre de ce projet. Ces tâches sont réalisées par un opérateur.

Ce mémoire est divisé en cinq chapitres. Le chapitre 1 présente une revue de la littérature scientifique au sujet des capteurs d'acquisition 3D, des structures de données pour la représentation 3D, des techniques de segmentation télémétrique et photométrique et des techniques de localisation du capteur pour la création d'une représentation étendue. Le chapitre 2 expose la conception d'un nouveau capteur pour réaliser l'acquisition de données télémétriques et photométriques à haute précision. La méthode de segmentation de ces données pour détecter les primitives géométriques et photométriques fait l'objet du chapitre 3. Au chapitre 4, nous décrivons l'algorithme de localisation de la plate-forme ainsi que le mécanisme

d'intégration des vues multiples. Finalement, au chapitre 5, nous présentons et analysons les résultats obtenus au niveau de la précision de notre capteur, de la qualité de la segmentation réalisée et de la précision de l'intégration des vues multiples. Ces résultats proviennent de simulations et d'expérimentations réalisées dans des environnements réels.

## CHAPITRE 1

### REVUE DE LA LITTÉRATURE

Plusieurs solutions robustes ont été développées pour la cartographie 2D qui consiste à détecter l'intersection de l'environnement 3D avec un plan. Habituellement, le plan de coupe est parallèle au sol et passe par le centre optique du capteur utilisé. Les équipes de recherche se tournent maintenant vers la cartographie 3D qui consiste à créer une représentation tridimensionnelle de l'environnement observé. Le passage à une représentation 3D apporte de nouveaux problèmes à résoudre :

1. Quel(s) capteur(s) utiliser ?
2. Comment représenter l'information obtenue ?
3. Comment traiter cette information pour obtenir la représentation désirée ?

L'objectif de ce chapitre est de décrire l'état de la recherche dans le domaine de la cartographie 3D afin de mieux situer notre travail parmi les recherches déjà réalisées. Nous commençons par une étude des capteurs utilisés par les autres équipes de recherche en robotique mobile. Par la suite, nous examinons les différentes représentations de cartes 3D que l'on retrouve dans la littérature. Finalement, nous résumons les principales techniques et algorithmes mis au point pour traiter les données fournies par les capteurs afin d'obtenir la représentation désirée.

## 1.1 Acquisition des données

Un problème rencontré en cartographie 3D est le choix du capteur à utiliser pour obtenir les données 3D. Nous appellerons « capteur 3D » un capteur qui fournit des données non-coplanaires. En deux dimensions, la majorité des équipes de recherche utilisent un télémètre laser dont toutes les données fournies sont coplanaires. Nous nommerons ce capteur un « capteur 2D ». En trois dimensions, aucun type de capteur ne s'est encore établi comme référence. Bien que de nombreuses compagnies offrent des capteurs 3D — ceux-ci seront étudiés au chapitre 2 — très peu d'équipes de recherche en robotique mobile les utilisent, principalement en raison de leur coût très élevé. La principale tendance en robotique mobile est de fabriquer un capteur 3D, à partir d'un capteur 2D.

### 1.1.1 Créer un capteur 3D à partir d'un capteur 2D

Devant la difficulté de trouver des capteurs 3D satisfaisants, plusieurs équipes de recherche ont travaillé à adapter des capteurs 2D qui ont fait leurs preuves. L'idée consiste à utiliser un télémètre laser qui fournit une coupe 2D de l'environnement et à le mettre en mouvement afin d'obtenir la troisième dimension. Selon le type de mouvement effectué, nous distinguons deux grandes approches : translation et rotation du télémètre.

#### 1.1.1.1 Translation d'un télémètre laser

La première technique consiste à installer un télémètre laser 2D sur une plate-forme mobile. Le déplacement de la plate-forme cause une modification du centre optique du capteur, ce qui permet l'acquisition de données 3D. La technique la

plus simple, utilisée par Hähnel et al. (Hähnel et al., 2003) et par Weingarten et al. (Weingarten et al., 2004), consiste à installer un télémètre laser de sorte que le plan d'acquisition des données soit perpendiculaire au déplacement de la plate-forme. Un second télémètre laser, parallèle au plancher, doit être utilisé pour la localisation et la détection d'obstacles lors du déplacement. Les principaux avantages de cette approche sont la simplicité et la densité de points obtenue. Cette densité est inversement proportionnelle à la vitesse de déplacement de la plate-forme mobile. Cette approche est principalement utile pour la numérisation à haute résolution de grandes surfaces. Früh et Zakhor (Früh et Zakhor, 2002) utilisent une technique similaire pour effectuer des numérisations de façades de bâtiments.

Tel que précisé par Surmann et al. (Surmann et al., 2001), cette technique comporte certains désavantages. Premièrement, la qualité des données acquises est directement dépendante de la qualité du positionnement de la plate-forme. Deuxièmement, il est impossible d'obtenir des données 3D sans déplacer la plate-forme, ce qui rend impossible la détection de tous les obstacles.

Une autre technique semblable, utilisant un seul télémètre laser, a été utilisée par Pujas et Aldon (Pujas et Aldon, 1997) et par Gerbaud (Gerbaud, 2004) avec un seul télémètre laser. Le télémètre est installé en hauteur sur la plate-forme mobile et est orienté à angle de façon à ce que le plan d'acquisition intersecte le sol à une certaine distance devant la plate-forme. De cette manière, le déplacement de la plate-forme mobile permet d'obtenir des données sur l'élévation du terrain. Cette technique possède la même limitation que celle de Hähnel et al. de nécessiter un déplacement de la plate-forme pour l'acquisition de données 3D, mais règle partiellement le problème de la détection préventive des obstacles devant la plate-forme (par exemple, un trou peut maintenant être détecté et évité).

### 1.1.1.2 Rotation d'un télémètre laser

La seconde approche consiste à faire pivoter un télémètre laser sur lui-même. La méthode employée par Surmann et al. (Surmann et al., 2001), consiste à faire pivoter le capteur de haut en bas et à prendre des acquisitions à intervalles réguliers lors de la rotation du capteur. Le principal avantage est que des données 3D fiables sur l'environnement peuvent être recueillies sans effectuer un déplacement de la plate-forme mobile. Puisque la position angulaire du télémètre laser est connue de façon absolue grâce à un encodeur optique, les données recueillies lors d'un scan complet de haut en bas, peuvent être parfaitement positionnées dans l'espace 3D. Il n'y a donc aucune accumulation d'erreur due au déplacement de la plate-forme mobile et aucune déformation géométrique n'est présente à l'intérieur d'une numérisation 3D. Le principal inconvénient est que la plate-forme doit être à l'arrêt pendant une acquisition complète du capteur 3D. Cette approche est également utilisée par Jensfelt et Kristensen (Jensfelt et Kristensen, 2001).

Brenneke et al. (Brenneke et al., 2003) ont développé une autre méthode où le capteur tourne cette fois de gauche à droite et de droite à gauche sur 360°. Cette approche permet d'agrandir l'angle d'ouverture du capteur. En fait, ce capteur peut couvrir pratiquement 100 % de l'espace observable en une seule numérisation. Un capteur similaire a également été conçu par Nüchter et al. (Nüchter et al., 2005).

### 1.1.1.3 Combinaison télémétrie/photométrie

Pour obtenir des données photométriques enregistrées aux données télémétriques, deux approches sont utilisées. La première de ces approches consiste à utiliser l'intensité du signal laser reçu comme information monochrome de la réflectance des objets. Cette technique, utilisée par Surmann et al. (Surmann et al., 2003) est peu

fiable, mais possède l'avantage de fournir des données parfaitement enregistrées. La seconde approche consiste à utiliser une caméra externe au capteur télémétrique. Cette méthode est plus complexe, car elle demande une calibration du système laser-caméra et ne donne généralement pas des données parfaitement enregistrées. Toutefois, l'erreur de registration est généralement négligeable et est largement surpassée par l'amélioration de la qualité visuelle du résultat obtenu.

La majorité des recherches effectuées utilisent une technique en trois étapes : acquérir les données télémétriques, traiter les données télémétriques pour en extraire des surfaces — généralement un maillage — et ensuite projeter les surfaces dans les images fournies par la caméra afin de pouvoir « coller » des textures sur chaque surface. Cette technique est utilisée, entre autre, par Sequeira et al. (Sequeira et al., 1998), Früh et Zakhor (Früh et Zakhor, 2002), Ramalingam et Lodha (Ramalingam et Lodha, 2003) et Pervözl et al. (Pervözl et al., 2004). L'article de Sequeira et al. se distingue par l'utilisation des données de réflectance du télémètre laser pour aider à la registration avec la caméra. Dans tous ces cas, la photométrie est utilisée uniquement à des fins d'amélioration de l'apparence visuelle et n'est pas utilisée pour offrir à la plate-forme mobile une meilleure compréhension de la scène. Pujas et Aldon (Pujas et Aldon, 1997) utilisent la télémétrie et la photométrie de manière combinée pour mieux interpréter la scène. Pour ce faire, ils attribuent une couleur à chaque point fourni par la télémétrie, plutôt que d'appliquer une texture aux surfaces extraites.

En robotique mobile, l'utilisation conjointe de la télémétrie et de la photométrie, pour des fins autres que visuelles est très rare. Lallement et al. (Lallement et al., 1998) présentent un exemple d'utilisation combinée de la photométrie et de la télémétrie pour détecter et classifier des points d'intérêt en milieu intérieur. Un télémètre laser est utilisé pour la détection et une caméra est utilisée pour la classification. Cet article démontre l'utilité de combiner ces deux modalités sensorielles.

### 1.1.2 Autres approches

Quelques équipes de recherche ont développé des capteurs qui n'utilisent pas de télémètre laser 2D. El-Hakim et al. (El-Hakim et al., 1997) utilisent le capteur BIRIS développé par le Conseil national de recherches Canada (CNRC). Ce capteur, d'une portée de 0,5 m à 4 m, utilise un laser projetant une ligne et une caméra à deux ouvertures pour trianguler la position des objets. Le CNRC a également développé une technologie appelée « balayage auto synchronisé d'un point laser » (Godin, 2003; CNRC, 2006). Cette technologie est employée dans deux capteurs du CNRC : le « scanner laser 3D couleur à haute résolution » (CNRC, 2006) et le « scanner laser 3D à grand volume et à accès aléatoire » (CNRC, 2006). Le premier capteur sert à la numérisation 3D avec couleur de petits objets et a été développé pour la numérisation d'oeuvres d'art. Le second capteur permet l'acquisition d'images 3D monochromes avec un portée variant de 0,5 m à 10 m.

La vision stéréoscopique peut également être utilisée pour créer des cartes 3D comme le montrent Sáez et Escolano (Sáez et Escolano, 2004). Cette approche présente plusieurs difficultés et est rarement utilisée.

### 1.1.3 Capteur utilisé

Nous verrons au chapitre 2 que les capteurs 3D disponibles commercialement sont inappropriés à la cartographie 3D en milieu intérieur, ce qui explique que les équipes de recherches construisent leur propre capteur. Nous avons donc conçu, nous aussi, notre propre capteur pour réaliser l'acquisition de données télémétriques et photométriques enregistrées. Notre capteur combine l'approche de Surmann et al. (Surmann et al., 2001) et de Jensfelt et Kristensen (Jensfelt et Kristensen, 2001) pour l'acquisition des données télémétriques et l'approche de Pujas et Aldon (Pujas et Aldon,

1997) pour l'acquisition des données photométriques. Un télémètre laser en rotation est utilisé pour obtenir les données télémétriques et une caméra est installée pour ajouter les informations photométriques à chaque point fourni par le télémètre laser. Le fonctionnement de notre capteur est décrit en détail au chapitre 2.

## 1.2 Différentes représentations de cartes 3D

Un autre problème important en cartographie 3D est le choix de la méthode de représentation de l'information. Par rapport aux cartes 2D, l'ajout de la troisième dimension augmente considérablement la quantité d'information à traiter. Le choix d'une représentation efficace dépend de la raison principale pour laquelle la carte est générée. Notre revue de la littérature a révélé que plusieurs recherches ont été menées avec l'objectif principal de créer une carte qui soit visuellement attrayante et proche de la réalité<sup>1</sup>. Une telle carte doit capturer tous les détails photométriques de la scène. Dans cette optique, les représentations les plus utilisées sont les maillages et les octrees. Lorsque la carte est utilisée directement par les robots mobiles ou manipulateurs pour interagir avec l'espace environnant, il est utile de sacrifier le niveau de détail visuel pour obtenir une meilleure synthèse des données recueillies. Dans ce cas, les données sont regroupées — segmentées — en entités ayant des caractéristiques semblables souvent appelées «objets». Dans cette section, nous effectuons une revue des structures de données les plus utilisées en robotique mobile pour la cartographie 3D. Il est à noter que ces modes de représentation ne sont pas mutuellement exclusifs. Par exemple, les maillages peuvent être utilisés pour représenter visuellement les objets qui composent la scène ou pour définir les portions d'objets réellement détectées.

---

<sup>1</sup>Le terme anglais employé est « photorealistic representation ».

### 1.2.1 Maillage

L'utilisation de maillages est très courante pour produire des cartes 3D avec une bonne qualité visuelle. Les données télémétriques obtenues sont transmises à un algorithme de maillage qui crée une triangulation de la surface des objets. Souvent, des photos prises par une caméra sont projetées sur le maillage obtenu lors de l'affichage à l'écran. Nous retrouvons une utilisation des maillages dans les travaux de El-Hakim et al., (El-Hakim et al., 1997), Axelsson (Axelsson, 1999), Thrun et al. (Thrun et al., 2000) et Pervözl et al. (Pervözl et al., 2004).

Un aspect important de l'utilisation des maillages est la technique de fusion utilisée pour incorporer une nouvelle vue à un modèle en construction. Deux techniques sont présentées par Sequeira et al. (Sequeira et al., 1998) et Ramalingam et Lodha (Ramalingam et Lodha, 2003). La technique proposée par Sequeira et al. (Sequeira et al., 1998) consiste à détecter les chevauchements entre les deux vues puis à supprimer les portions de la vue la moins précise de manière à éliminer le chevauchement. Un algorithme de raccord est ensuite utilisé pour lier les maillages de chaque vue. Ramalingam et Lodha (Ramalingam et Lodha, 2003) utilisent une approche qui consiste à utiliser des enveloppes convexes pour détecter les portions de maillage à supprimer et à trianguler de nouveau. Beaucoup de recherches sont également effectuées pour simplifier des maillages existants afin de réduire la taille de mémoire utilisée et accélérer l'affichage. Une telle méthode est présentée par Hoppe (Hoppe, 1996) et par Garland et Heckbert (Garland et Heckbert, 1997). La méthode de Hoppe possède l'avantage d'être réversible. Cette technique est très utile pour les applications demandant initialement un niveau de détail faible et éventuellement plus élevé.

Il ne faut pas confondre ces algorithmes de simplification de maillages avec les algorithmes de segmentation en surfaces mathématiques qui seront présentés à la

section 1.2.3. Dans le premier cas, l'objectif en est un d'optimisation alors que dans le second cas, l'objectif est de décrire l'environnement par les types de surfaces qu'on y retrouve. Hähnel et al. (Hähnel et al., 2003) et Früh et Zakhor (Früh et Zakhor, 2002) combinent d'ailleurs ces deux techniques. Dans le premier cas, une segmentation en surfaces planes permet de diminuer le bruit de mesure du capteur télémétrique avant de procéder à la création du maillage. Dans le second cas, une segmentation en surfaces planes permet de différencier les façades des bâtiments des autres éléments indésirables, comme les arbres, avant de procéder au maillage de la façade.

### 1.2.2 Octree

L'octree est la méthode la plus simple de représenter l'information obtenue et possède l'avantage de former une structure hiérarchique. Le principe consiste à subdiviser l'espace à représenter en huit sous-régions de volume égal (division par 2 selon chaque axe). Chaque région est ensuite subdivisée en huit de manière récursive jusqu'au niveau de détail désiré. Toutefois, il n'est pas nécessaire de subdiviser une région si toutes les subdivisions possèdent les mêmes caractéristiques. Cela permet de diminuer la taille de mémoire nécessaire et d'accélérer les algorithmes de recherche. Cette structure de données est utilisée entre autres par Payeur et al. (Payeur et al., 1998) et par Nüchter et al. (Nüchter et al., 2003b). L'utilisation des octrees — ou de grilles en général — est également utile pour la reconstruction de scènes 3D à partir de photos prises de points de vue différents. Cette technique est utilisée par Seitz et Dyer (Seitz et Dyer, 1999) et par Mueller et al. (Mueller et al., 2004). Dans le second article, un octree est utilisé pour la modélisation spatiale et est ensuite transformé en maillage pour la visualisation.

La version à deux dimensions de l'octree, appelée « quadtree », est utilisée depuis plusieurs années pour optimiser la recherche de trajectoires. Des exemples d'algorithmes utilisant un quadtree peuvent être retrouvés dans les articles de Zelinsky (Zelinsky, 1992) et de Kambhampati et Davis (Kambhampati et Davis, 1986). Ces techniques peuvent être facilement généralisées en 3D en utilisant un octree.

### 1.2.3 Subdivision de la scène en objets

Un autre mode de représentation consiste à segmenter la scène en objets. Le terme « objet » réfère ici à un ensemble de points qui ont des caractéristiques communes. Idéalement, ces ensembles de points devraient correspondre à ce que nous appelons « objet » dans le langage courant (livre, chaise, table, etc.).

Beaucoup de recherches ont été effectuées en robotique mobile pour identifier des surfaces planes dans une image 3D. Cela s'explique par le fait que la majorité des projets de recherche en cartographie 3D par des plates-formes mobiles sont réalisés en milieu intérieur où les murs sont prédominants. Hähnel et al. (Hähnel et al., 2003) présentent deux méthodes pour réaliser l'extraction de surfaces planes. La première méthode consiste à détecter des zones localement planes représentées sous forme de polygones. Ces polygones sont ensuite fusionnés suivant certaines règles préétablies pour former des surfaces planes de plus grande dimension. La seconde méthode consiste à extraire des segments de droite. Ces segments servent à former un histogramme dont les maximums fournissent les orientations des surfaces planes recherchées. Surmann et al. (Surmann et al., 2001) commencent eux aussi par une détection de segments de droite. Les segments ayant une orientation semblable et ayant des extrémités suffisamment proches les unes des autres sont fusionnés pour former des surfaces planes. Il existe également des méthodes probabilistes, dont celle de Weingarten et al. (Weingarten et al., 2004) qui permet de prendre en

considération les incertitudes sur les mesures afin d'obtenir de meilleures équations de plans. La photométrie peut également être ajoutée à la télémétrie pour aider à la détection des objets. Malgré les travaux de Pujas et Aldon (Pujas et Aldon, 1997), très peu de recherches ont été réalisées en ce sens en robotique mobile.

La segmentation en surfaces planes est utilisée par Nüchter et al. (Nüchter et al., 2003c) pour effectuer une analyse sémantique de la scène qui permet d'identifier les murs, le plancher, le plafond et les portes.

À notre connaissance, seule l'équipe de recherche de Surmann et al. regroupe les plans obtenus pour former des parallépipèdes carrés (Surmann et al., 2001). Certaines recherches ont montré qu'une segmentation de la scène en volumes peut être très efficace pour réaliser les tâches de planification de trajectoire et d'évitement d'obstacles. Sappa et Garcia (Sappa et Garcia, 2004) ont montré comment une structure hiérarchique de volumes sphériques accélère la recherche de collisions en 3D alors que Rimon et Boyd (Rimon et Boyd, 1997) ont montré comment réaliser la détection de collisions en utilisant des ellipsoïdes.

#### 1.2.4 Cartes hiérarchiques

La segmentation de la scène en objets est souvent accompagnée d'une structure hiérarchique. La lecture de nombreux articles en robotique mobile révèle qu'il existe deux notions de «représentation hiérarchique».

La première notion considère la hiérarchie comme un ensemble de niveaux de détails avec pour principal but l'optimisation des algorithmes de recherche et d'affichage. Dans cette catégorie, nous retrouvons l'octree et le quadtree. Ainsi, la carte 2D utilisant un quadtree de Kambhampati et Davis (Kambhampati et Davis, 1986) et l'octree utilisé par Mueller et al. (Mueller et al., 2004) sont considérés comme des

cartes hiérarchiques. Dans le même ordre d'idées, les maillages progressifs de Hoppe (Hoppe, 1996) forment également une structure hiérarchique. Ces techniques ont en commun le fait qu'elles ne tiennent pas compte des contours réels des objets présents dans la scène. En fait, ces techniques rigides imposent les frontières. Par exemple, dans le cas d'un octree, chaque axe est coupé en deux exactement au milieu, même si cela ne correspond pas à une frontière d'un objet. En conséquence, la représentation obtenue n'est pas naturelle dans la mesure où elle ne correspond pas nécessairement à la subdivision physique de la scène.

La seconde notion de « représentation hiérarchique », consiste à diviser la scène étudiée en objets en se basant sur les frontières naturelles de ceux-ci. Ces objets sont ensuite réunis en groupes d'objets ayant des caractéristiques semblables. Garcia et al. (Garcia et al., 1999) présentent une technique de fusion d'objets dans laquelle deux objets situés très proches l'un de l'autre ont une forte chance d'être regroupés. Puis ces groupes sont à leur tour fusionnés jusqu'à l'obtention d'un seul groupe qui contient la scène en entier. La description hiérarchique ainsi obtenue tient compte des relations spatiales entre les objets et est beaucoup plus naturelle que les représentations hiérarchiques basées sur les octrees et les maillages.

À notre connaissance, ce second type de carte hiérarchique n'est pas encore utilisé en robotique mobile. Cela est probablement dû à la complexité de segmenter les données brutes en objets de forme et de volume variés. Toutefois, plusieurs recherches démontrent que la seconde notion de représentation hiérarchique est utile à l'identification et à la reconnaissance des objets. Lyer et al. (Lyer et al., 2004) utilisent une description hiérarchique des objets pour classifier ceux-ci selon leur forme. Il est ensuite possible de rechercher un objet dont la forme ressemble à un autre objet ou à un modèle connu. Dans le même ordre d'idées, Grabska et al. (Grabska et al., 2003) présentent l'intérêt de cette technique en design.

### 1.2.5 Représentation adoptée dans ce mémoire

La représentation choisie pour nos cartes 3D est une représentation hiérarchique des surfaces des objets. Nous classifions les surfaces en trois types : surfaces planes, surfaces cylindriques et surfaces sphériques. D'autres types de surfaces pourraient être aisément ajoutés. Les surfaces sont subdivisées en fonction de la couleur en plus petites surfaces appelées « éléments surfaciques ». Les travaux d'analyse sémantique de Nüchter et al. (Nüchter et al., 2003c) pourraient alors être directement appliqués à cette représentation. Des travaux ultérieurs pourront être effectués pour ajouter un niveau hiérarchique supérieur où les faces d'un même objet seront fusionnées pour obtenir une représentation en objets. D'autres niveaux hiérarchiques pourraient être ensuite ajoutés en appliquant les travaux de Sappa et Garcia (Sappa et Garcia, 2004). L'usage d'une représentation hiérarchique permet l'utilisation de techniques de reconnaissance et de classification des objets à partir de leur structure hiérarchique semblables à celle proposée par Lyer et al. (Lyer et al., 2004).

## 1.3 Traitement des données

Pour obtenir la représentation désirée à partir des données brutes fournies par le capteur, plusieurs traitements doivent être effectués. Cette section porte sur la segmentation des données et sur l'intégration des données pour obtenir une carte étendue de l'environnement observé.

### 1.3.1 Segmentation

Nous avons choisi une représentation de la scène basée sur une segmentation en surfaces. Pour y arriver, plusieurs méthodes de segmentation peuvent être utilisées.

### 1.3.1.1 Segmentation basée sur la télémétrie

La forme de segmentation la plus retrouvée en robotique mobile pour la cartographie est la segmentation basée uniquement sur la télémétrie. Les techniques qui y sont actuellement employées diffèrent considérablement de celles employées dans le domaine du traitement d'images, et ce, même si la tâche à accomplir est la même. Nous avons déjà mentionné à la section 1.2.3 les principales méthodes de segmentation utilisées en robotique mobile. Dans cette section, nous allons discuter des autres techniques de segmentation en surfaces développées dans le domaine du traitement d'images. À notre connaissance, ces techniques n'ont pas encore été adaptées à la robotique mobile.

Un des principaux articles sur la segmentation en surfaces planes est celui de Hoover et al. (Hoover et al., 1996) où est présentée une méthodologie de comparaison d'algorithmes de segmentations. Les auteurs y définissent plusieurs métriques et bancs d'images qui sont utilisés pour évaluer quantitativement les algorithmes étudiés. Quatre algorithmes de segmentation sont alors présentés et analysés. Quatre ans plus tard, Jiang et al. (Jiang et al., 2000) poursuivent l'étude et présentent trois nouveaux algorithmes. Nos recherches ont mis à jour onze techniques de segmentations qui ont été évaluées au moyen de la méthode de Hoover et al : USF, WSU, UB et UE (Hoover et al., 1996), OU, PPU et UA (Jiang et al., 2000), UFPR version 1 (Bellon et Silva, 2002), version 2 (Silva et al., 2002) et version 3 (Gotardo et al., 2003) et ALKS (Lee et al., 1998). Ces techniques sont décrites brièvement à l'annexe I.

Selon l'étude réalisée par Hoover et al. (Hoover et al., 1996) comparant les techniques de segmentation USF, WSU, UB et UE, la technique WSU est celle qui obtient le plus haut taux de segmentation excessive, de segmentation insuffisante et de mauvaises détections et le plus bas taux de bonnes détections. À l'opposé, la

technique de segmentation UE donne généralement de meilleurs résultats que les autres. Toutefois, la nécessité de devoir ajuster 12 paramètres rend, selon nous, la solution trop dépendante du type de scène observée. La technique USF de son côté a l'inconvénient d'être beaucoup trop lente (30 images segmentées en 117 minutes) pour pouvoir être appliquée efficacement à la robotique mobile. La technique UB est de loin la plus rapide de celles étudiées, mais comporte plusieurs seuils dont les valeurs semblent dépendre des scènes observées. Les résultats de Jiang et al. (Jiang et al., 2000) ont montré que les techniques OU, PPU et UA n'offrent aucune amélioration par rapport à celles présentées dans l'article de Hoover et al. Les travaux de Bellon et Silva (Bellon et Silva, 2002) et de Silva et al. (Silva et al., 2002) sur les techniques UFPR1 et UFPR2 présentent des approches intéressantes, mais ne fournissent pas de données quantitatives permettant de juger de leur qualité. La technique UFPR3 présentée par Gotardo et al. (Gotardo et al., 2003) donne de très bons résultats et montre que l'approche d'extractions itératives de surfaces planes est très robuste. L'article de Lee et al. (Lee et al., 1998) sur l'algorithme ALKS ne fournit malheureusement pas de données quantitatives de manière suffisante pour pouvoir juger de la qualité de l'algorithme pour la segmentation de surfaces planes. Toutefois, puisque cet algorithme utilise l'extraction itérative de surfaces planes et que l'estimateur LKS est censé être plus robuste que l'estimateur MSAC, nous pouvons nous attendre à ce que la technique soit très fiable.

Outre la segmentation en surfaces planes, d'autres techniques existent pour segmenter les images 3D en surfaces plus complexes. Plusieurs approches sont utilisées selon les types de surfaces mathématiques recherchées. La technique de Sabata et al. (Sabata et al., 1990) consiste à segmenter les images en surfaces polynomiales bivariées d'ordre spécifié à l'avance. Gotardo et al. (Gotardo et al., 2004) ont présenté une technique de segmentation en surfaces planes et en quadriques. Ces méthodes fournissent de bons résultats, mais ont l'inconvénient de fournir des résultats qui ne

sont pas toujours représentatifs des formes réelles. Par exemple, le bruit de mesure peut faire en sorte qu'une portion de colonne cylindrique soit mieux représentée par une portion d'ellipsoïde, ce qui est plutôt fâcheux si on veut rechercher un objet d'après sa description géométrique.

Newman et al. (Newman et al., 1993) proposent une approche qui consiste à classer des points présegmentés dans l'un des modèles suivants : plan, cylindre, sphère et cône. Toutefois, la technique proposée utilise des algorithmes différents pour chaque type de surface. Ainsi, la classification des surfaces ne peut pas être facilement généralisée à d'autres modèles de surfaces.

Il existe également des techniques de segmentation des données télémétriques en surfaces aux propriétés inconnues. Par exemple, Specht et Devy (Specht et Devy, 2004) utilisent une version modifiée de l'algorithme de la balle pivotante<sup>2</sup>. Cet algorithme regroupe en un seul objet les points situés suffisamment proches les uns des autres et crée automatiquement un maillage. D'autres techniques du même genre, comme les cubes marchants<sup>3</sup> (Lorenson et Cline, 1987) et les formes alpha<sup>4</sup> (Edelsbrunner et Mücke, 1994) sont également proposées.

### 1.3.1.2 Segmentation basée sur la photométrie

De nombreuses recherches ont été effectuées en traitement d'images pour segmenter des images photométriques. Les différentes techniques développées se regroupent principalement en deux catégories<sup>5</sup> :

---

<sup>2</sup>Traduction libre de « ball-pivoting algorithm ».

<sup>3</sup>Traduction libre de « marching cubes ».

<sup>4</sup>Traduction libre de « alpha shapes ».

<sup>5</sup>Nous pouvons également ajouter une catégorie contenant les techniques qui modélisent la physique des propriétés lumineuses des surfaces. Pour plus de renseignements, le lecteur peut consulter l'article de Lucchese et Mitra.

- Segmentation basée sur les caractéristiques photométriques ;
- Segmentation basée sur la répartition spatiale des pixels.

Nous allons résumer ici très brièvement ces deux catégories. Pour un résumé plus approfondi, le lecteur est invité à lire l'article de Lucchese et Mitra (Lucchese et Mitra, 2001).

#### **1.3.1.2.1 Techniques basées sur les caractéristiques photométriques**

Les techniques de segmentation basées sur les caractéristiques photométriques tentent de regrouper les pixels de même couleur, de même teinte ou de même intensité. Parmi ces techniques, nous retrouvons le seuillage d'histogrammes et le k-means clustering et ses variantes. La technique du seuillage d'histogrammes consiste à calculer l'histogramme d'une caractéristique spécifique de l'image, par exemple la teinte, et à partitionner cet histogramme en modes en recherchant les extrema locaux. Comaniciu et Meer (Comaniciu et Meer, 1997) utilisent un histogramme tridimensionnel des pixels représentés dans l'espace de couleur LUV et un algorithme appelé « Mean Shift Algorithm » pour trouver itérativement les extrema locaux qui correspondent aux régions recherchées. La technique du k-means clustering consiste à diviser l'image en k groupes et à trouver les caractéristiques de ces groupes pour qu'ils soient les plus homogènes possible. Par exemple, Zhang et Wang (Zhang et Wang, 2000) effectuent un k-means clustering basé sur la teinte et l'intensité des pixels.

#### **1.3.1.2.2 Techniques utilisant la répartition spatiale des pixels**

Les techniques de segmentation basées sur la répartition spatiale des pixels de l'image cherchent à créer des régions de pixels spatialement proches les uns des

autres et ayant des caractéristiques semblables. Nous y retrouvons les techniques de split and merge, de grossissement de région et de détection de contours. La technique split and merge consiste à fusionner et à disloquer itérativement des régions pour maximiser une fonction objective qui évalue l'homogénéité des régions. La technique de grossissement de régions consiste à constituer une région restreinte de pixels et à grossir itérativement la région en englobant les pixels environnants qui ont des caractéristiques photométriques semblables. Les techniques de détection de contours consistent à rechercher les contours des objets à l'intérieur des images. Cette détection est obtenue en évaluant le gradient de couleur dans l'image. Par exemple, Gao et al. (Gao et al., 2001) présentent une technique de segmentation dans laquelle des opérations morphologiques de dilatation et d'érosion sont appliquées aux contours de l'image. Des zones homogènes sont ensuite détectées pour servir de marqueur à une technique de ligne de partage des eaux.

La ligne de partage des eaux est une technique de segmentation morphologique qui consiste à considérer l'image de gradient comme une image topographique où les zones de faible gradient sont des creux et les zones de fort gradient sont des montagnes. L'image est ensuite inondée lentement. Plusieurs lacs se forment dans les creux et se joignent éventuellement. Les jonctions des lacs forment les lignes de partage des eaux. Ces lignes sont les contours des régions recherchées. La première utilisation de la ligne de partage des eaux est attribuée à Digabel et Lantuejoul (Digabel et Lantuejoul, 1978). Par la suite, la technique a été modifiée et améliorée et il en existe maintenant plusieurs versions et variantes. Pour une revue des techniques et de leurs particularités, le lecteur est invité à lire l'article de Roerdink et Meijster (Roerdink et Meijster, 2000). La principale difficulté lors de l'application de cette technique est qu'elle crée généralement une segmentation excessive de l'image. Pour résoudre ce problème, deux stratégies sont généralement employées : utiliser des marqueurs et fusionner les régions obtenues. L'utilisation des marqueurs

consiste à identifier à l'avance le nombre de régions voulues et un pixel appartenant à chaque région. La ligne de partage des eaux agit alors comme une technique de grossissement de région à partir de ces pixels. L'autre stratégie consiste à fusionner les régions selon un critère préétabli. Par exemple, Eom et al. (Eom et al., 2002) proposent une approche où les régions sont fusionnées si le nombre de pixels à gradient élevé sur le contour commun entre les régions est inférieur à un seuil prédéterminé.

### **1.3.1.3 Segmentation combinant télémétrie et photométrie**

Quelques recherches ont été effectuées pour segmenter des images en combinant les informations télémétriques et photométriques. Les stratégies employées sont généralement des algorithmes de clustering ou de grossissement de régions dans un espace à plusieurs dimensions incluant des dimensions géométriques et des dimensions photométriques. Ainsi, Shen et Spann (Shen et Spann, 1997) segmentent des images 2D et 3D avec des niveaux de gris. Ils créent un graphe de connectivité dans lequel les points de l'image forment les noeuds. Deux noeuds sont reliés par une arête s'ils sont adjacents dans l'image. Un algorithme des plus proches voisins est ensuite utilisé pour segmenter ce graphe. Les plus proches voisins sont calculés en utilisant un modèle linéaire du niveau de gris des points en fonction de leur position spatiale. Une autre technique utilisant un graphe de connectivité est présentée par Yu et al. (Yu et al., 2001). Le graphe est partitionné en régions en utilisant l'algorithme normalized cut. Cet algorithme consiste à mettre un poids sur chaque arête du graphe et à diviser chaque région en deux en minimisant le coût de la séparation. Le poids associé à chaque arête est fonction de la distance spatiale entre les points, de leur différence de vecteur normal et de leur différence d'intensité lumineuse. Trouver la coupure optimale est un problème NP-complexe. Une autre approche, utilisant une technique de grossissement de régions, est pro-

posée par Pujas et Aldon (Pujas et Aldon, 1997). Partant d'un point de l'image, d'autres points sont agglomérés s'ils sont situés assez proche de ce point. La définition de distance utilisée combine la position spatiale, les informations de couleur et les vecteurs normaux évalués en chaque point.

Toutes ces approches ont en commun le fait qu'elles définissent, implicitement ou explicitement, un graphe de connectivité et qu'elles utilisent une métrique de distance combinant télémétrie et photométrie pour partitionner le graphe. Ces approches donnent de bons résultats et ont l'avantage de tenir compte de manière simultanée de toutes les modalités sensorielles. Toutefois, les métriques de distance utilisées combinent des données ayant des interprétations physiques complètement différentes et des constantes arbitraires doivent alors être utilisées pour uniformiser les données. Ces métriques ont donc peu de sens physique. De plus, ces techniques peuvent produire des nuages de points dont la forme ne correspond pas à des modèles de surfaces facilement identifiables, ce qui est peu pratique pour les algorithmes de classification, de recherche et d'identification d'objets.

#### **1.3.1.4 Méthode de segmentation utilisée dans ce mémoire**

Les données fournies par notre capteur 3D sont segmentées en tenant compte de la télémétrie et de la photométrie, mais, plutôt que d'utiliser l'approche traditionnelle qui consiste à combiner les deux modalités sensorielles en une seule métrique de distance, nous effectuons deux segmentations indépendantes. Une première segmentation est basée sur la télémétrie. Une technique itérative d'extraction de surfaces permet de segmenter l'image en surfaces planes. Ces surfaces sont ensuite utilisées pour extraire des surfaces plus complexes comme les surfaces sphériques et cylindriques. Une seconde segmentation, basée sur la photométrie, est réalisée au moyen d'un algorithme de ligne de partage des eaux. Les deux segmentations sont ensuite

fusionnées pour obtenir une segmentation complète basée à la fois sur la télémétrie et la photométrie. À notre connaissance, cette approche n'a jamais été utilisée. Cette technique possède les avantages de tirer profit des recherches de pointes réalisées indépendamment pour la segmentation de chaque modalité sensorielle et d'augmenter la robustesse en fournissant une segmentation cohérente même si l'une des modalités sensorielles fait défaut. Notre méthode de segmentation est détaillée au chapitre 3.

### 1.3.2 Intégration de vues multiples

Un des principaux problèmes de la cartographie est le traitement des occlusions : lorsqu'un objet est détecté, celui-ci cache les objets situés derrière lui. Afin d'obtenir une carte complète de l'environnement, il est nécessaire de prendre plusieurs images à partir de points de vue différents, d'où l'intérêt d'installer le capteur sur une plate-forme mobile. Pour fins d'analyse, le problème peut être divisé en trois sous-problèmes :

1. Déterminer les points de vue optimaux (où positionner le capteur pour obtenir de nouvelles données ?) ;
2. Déterminer les positions relatives de chaque point de vue (où insérer les nouvelles données dans la carte en construction ?) ;
3. Insérer les nouvelles données dans une structure de données préexistante (comment modifier la carte en construction pour y ajouter les données ?).

Le premier problème est complexe, car il nécessite de réaliser un compromis entre l'objectif de maximiser la quantité de nouvelles données, dans le but d'accélérer le processus de cartographie, et le besoin d'une certaine redondance pour optimiser l'intégration des nouvelles données. Ce problème a déjà fait l'objet de plusieurs

recherches, dont celles de Nüchter et al. (Nüchter et al., 2003a), Sequeira et al. (Sequeira et al., 1998) et Whaite et Ferrie (Whaite et P., 1997). Ce problème n'est pas traité dans le cadre de ce mémoire. Le deuxième problème a été traité non seulement dans le cas d'une seule plate-forme mobile, mais également dans le cas de plusieurs plates-formes et est décrit à la prochaine section. Le troisième problème est directement lié à la structure de données utilisée et il n'existe pas de solution générale.

### 1.3.2.1 Localisation

Pour positionner les nouvelles données dans la carte en création, l'approche la plus simple consiste à se baser uniquement sur les informations d'odométrie fournies par la plate-forme mobile. Malheureusement, cette méthode de positionnement accumule l'erreur très rapidement, de sorte que la position de la plate-forme devient vite erronée et que la carte générée est déformée.

L'algorithme le plus populaire pour la localisation d'une plate-forme dans un environnement connu ou partiellement connu est l'algorithme itératif des points les plus proches, généralement connu sous l'acronyme ICP<sup>6</sup>. Cet algorithme permet de prendre deux ensembles de points et de calculer la transformation géométrique devant être réalisée pour les superposer l'un sur l'autre le mieux possible. L'algorithme original provient de Besl et McKay (Besl et McKay, 1992). Plusieurs variantes de l'algorithme ont ensuite été proposées pour en augmenter la rapidité ou l'efficacité. Parmi celles-ci, notons l'algorithme parallèle de Langis et al. (Langis et al., 2001) et l'utilisation des points d'intérêts proposée par Sharp et al. (Sharp et al., 2002).

---

<sup>6</sup>Traduction libre de « Iterative Closest Points Algorithm ».

Une autre méthode de localisation populaire est l'utilisation d'un filtre de Kalman. Ce filtre modélise le bruit de mesure et le déplacement de la plate-forme mobile afin d'utiliser les nouvelles mesures prises par la plate-forme pour améliorer l'estimation de sa position. Cette approche permet également de combiner les données sensorielles provenant de plusieurs plates-formes mobiles. Des méthodes de localisation mutuelles entre plates-formes mobiles basées sur la détection de points de repères avec filtre de Kalman ont été proposées par Zalzal et Cohen (Zalzal et Cohen, 2006) et par Stroupe et Balch (Stroupe et Balch, 2002).

### 1.3.2.2 Cartes topologiques

Pour fusionner ensemble plusieurs cartes sans information initiale sur la position relative entre ces cartes, il est utile d'utiliser une approche topologique. La fusion de plusieurs cartes n'étant pas un objectif de ce mémoire, nous allons décrire très brièvement ce sujet. Une carte topologique est généralement un graphe dans lequel les noeuds représentent des lieux importants facilement identifiables, par exemple des locaux et des intersections, alors que les arêtes représentent les liens entre ces lieux, généralement les corridors. Il s'agit d'une représentation simplifiée de l'environnement. Huang et Beevers (Huang et Beevers, 2005) utilisent une telle représentation pour fusionner des portions de cartes d'un même environnement. Cela est réalisé à l'aide d'un algorithme de similitude entre deux graphes qui permet d'identifier les portions de chaque carte qui sont identiques d'un point de vue topologique. Leur carte topologique est obtenue en utilisant une technique de suivi de mur lors de la cartographie. Les discontinuités observées dans les murs servent à détecter les intersections. Konolige et al. (Konolige et al., 2003) réalisent des fusions de cartes en détectant manuellement les portes, les intersections et les corridors présents dans chaque carte. Ces informations servent de points de repère pour déterminer les zones de chevauchement entre les cartes.

### 1.3.2.3 Méthode de localisation utilisée dans ce mémoire

Pour obtenir une carte complète de l'environnement, plusieurs images doivent être fusionnées entre elles. Pour y arriver, nous utilisons une estimation de la position de la nouvelle image dans la carte. Cette estimation est fournie par l'odométrie de la plate-forme mobile. Au moyen de cette estimation, les surfaces de la nouvelle image sont associées aux surfaces présentes dans la carte et une version modifiée de la technique ICP est utilisée pour calculer la transformation qui minimise l'erreur de positionnement. Nous avons modifié l'algorithme ICP pour qu'il agisse sur les surfaces détectées plutôt que sur les données brutes fournies par le capteur. Le nombre de surfaces étant largement inférieur au nombre de points contenu dans les données brutes, nous obtenons ainsi un gain de performance appréciable. Cette méthode est décrite au chapitre 4.

## 1.4 Conclusion de la revue de la littérature

En bref, cette revue de la littérature a permis de mieux situer le contexte théorique de notre travail sur la base des recherches existantes. Les travaux de Surmann et al. (Surmann et al., 2001) et de Pujas et Aldon (Pujas et Aldon, 1997) ont servi de point de départ pour la conception de notre capteur 3D combinant télémétrie et photométrie. Nous avons préféré l'approche de Surmann, qui consiste à faire pivoter un télémètre laser, à celle de Thrun (Thrun et al., 2000), qui consiste à utiliser le déplacement de la plate-forme pour obtenir la troisième dimension. Ainsi, nous pouvons acquérir des données tridimensionnelles sans effectuer de déplacement de la plate-forme, ce qui fournit une bonne connaissance de l'environnement avant de devoir s'y déplacer. Cela est très utile pour le développement de stratégies

d'explorations et de navigation<sup>7</sup>.

Après analyse des différents types de représentations déjà développés pour les cartes 3D, nous avons choisi de représenter les surfaces d'objets positionnées dans l'espace. Ce type de représentation se rapproche des travaux effectués par Surmann et al. (Surmann et al., 2003) et Thrun et al. (Thrun et al., 2000). En plus d'offrir une représentation compacte et de permettre de caractériser les surfaces observées en fonction de leur forme géométrique et de leur couleur, cette représentation est un premier pas vers une analyse sémantique et topologique de l'environnement. En effet, les travaux d'analyse sémantique de Nüchter et al. (Nüchter et al., 2003c) pourraient alors être directement appliqués à notre représentation. L'analyse des surfaces pourrait faire ressortir la structure topologique de l'environnement observé. De plus, les développements futurs pourraient permettre de détecter et de fusionner les surfaces appartenant à un même objet. La représentation choisie n'empêche pas l'utilisation d'autres techniques, comme les maillages hiérarchiques (Hoppe, 1996), pour la représentation visuelle des surfaces.

Pour détecter les surfaces à l'intérieur des images, nous avons recherché les travaux existant dans les domaines de la segmentation d'images télémétriques, d'images photométriques et de combinaison de ces deux modalités. Devant la grande variété de techniques de segmentation télémétrique seule et photométrique seule, nous avons décidé de développer une méthode de segmentation qui utilise chaque modalité séparément et qui combine ensuite les résultats obtenus plutôt que d'utiliser les deux modalités en même temps. L'aspect modulaire de notre technique de segmentation permet de remplacer facilement le module de segmentation télémétrique ou celui de segmentation photométrique par un autre module utilisant une implantation différente. Cela nous permet d'adapter rapidement notre tech-

---

<sup>7</sup>Ces stratégies ne font pas l'objet du projet traité dans ce mémoire.

nique de segmentation à de nouveaux développements effectués dans les domaines de segmentation télémétrique ou photométrique. Notre implantation utilise l'estimateur RANSAC (Fischler et Bolles, 1981) pour la segmentation télémétrique et la ligne de partage des eaux (Digabel et Lantuejoul, 1978) pour la segmentation photométrique.

Finalement, pour la localisation de la plate-forme mobile et la fusion des vues multiples, nous avons décidé d'adapter la technique ICP (Besl et McKay, 1992) pour permettre la localisation à partir des surfaces extraites lors de la segmentation plutôt qu'à partir des données brutes du capteur. Cela nous permet de tirer profit de la segmentation déjà effectuée pour obtenir un algorithme beaucoup plus rapide que l'algorithme ICP classique. Cette technique nous permet de corriger l'estimation de la position de la plate-forme fournie par l'odométrie et ainsi augmenter la précision de la carte 3D générée.

## CHAPITRE 2

### CONCEPTION D'UN CAPTEUR

Dans le but de réaliser de la cartographie 3D, il est nécessaire d'avoir un capteur permettant d'acquérir des informations 3D sur l'environnement à explorer. Les données recueillies sont de nature télémétrique et photométrique. La télémétrie permet de connaître avec précision la taille, le volume, l'orientation et la position des objets observés. De plus, les mesures télémétriques sont insensibles à la luminosité ambiante. La télémétrie est toutefois incapable de détecter les transitions entre les objets si celles-ci ne sont pas accompagnées d'un changement de relief. De plus, elle ne fournit aucun indice sur la nature des objets observés et le champ de vision est limité en profondeur. À l'opposé, la photométrie permet de détecter les transitions entre les objets même si celles-ci ne sont pas accompagnées d'un changement de relief, dans la mesure où cette transition est accompagnée d'un changement de couleur ou de luminosité. De plus, les couleurs obtenues fournissent des indices sur la nature des objets observés et le champs de vision n'est pas limité en profondeur. Toutefois, la photométrie peut difficilement détecter la taille, le volume, l'orientation et la position des objets observés avec autant de précision que la télémétrie. De plus, les données recueillies sont fortement influencées par la luminosité ambiante.

Nous voulons un capteur qui fournit des données télémétriques et photométriques de manière enregistrée, c'est-à-dire que les deux modalités sensorielles doivent être en correspondance. Les données télémétriques sont nécessaires pour les tâches de navigation, car elles permettent de situer les obstacles dans l'espace. Les données photométriques peuvent aider à déterminer la nature de ces obstacles et à identifier les objets. De plus, certains objets, comme une affiche collée sur un mur, peuvent

être visibles par la photométrie tout en étant invisibles par la télémétrie. La combinaison de ces deux modalités sensorielles permet ainsi de tirer profit des avantages de chaque modalité et de combler les manques liés à l'usage d'une seule modalité sensorielle.

Tout d'abord, nous allons décrire les différents types de capteurs commerciaux existant actuellement sur le marché. Ensuite, nous présenterons un nouveau capteur que nous avons conçu. Nous détaillerons son principe de fonctionnement, son incorporation à l'architecture logicielle de notre plate-forme mobile et nous ferons part de certains problèmes rencontrés lors de la conception. Finalement, nous présenterons un simulateur pour le capteur proposé.

## 2.1 Revue des capteurs commerciaux

Avant de concevoir notre capteur, nous avons vérifié s'il n'existait pas déjà sur le marché un capteur pouvant être utilisé pour la cartographie 3D par une plate-forme mobile. Alors que les données photométriques sont toujours obtenues au moyen d'une caméra, plusieurs technologies différentes sont utilisées pour l'acquisition des données télémétriques :

- Temps de vol par mesure directe ;
- Temps de vol par mesure de phase ;
- Triangulation avec un motif lumineux ;
- Stéréoscopie.

Nous résumons ci-dessous ces différentes technologies en mettant l'accent sur leur intérêt pour la cartographie par une plate-forme mobile.

### 2.1.1 Technologie temps de vol par mesure directe

La technologie *temps de vol par mesure directe* consiste à mesurer le temps requis pour qu'un rayon lumineux parcoure l'aller-retour entre le capteur et l'obstacle. La vitesse de la lumière étant connue, la distance peut être calculée à partir de ce temps. La source lumineuse utilisée est généralement un laser à pulsations. Avec une vitesse d'acquisition de 10 000 à 25 000 points par seconde, cette technique est relativement lente. Cela est dû au fait que le laser permet de mesurer la distance d'un seul point à la fois et qu'il est nécessaire d'éteindre le laser entre deux points. Plusieurs compagnies offrent en option une caméra numérique fournissant des informations photométriques enregistrées. Le laser et la caméra demeurent fixes et un miroir rotatif permet d'acquérir des données avec un grand champ de vision pouvant atteindre plus de 360°x290°. La précision des capteurs utilisant cette technique varie de 5 à 10 mm. La portée minimale des capteurs varie généralement entre 2 et 4 m et la portée maximale atteint généralement plus de 100 m. Les capteurs commerciaux de ce type ont un poids voisin de 15 kg et entrent généralement à l'intérieur d'une enceinte cubique de 30 cm de côté.

Voici quelques compagnies qui offrent ce type de produit :

- 3rdTech <http://www.3rdTech.com>
- Leica Geosystems <http://www.leica-geosystems.com>
- Riegl <http://www.riegl.com>
- Measurement Devices Ltd. <http://www.mdl.co.uk>
- Mensi <http://www.mensi.com>
- Optech Inc. <http://www.optech.ca>

Ce type de capteur est généralement utilisé pour l'acquisition de scènes intérieures ou extérieures. Les caractéristiques sont quasi parfaites pour la cartographie 3D. Toutefois, les distances minimales de 2 à 4 m causent de graves problèmes pour

la navigation et l'évitement d'obstacles. En milieu intérieur, il arrive souvent que le champ de vision soit limité par des obstacles situés à moins de 4 m, de sorte que la distance maximale de 100 m de ces capteurs est rarement utile<sup>1</sup>. De plus, ces capteurs 3D coûtent plusieurs dizaines de milliers de dollars, ce qui rend cette technologie financièrement peu attrayante.

Notons que les télémètres laser 2D — appareils dont le laser tourne sur un seul axe et dont les mesures prises sont coplanaires — qui utilisent ce type de technologie sont probablement les capteurs les plus utilisés en robotique mobile pour la cartographie. Ces capteurs 2D sont toutefois beaucoup moins coûteux que leurs cousins 3D.

### 2.1.2 Technologie temps de vol par mesure de phase

La technologie *temps de vol par mesure de phase* ressemble à la technologie temps de vol par mesure directe. L'idée derrière cette technologie est de moduler en amplitude le signal lumineux émis et de mesurer le temps de vol au moyen de la différence de phase entre le signal émis et celui reçu. Ainsi, le laser peut demeurer allumé en tout temps plutôt que par impulsions, ce qui permet une vitesse d'acquisition plus élevée que la méthode de mesure directe. La vitesse d'acquisition des capteurs utilisant cette technologie varie de 100 000 à 500 000 points par seconde selon le modèle utilisé. Leur portée maximale est de l'ordre de 25 m. Certains capteurs offrent un mode d'opération avec une portée minimale de 0,1 m. Les capteurs que nous avons étudiés ne fournissent pas de données photométriques registrées aux données télémétriques. Les autres caractéristiques de ces capteurs ressemblent à celles utilisant le temps de vol par mesure directe.

---

<sup>1</sup>Les autres technologies de capteurs que nous présenterons ont une portée maximale plus faible, mais permettent d'obtenir une portée minimale également plus faible.

Nous avons recensé deux compagnies qui offrent ce type de produit :

- Leica Geosystems <http://www.leica-geosystems.com>
- iQvolution <http://www.faro.com>

Ce type de capteur est idéal pour la reconstruction de milieux intérieurs. Toutefois, le coût trop élevé et l'absence d'informations photométriques rendent ce capteur inintéressant pour notre application.

Une équipe de chercheurs suisses (Oggier et al., 2004) a développé un nouveau type de caméra qui utilise la technologie du temps de vol par mesure de phase pour fournir des images 3D. Leur capteur est équipé d'un flash infrarouge qui envoie un plan lumineux vers l'avant du capteur. La lumière rebondit sur les obstacles et le plan revient, déformé, vers un capteur CMOS/CCD qui mesure la différence de phase pour chaque pixel du capteur. Ce nouveau type de caméra est très prometteur. Les prototypes sont rapides et coûtent environ un dixième du prix des autres capteurs 3D utilisant le temps de vol par mesure directe ou par mesure de phase. Toutefois, ces capteurs sont encore au stade de prototype et ont une résolution, un champ de vision et une portée limités.

### **2.1.3 Technologie par triangulation avec motif lumineux**

Il existe plusieurs technologies qui utilisent toutes un principe semblable. Un motif lumineux est projeté sur un objet et est observé par une caméra. La position du motif dans l'image de la caméra et la connaissance de la position relative entre l'émetteur du motif et la caméra permettent d'obtenir par triangulation la position des obstacles ayant réfléchi le motif lumineux. La figure 2.1 présente un exemple de cette technique. Le motif peut être un seul point, une ligne, un grillage, etc.

Les capteurs commerciaux qui utilisent cette technologie offrent souvent une pré-

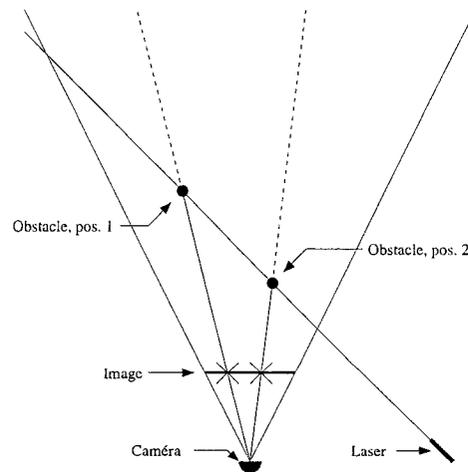


Figure 2.1 Triangulation avec un motif lumineux

cision supérieure à 0,02 mm, mais une portée limitée à 1 m. Le problème de cette technologie est que la précision est proportionnelle à la distance entre le laser et la caméra et est inversement proportionnelle à la distance de l'objet mesuré. Ainsi, pour obtenir les précisions indiquées à la portée de 1 m, les capteurs commerciaux ont une taille de près de 2x2x2 m. Cette technologie offre la possibilité d'une registration télémétrique et photométrique parfaite, car la même caméra sert à l'acquisition des deux modalités sensorielles. La vitesse d'acquisition peut atteindre les 1 300 000 points par seconde. Ces appareils sont utilisés principalement pour la numérisation de petits objets situés près du capteur prenant la mesure. Cette technologie n'est pas adaptée à la cartographie 3D.

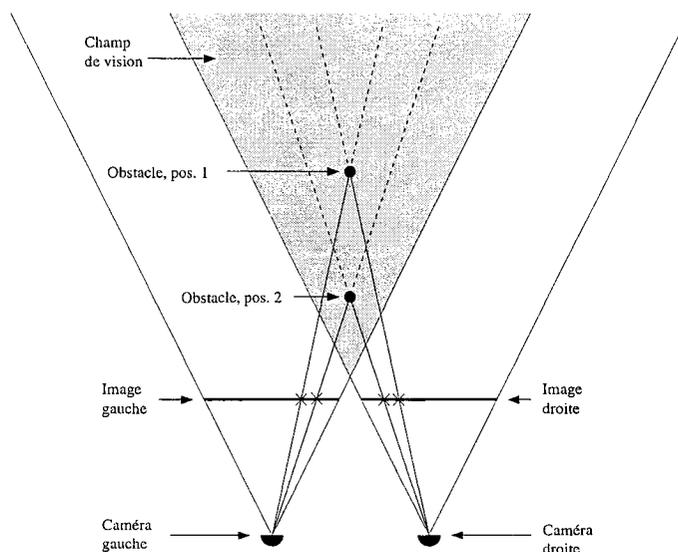


Figure 2.2 Stéréoscopie

Il existe de nombreuses compagnies qui offrent ce type de produit. En voici quelques unes :

- Breuckmann <http://www.breuckmann.com>
- Cyberware <http://www.cyberware.com>
- Laser Design Inc. <http://www.laserdesign.com>
- Minolta <http://www.minolta3d.com>
- Shape Grabber <http://www.shapegrabber.com>

#### 2.1.4 Technologie stéréoscopie

En utilisant deux caméras qui regardent la scène de deux points de vue différents, il est possible de déterminer la profondeur des objets visualisés en effectuant une triangulation à partir des points provenant des deux images. La figure 2.2 illustre ce principe.

Les caractéristiques (prix, précision, vitesse d'acquisition, etc.) dépendent généralement de la qualité des caméras utilisées. Comme pour la triangulation avec motifs lumineux, la précision est élevée à courte distance, mais devient très faible à grande distance. Par exemple, nous avons calculé que, pour obtenir une résolution de 1 cm à une distance de 5 m du capteur, la distance séparant les deux caméras doit être supérieure à 3,1 m. Si nous limitons cette distance à 0,3 m, alors la résolution sera d'environ 10,5 cm à 5 m du capteur et d'environ 27,3 cm à une distance de 8 m. Ces calculs prennent en considération des caméras possédant une distance focale de 807 pixel, ce qui correspond à un angle d'ouverture de  $42^\circ$  et une résolution de  $640 \times 480^2$ . En plus du problème de précision, cette technologie montre une difficulté à déterminer les distances pour des objets de couleur unie. En effet, pour déterminer la profondeur d'un point, il est nécessaire de le détecter sans ambiguïté dans les deux images, ce qui devient impossible si la couleur est unie.

Pour ces raisons, les caméras stéréoscopiques ne sont pas utilisées comme capteurs télémétrique de précision. Par conséquent, nous n'avons pas effectué de recherches approfondies des compagnies offrant ce type de capteur.

### 2.1.5 Conclusion de la revue des capteurs commerciaux

Les capteurs commerciaux de type temps de vol par mesure directe ou par mesure de phase sont les capteurs idéaux pour l'acquisition des données télémétriques nécessaires à la cartographie 3D. Malheureusement, ces capteurs sont très coûteux. Les technologies moins coûteuses, comme la triangulation avec motif lumineux et

---

<sup>2</sup>La formule  $z = \frac{Bf}{d}$  permet de déterminer la profondeur  $z$  d'un point en fonction de la disparité  $d$ , c'est-à-dire la différence de position de la projection du point dans chaque image.  $B$  est la ligne de base, c'est-à-dire la distance entre les centres optiques des deux caméras et  $f$  est la distance focale des caméras utilisées. Nous prenons également  $z + \Delta z = \frac{Bf}{d - \Delta d}$  et en combinant les deux formules, nous trouvons  $\Delta z = \frac{z^2 \Delta d}{Bf - z \Delta d}$ . Puis nous remplaçons  $B = 0,3$  m,  $f = 807$  pixels,  $\Delta d = 1$  pixel et  $z = 5$  m ou 8 m.

les caméras stéréoscopiques ne permettent pas d'obtenir une précision télémétrique acceptable à moyenne et à grande distance.

Nous en venons à la conclusion qu'il n'existe actuellement pas sur le marché de capteur disponible et abordable pour l'acquisition des données télémétriques nécessaires à la cartographie 3D par une plate-forme mobile. Ceci explique pourquoi les équipes de recherche en robotique mobile construisent toutes leur propre capteur d'acquisition 3D. Tel que montré au chapitre 1, les deux principaux designs de capteurs utilisés en robotique mobile pour la reconstruction de scènes 3D sont celui de Surmann et al. (Surmann et al., 2001; Surmann et al., 2003), qui consiste à faire tourner un capteur 2D sur lui-même, et celui de Hähnel et al. (Hähnel et al., 2003), qui consiste à utiliser un capteur 2D fixe et à déplacer la plate-forme mobile pour acquérir la troisième dimension. Une caméra peut être ajoutée à ces deux techniques pour réaliser l'acquisition des données photométriques.

## 2.2 Fonctionnement du capteur proposé

La conclusion de la section précédente nous oblige à concevoir notre propre capteur. Le capteur que nous proposons s'apparente à celui présenté par Surmann. Nous avons préféré cette approche à celle de Thrun parce qu'elle permet l'acquisition de données 3D sans nécessiter un déplacement de la plate-forme mobile. La position des obstacles est alors connue à l'avance et de meilleures stratégies de planification de trajectoires pourront être développées.

Notre capteur est basé sur le télémètre laser 2D LMS200 de la compagnie SICK. Il s'agit d'un capteur très populaire en robotique mobile en raison de sa vitesse d'acquisition acceptable de 13 575 points par seconde et de sa précision élevée. Nous ajoutons à ce capteur une caméra afin d'obtenir les informations photométriques

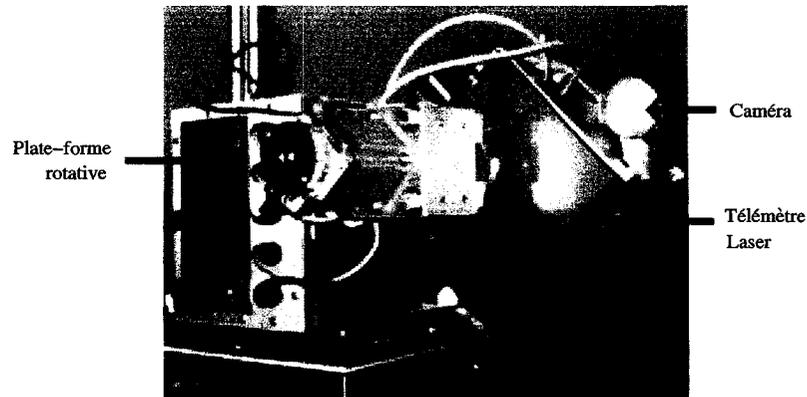


Figure 2.3 Vue d'ensemble du capteur

correspondant aux points observés par le télémètre laser. Ces deux capteurs sont situés sur une plate-forme qui effectue une rotation de haut en bas pour l'obtention de la troisième dimension. Le système complet est illustré à la figure 2.3. Sur cette illustration, le télémètre est incliné à un angle d'environ  $-45^\circ$  par rapport à l'horizontale. Le fonctionnement détaillé des diverses composantes et leur intégration sont décrits aux sections suivantes.

## 2.2.1 Télémètre laser

### 2.2.1.1 Principe de fonctionnement général

Le télémètre laser utilise la technologie « temps de vol par mesure directe ». Un miroir situé à l'intérieur du télémètre tourne à une vitesse d'au moins 75 tours par seconde et permet de modifier la direction pointée par le faisceau laser. Le champ de vision est limité par l'ouverture du boîtier à  $180^\circ$ . La figure 2.4 illustre une acquisition complète du télémètre laser prise avec un angle d'ouverture de  $180^\circ$  et une résolution angulaire de  $1,0^\circ$  : cette acquisition est donc constituée de 181

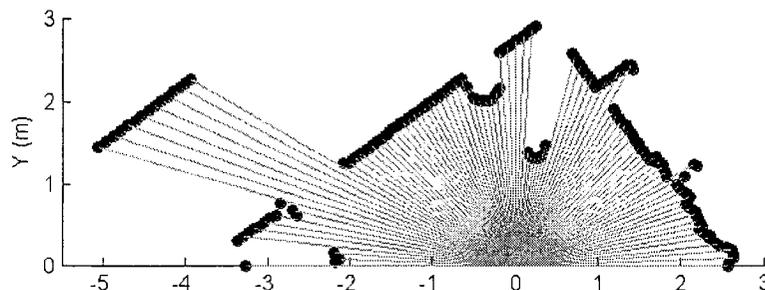


Figure 2.4 Coupe 2D parallèle au plancher

points. Nous appellerons ce type d'acquisition une « acquisition plane », car tous les points obtenus sont coplanaires. Les points noirs sont les données obtenues par le télémètre alors que les traits gris représentent les faisceaux du laser.

### 2.2.1.2 Repère lié au télémètre laser

Nous attribuons au télémètre laser le repère illustré à la figure 2.5. Les données nous sont fournies par le télémètre en coordonnées polaires conformément à l'équation 2.1. Pour les étapes suivantes, nous travaillons avec les coordonnées cartésiennes obtenues au moyen de l'équation 2.2. Dans ces équations,  $r_i$  est la distance mesurée par le capteur et  $\varphi_i$ , est l'angle pour lequel la distance  $r_i$  a été prise. Cet angle est obtenu au moyen de la formule 2.3 dans laquelle  $\varphi_{min}$  est l'angle de départ de l'acquisition,  $\Delta\varphi$  est le pas angulaire utilisé par le télémètre et  $n_{tl}$  est le nombre de points mesurés.

$$p_{i,tl\_polaire} = (r_i, \varphi_i) \quad (2.1)$$

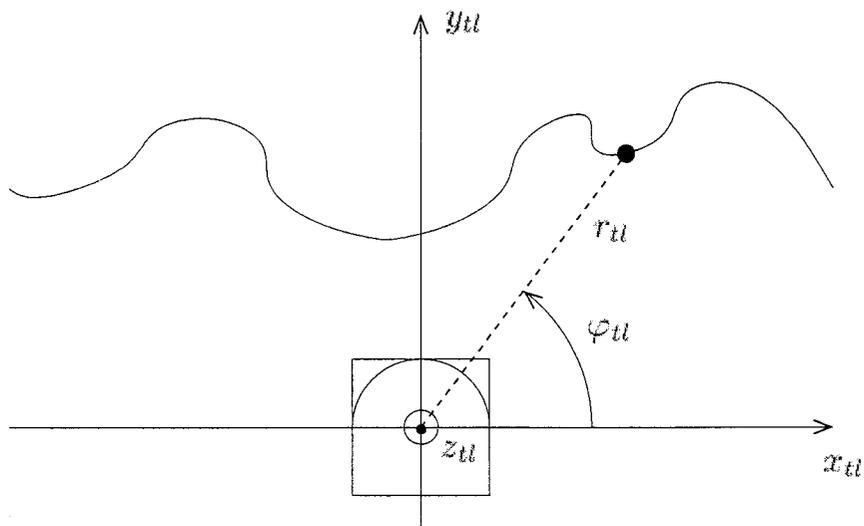


Figure 2.5 Repère lié au télémètre laser

$$p_{i,tl} = (r_i \cos \varphi_i, r_i \sin \varphi_i, 0) \quad (2.2)$$

$$\varphi_i = \varphi_{min} + i\Delta\varphi \quad i \in [0, n_{tl} - 1] \quad (2.3)$$

Le télémètre LMS200 possède plusieurs modes d'opération selon l'angle d'ouverture et la résolution angulaire désirée. En utilisant une carte d'acquisition haute vitesse RS422 spécialement modifiée, la vitesse de transmission des données entre le télémètre et la plate-forme est de 500 kbps. Le tableau 2.1 indique la vitesse d'acquisition (en nombre d'acquisitions planes par seconde) pouvant être obtenue pour chaque mode disponible ainsi que le nombre de points contenus dans chaque acquisition. Le mode de fonctionnement  $180^\circ@0,25^\circ$  n'existe pas en raison de la taille de mémoire limitée que possède le télémètre laser.

Angle d'ouverture	Résolution $\Delta\varphi$	Vitesse d'acquisition	Nombre de points $n_{tl}$
100°	0,25°	18,75 acq/s	401
100°	0,5°	37,5 acq/s	201
100°	1,0°	75 acq/s	101
180°	0,25°	Mode inexistant	
180°	0,5°	37,5 acq/s	361
180°	1,0°	75 acq/s	181

Tableau 2.1 Modes d'opération du télémètre laser

### 2.2.1.3 Fonctionnement détaillé

Il est important pour notre application de connaître quelques informations sur le fonctionnement interne du télémètre laser. La résolution angulaire utilisée à l'interne est toujours de 1,0°, peu importe le mode d'acquisition utilisé. Lorsque le télémètre laser est employé en mode 0,5°, deux acquisitions avec une résolution de 1,0° et décalées de 0,5° sont en réalité effectuées. Nous appellerons celles-ci des « acquisitions planes internes », car elles sont transparentes à l'utilisateur. Pour une utilisation en mode 0,25°, quatre acquisitions planes internes décalées chacune de 0,25° sont réalisées. La figure 2.6 illustre dans quel ordre les points d'une acquisition plane à une résolution de 0,25° sont réellement pris par le capteur. Ceci explique pourquoi l'acquisition plane en mode 0,25° est quatre fois plus lente que l'acquisition en mode 1,0°. L'interface de communication fait abstraction de ces détails d'implantation. Ainsi, lorsqu'une acquisition plane à une résolution de 0,25° est demandée, les quatre acquisitions planes internes sont effectuées puis les points sont transférés dans l'ordre normal d'utilisation, c'est-à-dire par ordre croissant d'angle. Puisque les angles sont connus par l'ordinateur connecté au télémètre, seules les distances mesurées ( $r_i$ ) sont réellement envoyées. Les valeurs de  $\varphi_i$  sont obtenues par l'équation 2.3. Ce fonctionnement interne, généralement sans impor-

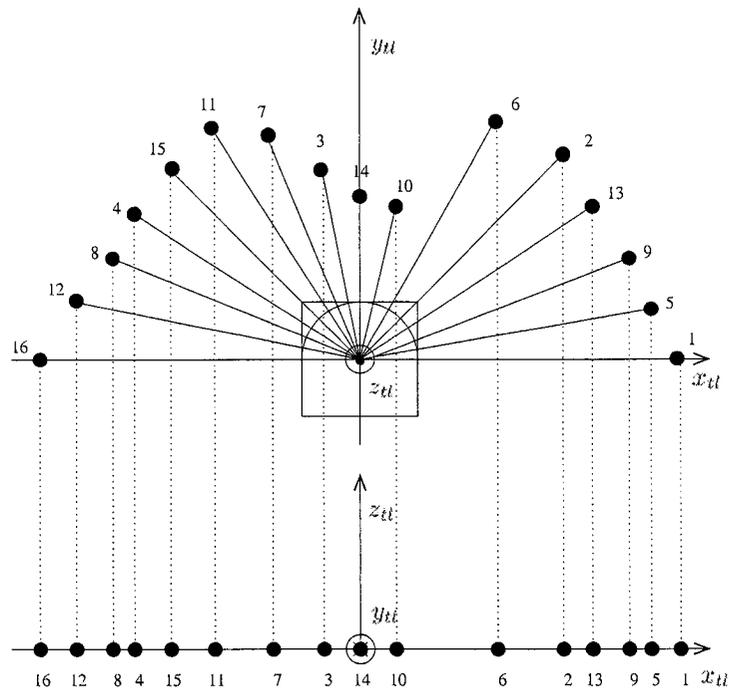


Figure 2.6 Ordre interne d'acquisition plane en mode 0,25°

tance, a un impact si nous faisons tourner le télémètre laser pendant qu'il effectue une acquisition, tel qu'il sera montré à la section 2.2.3.4.

### 2.2.2 Caméra

Le télémètre laser permet d'obtenir la position de certains points dans un demi-plan situé devant lui. Afin d'obtenir les informations photométriques reliées à ces points, nous ajoutons une caméra sur le dessus du télémètre laser.

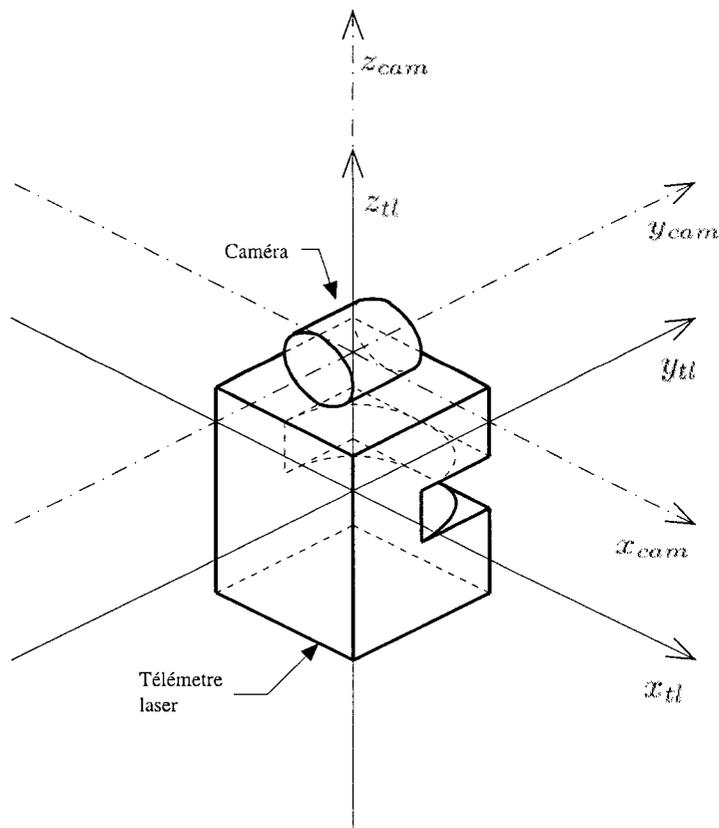


Figure 2.7 Repères liés au télémètre laser et à la caméra

### 2.2.2.1 Repère lié à la caméra

Le montage est assemblé de sorte que le centre optique de la caméra soit aux coordonnées  $c_{cam,tl} = (0, 0, c_{cam,\bar{k}})$  dans le repère du télémètre laser et que son axe optique soit parallèle à l'axe de visée du télémètre. Nous utilisons le repère lié à la caméra illustré à la figure 2.7. L'axe  $x_{cam}$  pointe vers la droite de l'image, l'axe  $y_{cam}$  correspond à l'axe optique et l'axe  $z_{cam}$  pointe vers le haut de l'image. L'origine de ce système est située au centre optique de la caméra.

### 2.2.2.2 Changement du repère lié au télémètre vers le repère lié à la caméra

Les points obtenus par le télémètre laser ont des coordonnées dans le repère lié à la caméra qui sont données par l'équation 2.4. Dans cette équation,  $T_{tl \rightarrow cam}$  et  $R_{tl \rightarrow cam}$  sont la translation et la rotation nécessaires pour passer du repère lié au télémètre laser vers le repère lié à la caméra. Les valeurs théoriques de ces transformations sont données aux équations 2.5 et 2.6.

$$p_{i,cam} = (p_{i,tl} - T_{tl \rightarrow cam}) R_{tl \rightarrow cam} \quad (2.4)$$

$$R_{tl \rightarrow cam} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

$$T_{tl \rightarrow cam} = \begin{bmatrix} 0 & 0 & c_{cam,\bar{k}} \end{bmatrix} \quad (2.6)$$

En pratique, le montage utilisé est légèrement imparfait : la caméra n'est ni parfaitement orientée, ni parfaitement centrée de sorte que les valeurs théoriques de  $T_{tl \rightarrow cam}$  et  $R_{tl \rightarrow cam}$  doivent être ajustées. Pour effectuer cette calibration, la méthode semi-automatique de Zhang et Pless (Zhang et Pless, 2004) peut être utilisée.

### 2.2.2.3 Projection dans le plan image

Lorsque nous avons les coordonnées des points dans le repère 3D lié à la caméra, nous pouvons les projeter dans le plan image de la caméra et ainsi obtenir les coor-

données des points dans l'image. Pour effectuer cette projection, nous utilisons le modèle de caméra à trou d'épingle. Selon ce modèle, les rayons lumineux passent tous par le centre optique de la caméra. Une portion rectangulaire des intersections de ces rayons avec le plan image forme l'image obtenue par la caméra. Les coordonnées des points dans l'image sont obtenues par la projection perspective de la formule 2.7.  $L_{img}$ ,  $H_{img}$  et  $f$  sont respectivement la largeur, la hauteur et la distance focale, exprimés en pixels. Le signe négatif du second terme de l'équation 2.7 a pour effet de placer l'origine de l'image dans son coin supérieur gauche.

$$p_{i,img} = \left( \frac{L_{img}}{2} + \frac{p_{i,cam_x}}{p_{i,cam_z}} f, \frac{H_{img}}{2} - \frac{p_{i,cam_y}}{p_{i,cam_z}} f \right) \quad (2.7)$$

Les paramètres intrinsèques de la caméra — distance focale, position du centre optique, distorsion de la lentille — peuvent être obtenus au moyen de la procédure de calibration proposée par Zhang (Zhang, 1999) qui est disponible dans Matlab. Les images utilisées par notre capteur proviennent d'une librairie créée par M. Vincent Zalzal, un ancien étudiant à la maîtrise au GRPR.

La figure 2.8 présente l'image de la scène qui a été prise par la caméra en même temps que l'acquisition du laser montrée à la figure 2.4 de la page 43. Les points  $p_{i,tl}$  dans le repère du télémètre laser sont transférés dans le repère de la caméra au moyen de l'équation 2.4 puis projetés dans l'image en utilisant l'équation 2.7 pour obtenir les points rouges de la figure 2.8. Les couleurs originales de l'image à ces endroits sont les couleurs attribuées aux points.

L'acquisition du télémètre laser de la figure 2.4 comportait 181 points alors que seulement 43 points sont visibles dans l'image de la caméra. Cela est dû au fait que l'angle d'ouverture de notre caméra n'est que de  $42^\circ$ . Pour les autres points acquis par le télémètre laser, aucune information photométrique n'est disponible. L'acqui-

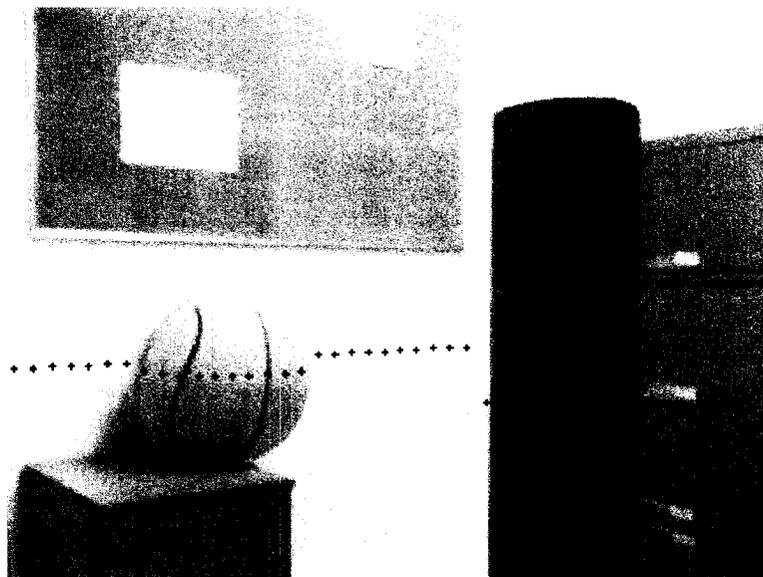


Figure 2.8 Image prise par la caméra pendant l'acquisition du télémètre laser de la figure 2.4

sition du télémètre laser avec l'ajout de l'information photométrique est présentée à la figure 2.9. Les points magenta sont les points pour lesquels aucune information photométrique n'est disponible. Ces points ne sont pas pris en considération dans le reste du présent mémoire.

### 2.2.3 Plate-forme rotative

Nous avons montré à la section 2.2.1 comment acquérir une coupe 2D de la scène et, à la section 2.2.2, comment récupérer les informations photométriques de ces points. Notre objectif est d'obtenir un capteur 3D. Pour y arriver, nous faisons pivoter l'ensemble télémètre-laser-caméra autour de l'axe  $x_{tl}$ , ce qui permet de prendre plusieurs coupes non coplanaires de la scène.

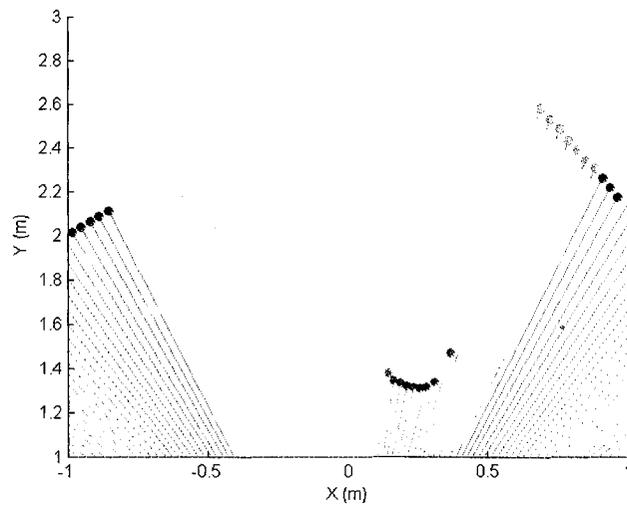


Figure 2.9 Acquisition télémétrique et photométrique 2D

### 2.2.3.1 Principe de fonctionnement

Le système télémètre-laser-caméra est assez volumineux et pèse environ 5 kg. Faire pivoter cette masse avec précision n'est pas simple. Pour ce faire, nous avons fait fabriquer sur mesure une plate-forme rotative par la compagnie *Scientific Instrumentation Limited*. Le modèle commandé est une version légèrement modifiée d'une plate-forme rotative conçue pour *Recherche et Développement de la Défense du Canada*. Il s'agit d'une plate-forme robuste et mécaniquement très précise. Un moteur pas à pas permet de faire tourner le télémètre laser et la caméra à des vitesses variant de 0 à 90 degrés par seconde. Un encodeur de rotation permet d'obtenir l'angle d'élévation avec une précision de  $0,1^\circ$ . Un système embarqué se charge de la communication avec le télémètre laser de sorte que l'ordinateur reçoit toutes les acquisitions planes avec les angles d'élévation auxquels ont été prises les acquisitions.

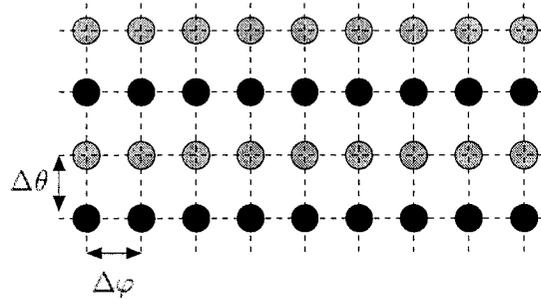


Figure 2.10 Motif de points pour le mode ligne par ligne

### 2.2.3.2 Repère lié au capteur 3D

Le repère que nous utilisons lié au capteur 3D est tel qu'il est égal au repère du télémètre laser lorsque celui-ci est parallèle au sol. Nous appelons  $\theta$  l'angle d'élévation du télémètre laser par rapport au plan  $xy$  du repère lié au capteur 3D. Cet angle est positif lorsque le télémètre est incliné vers le haut et négatif lorsqu'il est incliné vers le bas.

### 2.2.3.3 Mode d'acquisition ligne par ligne

Deux modes d'acquisition peuvent être utilisés : l'acquisition ligne par ligne et l'acquisition continue. Le premier consiste à déplacer le télémètre laser à l'angle désiré, à effectuer une acquisition à cet angle et à déplacer de nouveau le télémètre pour la prochaine acquisition plane. Dans ce mode, le télémètre laser n'est pas en mouvement pendant qu'il effectue une acquisition. Dans ce cas, les points observés par le capteur forment un motif ressemblant à celui de la figure 2.10. Les coordonnées sphériques de ces points sont données par l'équation 2.8 où  $i$  représente l'indice d'un point d'une acquisition plane et  $j$  représente le numéro de l'acquisition. Les valeurs de  $\varphi_i$  et  $\theta_j$  sont obtenues au moyen des équations 2.3 et 2.9.

$$p_{ij,cap\_spherique} = (r_{ij}, \theta_j, \varphi_i) \quad (2.8)$$

$$\theta_j = \theta_{max} - j\Delta\theta \quad j \in \left[0, \frac{n_{cap}}{n_{tl}} - 1\right] \quad (2.9)$$

La variable  $n_{cap}$  représente le nombre total de points d'une image 3D obtenue par le capteur. En pratique, les valeurs de  $\theta_j$  utilisées sont celles fournies par le capteur de rotation de la plate-forme rotative plutôt que celles fournies par l'équation 2.9.

#### 2.2.3.4 Mode d'acquisition continu

Le second mode d'opération est le mode continu. Dans ce mode, la plate-forme rotative effectue la rotation de haut en bas sans s'arrêter. Les acquisitions planes sont effectuées pendant que le capteur est en déplacement. Le motif de points ainsi obtenu est semblable à celui de la figure 2.11. Les points de même couleur appartiennent à la même acquisition plane. Les coordonnées sphériques des points obtenus sont données par l'équation 2.10. L'équation fournissant la valeur de  $\theta$  doit être redéfinie tel que montré à l'équation 2.11. Dans cette équation,  $n_{int}$  représente le nombre d'acquisitions planes internes effectuées pour chaque acquisition plane.

$$p_{ij,cap\_spherique} = (r_{ij}, \theta_{ij}, \varphi_i) \quad (2.10)$$

$$\theta_{ij} = \theta_{max} - \left( j\Delta\theta + \frac{\Delta\theta}{n_{int}} (i \bmod n_{int}) + \frac{\Delta\theta}{360n_{int}} \left\lfloor \frac{i}{n_{int}} \right\rfloor + \frac{\varphi_{min}}{360} \right) \quad (2.11)$$

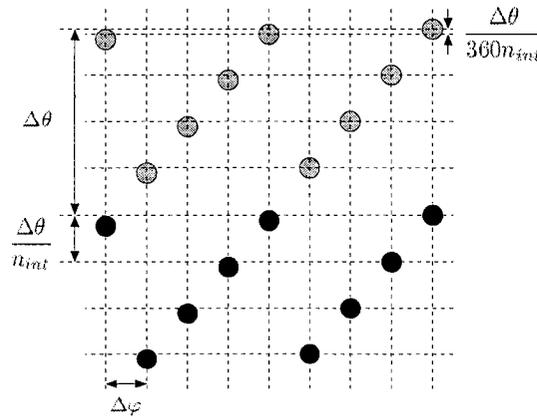


Figure 2.11 Motif de points pour le mode continu

La figure 2.11 illustre graphiquement les termes de cette équation. Le premier terme de la parenthèse de l'équation 2.11 est l'écart angulaire entre deux points de même angle  $\varphi$  de deux acquisitions planes successives. Le second terme est la distance angulaire parcourue entre le début de deux acquisitions planes internes successives. Le troisième terme est la distance angulaire parcourue entre deux points successifs d'une même acquisition plane interne. Ce terme peut être négligé en pratique. Le dernier terme est un décalage qui tient compte des points obtenus par le télémètre laser qui n'ont pas été observés par la caméra. En pratique, la valeur de  $\Delta\theta$  est évaluée pour chaque valeur de  $j$  en prenant les informations fournies par le capteur de rotation de la plate-forme rotative. Ce capteur de rotation nous fournit l'angle de début ( $j\Delta\theta$ ) et de fin  $((j+1)\Delta\theta)$  d'une acquisition plane et l'équation 2.11 est utilisée pour obtenir les angles de chaque point de l'acquisition par interpolation.

Pour obtenir les informations photométriques, il faut remplacer l'équation 2.2 de la page 44 par la nouvelle équation ci-dessous.

$$p_{i,tl} = (r_i \cos \varphi_i \sin \theta_i, r_i \sin \varphi_i \sin \theta_i, \cos \theta_i) \quad (2.12)$$

Dans cette équation,  $\theta_i$  est l'angle  $\theta_{ij}$  de l'équation 2.11 exprimé dans le repère du télémètre laser en au début de l'acquisition  $j^3$ . Cette valeur est donné par l'équation 2.13<sup>4</sup>.

$$\theta_i = -\frac{\Delta\theta}{n_{int}} (i \bmod n_{int}) \quad (2.13)$$

Que nous utilisons le mode ligne par ligne ou le mode continu, les positions des points en coordonnées cartésiennes dans le repère du capteur 3D sont toujours fournies par l'équation 2.14 où  $\theta_{ij}$  est fourni par l'équation 2.9 ou 2.11 selon le mode de fonctionnement utilisé.

$$p_{i,cap} = (r_{ij} \cos \varphi_i \sin \theta_{ij}, r_{ij} \sin \varphi_i \sin \theta_{ij}, \cos \theta_{ij}) \quad (2.14)$$

#### 2.2.4 Procédure d'acquisition complète

Nous avons opté pour l'utilisation de la plate-forme rotative en mode continu. Cela a l'inconvénient de rendre plus complexes les calculs et d'augmenter l'incertitude des résultats obtenus, car les délais internes du télémètre laser ne sont pas connus avec précision. D'un autre côté, arrêter et repartir la plate-forme rotative entre deux acquisitions planes induirait des vibrations qui causeraient une erreur beaucoup plus grande que l'imprécision des délais internes. La seule manière d'éviter

---

<sup>3</sup>Nous supposons que c'est à ce moment qu'est prise l'image photométrique correspondant à cette acquisition.

<sup>4</sup>Nous négligeons le terme  $\frac{\Delta\theta}{360n_{int}} \left\lfloor \frac{i}{n_{int}} \right\rfloor$ .

ces vibrations serait d'attendre entre l'arrêt du moteur et l'acquisition plane pour que les vibrations s'estompent. Cela augmenterait considérablement le temps d'acquisition.

En pratique, nous utilisons les valeurs  $\Delta\theta = \Delta\varphi = 0,25^\circ$ . Cette valeur est choisie pour avoir la résolution maximale du télémètre laser et une résolution uniforme dans les deux axes de rotation. Le mode  $0,25^\circ$  implique  $n_{int} = 4$ . En raison de l'angle d'ouverture de la caméra de  $42^\circ$ , nous avons  $\varphi_{min} = 69^\circ$ . L'angle d'ouverture de la caméra fixe le paramètre  $n_{tl} = 169$  ( $42^\circ @ 0,25^\circ$ ). Nous avons décidé arbitrairement de poser  $\theta_{max} = 45^\circ$  et d'avoir un angle ouverture vertical de  $90^\circ$ . Nos images 3D ont donc une résolution de  $169 \times 361$ , soit un total de  $n_{cap} = 61\,009$  points. La vitesse de rotation de la plate-forme rotative est fixée en tenant compte de la fréquence d'acquisition du télémètre laser qui est de 18,75 acquisitions planes par seconde. Pour obtenir  $\Delta\theta = 0,25^\circ$ , nous devons fixer la vitesse de rotation verticale à environ  $4,7^\circ$  par seconde.

La procédure complète d'acquisition d'une image 3D est indiquée ci-dessous :

1. Élever le capteur à la position angulaire  $\theta_{max} = 45^\circ$ .
2. Démarrer les acquisitions planes et la rotation du capteur vers le bas.
3. Pour chaque acquisition plane, calculer les coordonnées des points dans le repère 3D lié à la caméra au moyen des équations 2.12 et 2.4 puis les coordonnées des pixels dans l'image de la caméra au moyen de l'équation 2.7.
4. Calculer la position des points de chaque acquisition plane dans le repère 3D lié au capteur au moyen de l'équation 2.14.
5. Arrêter lorsque l'angle d'élévation minimal du capteur est atteint.

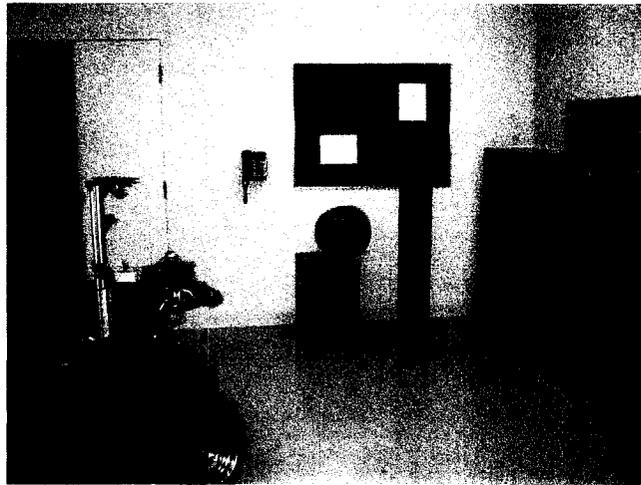
### 2.3 Exemple de résultat

L'image 3D résultant de ce processus est une matrice à deux dimensions dont chaque élément contient les coordonnées cartésiennes et la couleur d'un point dans l'espace 3D. La figure 2.12 présente une acquisition 3D vue sous deux angles différents en (b) et en (c); la figure 2.12 (a) est une photo prise avec une caméra numérique montrant à la fois le capteur et la scène observée.

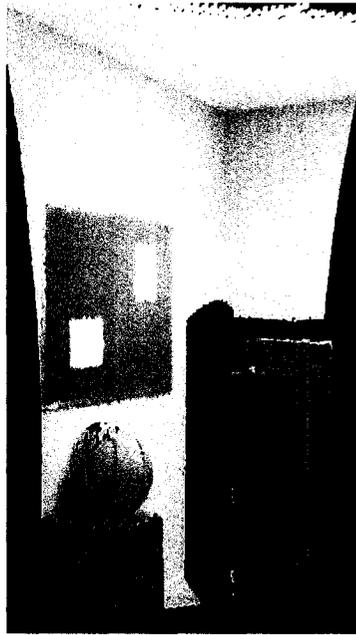
Les données obtenues au moyen du capteur peuvent également être représentées en dissociant les données télémétriques et les données photométriques. Les données photométriques seules donnent l'image de la figure 2.13 (a). Il s'agit d'une image à l'apparence habituelle. Les données télémétriques seules sont présentées à la figure 2.13 (b). Sur cette image, le niveau de gris est proportionnel à la profondeur des points observés. Ce type d'affichage met en évidence le fait que le résultat est une matrice 2D de points.

### 2.4 Intégration du capteur à la plate-forme mobile

La conception du capteur 3D a nécessité un grand nombre d'étapes : étude des capteurs existants, conception préliminaire, essais et preuve de fonctionnement du concept, conception finale, achat de la plate-forme rotative, fabrication de supports en métal pour fixer la caméra au capteur et le capteur à la plate-forme mobile, installation et stabilisation du capteur sur la plate-forme mobile et intégration de l'ensemble à l'architecture logicielle de la plate-forme mobile. De tous ces aspects, le dernier a été le plus laborieux en raison du nombre de couches logicielles devant être modifiées.



(a) Vue d'ensemble



(b) Image 3D vue du côté gauche



(c) Image 3D vue du côté droit

Figure 2.12 Exemple d'une numérisation 3D

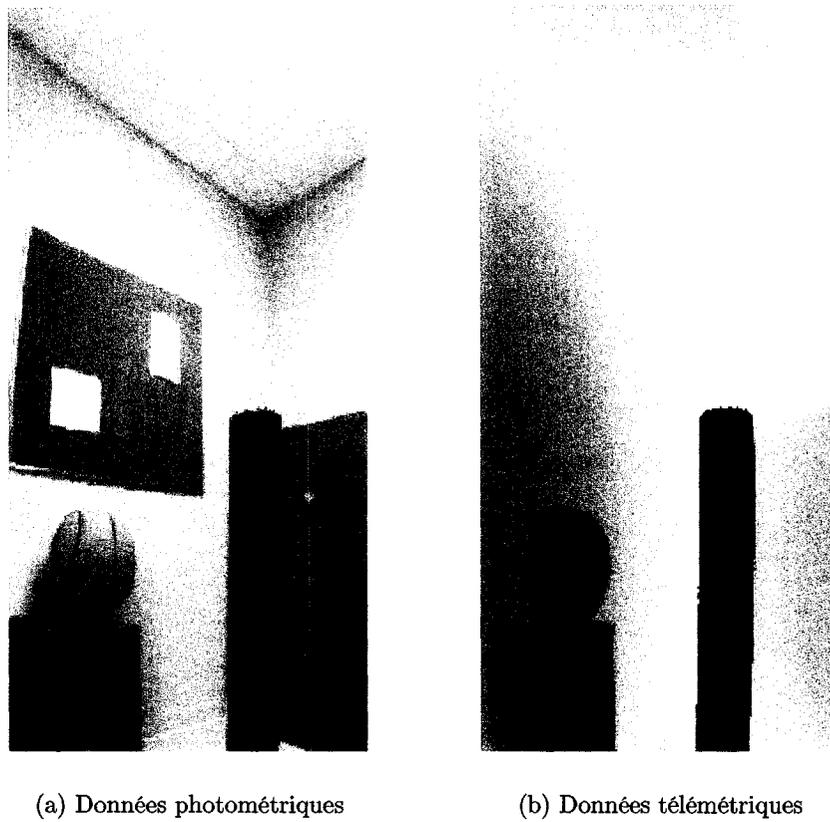


Figure 2.13 Dissociation des données télémétriques et photométriques de l'exemple de la figure 2.12

Notre plate-forme mobile est équipée du système d'exploitation Linux Fedora Core 3 et de l'architecture logicielle Player, version 1.6.2, disponible sous licence GPL. Une couche logicielle supplémentaire, nommée Acropolis, a été développée au GRPR principalement par M. Vincent Zalzal (Zalzal, 2005). L'annexe II décrit plus en détail l'architecture en place, la liste des modules existants avant la création du capteur 3D et les modules développés pour permettre l'intégration du capteur à cette architecture logicielle.

## 2.5 Problèmes rencontrés

Deux problèmes principaux ont été observés pendant la conception du capteur : les occlusions et la présence de points fantômes.

### 2.5.1 Occlusions

Puisque le point de vue de la caméra n'est pas le même que celui du télémètre laser, il peut arriver des situations où certains points sont vus par le télémètre laser, mais ne sont pas visibles par la caméra. Lors d'une occlusion, les couleurs obtenues par la caméra suite à la projection des points du télémètre laser ne correspondent pas aux vraies couleurs de ces points. La figure 2.14 illustre une telle situation. Étant donné que la caméra est située au-dessus du télémètre laser, les obstacles pouvant créer une occlusion sont toujours situés au-dessus du plan de coupe du télémètre. Puisque le mouvement de rotation est effectué de haut en bas, l'obstacle aura été détecté lors d'une acquisition plane précédente (sauf dans le cas de la première acquisition plane). Ainsi, il est possible de détecter les cas où il y a eu occlusion et où les données photométriques sont erronées.

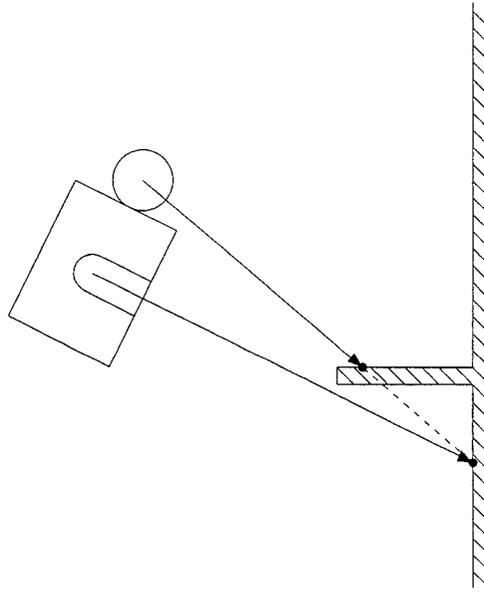


Figure 2.14 Exemple d'une situation d'occlusion

Pour détecter un cas d'occlusion, nous pouvons former un maillage 3D avec les points obtenus lors des acquisitions précédentes. En raison du processus d'acquisition utilisé, un maillage de quadrilatères peut être obtenu sans effort en prenant les points  $p_{i,j}$ ,  $p_{i+1,j}$ ,  $p_{i,j+1}$  et  $p_{i+1,j+1}$  comme sommets de chaque quadrilatère. À chaque nouvelle acquisition plane, nous vérifions si les droites reliant ces points au centre optique de la caméra traversent le maillage 3D obtenu en combinant les acquisitions planes précédentes. S'il y a intersection, ceci signifie qu'il y a une occlusion.

L'intersection d'une droite avec un quadrilatère est une opération assez intense en calculs, surtout si elle doit être faite  $\mathcal{O}(n_{cap}^2)$  fois. En pratique, nous avons simplifié la détection des occlusions en ramenant le problème 3D à un problème 2D. Pour ce faire, nous considérons uniquement les points qui ont la même valeur d'angle  $\varphi_i$  (donc le même indice  $i$ ). Nous obtenons alors le cas 2D illustré à la figure 2.15 et

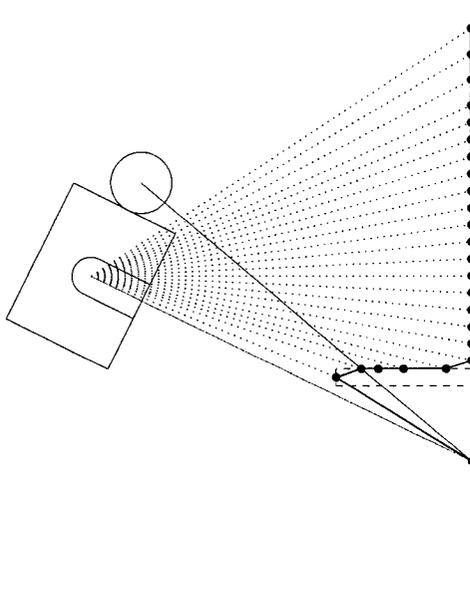


Figure 2.15 Méthode de détection des occlusions

le problème en est un de détection d'une intersection entre deux segments. Cette simplification n'est pas une méthode exacte, mais les essais effectués ont montré qu'elle fonctionne bien. La figure 2.16 (a) présente une numérisation effectuée sans détection des occlusions. Nous voyons que certains points du mur n'ont pas reçu la bonne couleur. La figure 2.16 (b) présente la même numérisation avec les occlusions bien détectées et affichées en cyan.

### 2.5.2 Points fantômes

Le second problème rencontré est l'existence de points fantômes situés aux frontières entre les objets. Certains points fantômes sont visibles sur la figure 2.12 (c) de la page 58. La figure 2.17 est un agrandissement où les points fantômes ont été encadrés en rouge. L'expérience acquise au laboratoire GRPR démontre que ce

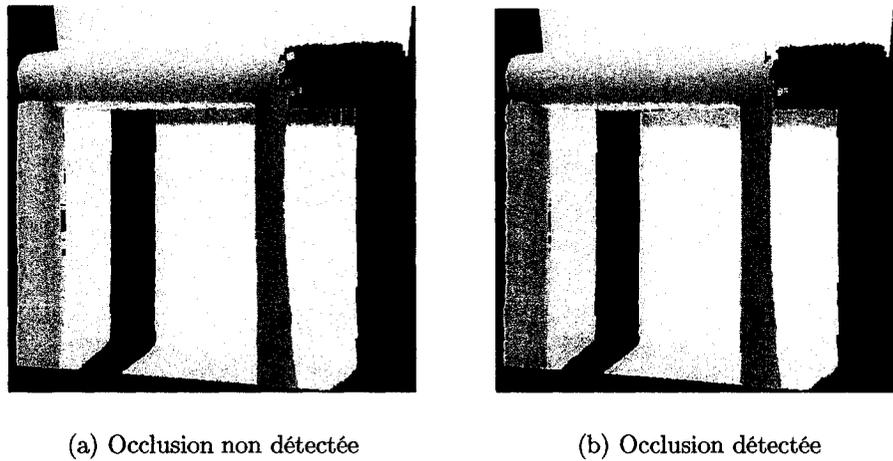


Figure 2.16 Exemple réel de détection des occlusions

problème provient du télémètre laser. Le problème est lié au fait que le faisceau laser n'est pas idéal et forme en réalité un cône très fin. Aux frontières des objets, une portion du faisceau frappe l'objet en avant-plan et l'autre portion frappe l'arrière-plan. La valeur résultante est un point situé dans le vide entre les deux obstacles. La position spatiale de ce point est alors erronée et le calcul de projection qui fournit la couleur reliée au point donne par conséquent une couleur également erronée. Ce problème est donc lié à la technologie employée par le télémètre laser et est hors de notre contrôle. La technique de segmentation présentée au chapitre suivant devra prendre en considération l'existence de ces points.

## 2.6 Simulateur

Nous avons présenté aux sections ci-dessus un nouveau capteur combinant télémétrie et photométrie. Nous avons également conçu un simulateur pour ce capteur. Ce simulateur permet de générer des scènes précises pouvant servir de base de comparaison pour valider la technique de segmentation présentée au chapitre 3 et

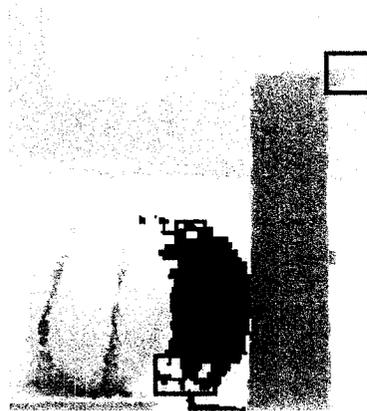


Figure 2.17 Points fantômes

l’algorithme de fusion de vues multiples présenté au chapitre 4. De plus, l’usage d’un simulateur permet de modifier facilement les paramètres de la scène et de vérifier leurs effets sur les algorithmes à valider.

Le simulateur que nous avons conçu contient trois types de surfaces simples : quadrilatères, cylindres et sphères. En combinant plusieurs quadrilatères, il est possible de créer des surfaces planes plus complexes.

Nous simulons une version simplifiée du capteur réel :

- le centre de la caméra est situé à la même position que le centre du laser, de sorte qu’il n’y a pas d’effet d’occlusion ;
- le capteur simulé opère en mode ligne par ligne, de sorte que tous les points situés sur une même ligne ont le même angle d’élévation  $\theta$ .

Pour obtenir les données télémétriques, nous simulons une demi-droite partant du centre du capteur et ayant l’orientation donnée par les angles  $\varphi_i$  et  $\theta_j$ . L’angle  $\theta_j$  varie par incrément de  $0,25^\circ$  dans l’intervalle  $[-45^\circ, 45^\circ]$ . Pour chaque valeur de  $\theta_j$ , l’angle  $\varphi_i$  varie avec le même incrément dans l’intervalle  $[-21^\circ, 21^\circ]$ . Ces valeurs

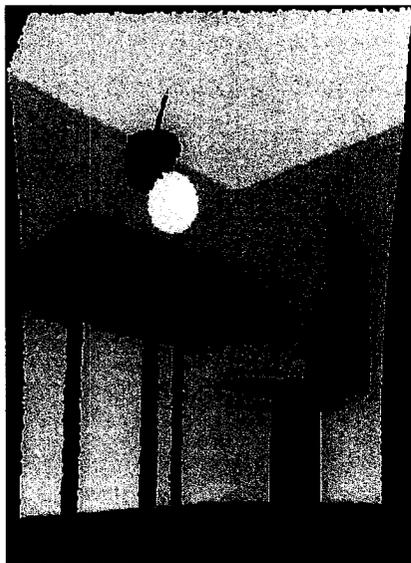


Figure 2.18 Image obtenue au moyen du simulateur

sont choisies pour être fidèles au mode d'acquisition utilisé par le vrai capteur. Pour chaque demi-droite, nous calculons les points d'intersection de cette demi-droite avec toutes les surfaces traversées par la demi-droite. Le point obtenu le plus proche du centre du capteur est retenu. Le processus est répété pour chaque demi-droite.

Pour obtenir les données photométriques, nous affichons la scène en utilisant OpenGL. La caméra est centrée au point d'où partent les demi-droites et possède un angle d'ouverture suffisant pour que tous les points calculés soient visibles dans l'image produite. La fonction d'OpenGL *gluProject* est utilisée pour connaître les pixels d'affichage à l'écran de chaque point d'intersection. Nous récupérons ainsi les informations photométriques de chaque point.

Nous attribuons un identificateur à chaque surface. Celui-ci permet de connaître à quelle surface appartient chaque point de l'image. Cette information sert au

chapitre 5 à produire la segmentation de référence utilisée pour évaluer les performances de l'algorithme de segmentation.

Un exemple d'image obtenu au moyen de notre simulateur est présenté à la figure 2.18.

## 2.7 Conclusion de la conception du capteur

Nous avons étudié les différents types de capteurs commerciaux disponibles sur le marché pour en venir à la conclusion qu'aucun d'entre eux ne convient pour la cartographie d'un environnement intérieur par une plate-forme mobile.

Nous avons donc conçu notre propre capteur. Un télémètre laser 2D est utilisé pour l'acquisition des données télémétriques et une caméra est ajoutée pour l'acquisition des données photométriques. L'ensemble est monté sur une plate-forme rotative qui permet d'obtenir des données dans un espace à trois dimensions. Le capteur est conçu de manière à ce que les données télémétriques et photométriques soient enregistrées.

Nous avons développé un algorithme permettant de détecter les situations d'occlusions pouvant mener à l'association d'une fausse couleur aux données télémétriques. Ces points sont conservés, mais un drapeau indique que leur couleur est erronée.

Les images obtenues au moyen du capteur montrent une bonne registration des données télémétriques et photométriques. Nous avons toutefois noté l'existence de points fantômes dont la présence, due au télémètre laser, est hors de notre contrôle. Au chapitre suivant, ces données seront traitées pour en faire ressortir les surfaces des objets présents dans l'environnement.

## CHAPITRE 3

### SEGMENTATION

Le capteur décrit au chapitre 2 fournit une grande quantité de données très précises. Une seule acquisition comporte plus de 60 000 points. Toutefois, ces données ne sont pas directement utiles à la plate-forme mobile dans leur état brut. Plutôt que de conserver tous les points individuels, il est largement préférable d'effectuer un traitement initial servant à regrouper entre eux les points ayant des caractéristiques communes ; c'est ce qu'on appelle la segmentation.

L'objectif de notre segmentation est de détecter adéquatement les surfaces des objets qui composent la scène afin d'en obtenir une description plus compacte et plus utile. Nous voulons classer les points selon deux critères :

1. Dans un premier temps, au niveau spatial, nous voulons détecter les surfaces des objets et les classifier en fonction de modèles mathématiques prédéterminés. Nous appelons « surface » un ensemble de points spatialement reliés les uns aux autres et dont la relation peut être mathématiquement paramétrée. Plus précisément, nous voulons détecter les portions de plans, et d'autres types de surfaces comme des cylindres et des sphères.
2. Dans un second temps, ces surfaces doivent être subdivisées en sous-ensembles, que nous appelons « éléments surfaciques » en fonction des caractéristiques photométriques.

La représentation ainsi créée facilite la navigation en simplifiant les tâches d'évitement d'obstacles. En effet, calculer les collisions potentielles avec quelques surfaces

est beaucoup plus performant que le faire avec plusieurs milliers de points. Cette représentation peut également être utilisée efficacement pour la localisation de la plate-forme mobile. Un algorithme de localisation utilisant cette structure sera présenté au chapitre 4. Finalement, la recherche d'objets simples à partir d'une description des surfaces est également facilitée. La segmentation en surfaces géométriques est une première étape dans l'analyse sémantique de l'environnement dans lequel évolue le robot. Les travaux d'analyse sémantique de Nüchter et al. (Nüchter et al., 2003c) dans lequel sont identifiés le plancher, le plafond, les murs et les portes, pourraient être directement appliqués à notre représentation. Des travaux ultérieurs pourront être effectués pour ajouter un niveau hiérarchique supérieur où les faces d'un même objet seront fusionnées pour obtenir une représentation en objets. Cela permettra ensuite de reconnaître et de classer les objets plus complexes en utilisant des techniques semblables à celle de Lyer et al. (Lyer et al., 2004). Des niveaux hiérarchiques supplémentaires pourront ensuite être ajoutés pour permettre d'exploiter les travaux de Sappa et Garcia (Sappa et Garcia, 2004) qui ont montré comment une structure hiérarchique peut être utilisée pour accélérer les tâches de recherche de collisions.

Une bonne segmentation devra :

1. Détecter adéquatement les éléments surfaciques de taille supérieure à 100 points<sup>1</sup> présents dans l'acquisition 3D. Une segmentation est adéquate si elle n'est ni excessive — trop d'éléments surfaciques différents détectés pour un même élément surfacique réel — ni insuffisante — plusieurs éléments surfaciques réels détectés comme étant un seul élément surfacique.
2. Respecter la géométrie spatiale. Les distances spatiales entre les points et les surfaces associées doivent être les plus faibles possible.

---

<sup>1</sup>Nous retrouvons cette valeur dans les travaux de Silva et al. (Silva et al., 2002) et de Gotardo et al. (Gotardo et al., 2003). Cela correspond à 0,16% de la taille des images traitées.

3. Respecter la photométrie. Les différences de couleur entre les couleurs des points et celles des éléments surfaciques auxquels ils sont associés doivent être les plus faibles possible.
4. La segmentation doit être robuste au bruit télémétrique et photométrique. Le nombre de fausses détections d'éléments surfaciques doit être faible.
5. La segmentation photométrique doit être la moins dépendante possible d'un éclairage non-uniforme de la scène.

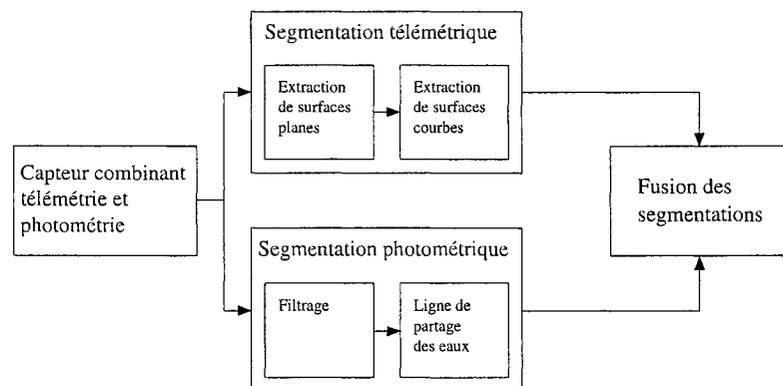


Figure 3.1 Vue d'ensemble du système de segmentation

La figure 3.1 montre une vue d'ensemble du système de segmentation qui a été mis au point. Les données provenant du capteur sont segmentées selon deux méthodes différentes, l'une basée sur la télémétrie et l'autre basée sur la photométrie. Les deux modélisations de l'environnement ainsi obtenues sont ensuite fusionnées afin d'obtenir un modèle plus complet. Séparer ainsi les segmentations télémétrique et photométrique offre plusieurs avantages par rapport à une approche combinée :

1. Chacune des segmentations travaille sur une modalité sensorielle différente et les modèles mathématiques ne sont pas les mêmes pour chaque modalité. Les techniques de segmentation utilisant en même temps la télémétrie et

la photométrie combinent dans une même équation des données ayant des sens physiques différents. Pour dimensionner correctement ces données, des facteurs multiplicatifs difficilement justifiables doivent être utilisés. Pour cette raison, ces techniques n'ont pas, à notre avis, un sens physique aussi important que le traitement séparé de chaque modalité.

2. Beaucoup de recherches ont été effectuées individuellement dans chacun des deux domaines de segmentation. En appliquant une approche de segmentation séparée, nous pouvons tirer profit des techniques qui ont déjà fait leurs preuves.
3. Si une des modalités sensorielles venait à faire défaut, une partie seulement de la segmentation serait affectée. Celle-ci pourrait alors être écartée et le résultat demeurerait cohérent. Nous augmentons donc ainsi la robustesse du système.

Ce chapitre est divisé comme suit. La section 3.1 présente le module de segmentation télémétrique tandis que la section 3.2 présente le module de segmentation photométrique. Le module de fusion de ces deux segmentations est présenté à la section 3.3 et quelques notes sur les régions créées forment la section 3.4.

### **3.1 Segmentation par la télémétrie**

Une des segmentations effectuées est basée sur la télémétrie. Cette segmentation est réalisée par deux modules tels qu'illustrés à la figure 3.1. Le premier module réalise une segmentation de la scène en surfaces planes alors que le second module utilise ces surfaces planes pour réaliser la détection de surfaces courbes. La séparation de ces deux modules permet de tirer profit des nombreuses recherches déjà réalisées dans ces domaines et d'éviter certains problèmes liés à l'extraction des surfaces

courbes directement à partir d'un nuage de points (temps de calcul élevé et sous-segmentation).

### 3.1.1 Extraction de surfaces planes

Les données du capteur arrivent à un premier module dont l'objectif est de créer une approximation de la scène au moyen d'un ensemble de portions de plans. Parmi les différentes techniques que nous avons étudiées pour réaliser cette tâche (voir section 1.3.1), nous avons décidé d'utiliser une technique de type « extraction itérative de surfaces planes », principal aspect des techniques proposées par Lee et al. (Lee et al., 1998) et Gotardo et al. (Gotardo et al., 2003). Nous avons choisi ce type de techniques pour trois grandes raisons : la robustesse, la précision et la simplicité.

**Robustesse.** Ces techniques fonctionnent très bien, même si de nombreux points (près de 50 %) sont erronés.

**Précision.** Les comparaisons effectuées par les auteurs avec d'autres techniques montrent de bons résultats.

**Simplicité.** La technique n'a que trois paramètres et ceux-ci sont facilement ajustables et physiquement justifiables (les autres techniques en ont généralement entre 5 et 12). Il n'est pas nécessaire de recalibrer le système entre chaque utilisation.

#### 3.1.1.1 Description de la technique retenue

La technique retenue fonctionne par extractions successives de surfaces planes. Au cours d'une itération, le plan le plus important d'un nuage de points est détecté. Puis la région de points 4-connectée la plus grande formée par les points de ce

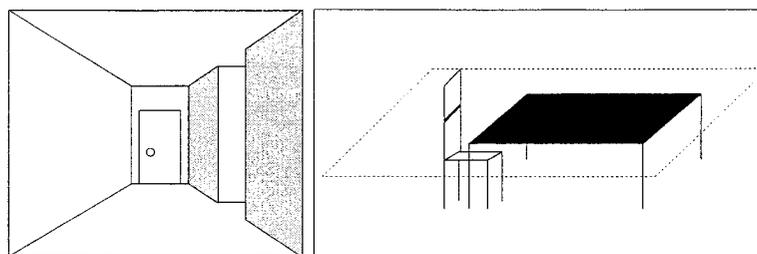
plan est conservée comme face résultant de cette itération. Les points restants sont divisés en nuages de points 4-connectés et la procédure est appliquée itérativement sur chacune de ces zones.

La détection du plan est réalisée au moyen de l'estimateur RANSAC (RANdom SAMpling Consensus (Fischler et Bolles, 1981)). L'algorithme prend en entrée un nuage de points et retourne le plus grand plan trouvé en procédant comme suit : trois points sont choisis au hasard dans le nuage de points et sont utilisés pour définir un plan. Ensuite, l'ensemble des points est parcouru afin de compter le nombre de points situés à une distance inférieure à un paramètre  $d_{max}$  du plan. Enfin, la procédure est répétée  $N_{iter}$  fois et seul le plan comptant le plus de points est conservé.

Il est à noter que cette technique est en fait une approximation d'une technique plus précise, mais énormément plus lente. En effet, nous pourrions essayer toutes les équations de plans obtenues en prenant tous les triplets de points possibles dans l'image. Cependant, un tel algorithme serait de complexité  $\mathcal{O}(n_{cap}^4)$  et prendrait, au mieux, plusieurs minutes à s'exécuter. Choisir seulement  $N_{iter}$  triplets parmi les  $n_{cap}^3$  existants, donne une bonne approximation de complexité  $\mathcal{O}(N_{iter}n_{cap})$ .

Prendre les points complètement aléatoirement permet de réduire l'effet du bruit de mesure sur les performances de l'algorithme. En effet, une petite erreur sur la position d'un point aura un grand effet sur l'orientation du plan résultant si la distance entre les points est du même ordre de grandeur que l'erreur. À l'opposé, l'effet sera imperceptible si les points sont très éloignés les uns des autres. Le choix aléatoire des points diminue considérablement les chances de tomber sur une situation particulière néfaste.

Pour chaque plan, nous extrayons la région la plus grande contenue dans ce plan.



(a) Deux murs sur un même plan

(b) Portion de chaise incluse dans le plan de la table

Figure 3.2 Nécessité de détecter les zones 4-connectées

La figure 3.2 illustre deux exemples de la nécessité de cette approche. Dans le cas 3.2a), deux sections de mur appartiennent au même plan mais doivent constituer deux surfaces distinctes. Le cas 3.2b) illustre une situation où des points, ceux de la chaise, appartiennent à plus d'un plan et seraient ainsi mal classés lors de la détection de la table si nous ne conservions pas que la région la plus grande.

Nous utilisons pour cela une connaissance sur le mode d'acquisition du capteur utilisé. En effet, une image 3D fournie par notre capteur peut être vue comme une grille 2D de pixels auxquels des informations télémétriques sont associées. Nous pouvons alors définir une notion de voisinage. Nous disons que deux points sont 4-connectés s'ils sont situés sur deux cases adjacentes de la grille (deux cases qui ont une arête en commun). Une région de points est dite 4-connectée si pour chaque paire de points de cette région, il existe au moins un chemin 4-connecté pour se rendre d'un point à l'autre. La surface principale du plan est ainsi constituée uniquement des points de ce plan qui forment la plus grande région 4-connectée dans l'image. Un numéro d'identification de la surface est attribué à ces points et ceux-ci sont retirés de l'image.

À la suite du retrait des points d'une surface, il arrive fréquemment que les points restants forment plusieurs nuages de points 4-connectés disjoints. La procédure d'extraction d'une surface est ensuite appliquée de manière récursive sur chacun de ces nuages jusqu'à ce que le nombre de points des nuages restants soit inférieur à un paramètre  $N_{min}$  prédéterminé. Le paramètre  $N_{min}$  permet de réduire les apparitions de fausses surfaces dues au bruit.

### 3.1.1.2 Détermination des paramètres

La méthode d'extraction de surfaces planes décrite ci-dessus implique l'usage de trois paramètres auxquels il faut assigner une valeur :

$d_{max}$  Distance maximale tolérée d'un point à la surface.

$N_{min}$  Nombre minimal de points pour avoir une surface.

$N_{iter}$  Nombre d'essais aléatoires pour détecter un plan.

La valeur de  $d_{max}$  doit être fixée égale (ou légèrement supérieure) à la précision en distance du capteur utilisé. Pour notre capteur, la valeur typique de  $d_{max}$  est de 2 cm (voir section 5.1.1).

La valeur de  $N_{min}$  devrait être fixée à un faible pourcentage de la taille de l'image originale. Pour un capteur donné, comme toutes les images sont de taille fixe, la valeur de  $N_{min}$  est également fixe. La valeur typique de  $N_{min}$  est de 100 points (0,16 %).

La valeur de  $N_{iter}$  peut être choisie statistiquement de sorte à avoir virtuellement près de 100 % des chances de trouver le bon plan. En supposant que le plan principal de l'image occupe une proportion supérieure ou égale à  $q_1$  des points de celle-ci, la valeur de  $N_{iter}$  à choisir pour détecter le plan principal avec une probabilité  $q_2$  est donnée par l'équation 3.1. Par exemple, pour avoir plus de 99 % de chance de

trouver un plan occupant seulement 20 % des points non assignés dans l'image, il faut avoir  $N_{iter} \geq 574$ .

$$N_{iter} = \frac{\log(1 - q_2)}{\log(1 - q_1^3)} \quad (3.1)$$

### 3.1.1.3 Modifications apportées à l'algorithme de base

La méthode expliquée ci-dessus utilise des idées similaires à celles utilisées par Gotardo et al. (Gotardo et al., 2003) et Lee et al. (Lee et al., 1998). Les expériences effectuées avec cette méthode ont permis d'y apporter une modification importante pour améliorer ses performances vis-à-vis des surfaces de petite taille. Au lieu d'effectuer l'opération de vérification de 4-connexion à la fin des  $N_{iter}$  essais aléatoires, nous la réalisons pour chacun des  $N_{iter}$  essais. Ainsi, nous détectons réellement la surface plane ayant le plus grand nombre de points plutôt que le plan le plus important. Il est en effet possible que le plan qui comporte le plus grand nombre de points ne soit pas celui qui possède la surface la plus grande. Cela est principalement vrai dans le cas où la scène est composée d'une multitude de petites surfaces. Le fait que l'algorithme fonctionne en une seule passe — lorsqu'une surface est retirée, elle l'est définitivement — implique qu'une erreur d'extraction de surface peut avoir des répercussions fâcheuses sur les surfaces extraites ultérieurement, d'où l'importance d'apporter cette modification. Notons que ce changement ralentit légèrement l'algorithme, mais ne modifie pas sa complexité algorithmique.

La figure 3.3 illustre un exemple 2D fictif où les petits segments sont mal détectés par l'algorithme de base. Les données originales sont présentées en a) et les premières régions détectées en b). La prochaine droite la plus importante est constituée soit des points bleus, soit des points verts de la partie c) et la région 4-connectée la

plus grande est composée d'un seul point comme celle encadrée en rouge. Si les régions possédant moins de deux points sont supprimées, alors le résultat est présenté en d). Nous observons que quatre segments sont alors perdus. Avec la modification apportée à l'algorithme, le résultat e) est obtenu et conserve correctement tous les points.

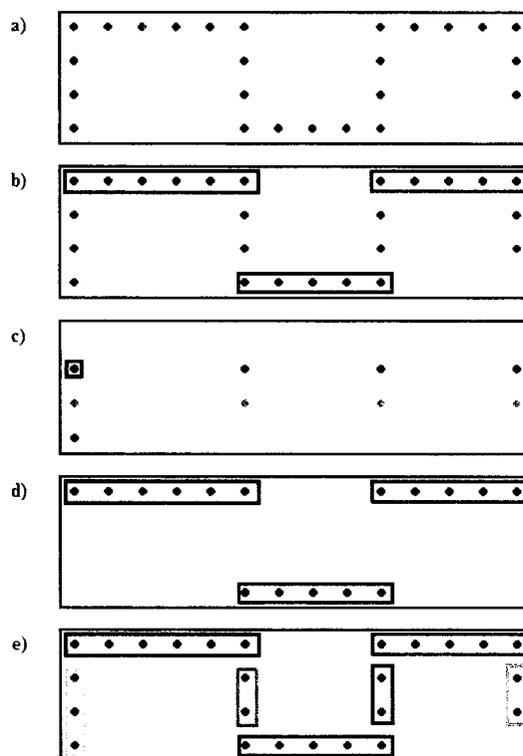


Figure 3.3 Exemple de l'utilité de vérifier les zones 4-connectées à chaque itération a) données brutes, b) premières régions bien détectées, c) problème observé avec l'algorithme de base, d) résultat de la segmentation obtenu avec l'algorithme de base, e) résultat obtenu suite à la modification apportée.

Une autre modification a été apportée à l'algorithme. Elle touche la manière de calculer les régions 4-connectées. Tel que mentionné à la section 2.5.2, le capteur que nous avons conçu possède des points fantômes. Pour remédier à ce problème,

un filtre est appliqué sur les données sources pour retirer ces points. Deux stratégies peuvent être utilisées :

1. Calculer numériquement le vecteur normal à la surface en chaque point et supprimer le point si la différence de vecteur normal avec les points voisins est supérieure à un seuil prédéterminé.
2. Supprimer le point si celui-ci possède un voisin plus éloigné qu'une certaine distance  $S_D$ .

Le concept de la première méthode paraît plus intéressant que celui de la seconde. En se basant sur les vecteurs normaux plutôt que sur les distances elles-mêmes, la méthode devrait être aussi robuste à courte distance qu'à grande distance. Toutefois, cette méthode est très sensible au bruit de mesure du laser. Lorsque les points sont très rapprochés les uns des autres, la différence entre les vecteurs normaux peut devenir très élevée même au milieu d'une surface numérisée. La figure 3.4, page 78, montre la répartition spatiale des vecteurs normaux obtenus en utilisant une fenêtre 5x5 pour le cas d'une surface plane. Au lieu d'obtenir une zone compacte autour de la valeur cible  $(0, 0, -1)$ , nous remarquons que les vecteurs normaux sont très étalés et que plusieurs d'entre-eux font même un angle de plus de  $90^\circ$  par rapport à la valeur théorique. Dans ces conditions, un filtrage au moyen des vecteurs normaux ne pourra pas donner les résultats escomptés.

Pour cette raison, nous avons choisi la seconde approche. Le seuil  $S_D$  peut être vu comme la résolution minimale acceptée. En effet, au-delà du seuil  $S_D$  nous pouvons considérer que les points n'offrent pas un niveau de détail suffisant pour déterminer s'ils appartiennent bel et bien à la même surface. Cette technique aura tendance à privilégier les points rapprochés et discréditer les points plus éloignés.

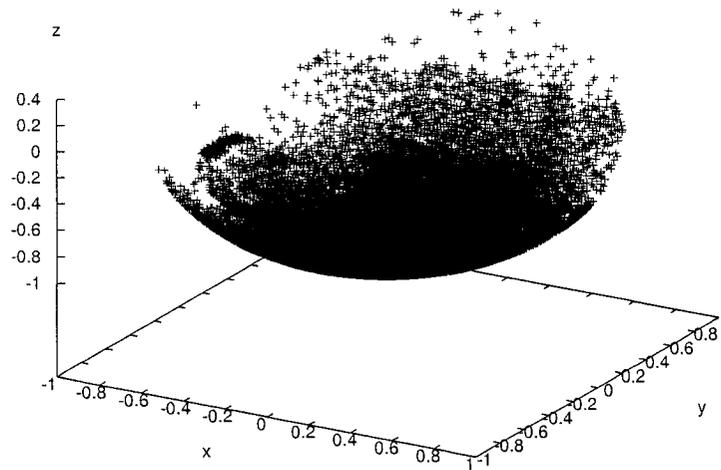


Figure 3.4 Répartition spatiale des vecteurs normaux obtenus à la suite de la numérisation d'un plan.

Les vecteurs normaux sont normalisés et partent tous du point  $(0, 0, 0)$ . Seuls les points d'arrivée sont affichés.

#### 3.1.1.4 Plans aux moindres carrés

Dans les sections précédentes, nous avons décrit les bases de l'algorithme utilisé pour l'extraction des surfaces planes. Une dernière opération est réalisée avant d'obtenir les résultats définitifs.

Plus la valeur choisie pour le paramètre  $d_{max}$  est élevée, plus les surfaces obtenues sont contaminées par des points appartenant à des surfaces adjacentes et situés près des intersections. Il est à noter que ces points sont peu nombreux en comparaison des points appartenant réellement à la surface. L'équation du plan peut alors être

améliorée en effectuant une régression linéaire par moindres carrés<sup>2</sup>. L'équation du plan minimisant les moindres carrés est obtenue en utilisant la technique expliquée par Szafran (Szafran, 2006). D'autres estimateurs pourraient également être utilisés comme le MSAC (Silva et al., 2002) ou le ALKS (Lee et al., 1998).

### 3.1.1.5 Exemple de résultat

La figure 3.5, page 80, présente un exemple type de segmentation en surfaces planes. L'image télémétrique originale vue par le capteur est présentée en (a) et le résultat de la segmentation est en (b). Toutes les principales surfaces ont été détectées convenablement. Une petite surface peut sembler étrange dans le coin supérieur droit. Celle-ci provient de la lumière néon située au plafond. La majorité des rayons du laser ont détecté le plastique protecteur du néon et ont ainsi fourni des points à la même hauteur que le reste du plafond. D'un point de vue télémétrique, ces points appartiennent donc au plafond. Toutefois, il existe un angle particulier pour lequel les rayons du laser passent au travers du plastique protecteur et détectent le fond du néon, créant ainsi la petite surface dans le coin supérieur droit. Il ne s'agit pas d'un artifice de segmentation, mais d'une surface bien réelle. Un phénomène semblable a été documenté dans un article de Früh et Zakhor (Früh et Zakhor, 2002).

Les surfaces courbes sont sursegmentées, c'est-à-dire qu'elles sont divisées en plusieurs surfaces au lieu d'en former une seule. Ce problème sera corrigé par le module décrit à la section 3.1.2. Des résultats supplémentaires sont fournis au chapitre 5.

---

<sup>2</sup>Plus la valeur de  $d_{max}$  est faible, plus l'approximation fournie par RANSAC se rapproche de celle obtenue par régression linéaire ; celle-ci devient alors peu utile.

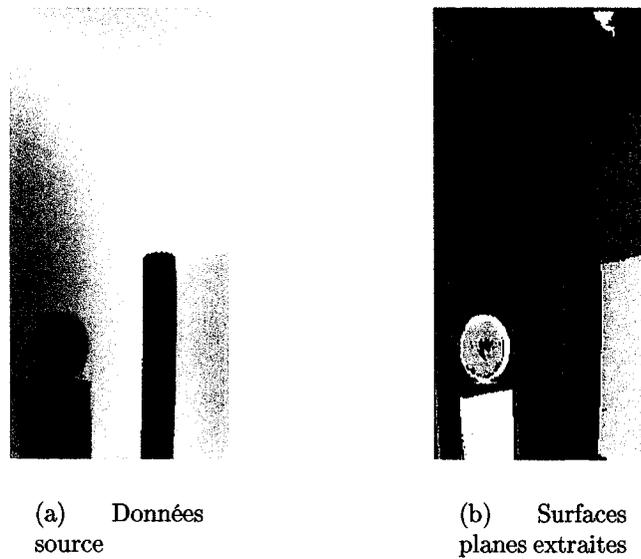


Figure 3.5 Résultat d'une extraction de surfaces planes

### 3.1.1.6 Résultat sous forme de graphe

Le résultat de la segmentation en surfaces planes est un graphe pour lequel les noeuds sont des surfaces et les arêtes sont les intersections entre les surfaces. Il y a une intersection entre deux surfaces si au moins un point d'une surface est 4-connecté à un point de l'autre surface.

Cette représentation topologique de la scène est très utile à plusieurs points de vue. Premièrement, le module de segmentation courbe présenté à la section suivante parcourt ce graphe afin de vérifier si deux noeuds adjacents ne peuvent pas être fusionnés en une surface de type plus complexe. Deuxièmement, il s'agit d'une information fort utile pour un module d'inférence logique qui voudrait déterminer la nature des objets dans la scène. Par exemple, le système de Nüchter et al. (Nüchter et al., 2003b; Nüchter et al., 2003c) permet de détecter planchers, plafonds, murs et portes en se basant sur les formes des surfaces et sur les relations entre elles.

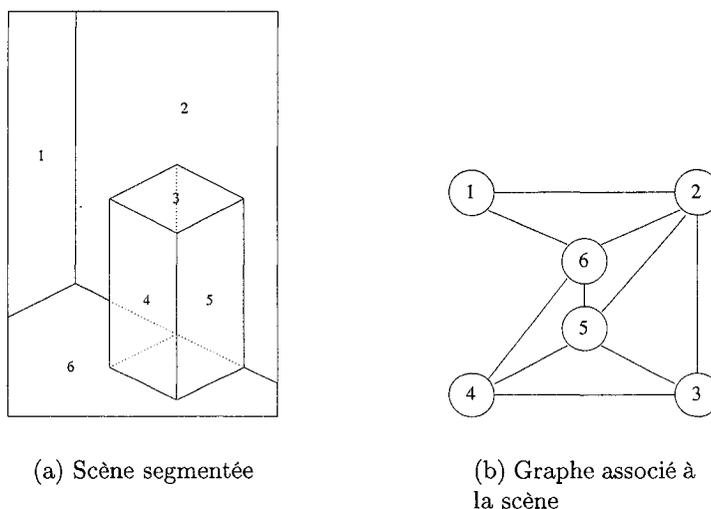


Figure 3.6 Exemple d'une scène et de son graphe.

La figure 3.6 illustre un exemple de scène et sa représentation sous forme de graphe.

### 3.1.2 Extraction de surfaces courbes

Le module décrit ci-dessus permet une segmentation de la scène en surfaces planes. En milieu structuré, les principales surfaces — murs, plafonds, planchers, portes, tables, étagères, etc. — sont planes et sont donc adéquatement segmentées par notre premier module de segmentation. Toutefois, les produits créés par l'homme ne sont pas tous plats et le modèle plan ne convient pas à toutes ces surfaces. C'est pourquoi nous avons développé le présent module d'extraction de surfaces courbes.

Si une surface courbe est numérisée et que l'approximation de la surface sous forme de plan n'est pas adéquate — avec une tolérance sur l'erreur de distance — la surface courbe sera divisée en plusieurs plans par le module précédent. Nous utilisons cette propriété pour chercher des surfaces courbes aux intersections entre deux surfaces sécantes. Cette technique offre plusieurs avantages par rapport à d'autres

techniques connues, notamment celles qui consistent à chercher des quadriques ou autres types de surfaces plus complexes directement à partir des données sources :

1. En limitant la recherche aux intersections, nous ne traitons qu'une portion restreinte des données à chaque fois, ce qui diminue le temps de calcul.
2. Nous privilégions le modèle le plus simple. Par exemple, si un plan et un cylindre peuvent tous deux être une approximation d'une surface, le modèle du plan sera choisi. Les modèles plus complexes sont alors choisis uniquement lorsque nécessaire. Notons, par exemple, qu'une surface plane est un cas particulier d'un cylindre (ou d'une sphère) dont le rayon tend vers l'infini. Puisque les équations de surfaces courbes offrent plus de degrés de liberté que l'équation d'un plan, il est fort probable, à cause des erreurs de mesures, qu'une surface pratiquement plane, comme un mur, soit mieux représentée par une surface courbe de très grand rayon. En privilégiant le modèle le plus simple, nous diminuons le risque de ce type de mauvaise classification.
3. La littérature scientifique fournit des exemples de sous-segmentations créées par l'extraction de surfaces courbes directement à partir des données sources. Un exemple typique (Lee et al., 1998) est l'intersection de deux plans inclinés, comme, par exemple un toit de maison, qui sont souvent remplacés à tort par un hyperboloïde. Ce type d'erreur est peu probable avec notre système, car les fusions de surfaces planes sont réalisées uniquement si elles réduisent l'erreur globale.
4. Notre algorithme ne se limite pas uniquement à certains types de surfaces. Notre méthode peut être adaptée à tout type de surface, simple ou complexe, pour lequel un modèle adéquat existe.

La figure 3.7 présente le fonctionnement global du module de segmentation courbe. L'entrée du module est une intersection de deux surfaces : deux surfaces planes, une

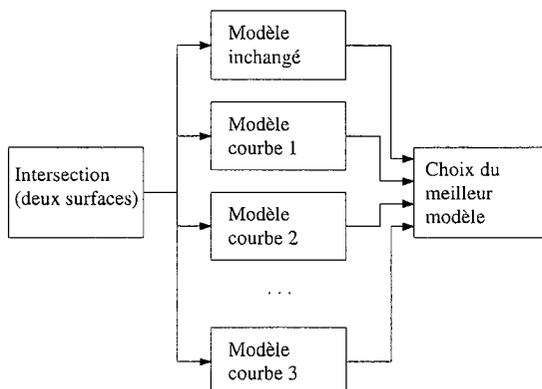


Figure 3.7 Procédé d'extraction des surfaces courbes

surface courbe et une plane ou deux surfaces courbes. Le module reçoit également en entrée une liste de modèles, c'est-à-dire une liste des types de surfaces à rechercher. Pour chaque modèle, les paramètres optimaux sont calculés et le modèle offrant l'erreur aux moindres carrés minimale est retenu.

Un modèle de surface est composé de trois éléments : une paramétrisation de la surface, un calcul de l'erreur commise et une approximation initiale des paramètres.

**Paramétrisation.** Il faut une paramétrisation  $\vec{P} = (P_0, \dots, P_N)$  de la surface mathématique sur laquelle devraient se trouver les points. La surface mathématique peut être infinie dans certaines directions (comme c'est le cas pour un plan ou un cylindre).

**Erreur.** Il faut également une expression pour calculer l'erreur en distance  $e_i(\vec{P}, p_i)$  entre un point  $p_i$  de l'espace et la surface paramétrée par  $\vec{P}$ . Il peut s'agir d'une formule exacte ou d'un calcul d'approximation numérique. Nous notons  $\vec{e} = (e_0, \dots, e_N)$  le vecteur formé des erreurs de distance des  $n$  points à la surface.

**Approximation.** Il faut finalement une méthode permettant de calculer une ap-

proximation initiale  $\vec{P}_0$  des paramètres de la surface à partir d'un nuage de points ou à partir des paramètres (position, orientation) des surfaces présentes avant la fusion en une surface plus complexe.

L'objectif est de trouver une valeur de  $\vec{P}$  qui minimise la somme des carrés de l'erreur de distance de chaque point à la surface, tel que décrit par l'équation 3.2.

$$f(\vec{P}) = \vec{e}\vec{e}^T = \sum_{i=0}^N e_i^2(\vec{P}, p_i) \quad (3.2)$$

Il est très important d'avoir une estimation initiale de bonne qualité, et ce, pour deux raisons :

1. Tel qu'il sera montré à la section III.1, l'algorithme peut converger vers un minimum local, ou même diverger si l'estimation initiale est trop éloignée de la bonne solution.
2. Plus la solution initiale est précise, moins il faudra d'itérations pour atteindre la solution recherchée et plus l'algorithme convergera rapidement.

### 3.1.2.1 Algorithme de Levenberg-Marquardt

Pour minimiser l'équation 3.2 et ainsi trouver la meilleure paramétrisation d'une surface pour un ensemble de points donné, l'algorithme de Levenberg-Marquardt (Levenberg, 1944; Marquardt, 1963) est utilisé. Il s'agit d'une méthode itérative qui débute avec l'estimation  $\vec{P}_0$  et qui tente d'améliorer cette estimation à chaque itération en combinant les algorithmes de Gauss-Newton et de descente du gradient.

À partir d'une approximation  $\vec{P}_k$ , une seconde approximation  $\vec{P}_{k+1}$  est obtenue en linéarisant  $\vec{e}$  autour  $\vec{P}_k$ . Cette linéarisation est donnée par l'équation 3.3 où  $J$

représente le jacobien de  $\vec{e}$  autour de  $\vec{P}_k$ .

$$\vec{e}(\vec{P}_{k+1}) = \vec{e}(\vec{P}_k + \Delta\vec{P}) \approx \vec{e}(\vec{P}_k) + J\Delta\vec{P} \quad (3.3)$$

Au minimum de  $f(\vec{P}_k + \Delta\vec{P})$ , nous avons  $\nabla_{\Delta\vec{P}} f(\vec{P}_k + \Delta\vec{P}) = 0$ . Après quelques calculs, nous obtenons l'équation 3.4.

$$(J^T J) \Delta\vec{P} = -J^T \vec{e}(\vec{P}_k) \quad (3.4)$$

Le terme  $J^T J$  est en fait une estimation du Hessien<sup>3</sup> de  $\vec{e}$  autour de  $\vec{P}_k$ . La technique de Levenberg-Marquardt consiste à amortir le Hessien en remplaçant  $J^T J$  par  $J^T J + \lambda I$  dans l'équation 3.4. Le facteur  $\lambda$  permet de contrôler la convergence de l'algorithme. En choisissant  $\lambda$  élevé, l'algorithme se comporte comme un algorithme de suivi de descente gradient et converge lentement vers le minimum recherché. Une fois dans le voisinage de ce minimum, la valeur de  $\lambda$  est revue à la baisse. L'algorithme se comporte alors comme de celui de Gauss-Newton et converge plus rapidement.

Durant une itération de Levenberg-Marquardt, une valeur initiale est fixée pour  $\lambda$ . Si la valeur de  $\vec{e}$  diminue, alors la valeur de  $\lambda$  est revue à la baisse pour l'itération suivante. Si la valeur de  $\vec{e}$  augmente, la valeur de  $\lambda$  est augmentée au cours de la même itération jusqu'à ce qu'une baisse de la valeur de  $\vec{e}$  soit obtenue.

L'algorithme itératif prend fin lorsque la valeur de  $f(\vec{P}_{k+1}) - f(\vec{P}_k)$  devient inférieure à une certaine valeur limite ou lorsqu'un nombre maximal d'itérations est atteint.

---

<sup>3</sup>Une matrice des dérivées partielles du second ordre.

Pour plus de renseignements sur l'algorithme de Levenberg-Marquardt, le lecteur peut se référer aux articles originaux (Levenberg, 1944; Marquardt, 1963). Le module de segmentation courbe utilise la librairie *levmar*, disponible sous licence GPL (Lourakis, 2006). Une revue des détails d'implantation de l'algorithme est également disponible sur le site web référencé.

### 3.1.2.2 Modèles utilisés

Nous avons décrit ci-dessus comment la paramétrisation d'une surface de type donné peut être obtenue en utilisant l'algorithme de Levenberg-Marquardt. Pour nos expériences, nous avons décidé d'utiliser deux types de surfaces courbes : des cylindres de révolution et des sphères. Il s'agit d'un cas particulier de surfaces pouvant être détectées en utilisant l'approche suggérée. Ces modèles sont présentés en détail à l'annexe III. D'autres types de surfaces pourraient être ajoutés en dérivant les modèles appropriés.

### 3.1.2.3 Processus de fusion des régions

Le module de segmentation par surfaces courbes reçoit en entrée le graphe produit par le module d'extraction des surfaces planes (voir section 3.1.1.4). Pour chaque arête du graphe — chaque arête correspond à une intersection — une recherche de surface courbe est réalisée tel que décrit aux sections ci-dessus. Si une surface courbe est trouvée, alors les deux noeuds du graphe reliés par l'arête en question sont fusionnés. Si aucune surface courbe n'est trouvée, les noeuds et l'arête sont laissés inchangés.

Selon les types de surfaces rencontrés pendant ce processus, 4 scénarios peuvent survenir :

**Deux plans.** Si les deux surfaces sont planes, alors tous les types de modèles courbes doivent être essayés.

**Une surface courbe et un plan.** Si une des surfaces est déjà courbe (à la suite d'une détection antérieure), alors le type de surface à rechercher est connu et les paramètres de la surface courbe sont utilisés comme approximation initiale de la surface. Les autres types de surfaces courbes ne sont pas analysés.

**Deux surfaces courbes de même type.** Si les deux surfaces sont courbes et de même type, alors les paramètres de la plus grande surface servent d'approximation initiale. Les autres types de surfaces courbes ne sont pas analysés.

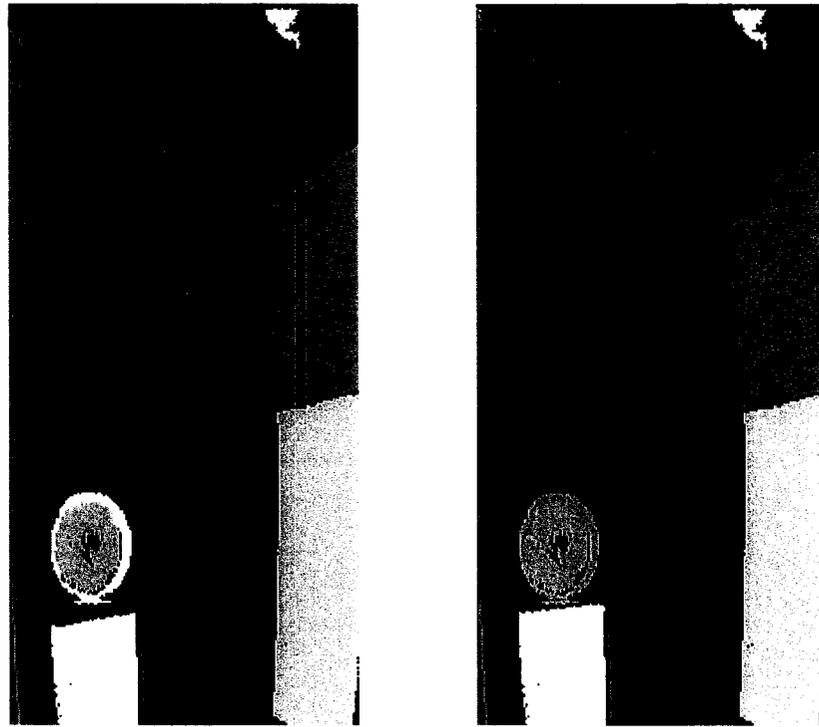
**Deux surfaces courbes de type différent.** Dans ce cas, les surfaces ne peuvent pas être fusionnées.

#### 3.1.2.4 Exemple de résultat

La figure 3.8 présente un résultat d'extraction des surfaces courbes. Toutes les surfaces planes correctement détectées demeurent inchangées alors que les surfaces appartenant au cylindre et à la sphère ont été adéquatement fusionnées. Des résultats supplémentaires sont fournis au chapitre 5.

## 3.2 Segmentation par la photométrie

La section 3.1 traitait de segmentation basée uniquement sur la télémétrie. Toutefois, notre capteur offre deux modalités sensorielles enregistrées : la télémétrie et la



(a) Segmentation en surfaces planes

(b) Segmentation en surfaces planes et courbes

Figure 3.8 Exemple d'extraction de surfaces courbes

photométrie. La présente section traite de segmentation de la scène basée uniquement sur la photométrie. La raison d'être de cette segmentation est double :

1. Certains détails de la scène observée sont invisibles par la télémétrie, mais peuvent être perçus très efficacement par la photométrie. Un exemple trivial est une affiche collée sur un mur.
2. En raison de l'incertitude en distance du capteur télémétrique, certaines frontières entre les surfaces peuvent être mal placées. Dans certains cas, comme il sera montré ci-dessous à la section 3.3.1, la photométrie peut aider à corriger la situation.

Nous avons choisi d'utiliser une technique morphologique, c'est-à-dire qui se base sur le contour des objets et non pas sur la couleur de ceux-ci. L'utilisation des contours rend la technique plus robuste aux conditions lumineuses. Un même objet vu sous plusieurs angles différents risque fort d'avoir une couleur différente, alors que ses contours vont rester sensiblement inchangés<sup>4</sup>. De plus, les contours des surfaces peuvent être associés aux intersections entre les surfaces obtenues par la segmentation télémétrique<sup>5</sup>.

Les techniques de détection de contours se basent sur le gradient de couleur de l'image. L'approche la plus simple, qui consiste à appliquer un seuil sur la norme du gradient, comporte quelques difficultés :

- Les contours obtenus au moyen du gradient sont souvent ouverts, c'est-à-dire que des portions des contours ne sont pas correctement détectées. Si ces contours ne sont pas correctement fermés, deux régions distinctes peuvent se retrouver

---

<sup>4</sup>Nous supposons ici qu'il n'y a pas de variation trop brusque de l'intensité lumineuse.

<sup>5</sup>Nous n'utilisons pas cette propriété actuellement parce que notre capteur fournit directement des images enregistrées. Cette propriété pourrait s'avérer utile si d'autres types de capteurs sont éventuellement utilisés (voir section 3.3.4).

fusionnées.

- Le bruit de mesure du capteur lumineux crée localement des gradients élevés dans l'image et peuvent être confondus avec les contours recherchés.
- La valeur du seuil dépend des conditions lumineuses. Si le seuil est trop élevé, les contours seront très bruités et si le seuil est trop faible, plusieurs trous seront présents dans les contours.

Notre segmentation photométrique s'appuie sur la technique de la ligne de partage des eaux, plus spécifiquement l'implantation de Vincent-Soille (Vincent et Soille, 1991). Il s'agit d'un algorithme séquentiel de type « simulation d'immersion ». Cette technique utilise une détection de contour basée sur la norme du gradient des couleurs de l'image pour en extraire les régions. L'image de contours est considérée comme une image topographique où les zones noires (pas de contours) correspondent à des creux et les zones blanches (contours) correspondent à des zones de haute altitude. Une simulation d'immersion inonde l'image topographique en plusieurs itérations. Des lacs sont créés aux minimums locaux dans l'image et la superficie des lacs augmente au fur et à mesure des itérations où le niveau d'eau est augmenté. Les points où deux lacs se touchent forment une ligne de partage des eaux. Ces lignes de partage des eaux forment les contours définitifs des surfaces. Cette technique permet d'éliminer le problème de fermeture des contours et de choix du seuil.

### 3.2.1 Détails d'implantation

L'image originale est d'abord filtrée au moyen d'un filtre passe-bas gaussien afin de réduire le bruit de mesure pouvant mener à la création de faux contours. La moyenne des normes du gradient de l'image pour chaque canal de couleur est ensuite calculée. L'image de contours ainsi obtenue est filtrée elle aussi au moyen d'un

filtre passe bas gaussien. Cette opération diminue l'intensité des faux contours dus au bruit ou aux variations d'intensité lumineuse et aide à fermer les contours de l'image. Un seuil est appliqué pour retirer tous les contours dont la valeur est inférieure à une valeur limite : les pixels qui ont une intensité inférieure à  $S_P$  voient leur valeur d'intensité mise à 0. Cela permet d'éliminer les minimums locaux qui créent une segmentation excessive de l'image suite à l'application de la ligne de partage des eaux. En appliquant ce seuil, nous favorisons un remplissage rapide du centre des régions et laissons l'algorithme de la ligne de partage des eaux déterminer les contours plus précisément. La ligne de partage des eaux est ensuite appliquée et les régions définitives sont retenues. L'usage de l'algorithme de Vincent-Soille permet d'obtenir une complexité algorithmique  $\mathcal{O}(n)$ . Cette segmentation est donc très rapide.

Le seul paramètre de cette technique est le seuil  $S_P$  qui détermine le niveau de segmentation désiré. Plus  $S_P$  est bas et plus les détails seront pris en compte. Le nombre de surfaces créées sera également plus grand. Si  $S_P$  est grand, seuls les contours importants de l'image, et donc seules les caractéristiques photométriques principales, seront retenus. Si  $S_P$  est trop petit, les variations de l'intensité lumineuse créeront une segmentation excessive de l'image alors que si  $S_P$  est trop grand, il y aura une segmentation insuffisante de l'image. Le paramètre  $S_P$  a donc un rôle équivalent au rôle du paramètre  $r$  de Comaniciu et Meer (Comaniciu et Meer, 1997). Le choix de  $S_P$  doit se faire par calibration en prenant quelques images dans l'environnement cible et en faisant varier  $S_P$ . La valeur de  $S_P$  demeure inchangée si les conditions d'éclairage demeurent également inchangées. En utilisant une cible calibrée, il serait possible de concevoir un algorithme de calibration automatique. Cet algorithme numériserait la scène calibrée et optimiserait la valeur de  $S_P$  pour obtenir les régions prédéterminées lors de la calibration de la cible.

### 3.2.2 Exemple de résultat

La figure 3.9, page 93, présente un résultat de segmentation photométrique. Il est intéressant de voir que cette segmentation présente plusieurs éléments qui étaient invisibles dans la segmentation télémétrique (voir figure 3.8, page 88). Parmi ces éléments, nous notons le babillard sur le mur, la séparation des tiroirs du petit classeur à la gauche de l'image et les rainures du ballon. Toutefois, la segmentation photométrique manque certains éléments importants telles que les divisions entre les deux murs et entre le mur et le plafond. Notons que les murs sont exactement de la même couleur ; d'un point de vue photométrique, il est donc juste qu'ils soient fusionnés. Le plafond a une couleur très proche de celle des murs et il n'est donc pas choquant de l'associer à cette région. Les zones noires dans l'image segmentée sont dues à la suppression des régions jugées trop petites (dont le nombre de points est inférieur à  $N_{min}$ ). Notons finalement que la surface correspondant au dessus du petit classeur à la gauche de l'image n'a pas été correctement segmentée en raison de sa faible taille et des jeux d'ombre créés par la sphère. Cela montre que plus les surfaces sont petites, plus les effets d'ombre et de bruit de mesure sont importants.

### 3.3 Fusion des segmentations télémétriques et photométriques

Les modules de segmentations télémétriques et photométriques (voir sections 3.1 et 3.2) fournissent deux segmentations différentes de la même réalité. Afin d'obtenir une représentation finale détaillée de la scène, nous procédons à la fusion de ces deux segmentations. Ce processus consiste à prendre les éléments pertinents des deux segmentations précédentes et à les combiner pour obtenir une nouvelle représentation plus riche que chacune des deux autres prise individuellement.

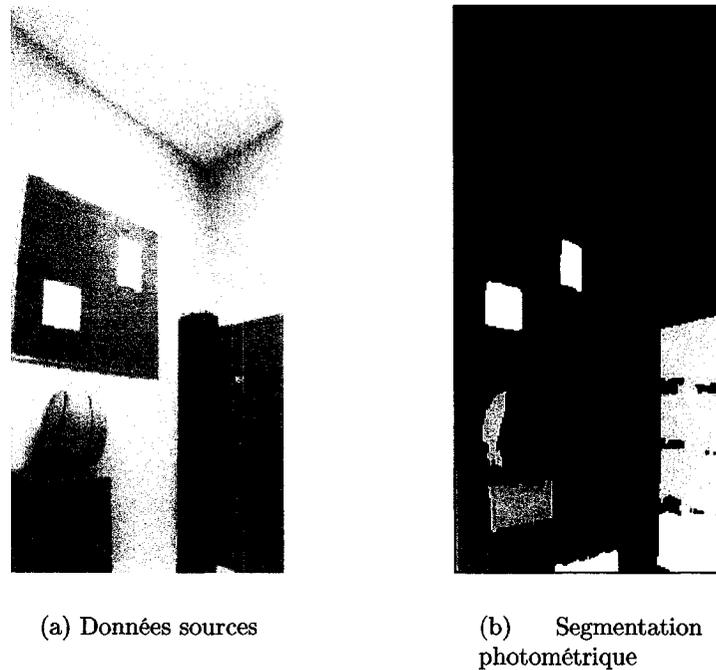


Figure 3.9 Résultat d'une segmentation photométrique

L'idée de base est de considérer chaque segmentation comme une image colorée sur un acétate et de superposer les deux acétates l'un sur l'autre. L'image résultante sera alors la somme des deux images précédentes. Cette superposition est possible parce que les données télémétriques et photométriques sont enregistrées à la source par le capteur.

La figure 3.10 illustre un problème obtenu si nous appliquons intégralement cette approche. L'analyse de ce résultat montre que l'idée de base fonctionne correctement. Toutefois, il y a une segmentation excessive visible près des intersections entre le plancher et les murs. Ce comportement indésirable s'explique en analysant le fonctionnement de la segmentation télémétrique. Lors de cette segmentation, nous avons accepté une erreur en distance de  $d_{max}$  entre un point et une surface pour tenir compte de l'imprécision de mesure du capteur. Une conséquence de cette

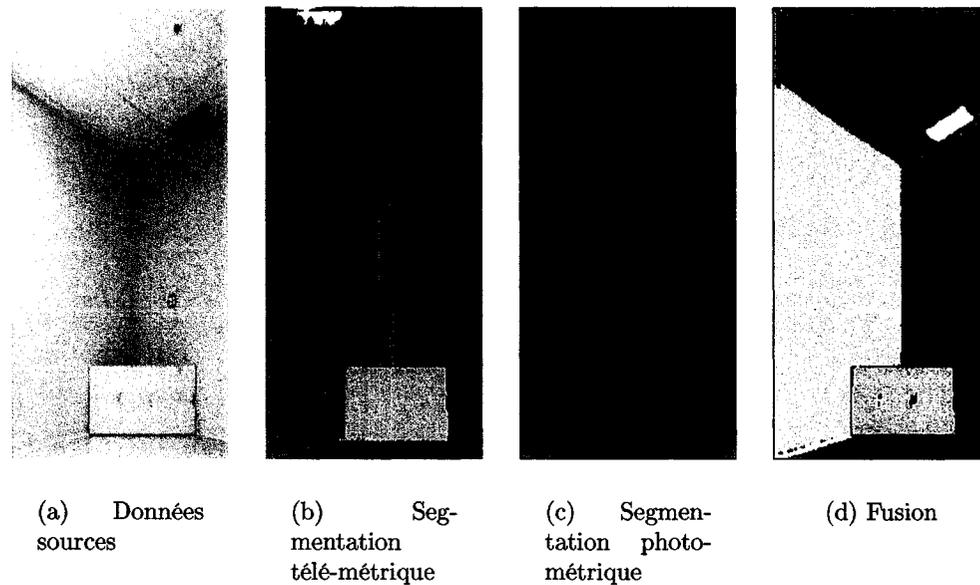


Figure 3.10 Fusion des segmentations

approche est que certains points situés à proximité des intersections peuvent alors appartenir à plus d'une surface. Le phénomène observé à la figure 3.10 s'explique par une mauvaise classification de certains points par l'algorithme de segmentation télémétrique liée à cette ambiguïté. Le problème des points mal catégorisés aux intersections est connu (voir (Lee et al., 1998)). Régler ce problème uniquement par la télémétrie est très compliqué et nécessite une consommation de temps CPU élevée. Une telle approche est l'utilisation d'algorithmes génétiques proposée par Gotardo et al. (Gotardo et al., 2003).

Nous suggérons plutôt ici de tirer profit du fait que notre capteur fournit télémétrie et photométrie. Plus particulièrement, nous allons montrer comment la photométrie peut être utilisée pour désambiguïser la télémétrie.

### 3.3.1 Utilisation de la photométrie pour désambiguïser la télémétrie

Premièrement, il est important de déterminer quels sont les points à risque dans l'image télémétrique. Pour ce faire, un post-traitement est appliqué à la segmentation télémétrique pour trouver tous les points situés à une distance inférieure à  $d_{max}$  de plus d'une surface. Nous associons alors à chaque point une liste des surfaces auxquelles il peut appartenir. L'idée est de prendre les points pouvant appartenir à plus d'une région télémétrique, de regarder à quelle région photométrique ils appartiennent et de trouver quelle est la région télémétrique parmi les choix possibles qui correspond le mieux à cette région photométrique. Cette idée a un équivalent qui est plus pratique à implanter. Nous effectuons la superposition des deux segmentations en prenant soin de garder toutes les attributions possibles pour chaque pixel de la segmentation télémétrique. Puis nous conservons les attributions menant à la création des régions finales fusionnées les plus grandes.

L'algorithme est plus facile à comprendre par l'usage d'un exemple fictif dans lequel nous pouvons identifier clairement tous les pixels et toutes les régions télémétriques et photométriques auxquelles chaque pixel peut appartenir. La figure 3.11 (a) présente le résultat de la segmentation télémétrique d'une image fictive, alors que le résultat de la segmentation photométrique est présenté en (b) et la fusion en (c). Suite à cette fusion, six régions peuvent être créées. La taille maximale possible de ces régions est donnée dans le tableau 3.1. Nous utilisons ces informations pour choisir les zones télémétriques finales, tel que montré au tableau 3.2. La segmentation télémétrique résultante est présentée à la figure 3.11 (d).

1	1	1/2	1/2	1/2	2	2	2/3	2/3	2/3	2/3	2/3	3	3	3	3	3	3	3	3
1	1	1/2	1/2	1/2	2	2	2/3	2/3	2/3	2/3	2/3	3	3	3	3	3	3	3	3
1	1	1/2	1/2	1/2	2	2	2/3	2/3	2/3	2/3	2/3	3	3	3	3	3	3	3	3
1	1	1/2	1/2	1/2	2	2	2/3	2/3	2/3	2/3	2/3	3	3	3	3	3	3	3	3

(a) Segmentation télémétrique

A	A	A	A	A	A	A	A	A	A	A	B	B	B	B	B	C	C	C	C
A	A	A	A	A	A	A	A	A	A	A	B	B	B	B	B	C	C	C	C
A	A	A	A	A	A	A	A	A	B	B	B	B	B	B	C	C	C	C	C
A	A	A	A	A	A	A	A	A	B	B	B	B	B	B	C	C	C	C	C

(b) Segmentation photométrique

1A	1A	1/2A	1/2A	1/2A	2A	2A	2/3A	2/3A	2/3A	2/3A	2/3B	3B	3B	3B	3B	3C	3C	3C	3C
1A	1A	1/2A	1/2A	1/2A	2A	2A	2/3A	2/3A	2/3A	2/3A	2/3B	3B	3B	3B	3B	3C	3C	3C	3C
1A	1A	1/2A	1/2A	1/2A	2A	2A	2/3A	2/3A	2/3B	2/3B	2/3B	3B	3B	3B	3B	3C	3C	3C	3C
1A	1A	1/2A	1/2A	1/2A	2A	2A	2/3A	2/3A	2/3B	2/3B	2/3B	3B	3B	3B	3B	3C	3C	3C	3C

(c) Fusion des segmentations

1	1	2	2	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3	3
1	1	2	2	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3	3
1	1	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3	3	3	3
1	1	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3	3	3	3

(d) Résultat de la désambiguïsation

Figure 3.11 Exemple de désambiguïsation de la télémétrie par la photométrie

Région	Taille maximale
1A	20
2A	32
2B	8
3A	12
3B	24
3C	16

Tableau 3.1 Taille des régions de l'exemple de la figure 3.11

Valeur	Choix	Note
1/2A	2 (car $2A > 1A$ )	La photométrie est inutile ici. Le choix est le même que celui qui aurait été fait par l'algorithme RANSAC.
2/3A	2 (car $2A > 3A$ )	La photométrie permet de désambiguïser entre les régions 2 et 3.
2/3B	3 (car $3B > 2B$ )	La photométrie permet de désambiguïser entre les régions 2 et 3.

Tableau 3.2 Choix des régions télémétriques et justifications accompagnant l'exemple de la figure 3.11

En appliquant cet algorithme à l'exemple réel présenté à la figure 3.10, nous obtenons le résultat présenté à la figure 3.12. La segmentation excessive de la scène a été complètement éliminée, tel que désiré.

La question qui se pose maintenant est : où mettre cette étape dans le schéma général de segmentation ? La première idée qui vient à l'esprit est de la mettre après que les deux segmentations, télémétrique et photométrique, soient terminées. Toutefois, l'ambiguïté entre en jeu uniquement dans le module d'extraction des surfaces planes puisque le paramètre  $d_{max}$  n'est utilisé que dans cette section. De plus, la désambiguïstation de la télémétrie, en rendant la segmentation télémétrique plus précise, peut avoir un impact sur l'extraction des surfaces courbes. Pour ces raisons, cette phase de la segmentation doit être placée entre le module d'extraction des surfaces planes et celui d'extraction des surfaces courbes. Le résultat de l'extraction des surfaces courbes peut ensuite être superposé simplement à la segmentation photométrique pour obtenir la segmentation finale de la scène.

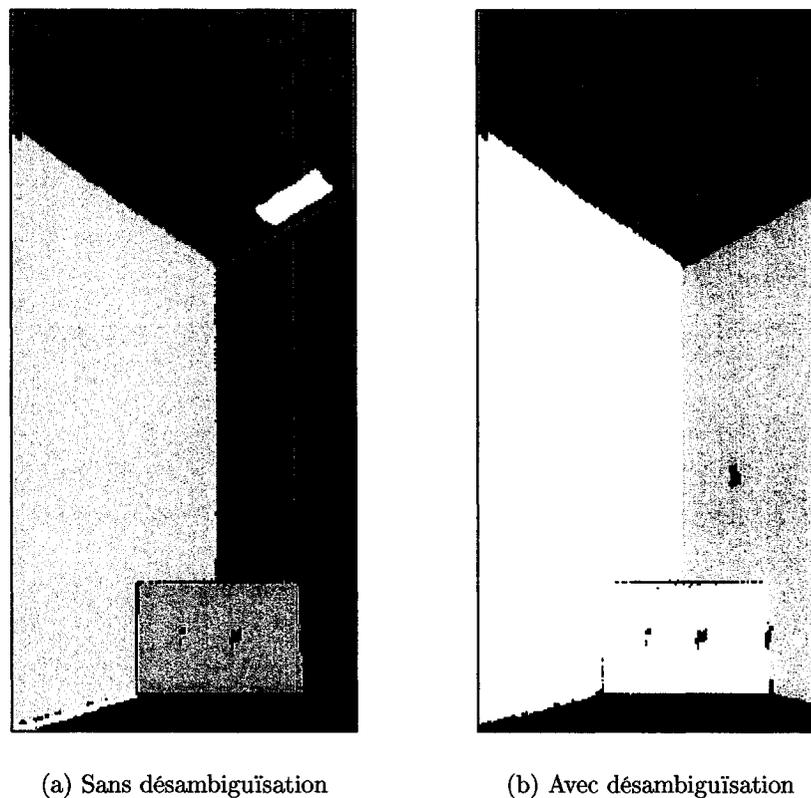


Figure 3.12 Effet de la désambiguïsation sur la qualité de la fusion

### 3.3.2 Schéma de segmentation détaillé

Le processus complet de segmentation est plus complexe que celui présenté initialement à la figure 3.1, page 69. Un schéma plus détaillé est présenté à la figure 3.13 avec la légende donnée au tableau 3.3.

Symbole	Description	Section référence	Lettre	Contenu
FS	Filtre spatial	3.1.1.3	a	Données brutes provenant du capteur
SP	Segmentation en surfaces planes	3.1.1	b	Données télémétriques validées
Aj	Désambiguïsation de la télémétrie	3.3.1	c	Surfaces planes
4C	Recherche des régions 4-connectées et calcul des intersections	3.1.1.6	d	Données photométriques filtrées
ESC	Extraction des surfaces courbes	3.1.2	e	Contours
Fu	Fusion des segmentations télémétriques et photométriques	3.3	f	Contours filtrés
FG	Filtre gaussien	3.2.1	g	Principaux contours
So	Sobel	3.2.1	h	Régions photométriques
Se1	Seuil $S_P$ sur l'intensité des contours	3.2.1	i	Surfaces planes corrigées
Se2	Retrait des régions de taille inférieure à $N_{min}$	3.1.1.1 et 3.2.2	j	Surfaces planes et intersections
LPE	Ligne de partage des eaux	3.2	k	Surfaces planes et courbes
			l	Éléments surfaciques
			m	Résultat final de la segmentation

Tableau 3.3 Légende de la figure 3.13

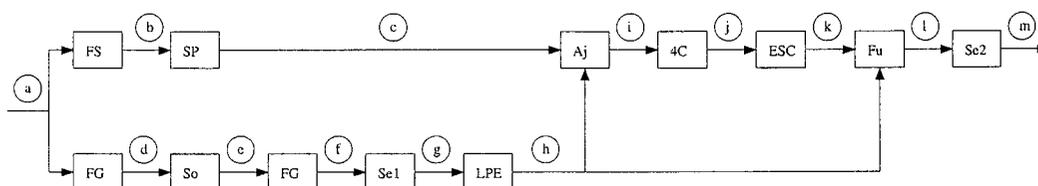


Figure 3.13 Schéma détaillé du processus complet de segmentation

### 3.3.3 Exemple de résultat

La figure 3.14 présente un résultat de la fusion des segmentations télémétriques et photométriques. Le résultat de la fusion comporte toutes les surfaces détectées par la segmentation télémétrique avec tous les détails de la segmentation photométrique. Nous notons que le néon est divisé en deux surfaces. La plus grande surface est située au niveau du plafond et correspond au plastique protecteur et la plus petite surface est située légèrement plus haut et provient des rayons du laser ayant traversé le plastique protecteur pour atteindre le fond de la boîte du néon.

### 3.3.4 Cas des données non enregistrées

La fusion des segmentations a été simplifiée par le fait que les données fournies par le capteur étaient déjà enregistrées. Que faire dans le cas où les données ne sont pas enregistrées à la source ? Cette section fournit quelques pistes de réflexion à ce sujet.

Dans le cas où les données télémétriques et photométriques ne sont pas initialement enregistrées, plusieurs problèmes se posent, dont celui des occlusions. Les données télémétriques et photométriques ne peuvent pas être prises du même point de vue et, par conséquent, il est possible que certains éléments soient visibles par un capteur et ne le soient pas par l'autre.

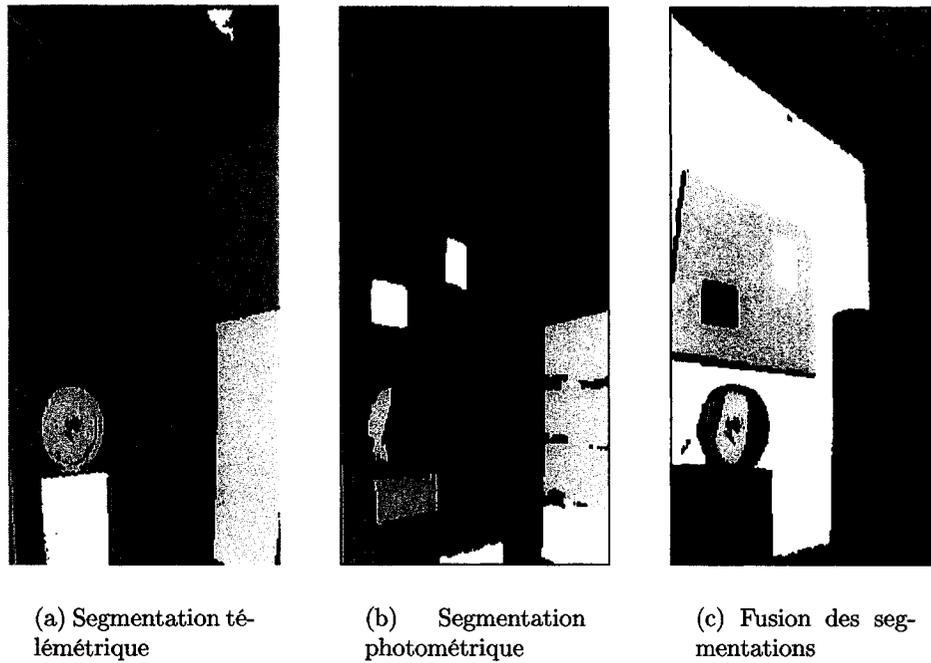


Figure 3.14 Exemple d'une fusion des segmentations télémétriques et photométriques

Nous supposons que les occlusions sont négligeables ou qu'il existe une méthode pour les détecter, comme ce fut le cas avec notre capteur (voir section 2.5.1).

Une discontinuité géométrique peut être accompagnée d'une discontinuité photométrique pour une des raisons suivantes :

- La discontinuité est liée à un changement du type de surface et les deux surfaces ont une couleur/texteure différente.
- La lumière n'est pas réfléchi de la même manière de part et d'autre de la discontinuité.

Ainsi, certains contours de l'image télémétrique devraient être visibles également dans l'image photométrique. Si la position de la caméra par rapport au télémètre laser 3D était connue avec précision, alors il serait possible de calculer la projection des arêtes 3D dans l'image obtenue par la caméra et certaines de ces arêtes concorderaient parfaitement avec les arêtes obtenues par la segmentation photométrique. Le problème de la registration serait alors résolu puisqu'il suffirait de projeter de la même manière toutes les surfaces 3D obtenues dans l'image 2D pour registrer les deux modalités sensorielles. Le problème de l'ambiguïté télémétrique discuté à la section 3.3 créerait toutefois une petite erreur. En calculant les intersections des surfaces mathématiques sur lesquelles se trouvent les surfaces et en projetant ces intersections dans l'image 2D au lieu d'y projeter les contours des surfaces, l'erreur devrait être corrigée.

La principale difficulté ici est que nous ne connaissons pas la position de la caméra par rapport au capteur (sinon les données seraient registrées!). Nous pouvons émettre l'hypothèse que nous connaissons une bonne approximation de cette position. Cette approximation peut être fournie par le montage ou par le mode d'acquisition des données. Nous pourrions alors utiliser l'algorithme itératif ICP

pour améliorer cette estimation. Pour ce faire, il faudrait procéder comme suit :

1. En utilisant l'estimation précédente, nous projetons les arêtes de la segmentation télémétrique sur l'image de contours photométrique.
2. Pour chaque point projeté, nous lui associons son plus proche voisin de contour de l'image photométrique. Nous conservons uniquement les paires de points dont la distance en pixels est inférieure à  $\tau_{dist}$ <sup>6</sup>.
3. La nouvelle estimation de la position relative des deux capteurs est celle qui minimise les distances entre les points associés (moindres carrés).
4. Il faut un critère d'arrêt. Le critère préférable est d'arrêter lorsque les deux dernières projections sont identiques ou lorsque la différence est très faible.

#### 3.3.4.1 Autocalibration du capteur

La technique décrite ci-dessus n'est pas utile dans le cas où le capteur fournit des données enregistrées. Toutefois, elle pourrait être utilisée pour effectuer une auto-calibration du capteur. Ainsi, si pour une raison inconnue, le capteur perd sa calibration, l'erreur pourrait être détectée et un mécanisme de rétroaction pourrait ajuster dynamiquement les paramètres intrinsèques du capteur. Une telle boucle de contrôle serait un ajout intéressant à réaliser lors d'un travail futur.

### 3.4 Régions finales

Une fois que tous les éléments surfaciques sont identifiés, la dernière étape consiste à tracer les contours de ces régions. Cette étape permet de conserver en mémoire

---

<sup>6</sup>Ce paramètre est présent dans l'algorithme classique d'ICP.

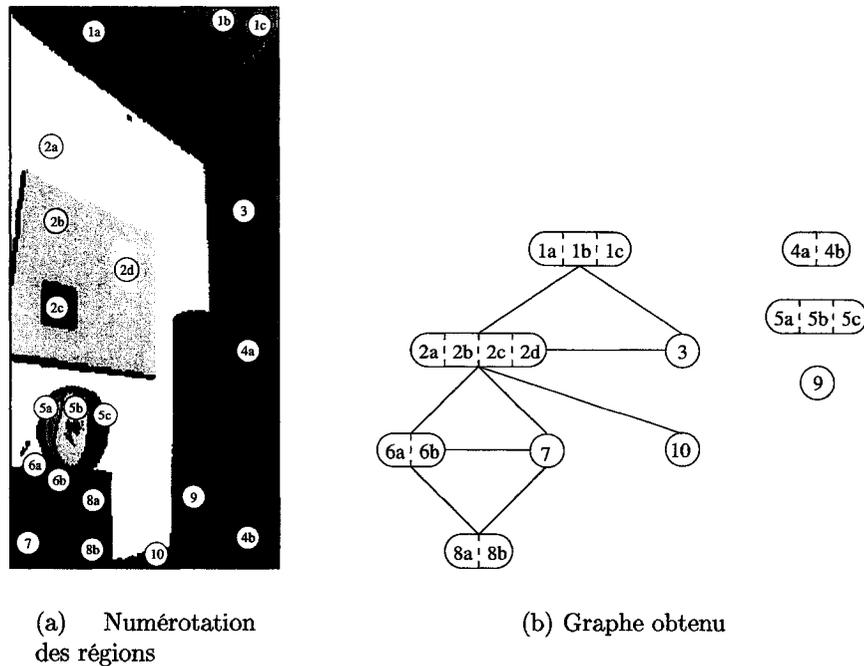


Figure 3.15 Graphe correspondant au résultat de segmentation montré à la figure 3.14

uniquement les paramètres de la surface mathématique où résident l'élément surfacique et le contour de cet élément. Cette représentation est suffisante pour la majorité des applications et est beaucoup plus compacte que de conserver l'ensemble des points de la surface.

Les contours des surfaces sont obtenus par l'algorithme de Square-Tracing (Ghuneim, 2000). L'algorithme original permet de trouver le contour d'une tache noire 4-connectée sur fond blanc. Nous l'avons adapté légèrement pour trouver le contour de toutes les surfaces de l'image.

Le résultat de la segmentation complète est un graphe où chaque noeud est une surface et où chaque arête est une intersection entre deux surfaces. Les surfaces sont le résultat de la segmentation télémétrique et chacune d'entre elles est composée

d'un ou de plusieurs éléments surfaciques. Les éléments surfaciques sont les résultats de la fusion des segmentations télémétriques et photométriques. Nous avons donc deux niveaux de détail. Le premier niveau est une division de l'environnement en surfaces basée sur la télémétrie et le second niveau est une division des surfaces en éléments surfaciques basée sur la photométrie. Le graphe obtenu pour le résultat de la figure 3.14 (c), page 101, est illustré à la figure 3.15. Sur cette figure, les nombres représentent les surfaces et les lettres représentent les éléments surfaciques.

Pour chaque surface et élément surfacique, nous conservons uniquement les renseignements suivants :

- Les paramètres de la surface mathématique contenant la région ;
- Le nombre de points que possédait la région (estimateur de la taille de la région) ;
- La moyenne des composantes rouges, vertes et bleues des points qui composaient la région<sup>7</sup> ;
- Le contour de la région, conservé comme une liste ordonnée de points.

En plus de ces caractéristiques, une surface contient aussi la liste des éléments surfaciques qu'elle possède.

### 3.5 Conclusion du chapitre sur la segmentation

Nous avons développé une nouvelle approche qui permet de segmenter des images combinant télémétrie et photométrie. Deux segmentations séparées sont effectuées : l'une basée sur la télémétrie et l'autre sur la photométrie. Ces segmentations sont ensuite fusionnées pour obtenir une segmentation qui prend en compte les deux modalités sensorielles. Cette approche nous permet de tirer profit des recherches

---

<sup>7</sup>Cette couleur moyenne n'a de sens en fait que pour les éléments surfaciques et les surfaces composées d'un seul élément surfacique.

de pointes réalisées séparément dans les deux domaines de segmentation et d'augmenter la robustesse de notre système.

La segmentation télémétrique est divisée en deux modules. Le premier module réalise une segmentation en surfaces planes et le second module utilise ces surfaces planes pour extraire les surfaces courbes. Le premier module utilise l'estimateur RANSAC alors que le second module utilise l'algorithme de Levenberg-Marquardt pour trouver le modèle qui minimise les erreurs de distance. Entre ces deux modules, s'insère un autre module qui permet de préciser les contours des surfaces planes au moyen de la photométrie. Les modèles de surfaces utilisés sont les surfaces planes, sphériques et cylindriques.

La segmentation photométrique est réalisée au moyen de l'algorithme de la ligne de partage des eaux. Un double filtrage et une technique de seuillage permettent d'éviter le problème de la segmentation excessive généralement associé à cette technique.

La fusion des deux segmentations permet d'obtenir des résultats où tous les détails détectés grâce à la télémétrie ou à la photométrie sont présents. Des résultats supplémentaires sont fournis au chapitre 5.

Finalement, une discussion sur le cas des données non enregistrées a permis de déterminer les fondements d'une technique d'autocalibration du capteur utilisé. L'implantation de cette technique pourra faire l'objet de travaux futurs.

Les surfaces et éléments surfaciques détectés grâce à notre méthode de segmentation doivent maintenant être intégrés à la carte en construction. Ce sujet fait l'objet du prochain chapitre.

## CHAPITRE 4

### INTÉGRATION DE VUES MULTIPLES

Au chapitre 2, nous avons présenté un nouveau capteur pour l'acquisition d'images 3D combinant télémétrie et photométrie. Puis, au chapitre 3, nous avons utilisé ces deux modalités sensorielles pour segmenter les images obtenues dans le but de détecter les surfaces des objets composant l'environnement observé. La dernière étape à réaliser dans le but d'effectuer de la cartographie 3D est de fusionner plusieurs images d'un même lieu afin d'obtenir une carte plus grande et plus complète de ce lieu.

Au chapitre 1, nous avons vu que le problème de l'intégration de vues multiples se divise en trois sous-problèmes :

1. Choisir les points de vue optimaux ;
2. Localiser la nouvelle image dans la carte en cours de construction ;
3. Intégrer la nouvelle image à la structure de données qui constitue la carte.

Dans le cadre de ce mémoire, nous avons laissé de côté le premier sous-problème. Il s'agit d'un problème complexe auquel il existe déjà certaines solutions dont celles de Nüchter et al. (Nüchter et al., 2003a), Sequeira et al. (Sequeira et al., 1998) et Whaite et Ferrie (Whaite et P., 1997). Pour notre projet, les choix des points de vue et le contrôle du déplacement de la plate-forme mobile entre ces différents points de vue sont réalisés par un opérateur qui guide la plate-forme mobile d'un point de vue au suivant. Sur demande de l'opérateur, la plate-forme mobile effectue une acquisition 3D, la segmente, détermine sa position dans la carte en cours de

construction et ajoute la nouvelle image à la carte en fusionnant les surfaces de la nouvelle image avec celles déjà présentes dans la carte.

Dans ce chapitre, nous présentons d'abord une solution au problème de la localisation, puis nous décrivons le processus d'intégration des nouvelles images à la carte en construction.

#### 4.1 Localisation

Trouver où insérer la nouvelle image dans la carte en construction revient à trouver la position de la plate-forme mobile dans cette carte. La plate-forme mobile que nous utilisons est équipée d'un système de positionnement relatif : des odomètres permettent de mesurer le déplacement effectué par chaque roue et d'en calculer les déplacements linéaires et angulaires de la plate-forme mobile. La position de la plate-forme est obtenue par intégration des déplacements effectués, ce qui a pour effet d'accumuler les erreurs de mesure. En conséquence, l'erreur de positionnement augmente en fonction du déplacement effectué par la plate-forme. L'objectif du module de localisation est de raffiner cette estimation de la position avant de procéder à l'intégration définitive des nouvelles données dans la carte en construction. Cette correction de la position est réalisée en comparant les surfaces présentes dans la nouvelle image avec les surfaces déjà présentes dans la carte.

Le principal algorithme utilisé en robotique mobile pour effectuer ces corrections est l'algorithme itératif des points les plus proches connu sous l'acronyme ICP<sup>1</sup> (Besl et McKay, 1992). Cet algorithme travaille directement sur les points acquis au moyen d'un télémètre laser. Une première estimation de la position est fournie, généralement, par l'odométrie de la plate-forme et est utilisée pour localiser

---

<sup>1</sup>Iterative Closest Points

approximativement la nouvelle acquisition dans la carte en construction. Ensuite, pour chaque point de la nouvelle acquisition, le point le plus proche dans la carte en construction lui est associé. Nous obtenons ainsi une série de paires de points. Normalement certains points de la nouvelle acquisition ne sont pas déjà présents dans la carte. Ces points n'auront donc pas de proches voisins et seront appariés, à tort, à des points éloignés. Pour contrer cet effet, un seuil est appliqué sur la distance entre les points pour conserver uniquement les paires dont la distance entre les points est inférieure à une valeur prédéterminée. La transformation géométrique minimisant les distances entre les points de chaque paire est calculée et appliquée à la nouvelle acquisition. L'algorithme est répété itérativement jusqu'à ce qu'il y ait convergence de la position obtenue.

Cet algorithme possède l'inconvénient d'être très lent en raison des recherches nécessaires pour trouver les plus proches voisins. De plus, le temps de recherche augmente à chaque nouvel ajout de point dans la carte. Nous présentons ici un nouvel algorithme basé sur une modification de l'algorithme ICP pour utiliser les surfaces plutôt que les points pour le positionnement de la plate-forme. Le principal avantage de notre algorithme par rapport à l'approche classique est sa rapidité : le nombre de surfaces étant de plusieurs ordres de grandeur inférieur au nombre de points, l'algorithme utilisé est beaucoup plus rapide. De plus, l'ajout de nouvelles données à une surface déjà existante n'a aucun effet sur les performances de l'algorithme. Cet algorithme est décrit ci-dessous.

#### 4.1.1 Description de l'algorithme utilisé

Tout comme l'algorithme ICP classique, nous utilisons l'odométrie pour obtenir une première estimation de la position de la plate-forme. Puis nous créons des paires d'éléments surfaciques composées d'un élément de la nouvelle acquisition et

d'un élément de la carte en construction. Pour être pairés, ces éléments surfaciques doivent être de même type (deux éléments plans, deux éléments cylindriques ou deux éléments sphériques), de même couleur, d'orientation semblable et proches les uns des autres. La taille des contours des éléments surfaciques n'est pas prise en compte, car il est probable que les éléments n'aient pas été vus en totalité. Nous calculons ensuite la rotation nécessaire pour minimiser l'écart d'orientation entre les surfaces de chaque paire puis la translation nécessaire pour minimiser l'écart de distance entre ces surfaces. Ces étapes sont exécutées de manière itérative jusqu'à ce qu'il y ait convergence de la position de la plate-forme. Cette position corrigée est ensuite utilisée comme nouvelle référence pour les calculs d'odométrie. De cette manière, la position fournie par l'odométrie de la plate-forme est corrigée à chaque nouvelle acquisition 3D.

Nous avons implanté cette stratégie en deux dimensions, ce qui est acceptable en milieu intérieur puisque la plate-forme mobile se déplace généralement sur un plancher plat. De plus, nous avons simplifié notre implantation en ne développant nos équations que pour les surfaces planes qui sont dominantes en milieu intérieur.

#### **4.1.1.1 Appariement des surfaces**

Deux éléments surfaciques plans, l'un provenant de la nouvelle acquisition et l'autre de la carte en construction, peuvent être appariés si, et seulement si, ils respectent les conditions suivantes :

1. Être de même couleur. Nous rappelons que la couleur d'un élément surfacique est déterminée à la suite de la segmentation en prenant la moyenne des points associés à cet élément. En raison des variations de l'éclairage, les deux éléments surfaciques ne seront probablement pas de la même couleur.

Nous devons alors vérifier s'ils sont de couleur semblable. Pour ce faire, nous considérons la couleur de l'élément surfacique comme un vecteur  $\vec{c}$  dans  $\mathbb{N}^3$  et vérifions si les vecteurs ont la même orientation, ce qui est déterminé par le respect de l'inégalité 4.1. En vérifiant l'orientation des vecteurs plutôt que la norme de la différence de ceux-ci, nous obtenons une relation moins sensible à une variation de luminosité.

$$\arccos \frac{|\vec{c}_1 \cdot \vec{c}_2|}{\|\vec{c}_1\| \|\vec{c}_2\|} \leq \tau_{coul} \quad (4.1)$$

2. Être d'orientation semblable. Pour ce faire, nous vérifions si l'inégalité 4.2 est respectée. Cette inégalité vérifie si l'angle entre les vecteurs normaux des deux surfaces (donc l'angle entre les deux plans) est inférieur à une valeur seuil prédéterminée. Cette valeur devrait être déterminée en fonction de la qualité de l'odométrie de la plate-forme mobile.

$$\arccos \frac{|\vec{n}_1 \cdot \vec{n}_2|}{\|\vec{n}_1\| \|\vec{n}_2\|} \leq \tau_{ang} \quad (4.2)$$

3. Être assez proches. Malheureusement, il n'existe pas de méthode universelle pour calculer la distance entre deux plans non-parallèles donnant un résultat non nul. Nous avons choisi de calculer la distance entre le barycentre de la nouvelle surface et le plan dans lequel se trouve la surface provenant de la carte. Nous vérifions que cette valeur est inférieure à une valeur  $\tau_{dist}$  prédéterminée. Cette métrique possède l'avantage d'être rapide à calculer et d'être peu influencée par les petites variations des orientations des surfaces qui peuvent être dues à l'imprécision du capteur. La valeur de  $\tau_{dist}$  devrait être déterminée en fonction de la précision de l'odométrie de la plate-forme mobile.

Si plusieurs éléments surfaciques de la carte peuvent être appariés à un même élément surfacique de la nouvelle acquisition, nous choisissons celui qui minimise la distance du critère 3. Nous choisissons donc l'élément surfacique le plus proche, conformément à l'algorithme ICP classique. Une fois les éléments surfaciques appariés, nous pouvons calculer la rotation et la translation optimales pour localiser la nouvelle image dans la carte en construction.

#### 4.1.1.2 Calcul de la rotation

Puisque nous travaillons en deux dimensions, nous calculons l'angle entre deux éléments surfaciques plans en projetant leur vecteur normal sur le plan  $z = 0$  et en calculant l'angle  $\gamma_i$  entre les projections ainsi obtenues. Les plans parallèles au plan  $z = 0$  ne sont pas pris en considération, car ils ne peuvent pas servir à l'orientation de la plate-forme. Nous cherchons à calculer la rotation d'angle  $\Delta\gamma$  qui minimise l'angle entre les éléments surfaciques de chaque paire  $i$ . Pour ce faire, nous utilisons la formule des moindres carrés 4.3. Les carrés des distances sont pondérés en fonction du nombre de point des éléments surfaciques appariés. Dans la méthode ICP classique, le nombre maximal de paires de points entre deux éléments surfaciques donnés est égal au nombre de points de l'élément qui en possède le moins. Ainsi, pour respecter l'algorithme ICP classique,  $\rho_i$  doit être égal au nombre de points de l'élément surfacique ayant le moins de points dans la paire  $i$ . Le minimum est atteint lorsque  $\Delta\gamma = \frac{\sum_i \rho_i \gamma_i}{\sum_i \rho_i}$ .

$$\min_{\Delta\gamma} \sum_i \rho_i (\gamma_i - \Delta\gamma)^2 \quad (4.3)$$

### 4.1.1.3 Calcul de la translation

Pour le calcul de la translation, nous cherchons à minimiser la distance entre les éléments surfaciques de chaque paire  $i$ . Plus précisément, nous minimisons la distance entre le barycentre du nouvel élément surfacique et le plan dans lequel habite l'élément surfacique contenu dans la carte. La distance du barycentre  $p_i = (x_i, y_i, z_i)$  au plan donné par l'équation  $a_i x + b_i y + c_i z + d_i = 0$  est donné par la formule 4.4.

$$dist = \frac{|a_i x_i + b_i y_i + c_i z_i + d_i|}{\sqrt{a_i^2 + b_i^2 + c_i^2}} \quad (4.4)$$

En choisissant la valeur de  $d_i$  appropriée, nous pouvons forcer  $a_i^2 + b_i^2 + c_i^2 = 1$  et la formule 4.4 devient alors la formule simplifiée 4.5.

$$dist = |a_i x_i + b_i y_i + c_i z_i + d_i| \quad (4.5)$$

L'objectif est de trouver la translation  $(\Delta x, \Delta y)$  qui minimise la somme des carrés des distances, conformément à l'équation 4.6. La solution de cette équation est donné par la résolution du système linéaire 4.7.

$$\min_{\Delta x, \Delta y} \sum_i \rho_i (a_i (x_i - \Delta x) + b_i (y_i - \Delta y) + c_i z_i + d_i)^2 \quad (4.6)$$

$$\begin{bmatrix} \sum_i \rho_i a_i^2 & \sum_i \rho_i a_i b_i \\ \sum_i \rho_i a_i b_i & \sum_i \rho_i b_i^2 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} \sum_i \rho_i a_i (a_i x_i + b_i y_i + c_i z_i + d_i) \\ \sum_i \rho_i b_i (a_i x_i + b_i y_i + c_i z_i + d_i) \end{bmatrix} \quad (4.7)$$

La forme de la matrice 2x2 de la partie gauche de l'équation 4.7 est très importante. Si cette matrice est singulière, alors le système admet une infinité de solutions, ce

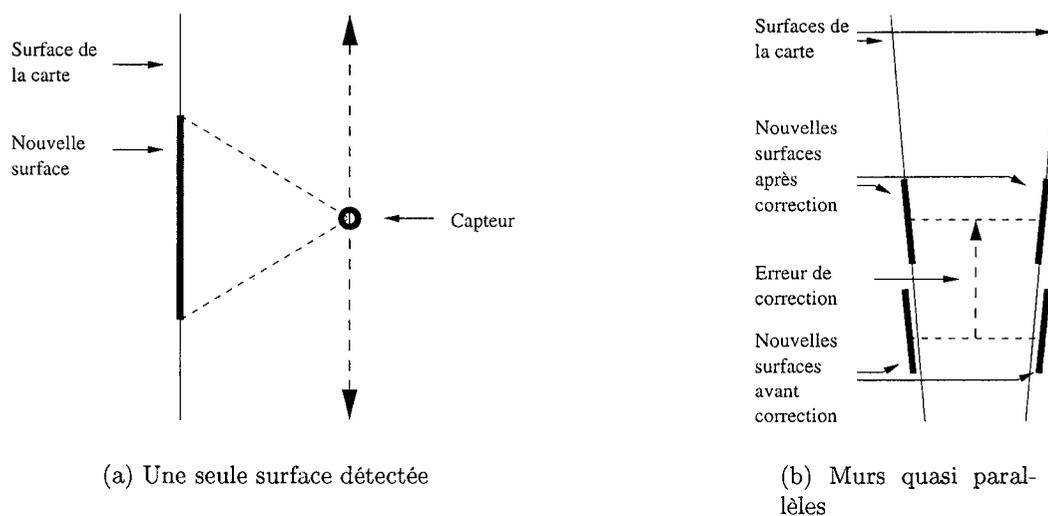


Figure 4.1 Problèmes pouvant survenir lors de la correction de la position de la plate-forme

qui arrive, par exemple, s'il y a une seule paire tel que le montre la figure 4.1 (a). Dans cette situation, il est impossible de déterminer le déplacement optimal dans une direction tangente au plan : le capteur peut être placé n'importe où le long de la double flèche pointillée de la figure 4.1 (a). Cette situation survient également si toutes les surfaces sont parallèles. Toutefois, des surfaces réellement parallèles, comme les murs d'un corridor, risquent d'être détectées comme quasi parallèles en raison des erreurs de mesure. Tel qu'illustré à la figure 4.1 (b), une petite erreur dans la position des surfaces détectées entraînera une erreur de déplacement importante lors de la correction de la position de la plate-forme. Dans ce cas, il est préférable de corriger la position uniquement dans la direction normale aux plans. Dans une telle situation, la matrice sera mal conditionnée. Nous pouvons détecter ces situations en évaluant les valeurs propres de la matrice  $2 \times 2$ .

Nous pouvons utiliser les valeurs propres de cette matrice pour déterminer si la translation obtenue par la résolution du système linéaire 4.7 est valide. Si les deux valeurs propres sont élevées, alors la matrice est bien conditionnée et nous acceptons

la translation obtenue sans modification. Si une des valeurs propres est trop faible, nous avons des surfaces quasi parallèles et nous effectuons la translation uniquement le long de l'axe obtenu en effectuant la moyenne pondérée par  $\rho_i$  des vecteurs normaux aux surfaces. Nous déterminons, arbitrairement, qu'une valeur propre est trop petite si elle est inférieure à 10 % de la somme des valeurs propres. En raison de la relation 4.8, qui est obtenue en calculant la trace de la matrice 2x2 à la gauche de l'équation 4.7 et la trace de la matrice composée des valeurs propres, et de la relation 4.9, qui est une conséquence d'avoir choisi  $a^2 + b^2 + c^2 = 1$  et d'avoir éliminé les plans parallèles au plancher, cette valeur de 10 % imposée aux valeurs propres signifie qu'il faut une contribution d'au moins 10 % des points de l'image dans chaque axe de déplacement.

$$\lambda_1 + \lambda_2 = \sum_i \rho_i a_i^2 + \sum_i \rho_i b_i^2 \quad (4.8)$$

$$\sum_i \rho_i a_i^2 + \sum_i \rho_i b_i^2 \approx \sum_i \rho_i \quad (4.9)$$

Dans le cas extrême où aucun appariement de surface n'est possible entre les surfaces de la carte et celles de la nouvelle acquisition, aucune correction ne peut être apportée à l'odométrie.

## 4.2 Fusion des surfaces

Le module de localisation décrit à la section 4.1 nous permet de déterminer la position des nouvelles surfaces dans la carte en construction. Il faut maintenant intégrer ces nouvelles surfaces à la structure de données qui contient la carte en construction. Pour les surfaces observées pour la première fois, cela ne pose aucun problème,

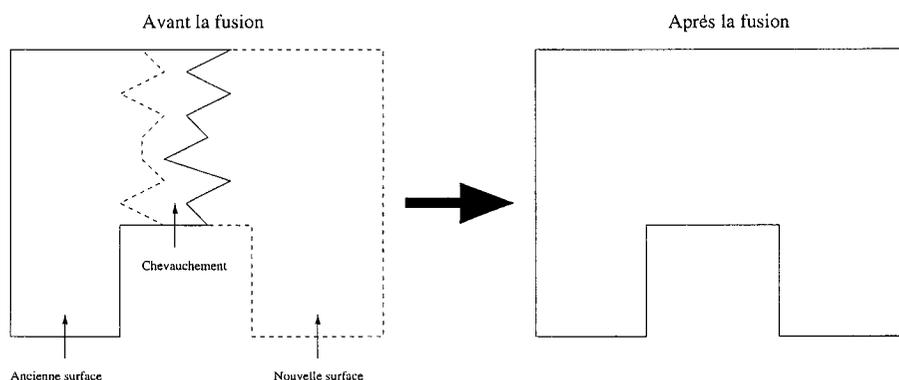


Figure 4.2 Fusion de deux surfaces planes

il s'agit de les ajouter à la liste des surfaces que contient la carte. Le problème est plus compliqué pour les surfaces déjà présentes dans la carte, car il faut alors fusionner les anciennes et les nouvelles surfaces. Cela est illustré schématiquement à la figure 4.2.

La méthode que nous utilisons est divisée en trois étapes :

1. Déterminer les éléments surfaciques pouvant être fusionnés ensemble. La méthode utilisée varie en fonction du type d'élément considéré.
2. Calculer les paramètres mathématiques du nouvel élément surfacique ainsi formé. Ces paramètres sont obtenus en prenant la moyenne des paramètres des éléments surfaciques avant la fusion. Les moyennes sont pondérées en fonction du nombre de points que représente chaque élément surfacique.
3. Calculer le contour du nouvel élément surfacique ainsi formé. Le traitement effectué dépend du type des éléments fusionnés.

Lorsque des éléments surfaciques sont fusionnés, les surfaces qui les contenaient sont également fusionnées en suivant la même technique que celle qui a été utilisée

pour les éléments surfaciques. Nous décrivons ci-dessous les particularités de chaque type de surface.

#### 4.2.1 Fusion des surfaces planes

Nous considérons que deux éléments surfaciques plans peuvent être fusionnés ensemble si les conditions suivantes sont respectées :

1. Ils sont de même couleur. Pour vérifier cela, nous utilisons la relation 4.1.
2. Ils ont la même orientation. Pour vérifier cela, nous utilisons la relation 4.2. Cette fois, nous fixons la valeur de  $\tau_{ang}$  en fonction de l'erreur d'orientation observée lors de la segmentation.
3. Ils sont proches l'un de l'autre. Pour vérifier cela, nous utilisons la relation 4.5. La distance obtenue doit être inférieure à une valeur  $\tau_{dist}$  fixée en fonction de l'erreur de distance du télémètre laser.

Si deux éléments surfaciques répondent à toutes ces conditions, ils sont quasi coplanaires. Nous les forçons à être coplanaires en les projetant sur le plan obtenu en prenant la moyenne pondérée des paramètres de chaque élément surfacique. Les contours des éléments surfaciques forment alors deux polygones coplanaires. Nous calculons l'union de ces polygones au moyen de l'algorithme de Vatti (Vatti, 1992) tel qu'implanté dans la librairie GPC (Murta, 2006) disponible sous license GPL.

#### 4.2.2 Fusion des surfaces cylindriques

Nous considérons que deux éléments surfaciques cylindriques peuvent être fusionnés ensemble si les conditions suivantes sont respectées :

1. Ils sont de même couleur. Pour vérifier cela, nous utilisons la relation 4.1.
2. Ils ont la même orientation. Pour vérifier cela, nous utilisons la relation 4.10 qui vérifie l'angle entre les vecteurs directeurs des axes des cylindres. Ces vecteurs sont obtenus à partir des paramètres des cylindres en utilisant l'équation III.11 présentée à l'annexe III. La valeur de  $\tau_{ang}$  est fixée en fonction de l'erreur d'orientation observée lors de la segmentation.

$$\arccos \frac{|\vec{v}_{C1} \cdot \vec{v}_{C2}|}{\|\vec{v}_{C1}\| \|\vec{v}_{C2}\|} \leq \tau_{ang} \quad (4.10)$$

3. Ils sont proches l'un de l'autre. Pour vérifier cela, nous calculons la distance entre les axes des cylindres auxquels appartiennent les surfaces<sup>2</sup>. Cette distance, ainsi que la différence entre les rayons des cylindres, doivent toutes deux être inférieures à la valeur  $\tau_{dist}$  qui est fixée en prenant compte de la précision du télémètre laser.

Pour réaliser la fusion des éléments surfaciques cylindriques, nous « déroulons » les cylindres pour obtenir des éléments surfaciques plans que nous fusionnons en utilisant l'algorithme de Vatti. Le résultat de la fusion est ensuite « enroulé » pour obtenir de nouveau un élément surfacique cylindrique. Les paramètres du nouveau cylindre sont obtenus en prenant les moyennes des paramètres des éléments surfaciques fusionnés.

### 4.2.3 Fusion des surfaces sphériques

Nous considérons que deux éléments surfaciques sphériques peuvent être fusionnés ensemble si les conditions suivantes sont respectées :

---

<sup>2</sup>Rappelons que la distance entre deux droites gauches dans  $\mathbb{R}^3$  est la plus petite distance possible entre un point de la première droite et un point de la seconde droite.

1. Ils sont de même couleur. Pour vérifier cela, nous utilisons la relation 4.1.
2. Ils sont proches l'un de l'autre. Pour vérifier cela, nous calculons la distance entre les centres des sphères. Cette distance et la différence entre les rayons des sphères doivent toutes deux être inférieures à la valeur  $\tau_{dist}$  qui est fixée en prenant compte de la précision du télémètre laser.

La fusion des éléments surfaciques sphériques est beaucoup plus difficile que la fusion des éléments surfaciques plans et cylindriques. De plus, les surfaces sphériques sont beaucoup moins présentes en milieu intérieur que les surfaces planes et cylindriques. Pour ces raisons, nous avons décidé de ne pas calculer le contour des surfaces sphériques fusionnées. Ainsi, lorsque deux surfaces sphériques sont fusionnées, nous considérons que le résultat est une sphère complète. Les paramètres de la sphère obtenue suite à la fusion sont calculés en effectuant la moyenne des paramètres des éléments surfaciques fusionnés.

Une approche qui pourrait être utilisée pour fusionner adéquatement les surfaces sphériques serait de projeter la sphère sur un cylindre qui l'englobe et qui est tangent à un grand cercle de la sphère. Il existe plusieurs manières de projeter une sphère sur un cylindre dont la projection de Mercator (Wikipédia, 2006). Les surfaces peuvent ainsi être fusionnées en utilisant la technique de fusion des surfaces cylindriques présentée à la section 4.2.2. Malheureusement, cette projection déforme les surfaces, surtout autour des points d'intersection de l'axe du cylindre et de la sphère. En effet, à ces points la projection n'est pas unique.

#### 4.2.4 Fusion d'une surface courbe et d'une surface plane

Un cas qui n'a pas encore été discuté est le cas où, dans l'une des deux images, une surface courbe a été segmentée en une surface plane. Cela survient quand l'information obtenue était insuffisante pour déterminer qu'il s'agissait d'une surface courbe. Il est donc nécessaire de passer en revue les surfaces planes qui n'ont pas été fusionnées à d'autres surfaces planes et de vérifier si elles peuvent se fusionner à une surface courbe. Nous considérons qu'un élément surfacique plan peut se fusionner à un élément surfacique courbe de même couleur si tous les points du contour de l'élément surfacique plan sont à une distance inférieure à  $\tau_{dist}$  de l'élément surfacique courbe. Pour la fusion, nous procédons de la même manière que pour la fusion de deux éléments surfaciques courbes identiques, mais nous conservons uniquement les paramètres de la surface courbe initiale.

#### 4.2.5 Fusion de deux éléments surfaciques courbes mal détectés

Le dernier cas qui peut survenir est celui où une surface courbe a été approximée par une surface plane à la fois dans la carte et dans la nouvelle acquisition. Dans le cas général, les surfaces ne seront pas parallèles et n'auront donc pas été fusionnées par la fusion des surfaces planes présentée à la section 4.2.1. Cette situation est illustrée à la figure 4.3.

Ce cas n'a pas été implémenté dans le cadre du présent travail. Une approche pour résoudre cette situation serait de réappliquer la technique de détection des surfaces courbes basée sur l'algorithme de Levenberg-Marquardt décrite au chapitre 3. Il faudrait appliquer cet algorithme chaque fois qu'une intersection est détectée entre une nouvelle surface plane et une surface plane déjà présente dans la carte. Toutefois, cette approche augmenterait de manière significative le temps de calcul et

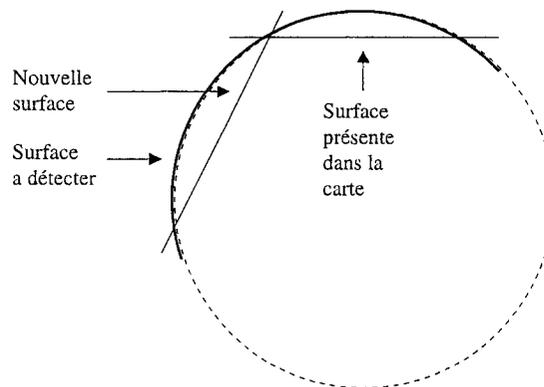


Figure 4.3 Exemple de surfaces courbes mal détectées

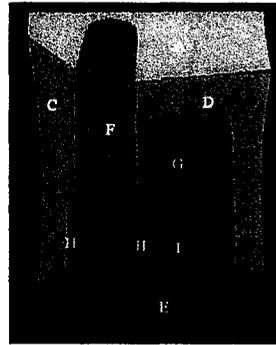
nécessiterait de conserver en mémoire toutes les données brutes.

#### 4.2.6 Cas général

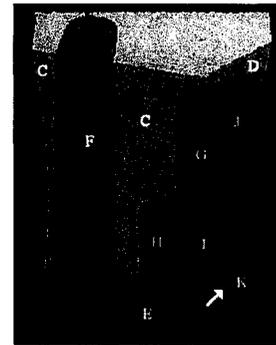
Si d'autres modèles de surfaces courbes sont utilisés, il est possible qu'il soit très difficile, voir même peut-être impossible, de continuer à utiliser l'algorithme de Vatti pour la fusion des surfaces et des éléments surfaciques. Une méthode générale qui pourrait être employée pour tous les types de surfaces consiste à utiliser des maillages pour identifier les portions des surfaces qui ont été détectées. Les maillages pourraient être fusionnés en utilisant les techniques de Sequeira et al. (Sequeira et al., 1998) ou de Ramalingam et Lodha (Ramalingam et Lodha, 2003).

#### 4.3 Exemple de résultat

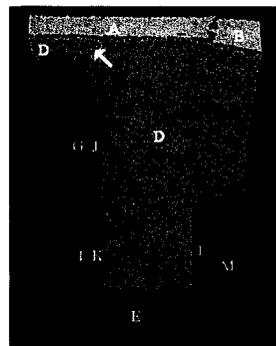
La figure 4.4, à la page 122, présente un exemple d'intégration de vues multiples réalisé au moyen de notre simulateur. Des exemples de scènes réelles seront présentés au chapitre 5. Les figures 4.4 (a) à (d) présentent les différentes acquisitions



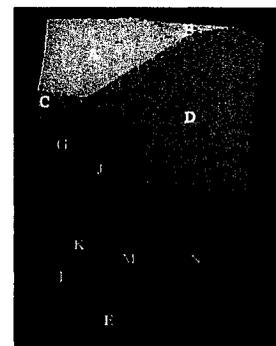
(a) Vue 1



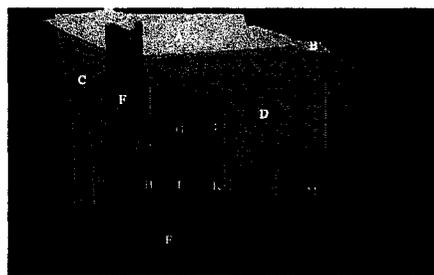
(b) Vue 2



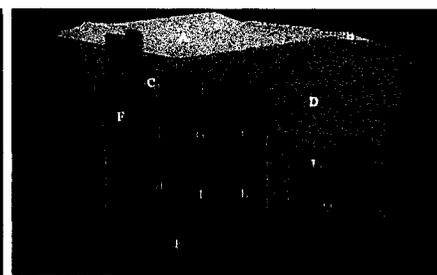
(c) Vue 3



(d) Vue 4



(e) Résultat (profil gauche)



(f) Résultat (centré)

Figure 4.4 Exemple d'intégration de vues multiples

qui ont été fusionnées pour obtenir la scène complète présentée en (e) et (f). Au total, 42 éléments surfaciques ont été détectés. Suite à la fusion, la carte créée en contient 14. Nous remarquons également que deux erreurs de segmentation, qui ont été mises en évidence par de petites flèches dans les figures 4.4 (b) et (c), ont été corrigées grâce à la fusion des vues multiples. Des résultats quantitatifs sont présentés au chapitre 5.

#### 4.4 Conclusion du chapitre de l'intégration de vues multiples

Nous avons présenté une technique de correction de la position de la plate-forme mobile dans un espace à deux dimensions. La technique développée compare les surfaces planes observées aux surfaces planes présentes dans la carte pour améliorer le positionnement fourni par l'odométrie de la plate-forme. L'utilisation des surfaces plutôt que des données brutes, rend la technique très rapide. Par exemple, pour ajouter une nouvelle acquisition comportant 14 éléments surfaciques au cas illustré à la figure 4.4 (e) et (f), le nombre de comparaisons pour trouver les points les plus proches en utilisant l'algorithme ICP classique devrait être de l'ordre de 1 000 000 alors que le nombre de comparaisons à effectuer avec notre algorithme est de l'ordre de 400. Il devrait donc y avoir un gain de performance de l'ordre de 250 000 %<sup>3</sup>. La localisation de la plate-forme est un bon exemple d'application dans laquelle la segmentation des données brutes en surfaces accélère la réalisation d'une tâche

---

<sup>3</sup>La carte de la figure 4.4 provient de 4 acquisitions 3D de 61 009 points chacune pour un total de 244 036 points. En supposant que la recherche du point le plus proche se fasse avec une complexité de  $\mathcal{O}(\log_2 n)$ , le nombre d'opérations requises pour trouver le point le plus proche d'un nouveau point est d'environ 18. Comme la nouvelle acquisition contient elle aussi 61 009 points, il faudra plus de 1 000 000 d'opérations pour trouver les plus proches voisins de tous les nouveaux points. La même carte comporte 14 surfaces. Si la nouvelle acquisition comporte elle aussi 14 surfaces, il faudra, dans le pire cas, réaliser  $14 \times 14 = 196$  vérifications pour apparier les surfaces. Chaque vérification demande deux opérations, l'une pour vérifier la limite d'angle et l'autre pour vérifier la limite de distance, pour un total d'environ 400 opérations. Ces chiffres sont approximatifs et ne tiennent pas compte de toutes les optimisations pouvant être effectuées. Ces valeurs sont valide pour une seule itération des algorithmes considérés.

importante en robotique mobile.

Nous avons ensuite montré comment l'algorithme de Vatti peut être utilisé pour la fusion des surfaces planes et cylindriques. Dans le cas des modèles de surfaces plus complexes qui pourraient être ajoutés à notre système de segmentation, et qui ne pourraient pas être facilement fusionnables par l'algorithme de Vatti, nous suggérons d'utiliser des techniques de fusion de maillages.

Plusieurs sujets pourraient faire l'objet de développements futurs. Premièrement, la technique de positionnement pourrait être généralisée à trois dimensions. En milieu intérieur, cela ne serait toutefois pas très utile, sauf dans des applications où le sol est jonché de débris et où la plate-forme ne se déplace plus toujours sur une surface plane.

Deuxièmement, la technique de positionnement pourrait inclure les surfaces courbes. Toutefois, il nous semble difficile de définir une technique générale qui pourrait fonctionner avec tous les types de surfaces (autre que considérer directement les données brutes). Puisque les surfaces planes sont toujours présentes en milieu structuré, nous suggérons de conserver la technique que nous avons présentée. Puisque les surfaces courbes sont moins nombreuses que les surfaces planes, elles pourraient être utilisées comme repères qui serviraient à obtenir une estimation de la position qui pourrait remplacer l'odométrie ou servir pour la fusion de cartes prises par plusieurs plate-formes mobiles.

Troisièmement, l'algorithme de Vatti pourrait être appliqué aux sphères. Une stratégie de gestion des points problématiques devra alors être développée.

Finalement des travaux pourraient porter sur la détection des surfaces courbes non détectées lors de la segmentation. Une méthode consisterait à détecter les intersections entre les surfaces planes et à appliquer la technique d'extraction des surfaces

courbes présentée au chapitre 3. Puisque le nombre de surfaces augmente généralement à chaque nouvelle acquisition<sup>4</sup>, cette recherche de surfaces courbes deviendra de plus en plus lente. Il faudrait donc développer une stratégie permettant de réduire au minimum les appels à l'algorithme de Levenberg-Marquardt.

---

<sup>4</sup>Nous supposons que la plate-forme explore un environnement inconnu.

## CHAPITRE 5

### RÉSULTATS ET ANALYSE

Nous avons d'abord décrit au chapitre 2 la conception d'un nouveau capteur permettant d'obtenir des images combinant télémétrie et photométrie. Puis, au chapitre 3, nous avons présenté une technique de segmentation qui utilise ces deux modalités sensorielles pour extraire de ces images les surfaces des objets composant la scène. Ensuite, nous avons développé, au chapitre 4, un algorithme de localisation et une technique de fusion de vues multiples qui utilisent les surfaces obtenues à la suite de la segmentation. Finalement, dans ce chapitre, nous allons présenter et analyser les résultats obtenus au moyen de notre capteur, de notre technique de segmentation et de nos algorithmes de localisation et de fusion de vues multiples.

#### 5.1 Caractéristiques du capteur

Cette section présente les résultats et performances de notre capteur 3D. D'abord, nous évaluerons la précision télémétrique de notre capteur, afin de pouvoir déterminer la valeur appropriée du paramètre de segmentation télémétrique  $d_{max}$ , qui représente la distance maximale permise entre un point et un plan pour que le point soit considéré comme faisant partie du plan. Ensuite, nous évaluerons la qualité de la registration entre les données télémétriques et photométriques, c'est-à-dire que nous vérifierons dans quelle mesure les bonnes couleurs sont associées aux points obtenus au moyen du télémètre laser.

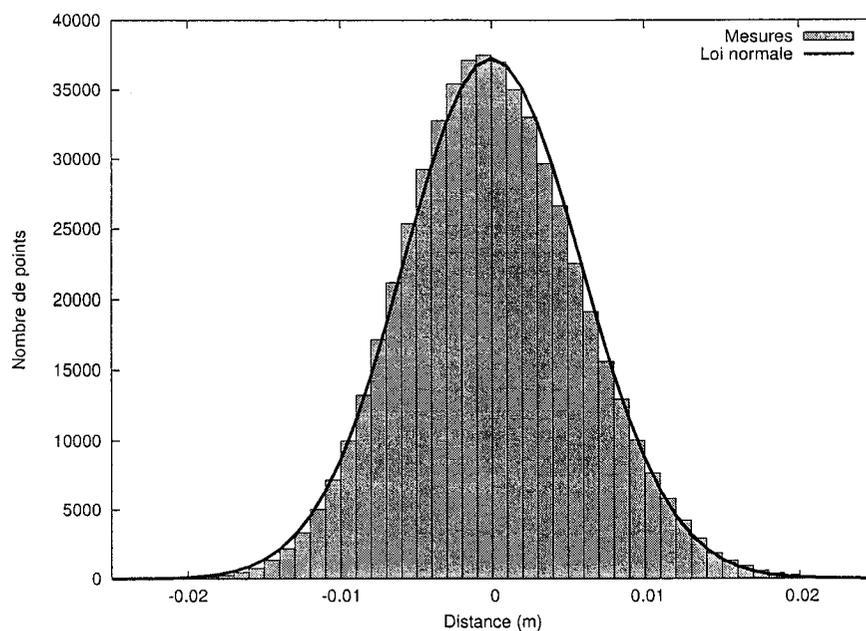


Figure 5.1 Distribution de l'erreur de distance

### 5.1.1 Précision télémétrique

Afin d'évaluer la précision des données télémétriques, nous avons placé notre télémètre laser en face d'une surface plane et avons pris plusieurs acquisitions en faisant varier la distance et l'orientation du capteur par rapport à celle-ci. Pour chaque acquisition, nous avons calculé l'équation du meilleur plan passant par le nuage de points en utilisant un algorithme de régression linéaire par calcul des moindres carrés. Le fait de numériser une seule surface plane élimine toute erreur potentielle due à une mauvaise segmentation. Finalement, nous évaluons les distances des points à la surface plane calculée. Le signe de la distance détermine de quel côté du plan se trouve le point. Nous obtenons ainsi la répartition des points de part et d'autre du plan. Les résultats cumulatifs sont présentés à la figure 5.1. Nous observons que les résultats obtenus suivent approximativement une loi normale de moyenne

nulle dont l'écart-type est de 5,9 mm. Par observation de la figure 5.1, nous pouvons déterminer le paramètre  $d_{max}$ . Nous utiliserons la valeur  $d_{max}=0,02$  m afin de contenir tous les points du plan lors de notre extraction de surfaces planes.

### 5.1.2 Précision de la registration

Pour évaluer la qualité de la registration, nous avons utilisé une scène composée d'un mur blanc et d'un panneau rose parallèle au mur et situé à un mètre de celui-ci. L'ensemble est ensuite numérisé plusieurs fois en faisant varier la distance et l'angle du capteur par rapport à la scène. Deux segmentations manuelles sont ensuite appliquées sur chaque image pour répartir les points en fonction de l'objet et de la couleur auxquels ils appartiennent. Puisque les deux plans auxquels appartiennent les surfaces sont parallèles, nous pouvons calculer manuellement l'équation d'un plan imaginaire passant entre ces deux plans. Ce plan sert de seuil pour la division spatiale des points : les points situés en avant de ce plan imaginaire appartiennent au panneau rose et les points situés derrière appartiennent au mur. Nous obtenons ainsi la première segmentation manuelle basée sur la télémétrie. La seconde segmentation manuelle est basée sur la photométrie et est effectuée en choisissant un seuil approprié sur la composante rouge des points. Dans le cas idéal, tous les points roses devraient être situés dans le plan le plus rapproché du capteur et tous les points blancs dans plan le plus éloigné. En comptant le nombre de points différents entre les deux segmentations, nous obtenons une bonne idée de la qualité de la registration.

Les résultats des essais effectués ont indiqué que sur un total de 209 392 points, 207 968 points avaient reçu une bonne couleur alors que 1 424 étaient d'une couleur erronée, donc mal enregistrés, ce qui donne une réussite de 99,32 %. La figure 5.2 présente un exemple typique d'acquisition. La figure 5.2 (a) présente le montage

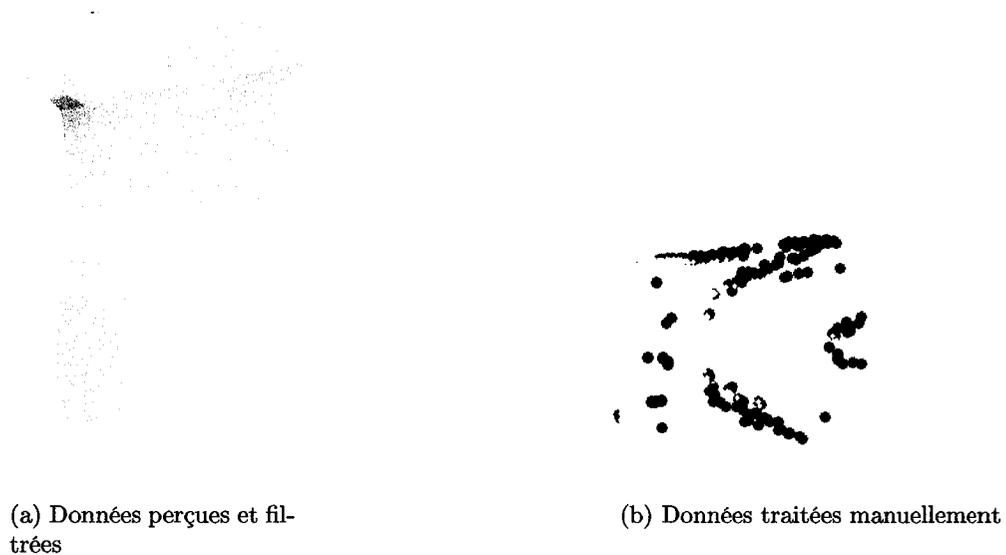


Figure 5.2 Répartition des erreurs de registration

réalisé tel que perçu par le capteur après filtrage des points fantômes<sup>1</sup>. Nous avons pris les données brutes, avant retrait des points fantômes, et avons appliqué les deux segmentations manuelles décrites au paragraphe précédent. La figure 5.2 (b) présente les résultats obtenus. Si les points appartiennent au plan du panneau rose et qu'ils sont effectivement roses, ou si les points appartiennent au mur blanc et qu'ils sont effectivement blancs, la registration est considérée correcte. Ces points sont affichés en gris sur la figure 5.2 (b). Dans les cas contraires, la registration est fautive et les points sont affichés en noir. Il est à noter que les points appartenant au plafond et au plancher ont été retirés pour ne garder que les deux plans parallèles. La figure 5.2 (b) est observée de profil pour mettre en évidence les points ayant subi une registration fautive. Nous remarquons, tel que prévu, que les points ayant subi une registration fautive se situent aux frontières entre les objets. Nous observons

<sup>1</sup>Ces points sont des artéfacts du télémètre laser (pour plus de détails, voir la section 2.5.2, page 62).



(a) Vue de face



(b) Vue de profil gauche



(c) Vue de profil droit

Figure 5.3 Image complexe bien enregistrée

également que les points dont la registration est fautive sont majoritairement des points fantômes dont la position est erronée. Il s'agit en fait des points fantômes créés par le télémètre laser dont il était fait mention à la section 2.5.2, page 62. Puisque la position de ces points est erronée, le calcul de projection pour obtenir la couleur est également erroné. Ces points ne sont pas affichés sur l'image 5.2 (a), car ils ont été éliminés au moyen du filtre de distance décrit à la section 3.1.1.3, page 75. Cette image démontre le bon fonctionnement du filtre proposé. À titre indicatif, l'image 5.2 (b) compte 176 points dont la registration est fautive. De ce nombre, 162 points sont des points fantômes. Le nombre de points bien positionnés ayant une mauvaise couleur est donc de 14. Le décalage maximal observé entre les données télémétriques et photométriques est de 1 pixel. La figure 5.3, à la page 130, présente une autre image prise au moyen de notre capteur. L'image est affichée sous trois angles différents pour permettre d'évaluer visuellement la qualité de la registration. Rappelons que les points cyan de l'image sont des points pour lesquels il n'y a pas d'information photométrique en raison des occlusions.

## 5.2 Résultats de segmentations

Cette section présente les résultats et performances de notre technique de segmentation. Dans un premier temps, la segmentation est appliquée à des images obtenues en utilisant le simulateur de notre capteur (voir section 2.6). L'usage du simulateur permet d'obtenir des images provenant d'un milieu connu ayant des caractéristiques bien définies. Ces images peuvent servir de référence fiable pour une analyse quantitative des résultats obtenus. Dans un second temps, notre technique de segmentation est appliquée à des images obtenues au moyen du vrai capteur dans un environnement réel. Ces scènes sont généralement plus complexes que les scènes simulées et nous ne connaissons pas avec autant de précision leurs caractéris-

tiques (dimensions, orientations et couleurs) exactes<sup>2</sup>. Notre analyse des résultats provenant d'images réelles sera qualitative.

Dans cette section, nous conservons l'emploi du terme « élément surfacique » pour identifier un élément surfacique réel, c'est-à-dire un élément surfacique devant être détecté, alors que nous employons le terme « segment » pour identifier un élément surfacique issu de la segmentation.

Au chapitre 3, nous avons établi qu'une bonne segmentation doit respecter les critères suivants :

**Critère 1 :** Détecter adéquatement les éléments surfaciques de taille supérieure à 100 points (0,16%) présents dans l'acquisition 3D. Une segmentation est adéquate si elle n'est ni excessive — trop de segments détectés pour un même élément surfacique réel — ni insuffisante — plusieurs éléments surfaciques détectés comme étant un seul segment.

**Critère 2 :** Respecter la géométrie spatiale. Les distances spatiales entre les points et les surfaces associées doivent être les plus faibles possible.

**Critère 3 :** Respecter la photométrie. Les différences de couleur entre les couleurs des points et celles des segments auxquels ils sont associés doivent être les plus faibles possible.

**Critère 4 :** La segmentation doit être robuste au bruit télémétrique et photométrique. Le nombre de fausses détections d'éléments surfaciques doit être faible.

**Critère 5 :** La segmentation photométrique doit dépendre le moins possible d'un éclairage non uniforme de la scène.

---

<sup>2</sup>L'instrument de mesure le plus précis que nous ayons pour mesurer ces valeurs est le capteur que nous avons conçu.

## 5.2.1 Segmentation d'images provenant d'une scène simulée

Nous commençons notre analyse par trois séries de simulations qui ont pour but d'évaluer les performances globales de la technique de segmentation. Une étude plus détaillée de certaines caractéristiques spécifiques est réalisée à la section 5.2.2, page 142. Nous présentons ici un résumé de l'étude réalisée. L'étude complète se trouve à l'annexe IV.

### 5.2.1.1 Métriques utilisées pour l'analyse

Afin d'analyser les résultats, nous utilisons la méthode proposée par Hoover et al. (Hoover et al., 1996). La méthode originale consiste à prendre des images réelles et à les segmenter de deux manières différentes. La première segmentation est effectuée à la main et est considérée comme la segmentation idéale, c'est-à-dire l'objectif à atteindre. La seconde segmentation est effectuée automatiquement par l'algorithme de segmentation analysé. Les deux segmentations sont ensuite comparées et 6 métriques sont calculées :

- Le nombre d'éléments surfaciques bien détectés ;
- Le nombre d'élément surfaciques excessivement segmentés ;
- Le nombre d'élément surfaciques insuffisamment segmentés ;
- Le nombre d'éléments surfaciques non détectés ;
- Le nombre de segments détectés en trop ;
- L'erreur d'orientation des segments détectés.

Ces six métriques permettront d'évaluer le critère numéro 1 — détecter adéquatement les éléments surfaciques de taille supérieure à 100 points (0,16%) présents dans l'acquisition 3D. Plutôt que de prendre des images réelles, nous utilisons des

images simulées. Le simulateur, en plus de nous fournir les images 3D, nous fournit également la segmentation de référence, ce qui retire le risque d'erreurs liées à une mauvaise segmentation manuelle de la part de l'utilisateur et diminue la subjectivité. Le simulateur nous fournit également l'orientation dans l'espace de toutes les surfaces. Nous pouvons ainsi mesurer directement l'erreur d'orientation entre les surfaces de la segmentation automatique et celles de la segmentation de référence. Nous utilisons cette métrique pour évaluer les erreurs d'orientation. Cette métrique est plus précise que l'évaluation de l'erreur d'orientation entre deux surfaces bien détectées, car elle élimine les cas où deux erreurs s'annulent mutuellement, ce qui arriverait si toute la scène était mal orientée.

Nous ajoutons à nos outils d'analyse deux métriques additionnelles :

- L'erreur spatiale : distance entre les points et les surfaces obtenues ;
- L'erreur photométrique : différence entre les couleurs des points et les couleurs des segments.

La première métrique consiste à calculer, en millimètres, l'erreur de distance de chaque point à la surface à laquelle il a été assigné. Elle permet d'évaluer le critère numéro 2 — les distances spatiales entre les points et les surfaces associées doivent être les plus faibles possible. La seconde métrique consiste à calculer l'erreur de distance euclidienne RVB de la couleur de chaque point à la couleur du segment auquel a été assigné le point. Chaque canal de couleur peut posséder une valeur entière dans l'intervalle  $[0, 255]$ , l'erreur sur la couleur peut donc prendre des valeurs discrètes dans l'intervalle  $[0, 255\sqrt{3}]$ . Cette métrique sert à évaluer le critère numéro 3 — les différences de couleur entre les couleurs des points et celles des segments auxquels ils sont associés doivent être les plus faibles possible.

### 5.2.1.2 Caractéristiques des simulations effectuées

Les erreurs obtenues à la suite de la segmentation peuvent provenir de trois sources :

**Scène** La scène elle-même peut être bruitée. Par exemple, les murs peuvent être légèrement bombés, il peut y avoir des aspérités, etc.

**Capteur** Les mesures fournies par le capteur possèdent une précision limitée et sont bruitées.

**Algorithme** L'algorithme de segmentation utilisé suppose une certaine connaissance du capteur et est prévu pour travailler avec des données bruitées. En rendant les algorithmes assez flexibles pour accepter ces données bruitées, nous ajoutons du fait même un risque d'erreur.

Dans toutes les simulations que nous avons effectuées, la scène est parfaite. C'est-à-dire qu'elle est constituée de sections de plans, de sphère et de cylindres parfaits. Les caractéristiques du capteur et de l'algorithme sont modifiées au cours de trois séries de simulations :

**Série 1** La première série de simulations est effectuée en considérant un capteur et un algorithme parfaits. L'objectif de cette série de simulations est de valider le fonctionnement de l'algorithme dans le cas idéal.

**Série 2** La deuxième série de simulation est effectuée avec un capteur parfait et un algorithme qui prend pour acquis que le capteur est imparfait. L'objectif de cette série de simulations est de déterminer l'effet des paramètres utilisés par l'algorithme sur les performances obtenues dans un cas où le capteur est idéal.

**Série 3** La troisième série de simulations est effectuée avec une lumière ambiante et dirigée, un capteur bruité et un algorithme qui prend pour acquis que

le capteur est imparfait. Il s'agit du mode de simulation le plus proche de la situation réelle. Il faut noter que nous avons simulé un bruit de mesure beaucoup plus intense que celui retrouvé en réalité. Cette série de simulations permet d'évaluer le critère numéro 4 — la segmentation doit être robuste au bruit télémétrique et photométrique et le nombre de fausses détections d'éléments surfaciques doit être faible.

### 5.2.1.3 Résumé des résultats obtenus

Les résultats des ces simulations sont présentés en détails à l'annexe IV. Nous reprenons ici les principaux résultats obtenus.

Si nous considérons uniquement les éléments surfaciques de taille supérieure à 100 points, nous obtenons les résultats fournis au tableau 5.1. La similitude entre les deux dernières colonnes du tableau 5.1, ainsi que les autres résultats obtenus et présentés à l'annexe IV, montrent que le résultat de la segmentation reste pratiquement inchangé sous l'effet du bruit et de la lumière. Le seul impact du bruit de mesure est l'augmentation du nombre de segments détectés en trop. En effet, pour les séries 1 et 2, le nombre d'éléments surfaciques non détectés est sensiblement égal au nombre de segments détectés en trop. Cela s'explique par le fait que plusieurs régions en trop sont en fait les régions non détectées, mais pour lesquelles la proportion de points correctement assignés est trop faible. Ces surfaces sont alors considérées comme étant de nouvelles surfaces. Pour la série 3, nous avons une plus grande proportion de régions en trop en raison de l'erreur que nous avons ajoutée sur le capteur.

	Série 1	Série 2	Série 3
Bonnes détections	98,0 %	93,6 %	93,4 %
Mauvaises détections	2,0 %	6,4 %	6,6 %
Éléments surfaciques en trop	0,8 %	5,4 %	12,0 %

Tableau 5.1 Taux de réussite pour les éléments surfaciques de plus de 100 points

Métrique	Série 1	Série 2	Série 3
Erreur spatiale (mm)	0,042 (0,231)	1,80 (4,11)	11,12 (9,00)
Erreur photométrique	1,12 (8,79)	1,36 (11,05)	5,81 (10,44)
Erreur angulaire (°)	0,005 (0,039)	0,30 (0,89)	0,99 (1,83)

Tableau 5.2 Erreurs moyennes et écarts-types obtenus à la suite des segmentations

Les valeurs entre parenthèses sont les écarts-types.

Pour les surfaces bien détectées, nous avons évalué les métriques d'erreur suivantes :

- Erreur spatiale ;
- Erreur photométrique ;
- Erreur angulaire.

Nous avons calculé les valeurs moyennes et les écarts-types que nous avons affichés dans le tableau 5.2.

L'erreur moyenne en distance, demeure toujours inférieure à la limite  $d_{max}$ <sup>3</sup> imposée et l'erreur moyenne d'angle nous semble très acceptable. Pour la série 3, il convient d'indiquer qu'une segmentation parfaite ne donnerait pas une erreur nulle, puisque les données sont bruitées. L'erreur photométrique obtenue est également excellente. En effet, la distance moyenne observée de 1,36 est pratiquement imperceptible à

---

<sup>3</sup>Rappel :  $d_{max}$  est la distance maximale tolérée entre un point et une surface pour qu'un point soit considéré comme faisant partie de la surface.

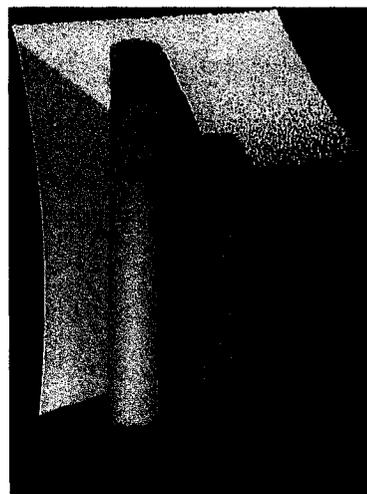
l'oeil alors que celle de 5,81 serait à peine perceptible pour un observateur averti.

#### 5.2.1.4 Exemples visuels

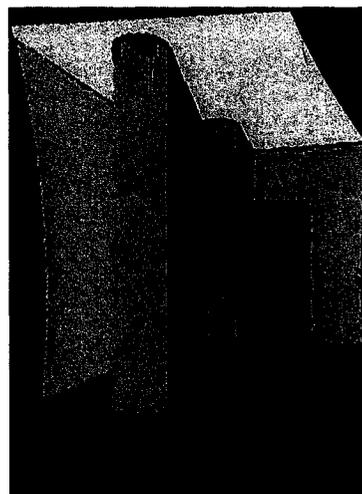
Nous présentons ici deux exemples visuels de segmentations réussies et un exemple de segmentation mal réussie. Les figures 5.4, page 139, et 5.5, page 140, présentent des exemples de segmentations bien réussies alors que la figure 5.6, page 141, présente la segmentation la moins bien réussie lors de nos simulations. Plusieurs erreurs sont en effet présentes dans la segmentation de la figure 5.6 :

- Une portion de chaise n'a pas été détectée (voir flèche no. 1). La taille de la surface était trop petite, principalement à cause du fait que la surface est quasi parallèle à la direction de vision du capteur.
- Il y a une mauvaise segmentation télémétrique du pied de la table en avant-plan (voir flèche no. 2). En effet, une portion du haut de la table se trouvait dans le même plan qu'une portion du pied de la table. Il s'agit d'un bon exemple de points pouvant appartenir à plus d'une surface à la fois (voir section 3.3, page 92). Malheureusement, dans ce cas la photométrie a été incapable de désambiguïser la situation, car le pied de la table était de la même couleur que le reste de la table. Cette erreur est aussi la cause d'une segmentation excessive du pied de la table (voir flèche no. 4) et de l'incapacité d'en détecter la forme cylindrique.
- Le plancher est excessivement segmenté (voir flèche no. 3). Cette segmentation excessive est due à un mauvais choix aléatoire des triplets de points utilisés pour détecter le plancher.

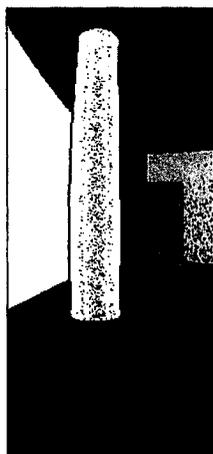
Malgré ces quelques problèmes, tous de nature télémétrique, nous notons que le résultat produit demeure cohérent au niveau de la position spatiale et de la couleur des éléments.



(a) Données brutes



(b) Résultat 3D



(c) Segmentation télémétrique

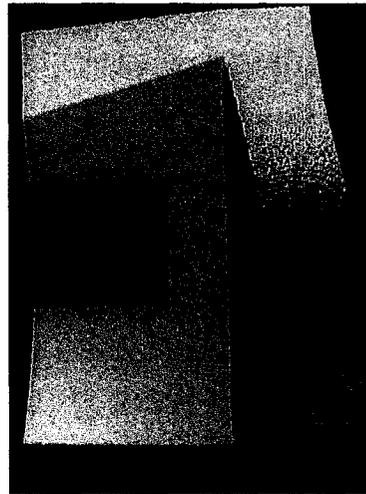


(d) Segmentation photométrique

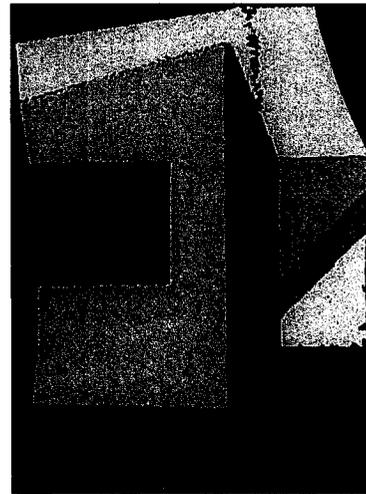


(e) Fusion

Figure 5.4 Exemple d'image simulée bien segmentée



(a) Données brutes



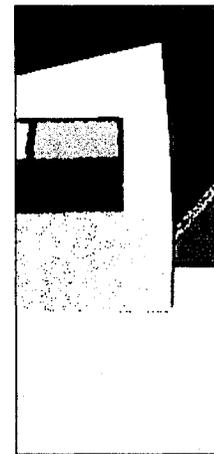
(b) Résultat 3D



(c) Segmentation télémétrique

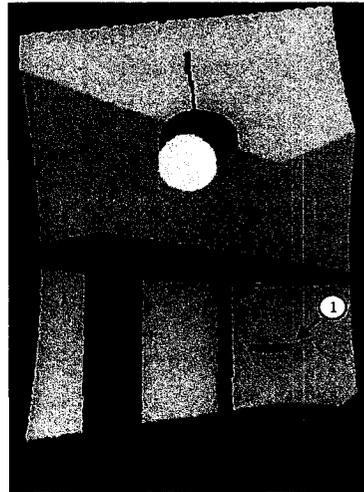


(d) Segmentation photométrique

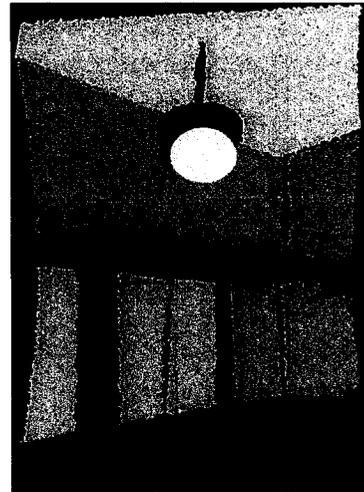


(e) Fusion

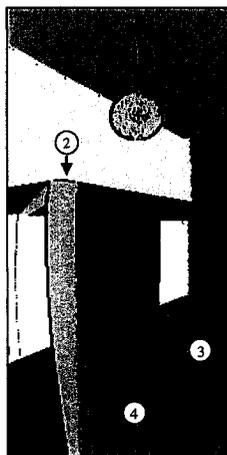
Figure 5.5 Autre exemple d'image simulée bien segmentée



(a) Données brutes



(b) Résultat 3D

(c) Segmentation  
télé-  
métrique(d) Seg-  
mentation  
photométrique

(e) Fusion

Figure 5.6 Exemple d'image simulée mal segmentée

### 5.2.2 Simulations spécifiques

Les trois séries de simulations présentées à la section 5.2.1 permettent d’avoir une bonne appréciation des performances de la méthode de segmentation proposée dans un environnement réaliste. Toutefois, la scène utilisée pour les simulations précédentes contenait une faible variété de surface cylindriques, de surfaces sphériques et de couleurs. Nous présentons à l’annexe V des simulations supplémentaires effectuées dans le but de caractériser plus précisément les performances de notre système de segmentation pour ces trois éléments. Plus précisément, nous avons caractérisé les aspects suivants :

**Simulation 1** Détection des surfaces cylindriques et sphériques en fonction de leur rayon ;

**Simulation 2** Détection des surfaces cylindriques et sphériques en fonction de leur distance au capteur ;

**Simulation 3** Détection des surfaces cylindriques en fonction de leur orientation par rapport au capteur ;

**Simulation 4** Détection des éléments surfaciques en fonction de la différence entre leurs couleurs ;

**Simulation 5** Détection d’un élément surfacique éclairé de manière non uniforme ;

**Simulation 6** Seuil photométrique de détection des éléments surfaciques en fonction du niveau de lumière.

Les deux derniers points permettent d’évaluer le critère numéro 5 — la segmentation photométrique doit dépendre le moins possible d’un éclairage non uniforme de la scène.

Nous reportons ici les conclusions obtenues à partir de ces simulations. Pour plus de détails, le lecteur est invité à consulter l'annexe V.

1. La première simulation montre que les surfaces sphériques sont bien détectées dans toute la plage étudiée à partir de 15 cm de rayon. Les surfaces cylindriques sont bien détectées avec une erreur inférieure à 2 % jusqu'à un rayon de 2,9 m. L'erreur grimpe ensuite jusqu'à atteindre 12 % à un rayon de 4,7 m. Après 6 m, la détection des surfaces cylindriques devient erratique. Cela est dû au fait que la courbure des cylindres est détectable dans un seul axe de l'image alors que la courbure des sphères est détectable dans les deux axes. L'explication détaillée du phénomène est donnée à l'annexe V.
2. La deuxième simulation montre que les surfaces cylindriques sont mieux détectées à grande distance que les surfaces sphériques. Les surfaces cylindriques ont été détectées jusqu'à une distance de 6,7 m alors que cette distance était de 4,3 m pour les surfaces sphériques. Cette différence s'explique par le fait que les surfaces sphériques comportaient en général moins de points que les surfaces cylindriques à cause de leur forme. Nous en venons à la conclusion (évidente) que les objets de plus petite taille sont moins bien détectés à grande distance. Dans les limites où les objets sont correctement détectés, la distance n'a pas d'effet important sur le rayon des surfaces obtenues.
3. La troisième simulation montre que les surfaces cylindriques sont bien détectées, peu importe leur orientation par rapport au capteur.
4. La quatrième simulation indique que deux éléments surfaciques sont bien détectés au moyen de la photométrie si la moyenne des différences de chaque canal de couleur, rouge, vert et bleu, est supérieure à une valeur de 8 sur l'échelle RVB utilisée. Cette valeur est très proche du seuil  $S_P$  — seuil appliqué à l'image de contours avant d'effectuer la ligne de partage des eaux — que nous avons fixé à 6. La différence s'explique par l'application du filtre gaus-

sien. Ceci montre que la segmentation photométrique peut être très sélective si la différence photométrique s'applique sur les trois canaux à la fois.

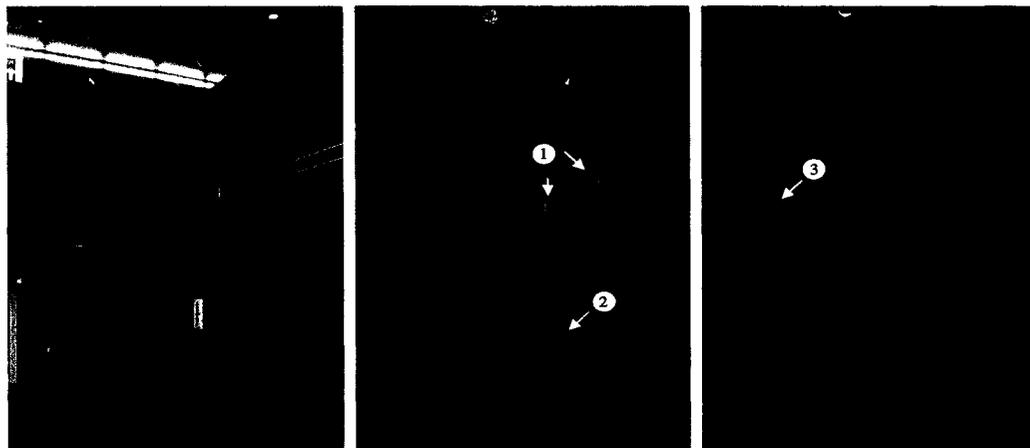
5. Les résultats de la cinquième simulation démontrent la robustesse de notre système par rapport à l'éclairage : l'intensité lumineuse appliquée à une seule surface n'a aucun effet sur la bonne détection de cette surface.
6. La sixième simulation confirme la robustesse du système par rapport à l'éclairage : la position des contours des surfaces n'est pas affectée par l'éclairage.

### 5.2.3 Résultats provenant d'images réelles

Les résultats et conclusions inscrits aux sections 5.2.1 et 5.2.2 ainsi qu'aux annexes IV et V ont fourni beaucoup d'informations quantitatives sur la technique de segmentation que nous avons développée. Ces informations ont été obtenues au moyen de simulations et il est maintenant temps de vérifier les performances de la segmentation sur des images réelles. Nous présentons ici quatre exemples typiques de scènes numérisées et segmentées.

#### Grande salle sur deux étages (figure 5.7, page 145)

Cette image a été prise au premier étage du pavillon Lassonde de l'École Polytechnique de Montréal, près des ascenseurs. Cette vue s'étend sur deux étages. Au deuxième étage se trouve une balustrade. Le plafond observé dans le haut de l'image est celui du deuxième étage. Il s'agit d'une image difficile à segmenter par la photométrie, car toutes les surfaces sont de la même couleur. Dans ce cas, les démarcations photométriques observées entre les surfaces sont dues à l'éclairage. Une inspection des images révèle que toutes les surfaces ont été correctement détectées, mis à part une segmentation excessive du mur gauche de la balustrade (voir flèche



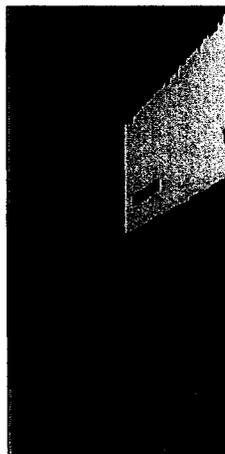
(a) Scène observée

(b) Données brutes

(c) Résultat 3D



(d) Segmentation télémétrique



(e) Segmentation photométrique



(f) Fusion

Figure 5.7 Segmentation de l'image réelle « Grande salle sur deux étages »

no. 3). La segmentation excessive apparente du plafond (voir flèche no. 4) n'est pas une erreur de l'algorithme, il existe bel et bien une coupure dans le plafond à cet endroit. Malheureusement, la majorité des éléments surfaciques situés à l'intérieur des surfaces (voir flèches nos. 1 et 2) n'ont pas été détectés, car ils étaient constitués d'une multitude de petites surfaces. Il s'agit là d'un problème rencontré avec les objets métalliques qui reflètent leur environnement au lieu d'avoir leur propre couleur.

### **Colonne et salle de cours (figure 5.8, page 147)**

Cette image a été prise au deuxième étage du pavillon Lassonde. Il s'agit d'une colonne située à côté d'une porte menant à une salle de cours. Entre la colonne et la porte, une vitre permet de voir l'intérieur du local. Les résultats de la segmentation montrent une très bonne segmentation télémétrique, mais une segmentation photométrique moins réussie. En effet, toutes les surfaces sont bien détectées, nous obtenons même un beau spécimen de colonne. La segmentation photométrique a beaucoup de difficultés à différencier les murs rouges des murs orange. Toutefois, cela a très peu de conséquences sur la segmentation finale, mis à part une segmentation excessive à gauche de l'image (voir flèche no. 1). Ce problème est dû en grande partie au mauvais éclairage de la scène. En effet, peu de lumière parvient à la partie renfoncée où se trouve la colonne. Nous notons également que le plafond est étrangement segmenté, ce qui est dû à la présence d'un grillage qui protège des tuyaux, câbles réseau et conduits de ventilation.

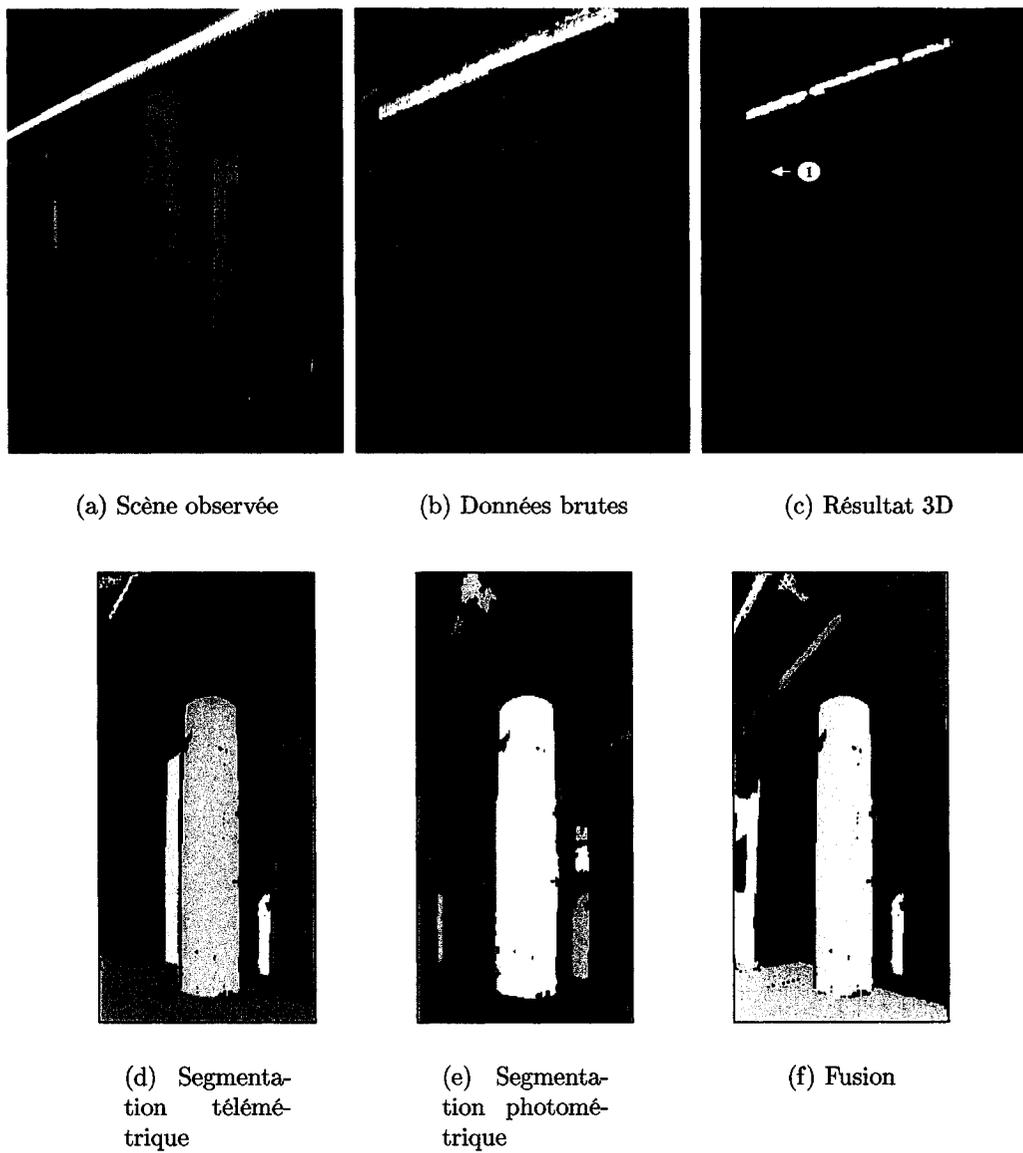
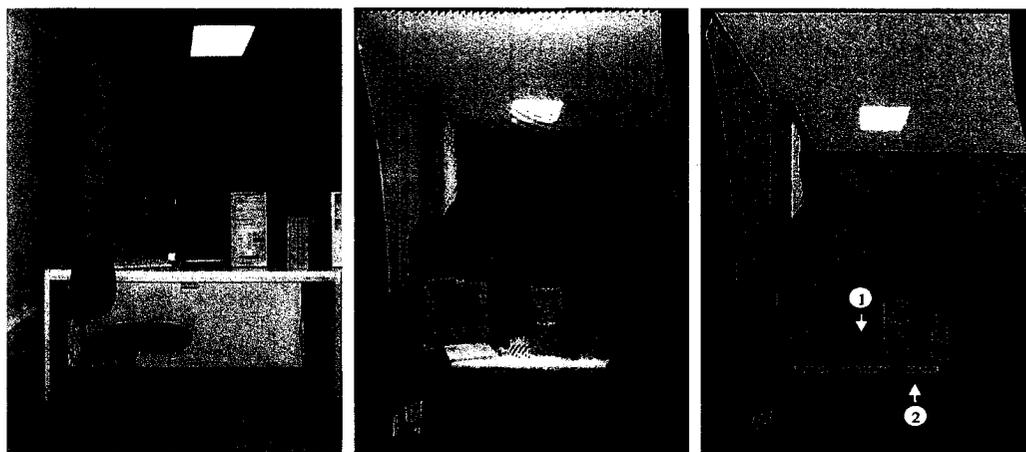


Figure 5.8 Segmentation de l'image réelle « Colonne et salle de cours »



(a) Scène observée

(b) Données brutes

(c) Résultat 3D



(d) Segmentation télémétrique



(e) Segmentation photométrique



(f) Fusion

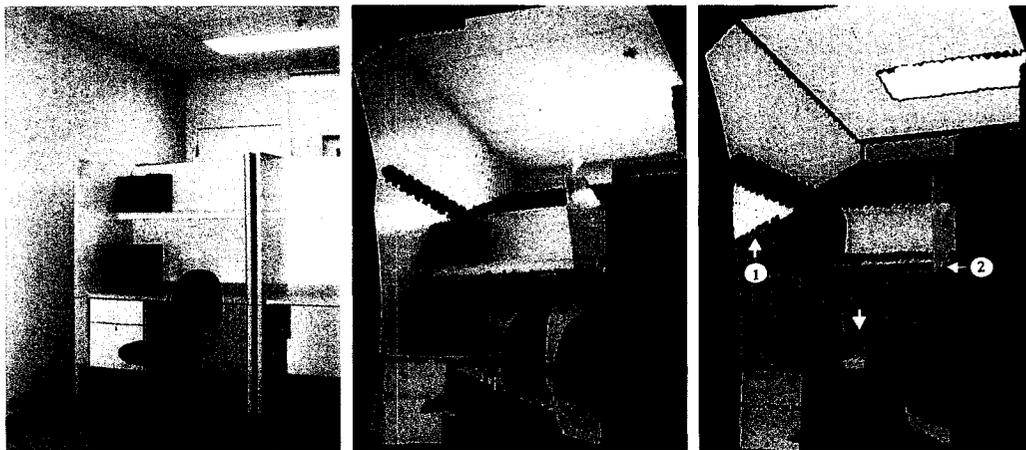
Figure 5.9 Segmentation de l'image réelle « Poste de travail 1 »

**Poste de travail 1 (figure 5.9, page 148)**

Cette image a été prise au local L5901 du pavillon Lassonde. Il s'agit d'une scène typique que l'on retrouve dans les bureaux. Il y a peu de choses à dire sur les résultats obtenus qui sont assez fidèles à la scène observée. Notons qu'il y a segmentation excessive du dessous de la table (voir flèche no. 2) causée par les points cyan de l'image. Ces points correspondent à des zones ayant subi une occlusion et ayant été vues uniquement par le télémètre laser. Aucune couleur n'est associée à ces points. Nous avons préféré conserver ces points, car ils sont utiles à la reconstruction géométrique de la scène. Toutefois, nous attribuons à ces éléments surfaciques un drapeau qui indique qu'ils ont une mauvaise couleur et qu'ils devront être éliminés lorsque de nouvelles données seront disponibles. Nous remarquons que beaucoup de points situés sur le dessus de la table ont été rejetés par l'algorithme de segmentation (voir flèche no. 1). Il s'agit d'un autre cas de rejet des surfaces de trop faible taille.

**Poste de travail 2 (figure 5.10, page 150)**

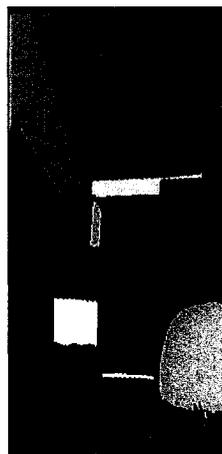
Cette image a été prise elle aussi au local L5901 du pavillon Lassonde. Il s'agit d'une aire de travail cloisonnée sur laquelle repose un ordinateur portable. Nous remarquons une segmentation photométrique excessive des cloisons droites et gauches de l'isoloir (voir flèches nos. 1 et 2). Cette segmentation excessive est due à des reflets lumineux très localisés. Il est étrange de voir que les simulations montraient une bonne tolérance de la technique de segmentation vis-à-vis de ce genre de situation alors que nous obtenons cette segmentation excessive en pratique. Toutefois, le modèle lumineux utilisé en simulation était très rudimentaire et ne tenait pas compte de la brillance des objets ni des ombres causées par les autres surfaces.



(a) Scène observée

(b) Données brutes

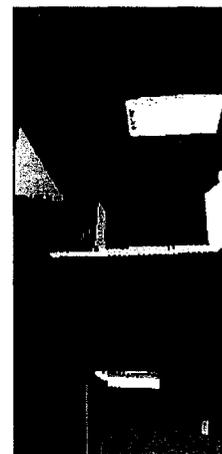
(c) Résultat 3D



(d) Segmentation télémétrique



(e) Segmentation photométrique



(f) Fusion

Figure 5.10 Segmentation de l'image réelle « Poste de travail 2 »

### 5.2.4 Conclusion des résultats de notre technique de segmentation

Les simulations effectuées nous permettent d'affirmer que notre algorithme de segmentation répond favorablement à tous les critères que nous avons établis :

**Critère 1 :** *Détecter adéquatement les éléments surfaciques de taille supérieure à 100 points (0,16%) présents dans l'acquisition 3D.* Les simulations ont montré que le taux de bonnes détections de notre algorithme est supérieur à 93 %.

**Critère 2 :** *Les distances spatiales entre les points et les surfaces associées doivent être les plus faibles possible.* Les simulations ont montré que l'erreur de distance moyenne entre les points et les surfaces est de l'ordre de 1,1 cm lorsque le bruit appliqué au capteur est de  $\pm 3$  cm. Nous avons montré à l'annexe IV que cette erreur n'est pas induite par notre algorithme de segmentation, car elle suit précisément l'erreur télémétrique ajoutée au capteur.

**Critère 3 :** *Les différences de couleur entre les couleurs des points et celles des segments auxquels ils sont associés doivent être les plus faibles possible.* Les résultats obtenus suite aux simulations ont montré que l'erreur photométrique obtenue, inférieure à 6 unités dans l'échelle RVB, est pratiquement imperceptible à l'œil.

**Critère 4 :** *La segmentation doit être robuste au bruit télémétrique et photométrique. Le nombre de fausses détections d'éléments surfaciques doit être faible.* Nous avons ajouté un bruit télémétrique au moins deux fois supérieur à celui observé en réalité et avons ajouté de la lumière dans notre troisième série de simulations. Nous avons montré que cela n'altère pratiquement pas les résultats de la segmentation puisque le taux de bonnes détections varie seulement de 0,2 %, passant de 93,6 % à 93,4 %. Le taux de fausses détections est cependant assez élevé avec une valeur supérieure à 5 % dans le cas non bruité et à 12 % dans le cas bruité.

**Critère 5 :** *La segmentation photométrique doit dépendre le moins possible d'un éclairage non uniforme de la scène.* Nous avons analysé l'effet de la lumière sur les performances de la segmentation photométrique et avons montré que la détection des éléments surfaciques n'est pas affecté par l'intensité de l'éclairage.

Les segmentations effectuées sur des images réelles donnent, en général, de bons résultats. Les surfaces télémétriques sont très bien segmentées et les volumes sont bien conservés. La segmentation photométrique donne des résultats acceptables, mais pose tout de même quelques difficultés. Les recherches futures devraient se diriger vers l'amélioration notamment de deux éléments principaux :

**Les petites surfaces** Il serait utile de concevoir un système permettant de réduire le rejet des petites surfaces. Une manière naïve d'y arriver serait de réduire la valeur de  $N_{min}$ , le nombre de points minimal que doit posséder une surface pour être conservée. Toutefois cela causerait une hausse importante du nombre de surfaces en trop. Une meilleure méthode serait de fusionner entre elles les petites régions ayant des caractéristiques photométriques communes. Pour cela, il faudrait établir une règle et un ordre de fusion.

**L'éclairage** La segmentation photométrique est encore assez sensible aux variations brusques de l'éclairage, que ce soit à cause des ombres ou d'une surface trop brillante. Pour réduire cet effet, il serait intéressant d'étudier l'usage d'autres espaces de couleur comme le HSV. Une autre solution serait d'incorporer une lumière au capteur. Nous aurions ainsi un meilleur contrôle de l'éclairage de la scène.

La fusion des segmentations télémétriques et photométriques fonctionne de manière impeccable. Le fait de séparer les deux segmentations permettra d'améliorer

facilement le système lorsque de nouvelles innovations seront faites dans l'un des deux domaines de segmentation.

Il découle de ces résultats que notre méthode de segmentation permet de préserver fidèlement la géométrie et la photométrie de la scène tout en réduisant massivement la quantité d'information. À titre d'exemple, la série de simulations 3 comportait 60 images de 61 009 points totalisant 3 660 540 points (99 Mo, binaire). Le résultat des segmentations a produit 488 segments (14 Mo, texte), soit une compression de 99,99 % du nombre d'éléments et de plus de 85 % d'espace.

### 5.3 Résultats d'intégrations de vues multiples

Nous nous intéressons maintenant aux résultats de l'intégration de vues multiples. Dans un premier temps, nous présenterons un résultat de simulation puis nous exposerons quelques résultats obtenus à partir d'images réelles.

Les modules de localisation et de fusion des surfaces utilisent trois paramètres pour apparier les éléments surfaciques. Le paramètre  $\tau_{coul}$  représente l'angle maximal toléré entre les vecteurs obtenus en représentant les couleurs dans  $\mathbb{N}^3$ . Le paramètre  $\tau_{ang}$  représente l'angle maximal toléré entre deux surfaces planes ou cylindriques. Enfin,  $\tau_{dist}$  représente la distance maximale tolérée entre deux surfaces de même type. Pour le module de localisation, les valeurs de  $\tau_{ang}$  et  $\tau_{dist}$  peuvent être déterminées en fonction de la précision de l'odométrie de la plate-forme mobile. Pour le module de fusion des surfaces, ces valeurs peuvent être déterminées en fonction de la précision de positionnement et d'angle des surfaces obtenues à la suite de la segmentation. Pour obtenir les résultats présentés dans ce chapitre, nous avons utilisé les valeurs présentées au tableau 5.3. Nous avons trouvé expérimentalement que les valeurs choisies donnent de bon résultats.

Paramètre	Valeur utilisée pour la localisation	Valeur utilisée pour la fusion
$\tau_{coul}$	8°	8°
$\tau_{ang}$	10°	5°
$\tau_{dist}$	0,3 m	0,10 m

Tableau 5.3 Valeurs des paramètres utilisés pour la localisation et la fusion des surfaces

### 5.3.1 Résultats de simulations

Afin de valider notre système de fusion des vues multiples, nous utilisons une fois de plus notre simulateur, car il nous permet de connaître avec exactitude les dimensions et positions des surfaces numérisées ainsi que la position exacte de la plate-forme mobile. La position de la plate-forme est transmise à un module qui y ajoute à chaque déplacement une erreur constante et une erreur qui varie aléatoirement de manière uniforme à chaque nouvelle mesure. Cette position erronée est utilisée et corrigée par l'algorithme de localisation de la plate-forme.

Afin de déterminer l'exactitude des résultats obtenus, nous avons effectué une coupe horizontale de la scène simulée et de sa reconstruction. Nous avons superposé ces deux coupes pour obtenir la figure 5.11. Les traits pleins visibles en périphérie de l'image sont les surfaces simulées. Ces surfaces de référence sont constituées de quatre murs, de deux boîtes collées côte à côte dans le coin supérieur droit, d'une boîte au centre à droite de l'image et d'une colonne dans la partie inférieure gauche. Les surfaces obtenues suite à l'intégration de dix acquisitions sont affichées en traits pointillés. Nous avons également superposé à ce graphique la trajectoire réelle suivie par la plate-forme mobile ainsi que les trajectoires additionnées d'erreur corrigées et non-corrigées. Finalement, les flèches présentent les points de vue d'où les dix

acquisitions 3D ont été prises : nous y trouvons les points de vue réels et entachés d'erreur avant et après correction.

Au début de la simulation, la plate-forme est située dans le coin supérieur gauche à la position (0,0) et prend deux acquisitions. À ce moment, la plate-forme mobile n'a pas encore accumulé d'erreur de positionnement de sorte qu'aucune correction n'est apportée. La plate-forme se déplace ensuite pour contourner la colonne et se rend à la position (1,-4). L'erreur s'accumule progressivement pour atteindre 9,4 cm au moment où les 4 acquisitions suivantes sont prises. Ces acquisitions permettent de corriger le positionnement de la plate-forme. Celle-ci poursuit ensuite son déplacement en ligne presque droite pour prendre 4 acquisitions supplémentaires. Une autre correction du positionnement est visible à la fin du parcours. L'observation des flèches à la fin du parcours nous indique que sans les corrections de l'odométrie, il y aurait eu une erreur de près de 30 cm entre la position mesurée par l'odométrie et la position réelle de la plate-forme. Les valeurs exactes des erreurs de positionnement avant et après correction sont présentées au tableau 5.4. Ces résultats montrent que le module de localisation développé exécute correctement sa tâche, en maintenant généralement l'erreur inférieure à 2 cm et à 0,5°. Toutefois deux situations problématiques sont survenues dont l'une a fait monter l'erreur angulaire à plus de 3° et dont l'autre a créé une erreur de plus de 7 cm. Ces deux situations montrent que lorsqu'il y a peu de surfaces communes entre la carte et la nouvelle acquisition, le système est très sensible à une erreur d'orientation ou de position des surfaces.

La superposition quasi parfaite sur la figure 5.11 des traits pleins correspondant aux surfaces réelles et des traits pointillés correspondant aux surfaces obtenues montre que les surfaces obtenues sont toutes bien positionnées et que les dimensions et orientations sont conservées. Pour plus de précision, nous avons reporté les mesures attendues et obtenues pour chaque surface au tableau 5.5. Les dimensions des

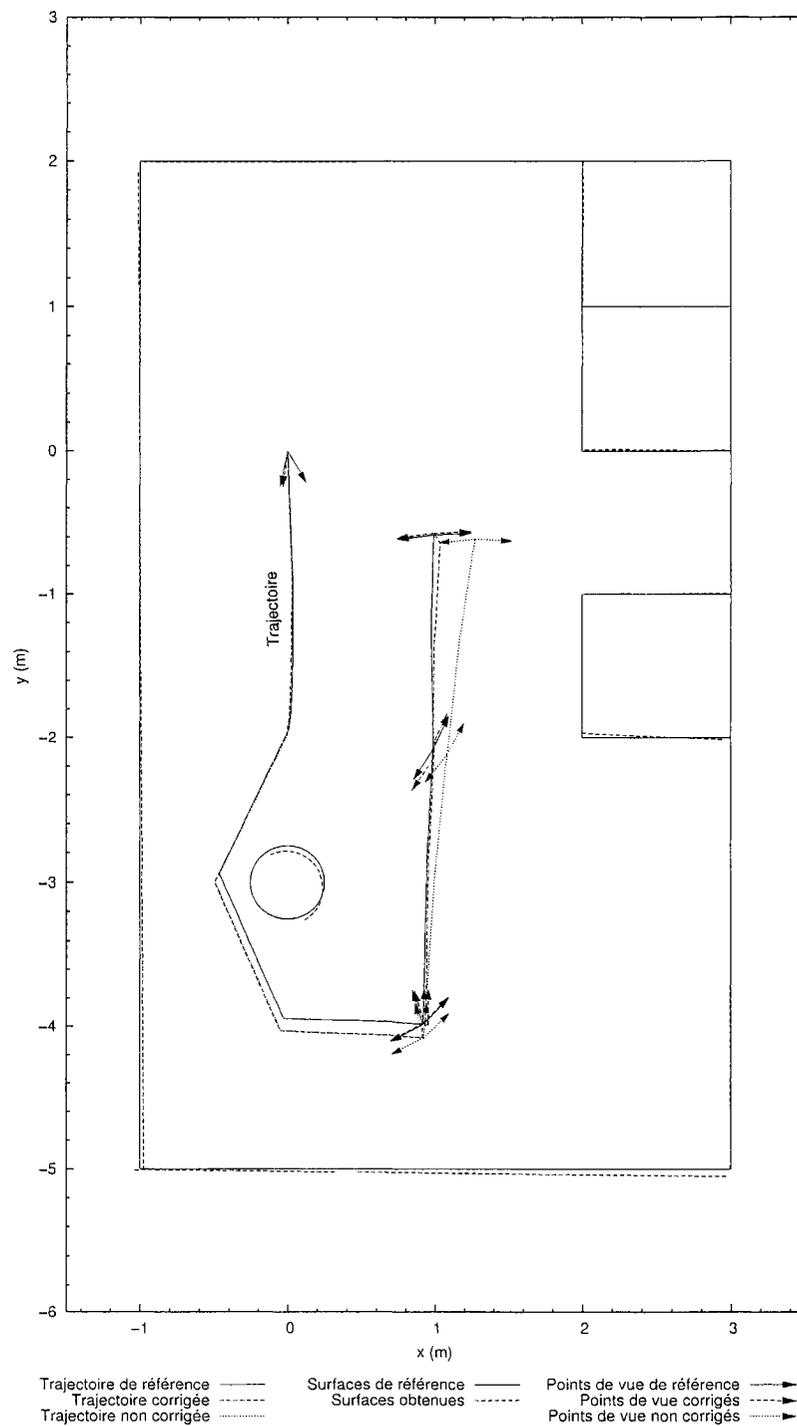


Figure 5.11 Intégration de vues multiples provenant d'une scène simulée

Erreur de positionnement Sans correction		Erreur de positionnement Avec correction	
Distance (m)	Angle (°)	Distance (m)	Angle (°)
0,000	0,00	0,000	0,00
0,000	0,08	0,000	0,00
0,094	0,73	0,008	0,40
0,094	1,52	0,008	0,00
<b>0,094</b>	<b>2,63</b>	<b>0,033</b>	<b>3,04</b>
0,094	2,60	0,017	0,16
0,104	2,42	0,019	0,29
<b>0,104</b>	<b>5,94</b>	<b>0,077</b>	<b>0,78</b>
0,285	5,28	0,011	0,18
0,285	1,91	0,012	0,30

Tableau 5.4 Erreurs de positionnement avec et sans correction

surfaces inscrites dans le tableau sont les dimensions des segments affichés sur la figure 5.11. L'erreur de dimension observée pour la dernière surface provient du fait que cette surface n'a pas été vue entièrement d'aucun point de vue. Si nous faisons abstraction de cette surface, nous obtenons que l'erreur maximale commise est de 5,9 cm et qu'elle est d'environ 3,6 cm en moyenne. L'erreur d'angle maximale observée est de 2,79°. Toutefois, il s'agit d'un cas isolé puisque la moyenne des autres valeurs est de  $0,23^{\circ 4}$ .

### 5.3.2 Résultats provenant d'images réelles

Nous présentons maintenant quelques expérimentations réalisées sur des scènes réelles. Pour la première expérimentation, présentée à la figure 5.12 de la page 159, nous avons repris la scène présentée à la page 147. Une photographie de la scène est présentée à la figure 5.12 (g). Cette fois, nous avons numérisé la scène sous

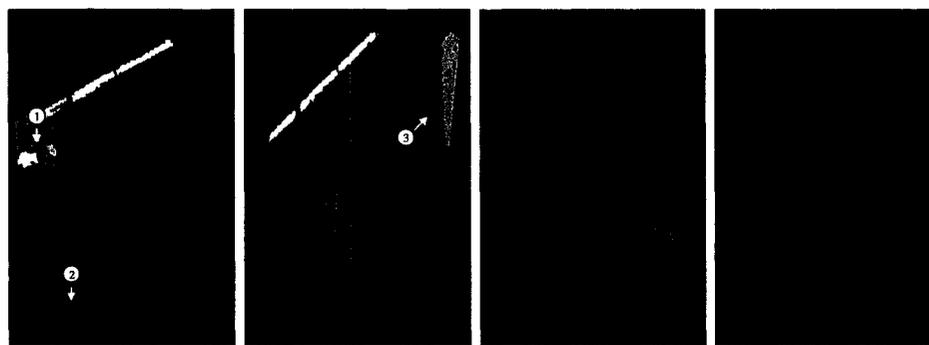
<sup>4</sup>Étrangement, cette valeur est proche de la résolution angulaire du télémètre laser qui est de  $0,25^{\circ}$ . Toutefois, ces deux valeurs n'ont pas de lien entre elles.

Dimension attendue (m)	Dimension obtenue (m)	Angle attendu (°)	Angle obtenu (°)	Erreur de dimension (m)	Erreur d'angle (°)
7,000	6,941	90,00	90,31	0,059	0,31
5,000	5,006	90,00	89,90	0,006	0,1
1,000	0,978	90,00	89,88	0,022	0,12
1,000	0,946	90,00	89,88	0,054	0,12
1,000	0,942	90,00	89,84	0,058	0,16
4,000	4,013	0,00	-0,63	0,013	0,63
3,000	2,971	0,00	0,07	0,029	0,07
1,000	0,962	0,00	-2,79	0,038	2,79
1,000	0,958	0,00	-0,43	0,042	0,43
1,000	0,713	0,00	-0,18	0,287	0,18

Tableau 5.5 Comparaison entre les surfaces attendues et obtenues

quatre points de vue différents. Les quatre segmentations résultant de chaque acquisition sont présentées aux figures 5.12 (a) à (d) dans l'ordre d'acquisition et de fusion. L'ajout de la figure 5.12 (b) présente une situation idéale où il est possible de localiser parfaitement la plate-forme, alors que l'ajout des figures 5.12 (c) et (d) présentent des situations où toutes les surfaces planes pouvant servir à la localisation sont parallèles. Dans ces situations, notre algorithme de localisation peut corriger uniquement l'angle de rotation et la position dans l'axe perpendiculaire aux surfaces ; il ne peut pas effectuer de correction dans la direction tangente aux surfaces. L'ajout de la figure 5.12 (b) présente également une situation où une portion de colonne a été perçue comme une surface plane (voir flèche no. 3) et doit être fusionnée à une surface courbe déjà existante.

Le résultat de la fusion est présenté à la figure 5.12 (e). Nous avons également présenté une coupe du résultat parallèle au plancher à laquelle nous avons superposé les surfaces telles que mesurées manuellement au moyen d'un ruban à mesurer. Cette comparaison est présentée à la figure 5.12 (f). Les résultats montrent que,

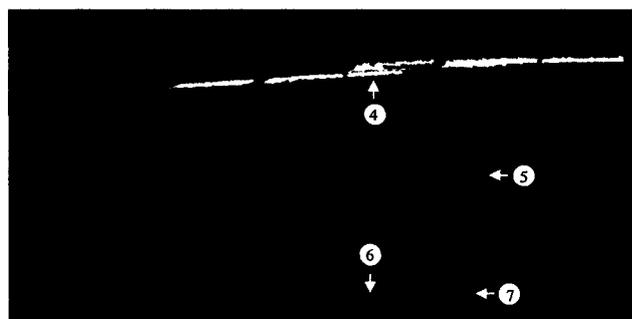


(a) Vue 1

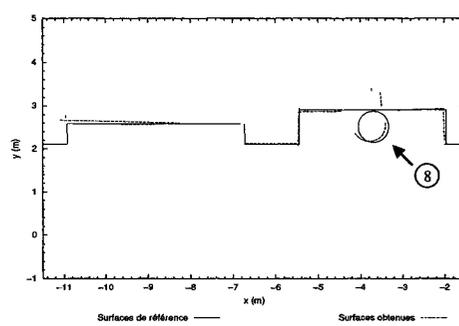
(b) Vue 2

(c) Vue 3

(d) Vue 4



(e) Résultat, vue de face



(f) Résultat, vue de dessus



(g) Scène observée

Figure 5.12 Intégration de vues multiples provenant d'une scène réelle

globalement, la fusion des vues multiples fonctionne bien. Les dimensions obtenues correspondent bien aux dimensions mesurées manuellement. Toutefois, quelques petits accrocs sont visibles dans la figure 5.12 (e). Premièrement, nous notons une segmentation excessive du plancher (voir flèche no. 6). Celle-ci était déjà présente dans la première acquisition (voir flèche no. 2) et n'est pas due à la fusion des images. Deuxièmement, nous notons un décalage vertical au niveau des lumières blanches dans le centre au haut de la figure 5.12 (e) (voir flèche no. 4). La cause de ce décalage est actuellement inconnue. Une hypothèse est que les rainures des roues tout-terrain de la plate-forme mobile font que celle-ci n'est pas toujours parfaitement parallèle au sol. Une petite déviation de l'angle d'élévation de la plate-forme pourrait créer le décalage vertical observé. Dans cette situation, il aurait été utile de corriger la positionnement de la plate-forme en trois dimensions plutôt qu'en deux dimensions tel que nous l'avons implanté. Troisièmement, nous remarquons que la surface orange, à droite de la colonne (voir flèche no. 5) a été mal fusionnée, ce qui a comme conséquence de bloquer partiellement la fenêtre existant à cet endroit. Cela est dû à une mauvaise localisation de la troisième acquisition : les surfaces permettant d'effectuer la localisation étaient toutes parallèles de sorte qu'il était impossible pour l'algorithme de corriger le déplacement dans la direction tangente au mur. Cette erreur a également eu pour effet de déplacer la colonne légèrement vers la gauche de l'image (voir flèches nos. 7 et 8). Dans une telle situation, l'utilisation des surfaces courbes pour le positionnement en plus des surfaces planes aurait pu permettre d'utiliser la colonne pour améliorer le résultat. Nous remarquons que la fusion des surfaces a permis de corriger une segmentation excessive visible dans le coin supérieur gauche de la figure 5.12 (a) (voir flèche no. 1).

La second essai expérimental que nous présentons a été pris à l'intérieur d'un local de bureaux d'étudiants. Une photographie de ce bureau est présentée à la figure 5.13 (g). Quatre acquisitions ont été effectuées de quatre points de vue dif-

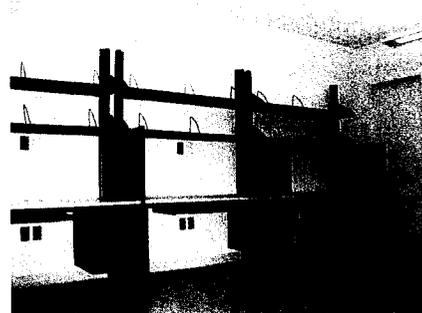
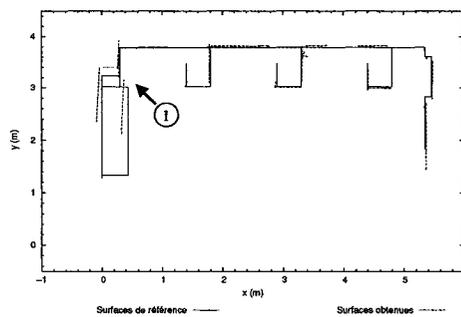
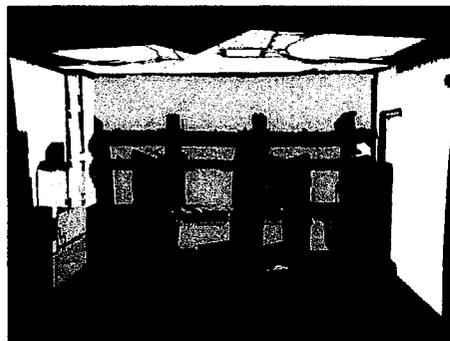
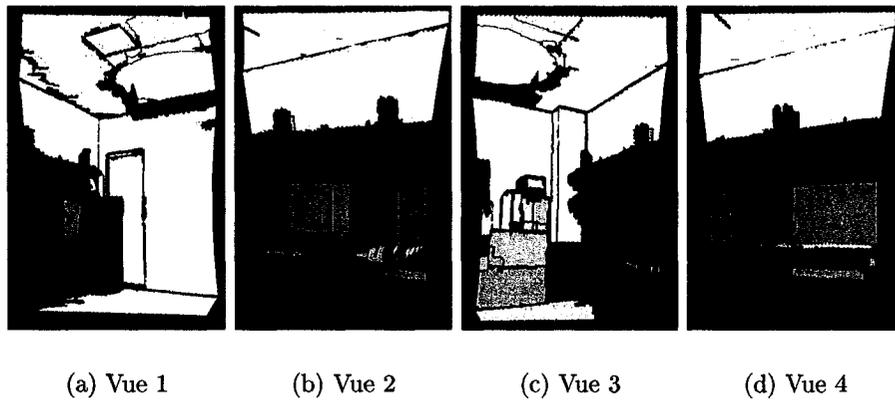


Figure 5.13 Intégration de vues multiples d'un local de travail

férents. Les résultats des segmentations de ces images sont présentés aux figures 5.13 (a) à (d). Il faut mentionner que c'est une scène très difficile à segmenter pour plusieurs raisons. Les surfaces des tables de travail sont presque au même niveau que le centre optique du capteur de sorte que ces surfaces sont quasi parallèles au champ de vision et donc pratiquement indétectables. Les tablettes sont également assez étroites de sorte que le capteur ne peut pas acquérir suffisamment de points pour bien les détecter. Seule la tablette la plus haute a été adéquatement détectée. La segmentation du plafond est décevante, principalement en raison de la présence des lumières néon qui, d'une part, ont causé une surexposition autour des néons et, d'autre part, ont augmenté les contrastes au plafond créant une segmentation photométrique excessive. Le résultat de l'intégration est présenté aux figures 5.13 (e) et (f). Les surfaces bien détectées sont globalement toutes à la bonne place, comme le montre la figure 5.13 (f), à l'exception de celles situées à la gauche de la figure qui ont subies un décalage d'environ 12 cm et une légère rotation (voir flèche no. 1). Peu d'informations permettaient de déterminer la position exacte de la troisième acquisition — celle de la figure 5.13 (c) — dans la scène.

Pour la dernière expérimentation que nous présentons, nous avons réalisé huit acquisitions 3D d'une portion d'un corridor du cinquième étage du pavillon Lassonde de l'École Polytechnique de Montréal. Ces huit acquisitions ont été segmentées pour donner les résultats présentés à la figure 5.14 et fusionnées pour donner le résultat présenté à la figure 5.15. Aucune erreur importante n'est visible dans le résultat obtenu, outre une segmentation excessive du plafond à la gauche de l'image 5.15 (a) (voir flèche no. 1). Nous remarquons un décalage d'environ 7 cm des surfaces situées dans le coin inférieur gauche de la figure 5.15 (b) (voir flèche no. 2). En fait, aucune des huit acquisitions effectuées ne voyait le mur principal (voir flèche no. 3) au complet. Il était donc impossible, à partir de ces images de déterminer en tout temps la position exacte de la plate-forme. L'erreur de 7 cm observée est

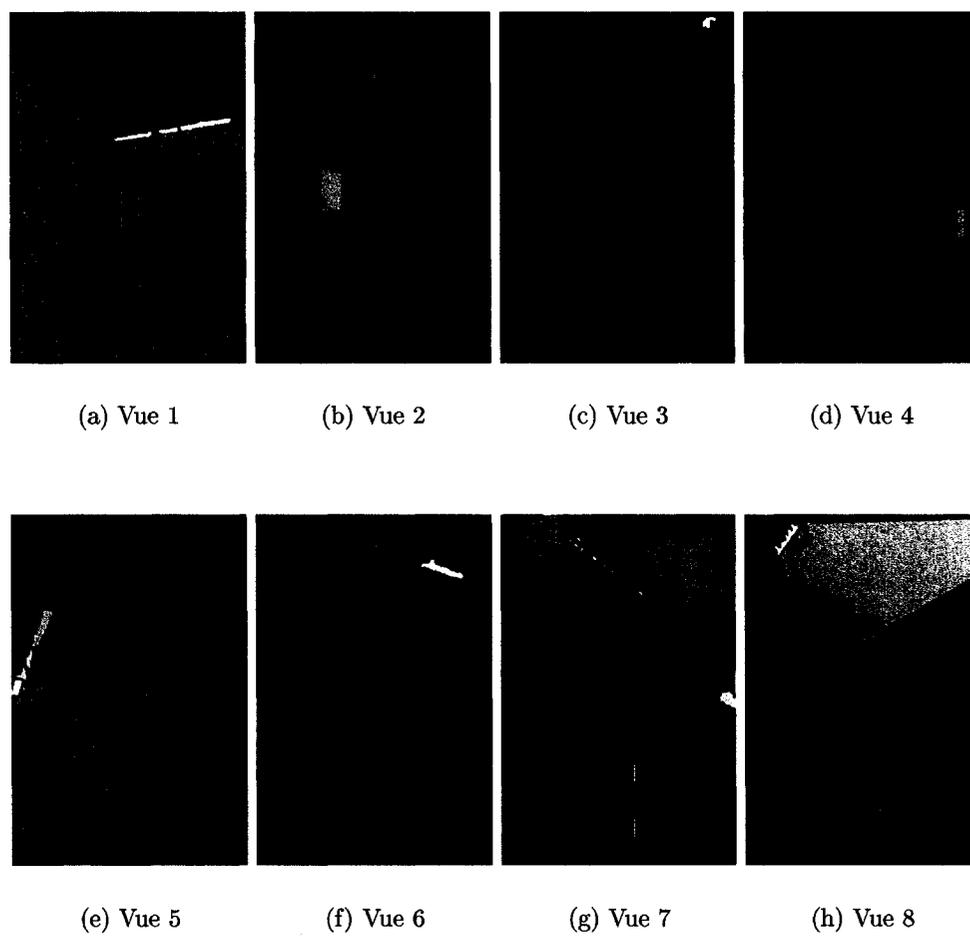
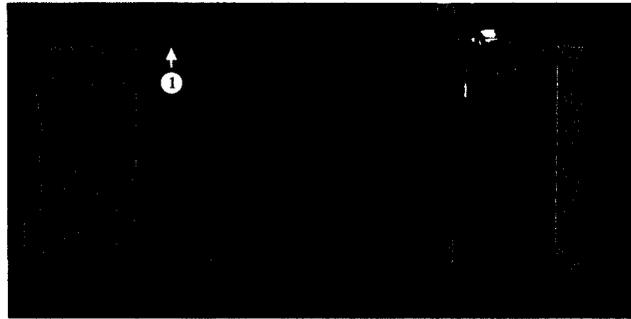
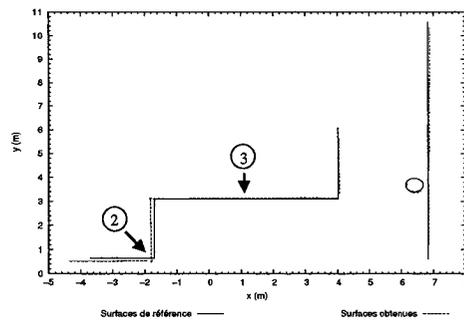


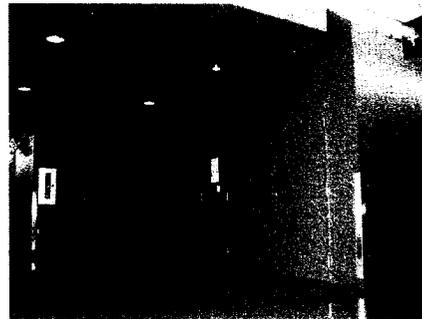
Figure 5.14 Vues multiples d'une portion d'un corridor, premier essai



(a) Résultat, vue de face



(b) Résultat, vue de dessus



(c) Scène observée

Figure 5.15 Intégration des vues multiples d'une portion d'un corridor, premier essai

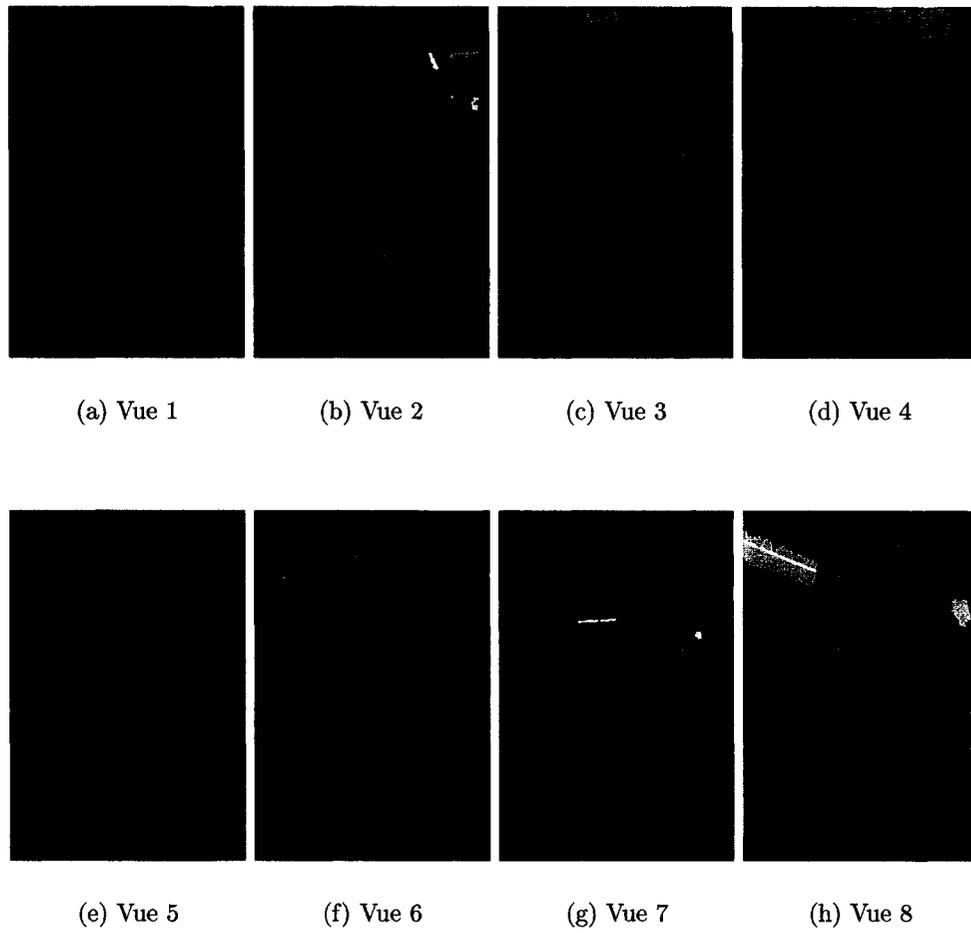
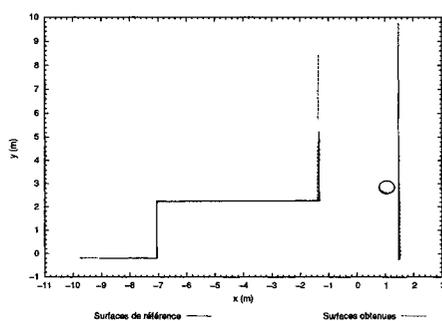


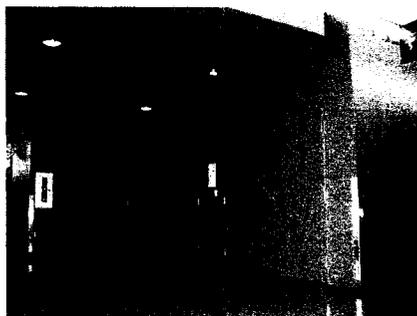
Figure 5.16 Vues multiples d'une portion d'un corridor, deuxième essai



(a) Résultat, vue de face



(b) Résultat, vue de dessus



(c) Scène observée

Figure 5.17 Intégration des vues multiples d'une portion d'un corridor, deuxième essai

donc principalement due à l'odométrie de la plate-forme mobile. Une photographie de la scène observée est présentée à la figure 5.15 (c).

Nous avons renumérisé la même scène en choisissant minutieusement huit nouveaux points de vue. Les résultats de cette seconde numérisation sont présentés aux figures 5.16 et 5.17. Les erreurs de dimensions observées précédemment ont été éliminées. Cela démontre l'importance de bien choisir les points de vue.

### **5.3.3 Conclusion des résultats de la fusion des vues multiples**

L'analyse des résultats de la fusion des vues multiples montre un succès mitigé. D'un côté l'algorithme de localisation basé sur la comparaison des surfaces planes présentes dans la carte et dans la nouvelle acquisition donne de très bons résultats quand il est possible de mettre plusieurs surfaces en correspondance. D'un autre côté, lorsque le nombre de surfaces en correspondance est restreint, l'algorithme de localisation devient très susceptible aux erreurs d'orientation et de position des surfaces. De plus, il arrive fréquemment que le nombre de surfaces pouvant être appariées est insuffisant pour corriger adéquatement les erreurs d'odométrie de la plate-forme mobile.

Ce problème montre la difficulté du choix des points de vue. En effet, le choix d'un point de vue est un compromis entre l'objectif de maximiser la quantité de nouvelles données sur l'environnement afin de minimiser le temps requis pour réaliser la cartographie et le besoin de maximiser la quantité d'informations déjà présentes dans la carte pour améliorer la qualité du positionnement de la plate-forme mobile. La dernière expérimentation, réalisée avec deux séries de points de vue différents de la même scène, démontre bien l'importance d'optimiser le choix des points de vue.

Puisque notre algorithme travaille au niveau des surfaces et non pas au niveau des points, il est très rapide, ce qui est un avantage incontestable en comparaison avec l'algorithme ICP classique. Notre algorithme pourrait même être exécuté en temps réel pendant le déplacement de la plate-forme. Pour cette raison, nous proposons, comme piste de recherche pour les travaux futurs, de placer le télémètre laser en position horizontale lors des déplacements de sorte qu'il acquiert une coupe horizontale de l'environnement. En corrélant les segments présents dans cette coupe avec les surfaces présentes dans la carte, il deviendra possible de corriger l'odométrie de la plate-forme pendant son déplacement et non pas seulement lors des nouvelles acquisitions 3D. En couplant cela à un bon modèle de l'erreur de l'odométrie, il devrait être possible de réduire significativement les quelques erreurs de fusion observées ci-dessus.

Finalement, nous avons pu constater que la fusion de plusieurs surfaces bien positionnées en une seule surface fonctionne très bien et permet même, dans certaines situations, de corriger quelques erreurs de segmentation excessive.

## CONCLUSION

La possibilité de reconstruire virtuellement en trois dimensions et en couleur un environnement structuré a des applications importantes dans plusieurs domaines. Les effets spéciaux au cinéma, l'incorporation de lieux réels dans les jeux vidéo, la visualisation 3D avec couleur pour les architectes et les ingénieurs et la planification de tâche de navigation par des plates-formes robotiques mobiles en milieu préalablement inconnu, sinistrés ou hostiles ne sont que quelques exemples d'applications.

Nous avons présenté une solution pour l'acquisition des données brutes, pour leur traitement afin d'extraire les surfaces des objets et pour l'intégration de ces surfaces pour obtenir une représentation géométrique 3D étendue avec couleur de l'environnement observé. Le projet que nous avons réalisé possède plusieurs aspects innovateurs :

Premièrement, nous avons conçu un nouveau capteur 3D pour l'acquisition de données télémétriques et photométriques de manière enregistrée de l'environnement à cartographier. Les résultats obtenus ont montré que l'erreur télémétrique moyenne de notre capteur est de 5,9 mm. Les données télémétriques et photométriques obtenues sont en superposition tridimensionnelle quasi exacte ; le décalage maximal observé entre ces données est de 1 pixel.

Deuxièmement, nous avons développé une nouvelle approche pour la segmentation d'images combinant télémétrie et photométrie. Nous réalisons une extraction de primitives géométriques correspondant à des portions de plans, de cylindres et de sphères à partir des images obtenues. L'extraction de ces primitives est basée sur une double segmentation de la scène, l'une basée sur la télémétrie et la se-

conde basée sur la photométrie. Les deux segmentations sont ensuite fusionnées pour obtenir une segmentation qui tient compte des deux modalités sensorielles. Cette approche possède l'avantage de pouvoir être rapidement adaptée à de nouvelles percées dans les domaines de la segmentation d'images télémétriques et de la segmentation d'images photométriques. La segmentation séparée des deux modalités sensorielles augmente également la robustesse du système : advenant la panne d'une modalité sensorielle, l'autre modalité peut continuer à être utilisée sans perturbation. L'implantation réalisée fournit un taux de bonnes détections de 93 %. Nous obtenons ainsi une représentation constituée de primitives géométriques et photométriques intrinsèque à la scène plutôt qu'en primitives extrinsèques (pixels, voxels, etc.).

Troisièmement, nous avons montré que la photométrie peut être utilisée dans certaines occasions pour éliminer l'ambiguïté des contours de la segmentation télémétrique due à l'imprécision du télémètre laser utilisé.

Finalement, nous avons présenté une nouvelle version de l'algorithme de positionnement ICP qui permet de réaliser la localisation du capteur à partir des surfaces détectées suite à la segmentation plutôt qu'à partir des données brutes. Cette technique permet de corriger les erreurs de positionnement accumulées par l'odométrie de la plate-forme et de conserver une erreur inférieure à 2 cm et à  $0,5^\circ$ . Le principal avantage de notre nouvel algorithme par rapport à l'algorithme ICP classique est sa très grande rapidité qui est due au fait que le nombre de surfaces présentes dans la scène est de plusieurs ordres de grandeur inférieur au nombre de points contenus dans les données brutes.

Nous avons également réalisé un module permettant l'intégration d'images segmentées provenant de plusieurs points de vue du même environnement. Nous obtenons ainsi une représentation étendue dans laquelle nous pouvons décrire des scènes

excédant les limites géométriques du champ de vision du senseur utilisé. Nous obtenons ainsi des cartes 3D dont la précision est de l'ordre de 5 cm et de  $0,25^\circ$ . Les erreurs obtenues sont dues en partie aux erreurs de mesure de notre capteur et aux difficultés de localisation lorsque les observations sont insuffisantes pour déterminer précisément la position du capteur dans la carte.

Plusieurs pistes de réflexion peuvent faire l'objet de développements futurs. Au niveau des améliorations pouvant être apportées au système implanté, il serait intéressant d'accroître le champ de vision photométrique du capteur. En effet, le télémètre laser possède, à la résolution employée, un angle d'ouverture de  $100^\circ$  alors que celui de la caméra utilisée est de  $42^\circ$ . Les données télémétriques situées en-dehors du champ de vision de la caméra sont actuellement inutilisées. Pour augmenter le champ de vision, il serait possible d'utiliser deux caméras ou une lentille grand angle. Cela permettrait d'augmenter la quantité de données acquises pendant le même intervalle de temps et donc d'augmenter la vitesse d'acquisition. Chaque image serait composée d'un plus grand nombre de surfaces, ce qui devrait aider à la localisation de la plate-forme sur laquelle est monté le capteur.

Les performances de la segmentation photométrique pourraient être améliorées en développant des stratégies pour diminuer le rejet des petites surfaces et pour diminuer les effets des brusques variations de l'éclairage. Le premier objectif pourrait être atteint en fusionnant les petites surfaces qui possèdent des caractéristiques communes alors que le second objectif pourrait être atteint en utilisant des espaces de couleurs comme le HSV (teinte, saturation, valeur) ou le YUV (luminance et chrominance) pour la segmentation ou en fournissant au capteur sa propre source lumineuse afin de mieux contrôler l'éclairage.

Au niveau du positionnement du capteur, il serait utile d'utiliser les surfaces courbes, surtout dans les situations où il y a peu de correspondances entre les

surfaces planes de la nouvelle acquisition et celles de la carte.

En modélisant l'erreur du déplacement de la plate-forme mobile et en utilisant possiblement un filtre de Kalman, il serait possible d'améliorer le positionnement de la plate-forme et d'augmenter la précision des surfaces issues de l'intégration des vues multiples.

Plusieurs nouvelles fonctionnalités pourraient s'ajouter au travail réalisé à ce jour. Au niveau du capteur, nous avons indiqué au chapitre 3 que la fusion des segmentations télémétriques et photométriques permettrait de réaliser une autocalibration du capteur.

Une autre fonctionnalité intéressante serait d'utiliser le télémètre laser pendant le déplacement de la plate-forme afin d'effectuer la correction de l'odométrie en temps réel en utilisant la méthode proposée à la fin du chapitre 5.

Finalement, des recherches pourraient être menées pour permettre à la plate-forme mobile de choisir de manière autonome le prochain point de vue à numériser et à intégrer à la carte. Cela, combiné à un module de navigation, permettrait à la plate-forme de réaliser de la cartographie 3D de manière autonome.

Ce projet a permis de mettre en place l'infrastructure nécessaire à l'acquisition de données 3D précises sur l'environnement, à la segmentation de ces données pour en faire ressortir les surfaces des objets présents dans l'environnement et à l'intégration de vues multiples dans le but d'obtenir une carte 3D plus complète. Une plate-forme mobile réalise de manière autonome les tâches de prise de mesure, de segmentation, de localisation et d'intégration des surfaces dans la carte. Des développements futurs portant sur la navigation autonome et le choix optimal des points de vue permettraient d'obtenir une solution complètement autonome au problème de localisation et cartographie simultanée (SLAM) en milieu structuré.

## RÉFÉRENCES

- Axelsson, P. (1999). Processing of laser scanner data - algorithms and applications. *ISPRS Journal of Photogrammetry & Remote Sensing*, **54**, 138–147.
- Bellon, O. R. et Silva, L. (2002). New improvements to range image segmentation by edge detection. *IEEE Signal Processing Letters*, **9**(2), 43–45.
- Besl, P. J. et McKay, N. D. (1992). A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **14**(2), 239–256.
- Brenneke, C., Wulf, O., et Wagner, B. (2003). Using 3d laser range data for slam in outdoor environments. *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 188–193.
- CNRC (2006). Technologies d'imagerie tridimensionnelle élaborées par le groupe TIV. Consulté le 20 octobre 2006, tiré de [http://iit-iti.nrc-cnrc.gc.ca/vit-tiv/3d-developed\\_f.html](http://iit-iti.nrc-cnrc.gc.ca/vit-tiv/3d-developed_f.html).
- Comaniciu, D. et Meer, P. (1997). Robust analysis of feature spaces : Color image segmentation. *IEEE Conference on Computer Vision and Pattern Recognition*, 750–755.
- Digabel, H. et Lantuejoul, C. (1978). Iterative algorithms. *Actes du Second Symposium Europeen d'Analyse Quantitative des Microstructures en Sciences des Matériaux, Biologie et Médecine*, 85–99.
- Edelsbrunner, H. et Mücke, E. P. (1994). Three-dimensional alpha shapes. *ACM Transactions on Graphics*, **13**(1), 43–72.
- El-Hakim, S. F., Boulanger, P., Blais, F., Beraldin, J.-A., et G., R. (1997). A mobile system for indoor 3-d mapping and positioning. *Proceedings of the Optical 3-D Measurement Techniques IV*, 275–282.

- Eom, S., Chang, S., et Ahn, B. (2002). Watershed-based region merging using conflicting regions. *Proceedings of the 2002 International Conference on Image Processing*, **2**, 781–784.
- Fischler, M. A. et Bolles, R. C. (1981). Random sample consensus : a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, **24**(6), 381–395.
- Früh, C. et Zakhor, A. (2002). Data processing algorithms for generating textured 3d building façade meshes from laser scans and camera images. *Proceedings of the First International Symposium on 3D Data Processing Visualization and Transmission (3DPVT '02)*, 834–847.
- Gao, H., Siu, W.-C., et Hou, C.-H. (2001). Improved techniques for automatic image segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*, **11**(12), 1273–1280.
- Garcia, M. A., Sappa, A. D., et Basañez, L. (1999). Efficient generation of object hierarchies from 3d scenes. *Proceedings of the IEEE International Conference on Robotics and Automation*, **2**, 1359–1364.
- Garland, M. et Heckbert, P. S. (1997). Surface simplification using quadric error metrics. *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*.
- Gerbaud, T. (2004). Reconstruction tridimensionnelle et exploration autonome d'environnements non structurés. Mémoire de maîtrise, École Polytechnique de Montréal, Québec, Canada.
- Gerkey, B., Vaughan, R., et Howard, A. (2003). The player/stage project : Tools for multi-robot and distributed sensor systems. *Proceedings of the International Conference on Advanced Robotics (ICAR 2003)*, 317–323.
- Ghuneim, A. G. (2000). Square tracing algorithm. Consulté le 14 juillet 2006, tiré de <http://www.cs.mcgill.ca/~aghnei/sqmain.html>.

- Godin, G. D. (2003). Three dimensional optical scanning. Brevet américain US 6,507,036, United States Patent and Trademark Office. Consulté le 25 octobre 2006, tiré de <http://www.uspto.gov/patft/index.html>.
- Gotardo, P. F., Bellon, O. R., et Silva, L. (2003). Range image segmentation by surface extraction using an improved robust estimator. *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'03)*, **2**, 33–38.
- Gotardo, P. F. U., Bellon, O. R. P., Boyer, K. L., et Silva, L. (2004). Range image segmentation into planar and quadric surfaces using an improved robust estimator and genetic algorithm. *IEEE Transactions on Systems, Man and Cybernetics*, **34**(6), 2303–2316.
- Grabska, E., Ślusarczyk, G., et Krzysztof, P. (2003). Interpretation of objects represented by hierarchical graphs. *Proceedings of Computer Pattern Recognition Systems (KOSYR)*, 287–293. Consulté le 30 août 2006, tiré de <http://www.ii.uj.edu.pl/~grazyna/>.
- Hoover, A., Jean-Baptiste, G., Jiang, X., Flynn, P. J., Bunke, H., Goldgof, D. B., Bowyer, K., Eggert, D. W., Fitzgibbon, A., et Fisher, R. B. (1996). An experimental comparison of range image segmentation algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **18**(7), 673–689.
- Hoppe, H. (1996). Progressive meshes. *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 99–108.
- Huang, W. H. et Beevers, K. R. (2005). Topological map merging. *International Journal of Robotics Research*, **24**(8), 601–613.
- Hähnel, D., Burgard, W., et Thrun, S. (2003). Learning compact 3d models of indoor and outdoor environments with a mobile robot. *Journal of Robotics and Autonomous Systems*, **44**, 15–27.

- Jensfelt, P. et Kristensen, S. (2001). Active global localization for a mobile robot using multiple hypothesis tracking. *IEEE Transaction on Robotics and Automation*, **17**(5), 748–760.
- Jiang, X., Bowyer, K., Morioka, Y., Hiura, S., Sato, K., Inokuchi, S., Bock, M., Guerra, C., Loke, R., et Buf, J. (2000). Some further results of experimental comparison of range image segmentation algorithms. *Proceedings of the 15th International Conference on Pattern Recognition*, **4**, 877–881.
- Kambhampati, S. et Davis, L. S. (1986). Multiresolution path planning for mobile robots. *IEEE Journal on Robotics and Automation*, **2**(3), 135–145.
- Konolige, K., Fox, D., Limketkai, B., Ko, J., et Stewart, B. (2003). Map merging for distributed robot navigation. *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 212–217.
- Lallement, A., Siadat, A., Dufaut, M., et Husson, R. (1998). Laser-vision cooperation for map building and landmarks recognition. *Proceedings of the 1998 IEEE ISIC/CIRA/ISAS Joint Conference*, 387–392.
- Langis, C., Greenspan, M., et Godin, G. (2001). The parallel iterative closest point algorithm. *Proceedings of the Third International Conference on 3-D Digital Imaging and Modeling*, 195–202.
- Lee, K.-M., Meer, P., et Park, R.-H. (1998). Robust adaptive segmentation of range images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **20**(2), 200–205.
- Levenberg, K. (1944). A method for the solution of certain problems in least squares. *Quarterly of Applied Mathematics*, **2**, 164–168.
- Lorensen, W. E. et Cline, H. E. (1987). Marching cubes : A high resolution 3d surface construction algorithm. *Computer Graphics (Proceedings of SIGGRAPH '87)*, **21**(4), 163–169.

Lourakis, M. (2006). levmar : Levenberg-marquardt nonlinear least squares algorithms in c/c++. Consulté le 14 juillet 2006, tiré de <http://www.ics.forth.gr/~lourakis/levmar/>.

Lucchese, L. et Mitra, S. K. (2001). Color image segmentation : A state-of-the-art survey. *Image Processing, Vision and Pattern Recognition, Proceedings of the Indian National Science Academy*, **64**(2), 207–221.

Lyer, N., Jayanti, S., Lou, K., Kalyanaraman, Y., et Ramani, K. (2004). A multi-scale hierarchical 3d shape representation for similar shape retrieval. *Proceedings of the International Symposium Series on Tools and Methods of Competitive Engineering*, 1117–1118.

Marquardt, D. (1963). An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal of Applied Mathematics*, **11**, 431–441.

Mueller, K., Smolic, P., Merkle, P., Kaspar, B., Eisert, P., et Wiegand, T. (2004). 3d reconstruction of natural scenes with view-adaptive multi-texturing. *Proceedings of the 2nd International Symposium on 3D Data Processing, Visualization, and Transmission*, 116–123.

Murta, A. (2006). A general polygon clipping library. Consulté le 16 octobre 2006, tiré de <http://www.cs.man.ac.uk/~toby/alan/software/gpc.html>.

Newman, T. S., Flynn, P. J., et Jain, A. K. (1993). Model-based classification of quadric surfaces. *CVGIP : Image Understanding*, **58**(2), 335–349.

Nüchter, A., Surmann, H., et Hertzberg, J. (2003a). Planning robot motion for 3d digitalization of indoor environments. *Proceedings of the 11th International Conference on Advanced Robotics*, 222–227.

Nüchter, A., Surmann, H., et Hertzberg, J. (2003b). Automatic model refinement for 3d reconstruction with mobile robot. *Proceedings of the 4th IEEE International Conference on Recent Advances in 3D Digital Imaging and Modeling*, 394–401.

- Nüchter, A., Surmann, H., Lingemann, K., et Hertzberg, J. (2003c). Semantic scene analysis of scanned 3d indoor environments. *Vision, Modeling and Visualization 2003*.
- Nüchter, A., Wulf, O., Lingemann, K., Hertzberg, J., Wagner, B., et Surmann, H. (2005). 3d mapping with semantic knowledge. *Proceedings of the RoboCup International Symposium 2005*.
- Oggier, T., Lehmann, M., Kaufmann, R., Schweizer, M., Richter, M., Metzler, P., Lang, G., Lustenberger, F., et Blanc, N. (2004). An all-solid-state optical range camera for 3d real-time imaging with sub-centimeter depth resolution (swissranger). *Proceedings of the SPIE - Optical Design and Engineering*, **5249**, 534–545.
- Payeur, P., Laurendeau, D., et M., G. C. (1998). Range data merging for probabilistic octree modeling of 3-d workspaces. *Proceedings of the 1998 IEEE International Conference on Robotics & Automation*, **4**, 3071–3078.
- Pervözl, K., Nüchter, A., Surmann, H., et Hertzberg, J. (2004). Automatic reconstruction of colored 3d models. *Proceedings of Robotik 2004*, 215–222.
- Pujas, P. et Aldon, M.-J. (1997). Dense 3d-color image segmentation. *4th Annual Conference on Mechatronics and Machine Vision in Practice*, 234–239.
- Ramalingam, S. et Lodha, S. K. (2003). Adaptive enhancement of 3d scenes using hierarchical registration of texture-mapped 3d models. *Fourth International Conference on 3-D Digital Imaging and Modeling*, 203–210.
- Rimon, E. et Boyd, S. P. (1997). Obstacle collision detection using best ellipsoid fit. *Journal of Intelligent and Robotic Systems*, **18**, 105–126.
- Roerdink, J. et Meijster, A. (2000). The watershed transform : Definitions, algorithms and parallelization strategies. *Fundamenta Informaticae*, **41**(1), 187–228.
- Sabata, B., Arman, F., et Aggarwal, J. K. (1990). Segmentation of 3-d range images using pyramidal data structures. *Proceedings of the Third International Conference on Computer Vision, IEEE*, 662–666.

- Sappa, A. D. et Garcia, M. A. (2004). Hierarchical clustering of 3d objects and its application to minimum distance computation. *Proceedings of the 2004 IEEE International Conference on Robotics & Automation*, 5287–5292.
- Seitz, S. M. et Dyer, C. R. (1999). Photorealistic scene reconstruction by voxel coloring. *Journal of Computer Vision*, **35**(2), 151–173.
- Sequeira, V., Ng, K., Wolfart, E., Gonçalves, J. a. G., et Hogg, D. (1998). Automated 3d reconstruction of interiors with multiple scan-views. *International Society for Optical Engineering proceedings series*, **3641**, 106–117.
- Sharp, G. C., Lee, S. W., et K., W. D. (2002). ICP registration using invariant features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**(1), 90–102.
- Shen, X. et Spann, M. (1997). Segmentation of 2d and 3d images through a hierarchical clustering based on region modelling. *Proceedings of the International Conference on Image Processing*, **3**, 50.
- Silva, L., Bellon, O. R., et Gotardo, P. F. (2002). A global-to-local approach for robust range image segmentation. *Proceedings of the 2002 IEEE International Conference on Image Processing*, **1**, 773–776.
- Specht, A. R. et Devy, M. (2004). Surface segmentation using a modified ball-pivoting algorithm. *International Conference on Image Processing*, **3**, 1931–1934.
- Stroupe, A. W. et Balch, T. (2002). Collaborative probabilistic constraint-based landmark localization. *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 447–453.
- Surmann, H., Lingemann, K., Nüchter, A., et Hertzberg, J. (2001). A 3d laser range finder for autonomous mobile robots. *Proceedings of the 32nd international Symposium on Robotics*, 153–158.

- Surmann, H., Nüchter, A., et Hertzbert, J. (2003). An autonomous mobile robot with a 3d laser range finder for 3d exploration and digitalization of indoor environments. *Journal of Robotics and Autonomous Systems*, **45**, 181–198.
- Szafran, N. (2006). Plan aux moindres carrés. [En ligne]. Consulté le 14 juillet 2006, tiré de <http://www-lmc.imag.fr/lmc-mga/Nicolas.Szafran/MASTER2/SM2-CS/node25.html>.
- Sáez, J. M. et Escolano, F. (2004). A global 3d map-building approach using stereo vision. *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*, 1197–1202.
- Thrun, S., Burgard, W., et Fox, D. (2000). A real-time algorithm for mobile robot mapping with applications to multi-robot and 3d mapping. *Proceedings of the IEEE International Conference on Robotics and Automation*, **1**, 321–328.
- Vatti, B. R. (1992). A generic solution to polygon clipping. *Communications of the ACM*, **35**(7), 56–63.
- Vincent, L. et Soille, P. (1991). Watersheds in digital spaces : an efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(6), 583–598.
- Weingarten, J. W., Gruener, G., et Siegwart, R. (2004). Probabilistic plane fitting in 3d and an application to robotic mapping. *Proceedings of the IEEE International Conference on Robotics and Automation*, **1**, 929–932.
- Whaite, P. et P., F. F. (1997). Autonomous exploration : Driven by uncertainty. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**(3), 193–205.
- Wikipédia (2006). Projection de Mercator. Consulté le 16 octobre 2006, tiré de [http://fr.wikipedia.org/wiki/Projection\\_de\\_Mercator](http://fr.wikipedia.org/wiki/Projection_de_Mercator).
- Yu, Y., Ferencz, A., et Malik, J. (2001). Extracting objects from range and radiance images. *IEEE Transactions on Visualization and Computer Graphics*, **7**(4), 351–364.

- Zalzal, V. (2005). Localisation mutuelle de plates-formes robotiques mobiles par vision omnidirectionnelle et filtrage de Kalman. Mémoire de maîtrise, École Polytechnique de Montréal, Québec, Canada.
- Zalzal, V. et Cohen, P. (2006). Mutual localization of mobile robotic platforms using Kalman filtering. *32nd Annual Conference of the IEEE Industrial Electronics Society*. Numéro d'acceptation : PF-001899.
- Zelinsky, A. (1992). A mobile robot exploration algorithm. *IEEE Transactions on Robotics and Automation*, **8**(6), 707–717.
- Zhang, C. et Wang, P. (2000). A new method of color image segmentation based on intensity and hue clustering. *Proceedings of the 15th International Conference on Pattern Recognition*, **3**, 613–616.
- Zhang, Q. et Pless, R. (2004). Extrinsic calibration of a camera and laser range finder (improves camera calibration). *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*, **3**, 2301–2306.
- Zhang, Z. (1999). Flexible camera calibration by viewing a plane from unknown orientations. *Proceedings of the Seventh IEEE International Conference on Computer Vision*, **1**, 666–673.

## ANNEXE I

RÉSUMÉ DE CERTAINES TECHNIQUES DE SEGMENTATION  
EN SURFACES PLANES

Cette annexe présente un bref résumé des techniques de segmentations que nous avons recensées et qui sont évaluées au moyen de la méthode de Hoover et al. (Hoover et al., 1996).

**USF (Hoover et al., 1996)** Cet algorithme calcule un vecteur normal en chaque point en cherchant le meilleur plan passant dans une fenêtre de  $n \times n$  points. Une technique spéciale est utilisée pour retirer les points erronés et diminuer le bruit. Par la suite, une méthode de grossissement de région est appliquée pour obtenir les surfaces planes. Cette technique comporte 5 paramètres.

**WSU (Hoover et al., 1996)** Une détection des points appartenant aux contours des surfaces est obtenue par seuillage de la différence de profondeur des points. Des vecteurs normaux sont ensuite calculés pour les autres points. En juxtaposant les coordonnées du point et les composantes du vecteur normal, un espace à 6 dimensions est créé. Une technique de clustering est ensuite appliquée sur ces points. Cette technique comporte plus de 7 paramètres dont 4 sont considérés « critiques » par les auteurs.

**UB (Hoover et al., 1996)** Cet algorithme segmente l'image en deux étapes. Lors de la première étape, chaque ligne de l'image est divisée en segments. Lors de la deuxième étape, les segments sont fusionnés les uns aux autres en fonction de leur longueur et de leur orientation pour former des surfaces planes. Cette méthode, qui comporte 7 paramètres, est très semblable à celle employée par Surmann et al. (Surmann et al., 2003).

**UE (Hoover et al., 1996)** La première étape de cet algorithme consiste à calculer un vecteur normal et une courbure moyenne en chaque point. Ces informations permettent de différencier les points appartenant à une surface plane des points du contour. Puis un grossissement de région est effectué en utilisant une méthode itérative dans laquelle les points peuvent changer de région et où les équations de plans sont recalculées. Une dernière étape itérative est appliquée pour raffiner les contours des surfaces. Cette méthode comporte environ 12 paramètres.

**ALKS (Lee et al., 1998)** L'algorithme ALKS est une méthode itérative d'extraction de surfaces planes. Trois points sont pris au hasard pour former un plan. La qualité du plan obtenu est évaluée en regardant la distance minimale au plan requise pour contenir  $k$  points. Une procédure automatique permet de choisir la valeur de  $k$ , d'où le nom *adaptive least k squares* (ALKS). Le tirage aléatoire est effectué plusieurs fois avant de retenir le meilleur plan et de retirer les points appartenant à la surface plane détectée. La procédure est répétée itérativement pour extraire les autres surfaces planes.

**OU (Jiang et al., 2000)** La scène est coupée par des plans parallèles. Dans chaque plan, une détection de segments est pratiquée en utilisant la transformée de Hough. Les segments sont ensuite fusionnés pour obtenir des plans. Cette méthode comporte 5 paramètres.

**PPU (Jiang et al., 2000)** Il s'agit d'une méthode aléatoire qui regroupe plusieurs stratégies. Un prétraitement est effectué pour détecter des segments dans l'image. À l'aide d'un segment et d'un point pris au hasard, un plan est formé. Plusieurs plans sont ainsi générés et ceux décrivant le mieux la scène sont conservés. Des plans peuvent également être générés à partir de trois points ou à partir de deux segments coplanaires. Le nombre total de paramètres utilisés n'est pas spécifié.

- UA (Jiang et al., 2000)** Une image en tons de gris est générée en attribuant à chaque point de l'image une valeur scalaire obtenue en effectuant une somme pondérée des composantes du vecteur normal de la surface au point considéré. Puis un quadtree est utilisé pour obtenir des niveaux de moins en moins détaillés. Un algorithme de clustering est appliqué au niveau le moins détaillé puis à chaque sous-niveau pour augmenter graduellement la précision des contours des régions. Cette méthode comporte 10 paramètres.
- UFPR1 (Bellon et Silva, 2002)** Les contours des surfaces sont obtenus en appliquant un seuil sur le gradient de l'image. Les contours sont ensuite raffinés au moyen d'opérations morphologiques en utilisant des noyaux 3x3.
- UFPR2 (Silva et al., 2002)** Il s'agit d'une méthode aléatoire utilisant l'estimateur MSAC (M-estimator SAmple Consensus) pour effectuer des extractions successives de surfaces planes. Trois points sont choisis au hasard et servent à calculer une équation de plan. Plusieurs essais aléatoires sont ainsi faits et le meilleur plan, calculé selon la métrique du MSAC est retenu. Les points appartenant à cette surface plane sont retirés et la procédure est répétée jusqu'à ce que toutes les surfaces planes soient retirées.
- UFPR3 (Gotardo et al., 2003)** Il s'agit de l'algorithme UFPR2 augmenté d'un algorithme génétique qui a pour tâche de minimiser l'erreur globale causée principalement par les points situés proches des frontières entre les surfaces et qui risquent d'être attribués à la mauvaise surface.

## ANNEXE II

### INTÉGRATION DU CAPTEUR 3D À L'ARCHITECTURE LOGICIELLE DE LA PLATE-FORME MOBILE

Cette annexe décrit l'architecture logicielle en place sur notre plate-forme mobile et les modules développés dans le but d'intégrer le nouveau capteur à cette architecture.

#### II.1 Architecture logicielle

Notre plate-forme mobile est équipée du système d'exploitation Linux Fedora Core 3 et de l'architecture logicielle Player, version 1.6.2, disponible sous licence GPL. Une couche logicielle supplémentaire, nommée Acropolis, a été développée au GRPR principalement par M. Vincent Zalzal (Zalzal, 2005).

##### II.1.1 Couche d'abstraction matérielle

L'architecture logicielle de la plate-forme mobile est divisée en deux couches logicielles : Player et Acropolis. Le logiciel libre Player, conçu à la University of South California (Gerkey et al., 2003), est utilisé pour faire abstraction du matériel. Player est une architecture client-serveur. Tous les capteurs et actuateurs de la plate-forme mobile sont gérés par leur propre pilote du côté serveur. Chaque marque de matériel a son propre pilote. Toutefois, les données fournies par les pilotes sont encapsulées dans des interfaces simples standardisées pour chaque type

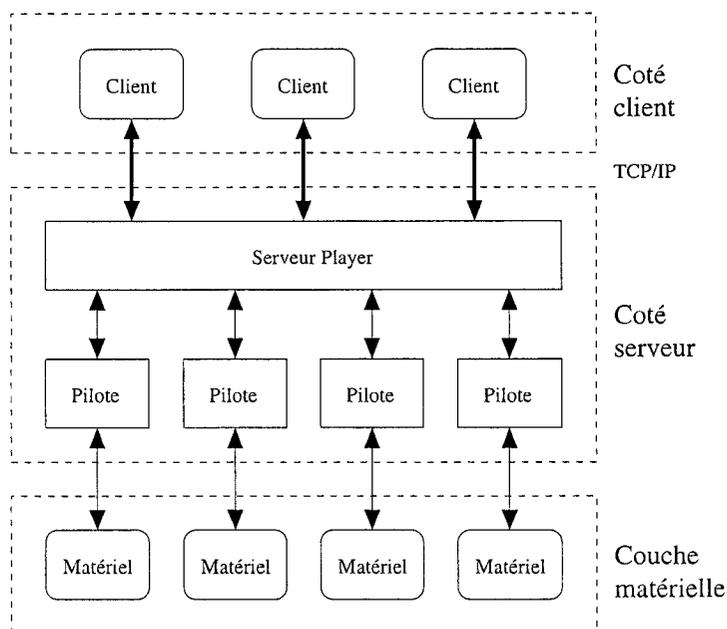


Figure II.1 Fonctionnement de Player

de matériel. Ainsi, si deux types différents de télémètres laser sont utilisés, chacun aura son propre pilote, mais les deux pilotes fourniront des données dans le même format. Les programmes clients peuvent se connecter au serveur Player par le protocole TCP/IP pour accéder aux données disponibles sur les interfaces désirées. Ainsi, les programmes clients n'ont pas à connaître directement le type de matériel utilisé, ni ses spécificités. Grâce à cette abstraction, l'ajout d'un nouveau matériel requiert théoriquement — ce fut plus compliqué dans le cas du capteur 3D — uniquement l'ajout du pilote approprié au serveur Player. Les clients peuvent alors utiliser ce nouveau matériel sans devoir subir de modification. La figure II.1 résume le fonctionnement de base de Player.

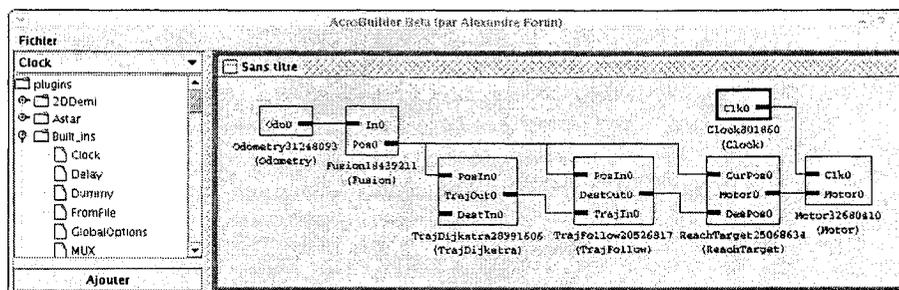


Figure II.2 Exemple d'un circuit Acropolis et de l'environnement Acrobuilder

### II.1.2 Couche d'abstraction logicielle

L'utilisation du serveur Player permet d'obtenir une abstraction efficace du matériel utilisé. Le GRPR a voulu étendre le concept d'abstraction au logiciel. Pour ce faire, l'architecture Acropolis a été créée. L'objectif d'Acropolis est d'encapsuler chaque algorithme dans une boîte noire appelée « module » et de standardiser les types de données échangées entre les modules. Chaque module définit son interface (les types de données de ses entrées et de ses sorties) et son comportement (le traitement effectué sur les entrées pour produire les sorties). Ainsi, il est possible d'avoir plusieurs modules possédant les mêmes interfaces, mais ayant des comportements différents. Acropolis permet d'agencer différemment, ou de remplacer, des comportements uniquement en modifiant un seul fichier de configuration. Un programme qui utilise l'architecture Acropolis peut être vu comme un circuit électronique. Chaque algorithme est encapsulé dans son module et les modules sont reliés entre eux par des fils qui représentent les transferts de données entre les modules. Cela permet une réutilisation rapide et facile des modules. L'auteur du présent mémoire a créé l'interface graphique Acrobuilder, qui est actuellement utilisée pour créer les fichiers de configuration utilisés par Acropolis. La figure II.2 présente un exemple de circuit Acropolis simple réalisé au moyen d'Acrobuilder.

Pour créer le lien entre les deux couches logicielles, la version d'Acropolis que nous utilisons intègre un client Player. En règle générale, à chaque interface de données de Player que nous utilisons correspond un module Acropolis qui assure la liaison entre le client Player et le reste du circuit d'Acropolis.

## II.2 Modules existants

Le développement logiciel relié à notre capteur 3D s'appuie sur des modules logiciels déjà existants au GRPR. Cette section résume les modules existants :

**Pilote du RDDC pour la plate-forme rotative** Recherche et Développement de la Défense du Canada nous a fourni le pilote Player utilisé pour leur modèle de plate-forme rotative. Toutefois, ce pilote n'est pas compatible avec Player 1.6 et a donc dû être adapté.

**Logiciel embarqué de SIL** La compagnie Scientific Instrumentation Limited fournit le micrologiciel qui doit être téléchargé sur la plate-forme rotative pour en assurer son fonctionnement. Ce programme, bien que fonctionnel, a dû être modifié pour ajouter plusieurs fonctionnalités.

**Librairie webcam** Une librairie pour l'acquisition d'images provenant d'une webcam a été conçue au GRPR par M. Vincent Zalzal. Cette librairie est utilisée sans modification.

## II.3 Modules développés

Normalement, l'ajout d'un nouveau matériel dans Player devrait être très facile, puisque cette opération devrait consister uniquement à créer un nouveau pilote. Toutefois, nous avons créé une nouvelle catégorie de capteur qui n'avait pas été

prévue initialement dans Player. Par conséquent, la tâche est beaucoup plus complexe. L'intégration du capteur à l'architecture logicielle demande de travailler et de communiquer entre 4 couches logicielles :

**Couche embarquée** La couche logicielle de plus bas niveau est le système embarqué qui gère la rotation du télémètre laser. Nous avons dû modifier le micrologiciel fourni par la compagnie SIL afin d'ajouter plusieurs fonctionnalités. Premièrement, nous avons ajouté une nouvelle requête permettant à la plate-forme d'atteindre une orientation désirée. Deuxièmement, nous avons ajouté un asservissement en position qui assure que la position atteinte est conservée malgré les vibrations induites par le déplacement de la plate-forme mobile. Troisièmement, nous avons adapté le micrologiciel pour supporter correctement le mode de fonctionnement à haute résolution du télémètre laser.

**Couche serveur de Player** La couche logicielle suivante est la couche de plus bas niveau située sur la plate-forme mobile. Il s'agit du pilote Player qui communique avec la couche embarquée. À ce niveau, il a été nécessaire de reprogrammer la communication avec le télémètre laser et avec la plate-forme rotative. Le code pour la communication avec le télémètre laser est inspiré en grande partie de celui fourni par Player alors que le code pour la plate-forme rotative s'inspire de celui du RDDC. Des modifications appropriées ont été apportées pour supporter les nouvelles fonctionnalités que nous avons ajoutées à la couche embarquée. C'est à ce niveau que se font les calculs pour associer les données photométriques aux données télémétriques. Les acquisitions 2D sont reçues à intervalles fixes de 52 ms, ce qui laisse suffisamment de temps entre chaque acquisition pour effectuer ces calculs. Les images sont acquises en utilisant la librairie développée au GRPR. Il n'y a pas de synchronisation entre le télémètre et la caméra. L'erreur maximale produite par

une éventuelle désynchronisation est de 1/30 de seconde, ce qui correspond à 1 pixel sur l'image de la caméra.

**Couche client de Player** Puisqu'il s'agit d'un nouveau type de capteur, le client Player ne possède pas de mandataire pouvant traiter le nouveau type de données. Ce mandataire est nécessaire pour faire le pont entre le serveur Player et Acropolis. Afin de réduire la quantité de données transmises par TCP/IP, seules les distances et couleurs sont transmises entre le serveur et le client. C'est la tâche du mandataire de calculer les coordonnées cartésiennes des points dans le repère lié au capteur 3D.

**Couche Acropolis** Un module Acropolis a été créé pour faire le pont entre la couche client de Player et les autres modules du circuit.

### ANNEXE III

## MODÈLES DE SURFACES COURBES UTILISÉS DANS CE MÉMOIRE

Cette annexe décrit les modèles de surfaces courbes utilisée dans ce mémoire : les sphères et les cylindres de révolution. Nous rappelons qu'un modèle doit comporter trois éléments :

1. Une paramétrisation de la surface mathématique ;
2. Une expression pour le calcul de l'erreur en distance d'un point à la surface modélisée ;
3. Une méthode pour calculer la première approximation des paramètres de la surface modélisée.

#### III.1 Modèle de sphère

Une équation mathématique d'une sphère quelconque dans l'espace peut être écrite sous la forme de l'équation III.1.

$$(x - h_S)^2 + (y - k_S)^2 + (z - l_S)^2 = r_S^2 \quad (\text{III.1})$$

Le modèle de la sphère requiert donc 4 paramètres : le centre  $(h_S, k_S, l_S)$  et le rayon  $r_S$ . Nous avons donc la paramétrisation suivante :

$$\vec{P}_S = (h_S, k_S, l_S, r_S) \quad (\text{III.2})$$

La distance d'un point  $p$  à la sphère est donnée par l'équation III.3 où  $c_S = (h_S, k_S, l_S)$ . La distance sera positive si le point  $p$  est situé à l'extérieur de la sphère et négative s'il est situé à l'intérieur.

$$\vec{e}_S(p) = \|\vec{c}_S \vec{p}\| - r_S \quad (\text{III.3})$$

La première approximation du centre de la sphère est obtenue en prenant le barycentre des points conformément à l'équation III.4. L'approximation du rayon est donnée par la moyenne des distances des points au centre, tel que décrit par l'équation III.5.

$$\hat{c}_S = (\hat{h}_S, \hat{k}_S, \hat{l}_S) = \frac{\sum_{i=1}^n p_i}{n} \quad (\text{III.4})$$

$$\hat{r}_S = \frac{\sum_{i=1}^n \|\vec{c}_S \vec{p}_i\|}{n} \quad (\text{III.5})$$

Cette première approximation fournirait de bons résultats si les points étaient répartis environ uniformément sur la sphère, ce qui, avec notre capteur, n'est pas le cas puisque celui-ci ne voit pas l'arrière des objets. Par conséquent, cette approximation donne une sphère de rayon beaucoup trop petit et dont le centre est très proche de la paroi numérisée, tel qu'illustré, pour le cas 2D d'un cercle, sur la figure III.1 (a). Dans ces conditions, les premières itérations de Levenberg-Marquardt ont tendance à vouloir à la fois agrandir le rayon et approcher davantage le centre de

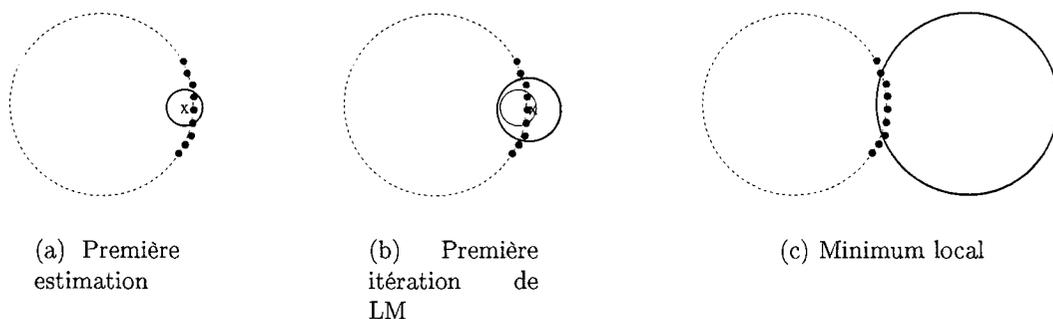


Figure III.1 Exemple de convergence vers un minimum local

la surface, tel que décrit à la figure III.1 (b). En conséquence, il arrive parfois que l'algorithme converge vers un minimum local tel qu'illustré en III.1 (c).

Pour remédier à cet inconvénient, il est nécessaire de raffiner cette approximation en utilisant les équations III.6 et III.7.

$$\check{r}_S = \max_i \left\| \overrightarrow{\check{c}_S p_i} \right\| \quad (\text{III.6})$$

$$\check{c}_S = (\check{h}_S, \check{k}_S, \check{l}_S) = \hat{c}_S + \frac{\sum_{i=1}^n \check{r}_S \left( \frac{p_i - \hat{c}_S}{\|\hat{c}_S p_i\|} \right)}{n} \quad (\text{III.7})$$

Cette transformation a pour effet d'éloigner le centre de la sphère de la paroi, et ce, dans la bonne direction. Les essais expérimentaux réalisés ont montré que cette nouvelle approximation améliore considérablement le taux de convergence de l'algorithme Levenberg-Marquardt.

Une autre méthode de calcul de la paramétrisation initiale a été envisagée. Nous savons que 4 points non coplanaires sont suffisants pour définir une sphère dans l'espace. Ainsi, nous pourrions tirer au hasard un certain nombre de quadruplets de

points, calculer les paramètres des sphères correspondantes et faire leur moyenne. Toutefois, à cause du bruit sur les données sources, cette solution est très peu fiable. Certains quadruplets peuvent même donner des centres situés du mauvais côté de la paroi. Pour obtenir une bonne estimation avec cette méthode, il faudrait choisir des points éloignés les uns des autres et effectuer une médiane sur les paramètres obtenus. Or, ces opérations sont beaucoup plus coûteuses en temps CPU que la méthode que nous avons proposée ci-dessus.

Nous avons maintenant les trois éléments requis pour le modèle de la sphère :

- La paramétrisation  $\vec{P}_S$  donnée par l'équation III.2 ;
- La formule de calcul de l'erreur  $\vec{e}_S$  donnée par l'équation III.3 ;
- La première estimation donnée par les formules III.6 et III.7.

### III.2 Modèle de cylindre

En milieu structuré, le cylindre est probablement le type de surface courbe le plus répandu, car colonnes, tuyaux, récipients et pieds de chaises ou de tables sont souvent de cette forme.

L'équation mathématique d'un cylindre d'orientation spatiale quelconque a la forme de l'équation III.8.

$$ax^2 + by^2 + cz^2 + dxy + exz + fyz + gx + hy + iz + j = 0 \quad (\text{III.8})$$

L'équation III.8 est en fait l'équation d'une quadrique quelconque. Pour avoir un cylindre, il est nécessaire d'avoir certaines relations spécifiques entre les 10 para-

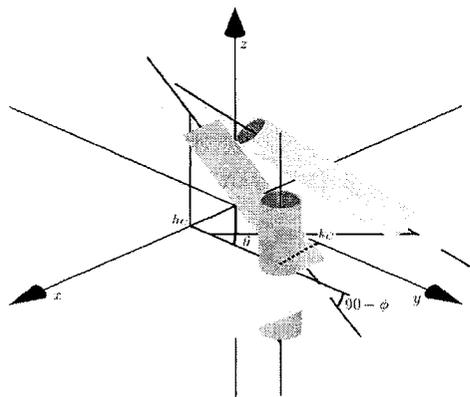


Figure III.2 Paramétrisation d'un cylindre

mètres de cette équation<sup>1</sup>. Minimiser directement l'équation III.8 fournirait donc une réponse qui ne serait probablement pas un cylindre.

Tout cylindre de l'espace de rayon  $r_C$  peut être ramené à un cylindre de rayon  $r_C$  dont l'axe de révolution coïncide avec l'axe  $z$  en appliquant trois transformations. Une manière d'y arriver est illustrée à la figure III.2. Une rotation d'angle  $\theta_C$  autour de l'axe  $z$  permet de ramener le cylindre parallèle au plan  $yz$  et une rotation d'angle  $\phi_C$  le ramène parallèle à l'axe  $z$ . Une translation de  $(h_C, k_C)$  fait coïncider l'axe de révolution du cylindre avec l'axe  $z$ .

La paramétrisation du cylindre est alors donnée par la formule III.9.

$$\vec{P}_C = (r_C, \theta_C, \phi_C, h_C, k_C) \quad (\text{III.9})$$

<sup>1</sup>Veillez noter que seuls 9 paramètres sont indépendants, mais qu'il est nécessaire de considérer l'équation à 10 paramètres pour avoir une équation qui fonctionne dans toutes les situations.

La distance d'un point  $p$  à la surface du cylindre est en fait la distance du point à l'axe du cylindre à laquelle on retranche le rayon du cylindre. Cette distance est donnée par l'équation III.10. La distance sera positive si le point est à l'extérieur du cylindre et négative s'il est à l'intérieur.

$$e_C(p) = \|(\overrightarrow{c_C p}) \times \vec{v}_C\| - r_C \quad (\text{III.10})$$

Dans l'équation III.10,  $\vec{v}_C$  représente un vecteur directeur unitaire de l'axe du cylindre et est donné par l'équation III.11 alors que  $c_C$  est le point de l'axe du cylindre le plus proche de l'origine et est donné par l'équation III.12.

$$\vec{v}_C = \begin{pmatrix} \cos \theta_C \sin \phi_C \\ \sin \theta_C \sin \phi_C \\ \cos \phi_C \end{pmatrix} \quad (\text{III.11})$$

$$c_C = \begin{pmatrix} h_C \cos \theta_C \cos \phi_C - k_C \sin \theta_C \\ h_C \sin \theta_C \cos \phi_C + k_C \cos \theta_C \\ -h_C \sin \phi_C \end{pmatrix} \quad (\text{III.12})$$

Pour trouver la première approximation des paramètres du cylindre, nous utilisons l'information sur l'intersection des deux plans. En effet, les plans trouvés doivent avoir un vecteur normal perpendiculaire à l'axe du cylindre, car s'il en était autrement, alors ces plans ne minimiseraient pas l'erreur de distance discutée à la section 3.1.1. Par conséquent, la droite d'intersection des deux plans doit être parallèle à l'axe du cylindre. Nous obtenons ainsi  $\hat{v}_C = (\hat{v}_{C\vec{i}}, \hat{v}_{C\vec{j}}, \hat{v}_{C\vec{k}})$ . Les paramètres  $\hat{\theta}_C$  et  $\hat{\phi}_C$  sont ensuite obtenus en utilisant les équations III.13 et III.14.

$$\hat{\theta}_C = \arctan \frac{\hat{v}_{C\vec{j}}}{\hat{v}_{C\vec{i}}} \quad (\text{III.13})$$

$$\hat{\phi}_C = \arccos \frac{\hat{v}_{C\vec{k}}}{\|\hat{\vec{v}}_C\|} \quad (\text{III.14})$$

Nous appliquons ensuite des rotations de  $\hat{\theta}_C$  et de  $\hat{\phi}_C$  aux points  $p_i$  pour obtenir des points  $p'_i$  situés sur un cylindre parallèle à l'axe  $z$ . Nous projetons ensuite les points sur le plan  $xy$  au moyen de la projection de l'équation III.15, ce qui donne des points cocirculaires.

$$p''_i = p'_i \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \quad (\text{III.15})$$

La première approximation des paramètres  $\hat{h}_C$ ,  $\hat{k}_C$  et  $\hat{r}_C$  est alors obtenue au moyen de ces points exactement de la même manière que pour le calcul de la sphère. Les calculs exacts sont repris aux équations III.16 et III.17.

$$\hat{c} = (\hat{h}_C, \hat{k}_C) = \frac{\sum_{i=1}^n p''_i}{n} \quad (\text{III.16})$$

$$\hat{r}_C = \frac{\sum_{i=1}^n \|\hat{c}p''_i\|}{n} \quad (\text{III.17})$$

Cette première approximation offre le même problème de convergence vers un minimum local que celui discuté dans le cas de la sphère. Nous appliquons la même

solution (éq. III.18 et III.19).

$$\check{r}_C = \max_i \left\| \frac{\vec{p}_i''}{\check{c}p_i} \right\| \quad (\text{III.18})$$

$$\check{c} = (\check{h}_C, \check{k}_C) = \hat{c} + \frac{\sum_{i=1}^n \check{r}_C \left( \frac{p_i'' - \hat{c}}{\check{c}p_i} \right)}{n} \quad (\text{III.19})$$

Nous avons maintenant les trois éléments requis pour le modèle du cylindre :

- La paramétrisation  $\vec{P}_C$  donnée par l'équation III.9 ;
- La formule de calcul de l'erreur  $\vec{e}_C$  donnée par l'équation III.10 ;
- La première estimation des paramètres donnée par les formules III.13 à III.19.

## ANNEXE IV

### SEGMENTATION D'IMAGES PROVENANT D'UNE SCÈNE SIMULÉE

Cette annexe présente les résultats obtenus en appliquant la technique de Hoover et al. (Hoover et al., 1996) pour évaluer les performances de notre algorithme de segmentation. Nous débutons par un rappel des métriques utilisées et des simulations effectuées. Puis nous présentons et analysons les résultats obtenus. Nous terminons par une comparaison de notre technique de segmentation avec d'autres techniques que l'on retrouve dans la littérature scientifique.

#### IV.1 Métriques utilisées pour l'analyse

La méthode d'analyse originale proposée par Hoover et al. (Hoover et al., 1996) consiste à prendre des images réelles et à les segmenter de deux manières différentes. La première segmentation est effectuée à la main et est considérée comme la segmentation idéale, c'est-à-dire l'objectif à atteindre. La seconde segmentation est effectuée automatiquement par l'algorithme de segmentation analysé. Plutôt que de prendre des images réelles, nous utilisons des images simulées. Le simulateur, en plus de nous fournir les images 3D, nous fournit également la segmentation de référence, ce qui retire le risque d'erreurs liées à une mauvaise segmentation manuelle de la part de l'utilisateur et diminue la subjectivité. Chaque segment est classé dans l'une des cinq classes suivantes : bien détecté, segmentation excessive, segmentation insuffisante, non détecté et en trop. Nous résumons ici les caractéristiques permettant d'effectuer ce classement. Une description plus complète de la

méthode est fournie dans l'article de Hoover et al. (Hoover et al., 1996). Un paramètre  $\tau \in ]0, 5, 1]$  permet de choisir le degré de similarité minimal requis pour que deux éléments surfaciques soient considérés identiques.

**Classe A : bien détecté** Un segment de la segmentation automatique est une bonne détection d'un élément surfacique de la segmentation de référence si les deux conditions suivantes sont remplies :

1. Une proportion supérieure ou égale à  $\tau$  des points du segment appartient à l'élément surfacique ;
2. Une proportion supérieure ou égale à  $\tau$  des points de l'élément surfacique appartient au segment.

**Classe B : segmentation excessive** Un élément surfacique de la segmentation de référence est excessivement segmenté en un ensemble de segments de la segmentation automatique si les deux conditions suivantes sont remplies :

1. Une proportion supérieure ou égale à  $\tau$  des points de chaque segment appartient à l'élément surfacique ;
2. Une proportion supérieure ou égale à  $\tau$  des points de l'élément surfacique appartient à l'union des segments de l'ensemble.

**Classe C : Segmentation insuffisante** Un ensemble d'éléments surfaciques de la segmentation de référence est insuffisamment segmenté en un segment dans la segmentation automatique si les deux conditions suivantes sont remplies :

1. Une proportion supérieure ou égale à  $\tau$  des points du segment appartient à l'union des éléments surfaciques ;
2. Une proportion supérieure ou égale à  $\tau$  des points de chaque élément surfacique de l'ensemble appartient au segment.

**Classe D : Non détecté** Un élément surfacique de la segmentation de référence est non détecté s'il n'appartient à aucune des catégories énumérées ci-dessus.

**Classe E : En trop** Un segment de la segmentation automatique est en trop s'il n'appartient à aucune des catégories énumérées ci-dessus.

Le simulateur nous fournit directement l'orientation dans l'espace de toutes les surfaces. Nous pouvons ainsi mesurer directement l'erreur d'orientation entre les surfaces de la segmentation automatique et celles de la segmentation de référence.

Nous ajoutons à nos outils d'analyse deux métriques additionnelles :

- L'erreur spatiale : distance entre les points et les surfaces obtenues ;
- L'erreur photométrique : différence entre les couleurs des points et les couleurs appliquées aux segments.

La première métrique consiste à calculer, en millimètres, l'erreur de distance de chaque point à la surface à laquelle il a été assigné. La seconde métrique consiste à calculer l'erreur de distance euclidienne RVB de la couleur de chaque point à la couleur du segment auquel a été assigné le point. Chaque canal de couleur peut posséder une valeur entière dans l'intervalle  $[0, 255]$ , l'erreur sur la couleur peut donc prendre des valeurs discrètes dans l'intervalle  $[0, 255\sqrt{3}]$ .

## IV.2 Caractéristiques des simulations effectuées

Dans toutes les simulations que nous avons effectuées, la scène est parfaite. C'est-à-dire qu'elle est constituée de sections de plans, de cylindres et de sphères parfaits. Les caractéristiques du capteur et de l'algorithme sont modifiées au cours de trois séries de simulations :

**Série 1** La première série de simulations est effectuée en considérant un capteur et un algorithme parfaits. L'objectif de cette série de simulations est de valider le fonctionnement de l'algorithme dans le cas idéal.

**Série 2** La deuxième série de simulations est effectuée avec un capteur parfait et un algorithme qui prend pour acquis que le capteur est imparfait. L'objectif de cette série de simulations est de déterminer l'effet des paramètres utilisés par l'algorithme sur les performances obtenues dans un cas où le capteur est idéal.

**Série 3** La troisième série de simulations est effectuée avec une lumière ambiante et dirigée, un capteur bruité et un algorithme qui prend pour acquis que le capteur est imparfait. Il s'agit du mode de simulation le plus proche de la situation réelle. Il faut noter que nous avons simulé un bruit de mesure beaucoup plus intense que celui retrouvé en réalité.

Les caractéristiques précises de ces simulations sont données au tableau IV.1. Rappelons que  $d_{max}$  est la distance maximale tolérée entre un point et la surface à laquelle il est assigné,  $N_{min}$  est le nombre minimal de points que doit posséder une surface pour être conservée,  $S_D$  est le seuil utilisé pour le filtrage spatial servant à éliminer les points fantômes,  $S_P$  est le seuil appliqué à l'image de contour avant l'exécution de la ligne de partage des eaux et  $N_{iter}$  est le nombre d'itérations réalisées pour l'extraction des surfaces planes.

Pour les trois modes de simulation, les mêmes numérisations (mêmes scènes et mêmes points de vue) ont été effectuées de manière à pouvoir comparer correctement les différentes simulations entre elles. Une scène contenant une table, une chaise, des boîtes et des colonnes a été créée dans le simulateur. Une illustration de cette scène est présentée à la figure IV.1, page 204. La figure IV.1 (a) illustre la disposition des éléments vus de haut alors que les figures IV.1 (b) et (c) montrent trois

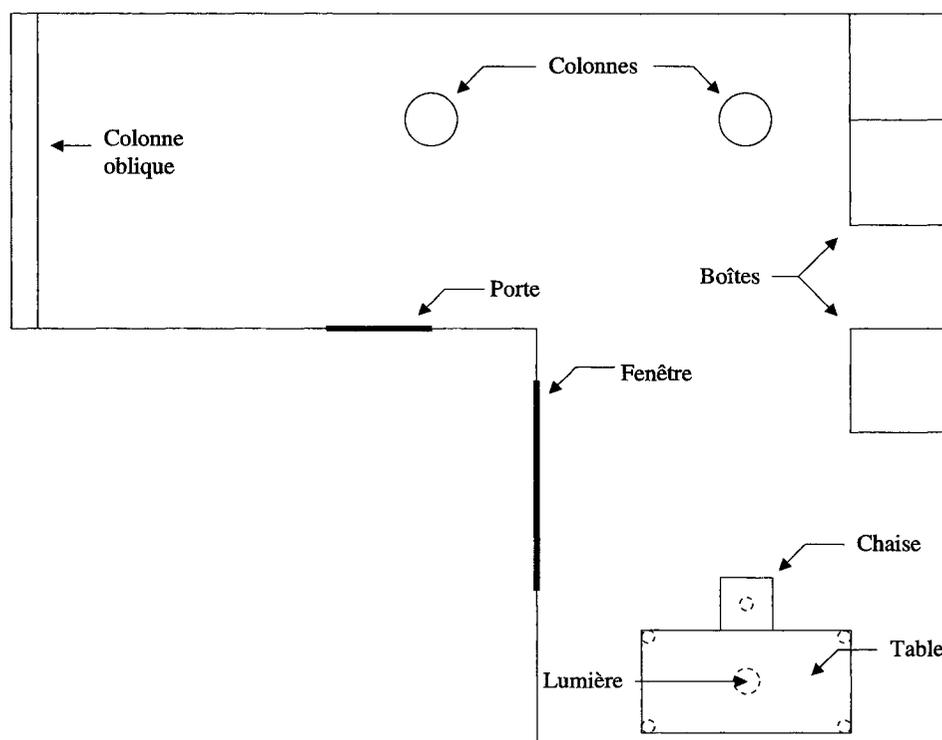
Paramètre	Série 1	Série 2	Série 3
Lumière ambiante	100%	100%	65%
Lumière dirigée	0%	0%	65%
Bruit de mesure	0 m	0 m	0,03 m
$d_{max}$	0,001 m	0,03 m	0,03 m
$N_{min}$	3 points	100 points	100 points
$S_D$	$\infty$ (Désactivé)	0,2 m	0,2 m
Filtre gaussien	Désactivé	Activé	Activé
$S_P$	6	6	6
$N_{iter}$	600 it.	600 it.	600 it.

Tableau IV.1 Caractéristiques des simulations effectuées

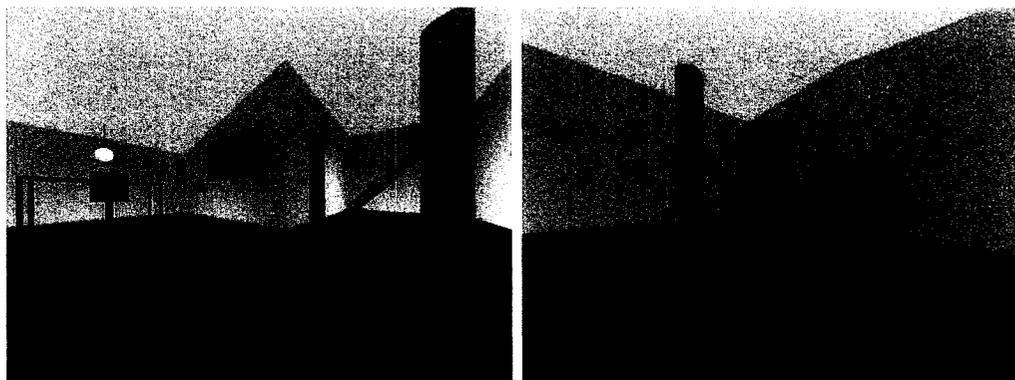
vues différentes de cette scène. Ces images sont déformées, à cause de l'angle d'ouverture de la caméra OpenGL simulée qui est de  $90^\circ$ <sup>1</sup>. Cette déformation n'affecte pas la précision des valeurs simulées.

Dix points de vue différents de cette pièce ont été choisis manuellement afin de s'assurer d'avoir des points de vue variés. Dix autres points de vue ont été générés de manière aléatoire afin de diminuer le risque d'avoir des points de vue biaisés par l'utilisateur. Pour chaque point de vue, des numérisations ont été effectuées à trois hauteurs différentes : 0,7 m, 1,5 m et 2,3 m. En choisissant ces trois hauteurs, nous diminuons le risque d'obtenir des résultats biaisés par la présence de surfaces quasi parallèles au capteur. Choisir uniquement une hauteur de 0,7 m serait plus représentatif du cas d'utilisation du capteur réel, mais l'objectif de cette section est d'évaluer l'efficacité de la méthode de segmentation indépendamment de l'application. Le même capteur pourrait d'ailleurs être éventuellement utilisé sur une plate-forme mobile ayant des dimensions physiques différentes. Il pourrait également s'agir d'un capteur différent fournissant des données dans le même format.

<sup>1</sup>Cet angle est nécessaire pour réaliser les simulations.



(a) Vue de dessus



(b) Vue 1

(c) Vue 2

Figure IV.1 Scène simulée pour valider la méthode de segmentation

Pour une série de simulations, 60 images représentant 444 éléments surfaciques sont d'abord simulées puis segmentées. Les 360 Mo de données ainsi obtenues sont analysées pour en évaluer les métriques mentionnées ci-dessus. Les résultats obtenus sont présentés aux sections suivantes.

### IV.3 Détection des éléments surfaciques

Les résultats généraux obtenus en fonction du paramètre  $\tau$ , qui représente le degré de similitude requis pour avoir une bonne détection, sont présentés au graphique IV.2. Afin d'alléger le graphique, nous avons volontairement omis les classes B et C, c'est-à-dire les éléments surfaciques excessivement segmentés et insuffisamment segmentés. La proportion de ces surfaces représente moins de 2 % des surfaces détectées. Les performances varient faiblement et de manière linéaire jusqu'à une valeur de  $\tau$  située entre 0,8 et 0,85. Nous ne sommes actuellement pas en mesure d'indiquer pourquoi la valeur  $\tau=0,8$  est une valeur critique, mais nous retrouvons exactement la même valeur dans les résultats fournis par Hoover et al. et ce, pour la majorité des segmenteurs qui y sont présentés<sup>2</sup>.

Les proportions indiquées sur le graphique IV.2 sont calculées par rapport à la valeur attendue du nombre d'éléments surfaciques bien détectés, soit 444. Nous observons que, pour la première série de simulations, la proportion d'éléments de la classe E, c'est-à-dire les segments en trop, est supérieure à la quantité d'éléments surfaciques devant être détectés ce qui donne un taux de segments détectés en trop d'environ 152 %. Toutefois, ce taux est d'environ 10 % pour les deux autres séries. Cet écart s'explique par la valeur du paramètre  $N_{min}$ , le nombre de points minimum que doit posséder un segment pour être retenu. Cette valeur était fixée à 3 points

---

<sup>2</sup>Il faut également noter que l'algorithme de comparaison a été entièrement recodé, de sorte qu'il est très peu probable que ce soit une erreur de programmation.

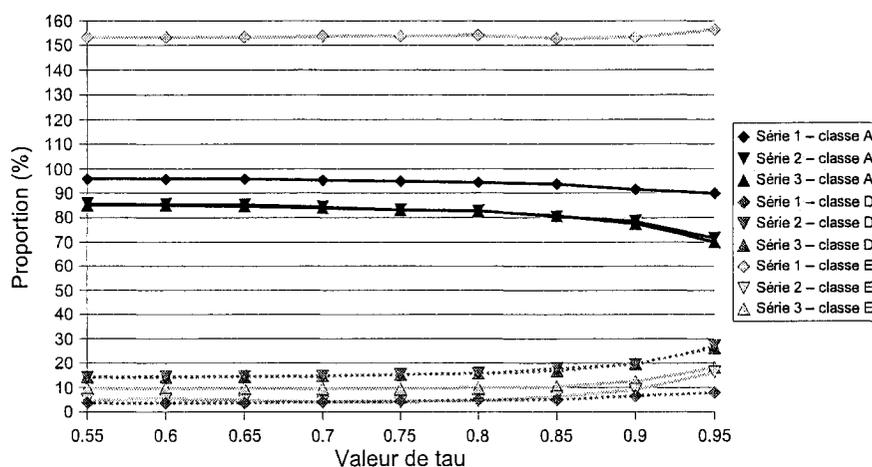


Figure IV.2 Classification des éléments surfaciques en fonction de  $\tau$

Les numéros des séries réfèrent au tableau IV.1 présenté à la page 203 alors que les lettres des classes de surfaces réfèrent à la description présentée à la page 200.

pour la première série de simulations et à 100 points pour les séries 2 et 3. Le but de ce paramètre est justement de diminuer le nombre de petits segments en trop ; le graphique IV.2 illustre que ce but est atteint. Le graphique montre également une baisse de performances d'environ 10 %, en comparant les proportions de surfaces de la classe A, entre la série 1 et les séries 2 et 3. Cette baisse est due au fait que les petites surfaces de taille inférieure à  $N_{min}$  points ne sont plus détectées. Une dernière observation très importante est que les séries 2 et 3 donnent pratiquement les mêmes performances. L'algorithme utilisé est donc très résistant au bruit de mesure et à la luminosité ambiante.

Pour la suite de l'analyse des résultats, nous utiliserons uniquement la valeur  $\tau=0,8$ . Cette valeur de 80 % est celle utilisée par Hoover et al. pour comparer les algorithmes entre eux. Les résultats du graphique IV.2 sont présentés de manière plus détaillée pour la valeur  $\tau=0,8$  à la figure IV.3.

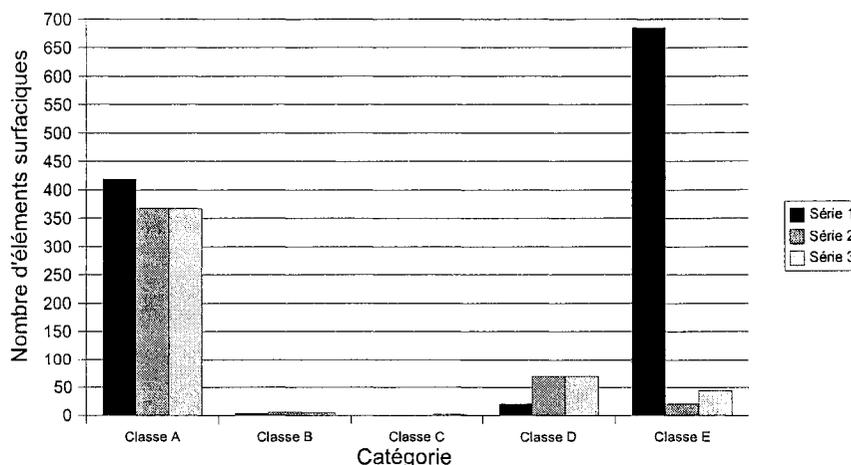


Figure IV.3 Classification des éléments surfaciques pour  $\tau=0,8$

Pour les éléments surfaciques mal détectés, c'est-à-dire les éléments des classes B, C, D et E, nous avons analysé le nombre de ceux-ci en fonction de leur taille. Nous évaluons la taille d'un élément surfacique par le nombre de points qui sont associés à cet élément. Ces résultats sont présentés au tableau IV.2, page 208. Nous remarquons que, pour la série 1, la majorité des segments appartenant à la classe E, c'est-à-dire les segments détectés en trop, ont une taille inférieure à 10 points. En fait, ces régions se situent aux frontières des grandes régions détectées. Ces frontières ne sont pas identiques dans la segmentation télémétrique et dans la segmentation photométrique et si, par endroits, la position de ces frontières diffère, ne serait-ce que de 1 pixel, il y a création de minuscules fausses surfaces lors de la fusion des segmentations.

De plus, la majorité des éléments surfaciques non détectés, c'est-à-dire les éléments de la classe D, ont un nombre de points inférieur au paramètre  $N_{min}$ . Ces régions trop petites n'ont pas réellement d'intérêt, car leur trop faible nombre de points empêche d'avoir des informations précises sur ces régions. Si nous retirons ces régions

	Série 1				Série 2				Série 3			
	Classe B	Classe C	Classe D	Classe E	Classe B	Classe C	Classe D	Classe E	Classe B	Classe C	Classe D	Classe E
[0,9]	0	0	10	659	0	0	28	0	0	0	28	0
[10,19]	0	0	1	21	0	0	6	0	0	0	6	0
[20,29]	0	0	3	1	0	0	5	0	0	0	5	0
[30,39]	0	0	1	0	0	0	4	0	0	0	4	0
[40,49]	0	0	0	1	0	0	0	0	0	0	0	0
[50,59]	0	0	2	1	0	0	5	0	0	0	5	0
[60,69]	0	0	0	1	0	0	1	0	0	0	1	0
[70,79]	0	0	0	0	0	0	0	0	0	0	0	0
[80,89]	0	0	0	2	0	0	1	0	0	0	1	0
[90,99]	0	0	0	0	0	0	1	0	0	0	1	0
[100,+]	4	0	4	3	6	0	19	20	5	2	19	44

Tableau IV.2 Répartition des éléments surfaciques mal segmentés et en trop en fonction de leur taille

trop petites des résultats présentés à la figure IV.3, c'est-à-dire si nous considérons uniquement les éléments surfaciques de taille supérieure à 100 points<sup>3</sup>, nous obtenons les résultats fournis au tableau IV.3.

Nous notons également au tableau IV.2 que, pour les séries 1 et 2, c'est-à-dire sans bruit de mesure, le nombre d'éléments surfaciques non détectés (classe D) de plus de 100 points est sensiblement égal au nombre de segments en trop (classe E) de plus de 100 points. Cela s'explique par le fait que plusieurs régions en trop sont en fait les régions non détectées, mais pour lesquelles la proportion de points correctement assignés est inférieure à  $\tau$ . Ces surfaces sont alors considérées comme étant de nouvelles surfaces. Pour la série 3, nous avons une plus grande proportion de régions en trop en raison de l'erreur que nous avons ajoutée sur le capteur.

<sup>3</sup>En fixant le paramètre  $N_{min} = 100$ , nous acceptons de ne pas détecter les éléments surfaciques ayant moins de 100 points. Il nous paraît donc pertinent de ne pas tenir compte de ces éléments

	Série 1	Série 2	Série 3
Bonnes détections (classe A)	98,0 %	93,6 %	93,4 %
Mauvaises détections (classes B, C et D)	2,0 %	6,4 %	6,6 %
Éléments surfaciques en trop (classe E)	0,8 %	5,4 %	12,0 %

Tableau IV.3 Taux de réussite pour les éléments surfaciques de plus de 100 points pour  $\tau=0,8$

La similitude entre les deux dernières colonnes du tableau IV.3 montre, une fois de plus, que le résultat de la segmentation reste pratiquement inchangé sous l'effet du bruit et de la lumière. Le seul impact du bruit de mesure est l'augmentation du nombre de segments détectés en trop.

#### IV.4 Erreurs spatiale, photométrique et angulaire

Pour les surfaces qui ont été bien détectées, nous évaluons les métriques d'erreur suivantes :

- Erreur spatiale;
- Erreur photométrique;
- Erreur angulaire.

Nous rappelons ici brièvement en quoi consistent ces métriques qui ont déjà été expliquées en détails à la section IV.1. L'erreur spatiale est la distance entre les points et les surfaces auxquelles ils ont été attribués par la technique de segmentation. L'erreur photométrique est la distance dans l'espace RVB entre la couleur des points et celle des segments auxquels ils ont été attribués. Finalement, l'erreur angulaire des surfaces est la différence d'orientation de la surface obtenue par notre 

---

 surfaciques dans l'évaluation finale des performances de notre technique de segmentation.

Métrique	Série 1	Série 2	Série 3
Erreur spatiale (mm)	0,042 (0,231)	1,80 (4,11)	11,12 (9,00)
Erreur photométrique	1,12 (8,79)	1,36 (11,05)	5,81 (10,44)
Erreur angulaire (°)	0,005 (0,039)	0,30 (0,89)	0,99 (1,83)

Tableau IV.4 Erreurs moyennes et écarts-types

Les valeurs entre parenthèses sont les écarts-types.

technique de segmentation par rapport à l'orientation réelle de la surface fournie par le simulateur.

Pour ces métriques, nous avons calculé les valeurs moyennes et les écarts-types que nous avons affichés dans le tableau IV.4. Ces résultats sont détaillés ci-dessous.

#### IV.4.1 Erreur spatiale

Afin de caractériser l'erreur spatiale plus précisément, nous avons tracé au graphique IV.4 la répartition de cette erreur pour la série 2. Rappelons que cette série consiste à simuler un capteur parfait et à utiliser un algorithme qui considère un capteur imparfait dans le but de détecter les erreurs induites par l'algorithme de segmentation. Cette figure montre que la répartition de l'erreur télémétrique n'est pas la même pour les plans que pour les objets courbes. Alors que seulement 75 % des points des plans sont à moins de 2 mm de la surface plane qui leur a été associée, cette proportion est pratiquement de 100 % pour les surfaces courbes. L'explication de ce phénomène est que les objets courbes numérisés ont peu de frontières communes avec d'autres objets et sont par conséquent moins contaminés par des points appartenant à une autre surface. À l'opposé, toutes les surfaces planes numérisées

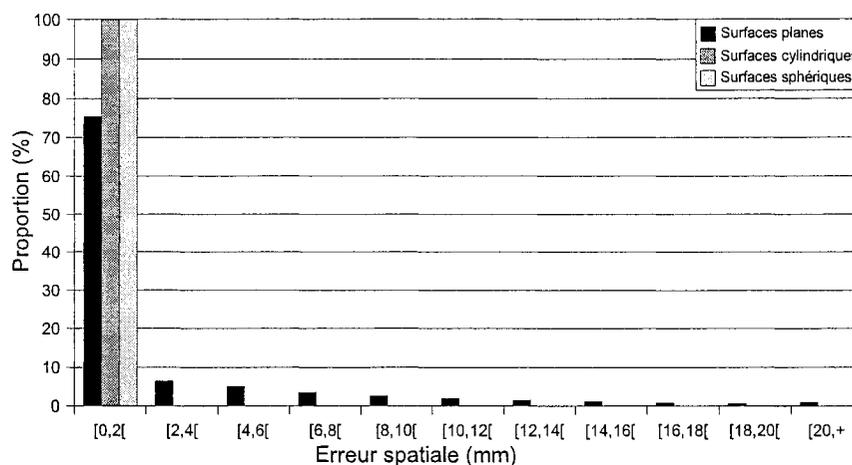


Figure IV.4 Distribution de l'erreur spatiale pour la série 2 avec  $\tau=0,8$

avaient de grandes intersections avec au moins quatre autres surfaces<sup>4</sup>. De plus, les surfaces planes adjacentes aux surfaces courbes sont généralement plus grandes de sorte qu'elles sont détectées en premier lors de l'extraction des surfaces planes. Or, ce sont les surfaces détectées en premier qui sont les plus contaminées par des points appartenant aux autres surfaces. En effet, une fois les points de ces surfaces retirés, ils ne peuvent plus contaminer les autres surfaces.

La distribution de l'erreur en distance pour la série 3 prend la forme affichée à la figure IV.5. Rappelons que la série 3 consistait, conformément au tableau IV.1, à simuler un capteur imparfait et à utiliser un algorithme qui suppose que le capteur est imparfait. Afin de bien analyser cette distribution, il faut la comparer à celle qui serait obtenue si la segmentation était parfaite. La figure IV.6 présente cette comparaison. Pour obtenir ces données, nous avons confondu tous les types de surfaces et avons calculé l'erreur télémétrique entre les points et les surfaces obtenues par segmentation, ce qui donne la distribution de l'erreur obtenue, et entre les mêmes

<sup>4</sup>Un mur touche généralement à deux autres murs, au plancher et au plafond.

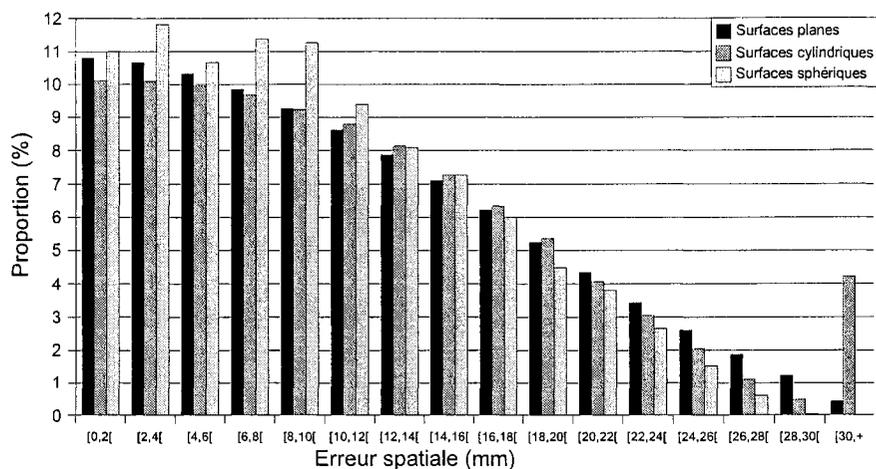


Figure IV.5 Distribution de l'erreur télémétrique obtenue pour  $\tau=0,8$

points et les surfaces fournies par le simulateur, ce qui donne la distribution de l'erreur optimale.

La comparaison de la figure IV.6 montre que la distribution de l'erreur en distance suit la distribution du bruit appliqué aux données du capteur. L'erreur causée par le processus de segmentation est très faible. En effet, si cette erreur était élevée, nous observerions un décalage vers la droite de la distribution obtenue par rapport à la distribution optimale, ce qui n'est pas le cas. Ceci suggère que l'erreur due à l'algorithme de segmentation doit suivre de près la distribution de la figure IV.4.

#### IV.4.2 Erreur photométrique

La distribution de l'erreur photométrique est présentée à la figure IV.7. Les distributions des séries 1 et 2 sont quasi identiques. Ce résultat, ainsi que la moyenne du

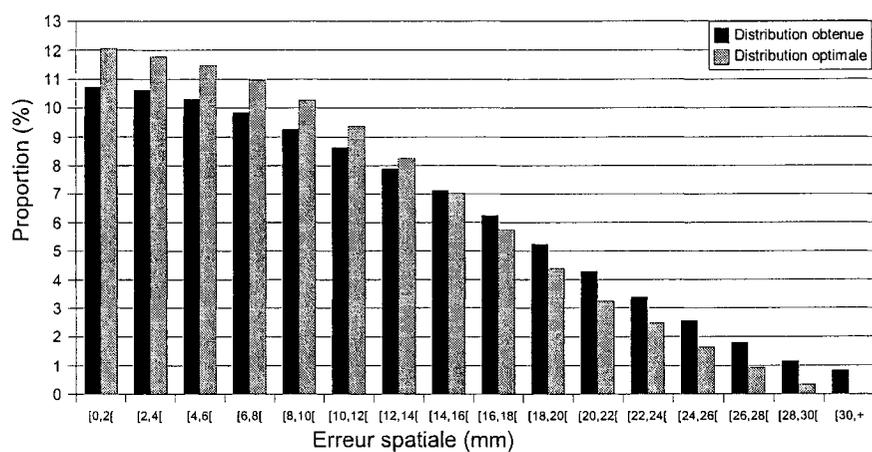


Figure IV.6 Comparaison entre l'erreur télémétrique obtenue et le bruit ajouté

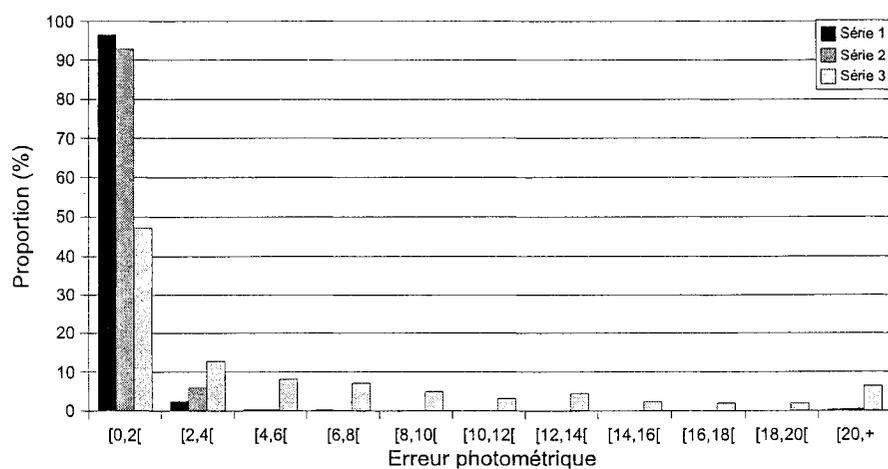


Figure IV.7 Distribution de l'erreur en couleur pour  $\tau=0,8$

tableau IV.4 de la page 210, démontrent que les paramètres  $d_{max}$ <sup>5</sup>,  $N_{min}$ <sup>6</sup> et le filtre gaussien appliqué sur l'image ont peu d'effet sur l'erreur photométrique. Pour les séries 1 et 2, la proportion de points qui ont une erreur supérieure à 4 est presque nulle. Les écarts-types observés au tableau IV.4 sont donc fortement influencés par des segments de faible taille ayant une erreur photométrique importante. De telles régions peuvent se retrouver aux frontières entre deux éléments surfaciques. L'étalement de la série 3 observé à la figure IV.7 est dû à l'ajout de la lumière.

#### IV.4.3 Erreur angulaire

La figure IV.8 illustre l'erreur angulaire pour les séries de simulations 2 et 3. Afin d'alléger le graphique, la série 1 a été volontairement omise. Toutes les surfaces de la série 1 avaient une erreur d'orientation inférieure à  $0,5^\circ$ . L'erreur angulaire entre deux surfaces planes est calculée en mesurant l'angle entre les vecteurs normaux des plans et l'erreur angulaire entre deux surfaces cylindriques est obtenue en mesurant l'angle entre les axes des cylindres. L'erreur angulaire n'a pas de sens pour les surfaces sphériques et n'est donc pas calculée.

Nous remarquons que l'erreur d'orientation des cylindres varie très peu lorsque nous ajoutons au capteur l'erreur de mesure en distance. Toutefois, l'erreur d'orientation des surfaces planes est plus vulnérable. Ce phénomène s'explique par une combinaison de plusieurs facteurs :

- Il y a plus de petites surfaces planes que de petites surfaces cylindriques. Les orientations des petites surfaces sont beaucoup plus sensibles à une erreur de mesure que celles des grandes surfaces.
- L'erreur n'est pas répartie de la même manière sur les cylindres que sur les plans

---

<sup>5</sup>Distance maximale permise pour qu'un point appartienne à une surface.

<sup>6</sup>Nombre de points minimal que doit posséder une surface.

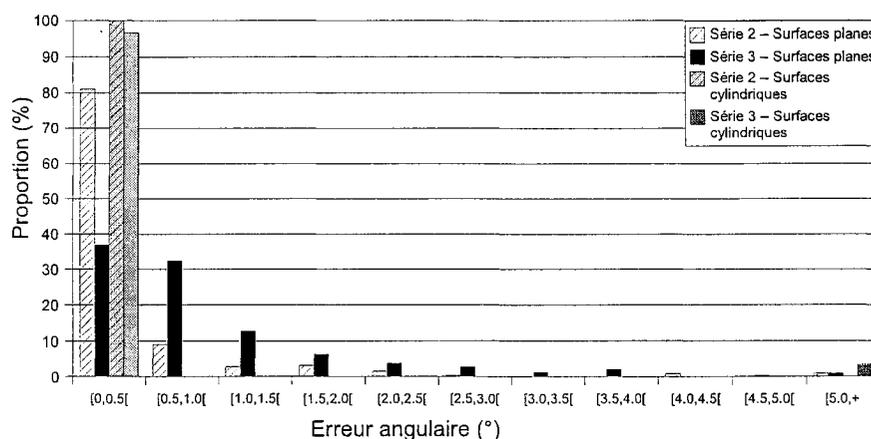


Figure IV.8 Distribution de l'erreur d'orientation pour  $\tau=0,8$

à cause de leur courbure. En effet, l'erreur en distance du capteur est dans la direction de propagation du faisceau laser. Or, le faisceau laser ne frappe pas tous les points du cylindre avec le même angle d'incidence. Par conséquent, l'erreur par rapport à la surface du cylindre, laquelle est normale à la surface, n'est pas uniforme : elle est maximale lorsque le faisceau du laser touche la paroi du cylindre à  $90^\circ$  et plus faible dans les autres cas<sup>7</sup>.

- Les surfaces courbes sont très peu contaminées par des points appartenant à d'autres surfaces.

Il est intéressant de voir que même si la détection des surfaces cylindriques est obtenue à partir d'une détection de surfaces planes moins précise, l'algorithme de minimisation de Levenberg-Marquardt parvient à obtenir une solution quasi optimale.

<sup>7</sup>Il faut noter que l'erreur n'est pas uniforme sur les plans non plus. Toutefois, contrairement aux cylindres, il n'y a pas d'effet dû à la courbure de la surface ; il y a seulement un effet dû à la rotation du laser.

#### IV.5 Comparaison avec d'autres travaux

Nous avons présenté les résultats de notre segmentation et les avons analysés afin de faire ressortir les facteurs ayant une influence sur les performances du système. Afin de leur donner plus de signification, nous avons comparé notre algorithme de segmentation à d'autres algorithmes que l'on retrouve dans la littérature scientifique.

Il convient de préciser que cette comparaison ne doit pas être perçue comme une comparaison objective prouvant hors de tout doute la supériorité d'une technique par rapport aux autres. En effet, pour ces comparaisons, nous n'avons pas utilisé les mêmes bancs d'essais que les autres algorithmes avec lesquels nous effectuons les comparaisons, car ces bancs d'essais sont constitués uniquement de surfaces planes et ne comportent pas d'informations photométriques. Nous présentons cette comparaison à titre approximatif uniquement afin de donner un aperçu des performances de notre méthode.

Les résultats de cette comparaison sont résumés au tableau IV.5, à la page 218. Les noms des techniques de segmentation sont ceux utilisés par les auteurs dans les ouvrages référencés. Les abréviations « PERC » et « ABW » réfèrent au banc de test utilisé. Afin d'effectuer une juste comparaison, les résultats que nous avons inscrits pour notre technique prennent en considération toutes les surfaces de plus de 10 points<sup>8</sup>. Les valeurs incrites dans le tableau sont celles fournies par les auteurs dans les ouvrages référencés alors que les pourcentages indiqués entre parenthèses ont été calculés par l'auteur du présent document. Pour l'erreur angulaire, la valeur entre parenthèses est l'écart type. Les classes de surfaces réfèrent à celles présentées

---

<sup>8</sup>Dans la méthode de Hoover et al., les images de référence sont segmentées à la main et les points n'appartenant à aucune surface apparente ne sont pas pris en considération. Nous considérons qu'une surface de moins de 10 points n'aurait pas été prise en compte par leur méthode.

à la page 200.

La dernière ligne du tableau IV.5 montre la position de notre technique par rapport aux autres. Cette comparaison est très prometteuse. Toutefois, une comparaison plus rigoureuse devrait être effectuée avant de pouvoir poser un jugement définitif.

#### IV.6 Conclusion

Les résultats montrent que la technique de segmentation est très résistante au bruit de mesure. La majorité des régions non détectées ont une taille inférieure à 100 points, alors que les plus grosses régions non détectées sont en nombre presque égal aux segments trouvés en trop. L'erreur de distance moyenne correspond, à quelques millimètres près à l'erreur en distance moyenne du capteur. La somme des deux erreurs est de l'ordre de 1 cm. L'erreur d'orientation moyenne est de l'ordre de  $1^\circ$  et l'erreur photométrique moyenne est de l'ordre de 5 unités dans l'espace RVB, une erreur à peine perceptible. Le taux de réussite de notre technique de segmentation est de 93,4 %.

	Référence	Nombre de régions	Classe A	Classe B	Classe C	Classe D	Classe E	Erreur angulaire (écart-typc)
USF (PERC)	(Hoover et al., 1996)	14.6	8.9 (61,0%)	0.4 (2,7%)	0.0 (0,0%)	5.3 (36,3%)	3.6 (24,7%)	2,7° (1,8)
WSU (PERC)	(Hoover et al., 1996)	14.6	5.9 (40,4%)	0.5 (3,4%)	0.6 (4,1%)	6.7 (45,9%)	4.8 (32,9%)	3,3° (1,6)
UB (PERC)	(Hoover et al., 1996)	14.6	9.6 (65,8%)	0.6 (4,1%)	0.1 (0,7%)	4.2 (28,8%)	2.8 (19,2%)	3,1° (1,7)
UE (PERC)	(Hoover et al., 1996)	14.6	10.0 (68,5%)	0.2 (1,4%)	0.3 (2,1%)	3.8 (26,0%)	2.1 (14,4%)	2,6° (1,5)
UFPR (PERC)	(Gotardo et al., 2003)	14.6	11.0 (75,3%)	0.3 (2,1%)	0.1 (0,7%)	3.0 (20,5%)	2.5 (17,1%)	2,5° (NC)
USF (ABW)	(Hoover et al., 1996)	15.2	12.7 (83,6%)	0.2 (1,3%)	0.1 (0,7%)	2.1 (13,8%)	1.2 (7,9%)	1,6° (0,8)
WSU (ABW)	(Hoover et al., 1996)	15.2	9.7 (63,8%)	0.5 (3,3%)	0.2 (1,3%)	4.5 (29,6%)	2.2 (14,5%)	1,6° (0,7)
UB (ABW)	(Hoover et al., 1996)	15.2	12.8 (84,2%)	0.5 (3,3%)	0.1 (0,7%)	1.7 (11,2%)	2.1 (13,8%)	1,3° (0,8)
UE (ABW)	(Hoover et al., 1996)	15.2	13.4 (88,2%)	0.4 (2,63%)	0.2 (1,3%)	1.1 (7,2%)	0.8 (5,3%)	1,6° (0,9)
OU (ABW)	(Jiang et al., 2000)	15.2	9.8 (64,5%)	0.2 (1,3%)	0.4 (2,6%)	4.4 (28,9%)	3.2 (21,1%)	NC (NC)
PPU (ABW)	(Jiang et al., 2000)	15.2	6.8 (44,7%)	0.1 (0,7%)	2.1 (13,8%)	3.4 (22,4%)	2.0 (13,2%)	NC (NC)
UA (ABW)	(Jiang et al., 2000)	15.2	4.9 (32,2%)	0.3 (2,0%)	2.2 (14,5%)	3.6 (23,7%)	3.2 (21,1%)	NC (NC)
UFPR (ABW)	(Gotardo et al., 2003)	15.2	13.0 (85,5%)	0.5 (3,3%)	0.1 (0,7%)	1.6 (10,5%)	1.4 (9,2%)	1,5° (NC)
Nous	Ce document	416	367 (88,2%)	5 (1,2%)	2 (0,5%)	42 (10,1%)	44 (10,6%)	0,99° (1,83)
Rang			1	2	2	3	4	1

Tableau IV.5 Comparaison des techniques de segmentation

## ANNEXE V

### SIMULATIONS SPÉCIFIQUES

Les trois séries de simulations présentées à l'annexe IV permettent d'avoir une bonne appréciation des performances de la méthode de segmentation proposée dans un environnement réaliste. Toutefois, la scène utilisée pour les simulations précédentes contenait une faible variété de surface cylindriques, de surfaces sphériques et de couleurs. Cette annexe présente des résultats de simulations effectuées dans le but de caractériser plus précisément les performances de notre système de segmentation pour ces trois éléments. Plus précisément, nous avons recherché les caractéristiques suivantes :

- Détection des surfaces cylindriques et sphériques en fonction de leur rayon ;
- Détection des surfaces cylindriques et sphériques en fonction de leur distance au capteur ;
- Détection des surfaces cylindriques en fonction de leur orientation par rapport au capteur ;
- Détection des éléments surfaciques en fonction de leur différence de couleur ;
- Détection d'un élément surfacique éclairé de manière non uniforme ;
- Seuil photométrique de détection des éléments surfaciques en fonction du niveau de lumière.

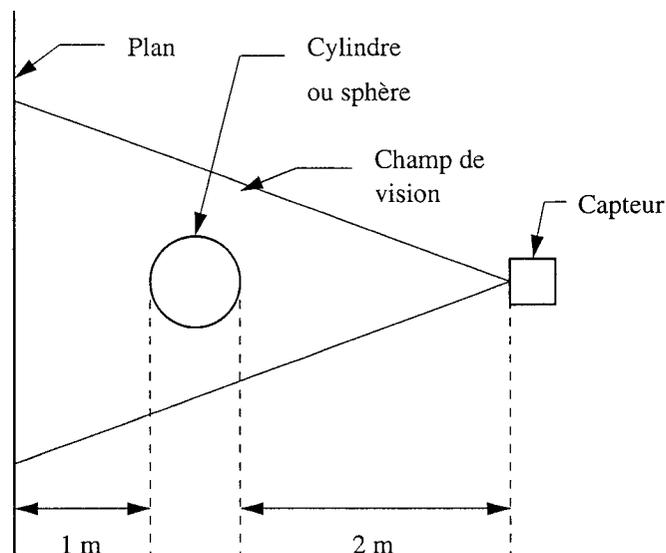


Figure V.1 Montage expérimental simulé pour la détection des surfaces courbes

#### V.1 Détection des surfaces cylindriques et sphériques en fonction de leur rayon

Pour ces simulations, nous plaçons des sphères et des cylindres conformément à l'illustration V.1 de manière à ce que le rayon du laser correspondant au point central de l'image soit normal à la surface des objets. Les centres des sphères et des cylindres sont situés de sorte que la plus petite distance entre le capteur et la surface numérisée soit toujours de 2 m. Une surface plane de très grandes dimensions est située à une distance de 1 m de l'arrière des objets numérisés. Le rayon des cylindres et des sphères est varié par incrément de 5 cm. Les résultats obtenus sont présentés à la figure V.2.

Il faut pointer d'abord le minuscule plateau obtenu pour les faibles rayons. Ce plateau indique que les surfaces cylindriques et sphériques de rayon 5 cm et 10 cm sont détectées comme des plans lorsqu'ils sont situés à 2 m du capteur. Cela est dû au fait que tous les points de ces objets sont situés à une distance inférieure

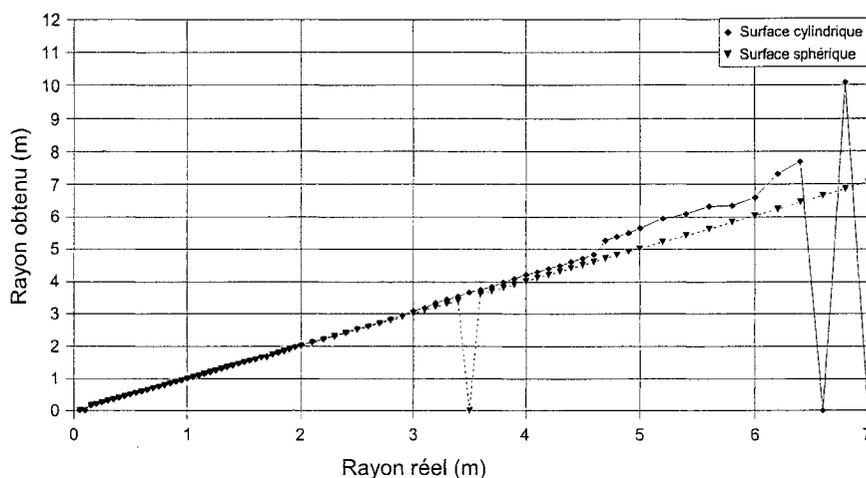


Figure V.2 Rayon obtenu en fonction du rayon réel des surfaces courbes numérisées à  $d_{max}$  d'un seul plan. Rappelons que le paramètre  $d_{max}$  détermine la distance maximale à laquelle un point est considéré comme faisant partie d'un plan. Pour ces simulations, nous avons posé  $d_{max}=3$  cm.

Les surfaces sphériques sont bien détectées dans toute la plage étudiée à partir de 15 cm de rayon. L'erreur obtenue pour un rayon de 3,5 m est selon toute évidence un artefact dû à une situation particulière du bruit aléatoire du capteur et de la composante aléatoire de la détection de plans.

Contrairement aux surfaces sphériques, les surfaces cylindriques sont bien détectées avec une erreur inférieure à 2 % jusqu'à un rayon de 2,9 m. L'erreur grimpe ensuite jusqu'à atteindre 12 % à un rayon de 4,7 m. Cette erreur de 12 % se maintient jusqu'à un rayon de 6 m à partir duquel la détection des surfaces cylindriques devient erratique. Ce comportement s'explique par le fait que la courbure du cylindre est perçue uniquement dans un seul axe de l'image et que l'angle d'ouverture de la caméra dans cet axe est limité à  $42^\circ$ . Lorsque le rayon du cylindre est inférieur à

1,1 m, le capteur voit la surface cylindrique et un morceau de plan de chaque côté. Lorsque le rayon atteint 1,1 m, le capteur ne voit plus que la surface cylindrique. Plus le rayon augmente et plus le capteur voit une proportion réduite de la surface cylindrique et cette portion visible se rapproche d'une surface plane. Ainsi, une petite erreur de distance sur les mesures a un impact de plus en plus important sur le calcul du rayon du cylindre.

Le même raisonnement s'applique aux surfaces sphériques. Toutefois, le rayon de courbure des sphères peut être perçu dans tous les axes. Si l'angle d'ouverture du capteur n'est que de  $42^\circ$  en lacet, il est de  $90^\circ$  en tangage. Ainsi, l'algorithme de segmentation possède plus d'informations pour déterminer le rayon de courbure des surfaces sphériques.

La majorité des surfaces cylindriques que l'on retrouve en milieu structuré appartiennent à des cylindres verticaux. Le résultat obtenu ci-dessus suggère qu'il pourrait être profitable de balayer l'espace de gauche à droite plutôt que de haut en bas afin d'augmenter l'angle d'ouverture horizontal et d'améliorer ainsi la détection des surfaces appartenant à des cylindres verticaux. Notons toutefois qu'il est rare d'observer une surface cylindrique dont le rayon est supérieur à 50 cm.

## V.2 Détection des surfaces cylindriques et sphériques en fonction de la distance

Cette simulation consiste à garder le rayon des surfaces courbes fixe et à modifier la distance séparant le capteur de la surface. Pour ces simulations, nous avons choisi un rayon de 0,3 m et nous avons fait varier la distance de 0,1 m à 8 m. Il faut noter que le capteur que nous utilisons réellement a une portée minimale d'environ 0,4 m et une portée maximale de 8 m. Les résultats sont présentés à la figure V.3.

L'analyse de la figure V.3 montre que les surfaces cylindriques sont mieux détectées

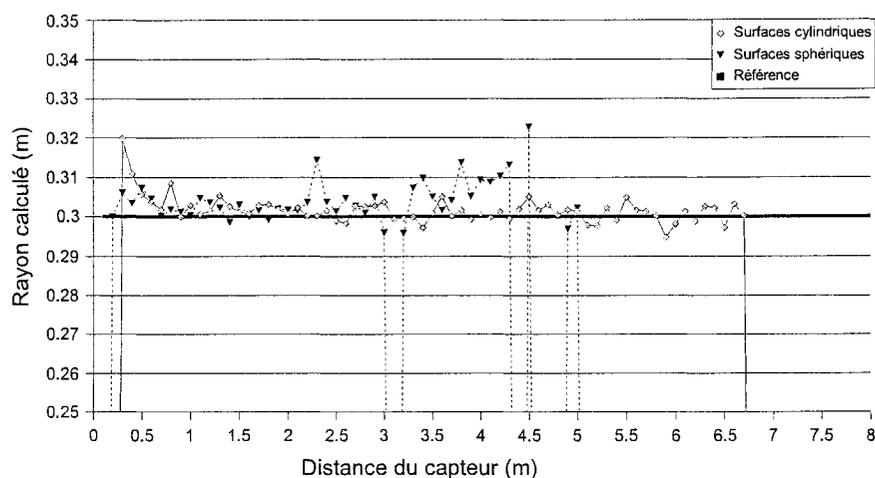


Figure V.3 Détection des surfaces cylindriques et sphériques en fonction de la distance au capteur

à grande distance que les surfaces sphériques. Les surfaces cylindriques ont été détectées jusqu'à une distance de 6,7 m alors que cette distance était de 4,3 m pour les surfaces sphériques. Cette différence s'explique par le fait que les surfaces sphériques comportaient en général moins de points que les surfaces cylindriques à cause de leur forme. Nous en venons à la conclusion (évidente) que les objets de plus petite surface sont moins bien détectés à grande distance. Il est également intéressant de noter que, dans les limites où les objets sont correctement détectés, la distance n'a pas d'effet important sur le rayon obtenu.

### V.3 Détection des surfaces cylindriques en fonction de leur orientation

Cette série de simulations a pour but de s'assurer que les surfaces cylindriques sont toujours bien détectées, peu importe leur orientation spatiale. Pour ce faire, nous faisons pivoter, de façon indépendante, un cylindre initialement vertical autour de

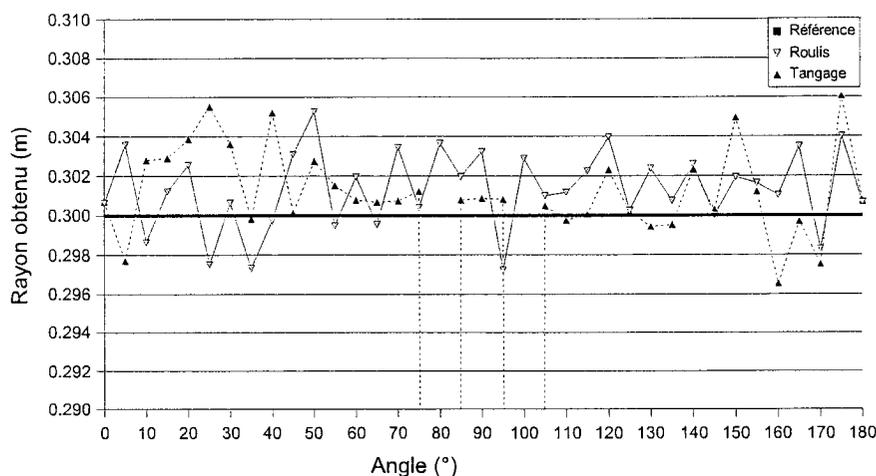


Figure V.4 Détection des surfaces cylindriques en fonction de l'inclinaison du cylindre

deux axes perpendiculaires entre eux. La première rotation est réalisée autour de l'axe correspondant au rayon laser passant par le centre de l'image. Cette rotation correspond au roulis. La seconde rotation est effectuée autour d'un axe perpendiculaire au premier axe et à l'axe de révolution du cylindre. Cette rotation correspond au tangage. La rotation autour de l'axe de révolution du cylindre, qui correspondrait au lacet, est inutile puisque le cylindre est symétrique autour de cet axe. Les résultats sont présentés à la figure V.4. Il en ressort que les surfaces cylindriques sont bien détectées, peu importe leur orientation par rapport au capteur. Nous notons deux points particuliers à  $80^\circ$  et à  $100^\circ$  de tangage. L'aspect symétrique de ces points autour de  $90^\circ$  suggère qu'il ne s'agit pas d'un artefact. En fait, l'aspect le plus étonnant est que les surfaces cylindriques sont détectables entre ces deux angles. En effet, à  $90^\circ$  de tangage, la surface devient alors parallèle à la direction d'observation. Seuls les effets de perspective permettent alors d'observer l'intérieur du cylindre.



Figure V.5 Montage expérimental simulé pour obtenir les résultats du tableau V.1.

#### V.4 Détection des éléments surfaciques en fonction de la différence de leur couleur

Cette série de simulations a pour objectif de déterminer quelle est la différence de couleur minimale dans l'échelle RVB nécessaire afin que notre algorithme de segmentation distingue deux couleurs. Ces simulations sont effectuées avec une lumière ambiante uniforme. Nous avons créé une scène composée de 3 éléments surfaciques tel que présenté à la figure V.5.

Nous avons étudié quatre cas différents : une variation sur chacun des trois canaux séparément et une variation sur les trois canaux simultanément. Pour chacun des quatre cas, nous avons effectué une simulation centrée sur la valeur minimale 0, une centrée sur la valeur intermédiaire 128 et une dernière centrée sur la valeur maximale 255. Pour chacune de ces valeurs, nous avons noté l'écart de couleur minimal  $\Delta c_{seuil}$  nécessaire pour obtenir une détection. Le tableau V.1, page 226, présente les paramètres de chaque simulation et les résultats obtenus se trouvent dans la dernière colonne.

Si nous calculons la moyenne des déviations des trois canaux, nous trouvons que

	Couleur du carré supérieur	Couleur du carré central	Couleur du carré inférieur	$\Delta c_{seuil}$ obtenu
Rouge clair	$\begin{pmatrix} 255 - 2\Delta cl \\ 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 255 - \Delta cl \\ 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 255 \\ 0 \\ 0 \end{pmatrix}$	24
Rouge moyen	$\begin{pmatrix} 128 - \Delta cl \\ 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 128 \\ 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 128 + \Delta cl \\ 0 \\ 0 \end{pmatrix}$	24
Rouge foncé	$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} \Delta cl \\ 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 2\Delta cl \\ 0 \\ 0 \end{pmatrix}$	24
Vert clair	$\begin{pmatrix} 0 \\ 255 - 2\Delta cl \\ 0 \end{pmatrix}$	$\begin{pmatrix} 0 \\ 255 - \Delta cl \\ 0 \end{pmatrix}$	$\begin{pmatrix} 0 \\ 255 \\ 0 \end{pmatrix}$	24
Vert moyen	$\begin{pmatrix} 0 \\ 128 - \Delta cl \\ 0 \end{pmatrix}$	$\begin{pmatrix} 0 \\ 128 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 0 \\ 128 + \Delta cl \\ 0 \end{pmatrix}$	24
Vert foncé	$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 0 \\ \Delta cl \\ 0 \end{pmatrix}$	$\begin{pmatrix} 0 \\ 2\Delta cl \\ 0 \end{pmatrix}$	24
Bleu clair	$\begin{pmatrix} 0 \\ 0 \\ 255 - 2\Delta cl \end{pmatrix}$	$\begin{pmatrix} 0 \\ 0 \\ 255 - \Delta cl \end{pmatrix}$	$\begin{pmatrix} 0 \\ 0 \\ 255 \end{pmatrix}$	24
Bleu moyen	$\begin{pmatrix} 0 \\ 0 \\ 128 - \Delta cl \end{pmatrix}$	$\begin{pmatrix} 0 \\ 0 \\ 128 \end{pmatrix}$	$\begin{pmatrix} 0 \\ 0 \\ 128 + \Delta cl \end{pmatrix}$	24
Bleu foncé	$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 0 \\ 0 \\ \Delta cl \end{pmatrix}$	$\begin{pmatrix} 0 \\ 0 \\ 2\Delta cl \end{pmatrix}$	24
Blanc	$\begin{pmatrix} 255 - 2\Delta cl \\ 255 - 2\Delta cl \\ 255 - 2\Delta cl \end{pmatrix}$	$\begin{pmatrix} 255 - \Delta cl \\ 255 - \Delta cl \\ 255 - \Delta cl \end{pmatrix}$	$\begin{pmatrix} 255 \\ 255 \\ 255 \end{pmatrix}$	8
Gris	$\begin{pmatrix} 128 - \Delta cl \\ 128 - \Delta cl \\ 128 - \Delta cl \end{pmatrix}$	$\begin{pmatrix} 128 \\ 128 \\ 128 \end{pmatrix}$	$\begin{pmatrix} 128 + \Delta cl \\ 128 + \Delta cl \\ 128 + \Delta cl \end{pmatrix}$	8
Noir	$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} \Delta cl \\ \Delta cl \\ \Delta cl \end{pmatrix}$	$\begin{pmatrix} 2\Delta cl \\ 2\Delta cl \\ 2\Delta cl \end{pmatrix}$	8

Tableau V.1 Seuils photométriques nécessaires pour discerner les éléments surfaciques

le seuil moyen est toujours de 8. Cette valeur est très proche du seuil  $S_P$  — seuil appliqué à l'image de contours avant d'effectuer la ligne de partage des eaux — que nous avons fixé à 6. La différence s'explique par l'application du filtre gaussien. Ceci montre que la segmentation photométrique peut être très sélective si la différence photométrique s'applique sur les trois canaux à la fois.

#### V.5 Détection d'un élément surfacique éclairé de manière non uniforme

La présente simulation a pour but de vérifier l'impact de l'éclairage sur la détection d'un élément surfacique. La scène est composée d'une surface grise éclairée par une lumière blanche. L'intensité de la lumière varie de 0 % à 100 % par incrément de 5 % : 0 % signifie qu'il n'y a pas de lumière alors que 100 % signifie que le centre de la tache lumineuse sur la surface est complètement blanc tel que montré à la figure V.6. Selon les résultats du tableau V.1, en choisissant une surface grise et une lumière blanche, nous vérifions l'effet obtenu dans le pire cas puisque les trois canaux varient pareillement. Les résultats sont affichés au graphique V.7. Ces résultats montrent que l'intensité lumineuse appliquée n'a aucun effet sur la bonne détection de la surface. Cela démontre la robustesse de notre système par rapport à l'éclairage.

#### V.6 Seuil photométrique de détection des éléments surfaciques en présence de lumière

Les simulations de la section précédente ont montré que la détection d'un élément surfacique est très robuste à l'éclairage. Cette dernière série de simulations a pour but de vérifier l'effet de la lumière sur la distinction entre deux éléments surfaciques. Les simulations effectuées ont pris la forme de celles de la section V.4, page 225,

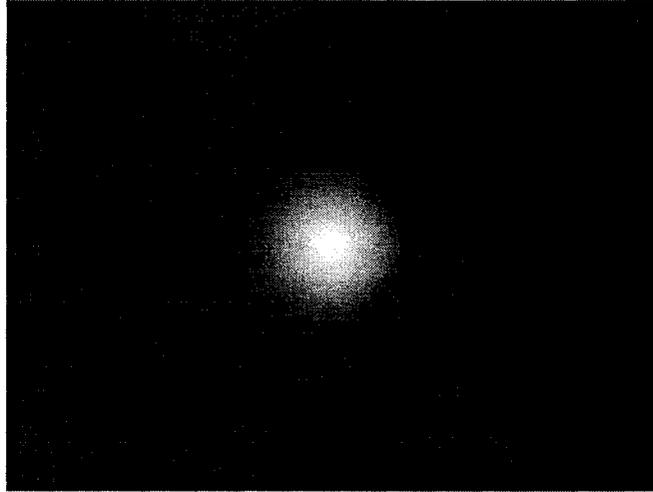


Figure V.6 Surface grise éclairée par une lumière blanche avec un niveau de lumière de 100 %

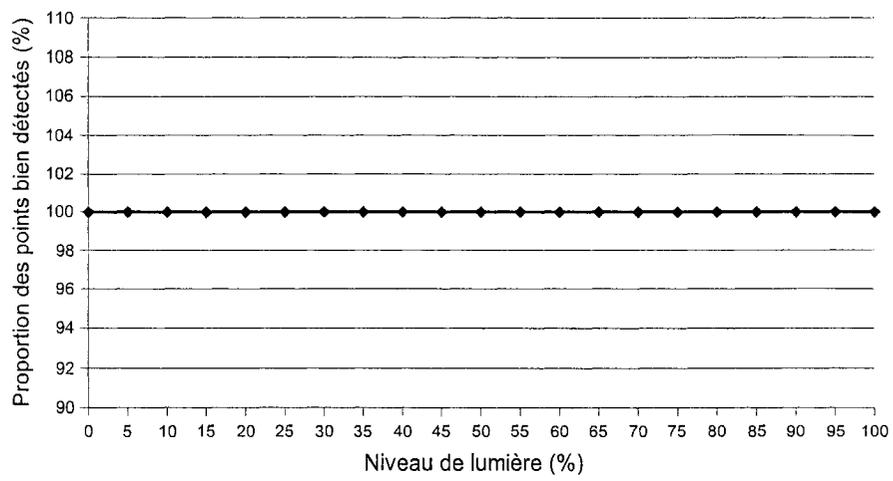


Figure V.7 Influence de l'éclairage non uniforme sur la détection d'un élément surfacique

auxquelles nous avons ajouté une lumière blanche qui éclaire l'élément surfacique central. Nous avons fait varier l'intensité de cette lumière de 0 % à 100 % par incrément de 10 %.

Les résultats obtenus ont montré que trois cas peuvent survenir :

1. Les trois éléments surfaciques ont été fusionnés. Dans ce cas, nous supposons que seuls les points du plus grand élément surfacique ont été assignés à la bonne surface. Nous obtenons alors un taux de réussite d'environ 40 %, car 40% des points appartiennent au plus grand éléments surfacique.
2. Deux des trois éléments surfaciques sont fusionnés. Pour les deux surfaces fusionnées, nous supposons que seuls les points du plus grand élément surfacique ont été assignés à la bonne surface. Nous obtenons alors un taux de réussite d'environ 70 %. En effet, l'élément surfacique bien détecté représente 30 % des points et le plus grand élément surfacique contient 40 % des points.
3. Les trois éléments surfaciques ont été bien segmentés. Le taux de réussite est alors d'environ 100 % puisque tous les points sont assignés aux bons éléments surfaciques.

Les résultats sont présentés à la figure V.8. Chaque courbe représente une intensité lumineuse différente de la lumière qui éclaire l'élément surfacique central. Plusieurs observations peuvent être tirées de ce graphique.

Premièrement, nous notons que la lumière a pour effet de déplacer le seuil de détection vers la gauche. Pour une intensité lumineuse donnée, il y a une différence de couleur minimale nécessaire entre deux éléments surfaciques pour que ceux-ci soient détectés. Plus il y a de lumière et plus cette différence de couleur est faible. Autrement dit, pour une différence de couleur donnée, il y a un niveau de lumière minimal nécessaire à la détection des éléments surfaciques.

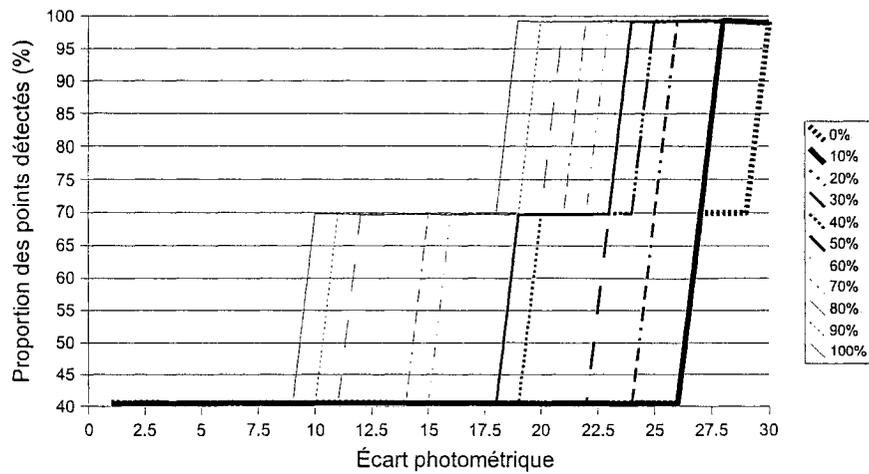


Figure V.8 Effet de la lumière sur la distinction entre les surfaces

Chaque courbe représente une intensité lumineuse différente.

Deuxièmement, nous notons la présence d'un plateau à 70 % de points détectés. Ce plateau survient lorsqu'un élément surfacique a été bien détecté et que deux éléments surfaciques ont été fusionnés ensemble. La largeur de ce plateau s'agrandit lorsque l'intensité de la lumière appliquée s'accroît. Cela s'explique par le fait que la lumière était dirigé vers l'élément surfacique central. Cela a eu pour effet de rendre cet élément surfacique plus pâle, diminuant ainsi le contraste de couleur avec l'élément surfacique qui était initialement plus pâle que lui et augmentant le contraste avec l'élément surfacique qui était plus foncé que lui.

Troisièmement, nous notons que lorsque les trois surfaces sont détectées, nous avons un taux de succès de 100 %. Cela signifie que la position des contours des surfaces n'est pas affectée par l'éclairage.

## V.7 Conclusion

Les résultats ont démontré l'existence d'une distance maximale de détection des surfaces cylindriques et sphériques ainsi que l'existence d'un rayon minimal. Il s'agit là de limitations dues à la fois à la faible résolution du capteur utilisé et au paramètre  $d_{max}$  de la méthode de segmentation. Les résultats obtenus ont également montré que la segmentation photométrique est peu affectée par la luminosité ambiante. L'éclairage affecte le seuil de détection des couleurs, mais n'affecte pas la position des frontières entre les éléments surfaciques.