

UNIVERSITÉ D'ÉVRY VAL-D'ESSONNE



École Doctorale Sciences & Ingénierie

Laboratoire d'Informatique, Biologie Intégrative et Systèmes Complexes (IBISC)
Équipe Interaction, Réalité Augmentée et Robotique Ambiante (IRA^2)

THÈSE

présentée par

Jean-Clément DEVAUX

et soutenue publiquement le 24 octobre 2014

pour obtenir le grade de

DOCTEUR ÈS SCIENCES

Spécialité « Vision et Robotique »

Perception multicapteur : Etalonnage extrinsèque de caméra 3D, télémètre laser et caméra conventionnelle

Application au déplacement autonome et téléopéré d'un robot mobile

---

Jury composé de

RAPPORTEURS

Dr. Patrick RIVES	Equipe Lagadic, INRIA Sophia Antipolis
Pr. Youcef MEZOUAR	Institut Pascal, IFMA

EXAMINATEURS

Dr. Guillaume CARON	Laboratoire MIS, Université de Picardie Jules Verne
Dr. David ROUSSEL	Laboratoire IBISC, ENSIIE

THÈSE DIRIGÉE PAR

Pr. Etienne COLLE	Laboratoire IBISC, Directeur de thèse
Dr. Hicham HADJ-ABDELKADER	Laboratoire IBISC, Co-encadrant

Remerciements

Avertissement : Difficile d'écrire en si peu de place cette partie pourtant souvent la plus agréable à lire pour le lecteur. Ne m'en veuillez pas en cas d'oubli.

Ce travail a été réalisé au laboratoire IBISC (Informatique, Biologie Intégrative et Systèmes Complexes) au sein de l'équipe IRA^2 (Interaction, Réalité Augmentée et Robotique Ambiante). Il y a beaucoup de personnes à remercier et à associer à cette thèse mais, faute de place, je vais n'en citer que quelques-uns ou quelques-unes.

Je tiens donc tout d'abord à remercier l'ensemble des membres du jury de cette thèse. Je commencerais par Patrick RIVES, directeur de recherche à l'INRIA Sophia-Antipolis et par Youcef MEZOUAR, Professeur à l'Université Blaise Pascal de Clermont-Ferrand pour leurs rapports minutieux, à la fois critiques et bienveillants, qui m'auront permis d'améliorer significativement la qualité de ces travaux et de ce manuscrit. Je veux également adresser de chaleureux remerciements à Guillaume CARON et à David ROUSSEL, Maîtres de Conférence respectivement à l'Université de Picardie Jules Verne à Amiens et à l'ENSIIE à Evry pour avoir accepté d'examiner cette thèse et avoir permis, au détour de certaines discussions, de faire éclore véritablement certaines idées. Enfin, je remercie particulièrement Youcef et David pour leurs conseils dans cette aventure initiatique de chercheur en tant que membres de mon comité de thèse.

Il est évident que ces travaux n'existeraient pas sans la disponibilité et la bienveillance d'Etienne COLLE et d'Hicham HADJ-ABDELKADER, qui ont respectivement dirigé et encadré cette thèse. Evidemment, leur concours scientifique et leurs conseils judicieux ont

permis d'élever le niveau de ces travaux mais ils ont aussi su me guider jusqu'au bout dans la bonne humeur et l'optimisme en me soutenant pour dépasser les difficultés. Ils ont été remarquables à tous les niveaux lors de cette thèse : motivation, science, méthodologie, bonne ambiance. Jusque dans la rédaction de ce manuscrit pour lequel leurs relectures multiples, auxquelles j'associe celles (nombreuses) de ma correctrice anonyme préférée, auront permis, je l'espère, d'aboutir à un document lisible et agréable. Pour tout cela, je souhaite leur adresser mes plus sincères et chaleureux remerciements personnels et professionnels.

Je souhaite ensuite remercier tous les membres du laboratoire IBISC de m'avoir accompagné ces 5 dernière années (3 ans de thèse et 2 stages) et surtout Sabine, Fred et Florent, toujours présents au bon moment et au bon endroit ! Merci à l'équipe HANDS (pendant mes stages) devenue IRA^2 (pendant cette thèse). Merci à tous les anciens que j'ai pu croiser : Sébastien, Antonio, Paul, Ha, Philippe et Cuong (par ordre de départ). Je n'oublie pas non plus Bruno MANTEL dont certains travaux ont amorcé une partie de cette thèse. Merci à tous les nouveaux, ceux de passage et les futurs. Merci aussi à tous les permanents et aux moins permanents de l'équipe.

Je veux maintenant remercier les camarades, tous ceux que j'ai embêtés presque tous les jours et qui en redemandaient... Merci au clan *café*, merci aussi, quand même, à celui de la *cloppe*. Je pense entre autres à Jean-Yves et à son look de viking, à Christophe, drogué au sport, et à ses *montagneries*, à Maxime, l'Ancien, à notre émulation réciproque et à nos différends politiques (même quand on est d'accord). Merci enfin à Maxime, le Jeune, fidèle compagnon de bureau, le dernier des Mohicans, gardien des animaux, supporter inconditionnel et encyclopédie à lui tout seul. Il est probable que j'en oublie. Honte à moi.

Je voudrais conclure cette page en m'adressant à toute ma famille. Alors voilà, simplement : Merci pour tout. Tous méritent des remerciements, chacun pour des raisons particulières, mais je les adresse en particulier à trois personnes :

A ma mère, A mon père, A mon épouse.

Résumé

Ce travail de thèse s'inscrit dans le domaine de la perception multicapteur pour la robotique mobile en milieu intérieur et, plus précisément, traite de l'exploitation des caméras 3D à bas coût en complément du système de perception d'un robot équipé d'un télémètre laser à balayage et de caméras conventionnelles. En effet, chacun de ces capteurs seuls souffre de défauts importants : limitation du champ de vision pour les caméras à bas coût et détection sur un unique plan pour le télémètre. L'utilisation de la caméra 3D en complément du télémètre doit permettre de percevoir les obstacles invisibles par le télémètre tout en conservant son large champ de vision. La caméra 3D apporte également des données très utiles pour augmenter les informations visuelles dans le cadre d'une interface homme-robot.

La perception multicapteur basée sur le télémètre laser, la caméra 3D et la caméra conventionnelle nécessite un étalonnage extrinsèque suffisamment précis pour l'application considérée. Au-delà de la précision, le processus lui-même se doit d'être pratique et facilement répétable puisque, notamment lorsque les capteurs sont mobiles mais pas seulement, l'étalonnage doit être renouvelé régulièrement. L'interrogation de la communauté utilisant les caméras 3D à bas coût montre que la moitié des chercheurs se contente d'un étalonnage approximatif du fait du caractère fastidieux des processus d'étalonnage. Nous proposons donc dans cette thèse quatre méthodes d'étalonnage extrinsèque dont deux en particulier pour lesquelles l'objectif est d'améliorer significativement la praticité du processus d'étalonnage.

La première méthode, reposant sur le télémètre et la caméra 3D, consiste en une ex-

tension de l'état de l'art de l'étalonnage laser et caméra conventionnelle à la caméra 3D. Nous proposons en outre un étalon tenant compte des spécificités des capteurs qui permet d'améliorer très significativement la précision de l'étalonnage. Les trois autres méthodes d'étalonnage concernent le couple caméra 3D et caméra conventionnelle. Tout d'abord, nous proposons une dérivation de la méthode précédente qui permet dans un seul et même processus d'étalonner la caméra 3D avec le télémètre laser et la caméra conventionnelle. Nous développons ensuite une autre méthode inédite totalement automatique qui repose sur une unique prise de vue et un asservissement visuel simulé entre les poses des deux capteurs. Enfin, nous démontrons qu'il est possible d'atteindre la précision de l'état de l'art avec une méthode automatique et basée sur une scène naturelle.

Nous présentons dans un premier temps une méthode de composition des données issues de la caméra 3D et du télémètre laser pour créer un évitement d'obstacles robuste. Cette méthode repose sur l'étalonnage extrinsèque entre les deux capteurs, la création d'un télémètre virtuel, la persistance des données basée sur un SLAM relatif et local et, enfin, un évitement d'obstacles réactif conventionnel. Enfin, nous verrons comment la caméra 3D et la caméra conventionnelle peuvent collaborer pour offrir une meilleure expérience utilisateur dans le cadre d'une téléopération en réalité augmentée. Notre approche se base sur la proposition d'un ensemble de méthodes d'étalonnage extrinsèque entre la caméra 3D et la caméra conventionnelle et d'une interface de téléopération en réalité augmentée.

Mots clés : Caméra 3D, télémètre laser, caméra conventionnelle, étalonnage extrinsèque, télémètre virtuel, évitement d'obstacles, SLAM, interface, réalité augmentée.

Table des matières

Introduction **17**

1 Etalonnage extrinsèque laser et Caméra 3D **21**
1.1 Introduction . . . 21
1.2 Caractérisation des capteurs . . . 22
1.2.1 Caméra 3D . . . 23
1.2.2 Télémètre à balayage laser . . . 28
1.2.3 Les données d'étalonnage . . . 32
1.3 Etat de l'art . . . 33
1.4 Modèle d'étalonnage extrinsèque . . . 38
1.4.1 Modèle linéaire . . . 40
1.4.2 Optimisation non linéaire de la translation $\mathbf{t}$ . . . 44
1.4.3 Optimisation non linéaire de la rotation $\mathbf{R}$ et de la translation $\mathbf{t}$ . . . 45
1.4.4 Métrique le long du rayon laser . . . 46
1.5 Evaluation . . . 47
1.5.1 Objectifs et métriques . . . 47
1.5.2 Protocole d'évaluation . . . 47
1.5.3 Résultats . . . 49
1.6 Conclusion . . . 54

2 Application à l'évitement d'obstacles basé sur un télémètre virtuel **57**
2.1 Introduction . . . 57
2.2 Etat de l'art . . . 58
2.3 Modèle d'évitement basé sur un télémètre virtuel laser-Kinect . . . 62
2.3.1 Modèle du télémètre virtuel . . . 62
2.3.2 Modèle de persistance des informations . . . 64
2.3.3 Modèle d'évitement d'obstacles . . . 67
2.4 Evaluation . . . 69
2.4.1 Objectifs et métriques . . . 69

2.4.2 Protocole expérimental 70
2.4.3 Résultats 71
2.5 Conclusion 80

3 Etalonnage extrinsèque Caméra conventionnelle-Caméra 3D 83
3.1 Introduction 83
3.2 Etat de l'art 85
3.3 Modèles 90
3.3.1 Méthode d'appartenance à un plan (M1) 90
3.3.2 Méthode complètement automatique (M2) 93
3.3.3 Méthode automatique et naturelle (M3) 99
3.3.4 Alignement de nuages de points 3D contenant des plans 106
3.4 Evaluation 110
3.4.1 Objectifs et métriques 110
3.4.2 Protocole expérimental 112
3.4.3 Résultats 113
3.5 Conclusion 124

4 Application à la robotique téléopérée ou autonome 127
4.1 Introduction 127
4.2 Application à la téléopération d'un robot en RA 129
4.2.1 Modèle de l'existant 129
4.2.2 Extension à la caméra 3D 135
4.2.3 Evaluation 138
4.3 Application au SLAM RGB-D 144
4.3.1 Modèle du SLAM RGB-D 145
4.3.2 Evaluation 147
4.4 Conclusion 150

Conclusion 153

Annexes 165

A Communications scientifiques 165

B Exemples de scènes pour le télémètre virtuel 167

Liste des tableaux

1.1 Comparaison des caractéristiques du télémètre laser et de la caméra 3D . . 23
1.2 Erreur de reprojection en cm pour l'étalonnage laser-caméra 3D utilisant un damier traditionnel ou le plan étalon modifié 51
1.3 Résultats des tests de Kruskal-Wallis entre les différentes méthodes d'étalonnage 53
1.4 Résultats des tests de Kruskal-Wallis entre l'étalonnage avec damier et l'étalonnage avec plan modifié 53

2.1 Règles concernant la vitesse angulaire 68
2.2 Mesures du télémètre laser et de la caméra 3D. Les vrais positifs et vrais négatifs sont déterminés à partir du télémètre virtuel et donc significatifs uniquement dans le secteur frontal. 74

3.1 Métrique d'optimisation dans les différentes méthodes comparées. 111
3.2 Comparaison globale des méthodes d'étalonnage. L'erreur moyenne et l'écart-type sont présentés en pixels en utilisant la métrique de la méthode stéréo. . 123

Table des figures

1.1 Vues de l'intérieur des capteurs Kinect et Kinect 2 (source : IHS). 24

1.2 Schéma de fonctionnement de la caméra 3D (source : PrimeSense.com). . . . 25

1.3 Images du motif projeté sur un plan et avec un élément à une profondeur différente (source : FuturePicture.org). 26

1.4 Erreur moyenne et écart-type de la Kinect selon les données de [Khoshelham et Elberink, 2012] et [Teichman *et al.*, 2013] 27

1.5 Résolution de la Kinect [Khoshelham et Elberink, 2012] 27

1.6 Résidus d'une régression planaire montrant la distorsion de disparité selon les données de [Smisek *et al.*, 2013] à deux profondeurs différentes 28

1.7 Schéma de fonctionnement d'un télémètre laser à balayage (domaine public/Mike1024) 29

1.8 Exemples de plan étalon : le plan damier traditionnel (à gauche) et le plan proposé pour l'usage du télémètre laser (à droite) 33

1.9 Représentation de la distance selon la métrique *orthogonal* (segment en bleu) ou *beam* (segment en rouge) d'un point mesuré (disque bleu) à un obstacle réel (en noir) 48

1.10 Position des capteurs sur le robot Lina de l'entreprise Droïds Company . . . 48

1.11 Erreur de reprojection en fonction des méthodes selon la distance orthogonale (gauche) ou le long du rayon (droite) pour le plan damier (A) et le plan modifié (B) 50

2.1 Fonctions d'appartenances du système flou 68

2.2 Obstacle vertical vu par la caméra 3D (gauche) et le télémètre virtuel (droite) 71

2.3 Table vue par la caméra 3D (gauche) et le télémètre virtuel (droite) 72

2.4 Annotation de la scène statique avec une personne se déplaçant 75

2.5 Comportement du télémètre virtuel en présence d'un obstacle mobile complexe 77

2.6 Scénario d'évitement utilisant seulement le télémètre laser. La première ligne correspond à une vue extérieure de la scène pour mieux appréhender la trajectoire du robot. 78

2.7 Scénario d'évitement utilisant le télémètre virtuel. La première ligne correspond à une vue extérieure de la scène pour mieux appréhender la trajectoire du robot. 79

3.1 Image de Profondeur (a), IR (b) et couleur (c) d'une même scène observée par différents capteurs *Kinect* (d) (e) (f). 84
3.2 Classification des méthodes d'étalonnage extrinsèque entre une caméra 3D et une caméra conventionnelle 89
3.3 Vues synthétiques du damier 3D utilisé pour l'étalonnage. Les points rouges matérialisent l'emplacement des points caractéristiques communs aux deux caméras. 95
3.4 Fonctions successives reprojetant les points d'intérêt de l'image couleur 1 (en bas) dans l'image couleur 2 (en haut) en passant par les images de profondeur 100
3.5 Mise en correspondance de deux vues d'une scène à l'aide des points caractéristiques SURF. Aucun faux appariement après réjection par RANSAC. . 104
3.6 Ajout d'un canal D à une image RGB pour former une image RGB-D à partir d'une image RGB et d'une image de profondeur 105
3.7 Images de profondeur (à droite) et couleur (à gauche) d'une même scène à partir de deux poses (une pose par ligne). 109
3.8 Rotation et translation du 1er nuage de points (bleu) vers l'estimation de la deuxième pose (vert) en utilisant les résultats de l'alignement 3D basé plans. Le nuage de points réels observés est montré en rouge. 110
3.9 Erreurs de la méthode M1 en fonction du nombre de plans utilisés 113
3.10 Reprojection des coins du damier dans les données issues de la caméra 3D . 114
3.11 Exemple d'asservissement visuel virtuel entre une image de profondeur et une image conventionnelle 115
3.12 Vue rapprochée de la mire d'étalonnage (caméra conventionnelle) 115
3.13 Convergence des erreurs (en px) et des vitesses (en m/s et rad/s) vers 0 . . 116
3.14 Fusion des nuages de points 3D après étalonnage 117
3.15 Reprojection des points d'intérêt après étalonnage 118
3.16 Erreur moyenne de reprojection en pixels (échelle logarithmique) en fonction du nombre de paires utilisées dans la minimisation 119
3.17 Effet de l'erreur d'estimation de $\mathbf{T}'$ (en cm) sur l'erreur de reprojection (en px) en translation pure. 120
3.18 Effet de la norme de la translation (en cm) sur l'erreur de reprojection (en px). La courbe orange représente la tendance affine. 121
3.19 Histogramme des erreurs de 142 reprojections (en bleu) et pourcentage cumulé (en jaune). 122

4.1 Spécification de la passabilité dans l'approche écologique (B. Mantel). . . . 130
4.2 Spécification du TTC dans l'approche écologique (B. Mantel). 131

4.3 Démarche de conception écologique d'une plate-forme de téléopération. . . . 133
4.4 Conception itérative de l'interface de téléopération écologique. 135
4.5 Interface de téléopération en réalité augmentée. 136
4.6 Utilisation de l'interface de téléopération avec le laser seul. 136
4.7 Positions relatives des capteurs et de la caméra virtuelle. 137
4.8 Augmentation basée caméra 3D sur l'interface de téléopération. 139
4.9 Obstacle complexe de type trépied. 141
4.10 Obstacle de type table 142
4.11 Obstacle de type pied de ventilateur 143
4.12 Pipeline de traitement du RGB-D SLAM. ([Endres *et al.*, 2014]) 146
4.13 Effet d'une erreur d'étalonnage en translation sur l'estimation de la trajectoire148
4.14 Effet d'une erreur d'étalonnage en rotation sur l'estimation de la trajectoire 149

B.1 Scène 1 : Obstacle vertical 167
B.2 Scène 2 : Obstacle de type table 167
B.3 Scène 3 : Obstacle incliné 168
B.4 Scène 4 : Obstacle droit 168
B.5 Scène 5 : Obstacle non posé sur le sol 168

Introduction

Les robots mobiles peuvent répondre à une grande variété de situations. Tantôt téléopérés ou encore tantôt autonomes, ces outils peuvent aussi bien être présents sur un champ de bataille pour réduire les pertes humaines qu'au domicile d'une personne présentant des pertes d'autonomie pour l'assister dans la vie quotidienne. C'est ce dernier cas qui nous intéresse et qui pose le contexte de ces travaux. Dans le domaine de la robotique de service et a fortiori pour l'assistance à la personne, les robots se déplacent et agissent à proximité de la personne. Lorsqu'il s'agit d'aider ou de s'assurer de la sécurité, le robot évolue principalement au domicile de la personne. Cet environnement, naturel pour l'être humain, reste particulièrement difficile pour le robot du fait de la complexité et de la mobilité des nombreux objets et des personnes présents habituellement dans une habitation. Cette problématique se retrouve dans toutes les fonctions du robot, y compris lors de l'activité de navigation simple.

Ces travaux s'incrivent dans la suite des projets ANR QuaVadis TEcSan 2007-2011 et 7FP CompanionAble 2008-2013. Certains résultats expérimentaux ont mis en évidence les difficultés de téléopération d'un robot sans une concentration importante de la part de l'opérateur. Dans le cadre de l'assistance à la personne, et à plus forte raison en situation d'urgence, il est nécessaire d'apporter des assistances à la téléopération ou même de donner des capacités d'autonomie au robot. Tout en respectant les différentes contraintes médicales, éthiques et socio-économiques, le robot est prévu comme un outil permettant d'alléger la charge de travail des services de télévigilance. Pour ce faire, l'équipe IRA^2

travaille sur une approche mettant en jeu un continuum entre la téléopération pure (non souhaitable) et l'autonomie pure (irréaliste à l'heure actuelle et pas non plus forcément souhaitable) du robot. Dans ce continuum, nous retrouvons différents modes de commande qui vont de la commande directe avec alerte de danger en passant par la commande directe avec évitement d'obstacle, la commande par objectifs successifs proches pour finir sur la planification à plus ou moins long terme et l'autonomie totale du robot. Quel que soit le mode de commande, l'assistance nécessite que le robot ait conscience de l'environnement qui l'entoure. Si on ne considère que la tâche de navigation, le robot devra alors connaître la position des obstacles pour activer l'assistance en conséquence. Ce dernier propos paraît énoncer une évidence, pourtant la plupart des capteurs traditionnels de la robotique échoue, lorsqu'ils sont utilisés seuls, à détecter un grand nombre d'obstacles. L'objectif de ces travaux est de fiabiliser cette détection afin de robustifier les éventuelles assistances que nous pourrons développer.

Dans cette thèse, nous pensons que les capacités de perception du robot (ou de l'environnement communicant) sont la clef de résolution des problématiques de navigation autonome ou téléopérée du robot mobile. En effet, comment naviguer sereinement si tous les obstacles ne sont pas détectés. L'objectif de cette thèse est de lever les verrous scientifiques mis en exergue par l'intégration d'une caméra 3D à bas coût au sein d'un robot déjà équipé de caméras conventionnelles et d'un télémètre laser à balayage. Ces verrous résident principalement dans l'estimation de la transformation entre les repères associés aux différents capteurs et posent donc la problématique de l'étalonnage extrinsèque entre les différents capteurs. Le contexte applicatif visé - la robotique d'assistance à domicile - implique des limites de coût très importantes sur les robots mobiles, ce qui induit des contraintes à la fois sur la qualité des capteurs envisagés mais aussi sur les capacités embarquées de calcul et de stockage ou de transmission. Si le capteur 3D semble être une solution aux difficultés de perception, il n'est pas possible d'utiliser des télémètres laser 3D qui présentent un coût très important. Seules les caméras 3D à bas coût semblent envisageables pour permettre une viabilité économique des robots de service. Dans cette thèse,

nous employons le terme de caméra 3D pour désigner tout capteur ou système de capteur permettant de mesurer une profondeur et exprimant le résultat sous la forme d'une image de profondeur respectant le modèle du sténopé. Une caméra 3D peut donc aussi bien être active (c'est le cas des capteurs de type Kinect) que passive (paire stéréoscopique de caméras conventionnelles). La totalité des propositions de cette thèse a été développée avec une caméra 3D active mais est immédiatement transposable à tout type de caméra 3D, voire de capteur télémétrique 3D.

Nous choisissons de développer ces travaux en deux axes qui sont associés aux capteurs très fréquemment présents sur les plateformes courantes de robotique mobile : le télémètre laser 2D à balayage et la caméra conventionnelle. Pour chacun de ces axes, nous proposerons une ou plusieurs méthodes d'étalonnage extrinsèque entre le capteur correspondant et la caméra 3D. Ensuite, pour chaque méthode, nous adopterons la démarche de réaliser une évaluation statistique par comparaison avec d'autres méthodes de la littérature et, pour chaque capteur utilisé conjointement avec la caméra 3D, nous effectuerons une évaluation pratique de l'apport de la caméra 3D (et donc de son étalonnage) dans un contexte applicatif donné.

Ainsi, concernant le premier axe, nous proposons dans le chapitre 1 une méthode d'étalonnage extrinsèque entre un télémètre laser à balayage et une caméra 3D. Nous expliquerons les défauts des télémètres laser à bas coût et nous montrerons comment un choix judicieux d'un plan étalon permet de réduire ces défauts et d'améliorer significativement la précision de l'étalonnage.

Le laser comme la caméra 3D, tous deux à bas coût, présentent des défauts et des avantages complémentaires. Dans le chapitre 2, nous montrerons que la construction d'un télémètre virtuel issu de la combinaison des données du laser et de la caméra 3D permet de gommer les inconvénients de chacun des capteurs pris séparément. En effet, si le télémètre laser propose un large champ de vision, il ne permet pas de percevoir les obstacles qui ne sont pas situés sur son unique plan de détection. Au contraire, la caméra 3D possède un champ de vision plus faible horizontalement mais offre une détection tridimensionnelle qui

permet de détecter quasiment tous les obstacles. En construisant un tel télémètre virtuel, nous montrons qu'il est possible d'obtenir un capteur virtuel qui conserve les défauts du télémètre réel uniquement sur la partie du champ de vision non commune aux deux capteurs. Dans la suite du chapitre, nous proposerons une application d'évitement d'obstacles qui met en évidence les capacités de perception supérieures du télémètre virtuel par rapport au télémètre laser seul ou à la caméra 3D seule.

S'agissant du second axe de cette thèse, autour des caméras conventionnelles, nous présentons dans le chapitre 3 trois méthodes d'étalonnage extrinsèque. La première méthode développée en section 3.3.1 est une extension des travaux présentés au chapitre 1 et peut être utilisée simultanément avec un étalonnage entre le laser et la caméra 3D. Proposées en section 3.3.2 et 3.3.3, les deux dernières méthodes permettent de répondre à la problématique de la complexité des processus traditionnels d'étalonnage en offrant un meilleur automatisme et en raccourcissant le temps de mise en oeuvre.

Enfin, dans le chapitre 4, nous évaluerons en section 4.2 l'apport de la caméra 3D par rapport à l'utilisation de la caméra conventionnelle seule au sein d'une interface de téléopération de robot en réalité augmentée et nous quantifierons en section 4.3 l'influence de la précision d'un étalonnage extrinsèque entre la caméra 3D et la caméra conventionnelle dans une application de SLAM RGB-D.

Notations : *Dans cette thèse, les valeurs scalaires, les vecteurs et les matrices sont respectivement écrits en minuscules (x), en minuscules et en gras ($\mathbf{x}$) et en majuscules et en gras ($\mathbf{X}$). Les exposants en préfixes représentent l'espace dans lequel est exprimée la variable préfixée et les indices en suffixe représentent l'indice d'un élément spécifique de la variable. Par exemple, ${}^{\mathcal{S}}\mathbf{v}_i$ exprime le i^{eme} élément du vecteur $\mathbf{v}$ exprimé dans le repère ou l'espace $\mathcal{S}$.*

Chapitre 1

Etalonnage extrinsèque laser et Caméra 3D

1.1 Introduction

L'intégration d'une caméra 3D dans un système robotique disposant d'un télémètre laser à balayage nécessite d'étudier les avantages et inconvénients relatifs des différents capteurs. Il est globalement admis que pour une application robotique en milieu intérieur, les caméras 3D à bas coût souffrent de défauts importants tels qu'un champ de vision restreint et la plage de détection en profondeur limitée à un intervalle par l'existence d'une zone aveugle et d'une portée limitée. Ainsi, la zone de détection d'une caméra 3D à bas coût (de type Microsoft Kinect ou Asus XTion Pro) ne couvre qu'une portion de cône d'une cinquantaine de degrés à des distances situées entre 40 ou 80 cm et 3 à 5 m selon les conditions d'éclairage de l'environnement. En comparaison, la majorité des télémètres laser à bas coût du marché (en 2013) est supposée capable de détecter au centimètre près un obstacle situé entre 0 et 4 m avec un champ de vision d'au moins 240 degrés. Pour autant, les télémètres laser à bas coût sont tout de même 7 à 8 fois plus coûteux que les caméras 3D à bas coût et, comme ils ne produisent des mesures que sur un unique plan, ils ne permettent pas de détecter des obstacles de forme complexe. L'utilisation conjointe

de ces deux types de capteurs semble intéressante pour pallier aux déficiences à la fois des télémètres laser à bas coût et des caméras 3D à bas coût. Le large champ de vision du télémètre laser apportera des informations que la caméra 3D ne peut produire alors que cette dernière sera capable de détecter les obstacles sur d'autres plans que celui scruté par le télémètre laser. Pour autant, le préalable nécessaire à l'utilisation conjointe du télémètre laser et de la caméra 3D est la connaissance précise de la transformation rigide entre les deux capteurs. L'obtention de cette transformation géométrique est l'objet de l'étalonnage extrinsèque.

L'objectif de ce chapitre est de proposer une technique d'étalonnage extrinsèque d'un couple de capteurs formé par une caméra 3D à bas coût et un télémètre laser 2D à balayage lui aussi à bas coût. La démarche employée repose sur quatre axes : la caractérisation et le traitement des données des capteurs, la formulation d'une solution linéaire et d'une optimisation non linéaire, l'adaptation du processus d'étalonnage aux données imparfaites et l'évaluation de cet étalonnage.

Nous présenterons donc dans la section 1.2 les caractéristiques, le fonctionnement, les qualités et les limites de chacun de ces capteurs, puis nous nous intéresserons dans la section 1.3 à l'état de l'art de l'étalonnage extrinsèque d'un télémètre et d'une caméra 3D. Forts de ces constatations, nous présenterons dans la section 1.4 notre modèle d'étalonnage extrinsèque que nous évaluerons finalement en section 1.5.

1.2 Caractérisation des capteurs

Les capteurs choisis dans ce chapitre sont ceux présents sur notre plateforme expérimentale : le télémètre laser à balayage 2D Hokuyo URG-04LX et la caméra 3D de PrimeSense équipant la Kinect de Microsoft. Ce sont tous deux des capteurs à bas coût dans leur catégorie. Avant de parler plus longuement des méthodes et des protocoles d'étalonnage, intéressons-nous aux capteurs et à leurs données. Nous verrons tout d'abord les caractéristiques (Cf. tableau 1.1) et le principe de fonctionnement des capteurs utilisés puis nous

nous focaliserons sur la représentation et la qualité des données du télémètre laser ou de la caméra 3D. Les unes comme les autres souffrent d'imprécisions dues à de nombreux facteurs.

	Télémètre Laser 2D Type Hokuyo URG-04LX	Caméra 3D à bas coût Type PrimeSense
Technologie	Mesure de déphasage	Lumière IR structurée
Prix	1000 $	150 $
Champ de vision horizontal Champ de vision vertical	240 ° N.A.	57 ° 43 °
Nb de points à l'horizontale Nb de points à la verticale	672 N.A.	320 240
Densité horizontale Densité verticale	2.8 pts/ ° N.A.	5.61 pts/ ° 5.58 pts/ °
Portée minimale Portée maximale (dégradée)	2 cm 400 cm (500 cm)	80 cm 350 cm (500 cm)
Précision	1 cm	de 1 cm à 15 cm
Fréquence	10 Hz	33 Hz

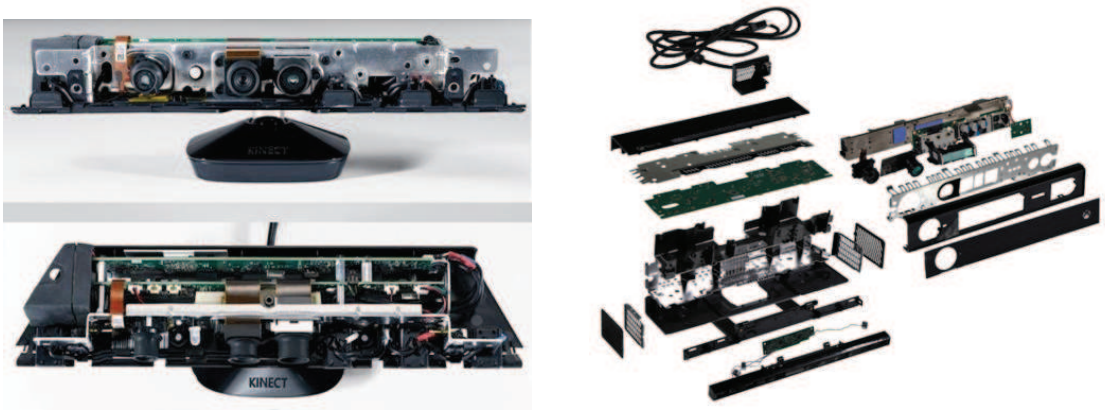
TABLE 1.1 – Comparaison des caractéristiques du télémètre laser et de la caméra 3D

1.2.1 Caméra 3D

Fonctionnement

L'exemple le plus connu de caméra 3D à bas coût est le capteur Kinect commercialisé par Microsoft en novembre 2010 qui a atteint les 25 millions d'unités vendues courant 2013. Ce capteur, comme la plupart de ses concurrents et notamment l'Asus Xtion Pro, repose sur la technologie de mesure de profondeur de la startup PrimeSense. Comme les figures 1.1 et 1.2 le montrent, ce type de caméra 3D fonctionne grâce à la présence d'une caméra (infrarouge) qui va capter l'image dans l'environnement d'un motif connu émis par un projecteur (infrarouge).

Ce principe est désigné en acquisition et modélisation 3D sous le nom de lumière structurée. Dans le cas de la Kinect, la lumière choisie se place dans le spectre du proche infrarouge la rendant invisible à l'oeil nu. La caméra infrarouge acquiert ainsi une image *a priori* indépendante des conditions lumineuses ambiantes ce qui permet d'utiliser le capteur



(a) Kinect (sortie en nov. 2010) (b) Kinect v2 (prévue pour l'été 2014)

FIGURE 1.1 – Vues de l'intérieur des capteurs Kinect et Kinect 2 (source : IHS).

dans le noir complet. Pour autant, si la caméra 3D n'est pas sensible à la lumière visible, elle n'est pas utilisable de jour, en extérieur, du fait de la présence des longueurs d'onde du proche infrarouge dans la lumière du Soleil (l'infrarouge constitue plus de la moitié du spectre d'émission du Soleil). Lorsque le milieu présente trop de rayons infrarouges, le motif projeté n'est plus suffisamment visible par la caméra et le calcul des profondeurs ne peut aboutir.

La figure 1.3 montre l'aspect du motif projeté perçu par la caméra infrarouge lorsque la scène est constituée d'un unique plan (à gauche) ou lorsqu'un objet rectangulaire est placé devant ce plan. La lumière projetée est en fait matérialisée par cette multitude de points clairs qui forment un motif non aléatoire dont la différence avec le motif projeté connu permet de calculer une disparité de laquelle est issue l'information de profondeur.

A l'image de la construction de carte de profondeur par caméra stéréoscopique, la caméra 3D à bas coût à lumière structurée repose sur le parallélisme du plan image de la caméra infrarouge et du projecteur. En effet, si les plans images sont parallèles et si les centres optiques sont à la même hauteur, alors les lignes épipolaires sont horizontales dans les images (image acquise et image de référence représentant le motif connu). Il est alors rapide et facile de trouver dans l'image la ligne épipolaire correspondante et de trouver pour chaque point lumineux du motif projeté le meilleur pixel dans l'image infrarouge.

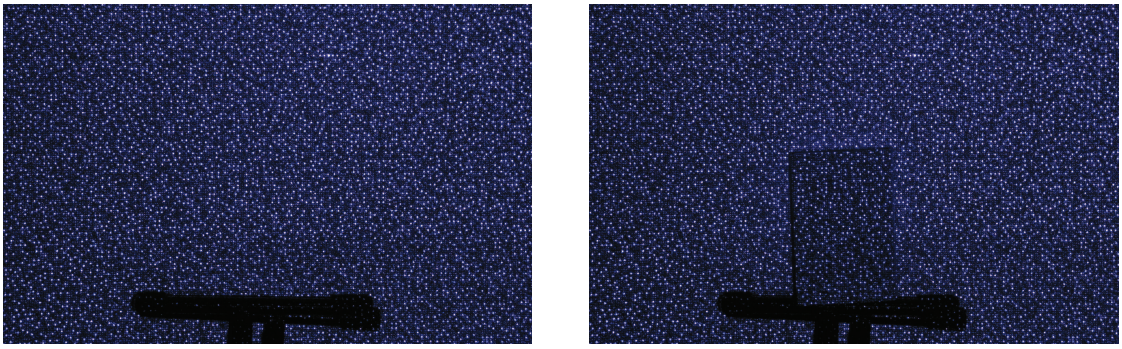


FIGURE 1.2 – Schéma de fonctionnement de la caméra 3D (source : PrimeSense.com).

Avec cette mise en correspondance, il suffit ensuite de trianguler les points afin d'obtenir l'information de profondeur :

$$d = \frac{B * f}{z}$$

avec d la disparité ou encore la différence de position d'un même point dans les deux images), B la *baseline* (la distance entre le centre de la source et le centre du capteur), f la focale et z la profondeur.



(a) Projection sur un plan (b) ... et avec un autre objet

FIGURE 1.3 – Images du motif projeté sur un plan et avec un élément à une profondeur différente (source : FuturePicture.org).

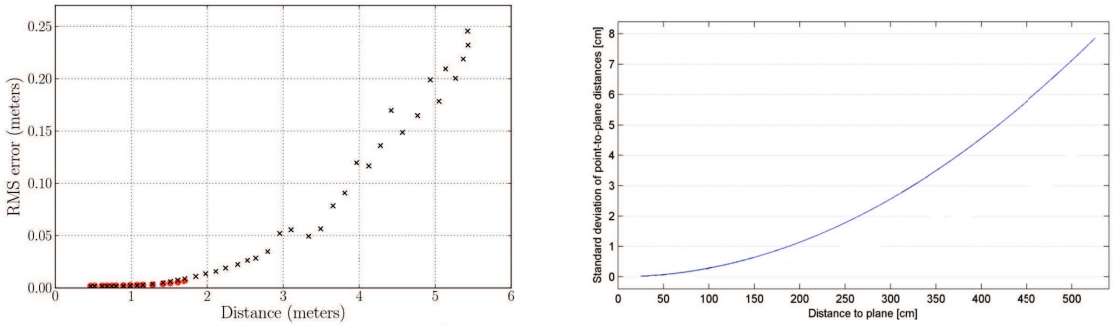
Limites

Dans [Khoshelham et Elberink, 2012], les auteurs analysent les points faibles du capteur de profondeur de la Kinect pour la cartographie en milieu intérieur. Du point de vue de l'application envisagée, moins stricte que les applications de métrologie pure, la précision et la résolution de la caméra 3D peuvent être suffisantes si certaines conditions sont respectées.

Les auteurs mettent tout d'abord en exergue la nécessité d'un étalonnage précis entre les capteurs. Cette étape doit pouvoir corriger les erreurs systématiques. S'agissant du capteur de profondeur, ils choisissent d'étalonner la caméra IR et donc prennent en considération un modèle du sténopé augmenté par la gestion des distorsions optiques radiales. Il s'agit ensuite d'estimer les paramètres de rotation et de translation entre les capteurs. Pour eux, il s'agit d'une caméra conventionnelle et de la caméra IR. Pour autant, l'étalonnage proposé n'est pas suffisant puisqu'en refaisant un étalonnage complet, il est montré dans [Smisek *et al.*, 2013] que la caméra IR de la Kinect présente plus de distorsions tangentielles que radiales.

Par ailleurs, les auteurs montrent qu'il existe aussi une erreur aléatoire sur la mesure de profondeur qui semble être quadratiquement croissante avec la distance de mesure (4cm d'erreur à 5m). De plus, cette erreur semble d'autant plus fréquente que le point observé est situé aux extrémités du nuage de points détecté (c'est-à-dire sur les bords de l'image de

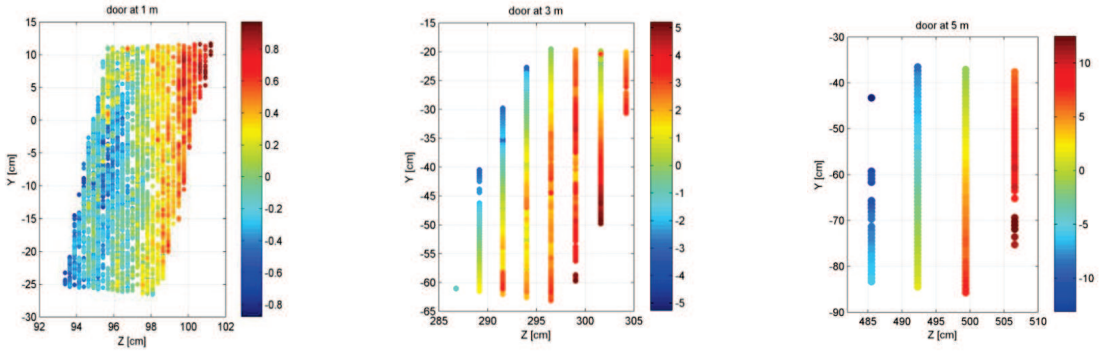
profondeur). L'écart-type des résidus d'une modélisation par un plan d'un nuage de points réellement planaire est ainsi quatre fois plus important à 5m qu'à 2.50m. Ces données sont confirmées par [Khoshelham et Elberink, 2012] et [Teichman *et al.*, 2013] comme le montre la figure 1.4.



(a) Plan à 1 m (b) Plan à 3 m

FIGURE 1.4 – Erreur moyenne et écart-type de la Kinect selon les données de [Khoshelham et Elberink, 2012] et [Teichman *et al.*, 2013]

Enfin, la résolution de mesure de profondeur (c'est-à-dire le long de l'axe Z de la caméra 3D), pour sa part, décroît quadratiquement avec la distance. En effet, l'espacement entre les mesures atteint environ 7cm le long de l'axe optique à une distance de 5m comme le montre la figure 1.5.



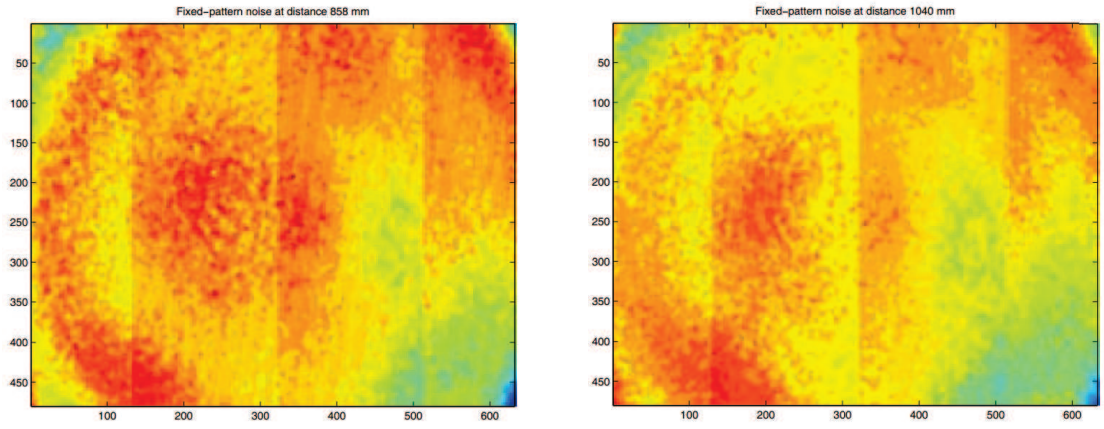
(a) Plan à 1 m (b) Plan à 3 m (c) Plan à 5 m

FIGURE 1.5 – Résolution de la Kinect [Khoshelham et Elberink, 2012]

L'erreur aléatoire mentionnée ci-dessus est confirmée et analysée par [Smisek *et al.*, 2013] qui met en évidence une distorsion semblable à une distorsion optique classique

mais portant sur les valeurs de profondeur (une image de cette distorsion est visible en figure 1.6). Plus tard, [Herrera *et al.*, 2012] appellera cette erreur *distorsion de disparité* et reformulera la correction proposée en ajoutant à la disparité d le produit du motif de distorsion D_δ connu pour chaque pixel de coordonnées (u, v) et d'une fonction exponentielle de la disparité et des coefficients de distorsion α_0 et α_1.

$$d_k = d + D_\delta(u, v) \ . \ e^{\alpha_0 - \alpha_1 d}$$



(a) Plan à 85 cm

(b) Plan à 1 m

FIGURE 1.6 – Résidus d'une régression planaire montrant la distorsion de disparité selon les données de [Smisek *et al.*, 2013] à deux profondeurs différentes

1.2.2 Télémètre à balayage laser

Fonctionnement

Le télémètre laser 2D à balayage à bas coût présente un fonctionnement simple basé sur des composants économiques. Il n'est souvent pas possible de faire des mesures de temps de vol du fait de la précision nécessaire des horloges (environ 60 ps pour une mesure de 1cm de précision). Dans les télémètres à bas coût, une source laser modulée en amplitude émet un signal périodique qui sera capté en retour par une photodiode. La distance d parcourue par l'onde (égale au double de la distance de l'obstacle) est obtenue en utilisant le décalage de

phase entre l'émetteur et le récepteur et la vitesse de propagation de l'onde c. Le déphasage ϕ intervient dans une relation entre le temps de vol t et la fréquence de modulation f :

$$\phi = 2\pi * t * f = 2\pi * \frac{d}{c} * f \quad => \quad d = \frac{\phi * c}{2\pi * f}$$

Une mesure est réalisée pour chacune des positions d'un miroir rotatif comme le montre la figure 1.7 ce qui donne le balayage complet d'un plan en environ 100 ms.

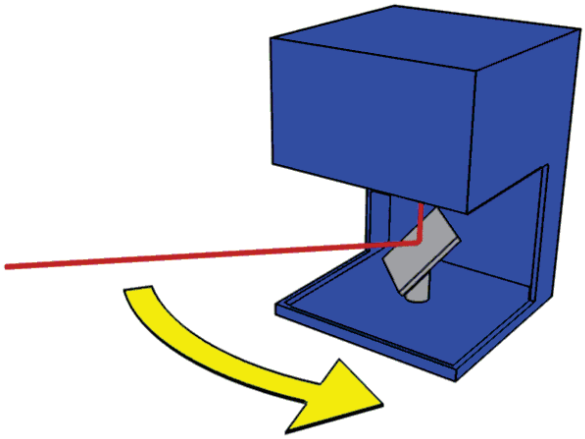


FIGURE 1.7 – Schéma de fonctionnement d'un télémètre laser à balayage (domaine public/Mike1024)

Limites

Il existe de nombreux travaux et notamment des rapports techniques très complets sur la caractérisation de divers capteurs et particulièrement de nombreux télémètres laser à balayage [Kweon *et al.*, 1991] [Alwan *et al.*, 2005] [Pito, 1995]. Parmi eux, [Okubo *et al.*, 2009] porte sur le même capteur à bas coût que celui utilisé pour les travaux de cette thèse : le Hokuyo URG-04LX.

Dans l'ensemble de ces travaux, plusieurs facteurs influençant les mesures sont mis en évidence :

— **La température et le temps de fonctionnement**

Un télémètre laser ne produira une mesure stable qu'après un certain temps de

fonctionnement. Cette durée correspond à la stabilisation de la différence entre la température ambiante et la température du télémètre. En effet, lorsque le laser est en fonctionnement, les composants électriques, électroniques ou mécaniques génèrent une chaleur qui va élever la température du télémètre au-delà de la température ambiante. Cette élévation cesse à partir d'un certain temps (lorsque la dissipation compense l'échauffement), stabilisant la mesure de distance du capteur. La dérive est estimée par [Okubo *et al.*, 2009] à 0.3% de la distance mesurée (soit 6 mm à 2000 mm) après 90 minutes sur le Hokuyo URG-04LX. [Pascoal *et al.*, 2008] montre une dérive de 2 cm après 40 minutes lorsque la température ambiante est maintenue à 20 ° C.

— **Le temps de balayage et le temps de transfert des mesures**

Le délai entre le début et la fin du balayage et celui entre la fin du balayage et la réception des données sur l'ordinateur induisent deux types d'erreurs différents. Tout d'abord, l'échantillon de mesures subit une déformation des données dès lors que le centre de rotation du télémètre laser subit une translation ou une rotation. A titre d'exemple, si la durée de balayage est fixée à 100 ms et que le télémètre laser se déplace à 50 cm/s (vitesse acceptable pour un robot d'intérieur), le centre de mesure se sera déplacé de 5 cm entre la première mesure et la dernière. De même, si le robot effectue une rotation, le balayage laser n'est plus effectué sur l'angle déterminé par le constructeur mais sur l'angle augmenté ou diminué de l'angle de rotation du robot pendant la durée du balayage. Enfin, le temps de balayage et le temps de transfert des données influent sur la datation des mesures, ce qui peut avoir des conséquences dans l'exploitation des échantillons.

— **La distance de l'objet à mesurer**

Les télémètres laser offrent une résolution angulaire constante. La distance de l'objet à mesurer influe donc sur la qualité du périmètre de l'objet (la quantité de points sur une surface diminue avec la distance de l'objet). L'ensemble des études et des fiches techniques présente les télémètres laser avec une précision constante jusqu'à

une certaine distance puis avec une précision égale à une fraction de la mesure au-delà (par exemple pour le Hokuyo URG-04LX : 10 mm entre 20 et 1000 mm puis 1% au-delà).

— **L'angle d'incidence du rayon laser sur la surface de l'objet à mesurer**
Comme le montre [Okubo *et al.*, 2009], la répartition des mesures d'un objet à distance constante s'élargit avec l'augmentation de l'angle d'incidence. Ainsi la variabilité des mesures augmente significativement. Au delà de 40 ° , les auteurs n'ont pas pu faire de mesure fiable de distance.

— **Le matériau et la couleur de la surface de l'objet à mesurer**
Tous les travaux de caractérisation des télémètres laser mettent en évidence l'influence de la surface du matériau, de sa couleur et de sa capacité d'absorption de la lumière, sur la mesure. L'état de surface (que l'on peut résumer en mat ou brillant) est le paramètre le plus influant. Une surface trop brillante va perturber les capteurs qui produiront des erreurs parmi les mesures. Si la couleur de l'objet change sans que sa brillance ne varie, la mesure sera aussi modifiée. Enfin, l'intensité lumineuse reçue par le télémètre laser semble avoir une influence puisqu'un objet noir, blanc ou gris ne sera pas mesuré à la même distance.

— **La taille du point laser**
La taille du point (qui augmente avec la distance) augmente l'effet de chatoiement. [Baribeau et Rioux, 1991] et [Dorsch *et al.*, 1994] montrent que les tavelures ou le chatoiement (*speckle* en anglais) ont une influence sur les mesures de certains types de télémètre laser. Une taille trop importante provoque un effet dit de *Mixed pixel problem* .

— **La division du faisceau sur plus d'un plan** due à la non ponctualité du laser et à la présence d'une discontinuité de surface (un coin par exemple) provoque un mélange de mesures qui conduit à une mesure unique erronnée (*Mixed pixel problem* [Ye et Borenstein, 2002]). Ce problème fréquent fait l'objet de travaux de filtrage pour le limiter. Ces filtrages vont du plus simple filtre médian spatial (qui respecte

l'emplacement des discontinuités) aux filtres de Kalman étendus comme dans [Ye, 2008].

— **L'opacité du milieu**

[Pascoal *et al.*, 2008] utilisent une machine à fumée pour mesurer l'influence de l'opacité ambiante sur la distance de mesure des télémètres laser et trouvent une forte corrélation entre la quantité de fumée et l'erreur de mesure. D'autres travaux ont établi les mêmes types de mesure en utilisant des brumisateurs pour générer des nuages de goutelettes d'eau avec les mêmes conclusions.

— **La radiance calorifique**

[Pascoal *et al.*, 2008] ont aussi mis en évidence l'influence de la radiance calorifique de certaines surfaces sur la variabilité des mesures ainsi que sur la quantité de mesures non abouties. Plus la surface est chaude (dans un milieu domestique il s'agira des radiateurs, fours, plaques électriques, ...) plus la mesure sera dégradée jusqu'à ne plus être possible pour des hautes températures présentes en milieu industriel.

1.2.3 Les données d'étalonnage

La conclusion des travaux cités sur la Kinect est donc claire : il ne faut utiliser le capteur de profondeur de la Kinect que dans un volume réduit, entre 1 à 3 m de profondeur, pour conserver une erreur limitée au centimètre.

Concernant le télémètre laser à balayage, effectivement, l'observation d'un plan ne conduit pas à l'obtention d'un segment parfait. On observe très facilement l'influence des quatre facteurs prépondérants que sont la distance, l'angle, le matériau et la couleur. Il suffit de placer divers plans (noir, blanc, brillant, mat) à différents angles et distances devant le télémètre laser. L'observation attendue doit représenter un segment à l'endroit de l'intersection entre le plan du laser et le plan observé. Les problèmes de mesure génèrent un nuage de points qui peut être très différent d'un segment. Deux solutions peuvent être mises en oeuvre conjointement. La première consiste à choisir ce plan tel que son matériau et sa couleur n'induisent pas d'erreur de mesure. Un tel plan est présenté à droite sur

la figure 1.8 au côté d'un plan étalon traditionnel de type damier. Dans la section 1.5, nous montrerons l'influence du choix du plan sur la qualité de l'étalonnage. La deuxième solution repose sur le post-traitement des données par une régression de l'observation laser pour obtenir un segment correct. Malheureusement, cette régression a pour effet de limiter l'information utile des données laser à deux paramètres linéairement indépendants que seule une augmentation du nombre d'observations pourra compenser.

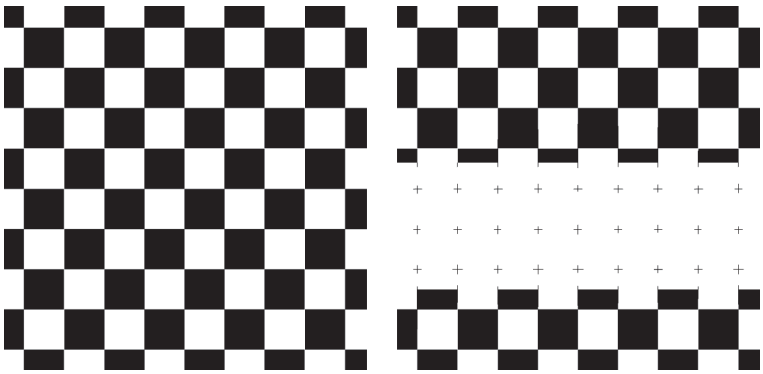


FIGURE 1.8 – Exemples de plan étalon : le plan damier traditionnel (à gauche) et le plan proposé pour l'usage du télémètre laser (à droite)

1.3 Etat de l'art

L'étalonnage extrinsèque entre les deux capteurs repose sur l'hypothèse fondamentale que la transformation géométrique entre le télémètre laser et la caméra 3D est rigide, c'est-à-dire constante (invariante quel que soit le paramètre considéré) et connue. La connaissance précise et aisée de cette transformation n'étant pas compatible avec une mesure manuelle, elle fait l'objet de l'étalonnage dit *extrinsèque* - par opposition à l'étalonnage *intrinsèque* - entre la caméra 3D et le télémètre laser.

De nombreux travaux ont traité de l'étalonnage intrinsèque et extrinsèque des caméras conventionnelles. Z. Zhang dans [Zhang, 1999] et [Zhang, 2000] a développé une méthode à base de plan d'étalonnage très largement utilisée de part le monde dans de nombreux outils d'étalonnage et notamment dans le fameux logiciel libre *Camera Calibration Toolbox for Matlab* de J.-Y. Bouguet [Bouguet, 2003].

Cette méthode se base sur la minimisation d'une erreur de reprojection de point d'intérêt d'un motif connu (par exemple un damier noir et blanc placé sur un plan) dans l'image de la caméra obervant ce motif. Les paramètres de cette minimisation sont au nombre de 15, à savoir 9 paramètres intrinsèques (5 paramètres modélisent les distorsions optiques et 4 paramètres représentent la distance focale et le point principal) et 6 paramètres extrinsèques qui font écho aux 6 degrés de liberté d'un objet rigide (3 paramètres représentent la rotation et 3 la translation entre le repère caméra et le repère monde généralement placé sur le plan étalon).

A partir de ces travaux, d'autres ont proposé une solution pour étalonner conjointement un télémètre laser et une caméra. Les premières solutions [Jokinen, 1999, McIvor, 1999] reposent sur la visualisation par la caméra du point d'intersection entre un plan étalon et le rayon laser. Cela correspond à deux situations : soit le rayon laser est émis dans le domaine du visible et une caméra conventionnelle peut être capable de visualiser le point, soit la caméra est capable de percevoir une image sur la même longueur d'onde que le laser (souvent le proche infrarouge).

Ces travaux n'utilisent pas de télémètre mais un laser *ligne* et le principe inverse de la télémétrie par lumière structurée. En effet, pour un objet de forme, de position et d'orientation connues, il est possible de prédire la déformation d'une ligne laser projetée sur l'objet. A l'inverse, c'est-à-dire lors de l'observation de la déformation d'une ligne projetée sur un objet de forme connue, il devient possible de déterminer la position relative de la source laser par rapport à l'objet. En déterminant la position relative de la caméra par rapport à l'objet grâce à la méthode de Z. Zhang, la détermination de la transformation entre la source laser et la caméra est alors immédiate.

Pour les nombreux cas où il n'est pas possible d'observer directement le faisceau laser, Q. Zhang and R. Pless ont proposé dans [Zhang et Pless, 2004] un processus d'étalonnage extrinsèque entre un télémètre laser et une caméra couleur. Cette méthode utilise aussi un damier placé sur un plan étalon mais repose sur la forme planaire de la mire d'étalonnage.

Dans cette méthode, il est simplement observé que tout point $\mathbf{p}_L$ relevé lors du balayage

$scan_i$ du télémètre laser et appartenant à la mire d'étalonnage est situé sur un plan (le plan constitué par la mire d'étalonnage). De fait, en appelant $\mathbf{H}_L^C$ la transformation rigide entre la caméra conventionnelle et le télémètre laser, $\mathbf{n}_C$ et d_C les paramètres du plan étalon dans le repère caméra (normale au plan et distance du plan), on en déduit la contrainte suivante :

$$\forall \mathbf{p}_L \in scan_i \cap mire_i, \quad \mathbf{n}_C^\top \mathbf{H}_L^C \mathbf{p}_L - d_C = 0 \tag{1.1}$$

avec $\{scan_i\}_{\forall i \in \mathbb{N}}$ et $\{mire_i\}_{\forall i \in \mathbb{N}}$ respectivement l'ensemble des points détectés par le télémètre laser et l'ensemble des points consituant le plan étalon.

En utilisant la méthode de Z. Zhang, il est possible d'extraire les paramètres intrinsèques de la caméra si on dispose de la connaissance exacte du motif imprimé sur la mire d'étalonnage. $\mathbf{p}_L$ étant une observation du laser, $\mathbf{n}_C$ et d_C étant calculés à partir des observations des points d'intérêt d'un motif (un damier) par la caméra conventionnelle, il devient possible d'estimer $\mathbf{H}_L^C$ si le nombre d'observation est suffisamment grand.

Dans la littérature, il n'existe pas, à ce jour, d'étalonnage extrinsèque entre une caméra 3D à bas coût et un télémètre à balayage laser. Les travaux les plus proches [Maddern *et al.*, 2012] consistent à réaliser cet étalonnage entre un télémètre 3D et un couple de télémètres 2D. Pour autant, les auteurs indiquent la nécessité d'effectuer une mesure de la transformation relative qui sera raffinée par leur méthode. De plus, ils mentionnent la nécessité que le véhicule transportant les capteurs effectue des accélérations dans les 6 degrés de liberté. Cette méthode n'est donc pas applicable dans le cas d'un robot mobile en milieu intérieur évoluant sur un plan (principales accélérations sur trois degrés de liberté, les accélérations sur les trois autres degrés sont éventuellement dues aux défauts mécaniques et négligeables).

Dans notre situation, nous ne pouvons ni observer le faisceau laser ni générer des accélérations dans les 6 degrés de liberté. La méthode la plus adaptée semble donc reposer sur le paradigme de l'appartenance au plan tel que proposé par Q. Zhang et R. Pless dans [Zhang et Pless, 2004]. Néanmoins, le type d'image produit par la caméra 3D ne permet

pas de représenter simplement un motif d'étalonnage puisqu'elle représente l'information de profondeur. L'observation de la mire proposée génère donc simplement un nuage de points bruité représentant un plan uni sans possibilité d'en extraire les points remarquables nécessaires à cette méthode.

Comme nous l'avons déjà évoqué, les caméras 3D à bas coût de type Kinect sont composées d'une caméra couleur (RGB), d'un projecteur et d'une caméra infrarouge (IR). Le premier dispositif permet la génération d'une image couleur alors que le deuxième et le troisième sont utilisés pour produire l'image de profondeur. Dans ce chapitre et les suivants, nous ferons l'hypothèse que cette image de profondeur peut être considérée comme générée par une caméra virtuelle de profondeur qui respecte le modèle du sténopé (en anglais *pin-hole model*).

Il existe donc une matrice $\mathbf{K}_K$ appelée matrice des paramètres intrinsèques qui, associée à une matrice de rotation $\mathbf{R}_W^K$ et un vecteur de translation $\mathbf{t}_W^K$ permet de passer d'un repère $\mathcal{F}_W$ exprimé en coordonnées monde au repère image de la caméra 3D. Un point observé de coordonnées $\mathbf{p}_W = [x_W, y_W, z_W]^\top$ dans $\mathcal{F}_W$ sera représenté dans l'image de la caméra 3D par un point de coordonnées $\mathbf{m}_K = [u_K, v_K]^\top$ en utilisant le modèle du sténopé :

$$\mathbf{m}_K \sim \mathbf{K}_K(\mathbf{R}_W^K.\mathbf{p}_W + \mathbf{t}_W^K) \tag{1.2}$$

avec

$$\mathbf{K}_K = \begin{pmatrix} f_u & s_{uv} & c_u \\ 0 & f_v & c_v \\ 0 & 0 & 1 \end{pmatrix}$$

c_u et c_v sont les coordonnées du point principal dans le plan image. f_u et f_v représentent les distances focales selon les axes u et v. On suppose que le *skew* s_{uv} est négligeable, les pixels étant supposés rectangulaires. Notons que le modèle utilisé ne tient pas compte des distorsions optiques qui ne sont pas très importantes dans le cas de la caméra Kinect [Smisek *et al.*, 2013].

La plupart des travaux basés sur une caméra 3D à bas coût de type Kinect et utilisant à la fois la caméra couleur et la caméra de profondeur repose sur la connaissance de la transformation géométrique entre la caméra RGB et la caméra IR [Burrus, 2011]. Une telle transformation est tout à fait valide pour représenter les pixels de la caméra RGB dans l'image IR (notamment par une droite épipolaire) et vice versa. Pour appliquer le même étalonnage entre la caméra RGB et la caméra virtuelle de profondeur, ces travaux formulent implicitement que les paramètres intrinsèques de la caméra IR et de la caméra virtuelle de profondeur sont identiques, que les repères associés à ces deux capteurs sont confondus et que les distorsions de la caméra de profondeur sont identiques à celles de la caméra IR.

D'après ces travaux, l'utilisation d'un étalonnage entre caméra IR et caméra RGB permet de produire des reprojections suffisamment précises pour leurs applications. Pourtant, il n'est pas démontré que l'on peut confondre totalement cette caméra virtuelle de profondeur avec la caméra IR. Si les deux caméras sont identiques alors leurs paramètres intrinsèques, extrinsèques et distorsions doivent aussi être identiques et tout point du repère monde doit avoir les mêmes coordonnées dans l'image de la caméra IR et dans l'image de la caméra de profondeur.

Quand bien même la lumière du télémètre laser serait visible par la caméra IR, pour une même scène, en comparant les images produites par la caméra IR et par la caméra de profondeur, il est évident que la caméra IR et la caméra 3D ne sont pas identiques puisque les objets de la scène ne sont pas représentés au même endroit de l'image (nous en faisons la démonstration dans le chapitre 3 avec la figure 3.1). Il semble donc difficile de pouvoir utiliser un étalonnage entre le laser et la caméra IR en lieu et place d'un étalonnage entre le laser et la caméra 3D. Il paraît donc nécessaire de réaliser un nouvel étalonnage directement pour chaque couple de caméra 3D et de télémètre laser.

1.4 Modèle d'étalonnage extrinsèque

Pour réaliser cet étalonnage, nous allons supposer (sans perte de généralité) que le repère monde et le repère caméra 3D sont confondus. Ainsi, la matrice de rotation et le vecteur de translation permettant de passer d'un point exprimé dans le repère monde vers le repère caméra 3D sont respectivement l'identité et le vecteur nul de dimension 3 ($\mathbf{R}_K = \mathbf{I_{3X3}}$ and $\mathbf{t}_K = [0,0,0]^\top$). L'équation (1.2) se simplifie et devient :

$$\mathbf{m}_K \sim \mathbf{K}_K.\mathbf{p}_K = \mathbf{K}_K(\mathbf{R}_K.\mathbf{p}_W + \mathbf{t}_K) = \mathbf{K}_K.\mathbf{p}_W$$

Si on développe le vecteur $\mathbf{p}_K$ qui contient les coordonnées du point 3D observé par la caméra virtuelle de profondeur, on a $\mathbf{p}_K = [x_K, y_K, z_K]^\top$. L'image de la caméra 3D encode une profondeur $z_K = \text{depth}(u_K, v_K)$ pour chaque couple de coordonnées pixelliques (u_K, v_K). On calcule alors les coordonnées restantes :

$$\mathbf{p}_K = \begin{bmatrix} z_K \dfrac{u_K - c_u}{f_u} \\ z_K \dfrac{v_K - c_v}{f_v} \\ \\ z_K \end{bmatrix}$$

Soient $\mathbf{R}_L^K$ et $\mathbf{t}_L^K$ respectivement la matrice de rotation et le vecteur de translation permettant le passage du repère lié au télémètre laser $\mathcal{F}_L$ vers la caméra 3D $\mathcal{F}_K$. Le but du processus d'étalonnage extrinsèque entre la caméra 3D et le télémètre laser à balayage est d'estimer $\mathbf{R}_L^K$ et $\mathbf{t}_L^K$ qu'on appellera par la suite *paramètres extrinsèques* de l'étalonnage laser - caméra 3D. L'équation suivante décrit la transformation d'un point $\mathbf{p}_L$ observé par le télémètre laser vers le repère $\mathcal{F}_K$ de la caméra 3D :

$$\mathbf{p}_K = \mathbf{R}_L^K.\mathbf{p}_L + \mathbf{t}_L^K \tag{1.3}$$

Soit (π) un plan qui peut être décrit par quatre paramètres $[a, b, c, d]$ qui respectent la

contrainte $ax + by + cz + d = 0$. Le vecteur unitaire normal au plan $\mathbf{n}_{(\pi)}$ et la distance euclidienne $d_{(\pi)}$ du plan à l'origine du repère $\mathcal{F}_K$ sont définis comme :

$$\mathbf{n}_{(\pi)} = \frac{1}{\sqrt{a^2+b^2+c^2}} \cdot \begin{bmatrix} a \\ b \\ c \end{bmatrix} \qquad\qquad d_{(\pi)} = \frac{1}{\sqrt{a^2+b^2+c^2}} \cdot d$$

Comme exprimé par l'équation (1.1) et en utilisant l'équation (1.3), tout point $\mathbf{p}_L$ supposé appartenir au plan (π) doit respecter l'égalité suivante :

$$\mathbf{n}_{(\pi)}{}^{\top}\mathbf{p}_K - d_{(\pi)} = \mathbf{n}_{(\pi)}{}^{\top}(\mathbf{R}_L^K \mathbf{p}_L + \mathbf{t}_L^K) - d_{(\pi)} = 0 \tag{1.4}$$

Le télémètre laser à balayage mesure des obstacles sur un unique plan (π_L). Pour calculer $\mathbf{n}_{(\pi)}$ et $d_{(\pi)}$, il faut que la caméra 3D puisse observer un plan dans la scène. Pour disposer des points $\mathbf{p}_L$ issus du télémètre laser, il faut que ce même plan (π) soit observable par ce capteur. On peut indifféremment - c'est-à-dire sans ajouter de contrainte particulière - fixer $\mathcal{F}_L$ tel que (π_L) corresponde au plan défini par les axes x et y de $\mathcal{F}_L$. C'est à dire que seules les coordonnées selon les axes x et y seront non nulles et celles selon l'axe z seront toutes nulles. Alors, le vecteur $\mathbf{p_L} = [x_L, y_L, z_L]^{\top} \in \mathbb{R}^3$ correspondant à une observation du télémètre laser peut être réécrit comme $[x_L, y_L, 0]^{\top}$ ou encore comme le vecteur de $[x_L, y_L]^{\top} \in \mathbb{R}^2$ dont les coordonnées homogènes sont $\tilde{\mathbf{p}}_L = [x_L, y_L, 1]^{\top}$.

Cette réécriture en coordonnées homogènes permet de simplifier l'expression de la transformation rigide définie par $\mathbf{R}_L^K$ et $\mathbf{t}_L^K$. Cette transformation s'exprime alors par une matrice $\mathbf{H}_L^K \in \mathbb{R}^3 \times \mathbb{R}^3$ dont la construction particulière à partir de la matrice de rotation et du vecteur de translation découle de l'uniplanarité des observations du télémètre laser :

$$\mathbf{H}_L^K = \mathbf{R_L^K} \begin{pmatrix} 1 & 0 & \\ 0 & 1 & \mathbf{t_L^K} \\ 0 & 0 & \end{pmatrix} \tag{1.5}$$

L'équation (1.4) devient alors :

$$\mathbf{n}_{(\pi)}{}^{\top}\mathbf{H}_L^K\tilde{\mathbf{p}}_L - d_{(\pi)} = 0 \tag{1.6}$$

Puisque $\mathbf{H}_L^K \in \mathbb{R}^3 \times \mathbb{R}^3$ est l'expression de la transformation rigide à trouver, il y a donc 9 inconnues h_i à estimer :

$$\mathbf{H_L^K} = \begin{pmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{pmatrix} \tag{1.7}$$

L'équation (1.6) exprime une contrainte unique, elle ne permet pas seule de résoudre le problème de l'étalonnage. Pour augmenter le nombre de contraintes, il faut considérer l'ensemble des points issus des observations du télémètre laser (théoriquement au moins 9 contraintes sont nécessaires).

$$\begin{cases} \mathbf{n}_{(\pi)}{}^{\top}\mathbf{H}_L^K\tilde{\mathbf{p}}_{L1} - d_{(\pi)} = 0 \\ \mathbf{n}_{(\pi)}{}^{\top}\mathbf{H}_L^K\tilde{\mathbf{p}}_{L2} - d_{(\pi)} = 0 \\ \qquad\qquad \vdots \\ \mathbf{n}_{(\pi)}{}^{\top}\mathbf{H}_L^K\tilde{\mathbf{p}}_{Ln} - d_{(\pi)} = 0 \end{cases} \tag{1.8}$$

1.4.1 Modèle linéaire

La matrice $\mathbf{H}_L^K$ étant une matrice 3x3, ce système de n équations pour 9 inconnues mène théoriquement à une solution unique si $n \geq 9$. Il suffit alors de mesurer 9 points du plan à l'aide du télémètre laser à balayage pour obtenir une solution à ce problème d'étalonnage extrinsèque. Etant donné l'imprécision des mesures issues des capteurs, ni $\mathbf{n}_{(\pi)}$ ni $\tilde{\mathbf{p}}_{Li}$ ni $d_{(\pi)}$ ne sont connus exactement. Cette incertitude nécessite d'avoir plus d'équations que d'inconnues et plus la différence sera importante moins la solution sera sensible au bruit de mesure.

Lorsque cela est possible, il est d'usage de réécrire le système sous une forme linéaire plus lisible et dont l'expression de la solution est directe. Pour ce faire, il faut réécrire la matrice $\mathbf{H}_L^K$ sous forme d'un vecteur $\mathbf{h}_L^K = [h_1, h_2, h_3, h_4, h_5, h_6, h_7, h_8, h_9]^\top$.

On obtient alors n équations (où n est le nombre de points produits par le télémètre laser correspondant à des mesures de distance du plan observé) de la forme :

$$\left(\mathbf{n}_{(\pi)}{}^\top \otimes \tilde{\mathbf{p}}_{Li}^\top\right) \mathbf{h}_L^K = d_{(\pi)} \otimes \mathbb{1}_n \tag{1.9}$$

Le symbole $\otimes$ représente le produit tensoriel de Kronecker.

En regroupant les termes $\left(\mathbf{n}_{(\pi)}{}^\top \otimes \tilde{\mathbf{p}}_{Li}^\top\right)$ en une matrice $\mathbf{M}$ et $d_{(\pi)} \otimes \mathbb{1}_n$ en un vecteur $\mathbf{d}$, ces équations produisent le problème linéaire suivant :

$$\mathbf{M}\mathbf{h}_L^K = \mathbf{d} \tag{1.10}$$

La solution $\tilde{\mathbf{h}}_L^K$ du problème (1.10) est donc $\tilde{\mathbf{h}}_L^K = \mathbf{M}\backslash\mathbf{d}$.

L'opérateur $\backslash$ représente la division matricielle à gauche telle que $x = A\backslash b$ est une solution de $Ax = b$. Si A est carrée et de rang maximal, $x = A\backslash b$ est équivalent mathématiquement à $x = A^{-1} * b$. Si A n'est pas de rang maximal ou si A n'est pas carrée (problème surdéterminé), x est tel que $\|Ax - b\|_2$ est minimale (au sens des moindres carrés).

Il existe un unique vecteur $x \in C^N$ de norme minimale qui minimise $||Ax - b||_2{}^2$ dans C^M et cette solution est donnée par $x = A^+ b$ où A^+ est la pseudo-inverse de A. On utilisera alors une décomposition en valeurs singulières (SVD - Singular Value Decomposition) pour déterminer la pseudo-inverse et écrire la solution $x = A^+ * b$ (où A^+ représente la pseudo-inverse de A).

Les données étant bruitées, l'observation par le télémètre laser d'un seul plan ne permet pas d'avoir suffisamment de redondance pour que la solution soit précise. Il est alors courant de multiplier les plans à observer. Le système d'équations (1.8) devient :

$$\begin{cases} {\mathbf{n}_{(\pi_1)}}^\top \mathbf{H}_L^K \tilde{\mathbf{p}}_{L11} - d_{(\pi_1)} = 0 \\ {\mathbf{n}_{(\pi_1)}}^\top \mathbf{H}_L^K \tilde{\mathbf{p}}_{L21} - d_{(\pi_1)} = 0 \\ \vdots \\ {\mathbf{n}_{(\pi_1)}}^\top \mathbf{H}_L^K \tilde{\mathbf{p}}_{Ln1} - d_{(\pi_1)} = 0 \\ {\mathbf{n}_{(\pi_2)}}^\top \mathbf{H}_L^K \tilde{\mathbf{p}}_{L12} - d_{(\pi_2)} = 0 \\ {\mathbf{n}_{(\pi_2)}}^\top \mathbf{H}_L^K \tilde{\mathbf{p}}_{L22} - d_{(\pi_2)} = 0 \\ \vdots \\ {\mathbf{n}_{(\pi_2)}}^\top \mathbf{H}_L^K \tilde{\mathbf{p}}_{Ln2} - d_{(\pi_2)} = 0 \\ \vdots \\ {\mathbf{n}_{(\pi_m)}}^\top \mathbf{H}_L^K \tilde{\mathbf{p}}_{L1m} - d_{(\pi_m)} = 0 \\ {\mathbf{n}_{(\pi_m)}}^\top \mathbf{H}_L^K \tilde{\mathbf{p}}_{L2m} - d_{(\pi_m)} = 0 \\ \vdots \\ {\mathbf{n}_{(\pi_m)}}^\top \mathbf{H}_L^K \tilde{\mathbf{p}}_{Lnm} - d_{(\pi_m)} = 0 \end{cases} \tag{1.11}$$

De même que pour l'équation (1.10), on peut réécrire le système (1.11) comme $n \times m$ équations de la forme :

$$\left({\mathbf{n}_{(\pi_j)}}^\top \otimes \tilde{\mathbf{p}}_{Lij}^\top\right) \mathbf{h}_L^K = d_{(\pi_j)} \otimes \mathbb{1}_n \tag{1.12}$$

L'énoncé du problème linéaire est le même (seule la taille des matrices et des vecteurs change). Sa résolution rend possible l'extraction de $\widehat{\mathbf{R}}_L^K$ et $\widehat{\mathbf{t}}_L^K$ à partir de la solution $\widehat{\mathbf{h}}_L^K$. En reprenant les équations (1.7) et (1.5), on obtient :

$$\widehat{\mathbf{h}}_L^K = \left[\widehat{h}_1, \widehat{h}_2, \widehat{h}_3, \widehat{h}_4, \widehat{h}_5, \widehat{h}_6, \widehat{h}_7, \widehat{h}_8, \widehat{h}_9\right]$$

$$\widehat{\mathbf{H}}_L^K = \begin{pmatrix} \widehat{h}_1 & \widehat{h}_2 & \widehat{h}_3 \\ \widehat{h}_4 & \widehat{h}_5 & \widehat{h}_6 \\ \widehat{h}_7 & \widehat{h}_8 & \widehat{h}_9 \end{pmatrix} = \widehat{\mathbf{R}}_L^K \begin{pmatrix} 1 & 0 & \vdots \\ 0 & 1 & \widehat{\mathbf{t}}_L^K \\ 0 & 0 & \vdots \end{pmatrix}$$

Et on en déduit :

$$\widehat{\mathbf{R}}_L^K = \left(\begin{bmatrix} \widehat{h}_1 \\ \widehat{h}_4 \\ \widehat{h}_7 \end{bmatrix}, \begin{bmatrix} \widehat{h}_2 \\ \widehat{h}_5 \\ \widehat{h}_8 \end{bmatrix}, \begin{bmatrix} \widehat{h}_1 \\ \widehat{h}_4 \\ \widehat{h}_7 \end{bmatrix} \times \begin{bmatrix} \widehat{h}_2 \\ \widehat{h}_5 \\ \widehat{h}_8 \end{bmatrix} \right) \tag{1.13}$$

$$\widehat{\mathbf{t}}_L^K = \widehat{\mathbf{R}}_L^{K\top} \begin{bmatrix} \widehat{h}_3 \\ \widehat{h}_6 \\ \widehat{h}_9 \end{bmatrix} \tag{1.14}$$

La solution $\widehat{\mathbf{h}}_L^K$ trouvée en résolvant le problème linéaire exprimé dans l'équation (1.10) étant perturbée par le bruit de mesure des capteurs, l'extraction de $\widehat{\mathbf{R}}_L^K$ ne produit pas une matrice de rotation : l'orthonormalité de la matrice $\mathbf{R}_L^K$ n'est pas respectée puisque l'équation (1.12) n'introduit aucune contrainte sur l'orthonormalité des deux premières colonnes de $\mathbf{H}_L^K$ telle que décrite dans l'équation (1.5).

Si $\mathbf{Q}$ est une matrice quelconque, le problème qui consiste à trouver la meilleure matrice de rotation 3×3 (au sens de la norme de Frobenius) s'écrit :

$$\min_{\mathbf{R}} \|\mathbf{R} - \mathbf{Q}\|_f \quad s.c. \;\; \mathbf{R}^\top \mathbf{R} = \mathbf{I}_{3\times 3}$$

et admet pour optimum $\mathbf{R} = \mathbf{U}\mathbf{V}^\top$ avec $\mathbf{Q} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$

Relevée par [Zhang, 2000] dans l'annexe C, cette propriété établie dans [Golub et Van Loan, 1996] montre que pour une matrice de rotation $\mathbf{R}$ approximée par une estimation dégénérée $\widehat{\mathbf{R}}$, on peut trouver une matrice de rotation $\widetilde{\mathbf{R}}$ qui minimise la norme de Frobenius $\left\|\mathbf{R} - \widetilde{\mathbf{R}}\right\|_f$ sous la contrainte $\widetilde{\mathbf{R}}\widetilde{\mathbf{R}}^\top = \mathbb{I}_{\mathbf{3}}$ en utilisant une décomposition en

valeur singulière :

$$Si \quad \widehat{\mathbf{R}}_L^K = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top \quad alors \quad \mathbf{R}_L^K \approx \widetilde{\mathbf{R}}_L^K = \mathbf{U}\mathbf{V}^\top$$
$$et \qquad \mathbf{t}_L^K \approx \widetilde{\mathbf{t}}_L^K = \widetilde{\mathbf{R}}_L^{K\top}\mathbf{h_3}$$

Comme l'approximation linéaire de $\widehat{\mathbf{R}}_L^K$ et $\widehat{\mathbf{t}}_L^K$ va être utilisée comme initialisation de méthodes non linéaires, la précision de cette extraction n'est pas primordiale.

1.4.2 Optimisation non linéaire de la translation t

L'observation de la reprojection des données de la caméra 3D dans le référentiel laser montre que la solution linéaire n'est pas suffisamment précise. Néanmoins cette solution linéaire semble fournir une bonne estimation de la matrice de rotation $\widetilde{\mathbf{R}}$. Dans un premier temps nous allons donc réaliser une optimisation non linéaire de la transformation géométrique en considérant seulement la translation $\widetilde{\mathbf{t}}$ comme variable.

Ainsi, cette optimisation concerne la minimisation de l'erreur de reprojection des points observés par le télémètre laser dans le référentiel de la caméra 3D. Pour ce faire, de même que pour le problème linéaire, nous allons utiliser la distance orthogonale d'un point à un plan afin d'obtenir un vecteur de translation optimisé $\breve{\mathbf{t}}$:

$$\min_{\breve{\mathbf{t}}} \sum_{j=1}^{m}\sum_{i=1}^{n} \left\| {\mathbf{n}_{(\pi_j)}}^\top.(\widetilde{\mathbf{R}}\mathbf{p}_{L_{ij}} + \breve{\mathbf{t}}) - d_{(\pi_j)} \right\|^2 \tag{1.15}$$

Les vecteurs normaux, les distances, les points du laser restent les mêmes que ceux employés dans le modèle linéaire. La matrice de rotation utilisée est celle obtenue à l'issue de la minimisation de la norme de Frobenius.

1.4.3 Optimisation non linéaire de la rotation R et de la translation t

Pour confirmer l'observation de bonne estimation de la rotation par la méthode linéaire, nous optimisons la rotation et la translation en même temps. En utilisant la même métrique que dans l'équation (1.15) ou en utilisant directement $\mathbf{H}$ comme dans l'équation (1.6), on peut écrire le problème sous deux formes :

$$\min_{\breve{\mathbf{H}}} \sum_{j=1}^{m} \sum_{i=1}^{n} \left\| {\mathbf{n}_{(\pi_j)}}^{\top} . \breve{\mathbf{H}} \mathbf{p}_{Lij} - d_{(\pi_j)} \right\|^2$$
$$s.c. \left\| \breve{\mathbf{h}}_{\mathbf{1}} \right\| = \left\| \breve{\mathbf{h}}_{\mathbf{2}} \right\| = 1 \; et \; \breve{\mathbf{h}}_{\mathbf{1}} . \breve{\mathbf{h}}_{\mathbf{2}} = 0$$

avec $\breve{\mathbf{h}}_{\mathbf{k}}$, la k-ième colonne de $\breve{\mathbf{H}}$.

Ces contraintes sur les colonnes $\breve{\mathbf{h}}_{\mathbf{1}}$ et $\breve{\mathbf{h}}_{\mathbf{2}}$ de $\breve{\mathbf{H}}$ impliquent l'obtention finale d'une matrice de rotation correcte et bien formée (groupe SO(3), orthonormalité respectée) à l'issue de l'extraction évoquée dans l'équation (1.13). En effet, ces contraintes forcent les deux premières colonnes de $\breve{\mathbf{H}}$ à être unitaires et orthogonales entre elles. Les matrices $\breve{\mathbf{R}}$ et le vecteur $\breve{\mathbf{t}}$ peuvent donc être extraits de $\breve{\mathbf{H}}$ comme le montrent les équations (1.13) et (1.14).

Ou encore :

$$\min_{\breve{\mathbf{r}},\breve{\mathbf{t}}} \sum_{j=1}^{m} \sum_{i=1}^{n} \left\| {\mathbf{n}_{(\pi_j)}}^{\top} . \left(\mathbf{exp} \left(\breve{\mathbf{r}} \right) \mathbf{p}_{L_{ij}} + \breve{\mathbf{t}} \right) - d_{(\pi_j)} \right\|^2 \tag{1.16}$$

avec **exp** l'application exponentielle (*exponential map*) permettant de transformer le vecteur de rotation $\breve{\mathbf{r}} = (r_x, r_y, r_z)^{\top}$ en une matrice de rotation.

On préfèrera la deuxième expression puisqu'elle permet d'utiliser des méthodes d'optimisations sans contrainte qui sont généralement plus performantes. De plus, cette formulation réduit le nombre de variables à 6 : r_x, r_y et r_z qui composent la rotation sous la forme vectorielle ainsi que les trois éléments du vecteur de translation. Comme il est couramment admis que l'augmentation de la dimension de l'espace de recherche mène à

une augmentation des minima locaux, l'équation 1.16 semble plus appropriée.

1.4.4 Métrique le long du rayon laser

L'article de [Dupont *et al.*, 2005] introduit une métrique différente spécialement conçue pour les télémètres laser à balayage. Cette métrique est en fait généralisable à n'importe quel capteur de distance basé sur la projection d'un rayon. Conçue pour une utilisation en milieu extérieur et avec des télémètres de grande portée ($> 10m$), cette métrique met de côté l'utilisation de la distance orthogonale d'un point à un plan au profit d'une distance le long du rayon de projection. Cette métrique a été introduite afin de mieux évaluer l'erreur réelle d'estimation puisqu'elle correspond à une grandeur physique. C'est lorsque l'angle d'incidence entre le rayon et la surface est important que la différence entre la métrique orthogonale traditionnelle et la métrique le long du rayon est la plus forte.

Afin de comparer dans notre contexte cette métrique avec la métrique orthogonale, nous avons implémenté cette métrique dans une optimisation non linéaire. La distance d_b le long du rayon est donnée par la fonction suivante :

$$d_b(R,t) = \left\| \mathbf{p}_K - \mathbf{s}(R,t) + \overrightarrow{\mathbf{r}(R,t)}\ e(R,t) \right\|$$

où $\mathbf{s}(\mathbf{R},\mathbf{t})$ est la position de la source du rayon (c'est-à-dire l'origine de $\mathcal{F}_L$, le repère associé au capteur télémétrique laser, exprimé dans le repère $\mathcal{F}_K$ associé à la caméra 3D). $\overrightarrow{\mathbf{r}}(\mathbf{R},\mathbf{t})$ représente alors le vecteur colinéaire au rayon, $e(\mathbf{R},\mathbf{t})$ peut être assimilé à la distance parcourue par le rayon entre sa source et le point d'impact sur le plan et $\mathbf{n}_{(\pi_j)}$ et $d_{(\pi_j)}$ sont les paramètres du plan tels que décrits précédemment :

$$\begin{aligned}
\mathbf{s}(\mathbf{R},\mathbf{t}) &= \mathbf{R}\mathbf{t} \\
\overrightarrow{\mathbf{r}(\mathbf{R},\mathbf{t})} &= \mathbf{p}_K - \mathbf{s}(\mathbf{R},\mathbf{t}) \\
e(\mathbf{R},\mathbf{t}) &= \frac{d_{(\pi_j)} - \mathbf{n}_{(\pi_j)}.\mathbf{s}(\mathbf{R},\mathbf{t})}{\mathbf{n}_{(\pi_j)}.\overrightarrow{\mathbf{r}(\mathbf{R},\mathbf{t})}}
\end{aligned}$$

Il est aisé de réécrire le problème de minimisation (1.15) sous la forme :

$$\min_{\mathbf{R},\mathbf{t}} \sum_{j=1}^{m} \sum_{i=1}^{n} d_{bij}(\mathbf{R},\mathbf{t})^2.$$

En utilisant de nouveau la formulation de Rodrigues pour écrire la matrice de rotation, le problème ci-dessus est strictement équivalent en terme de nombre de variables au problème représenté dans l'équation (1.16).

1.5 Evaluation

1.5.1 Objectifs et métriques

L'objectif de cette évaluation est de connaître la précision atteinte par les modèles d'étalonnage présentés en fonction des métriques utilisées et de déterminer la méthode la plus adaptée à notre couple de capteurs (télémètre laser Hokuyo URG-04LX et caméra 3D Kinect) et à notre situation (robotique mobile en milieu intérieur).

Dans la suite de notre évaluation, nous utiliserons deux métriques. L'une, dite *orthogonal*, correspond à la distance orthogonale entre le capteur et le plan étalon. La seconde, dite *beam*, se réfère à la distance le long du rayon projeté par le télémètre laser. La figure 1.9 montre la différence de mesures donnée par ces deux métriques. Pour une distance orthogonale d, la métrique *beam* propose une distance b dépendant de l'angle d'incidence α du rayon laser sur l'obstacle :

$$d = b * sin(\alpha) \quad => \quad \forall \alpha \in [0;\pi], \;\; b \geq d$$

1.5.2 Protocole d'évaluation

La robotique mobile d'intérieur à bas coût utilise nécessairement des capteurs embarqués à bas coût. Un exemple de ce type de robot est montré en figure 1.10. Dans cette

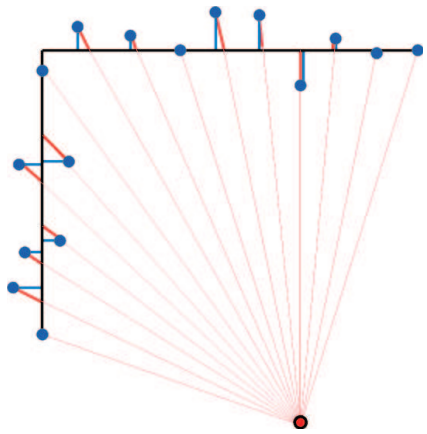


FIGURE 1.9 – Représentation de la distance selon la métrique *orthogonal* (segment en bleu) ou *beam* (segment en rouge) d'un point mesuré (disque bleu) à un obstacle réel (en noir)

évaluation, nous avons pour objectif de déterminer la précision atteinte par notre modèle d'étalonnage dans le cas d'un télémètre laser à bas coût d'une portée maximum de 4m et d'une caméra 3D à bas coût de type Kinect de 1ère génération. Ces deux capteurs sont embarqués au sein de notre robot mobile, nos jeux de données correspondent donc à des cas réels d'étalonnage et d'utilisation.

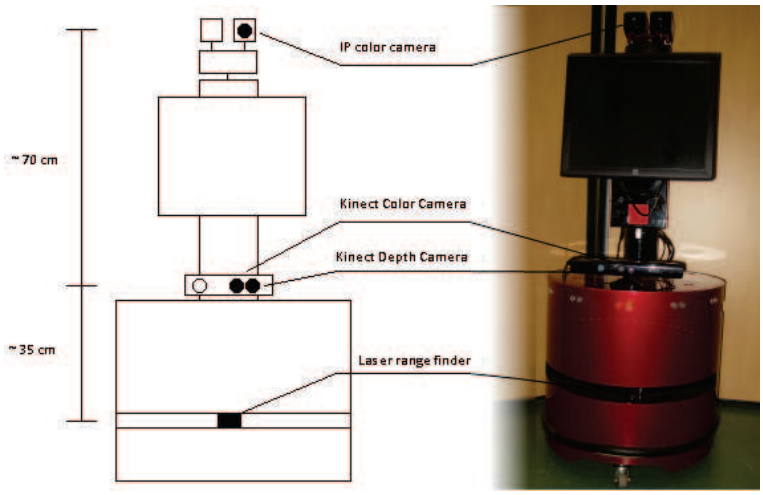


FIGURE 1.10 – Position des capteurs sur le robot Lina de l'entreprise Droïds Company

Deux jeux de données sont considérés dans cette évaluation. Chaque jeu de données est composé de 33 observations d'un plan étalon de 80×80 cm par les deux capteurs à différentes orientations et distances. Chacun de ces jeux de données est subdivisé en 16

observations utilisées pour l'étalonnage (le reste étant utilisé pour l'évaluation). Le choix de 16 observations parmi 33 permet plus d'un milliard d'évaluations. Pour des raisons de temps de calcul, seules 30 combinaisons tirées au hasard ont été évaluées. Le premier jeu de données repose sur l'observation d'un plan étalon traditionnel de type damier composé de cases de 89×89 mm. Le deuxième jeu de données est composé de mesures utilisant le plan étalon modifié tel que présenté dans la figure 1.8.

Les deux métriques employées dans cette évaluation sont celles présentées dans l'équation 1.6 et dans la section 1.4.4, soit respectivement la distance orthogonale ou le long du rayon laser entre le point observé par la caméra 3D et reprojeté dans l'espace laser 2D et le plan étalon. Pour chacune de ces métriques seront présentés les moyennes, les variances ainsi que les tests statistiques d'analyse de la variance de Kruskal-Wallis. Ce test statistique est une méthode non paramétrique qui permet de déterminer sans hypothèse de distribution (gaussienne ou autre) si la différence entre deux groupes statistiques est significativement différente.

1.5.3 Résultats

La figure 1.11 présente ainsi la moyenne et l'écart-type des deux métriques pour les deux plans étalons et pour chaque méthode d'étalonnage. Les résultats sont présentés simultanément pour les deux jeux de données : le plan traditionnel de type damier (A, en bleu foncé) et le plan modifié (B, en bleu clair).

La première observation que nous pouvons établir concerne la différence entre la métrique orthogonale et celle du rayon. Que ce soit pour les moyennes ou les écart-types, les résultats semblent sensiblement similaires : la méthode linéaire produit une erreur plus importante que les méthodes non linéaires (ce qui était prévisible) et le jeu de données B présente des performances meilleures que le jeu A et ce dans une proportion à peu près équivalente entre les deux métriques.

Si les résultats d'étalonnage avec la méthode linéaire ne sont pas suffisants pour une

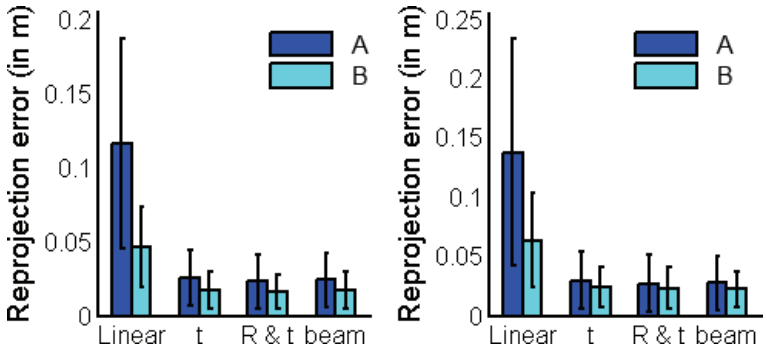


FIGURE 1.11 – Erreur de reprojection en fonction des méthodes selon la distance orthogonale (gauche) ou le long du rayon (droite) pour le plan damier (A) et le plan modifié (B)

utilisation robotique (en moyenne 11 cm et 14 cm pour les métriques orthogonale et le long du rayon avec le jeu A), nous pouvons observer que les résultats entre l'optimisation seule de $\mathbf{t}$ et l'optimisation de $\mathbf{R}$ et de $\mathbf{t}$ sont très similaires (environ 2cm). On peut en déduire que la matrice de rotation $\mathbf{R}$ est bien estimée par la méthode linéaire (ceci étant possible grâce à la matrice $\mathbf{H}$ fortement contrainte par construction par les propriétés du groupe $SO(3)$). La différence entre la méthode linéaire et les optimisations non linéaires nous permet par contre de dire que la méthode linéaire n'est pas adaptée à la détermination de la translation $\mathbf{t}$ qui évolue dans $\mathbb{R}^3$, un espace moins contraignant que $SO(3)$. La dernière observation que nous pouvons formuler concerne la différence entre l'utilisation d'un plan d'étalonnage traditionnel de type damier et le plan modifié proposé. Au vu des résultats présentés dans le graphique, le plan modifié permet une amélioration significative de l'ordre de 50% de la précision de la méthode linéaire et de 25% pour les méthodes non linéaires. Cette amélioration est fondamentale concernant la méthode linéaire puisqu'elle permet d'augmenter les performances des méthodes non linéaires aussi bien en améliorant le temps de calcul qu'en réduisant le risque de tomber dans un minimum local.

Le tableau 3.2 résume quantitativement les performances (en cm) des différentes méthodes avec les différentes métriques pour les deux types de plan étalon. Les cases vertes mettent en valeur les meilleures performances pour chaque ligne (moyenne, écart-type, minimum, maximum et pour chaque métrique). Comme la figure 1.11 le suggérait, la diffé-

rence entre l'optimisation seule de la translation et celle de la rotation et de la translation (que ce soit pour la métrique orthogonale ou celle le long du rayon) est inférieure à 1 cm. Enfin, nous pouvons remarquer que la ligne présentant les erreurs minimales est toujours négligeable (à $0 \pm \epsilon$ cm) sauf dans le cas de l'étalonnage linéaire avec un plan modifié.

TABLE 1.2 – Erreur de reprojection en cm pour l'étalonnage laser-caméra 3D utilisant un damier traditionnel ou le plan étalon modifié

Plan étalon damier traditionnel

Stat	Reproj. err	Linear	t opt.	R & t opt.	beam opt.
Mean	Orthogonal	11.62	2.55	2.33	2.42
	Beam	13.80	3.00	2.71	2.79
Std	Orthogonal	7.11	1.92	1.85	1.79
	Beam	9.54	2.43	2.45	2.29
Min	Orthogonal	0.03	0.03	0.03	0.00
	Beam	0.03	0.03	0.03	0.00
Max	Orthogonal	26.01	7.92	10.58	8.81
	Beam	41.01	12.49	15.64	13.23

Plan étalon modifié

Stat	Reproj. err	Linear	t opt.	R & t opt.	beam opt.
Mean	Orthogonal	4.66	1.75	1.64	1.71
	Beam	6.37	2.37	2.30	2.23
Std	Orthogonal	2.74	1.27	1.16	1.27
	Beam	4.01	1.69	1.77	1.54
Min	Orthogonal	0.51	0.02	0.04	0.02
	Beam	0.15	0.02	0.04	0.02
Max	Orthogonal	12.35	6.60	4.89	6.66
	Beam	16.91	8.05	7.71	6.80

En comparant les résultats d'étalonnage utilisant le damier traditionnel et le plan modifié, nous pouvons noter que l'erreur de reprojection est toujours plus faible avec le plan modifié qu'avec le damier et ce, quelle que soit la méthode d'étalonnage employée. Lorsque l'on considère la différence entre la métrique orthogonale et la métrique du rayon, il apparaît logiquement que l'erreur moyenne, la variance et l'erreur maximale sont toujours plus faibles lorsque la métrique optimisée et la métrique d'évaluation sont les mêmes. Il est donc difficile de les départager sur ce point.

Afin de mieux comprendre les différences de résultats entre les différentes méthodes et

de savoir si elles sont significatives, nous présentons les résultats de tests de Kruskal-Wallis dans le tableau 1.3. Ce tableau contient la P-value qui correspond à la probabilité de rejeter à tort l'hypothèse nulle H_0 pour chaque test statistique effectué. L'hypothèse testée pour chaque cellule (de ligne A et de colonne B) du tableau est : *La méthode A présente une erreur s_A identique à celle de la méthode B, s_B*. Soit $H_0 : s_A = s_B$. Le test statistique étant commutatif, seule la moitié du tableau est complétée. De même, les valeurs sur la diagonale ne sont pas mentionnées puisqu'elles sont théoriquement connues et égales à 1. En effet, si les méthodes sont réellement identiques, la probabilité de rejeter à tort H_0 est totale (tout rejet est une erreur) soit 1.

Dans le tableau 1.3, nous pouvons remarquer que la P-value est toujours inférieure à 10^{-6} lorsque le test est réalisé entre la méthode linéaire et n'importe quelle optimisation non linéaire. Cela signifie que la probabilité de rejeter à tort H_0 est très faible ; autant dire que l'hypothèse H_0 est fausse. Nous pouvons donc conclure que l'optimisation seule de $\mathbf{t}$ et l'optimisation de $\mathbf{R}$ et de $\mathbf{t}$ sont significativement différentes de la méthode linéaire. Par extension, on peut reformuler cette conclusion en disant que les optimisations de $\mathbf{t}$ seul et de $\mathbf{R}$ et de $\mathbf{t}$ conjointement ont une précision significativement meilleure que la méthode linéaire que ce soit avec le damier traditionnel qu'avec le plan étalon proposé. Comme nous l'avons formulé à l'observation de la figure 1.11, les tests de Kruskal-Wallis ne permettent pas de déterminer une quelconque différence significative entre les différentes méthodes non linéaires. Cela étant dit, le test de Kruskal-Wallis ne permet pas de déterminer l'acceptation à tort ou à raison de H_0 et donc de conclure sur la similarité des résultats des méthodes.

Le tableau 1.4 présente lui-aussi les résultats de tests de Kruskal-Wallis concernant la différence entre l'utilisation du plan étalon traditionnel, le damier (jeu de données A), et du plan étalon modifié présenté figure 1.8 (jeu de données B). L'hypothèse utilisée est la suivante : *La méthode X en utilisant le damier présente une erreur identique en utilisant le plan modifié*. Nous pouvons observer que toutes les P-values sont inférieures à 10^{-2} avec la métrique orthogonale. Cependant, seules la méthode linéaire et l'optimisation basée sur la métrique du rayon présentent une P-value reposant sur la métrique du rayon inférieure

à 10^{-1}. On peut ainsi conclure que le plan étalon modifié améliore significativement la précision d'étalonnage pour toutes les méthodes lorsque la précision est évaluée avec la métrique orthogonale et seulement pour la méthode linéaire et l'optimisation le long du rayon lorsque la précision est évaluée avec la métrique du rayon.

TABLE 1.3 – Résultats des tests de Kruskal-Wallis entre les différentes méthodes d'étalonnage

Plan étalon damier traditionnel

p > KW	**Linear**	**t opt.**	**R & t opt.**	**beam opt.**
Linear	X	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-6}$
t opt.	X	X	0.3428	0.6856
R & t opt.	X	X	X	0.5203

Plan étalon modifié

p > KW	**Linear**	**t opt.**	**R & t opt.**	**beam opt.**
Linear	X	$< 10^{-6}$	$< 10^{-6}$	$< 10^{-6}$
t opt.	X	X	0.6999	0.8438
R & t opt.	X	X	X	0.8514

TABLE 1.4 – Résultats des tests de Kruskal-Wallis entre l'étalonnage avec damier et l'étalonnage avec plan modifié

p > KW	**Linear**	**t opt.**	**R & t opt.**	**beam opt.**
Orthogonal	$< 10^{-6}$	$< 10^{-3}$	$< 10^{-2}$	$< 10^{-3}$
Beam	$< 10^{-6}$	0.7125	0.1242	$< 10^{-1}$

Pour résumer ce qui a été significativement établi en utilisant seulement un plan étalon de type damier :

— L'optimisation non-linéaire améliore en moyenne l'erreur de reprojection de la méthode linéaire de 79% si on évalue la précision avec la métrique orthogonale,

— L'optimisation non-linéaire améliore en moyenne l'erreur de reprojection de la méthode linéaire de 79% si on évalue la précision avec la métrique du rayon,

— L'optimisation non-linéaire améliore en moyenne l'écart-type de l'erreur de la méthode linéaire de 79% si on évalue la précision avec la métrique orthogonale,

— L'optimisation non-linéaire améliore en moyenne l'écart-type de l'erreur de la méthode linéaire de 76% si on évalue la précision avec la métrique du rayon.

Pour résumer ce qui a été significativement établi en utilisant seulement le plan étalon proposé :

— L'optimisation non-linéaire améliore en moyenne l'erreur de reprojection de la méthode linéaire de 64% si on évalue la précision avec la métrique orthogonale,
— L'optimisation non-linéaire améliore en moyenne l'erreur de reprojection de la méthode linéaire de 64% si on évalue la précision avec la métrique du rayon,
— L'optimisation non-linéaire améliore en moyenne l'écart-type de l'erreur de la méthode linéaire de 61% si on évalue la précision avec la métrique orthogonale,
— L'optimisation non-linéaire améliore en moyenne l'écart-type de l'erreur de la méthode linéaire de 58% si on évalue la précision avec la métrique du rayon.

Pour résumer ce qui a été significativement établi entre les deux types de plan :

— L'utilisation du plan proposé améliore l'erreur de reprojection de la méthode linéaire de 60% et son écart-type de 58%,
— L'utilisation du plan proposé améliore l'erreur de reprojection des méthodes non-linéaires de 30% et son écart-type de 36%.

1.6 Conclusion

L'objectif de ce chapitre était de proposer et d'évaluer une technique d'étalonnage extrinsèque d'une caméra 3D à bas coût et d'un télémètre laser 2D à balayage.

La méthode d'étalonnage présentée est une extension aux caméras 3D de la méthode de Zhang dédiée à l'étalonnage extrinsèque des télémètres laser 2D avec des caméras conventionnelles. L'usage de la caméra 3D permet d'estimer directement et plus précisément les paramètres du plan étalon en utilisant le nuage de points dense fourni. Nous levons ainsi l'obligation de créer un plan étalon avec un motif (de type damier) nécessaire à l'évalua-

tion des paramètres du plan avec une caméra conventionnelle. Cela nous permet enfin de proposer un plan étalon modifié tenant compte des spécificités des télémètres laser 2D à bas coût qui présentent de nombreuses erreurs de mesure lors de l'utilisation du damier.

Notre évaluation permet de conclure sur l'apport de cette méthode et d'un tel plan modifié. En effet, nous estimons un gain de précision de 30% et une plus faible dispersion des résultats grâce à l'usage du plan étalon proposé. L'erreur de reprojection finale moyenne est de 1.64 cm ce qui représente un erreur de 0.7% pour un point situé à 2.50 m. Cette précision est de l'ordre de la précision maximale de la caméra 3D utilisée et est amplement suffisante pour une tâche d'évitement d'obstacles en milieu intérieur.

Les principales difficultés inhérentes à la méthode présentée sont la nécessité de prendre un grand nombre de prises de vues à des positions et orientations variées afin d'obtenir une bonne précision. Cette contrainte se retrouve de la même façon dans la plupart des autres méthodes de la littérature. La méthode présentée souffre des imprécisions issues de la caméra 3D à bas coût. Si les résultats sont tout à fait suffisants pour la suite de la thèse et pour la majorité des applications robotiques, le champ de l'étalonnage intrinsèque de la caméra 3D reste à explorer plus en avant comme perspective. Notamment, les distorsions hors du modèle pin-hole, pour lesquelles la littérature ne propose pas de méthode de correction satisfaisante, à notre avis, pourraient être la cause partielle de l'erreur subsistante.

Une fois l'étalonnage entre les capteurs réalisé, la transformation rigide obtenue peut être directement utilisée pour exprimer les données de l'un des capteurs dans le référentiel de l'autre. Le chapitre suivant, montrera qu'en partant de cette connaissance, il est possible de créer un télémètre virtuel consolidé à partir des deux capteurs indépendants et produisant des données fiables pour une application d'évitement d'obstacles pour un robot mobile en milieu intérieur.

L'objectif du chapitre 2 suivant est de montrer comment cet étalonnage améliore la robustesse d'un évitement d'obstacles en permettant d'agréger les données de la caméra 3D et du télémètre laser pour créer un capteur virtuel plus fiable et robuste que chacun des capteurs pris séparément.

Chapitre 2

Application à l'évitement d'obstacles basé sur un télémètre virtuel

2.1 Introduction

Le télémètre laser et la caméra 3D ont des caractéristiques très différentes. L'un possède un champ de vision très étendu alors que la seconde offre des données denses tridimensionnelles. Les qualités de ces capteurs ont poussé le roboticien à les utiliser dans bien des situations. Le télémètre laser 2D est très souvent utilisé comme capteur de base pour un évitement d'obstacles fiable. Les caméras 3D telles que la Kinect ont été très rapidement adoptées et sont très expérimentées depuis 2011 dans les laboratoires de robotique. Pour autant, chacun de ces capteurs possède des défauts qui ne permettent pas de les utiliser seuls dans une situation réelle de robotique mobile en milieu intérieur. C'est, nous en sommes convaincus, dans l'utilisation conjointe du télémètre laser et de la caméra 3D que nous trouverons fiabilité, précision et robustesse suffisantes pour améliorer les capacités sensorielles de nos robots, l'objectif final étant l'amélioration de l'autonomie de ce robot.

Une scène est observée à la fois par le télémètre laser et par la caméra 3D. Pour que chaque capteur puisse pallier certaines déficiences de l'autre, il est nécessaire que les

champs de vision des deux capteurs se recouvrent. Selon l'application envisagée, la position et l'orientation relative des deux capteurs peuvent néanmoins varier. Par exemple, dans des applications de cartographie, de localisation, ou encore d'évitement d'obstacles, il est courant que les capteurs télémétriques soient dirigés dans la direction du mouvement du robot.

L'idée est de compléter les observations du télémètre laser par celles captées par la caméra 3D en transformant les informations de la caméra 3D vers le repère du télémètre laser pour créer un télémètre à balayage virtuel plus performant que le laser seul. Ainsi, le télémètre laser produit des couples angle/distance qui permettent de modéliser l'environnement. La caméra 3D produit une image de profondeur ou un nuage de points 3D qu'il va falloir traiter pour les rendre compatibles avec la représentation utilisée par le télémètre laser. Pour une application d'évitement d'obstacles par exemple, on pourra choisir de priviligier une sélection sécuritaire des données : lorsque le télémètre laser et la caméra 3D feront deux observations différentes, on préfèrera sélectionner l'information qui encode un risque plus fort (c'est-à-dire la distance la plus faible).

Dans ce chapitre, nous présenterons un rapide passage en revue des méthodes de la littérature dans la section 2.2 pour chacun des maillons de la chaîne de traitement de la perception vers la commande. Nous développerons ensuite les éléments de la chaîne choisie avec la perception dans la section 2.3.1, la représentation des données dans la section 2.3.2 et la commande dans la section 2.3.3. Dans la section 2.4, nous appliquerons cette utilisation conjointe du télémètre laser 2D à bas coût et de la caméra 3D à bas coût à l'évaluation de l'évitement d'obstacles dans le cas particulier d'un robot mobile en milieu intérieur inconnu.

2.2 Etat de l'art

Dans la littérature, il existe un grand nombre d'algorithmes temps réels pour la robotique basés sur l'utilisation des données d'un télémètre laser seul. On compte probablement

davantage d'algorithmes basés sur ce capteur classique de la robotique mobile que de méthodes utilisant les caméras 3D qui, avant l'apparition de la Kinect en 2010, présentaient un coût prohibitif non compatible avec la robotique de service, domestique ou personnelle. La quantité de données générées par ces caméras 3D est aussi beaucoup plus importante que celles produites par les télémètres laser 2D. Cela peut être un frein à l'utilisation embarquée d'algorithmes pour les caméras 3D. Nous proposons donc la création d'un modèle de capteur virtuel compatible avec les algorithmes basés sur les télémètres laser 2D, c'est-à-dire représentant les obstacles dans un plan avec un système de coordonnées polaires.

Afin de montrer le bien-fondé de la création d'un tel capteur virtuel et d'évaluer son efficacité dans une situation réelle de robotique, il est nécessaire de construire la chaîne complète de la perception à la commande en passant par la représentation des données. Le but de cette section n'est donc pas de réaliser un état de l'art exhaustif de chacun des maillons de cette chaîne mais de pouvoir se positionner par rapport aux références de la littérature afin de démontrer l'intérêt de l'étalonnage décrit au chapitre précédent pour la construction d'un capteur virtuel consolidant les données du télémètre laser 2D et de la caméra 3D.

La notion de capteur virtuel est utilisée principalement dans deux champs d'applications : la simulation d'un capteur avec la reproduction du comportement du capteur réel incluant la génération de bruit de mesure [Costantini et Susstrunk, 2004] et l'encapsulation d'un ou de plusieurs capteurs en incluant un traitement des données mesurées réelles [Kabadayi *et al.*, 2006]. La seconde situation se rapproche davantage de notre cas puisqu'il s'agit pour nous de réaliser la fusion de deux capteurs télémètriques pour n'en former qu'un seul. Dans ces travaux, nous employons donc le terme de capteur virtuel pour désigner une abstraction d'un producteur de données. A partir des éléments présentés par [Shafer *et al.*, 1986] et [Kabadayi *et al.*, 2006] nous proposons la définition suivante :

Définition Un capteur virtuel est un capteur logiciel qui produit des mesures indirectes issues de traitements sur l'agrégation, la sélection ou la combinaison des données d'autres

capteurs virtuels et/ou réels.

Depuis les années 90, le champ de la fusion de capteurs a fait l'objet d'un grand nombre d'études. Il y a diverses raisons d'utiliser la fusion de capteurs. Différents capteurs peuvent mesurer la même grandeur, alors la fusion va permettre de compléter l'espace de mesure, de rejeter des mauvaises mesures ou d'améliorer la précision de mesure [Diosi et Kleeman, 2004]. Les capteurs peuvent ne pas travailler à des fréquences identiques, auquel cas la fusion permettra de disposer d'une information consolidée cohérente temporellement [Luo *et al.*, 2002]. Ou enfin, chaque capteur peut fournir des informations complémentaires mais hétérogènes qui permettront à l'issue de la fusion de fournir une information de plus haut niveau (inférence hiérarchique présentée dans [Hall et Llinas, 1997]). Le capteur virtuel peut alors être un mode d'encapsulation du traitement de fusion de capteur et de gestion des défaillances [Long *et al.*, 1999].

La plupart des méthodes de fusion a une approche probabiliste ou bayésienne [Moravec, 1988]. En déduisant la probabilité d'un événement à partir d'autres événements déjà évalués, ces approches nécessitent donc de connaître ou d'imposer un degré de confiance dans les données et peuvent éventuellement faire du suivi multi-hypothèses [Martin *et al.*, 2006]. Il est courant d'utiliser cette estimation de l'incertitude pour ensuite simplement pondérer une somme. Lorsqu'on estime que les capteurs sont identiques, cela revient à faire la moyenne des données comme dans [Carballo *et al.*, 2008] avec deux télémètres laser identiques. D'autres travaux utiliseront plutôt un filtre de type Kalman et des dérivés [Diosi et Kleeman, 2004] qui nécessitent de la même façon une pondération par l'incertitude qui permet de calculer une matrice de covariance. Enfin, face aux difficultés de déterminer ce poids, [Goodridge et Luo, 1994] proposent de représenter l'incertitude comme une fonction d'appartenance à un ensemble de la logique floue ce qui permettra de poser un cadre formel de raisonnement de fusion.

Certains de ces travaux de fusion [Moravec, 1988] [Stepan *et al.*, 2005], proposent de réaliser la fusion des données des capteurs dans une grille d'occupation. Ces approches

répartissent les données dans un espace discret dont le concepteur choisit la résolution et aboutit à la création d'une carte issue de la fusion des informations. Un bon nombre d'algorithmes de SLAM base leur fonctionnement sur ce principe de fusion en introduisant la persistance de l'information au cours du temps [Bailey et Durrant-Whyte, 2006] [Thrun, 1998] [Thrun *et al.*, 1999]. Parmi ces travaux, le tinySLAM [Steux et El Hamzaoui, 2010] présente l'avantage d'avoir été conçu pour les télémètres laser à balayage à bas coût. C'est un algorithme particulièrement léger et très efficace à court et à moyen terme utilisé notamment pour la planification de chemin pour un robot semi-autonome [Devaux *et al.*, 2011].

Lorsque l'information de détection d'obstacles est fiable, un évitement d'obstacles verra ses performances augmenter. Parmi les méthodes classiques de la littérature, nous pouvons noter les méthodes par champ de potentiel [Arkin, 1998] qui ont l'inconvénient de présenter des minima locaux pour lesquels il faudra complexifier le modèle en introduisant des champs rotationnels ou tangentiels [Kim et Kim, 2003] ou aléatoires [Goodrich, 2002]. Il existe aussi les méthodes par histogramme d'occupation [Borenstein et Koren, 1991], très simples et très rapides, pour lesquelles il s'agit d'orienter la trajectoire vers un minimum dans l'histogramme des obstacles. Ce minimum correspond à un angle qui détermine la direction du robot. Pour les robots ou les véhicules présentant des contraintes de dynamique importantes, ont été mises au point les méthodes de la fenètre dynamique [Fox *et al.*, 1997] qui cherchent dans l'espace des vitesses (ou des accélérations) l'optimum à chaque instant en interdisant les vitesses présentant un danger. Enfin, la réactivité des méthodes étant une contrainte importante, il existe un autre groupe de méthodes simples basées sur le raisonnement par logique floue et les contrôleurs flous [Hoppenot *et al.*, 1996].

2.3 Modèle d'évitement basé sur un télémètre virtuel laser-Kinect

2.3.1 Modèle du télémètre virtuel

Pour réaliser un tel télémètre virtuel capable de détecter des obstacles complexes, nous supposons que la caméra 3D produit une image de profondeur $\mathcal{D} = \{(u_i, v_i, z_i)\}$ dont la valeur z_i de l'ensemble des n pixels de coordonnées (u_i, v_i) correspond à la distance du point au centre optique associé au repère de la caméra 3D $\mathcal{F}_d$. Le télémètre laser, quant à lui, produit un balayage $\mathcal{L} = \{(\theta_j, d_j)\}$ pour lequel chacun des m mesures à un angle θ_j donne une distance le long du rayon d_j. L'objectif est de produire un télémètre virtuel dont le balayage $\mathcal{V} = \left\{(\theta_j, d'_j)\right\}$ est composé de m mesures donnant pour un angle θ_j une distance le long du rayon d'_j et dont le repère est confondu avec celui du télémètre laser : $\mathcal{F}_v = \mathcal{F}_l$.

L'algorithme proposé est décomposé en 5 étapes :

— La génération d'un nuage de points $\mathbf{P}_d$ à partir de l'image $\mathcal{D}$:

En supposant que la caméra 3D est correctement modélisée par le sténopé, on a :

$$\mathbf{P}_d = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}_{\forall i \in 1..n} = \begin{pmatrix} \frac{(u_i - u_0) z_i}{f_x} \\ \frac{(v_i - v_0) z_i}{f_y} \\ z_i \end{pmatrix}_{\forall i \in 1..n}$$

où u_0, v_0, f_x et f_y sont les paramètres intrinsèques de la caméra 3D selon le modèle du sténopé. Dans ces travaux, ces paramètres sont supposés connus mais il est possible de les obtenir par étalonnage basé mire 3D [Devaux *et al.*, 2013].

— La transformation de ${}^{\mathcal{F}_d}\mathbf{P}_d$ en ${}^{\mathcal{F}_l}\mathbf{P}_d$ depuis le repère caméra 3D $\mathcal{F}_d$ vers le repère laser $\mathcal{F}_l$:

$$\forall i \in 1..n, \quad \left({}^{\mathcal{F}_l}\mathbf{P}_d\right)_i = \mathbf{R}.\left({}^{\mathcal{F}_d}\mathbf{P}_d\right)_i + \mathbf{t}$$

où $\mathbf{R}$ et $\mathbf{t}$ sont respectivement la matrice de rotation et le vecteur de translation, paramètres extrinsèques issus de l'étalonnage laser-caméra 3D (Cf. chapitre 1).

— La suppression dans ${}^{\mathcal{F}_l}\mathbf{P}_d$ du sol et du plafond :

Elle s'effectue simplement par seuillage en utilisant les dimensions connues du robot. En effet, si l'on suppose que le plan laser est parallèle au plan d'évolution du robot mobile et que la forme de celui-ci peut être approximée par une enveloppe cylindrique, il faut et il suffit de conserver les données de ${}^{\mathcal{F}_l}\mathbf{P}_d$ si et seulement si la composante y_i est telle que $\rho_{min} < y_i < \rho_{max}$. Avec ρ_{min} et ρ_{max} respectivement la distance du plan laser au sol et la distance du plan laser au sommet du robot et $i = 1..p$ où p correspond au nombre de points respectant la contrainte.

— La projection sur le plan laser (en coordonnées polaires) $\mathbf{P'}_l = ((\theta''_i, d''_i))_{\forall i=1..p}$ de ${}^{\mathcal{F}_l}\mathbf{P}_d$:

Si l'on suppose que le plan laser est parallèle au plan d'évolution du robot mobile alors :

$$\forall i \in 1..p, \quad \left(\mathbf{P'}_l\right)_i = \begin{pmatrix} \theta''_i = \begin{cases} arctan(\frac{y_i}{x_i}) & si\ x_i > 0 \\ arctan(\frac{y_i}{x_i}) + signe(y_i).\pi & si\ x_i < 0 \\ signe(y_i).\frac{\pi}{2} & si\ x_i = 0 \end{cases} \\ d''_i = \sqrt{{x_i}^2 + {y_i}^2} \end{pmatrix}$$

— La sélection par interpolation, pour chaque θ_j, d'une mesure d'_j de risque le plus fort :

Tout d'abord, nous regroupons dans N classes les points du nuage de points $\mathbf{P'}_l$ en fonction de leur coordonnée d'angle afin que chaque classe représente un pas de balayage d'un télémètre. Pour chaque classe, nous cherchons le point dont la distance est la plus faible que nous appelons le représentant de la classe. Nous effectuons le même traitement sur les données du télémètre laser que sur les données de la caméra 3D pour obtenir respectivement les ensembles de classes $\{\mathcal{C}_l\}$ et $\{\mathcal{C}_d\}$.

Ensuite, pour chaque mesure du télémètre virtuel à un angle θ, nous interpolons $\{\mathcal{C}_l\}$ et

$\{\mathcal{C}_d\}$ pour obtenir deux distances d_j et d''_j. La mesure virtuelle sera alors $d'_j = min(d_j, d''_j)$. La fonction *minimum* a été choisie pour minimiser le risque dans le cas d'un évitement d'obstacle. Elle correspond dans les fusions floues à l'opérateur le plus classique pour la composition des variables d'entrées. D'autres fonctions sont envisageables selon l'application envisagée, et notamment des fonctions de fusion plus complexes tenant compte de l'asynchronisme des capteurs. Néanmoins dans le cadre de la robotique mobile en milieu intérieur habité (l'humain peut être présent aux côtés du robot) et destiné à des personnes en perte d'autonomie, nous limitons volontairement, pour des questions d'acceptabilité la vitesse du robot et, par conséquent, dans le pire des cas, la différence de temps d'acquisition des données ne correspond qu'à un déplacement inférieur à 5 cm pour un télémètre laser à 10 Hz et moins de 2 cm pour les caméras 3D à 33 Hz pour un robot évoluant à une vitesse maximale de 0.5 m/s. Pour des robots plus rapides, il serait alors possible d'utiliser la date de dernière mesure pour estimer une incertitude pour l'introduire dans des modèles de fusion plus complexes.

2.3.2 Modèle de persistance des informations

Comme nous l'avons déjà évoqué plusieurs fois, les caméras 3D à bas coût ont un champ de vision horizontal assez restreint alors que le télémètre laser présente en général un champ de vision très large. L'usage du télémètre virtuel construit ci-dessus montre que les obstacles correctement détectés par la caméra 3D sortent rapidement du champ de vision d'un robot mobile lors de sa navigation en milieu intérieur.

La primo-détection par le télémètre virtuel donne néanmoins une information forte qu'il est nécessaire de faire vivre dans le temps pour une application d'évitement d'obstacle. Ainsi, supposons avoir détecté un obstacle O à un moment t, il est possible d'estimer la position relative de O à l'instant $t+1$ en ayant connaissance du mouvement du robot entre t et $t+1$. La construction d'une carte $\mathcal{M}$ de l'ensemble des obstacles observés au cours du temps est ainsi réalisable.

Si l'odométrie du robot donne des résultats utilisables directement pour réaliser un

tel modèle de persistance, il n'est pas possible de s'y fier pour un évitement d'obstacle. La littérature est très fournie en algorithmes nécessitant une précision meilleure que celle offerte par l'odométrie. Les techniques de SLAM, notamment, permettent d'établir une carte précise de l'environnement tout en localisant le robot au sein de cette carte. Des techniques de SLAM ont été développées pour de nombreux capteurs dont les caméras et les télémètres laser.

En général, l'objectif des algorithmes de SLAM est de cartographier et de localiser le robot à long terme. Ces méthodes nécessitent donc beaucoup de mémoire et une puissance de calcul conséquente étant données les dimensions du problème sous-jacent. Plus l'environnement est grand, plus les techniques de SLAM sont coûteuses. Puisqu'il n'est pas envisageable de ralentir l'évitement d'obstacles, nous proposons dans cette section, d'utiliser les mécanismes de SLAM sur une carte locale de l'environnement. Le concept de ces cartes locales provient de l'utilisation des algorithmes hiérarchiques de SLAM dans des environnements de grande taille [Estrada *et al.*, 2005]. Le principal avantage de ces cartes locales réside dans le fait de pouvoir contenir le temps d'exécution quel que soit le nombre de cartes locales produites.

Dans notre cas, il s'agit ainsi de concevoir une carte de l'environnement relative au robot et de n'y conserver les informations que si elles satisfont certains critères. Nous avons identifié deux critères principaux. Tout d'abord, l'obsolescence des données qui se base sur la date de dernière détection d'un obstacle permet de s'assurer qu'un obstacle observé il y a longtemps à une certaine localisation n'est pas pris en compte. Ce critère règle deux problèmes majeurs qui sont les obstacles dynamiques ou mobiles et l'éventuelle mauvaise qualité de la localisation du robot. Le deuxième critère concerne l'éloignement de l'obstacle. En effet, considérant les vitesses acceptables d'un robot d'intérieur, un obstacle situé au delà d'une distance max sera *oublié* volontairement. Ce dernier critère permet de conserver bornée la taille de la carte et donc le temps de calcul du SLAM et par extension il permet de garantir une fréquence minimale de fonctionnement de l'évitement d'obstacles.

A tout balayage $\mathcal{V}$ du télémètre virtuel est associé un état $\mathbf{x}_t = (x, y, \theta)$ du robot mobile

à un instant t. On peut estimer cet état en minimisant la différence entre la connaissance contenue dans la carte $\mathcal{M}_{t-1}$ à l'instant $t-1$ et celle contenue dans le balayage $\mathcal{V}_t$:

$$\mathbf{x}_t = \arg\min_{\mathbf{x}} \{\|h(\mathcal{M}_{t-1}, \mathbf{x}) - m(\mathcal{V}_t)\|_{\mathcal{M}}\}$$

où m et h sont respectivement les fonctions de construction d'une carte à partir d'un balayage (il s'agit d'une grille d'occupation de valeur 0 pour un espace libre et de valeur 1 pour un obstacle) et de transformation géométrique de la carte à partir de 3 paramètres : une rotation et deux translations. La norme utilisée $\|.\|_{\mathcal{M}}$ consiste simplement à compter le nombre de valeurs non nulles dans une carte.

Une telle minimisation peut être réalisée par une méthode de Monte-Carlo par chaînes de Markov comme dans tinySLAM [Steux et El Hamzaoui, 2010] ou plus simplement par un échantillonnage exhaustif dans un espace discret puisque l'espace de recherche est assez restreint.

Il s'agit ensuite de réaliser la fusion de l'observation $\mathcal{V}_t$ et de la carte $\mathcal{M}_{t-1}$ sachant le déplacement $\tilde{\mathbf{x}}_t$ du robot. Cette fusion peut s'apparenter à un OU logique :

$$\mathcal{M}_t = h(\mathcal{M}_{t-1}, \tilde{\mathbf{x}}_t) \; OU \; m(\mathcal{V}_t)$$

La carte ainsi produite peut être utilisée comme une source de données pour une planification locale ou comme une base pour recréer un télémètre virtuel qui possèdera alors un mécanisme de persistance sensorielle et aura alors un champ de vision à 360 °. Nous réalisons ce nouveau télémètre virtuel en transformant la carte en une liste d'obstacles en coordonnées polaires dont on ne conservera que les obstacles dont la distance est minimale à l'intérieur d'un secteur angulaire donné.

2.3.3 Modèle d'évitement d'obstacles

Une fois que nous disposons d'une source de données fiable, n'importe quel évitement d'obstacles sera nécessairement plus performant. Nous souhaitons montrer que la caméra 3D permet d'améliorer la sécurité de fonctionnement du robot, l'évitement d'obstacles choisi sera donc un simple évitement réactif. L'objectif n'étant pas de proposer un évitement d'obstacles plus performant mais de montrer l'influence de la qualité de détection, nous choisissons d'utiliser un modèle existant dans la littérature [Hoppenot *et al.*, 1996], modèle basé sur la logique floue, rapide (donc adapté pour un évitement réactif) et ayant démontré toute son efficacité sur différentes architectures de robots mobiles en milieu intérieur [Delarue *et al.*, 2010] [Nadrag, 2011] [Mantel *et al.*, 2012].

Supposons un robot mobile à deux roues motrices différentielles et symétriques, ce robot peut être résumé en un robot non-holonome de type unicycle.

Le modèle cinématique du robot peut être décrit par l'équation suivante :

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} cos(\theta) & 0 \\ sin(\theta) & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} v \\ \omega \end{pmatrix}$$

avec x, y et θ la position et orientation du centre de rotation du robot évoluant sur le plan xy et $\dot{x}$, $\dot{y}$ et $\dot{\theta}$, leurs dérivées. Le couple (v, ω) est composé des vitesses linéaires et angulaires qui correspondent aux entrées du système, le robot étant commandé en vitesse.

Supposons que l'environnement autour du robot peut être divisé en secteurs gauche, central et droit. Chacun de ces secteurs est virtuellement équipé d'un télémètre dont la mesure donne la distance du plus proche obstacle dans le secteur concerné. On a donc R, L et F les distances mesurées dans les trois secteurs et $R_n = \frac{R}{L+R}$, $L_n = \frac{L}{L+R}$ et $F_n = \frac{F}{Inf}$ les distances normalisées correspondantes (avec Inf, la distance maximale de détection).

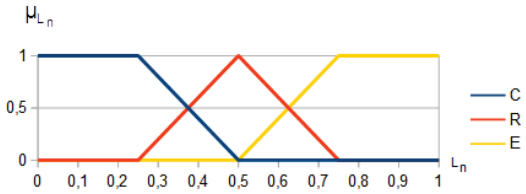
En utilisant la logique floue, chaque univers de discours est divisé en trois sous-ensembles. Pour les entrées capteurs L_n et R_n, nous labellisons les ensembles C, R, E pour respective-

ment les obstacles au Contact, les obstacles Rapprochés et les obstacles Eloignés. Concernant la vitesse angulaire ω nous disposons des labels N, Z, P pour respectivement une vitesse Négative, nulle (Zéro) ou Positive.

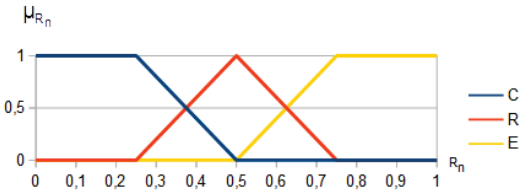
Les règles de contrôle déduites de l'expérience d'un opérateur humain sont résumées dans le tableau 2.1 et les fonctions d'appartenance sont représentées dans la figure 2.1 (la largeur des intervalles n'est donnée qu'à titre d'exemple). Par exemple, si l'obstacle représenté par R_n est *au Contact* et l'obstacle représenté par L_n est *Eloigné* alors la vitesse angulaire ω est *Positive*. Une vitesse angulaire positive correspond ainsi à un virage vers la gauche permettant d'augmenter R_n.

$R_n \backslash L_n$	C	R	E
C	Z	P	P
R	N	Z	P
E	N	N	Z

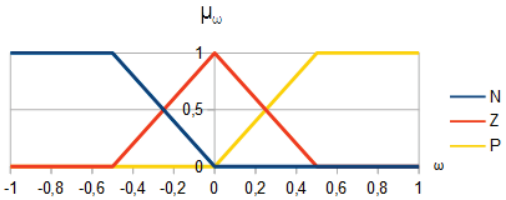
TABLE 2.1 – Règles concernant la vitesse angulaire



(a) Fonctions d'appartenance pour L_n



(b) Fonctions d'appartenance pour R_n



(c) Fonctions d'appartenance pour ω

FIGURE 2.1 – Fonctions d'appartenances du système flou

La méthode de passage de la représentation floue de la vitesse à une représentation métrique repose sur le centre de gravité [Lee, 1990] :

$$\omega = \frac{\sum_i \omega_i . \mu_{\omega_i}}{\sum_i \mu_{\omega_i}}$$

Le comportement de cet évitement flou simple commandant uniquement en vitesse angulaire correspond à l'attraction par l'équilibre entre les mesures du secteur gauche et du secteur droit ou, autrement dit, au centrage au milieu de l'espace libre. L'implémentation d'un tel contrôleur d'évitement d'obstacles est très simple et un tel programme fonctionne à des fréquences très élevées même sur des architectures embarquées les plus simples comme les microcontrôleurs.

L'ensemble de la chaîne de fusion-mémorisation-évitement fonctionne sur une architecture de type Core2 Duo à une fréquence de 40 Hz sans travail d'optimisation particulier et avec une implémentation purement séquentielle. 60% du temps de calcul (≈15 ms) sont dépensés par le modèle de persistance des informations et plus particulièrement par l'étape de minimisation. Il est néanmoins aisé de régler cette durée de calcul en manipulant la taille de la fenètre de recherche (et, par conséquent, en diminuant éventuellement la vitesse de déplacement du robot).

2.4 Evaluation

2.4.1 Objectifs et métriques

Les objectifs de l'évaluation sont de montrer l'apport de la caméra 3D pour la détection d'obstacles complexes en milieu intérieur, la complémentarité de la caméra 3D et du télémètre laser ainsi que la réactivité du système complet afin de réaliser un évitement d'obstacles robuste (aux obstacles complexes) et fiable.

Pour évaluer la bonne réalisation de ces trois objectifs, nous définissons les métriques suivantes pour le champ complet du télémètre virtuel ainsi que pour le champ de vision restreint à la portion perçue à la fois par le télémètre laser et par la caméra 3D :

— Le nombre de vrais positifs pour la caméra 3D

— Le nombre de vrais positifs pour le télémètre laser

— Le nombre de faux positifs pour la caméra 3D

— Le nombre de faux positifs pour le télémètre laser

— Le nombre de faux négatifs pour la caméra 3D

— Le nombre de faux négatifs pour le télémètre laser

— La sensibilité du télémètre virtuel, du télémètre laser et de la caméra 3D

L'évaluation de la chaîne de traitement complète repose sur l'information binaire correspondant au fait de devoir manuellement arrêter (ou non) le robot avant qu'une collision n'advienne.

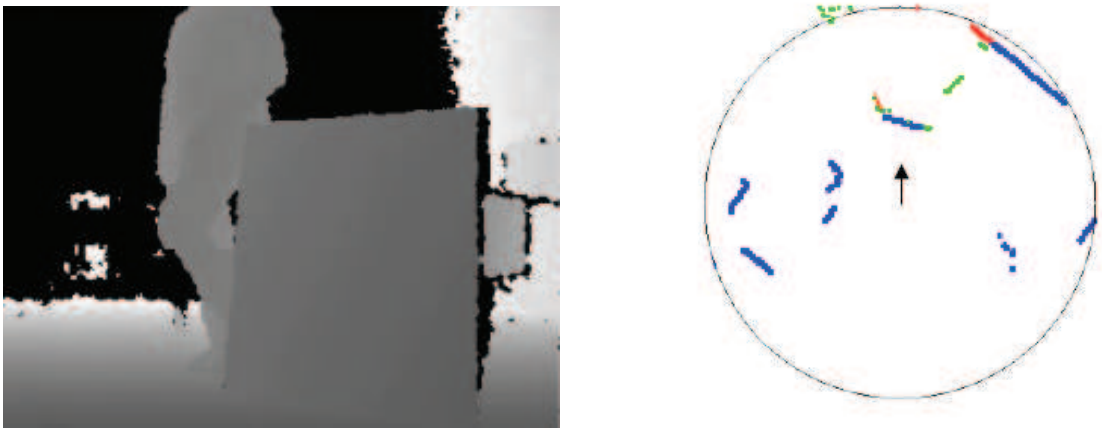
2.4.2 Protocole expérimental

Nous évaluerons la détection à la fois en statique et en dynamique. Les scènes statiques ont été choisies pour montrer deux situations où l'une est *a priori* plutôt favorable au laser et l'autre est *a priori* plutôt avantageuse pour la caméra 3D. La détection sur les scènes statiques fera l'objet d'une étude quantitative avec les métriques décrites ci-dessus. Le robot sera placé dans un environnement de bureau usuel (que nous appelons environnement naturel) constitué de murs, de meubles (tables, chaises) et de panneaux et dans lequel des personnes se déplacent.

Dans l'évaluation dynamique, nous placerons le robot dans ce même environnement avec deux situations : le robot statique et la scène dynamique (une personne se déplace dans l'environnement) ou le robot mobile dans une scène statique présentant notamment une table face à la position de départ du robot. Dans cette seconde situation, l'opérateur donnera au robot l'ordre d'avancer avec une vitesse linéaire constante et une vitesse angulaire nulle. Un deuxième opérateur arrêtera manuellement le robot immédiatement avant que celui-ci ne rentre en collision avec un obstacle.

2.4.3 Résultats

Lors de la navigation d'un robot mobile, les améliorations des données du télémètre laser par la caméra 3D ont été constamment observées. Les figures 2.2 et 2.3 représentent les images de profondeur acquises par la caméra 3D (à gauche) aux côtés des représentations graphiques des données télémétriques (à droite). Sur ces graphiques, le cercle noir représente la portée maximale du télémètre laser utilisé (4 m), alors que la flèche noire annote la position et l'orientation du robot. Les points de couleurs représentent les obstacles détectés par le télémètre laser ou le télémètre virtuel basé sur le télémètre laser et la caméra 3D. Les points bleus représentent les obstacles détectés par le télémètre laser pour lesquels le télémètre virtuel n'apporte pas de correction (soit l'obstacle est hors champ de la caméra 3D, soit la caméra 3D le localise précisément au même endroit que le télémètre laser). Les points verts représentent les obstacles détectés par le télémètre virtuel et non détectés par le télémètre laser alors que les points rouges correspondent à la détection du télémètre laser lorsque le télémètre virtuel apporte une correction dans la même direction.



(a) Kinect depth image

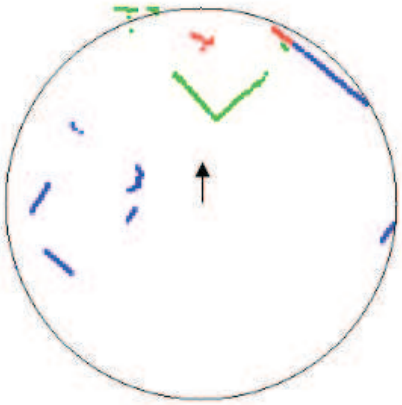
(b) Kinect-Laser fusion

FIGURE 2.2 – Obstacle vertical vu par la caméra 3D (gauche) et le télémètre virtuel (droite)

La figure 2.2 représente une scène simple composée principalement d'un plan vertical et d'une personne debout le genou en avant alors que la figure 2.3 représente une scène



(a) Kinect depth image



(b) Kinect-Laser fusion

FIGURE 2.3 – Table vue par la caméra 3D (gauche) et le télémètre virtuel (droite)

composée d'obtacles complexes (tables, chaises, ...). Sur les deux figures, nous pouvons observer que la totalité des points sur les zones à gauche et à droite de la flèche matérialisant le robot sont bleus. En effet, le télémètre laser ayant un champ de vision bien plus large que le champ de vision horizontal de la caméra 3D, le capteur 3D ne peut influer sur ces obstacles latéraux uniquement détectés par le télémètre laser.

La figure 2.2 représente une scène composée principalement d'un plan vertical touchant le sol et donc détectable à la fois par le télémètre laser et la caméra 3D. Sur ce cas idéal pour le télémètre laser, on observe que la plupart des mesures effectuées par la caméra 3D est localisée au même endroit par le télémètre laser (points bleus face à la flèche). Les seules détections produites par la caméra 3D proviennent d'objets complexes plus éloignés dans l'environnement (un bloc de tiroirs suspendus au coin du bureau à droite et une desserte roulante à gauche) ou de la personne dont le corps n'est pas exactement vertical et dont la mesure du laser au niveau des mollets n'est pas représentative de la projection sur le plan horizontal de l'enveloppe totale du corps. Ainsi, cette scène montre tout d'abord que les mesures du laser ne sont pas dégradées par la création du télémètre virtuel. Elle montre ensuite que l'étalonnage réalisé (présenté dans le chapitre 1) permet un alignement correct des capteurs laser et caméra 3D afin de créer un télémètre virtuel produisant des

données cohérentes. Enfin, elle suggère les meilleures performances de la caméra 3D pour la détection du corps de l'humain et des obstacles suspendus (comme le bloc de tiroirs au coin du bureau).

Les points verts montrent que le télémètre virtuel ajoute effectivement le corps de l'humain et le bloc de tiroirs dans la carte. La desserte est aussi représentée par le nuage de points verts à l'extrémité du cercle noir mais en raison de sa distance supérieure à 4 m (distance limite pour la caméra 3D à bas coût : nous rappelons que la plage utile est plutôt située entre 80 cm et 3 m), la localisation et la forme de l'objet sont imprécises. En revanche, le télémètre laser n'étant pas capable de détecter le bloc-tiroirs à droite suspendu au-dessus du plan de détection, la mesure effectuée (segment rouge colinéaire au segment bleu) correspond à la distance du télémètre laser jusqu'au mur de la salle, partiellement visible à l'extrémité droite de l'arrière plan de l'image de profondeur.

Sur la figure 2.3 on observe plus en détail la situation où l'obstacle n'est pas situé au niveau du plan laser et donc n'est pas détectable par le télémètre laser. L'image de profondeur montre la présence de l'arête du plateau de la table en gris foncé un peu au-dessus du centre de l'image. Cette arête est donc détectable par la caméra 3D. L'observation des points verts sur le graphique confirme que la caméra 3D a effectivement détecté le plateau de la table. En effet, les points verts matérialisent deux segments orthogonaux dans les données produites par le télémètre virtuel. Manifestement, un évitement d'obstacles basé sur un télémètre laser seul aurait conduit à une collision inévitable. Cette collision pourrait être évitée en utilisant les données issues du télémètre virtuel.

Le tableau 2.2 résume la contribution de chaque capteur en terme de faux positifs et de vrais positifs sur les deux scènes ci-dessus. Nous présentons le nombre de points mesurés par chaque capteur dans la totalité du champ de vision du télémètre virtuel (champ global) ou seulement dans le secteur frontal (couvert à la fois par la caméra 3D et par le télémètre laser). On constate que le taux de vrais positifs est globalement avantageux pour le laser (69%) par rapport à la caméra 3D (36%). Cette situation s'inverse lorsque l'on considère seulement le secteur couvert par les deux capteurs simultanément. On a alors en moyenne

			Obstacle vertical				Table			
			Champ glob.		Secteur front.		Champ glob.		Secteur front.	
			Nb	%	Nb	%	Nb	%	Nb	%
Positif	Vrai	Laser	71	78.9%	9	32.1%	58	59.2%	0	0%
		Caméra 3D	28	31.1%	28	100%	40	40.8%	40	100%
	Faux	Laser	9	10%	9	32.1%	11	11.2%	11	27.5%
		Caméra 3D	0	0%	0	0%	0	0%	0	0%
	Vrais positifs		90	100%	28	100%	98	100%	40	100%
Négatif	Vrai	Laser	NA				NA			
		Caméra 3D	NA				NA			
	Faux	Laser	19	3.3%	19	17.9%	40	6.9%	40	42.6%
		Caméra 3D	0	0%	0	0%	0	0%	0	0%
	Vrais négatifs		582	100%	106	100%	574	100%	94	100%

TABLE 2.2 – Mesures du télémètre laser et de la caméra 3D. Les vrais positifs et vrais négatifs sont déterminés à partir du télémètre virtuel et donc significatifs uniquement dans le secteur frontal.

16% pour le laser contre 100% pour la caméra 3D. Cela s'explique facilement par le faible champ de vision de la caméra 3D qui ne réalise aucune mesure sur plus de 80% du champ de balayage du télémètre virtuel. Cette mesure de vrais positifs montre bien l'utilité de chacun des capteurs et de l'agrégation réalisée.

La mesure des taux de faux négatifs (c'est-à-dire d'obstacles non détectés) donne une information cruciale pour une application d'évitement d'obstacles. La sensibilité S est une mesure reflétant la probabilité qu'un obstacle présent soit détecté. Elle est calculée à partir du nombre de vrais positifs V_P et du nombre de faux négatifs F_N.

$$S = \frac{V_P}{V_P + F_N}$$

En moyenne sur les deux scènes, la sensibilité du laser sur son champ de vision est de 0.16 alors que celle de la caméra 3D sur son champ de vision est de 1. Sur l'ensemble du champ de vision du télémètre virtuel, on peut estimer la sensilité à 0.33, soit une augmentation de 200% par rapport au télémètre laser seul alors même que la caméra 3D ne participe qu'à 20% du champ de vision du télémètre virtuel. D'autres exemples de scènes sont présentés en annexe B. L'évaluation dynamique d'un tel évitement d'obstacles est présentée ci-après.

La figure 2.5 montre une scène statique dans laquelle une personne se déplace. La première ligne représente l'image capturée par une caméra conventionnelle alors que la dernière ligne montre l'image capturée par la caméra 3D. La deuxième ligne matérialise les données du télémètre virtuel (bleu : mesures cohérentes entre le télémètre laser et la caméra 3D et vert : mesures de la caméra 3D présentant un danger supérieur aux mesures données par le télémètre laser) et du télémètre laser seul (rouge : mesures données par le télémètre laser lorsque la caméra 3D propose des mesures plus proches du robot). Ces dernières mesures du laser ne sont pas prises en compte dans le télémètre virtuel car elles traduisent un danger plus faible que les mesures de la caméra 3D. Chaque colonne correspond à un instant donné variant de t à $t + 3$. Nous montrons et annotons dans la figure 2.4 les éléments principaux de la scène avec notamment la personne se déplaçant (D) ainsi que sa trajectoire, le panneau utilisé texturé avec un damier (C), le bureau (B) ainsi que le mur du fond (A).

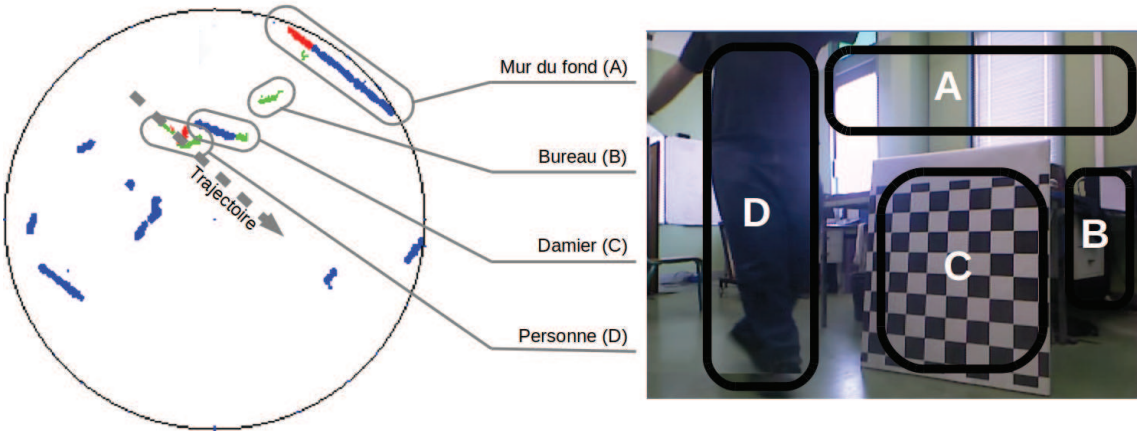


FIGURE 2.4 – Annotation de la scène statique avec une personne se déplaçant

La première colonne de la figure 2.5 montre la scène statique alors que la personne mobile n'est pas présente dans le champ de vision de la caméra 3D. La personne est détectée par le télémètre laser comme un groupe de deux petits obstacles. Il s'agit en fait uniquement des jambes de la personne. Le panneau apparaît clairement sur le relevé laser alors que le bureau est seulement détecté par la caméra 3D. A la place de détecter le bureau, le télémètre laser seul ne tenait compte que du mur du fond, parfaitement détecté.

Sur la deuxième colonne, la personne apparaît dans le champ de vision de la caméra 3D. La personne est toujours matérialisée par deux groupes de points. Le premier est parfaitement identifié par le laser seul alors que pour le second, le laser seul sous estime la surface au sol de l'enveloppe de la personne dont le bras n'est pas situé exactement à la verticale de la jambe. Sur la troisième colonne, les deux bras de la personne sont écartés du corps. Si le télémétre laser détecte toujours ses jambes, l'enveloppe de la personne n'est plus correctement détectée par le télémètre laser seul. Au contraire, comme le montrent les deux segments verts, le télémètre virtuel utilisant la caméra 3D est capable de détecter ces bras écartés et de retranscrire leur présence dans le plan de balayage du télémètre virtuel. Enfin, sur la dernière colonne, la personne occulte le panneau et, de la même façon la caméra 3D permet de détecter correctement l'ensemble du corps de la personne alors que le télémètre laser ne présentait toujours que les deux jambes. L'écart mesuré entre la jambe détectée par le télémètre laser et l'extrémité de l'enveloppe mesurée par la caméra 3D est d'environ 40 cm.

Cette figure 2.5 montre bien que la différence de détection entre le télémètre laser seul et le télémètre virtuel n'est pas négligeable. Ces mesures effectuées dans le cadre d'une personne se déplaçant devant les capteurs font ressortir la grande différence de perception entre le capteur 2D, le télémètre laser, et le capteur 3D, la caméra de profondeur.

Sur les figures 2.6 et 2.7, nous représentons l'évolution au cours du temps de la vue extérieure de la scène pour montrer la trajectoire du robot ainsi que de la perception par le robot de l'environnement à l'aide du télémètre laser seul pour la figure 2.6 et avec le télémètre virtuel utilisant le télémètre laser et la caméra 3D pour la figure 2.7. Le robot est positionné face à une table à une distance de 2m environ. Il lui est donné la consigne en vitesse suivante : $(v = 0.5m/s, \omega = 0rad/s)$ où v et ω sont respectivement les vitesses linéaire et angulaire appliquées par le robot. L'évitement d'obstacles se base uniquement sur les informations issues du télémètre laser dans le cas de la figure 2.6 alors que l'évitement d'obstacles reçoit les données du télémètre virtuel dans le cas de la figure 2.7.

Sur la figure 2.6, nous avons matérialisé la position réelle de la table par le segment

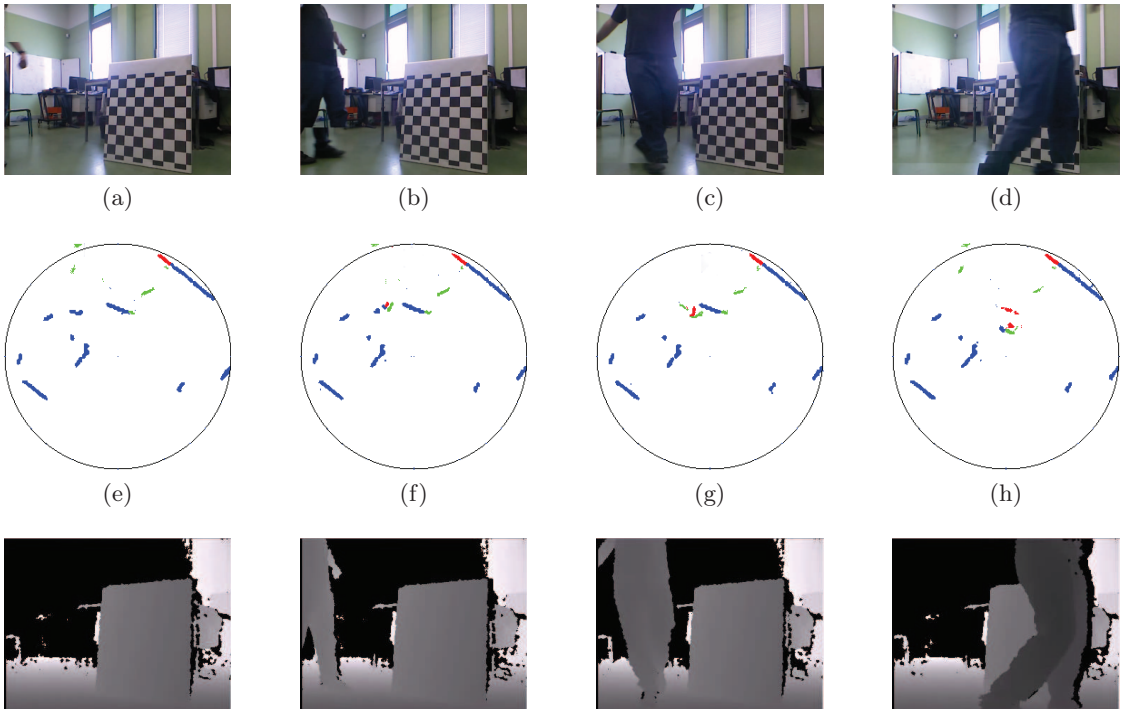


FIGURE 2.5 – Comportement du télémètre virtuel en présence d'un obstacle mobile complexe

gris. Précisons que ce segment n'est pas détecté par le télémètre laser seul et est seulement présent afin de rendre compte de la vérité terrain et de la position réelle de l'obstacle. Nous observons d'ailleurs qu'au cours du temps le segment gris se rapproche de plus en plus du centre du cercle, c'est-à-dire que la table se rapproche de plus en plus du robot. On observe par ailleurs que le robot ne détecte jamais la table et qu'il ne détectera les pieds de la table que peu de temps avant la collision. Le télémètre laser ne détectant pas l'obstacle, la trajectoire du robot suit la consigne envoyée et mène le robot à la collision avec la table (collision évitée par un arrêt manuel du robot).

Sur la figure 2.7, nous retrouvons en plus l'image de profondeur captée par la caméra 3D. Lorsqu'on observe la première image de profondeur (figure 2.7i), on voit nettement la table se détacher. En particulier, le plateau de la table est bien visible alors que les pieds semblent moins bien détectés. Il faut préciser que les pieds de la table sont en acier

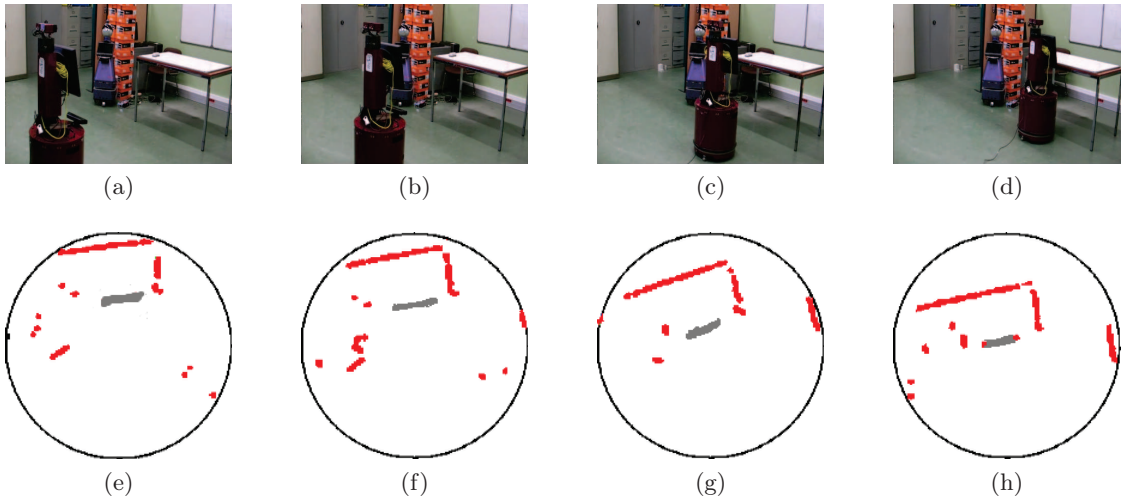


FIGURE 2.6 – Scénario d'évitement utilisant seulement le télémètre laser. La première ligne correspond à une vue extérieure de la scène pour mieux appréhender la trajectoire du robot.

avec une surface en finition *poli-miroir*. De la même façon avec le télémètre laser seul, la détection des pieds de la table semble être difficile du fait de cette surface particulièrement réfléchissante.

Alors que les mêmes consignes de vitesses que pour la figure 2.6 sont envoyées au robot ($v = 0.5m/s, \omega = 0rad/s$), on observe que le robot n'avance pas en ligne droite. L'observation directe de la trajectoire du robot nous permet d'affirmer que l'évitement d'obstacles a modifié les consignes données au robot. A partir de la troisième étape, la caméra 3D ne détecte plus la table préalablement identifiée. Néanmoins la persistance des informations telle que présentée dans la section 2.3.2 (non représentée dans les figures précédentes) permet à l'évitement de continuer sa manoeuvre de contournement puisqu'on observe que la rotation entre 2.7f et 2.7g continue entre 2.7g et 2.7h.

L'observation des images 2.7a, 2.7b, 2.7c et 2.7d permet de se représenter la trajectoire suivie par le robot mobile dont la consigne était d'avancer tout droit face à la table à vitesse constante. Dans les faits, le robot tourne vers la gauche à mesure qu'il se rapproche de la table jusqu'à adopter une trajectoire d'évitement de la table. Lors de la réalisation de ce scénario, il n'a pas été nécessaire d'ajuster la consigne de vitesse ni d'arrêter manuellement

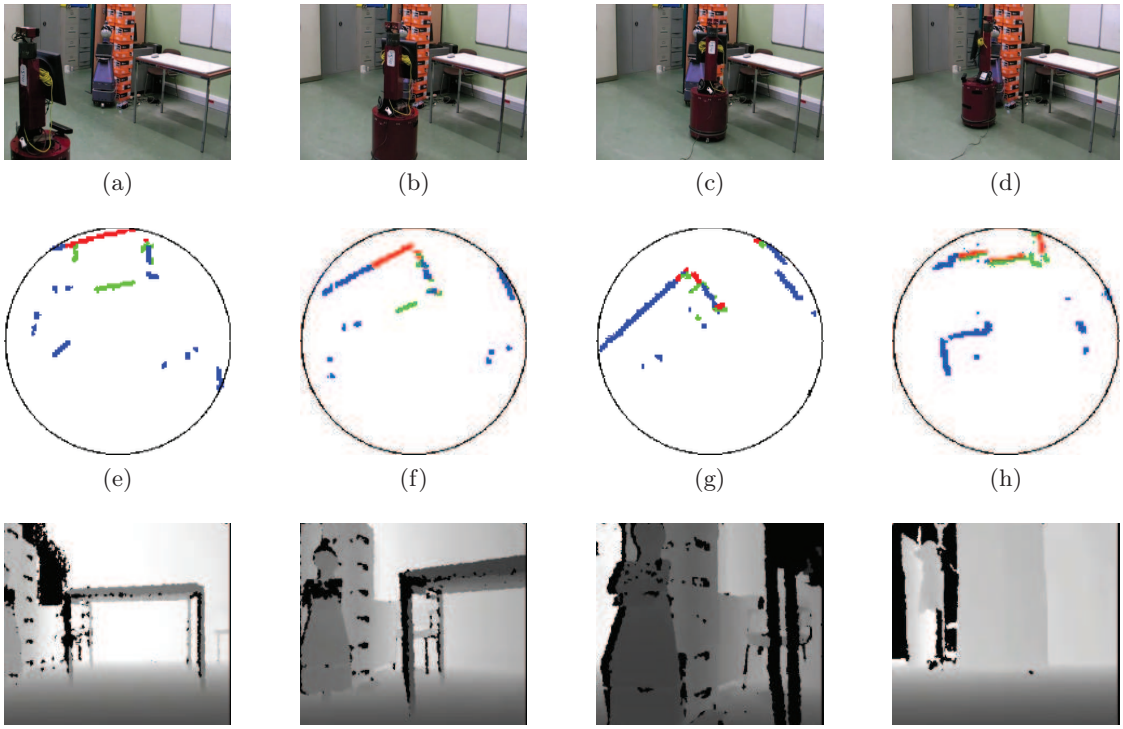


FIGURE 2.7 – Scénario d'évitement utilisant le télémètre virtuel. La première ligne correspond à une vue extérieure de la scène pour mieux appréhender la trajectoire du robot.

le robot, la chaîne de détection et d'évitement se chargeant du danger.

Comme nous avons pu le constater entre la figure 2.6 et 2.7, un même algorithme d'évitement d'obstacles ne réagit pas de la même façon selon les données en entrée. En effet, lorsque le télémètre laser est utilisé seul comme source de l'évitement d'obstacles, le robot rentre en collision avec les obstacles de forme complexe (bien que très courants dans un environnement naturel : les tables, les chaises, les étagères suspendues, ...). Cette collision n'est pas incriminable à l'évitement d'obstacles mais à une perception non fiable. Au contraire, lorsque le télémètre laser est assisté de la caméra 3D pour former le télémètre virtuel avec persistance de l'information, les capacités de perception du robot sont nettement améliorées. Cette meilleure fiabilité de détection intervient directement dans les résultats d'évitement d'obstacles puisque le robot est maintenant capable d'éviter tous les obstacles détectables par la caméra 3D en environ 25 ms (5 ms pour la construction du

télémètre virtuel, 15 ms pour le recalage et la prise en compte des obstacles précédemment observés et 5 ms pour le calcul d'évitement). De fait, le système est capable de d'éviter même les obstacles dynamiques présents en milieu intérieur à environ de 5 cm du robot et dont la vitesse relative maximale peut aller jusqu'à 1.5m/s (somme de la vitesse maximale usuelle du robot et de la vitesse habituelle de marche d'une personne en milieu intérieur).

2.5 Conclusion

L'objectif de ce chapitre était de montrer l'apport de la caméra 3D à bas coût pour compléter les données d'un télémètre laser à balayage dans le cadre de la navigation autonome d'un robot mobile.

La proposition de construction d'un télémètre virtuel répond à cet objectif et offre l'avantage de pouvoir remplacer de façon totalement transparente un télémètre à balayage (capteur fréquemment utilisé en robotique mobile) tout en réduisant la quantité de données à manipuler (la caméra 3D offre un nuage de points dense) et en permettant une fusion des données des capteurs. Nous avons montré que l'amélioration de la perception est très importante dans la zone couverte par les deux capteurs mais avons dû faire face au problème de la sortie des obstacles du champ de vision de la caméra 3D. Nous avons alors proposé un modèle de persistance des données basé sur un SLAM local et relatif qui permet de conserver l'information de la présence d'un obstacle en dehors du champ de vision de la caméra 3D même si le télémètre laser n'est pas capable de le percevoir. Enfin nous avons appliqué cette chaîne de traitement au déplacement autonome d'un robot mobile avec un évitement d'obstacles traditionnel.

Hors persistance des informations, nous avons montré que la caméra 3D, bien que ne participant qu'à 20% du champ de vision du télémètre virtuel, permet de doubler sa sensibilité par rapport au télémètre laser seul sans pour autant augmenter le nombre de faux positifs ou de faux négatifs. Enfin, par différentes expérimentations en situation réelle avec un robot mobile doté d'un évitement d'obstacles conventionnel, nous avons montré

que l'utilisation du télémètre virtuel permet d'éviter l'obstacle là où le télémètre laser seul ne permettait même pas de le percevoir.

Ce chapitre a permis de répondre au besoin d'une perception robuste de l'environnement pour l'évitement d'obstacles lorsque le système est équipé d'une caméra 3D et d'un télémètre laser à balayage. Dans le cadre de la robotique de service (et donc de la robotique mobile à bas coût), de nombreux constructeurs font le choix de mettre de côté les télémètres laser à balayage en raison de leur coût d'achat et de leur faible durée de vie (présence de pièces mécaniques de précision). Par contre, une grande majorité de ces robots inclut des caméras conventionnelles. L'objet des chapitres 3 et suivant est donc de proposer des méthodes d'utilisation conjointe d'une caméra 3D et d'une caméra conventionnelle.

Dans le chapitre 3 ci-après, après un état de l'art sur l'étalonnage extrinsèque entre une caméra 3D et une caméra conventionnelle, nous proposerons une classification selon les deux critères que sont le caractère automatique du processus d'étalonnage et le caractère naturel de la scène utilisée. Enfin, nous proposerons trois modèles d'étalonnage assez éloignés dans la classification proposée puis nous évaluerons leur précision en les comparant avec plusieurs méthodes de l'état de l'art.

Chapitre 3

Etalonnage extrinsèque Caméra conventionnelle-Caméra 3D

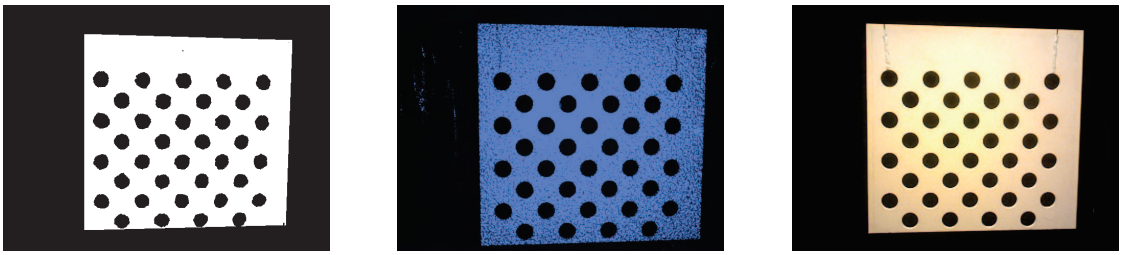
3.1 Introduction

Le robot présenté dans l'introduction dispose - comme l'immense majorité des robots mobiles - de caméras couleur (que nous appellerons aussi caméras conventionnelles). Intégrer une caméra 3D au sein de ce robot nécessite, de la même façon qu'avec le laser 2D au chapitre 1, de réaliser l'étalonnage extrinsèque de la caméra 3D avec les caméras conventionnelles. Un tel étalonnage rend possible la production d'images dites RGB-D apportant des solutions intéressantes à différents problèmes de vision pure ou de télémétrie pure. La plupart des caméras 3D à bas coût embarque elles-mêmes aussi une caméra conventionnelle.

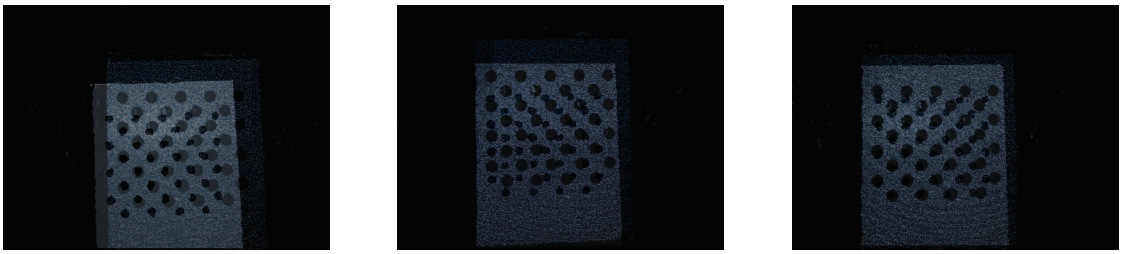
Lors de l'apparition de la première caméra 3D à bas coût en 2011, les premières personnes à rendre publics des travaux sur l'étalonnage extrinsèque entre la caméra conventionnelle intégrée à la Kinect et la caméra 3D ont proposé de réaliser un étalonnage similaire à celui réalisé lors d'un étalonnage de caméras stéréoscopiques en utilisant les images de la caméra conventionnelle et de la caméra infrarouge (IR) sur laquelle est basée la caméra 3D.

En effet, l'hypothèse est simple : puisque l'image de profondeur fournie est calculée à l'aide d'une caméra infrarouge qui observe et compare un motif connu projeté par un projecteur infrarouge avec son image, alors, étalonner la caméra infrarouge revient à étalonner la caméra 3D. Cela signifierait que les paramètres intrinsèques et/ou extrinsèques sont identiques.

Nous pensons que cette hypothèse n'est pas vérifiée et pour le démontrer, il suffit de considérer un contre-exemple : la figure 3.1 représente sur la première ligne, de gauche à droite, les images de profondeur, infrarouge et couleur d'une même scène observée par un même capteur à une position et orientation constantes. La seconde ligne montre la superposition de l'image de profondeur issue de la caméra 3D avec l'image infrarouge en transparence par-dessus pour trois capteurs Kinect différents.



(a) Image de profondeur (b) Image IR (c) Image couleur



(d) Kinect 1 IR & Profondeur (e) Kinect 2 IR & Profondeur (f) Kinect 3 IR & Profondeur

FIGURE 3.1 – Image de Profondeur (a), IR (b) et couleur (c) d'une même scène observée par différents capteurs *Kinect* (d) (e) (f).

L'image 3.1c montre le type de scène employé. Il s'agit d'un plan percé de trous de diamètres identiques selon une grille régulière. L'intérêt d'une telle scène réside dans la possibilité d'observer simultanément sur l'image de profondeur et les images IR et couleur des motifs identiques. Sur la seconde ligne, on peut observer le décalage systématique de

ces motifs entre l'image de profondeur et l'image infrarouge. L'hypothèse mentionnée ci-dessus n'est donc pas valide et la caméra de profondeur (caméra virtuelle), si elle respecte le modèle du sténopé, ne possède pas les mêmes paramètres d'étalonnage que la caméra IR. Puisque l'étalonnage de la caméra conventionnelle avec la caméra IR n'est pas applicable à la caméra 3D, il est donc nécessaire d'étalonner la caméra 3D directement avec la caméra conventionnelle.

Nous verrons dans un premier temps dans la section 3.2, les propositions formulées dans la littérature pour un tel étalonnage et nous présenterons une classification basée sur le critère de l'automaticité du processus et du caractère naturel de la scène utilisée. Nous proposerons dans la section 3.3 trois modèles pour résoudre le problème de l'étalonnage extrinsèque entre une caméra 3D et une caméra conventionnelle. Enfin dans la section 3.4, nous comparerons les modèles produits avec ceux utilisés ou connus dans la littérature.

3.2 Etat de l'art

Les fabricants de caméras 3D et conventionnelles fournissent généralement les paramètres intrinsèques de leurs capteurs. Pour des résultats plus précis, il existe de nombreuses méthodes d'étalonnage intrinsèque d'une caméra respectant le modèle du sténopé. Ces paramètres sont constants aussi longtemps qu'il n'y a pas de changement optique (zoom optique, modification de la mise au point, ...).

Il existe de nombreux travaux dans la littérature sur l'étalonnage extrinsèque et intrinsèque des caméras couleur. Les travaux les plus notables ont été menés par Z. Zhang [Zhang, 1999] [Zhang, 2000] et ont abouti à une méthode maintenant utilisée dans la plupart des outils d'étalonnage. Par exemple, cette méthode a été implémentée dans plusieurs toolboxes d'étalonnage pour Matlab comme la fameuse *Camera Calibration Toolbox for Matlab*[1] de J.-Y. Bouguet ou encore dans le module d'étalonnage de la bibliothèque de vision artificielle *OpenCV*[2]. Cette méthode a par la suite été étendue et d'autres études

1. http ://www.vision.caltech.edu/bouguetj/calib_doc/
2. http ://opencv.org

ont proposé des méthodes pour l'étalonnage de paires de caméras stéréo en utilisant les contraintes épipolaires [Hartley et Zisserman, 2000] [Xu et Zhang, 1996].

Nous faisons l'hypothèse que la caméra 3D est un capteur virtuel respectant le modèle du sténopé dont les paramètres intrinsèques sont connus et fournis par le constructeur. Les nombreux travaux à propos de ces paramètres intrinsèques [Smisek *et al.*, 2013] [Herrera *et al.*, 2012] [Khoshelham et Elberink, 2012] n'infirment pas cette hypothèse puisque leurs résultats montrent des paramètres intrinsèques proches des valeurs du constructeur. Il n'existe pas, à ce jour, de modèle paramétrique décrivant les distorsions (notamment de disparité) présentées dans ces travaux. En pratique, ces distorsions n'apparaissent qu'à partir d'une certaine portée (2,5-3 m pour le capteur Kinect) et donc sont peu présentes dans une habitation en raison de la taille généralement raisonnable des pièces. Nous ne nous intéresserons donc dans cette thèse qu'aux paramètres extrinsèques qui expriment la transformation rigide (rotation et translation) entre la caméra conventionnelle et la caméra virtuelle de profondeur. Nous avons ainsi identifié plusieurs catégories de méthodes :

— **Méthodes de type Stéréo :** Cette première catégorie de méthodes repose sur l'extension de l'étalonnage des paires de caméras stéréoscopiques à l'étalonnage d'une paire de caméra 3D/conventionnelle. Si ces méthodes n'ont pas été évaluées dans ce dernier cas, elles ont été décrites très précisément pour les caméras stéréo [Thacker et Mayhew, 1991] [Hartley et Zisserman, 2000] [Faugeras, 1993]. L'idée générale de ces méthodes est de considérer l'image de profondeur et l'image conventionnelle en lieu et place des deux images conventionnelles (pour l'étalonnage stéréo traditionnel) [Hartley et Zisserman, 2000] [Xu et Zhang, 1996]. Pour de telles méthodes, les deux capteurs doivent être capables de détecter les mêmes formes (ou points remarquables) dans leurs images respectives. Outre les méthodes proposées du même type que les travaux de N. Burrus sur la caméra IR [3] dont nous avons déjà discuté l'hypothèse de départ, il est nécessaire d'utiliser une scène particulièrement contrôlée. Alors que les méthodes pour les paires de caméras stéréoscopiques utilisent très

3. http ://nicolas.burrus.name/index.php/Research/KinectCalibration

souvent un plan étalon de type damier, il faut ici produire un étalon dont le motif doit être localisé au même endroit visuellement (pour la caméra conventionnelle) et géométriquement (pour la caméra 3D). Malgré cette difficulté, ces méthodes ont l'avantage d'estimer conjointement les paramètres intrinsèques et extrinsèques.

— **Méthodes d'appartenance à un plan :** Les premiers travaux autour de l'étalonnage entre une caméra conventionnelle et un capteur 3D ont d'abord été réalisés en considérant les télémètres traditionnels (comme le télémètre laser 2D à balayage). Dans le chapitre 1, nous avons déjà évoqué le résultat le plus notable que l'on peut trouver dans [Zhang et Pless, 2004]. Les auteurs proposent en effet, une méthode d'étalonnage d'une caméra conventionnelle avec un télémètre laser 2D qui utilise des contraintes sur les normales et les distances d'un plan étalon texturé d'un motif de damier. Dans la section 3.3.1, nous montrons qu'il est possible d'étendre cette méthode au cas de l'étalonnage entre une caméra 3D et une caméra conventionnelle. Il s'agit en effet de substituer les capteurs en appliquant les traitements adéquats. Les télémètres laser 3D ont aussi fait l'objet de beaucoup d'études, notamment dans [Unnikrishnan et Hebert, 2005] et [Unnikrishnan, 2006]. La méthode de Zhang a ainsi été employée en remplaçant simplement le télémètre 2D par un télémètre 3D. Dans ces méthodes, l'utilisateur doit manuellement sélectionner la région qui contient le plan étalon dans l'image conventionnelle et dans les données laser. Pour finir, la méthode de Zhang a aussi été appliquée aux caméras 3D dans [Herrera *et al.*, 2012] en remplaçant le télémètre 3D par le nuage de points 3D extrait à partir des images de profondeur d'une caméra 3D. A chaque fois, ces méthodes reposent sur l'observation de multiples plans étalons au travers de différentes scènes où la position et l'orientation du plan varient. [Geiger *et al.*, 2012] propose enfin une méthode permettant de s'affranchir de ces différentes manipulations à condition de positionner dans la scène non plus un plan étalon mais de multiples plans à la fois.

— **Méthodes basées sur une scène naturelle :** Dans [Scaramuzza *et al.*, 2007], les auteurs proposent l'étalonnage d'un très coûteux et précis LIDAR 3D avec une

caméra couleur omnidirectionnelle. Il s'agit ici pour l'utilisateur de sélectionner manuellement les correspondances entre un (grand) nombre de points de l'image couleur et leur homologue dans l'image de profondeur issue du nuage de points du télémètre 3D. Cette méthode peut être directement appliquée aux caméras 3D à bas coût même si les résultats seront nécessairement moins précis en raison de la grande imprécision relative des caméras 3D à bas coût face aux LIDAR 3D employés de type Vélodyne.

Nous l'avons vu, chacune de ces méthodes implique des contraintes fortes qui complexifient largement le processus d'étalonnage. Les méthodes d'appartenance à un plan ou les méthodes basées stéréo, bien que facilement automatisables, imposent d'utiliser un plan artificiel ainsi qu'un grand nombre de vues (à des orientations et positions très diverses) ou un grand nombre de plans (à des orientations et positions très diverses). Le plan nécessaire à l'étalonnage stéréo est en plus particulièrement complexe à contruire précisément sans moyen industriel de type fraiseuse numérique (CNC). Les méthodes naturelles devraient simplifier le processus d'étalonnage puisqu'elles ne nécessitent plus de plan étalon particulier mais en demandant à l'utilisateur de sélectionner manuellement les correspondances, ces méthodes perdent l'automaticité.

Il est d'usage de ne tenir compte que de la précision d'un étalonnage pour estimer sa valeur. Bien que nécessaire, cette métrique ne considère pas la possible erreur humaine. En effet, moins une méthode est simple à mettre en oeuvre, plus l'opérateur (ou l'utilisateur) est susceptible d'introduire des erreurs (précision de la réalisation du plan étalon, de la sélection des correspondances, des manipulations des capteurs, ...) impactant la précision des méthodes. Nous proposons donc de classifier les méthodes d'étalonnage selon deux axes montrant mieux la complexité ou la simplicité du processus : le caractère automatique de la méthode et le caractère naturel de la scène utilisée. Une telle classification est proposée dans la figure 3.2.

Cette figure établit 4 groupes de méthodes dont le premier, *Damier 3D*, représente des méthodes a priori complétement automatiques mais dont l'étalon utilisé est plutôt com-

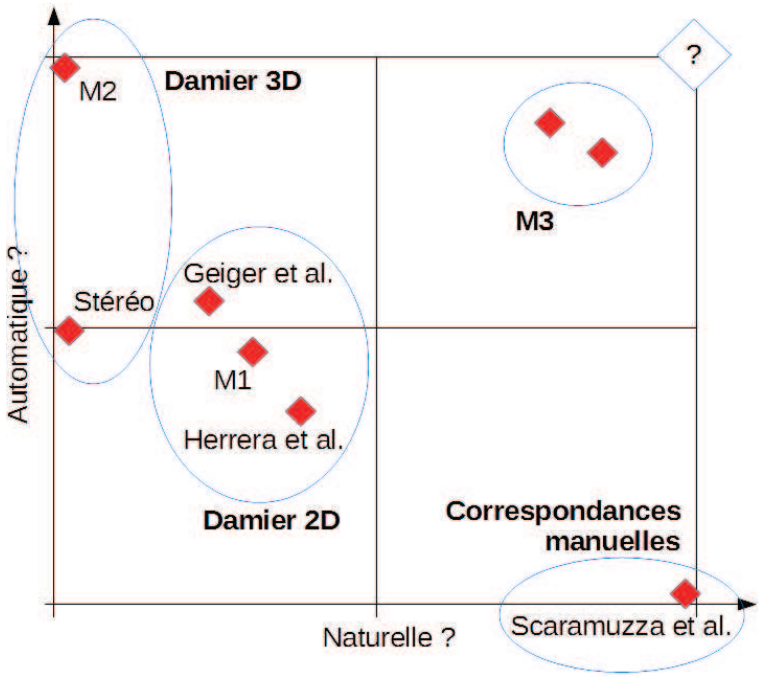


FIGURE 3.2 – Classification des méthodes d'étalonnage extrinsèque entre une caméra 3D et une caméra conventionnelle

plexe car basé sur un motif à la fois visuel et géométrique, aussi bien dans sa réalisation précise que dans son utilisation. L'intérêt d'un tel étalon est justement d'améliorer l'automaticité des méthodes. Le deuxième groupe, *Damier 2D*, est un groupe de méthodes que nous qualifierons d'historique au sens qu'elles sont une continuité naturelle et logique des travaux d'étalonnage des caméras conventionnelles pour les étendre aux capteurs 3D. Elles sont honorablement automatisables et, même si elles utilisent toujours un étalon à manipuler avec soin, celui-ci, basé seulement sur un motif visuel, n'est pas aussi complexe que le *Damier 3D*. Le troisième groupe, *Correspondances manuelles*, est le symétrique du groupe *Damier 3D*, puisqu'il s'agit de méthodes dénuées de tout étalon (méthodes dites naturelles) qui par conséquent perdent l'automaticité des méthodes *Damier 3D*. Enfin, le dernier groupe se rapproche de l'idéal des méthodes complètement automatiques et basées sur scène complétement naturelles. Nous verrons par la suite notre proposition concernant ce groupe *M3*.

Dans la section suivante, nous décrirons les modèles de trois méthodes différentes.

La première, M1, peu naturelle et partiellement automatisable, utilise un plan étalon de type damier 2D. C'est une extension de la méthode de Zhang. La deuxième, M2, est complètement automatique mais pas du tout naturelle puisqu'elle repose sur la détection du plan étalon 3D du même type que celui présenté dans la figure 3.1c. Et enfin, la troisième, M3, est un compromis entre le caractère naturel et le caractère automatique. Elle est en effet basée sur une scène naturelle mais ne nécessite pas une sélection manuelle des correspondances entre l'image couleur et l'image de profondeur.

3.3 Modèles

3.3.1 Méthode d'appartenance à un plan (M1)

Comme nous l'avons montré dans le chapitre 1, l'étalonnage d'une caméra conventionnelle et d'un télémètre 2D se base sur l'estimation des paramètres d'un plan étalon à l'aide de la caméra conventionnelle et l'appartenance des points du télémètre au même plan étalon. Ce type de méthode peut être facilement étendu au cas de l'étalonnage entre une caméra conventionnelle et un télémètre 3D ou une caméra 3D. Il suffit ainsi, soit de remplacer le télémètre 2D par le télémètre 3D (ou la caméra 3D), soit d'estimer les paramètres du plan à l'aide de la caméra 3D et d'utiliser les points issus de la caméra conventionnelle pour l'appartenance au plan étalon.

Parce que les données sont nettement plus denses lorsqu'elles sont issues de la caméra 3D, nous choisissons d'estimer les paramètres du plan étalon à l'aide de la caméra 3D et nous utiliserons les images de la caméra conventionnelle pour produire les points appartenant au plan à l'aide du motif visuel connu imprimé sur le plan étalon (damier). Ce faisant, nous réduisons, pour une pose, l'ensemble des données de la caméra 3D à 4 paramètres (un vecteur normal au plan $\mathbf{n}_{(\pi)}$ de dimension 3 et une distance $d_{(\pi)}$) décrivant le plan étalon et l'ensemble des données de la caméra conventionnelle aux coordonnées des points $\mathbf{p}_C$ représentant chacun un coin dans le motif étalon.

Comme exprimé dans le chapitre 1, tout point $\mathbf{p}_C$ supposé appartenir au plan (π) doit

respecter l'égalité suivante :

$$\mathbf{n}_{(\pi)}^{\top}(\mathbf{R}_K^C\mathbf{p}_C + \mathbf{t}_K^C) - d_{(\pi)} = 0 \tag{3.1}$$

Les points de la caméra conventionnelle $\mathbf{p}_C$ sont issus de la détection automatique de coins du motif damier basée sur un détecteur de Harris simple. Cette méthode efficace est très utilisée dans la plupart des méthodes d'étalonnage de caméra conventionnelle. On trouve une bonne revue de ce genre de détection automatique de motif damier dans [De la Escalera et Armingol, 2010]. Pour calculer $\mathbf{n}_{(\pi)}$ et $d_{(\pi)}$, nous utilisons la même méthode que dans le chapitre 1 où nous exploitons un algorithme RANSAC.

Pour plus de commodités, nous réécrivons la rotation et la translation en une matrice unique $\mathbf{H}_C^K$ qui impose l'utilisation des coordonnées homogènes $\tilde{\mathbf{p}}_C$ pour le point $\mathbf{p}_C$ par la suite :

$$\mathbf{H}_C^K = \mathbf{R_C^K}\begin{pmatrix} 1 & 0 & 0 & \vdots \\ 0 & 1 & 0 & \mathbf{t_C^K} \\ 0 & 0 & 1 & \vdots \end{pmatrix} \tag{3.2}$$

L'équation (3.1) devient alors :

$$\mathbf{n}_{(\pi)}^{\top}\mathbf{H}_C^K\tilde{\mathbf{p}}_C - d_{(\pi)} = 0 \tag{3.3}$$

Puisque $\mathbf{H}_C^K \in \mathbb{R}^{3\times4}$ est l'expression de la transformation rigide à trouver, il y a donc 12 inconnues h_i à estimer :

$$\mathbf{H_C^K} = \begin{pmatrix} h_1 & h_2 & h_3 & h_4 \\ h_5 & h_6 & h_7 & h_8 \\ h_9 & h_{10} & h_{11} & h_{12} \end{pmatrix} \tag{3.4}$$

En considérant l'ensemble des points de la caméra conventionnelle, nous pouvons écrire

n équations en utilisant la contrainte décrite par l'équation (3.3).

$$\begin{cases} \mathbf{n}_{(\pi)}^{\top}\mathbf{H}_C^K\tilde{\mathbf{p}}_{C1} - d_{(\pi)} = 0 \\ \mathbf{n}_{(\pi)}^{\top}\mathbf{H}_C^K\tilde{\mathbf{p}}_{C2} - d_{(\pi)} = 0 \\ \vdots \\ \mathbf{n}_{(\pi)}^{\top}\mathbf{H}_C^K\tilde{\mathbf{p}}_{Cn} - d_{(\pi)} = 0 \end{cases} \tag{3.5}$$

Ce système à 12 inconnues (les éléments de $\mathbf{H}_C^K$) possède une solution unique si $n \geq 12$. A nouveau, nous réécrivons le problème sous une forme linéaire simple en réécrivant la matrice $\mathbf{H}_C^K$ sous la forme d'un vecteur $\mathbf{h}_C^K = [h_1, h_2, h_3, h_4, h_5, h_6, h_7, h_8, h_9, h_{10}, h_{11}, h_{12}]^{\top}$.

On obtient alors n équations (où n est le nombre de points produits par la caméra conventionnelle) de la forme :

$$\left(\mathbf{n}_{(\pi)}^{\top} \otimes \tilde{\mathbf{p}}_{Ci}^{\top}\right)\mathbf{h}_C^K = d_{(\pi)} \otimes \mathbb{1}_n \tag{3.6}$$

En regroupant les termes $\left(\mathbf{n}_{(\pi)}^{\top} \otimes \tilde{\mathbf{p}}_{Ci}^{\top}\right)$ en une matrice $\mathbf{M}$ et $d_{(\pi)} \otimes \mathbb{1}_n$ en un vecteur $\mathbf{d}$, ces équations produisent le problème linéaire suivant :

$$\mathbf{M}\,\mathbf{h}_C^K = \mathbf{d} \tag{3.7}$$

La solution du problème (3.7) est donc $\tilde{\mathbf{h}}_C^K = \mathbf{M}\backslash\mathbf{d}$.

L'opérateur \ représente la division matricielle à gauche telle que $x = A\backslash b$ est une solution de $Ax = b$. Si A est carrée et de rang maximal, $x = A\backslash b$ est équivalent mathématiquement à $x = A^{-1}\,b$. Si A n'est pas de rang maximal ou si A n'est pas carrée (problème surdéterminé), x est tel que $\|Ax - b\|_2$ est minimale (au sens des moindres carrés). Cette solution du problème surdéterminé est calculée à l'aide de la décomposition en valeurs singulières.

A cause du bruit de mesure des données (position des coins), il est alors courant de multiplier les vues du plan étalon. On peut réécrire le système (3.6) comme $n \times m$ équations

de la forme :

$$\forall j = 1..m,\ \forall i = 1..n,\ \left({\mathbf{n}_{(\pi_j)}}^{\top} \otimes \tilde{\mathbf{p}}_{Lij}^{\top}\right) \mathbf{h}_L^K = \mathbf{d}_j \tag{3.8}$$

avec n le nombre de coins et m le nombre de plans.

Par la suite nous optimisons la solution $\breve{\mathbf{H}}$ en résolvant le problème de minimisation suivant :

$$\min_{\breve{\mathbf{H}}} \sum_{j=1}^{m} \sum_{i=1}^{n} \left\| \mathbf{n}_{(\pi_j)}^{\top} . \breve{\mathbf{H}} \mathbf{p}_{Cij} - d_{(\pi_j)} \right\|^2$$
$$s.c. \left\| \breve{\mathbf{h}}_\mathbf{1} \right\| = \left\| \breve{\mathbf{h}}_\mathbf{2} \right\| = \left\| \breve{\mathbf{h}}_\mathbf{3} \right\| = 1,\ \breve{\mathbf{h}}_\mathbf{1} . \breve{\mathbf{h}}_\mathbf{2} = 0\ et\ \breve{\mathbf{h}}_\mathbf{1} . \breve{\mathbf{h}}_\mathbf{3} = 0$$

Ou encore :

$$\min_{r_1, r_2, r_3, \breve{\mathbf{t}}} \sum_{j=1}^{m} \sum_{i=1}^{n} \left\| \mathbf{n}_{(\pi_j)}^{\top} . \left(\mathbf{exp}(\mathbf{r})\, \mathbf{p}_{C_{ij}} + \breve{\mathbf{t}} \right) - d_{(\pi_j)} \right\|^2 \tag{3.9}$$

avec **exp** l'application exponentielle (*exponential map*) permettant de transformer le vecteur de rotation $\mathbf{r} = [r_x, r_y, r_z]^{\top}$ en une matrice de rotation.

En résolvant le problème précédent, on peut finalement extraire $\widehat{\mathbf{R}}_C^K$ et $\widehat{\mathbf{t}}_C^K$:

$$\widehat{\mathbf{R}}_C^K = \mathbf{exp}(\mathbf{r}) \qquad et \qquad \widehat{\mathbf{t}}_C^K = \breve{\mathbf{t}} \tag{3.10}$$

3.3.2 Méthode complètement automatique (M2)

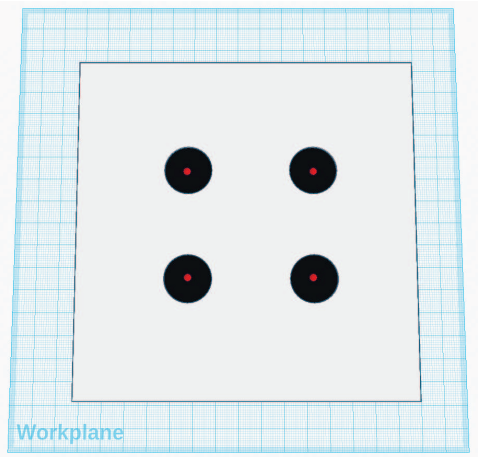
Contrairement à la méthode M1 (section 3.3.1) utilisant de nombreuses poses d'un plan étalon de type damier, il s'agit ici de procéder à l'étalonnage extrinsèque entre une caméra conventionnelle et une caméra 3D en exploitant un unique couple d'images couleur et de profondeur. Nous pouvons diviser cette méthode en deux problèmes séquentiels : la détection de points caractéristiques communs à l'image couleur et à l'image de profondeur

et la minimisation d'une erreur d'alignement entre ces points d'une image à l'autre.

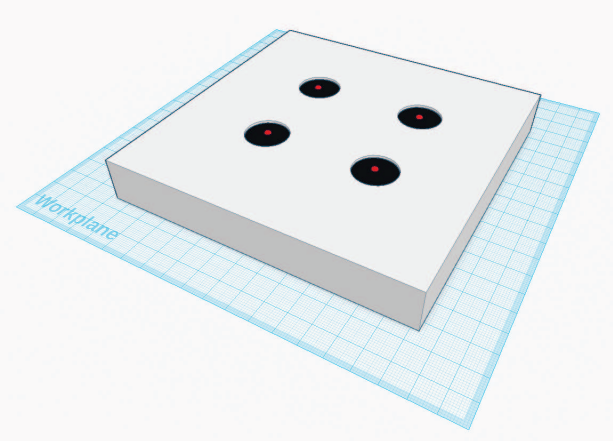
Points caractéristiques communs

Le processus d'étalonnage nécessite de pouvoir détecter des points caractéristiques communs aux images de profondeur et aux images conventionnelles. Pour ce faire, il s'agit d'utiliser une scène qui présente de tels points, c'est-à-dire une scène où il existe des éléments saillants à la fois visuellement et géométriquement. Par mesure de facilité nous proposons d'utiliser une mire *damier 3D* telle que présentée dans la figure 3.3. Il s'agit d'une boîte dont la surface supérieure, blanche, est percée de trous circulaires. La face inférieure et intérieure est colorée en noir de telle sorte que les trous forment une ellipse sombre sur une surface claire quel que soit le point de vue et soient facilement détectables dans l'image couleur. Dans l'image de profondeur, la différence de profondeur entre les surfaces supérieure (blanche) et inférieure (noire) de la boîte se matérialise par une discontinuité dans l'image de profondeur à la circonférence des trous. Cela forme un motif à trous circulaires perçu de la même façon par la caméra couleur et la caméra de profondeur. Nous utilisons une détection automatique d'ellipses dans chaque image après les avoir binarisées afin de trouver les centres des cercles. Dans l'image 3D, comme dans l'image couleur, le centre du cercle ne correspond pas à un point physiquement placé sur une surface mais, pour chaque trou, au point central coplanaire de la surface supérieure.

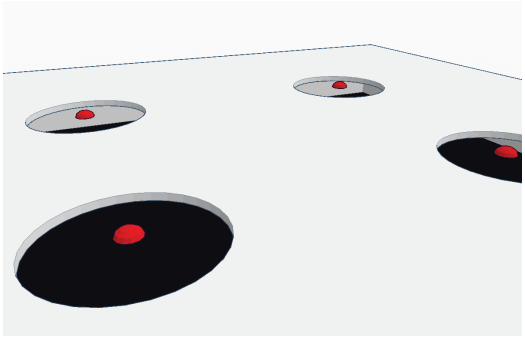
Si on dipose donc facilement des coordonnées des centres des trous dans les deux images (couleur et profondeur), nous avons besoin de connaître les coordonnées 3D dans le repère de la caméra 3D. Habituellement, il suffit d'utiliser la valeur correspondant aux coordonnées relevées dans l'image de profondeur mais dans notre cas, la profondeur ne correspond pas à celle de la surface supérieure mais à la profondeur du point projeté sur la surface inférieure. L'estimation correcte des coordonnées 3D des points caractéristiques (les centres des trous) se base donc sur l'interpolation de la surface supérieure en lieu et place des trous. Il est raisonnable de prendre pour hypothèse que la surface supérieure est planaire (par construction de la mire *damier3D*). Il suffit alors de modéliser ce plan à l'aide d'un certain



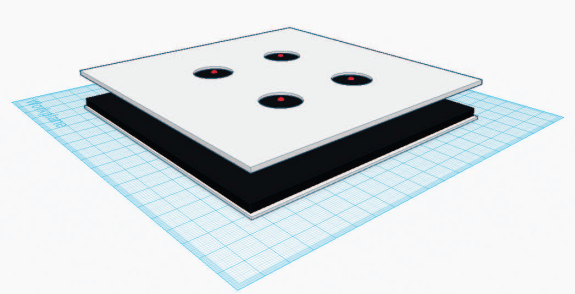
(a) Vue du haut



(b) Vue 3D



(c) Vue rapprochée des points caractéristiques



(d) Vue des surfaces supérieure et inférieure seules

FIGURE 3.3 – Vues synthétiques du damier 3D utilisé pour l'étalonnage. Les points rouges matérialisent l'emplacement des points caractéristiques communs aux deux caméras.

nombre de points placés sur la surface supérieure (on choisira des points placés à l'extérieur de la circonférence des trous) pour pouvoir ensuite obtenir les coordonnées 3D des 4 points caractéristiques.

Il est aisé d'étendre le raisonnement à N points caractéristiques (N supérieur à 3 pour la suite de l'étalonnage). Nous avons choisi N=4 pour faciliter la construction et la détection automatique. Nous pouvons noter qu'il n'est pas nécessaire de disposer précisément les trous, ce qui facilite considérablement le processus d'étalonnage. En effet, dans le cas de l'étalonnage extrinsèque seul et contrairement aux méthodes d'étalonnage conjoint des paramètres intrinsèques et extrinsèques, il n'est pas nécessaire de connaître le modèle de la mire. Pour la détection automatique, il faut que la forme des trous dans les images couleur

et de profondeur soit une ellipse quelle que soit la pose de la mire. Pour ce faire, il suffit de s'assurer que les trous sont effectivement bien circulaires. Une telle mire d'étalonnage peut donc être construite facilement et à peu de frais en quelques minutes avec un carton (pour la structure de la boîte), une feuille de papier noire (pour le fond de la boîte), une feuille de papier blanche (pour la surface supérieure), un compas ou une imprimante (pour dessiner des trous bien circulaires sur la feuille blanche) et une paire de ciseaux.

Asservissement visuel virtuel

Une fois les points caractéristiques correctement détectés dans l'image couleur et profondeur, nous cherchons à calculer la transformation rigide pour passer des points 3D dans le repère de la caméra couleur aux points 3D dans le repère de la caméra 3D. Pour ce faire, nous proposons dans cette section une technique basée sur l'approche d'asservissement visuel. En effet, l'asservissement visuel consiste à proposer une loi de commande permettant de positionner un repère mobile (l'effecteur d'un bras robotisé ou une caméra embarquée sur un robot) d'une configuration initiale vers une configuration désirée en utilisant les mesures issues de la caméra [Chaumette et Hutchinson, 2006]. Lorsque les commandes en vitesse sont appliquées sur le repère d'une caméra mobile, l'intégration du déplacement de cette dernière, de la position initiale vers la position désirée correspond à la transformation rigide entre les configurations, initiale et désirée.

Soit une caméra virtuelle définie par le repère $\mathcal{F}_v$ dont sa position initiale est confondue avec la position de la caméra 3D. En considérant la position de la caméra conventionnelle comme la configuration désirée pour la caméra virtuelle, il est possible de définir une fonction de tâche [Samson *et al.*, 1991] qui utilise les primitives visuelles normalisées issues des caméras 3D et conventionnelle. Cette fonction de tâche peut être utilisée pour définir une loi de commande en asservissement visuel permettant de déplacer la caméra virtuelle de la configuration initiale (la position de la caméra 3D) vers la configuration désirée (la position de la caméra conventionnelle). Par la suite, nous allons utiliser les centres des trous de la mire (proposée précédemment) comme primitives visuelles pour établir une loi

de commande en asservissement visuel virtuel.

En connaissant les matrices des paramètres intrinsèques $\mathbf{K}_d$ et $\mathbf{K}_c$ des caméras 3D et conventionnelle respectivement, il est possible d'exprimer les coordonnées des points caractéristiques de l'image de la caméra 3D et de l'image de la caméra conventionnelle en coordonnées normalisées :

$$\begin{pmatrix} x_c \\ y_c \\ 1 \end{pmatrix} = \mathbf{K}_c^{-1} \begin{pmatrix} u_c \\ v_c \\ 1 \end{pmatrix} \quad \text{et} \quad \begin{pmatrix} x_d \\ y_d \\ 1 \end{pmatrix} = \mathbf{K}_d^{-1} \begin{pmatrix} u_d \\ v_d \\ 1 \end{pmatrix}$$

avec $(u_d \; v_d \; 1)^\top$ et $(u_c \; v_c \; 1)^\top$ les coordonnées homogènes d'un point caractéristique, exprimées en pixel respectivement dans les repères de la caméra 3D et de la caméra conventionnelle.

Soit $\mathbf{s}^*$ (respectivement $\mathbf{s}$) le vecteur des primitives visuelles désirées (respectivement initiales/courantes) contenant les coordonnées normalisées des centres des trous dans l'image de la caméra conventionnelle (respectivement caméra 3D). Une tâche de positionnement contrôlant les six degrés de liberté de la caméra virtuelle (repère $\mathcal{F}_v$) peut être décrite par la régulation d'une fonction de tâche [Samson *et al.*, 1991] :

$$\mathbf{e} = \mathbf{s} - \mathbf{s}^*$$

Il s'agira donc de réaliser une tâche de positionnement par asservissement visuel virtuel entre la caméra 3D virtuellement mobile (attachée au repère $\mathcal{F}_v$) et la caméra conventionnelle.

Soit $\mathbf{v}$ le torseur cinématique de la caméra virtuelle exprimé dans $\mathcal{F}_v$. La relation entre la variation $\dot{\mathbf{s}}$ et le torseur cinématique $\mathbf{v}$ est donnée par :

$$\dot{\mathbf{s}} = \mathbf{L}_\mathbf{s} \, \mathbf{v}$$

où $\mathbf{L_s}$ est appelée la matrice d'interaction de dimension $2N \times 6$ avec N le nombre de points caractéristiques utiles (les centres des trous) définissant $\mathbf{s}$. L'expression de la matrice d'interaction pour une seule primitive visuelle de type points de coordonnées normalisées $\mathbf{x} = (x \ \ y)^\top$, est donnée par [Chaumette et Hutchinson, 2006] :

$$\mathbf{L_x} = \begin{pmatrix} -\frac{1}{Z} & 0 & \frac{x}{Z} & xy & -(1+x^2) & y \\ 0 & -\frac{1}{Z} & \frac{y}{Z} & 1+y^2 & -xy & -x \end{pmatrix}$$

Pour obtenir une décroissance exponentielle de l'erreur $\mathbf{e} = \mathbf{s} - \mathbf{s}^*$, nous avons utilisé la loi de commande classique suivante :

$$\mathbf{v} = -\lambda \, \mathbf{L_s^+} \, \mathbf{e}$$

où λ est un gain positif régissant la vitesse de décroissance de l'erreur $\mathbf{e}$.

Etant en asservissement visuel virtuel, on peut définir un temps d'échantillonnage constant qui permet de calculer simplement une rotation et une translation entre deux instants $t-1$ et t en intégrant la vitesse $\mathbf{v}$. On obtient alors les paramètres extrinsèques $\mathbf{R}$ et $\mathbf{t}$ de l'étalonnage caméra 3D et conventionnelle, en intégrant sur toute la durée de l'asservissement visuel :

$$\begin{cases} \mathbf{R} = \prod_{0}^{i=n} \mathbf{R}_i \\ \mathbf{t} = \left(\prod_{0}^{i=n} \mathbf{R}_i \right) \mathbf{t}_0 + \sum_{k=1}^{n} \left[\left(\prod_{n}^{i=1+k} \mathbf{R}_i \right) \mathbf{t}_k \right] \\ \text{avec } \mathbf{t}_0 = (0 \ 0 \ 0)^\top \text{ et } \mathbf{R}_0 = \mathbb{1}_{3\times 3} \end{cases}$$

Comme nous pouvons le constater, le modèle d'étalonnage proposé dans cette section nécessite la construction (simple) d'un motif étalon 3D. Comme le processus ne requiert qu'une unique prise de vue et que la détection des points caractéristiques peut être complètement automatique, on peut considérer que la quantité d'interaction avec l'utilisateur

est quasi nulle. Cela permet d'affirmer que la méthode est complètement automatique. Cependant, la nécessité d'utiliser des points caractéristiques communs aux deux caméras conventionnelles et 3D ne permet pas d'utiliser une scène naturelle et de se libérer de l'utilisation d'une mire d'étalonnage pour laquelle nous avons proposé une forme facile à construire.

La première méthode présentée (M1) était finalement moyennement automatique et faiblement naturelle. Le modèle formulé dans cette section (M2) n'est pas du tout naturel mais complètement automatique. Dans la section suivante, nous nous attacherons à proposer un processus d'étalonnage (M3) présentant les deux avantages de l'automaticité et du caractère naturel de la scène.

3.3.3 Méthode automatique et naturelle (M3)

Encore une fois, il s'agit ici de n'effectuer qu'un étalonnage extrinsèque entre une caméra conventionnelle et une caméra 3D. Les paramètres intrinsèques sont donc supposés connus. Comme nous l'avons évoqué dans la section 3.3.2, l'une des pistes d'automatisation envisagée par la communauté et notamment par [Geiger *et al.*, 2012] se base sur la réduction du nombre de vues nécessaires aux algorithmes d'étalonnage. Dans cette section, nous développerons une méthode qui nécessite seulement deux couples d'images couleur associées aux repères $\mathcal{C}_1$ et $\mathcal{C}_2$ et de profondeur associées aux repères $\mathcal{D}_1$ et $\mathcal{D}_2$. Nous supposons que les différentes distorsions sont corrigées aussi bien pour l'image couleur [Tsai, 1987] que pour l'image de profondeur [Teichman *et al.*, 2013].

Soit w une fonction générale non-linéaire de *warping* qui tranforme un pixel ${}^{\mathcal{C}_1}\mathbf{m}_{1i}$ représenté dans le repère image $\mathcal{C}_1$ en un pixel ${}^{\mathcal{C}_2}\mathbf{m}_{1i}$ représenté dans le repère image $\mathcal{C}_2$:

$$ {}^{\mathcal{C}_2}\mathbf{m}_{1i} = w({}^{\mathcal{C}_1}\mathbf{m}_{1i}, {}^{\mathcal{V}}z_{1i}, \mathbf{K}, \mathbf{K}', \mathbf{T}', \widetilde{\mathbf{T}}) \;\; \forall\, {}^{\mathcal{C}_1}\mathbf{m}_{1i} \; un\ point\ de\ l'image\ 1\ inclu\ dans\ \mathcal{M}^* \quad (3.11) $$

avec $\mathcal{M}^* = \{(\mathbf{m}_{1i}, \mathbf{m}_{2i})\} = \left\{([u_{1i}, v_{1i}, 1]^\top, [u_{2i}, v_{2i}, 1]^\top)\right\}$, l'ensemble des paires de points d'intérêt mis en correspondance entre les images couleur 1 et 2 exprimés en coordonnées

homogènes. ${}^{\mathcal{V}}z_{1i}$ correspond à la profondeur du point que nous explicitons plus loin. $\mathbf{K}$ et $\mathbf{K}'$ sont les matrices des paramètres intrinsèques respectivement pour les caméras conventionnelle et 3D. $\mathbf{T}' \sim (\mathbf{R}', \mathbf{t}') \in \mathbb{SE}^3$ est la transformation rigide permettant de passer de la première pose de la caméra 3D à la deuxième pose. Ces deux poses correspondent respectivement aux images de profondeur $\mathcal{D}_1$ et $\mathcal{D}_2$. La transformation rigide $\widetilde{\mathbf{T}} \sim (\widetilde{\mathbf{R}}, \widetilde{\mathbf{t}}) \in \mathbb{SE}^3$ représente la transformation entre la caméra 3D et la caméra conventionnelle. La matrice $\widetilde{\mathbf{T}}$ est construite à partir des paramètres extrinsèques à estimer par l'étalonnage.

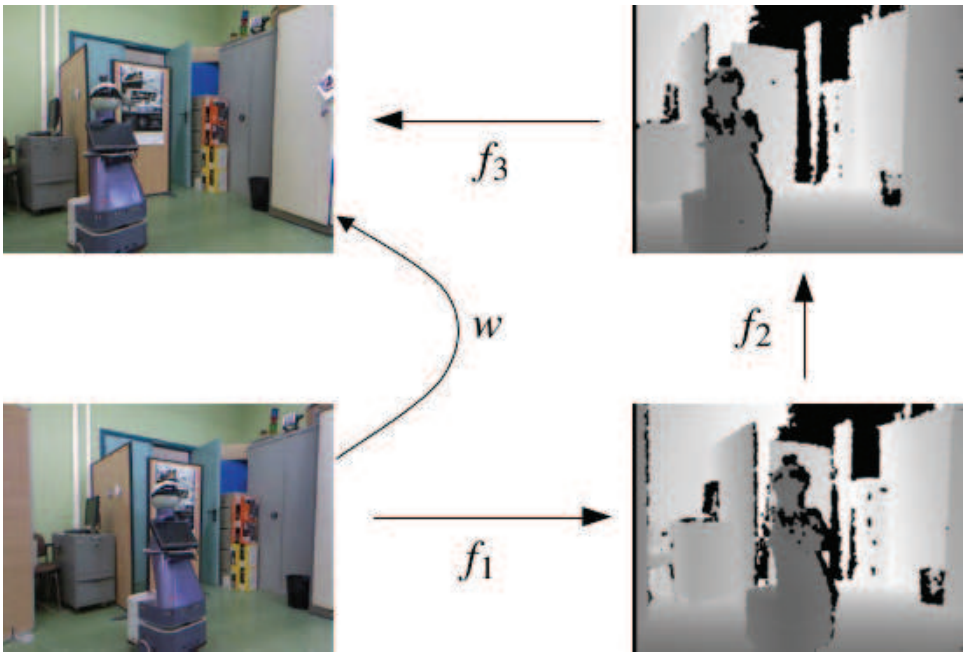


FIGURE 3.4 – Fonctions successives reprojetant les points d'intérêt de l'image couleur 1 (en bas) dans l'image couleur 2 (en haut) en passant par les images de profondeur

Comme le montre la figure 3.4 qui représente les images couleur et de profondeur à partir de deux points de vues différents, la fonction de *warping* w peut être décomposée en 3 fonctions f_1, f_2 et f_3 :

$$w = f_1 \circ f_2 \circ f_3$$

En écrivant les variables et les paramètres des différentes fonctions, on obtient l'équation

suivante :

$$\begin{aligned} w({}^{\mathcal{C}_1}\mathbf{m}_{1i}, {}^{\mathcal{V}}z_{1i}, \mathbf{K}, \mathbf{K}', \mathbf{T}'; \widetilde{\mathbf{T}}) \\ = f_3(f_2(f_1({}^{\mathcal{C}_1}\mathbf{m}_{1i}, {}^{\mathcal{V}}z_{1i}|\mathbf{K}, \mathbf{K}'; \widetilde{\mathbf{T}})|\mathbf{K}', \mathbf{T}')|\mathbf{K}, \mathbf{K}'; \widetilde{\mathbf{T}}) \end{aligned} \tag{3.12}$$

— La fonction f_1 lie chaque point de l'image couleur $\mathcal{C}_1$ avec son correspondant dans l'image de profondeur $\mathcal{D}_1$. Ainsi, nous définissons une image virtuelle de profondeur $\mathcal{V}$ à partir de laquelle il est possible de connaître la profondeur ${}^{\mathcal{V}}z_{1i}$ de chaque pixel de l'image couleur. Par exemple, ${}^{\mathcal{V}}z_{1i}$ correspond à la profondeur du point de coordonnées ${}^{\mathcal{C}_1}\mathbf{m}_{1i}$ dans l'image conventionnelle $\mathcal{C}_1$. f_1 est donc définie telle que :

$${}^{\mathcal{D}_1}\mathbf{m}_{1i} = f_1({}^{\mathcal{C}_1}\mathbf{m}_{1i}, {}^{\mathcal{V}}z_{1i}|\mathbf{K}, \mathbf{K}'; \widetilde{\mathbf{T}}) \tag{3.13}$$

— La fonction f_2 est utilisée pour exprimer, pour tout point de l'image de profondeur $\mathcal{D}_1$, le point correspondant dans le repère de l'image de profondeur $\mathcal{D}_2$:

$${}^{\mathcal{D}_2}\mathbf{m}_{1i} = f_2({}^{\mathcal{D}_1}\mathbf{m}_{1i}|{}^{\mathcal{V}}z_{1i}, \mathbf{K}', \mathbf{T}') \tag{3.14}$$

${}^{\mathcal{D}_2}\mathbf{m}_{1i}$ et ${}^{\mathcal{D}_1}\mathbf{m}_{1i}$ représentent un même point exprimé sous la forme de deux pixels dans les images de profondeur associées aux repères 1 et 2 de la caméra 3D.
Il s'agit soit d'une mise en correspondance totale des images de profondeur soit de transformer les données de l'image de profondeur $\mathcal{D}_1$ vers l'image de profondeur $\mathcal{D}_2$. Comme nous admettons l'hypothèse que la caméra 3D respecte le modèle du sténopé, les paramètres instrinsèques représentés par la matrice $\mathbf{K}'$ sont connus et constants entre les deux images de profondeur. De cette façon, la fonction dépend principalement de la transformation rigide entre les deux points de vues de la caméra 3D. Plusieurs stratégies peuvent être employées pour estimer cette tranformation : odométrie basée caméra 3D, SLAM, appariement 3D, ICP, ...
Théoriquement, n'importe quelle méthode calculant une mise en correspondance de points 3D (ou d'alignement de nuages de points) peut être utilisée. En pratique, on

cherchera une méthode rapide, robuste au bruit et aux manques de données (très fréquents dans les caméras 3D à bas coût), ce qui nous pousse à proposer en section 3.3.4 un algorithme dédié.

— la fonction f_3 assure la projection des données de l'image de profondeur $\mathcal{D}_2$ vers l'image couleur $\mathcal{C}_2$. On peut aussi définir f_3 comme étant la fonction inverse de f_1.

$$^{\mathcal{C}_2}\mathbf{m}_{1i} = f_3(^{\mathcal{D}_2}\mathbf{m}_{1i}|\mathbf{K}, \mathbf{K}'; \widetilde{\mathbf{T}}) = f_1^{-1}(^{\mathcal{D}_2}\mathbf{m}_{1i}|\mathbf{K}, \mathbf{K}'; \widetilde{\mathbf{T}}) \tag{3.15}$$

Maintenant que nous avons défini tous les éléments de la fonction globale de *warping*, il est facile de déterminer les paramètres inconnus, les paramètres connus, les variables à estimer et les données nécessaires en entrée du problème :

— Mesures : $^{\mathcal{C}_1}\mathbf{m}_{1i}$, $^{\mathcal{C}_2}\mathbf{m}_{2i}$ et $^{\mathcal{V}}z_{1i}$ (5 éléments)

— Paramètres connus : $\mathbf{K}$ et $\mathbf{K}'$ (8 éléments)

— Paramètres inconnus : $\widetilde{\mathbf{T}}$ et $\mathbf{T}'$ (12 éléments)

Comme nous l'avons dit précédemment, nous proposons en section 3.3.4 une méthode pour estimer $\mathbf{T}'$. Cette méthode se base uniquement sur les données de la caméra 3D et est donc totalement indépendante de l'estimation de $\widetilde{\mathbf{T}}$. Nous pouvons donc considérer $\widetilde{\mathbf{T}}$ comme la seule inconnue. $\widetilde{\mathbf{T}}$ est une matrice représentant une transformation rigide à 6 degrés de liberté et équivaut donc à 6 paramètres inconnus à estimer.

Soit r une fonction de coût de *warping* d'un point d'intérêt de $\mathcal{C}_1$ dans $\mathcal{C}_2$ en utilisant la distance euclidienne (en pixels) :

$$\begin{aligned} r &= \sum_i \left\| ^{\mathcal{C}_2}\mathbf{m}_{2i} - ^{\mathcal{C}_2}\mathbf{m}_{1i} \right\|^2 \\ &= \sum_i \left\| ^{\mathcal{C}_2}\mathbf{m}_{2i} - w(^{\mathcal{C}_1}\mathbf{m}_{1i}, ^{\mathcal{V}}z_{1i}|\mathbf{K}, \mathbf{K}', \mathbf{T}'; \widetilde{\mathbf{T}}) \right\|^2 \end{aligned} \tag{3.16}$$

Où $(^{\mathcal{C}_2}\mathbf{m}_{2i}, ^{\mathcal{C}_1}\mathbf{m}_{1i}) \in \mathcal{M}^*$ et $^{\mathcal{V}}z_{1i} \in \mathbb{R}$. Alors, nous pouvons réécrire le problème d'éta-

lonnage comme la minimisation de la fonction de coût r :

$$\widetilde{\mathbf{T}} = \underset{\mathbf{T}\in\mathbb{SE}^3}{\arg\min} \left\{ \sum_i \left\| {}^{\mathcal{C}_2}\mathbf{m}_{2i} - w({}^{\mathcal{C}_1}\mathbf{m}_{1i}, {}^{\mathcal{V}}z_{1i} | \mathbf{K}, \mathbf{K}', \mathbf{T}'; \mathbf{T}) \right\|^2 \right\} \tag{3.17}$$

Si ${}^{\mathcal{C}_2}\mathbf{m}_{2i}$, ${}^{\mathcal{C}_1}\mathbf{m}_{1i}$, $\mathbf{K}$, $\mathbf{K}'$ et $\mathbf{T}'$ sont connus, il est alors possible de calculer une solution numérique en employant une minimisation par les moindres carrés utilisant l'algorithme de Levenberg-Marquardt [Marquardt, 1963] [Levenberg, 1944]. Nous obtenons ainsi les 6 paramètres non linéaires (3 pour la rotation et 3 pour la translation) à condition de disposer d'une initialisation pour $\widetilde{\mathbf{T}}$.

Processus d'étalonnage

Finalement, le processus d'étalonnage requiert 7 étapes :

— *1ère étape :* Acquisition des données ${}^{\mathcal{C}_1}\mathbf{m}_{1i}$ et ${}^{\mathcal{D}_1}\mathbf{m}_{1i}$ à la première position du robot transportant la caméra conventionnelle et la caméra 3D.

— *2ème étape :* Le robot effectue un déplacement qui provoque un déplacement $\mathbf{T}'$ de la caméra 3D.

— *3ème étape :* Acquisition des données ${}^{\mathcal{C}_2}\mathbf{m}_{2i}$ et ${}^{\mathcal{D}_2}\mathbf{m}_{2i}$ à la nouvelle position du robot.

— *4ème étape :* Estimation de $\mathbf{T}'$ (Cf. section 3.3.4).

— *5ème étape :* Extraction de points d'intérêt et mise en correspondance : $\mathcal{M}^*$,

Le processus d'étalonnage repose sur la possibilité d'extraire et de mettre en correspondance des points d'intérêt ${}^{\mathcal{C}_1}\mathbf{m}_{1i}$ et ${}^{\mathcal{C}_2}\mathbf{m}_{2i}$, respectivement des images couleurs 1 et 2.

Dans ces travaux, nous utilisons principalement les détecteurs et descripteurs SURF [Bay *et al.*, 2008]. En effet, selon [Bauer *et al.*, 2007], lorsqu'ils sont associés même à de très simples algorithmes de mise en correspondance basée sur une simple distance, ils offrent un faible taux de faux appariements tout en consommant un temps de calcul raisonnable ainsi qu'un faible nombre de points. Le nombre de points mis en correspondance a une importance car il influe sur les résultats d'étalonnage ainsi que sur le temps d'optimisation.

En effet, la fonction de coût utilisée lors de la minimisation se base sur une distance euclidienne entre un point mis en correspondance et la reprojection de son homologue dans un même espace image. Notons que l'étalonnage est aussi tout à fait fonctionnel en utilisant d'autres détecteurs et descripteurs (SIFT, ORB, BRISK, ...).

L'importance de la qualité de la mise en correspondance pousse à introduire des améliorations, par exemple l'utilisation de l'algorithme RANSAC comme dans [Kitt *et al.*, 2010] ou l'introduction de contraintes particulières sur la scène : si la scène naturelle est un mur texturé ou un poster, alors nous pouvons utiliser l'homographie pour rejeter les appariements qui ne respectent pas cette contrainte. La figure 3.5 montre un exemple d'appariement utilisant les points d'intérêt SURF sur une scène en milieu intérieur.

La dernière contrainte de l'étalonnage consiste à disposer de l'information de profondeur pour les points mis en correspondance. En effet, les caméras 3D à bas coût ont la particularité de ne pas être capables de réaliser des mesures de profondeur sur certaines zones (en particulier sur les arêtes des objets). Plus le nombre de points mis en correspondance est important, plus la probabilité de ne pas pouvoir extraire l'information de profondeur augmente. N'importe quel détecteur peut être utilisé mais en raison des mesures manquantes fréquentes sur les bords des objets dans l'image de profondeur, il faut favoriser les détecteurs qui privilégient les surfaces plutôt que les bords.

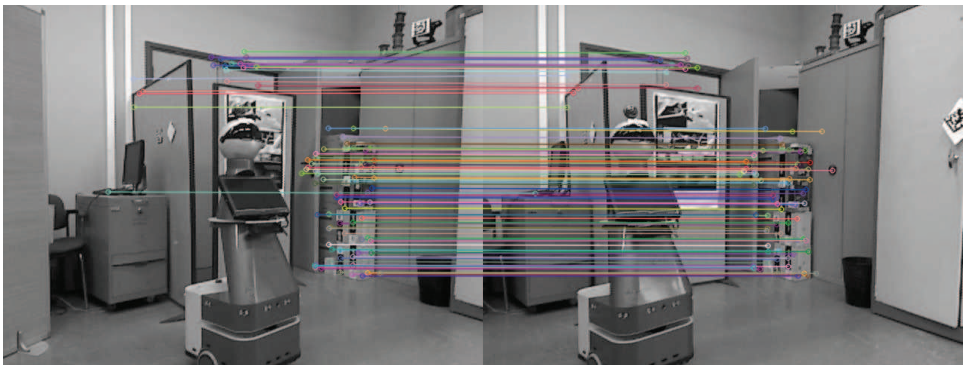


FIGURE 3.5 – Mise en correspondance de deux vues d'une scène à l'aide des points caractéristiques SURF. Aucun faux appariement après réjection par RANSAC.

— *6ème étape :* Initialisation de $\widetilde{\mathbf{T}}$ et génération de l'image virtuelle de profondeur

$\{^{\mathcal{V}}z_{1i}\}$.

Le processus de minimisation impose de commencer par la génération d'une image virtuelle de profondeur avec les paramètres intrinsèques et la pose de la caméra conventionnelle comme le montre la figure 3.6. En d'autres termes, il est nécessaire de créer une image de profondeur dans laquelle nous trouverons la profondeur aux mêmes coordonnées que la couleur dans l'image de la caméra conventionnelle. Ceci est strictement équivalent à la création d'une image RGB-D et nécessite une valeur pour $\widetilde{\mathbf{T}}$.

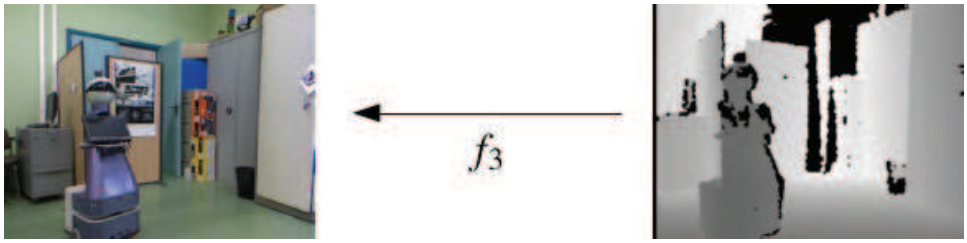


FIGURE 3.6 – Ajout d'un canal D à une image RGB pour former une image RGB-D à partir d'une image RGB et d'une image de profondeur

Cette opération nécessite d'utiliser la fonction $f_3(^{\mathcal{D}_2}\mathbf{m}_{1i}|\mathbf{K},\mathbf{K}';\widetilde{\mathbf{T}})$ définie dans la section 3.3.3 pour tous les points $^{\mathcal{D}_2}\mathbf{m}_{1i}$ dans l'image de profondeur $\mathcal{D}_2$. Pour l'étalonnage des caméras couleur et 3D internes aux caméras 3D à bas coût, la transformation $\widetilde{\mathbf{T}}$ peut être initialisée sans rotation ni translation du fait que ces dernières sont de faible grandeur dans ce type d'équipement. S'agissant de l'étalonnage de caméras plus distantes, une initialisation mesurée à main levée est utile puisqu'elle permet de limiter les risques de tomber dans un minimum local et diminue le temps de minimisation. Evidemment, à cette étape l'image virtuelle de profondeur ou l'image RGB-D ne représente pas une vue correcte puisque l'initialisation imprécise de $\widetilde{\mathbf{T}}$ génère probablement des inexactitudes.

— *7ème étape :* Minimisation de la fonction de coût r.

La fonction f_1 se base sur le nouveau calcul de $\mathcal{V}$ à chaque étape de la minimisation, dès que $\widetilde{\mathbf{T}}$ est mise à jour par l'algorithme de minimisation. Au début totalement erronée, l'image virtuelle de profondeur va être progressivement corrigée à mesure que la minimisation converge vers la solution.

3.3.4 Alignement de nuages de points 3D contenant des plans

La mise en correspondance des points 3D issus des deux images de profondeur peut être estimée de différentes façons selon la nature du mouvement de la caméra 3D. Le cas le plus simple consiste en une translation pure du robot. Dans ce cas, la caméra 3D effectue exactement le même mouvement. Il est ainsi possible d'obtenir une estimation précise du mouvement de la caméra 3D en faisant une mesure ou en utilisant le capteur odométrique du robot.

Si le mouvement est plus complexe et est composé à la fois d'une translation et d'une rotation, utiliser l'odométrie du robot implique de connaître la transformation entre le centre de la base mobile et la caméra 3D. Certains travaux considèrent ce type de problème notamment pour le calcul de la position et de l'orientation d'un capteur sur un bras robotisé [Ilonen et Kyrki, 2011] [Jordt *et al.*, 2009]. Ces travaux peuvent être aisément appliqués au cas simplifié du robot mobile.

L'ajout d'un autre étalonnage complexifie le processus et augmente l'incertitude des résultats. Disposant de deux images de profondeur, nous proposons d'utiliser plutôt un algorithme d'alignement de nuages de points 3D. Dans ses travaux, [Geiger *et al.*, 2012] utilise aussi un algorithme d'alignement 3D pour trouver la rotation entre deux ensembles de vecteurs. Dans les faits, l'algorithme proposé par Geiger et al. est une simplification (selon certaines hypothèses) de l'algorithme de Kabsch [Kabsch, 1976] conçu à l'origine pour trouver la rotation qui aligne au mieux deux ensembles de sommets dans une structure cristallographique.

L'algorithme de Kabsch se base sur le concept élémentaire du produit scalaire :

$$a.b = \|a\| \ \|b\| \ cos(\phi)$$

où ϕ est l'angle orienté entre le vecteur a et le vecteur b.

$$\frac{a}{\|a\|} . \frac{b}{\|b\|} = cos(\phi)$$

On retrouve ici l'expression du cosinus de l'angle entre les vecteurs a et b qui correspond au produit scalaire de ces vecteurs normalisés. Intuitivement, on voit bien que si les deux vecteurs sont colinéaires, c'est-à-dire si $\phi = 0$ alors $cos(\phi) = 1$. De même si tous les n vecteurs sont colinéaires deux à deux alors :

$$\sum_{i \in 1..n} \frac{a_i}{\|a_i\|} . \frac{b_i}{\|b_i\|} = n$$

En introduisant une matrice de rotation quelconque $\mathbf{R} \in \mathbb{SO}(3), \forall b_i \in \mathbb{R}^3, \exists! a'_i \in \mathbb{R}^3$ tel que $\frac{b_i}{\|b_i\|} = \mathbf{R} \frac{a'_i}{\|a'_i\|}$ et si tous les a_i sont colinéaires avec les b_i alors :

$$\sum_{i \in 1..n} \frac{a_i}{\|a_i\|} . \mathbf{R} \frac{a'_i}{\|a'_i\|} = n \tag{3.18}$$

En milieu intérieur, l'information la plus facile à extraire et la plus robuste dans la phase de détection est la présence de plans (sol, plafond, murs, meubles). Nous choisissons donc d'utiliser les vecteurs normaux de ces plans extraits à partir d'estimations successives des plans principaux en utilisant l'algorithme RANSAC.

Soient ${}^{\mathcal{D}_1}\mathbf{n}_i$ et ${}^{\mathcal{D}_2}\mathbf{n}_i$ l'ensemble des vecteurs normaux aux plans observés par la caméra 3D respectivement aux poses 1 et 2. Si ces normales sont supposées mises en correspondance alors, l'algorithme de Kabsch permet d'estimer la matrice de rotation qui aligne de façon optimale ${}^{\mathcal{D}_1}\mathbf{n}_i$ et ${}^{\mathcal{D}_2}\mathbf{n}_i$. Cette matrice de rotation $\mathbf{R}$ s'exprime en reprenant l'équation 3.18 comme :

$$\begin{aligned} \widetilde{\mathbf{R}} &= \underset{\mathbf{R} \in \mathbb{SO}(3)}{\arg\max} \left\{ \sum_i {}^{\mathcal{D}_1}\mathbf{n}_i^\top \mathbf{R}\, {}^{\mathcal{D}_2}\mathbf{n}_i \right\} \\ &= \mathbf{V} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & d \end{pmatrix} \mathbf{U}^\top \end{aligned} \tag{3.19}$$

où $\mathbf{U}$ et $\mathbf{V}$ proviennent de la décomposition en valeurs singulières (SVD) de la matrice

$\sum_i^{\mathcal{D}_2} \mathbf{n}_i^{\mathcal{D}_1} \mathbf{n}_i^\top = \mathbf{U}\Sigma\mathbf{V}^\top$ et d est le signe du déterminant $det(\mathbf{V}\mathbf{U}^\top)$.

Finalement, la translation $\mathbf{t}$ entre les deux poses de la caméra 3D est donnée par l'estimation $\widetilde{\mathbf{t}}$. $\widetilde{\mathbf{t}}$ minimise la distance au plan des centres de gravité ${}^{\mathcal{D}_1}\mathbf{c}_i$ et ${}^{\mathcal{D}_2}\mathbf{c}_i$ des différents plans. Alors, on obtient un nouveau problème d'optimisation :

$$\widetilde{\mathbf{t}} = \arg\min_{\mathbf{t}} \left\{ \sum_i ((\widetilde{\mathbf{R}}^{\mathcal{D}_2}\mathbf{c}_i + \mathbf{t} - {}^{\mathcal{D}_1}\mathbf{c}_i)^\top \, {}^{\mathcal{D}_1}\mathbf{n}_i)^2 \right\} \tag{3.20}$$

dont l'expression définit un score global d'alignement des nuages de points 3D.

Au début de ce raisonnement, nous avions supposé les mises en correspondance des normales (et des centres de gravité) connues. Penchons-nous maintenant sur cette mise en correspondance. Dans l'exemple suivant, nous prendrons l'exemple d'une scène composée de k plans. Nous disposons de deux vues de ces k plans à des poses différentes et nous faisons l'extraction des k plans par un algorithme RANSAC dans chacune des vues. Nous obtenons donc k vecteurs normaux ainsi que k centres de gravité pour chaque vue.

Il existe $k!$ façons de mettre en correspondance k plans entre les deux vues. Tant que k est petit, nous pouvons calculer $\widetilde{\mathbf{R}}$ et $\widetilde{\mathbf{t}}$ pour chaque combinaison possible et nous choisissons $\widetilde{\mathbf{R}}$ et $\widetilde{\mathbf{t}}$ tels qu'ils minimisent le score présenté dans l'équation 3.20. L'arrangement qui minimise ce score représente donc la mise en correspondance des plans entre les deux vues.

La figure 3.8 montre la qualité d'alignement des deux vues d'une scène représentée dans la figure 3.7 et composée de 3 plans à partir des données de la caméra 3D. On observe en bleu le nuage de points composé de trois plans capturés à la première pose. Le nuage de points vert correspond à la transformation de ce premier nuage de points à l'aide de la matrice de rotation estimée $\widetilde{\mathbf{R}}$ et du vecteur de translation estimé $\widetilde{\mathbf{t}}$. Le nuage de points rouge correspond à ce qui a été observé par la caméra 3D à la deuxième pose. Nous pouvons noter que le nuage de points vert est parfaitement aligné avec le nuage de points rouge, ce qui prouve la bonne estimation de la transformation entre la première et la deuxième pose.

Finalement, la procédure d'alignement 3D repose sur 4 étapes :

— L'estimation des normales $\mathbf{n}_i$ en utilisant un RANSAC et le calcul des centres de



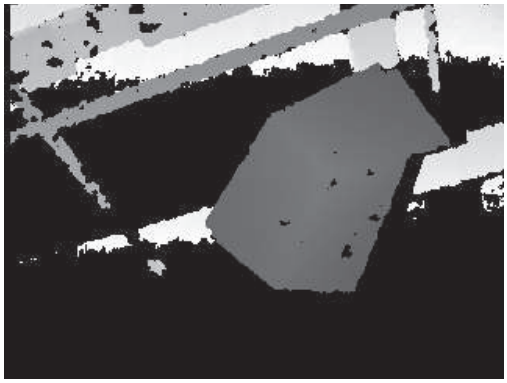
(a) Image couleur, Pose 1



(b) Image de profondeur, Pose 1



(c) Image couleur, Pose 2



(d) Image de profondeur, Pose 2

FIGURE 3.7 – Images de profondeur (à droite) et couleur (à gauche) d'une même scène à partir de deux poses (une pose par ligne).

gravité des plans $\mathbf{c}_i$ en utilisant seulement les *inliers*,

— L'estimation de la rotation en utilisant l'équation (3.19) pour chaque mise en correspondance possible,

— L'estimation de la translation en utilisant l'équation (3.20) pour chaque mise en correspondance possible,

— La sélection de la bonne mise en correspondance en utilisant la fonction de score de l'équation (3.20).

Nous obtenons finalement deux informations : la mise en correspondance des plans entre les deux vues et la transformation rigide entre les deux poses. Cette dernière peut être éventuellement améliorée en utilisant une optimisation globale de la rotation et de la

translation dans l'équation (3.20).

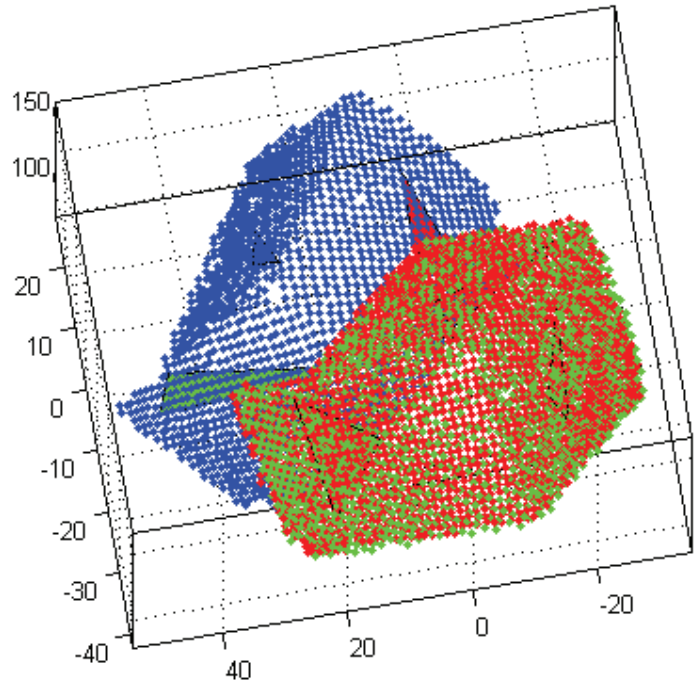


FIGURE 3.8 – Rotation et translation du 1er nuage de points (bleu) vers l'estimation de la deuxième pose (vert) en utilisant les résultats de l'alignement 3D basé plans. Le nuage de points réels observés est montré en rouge.

3.4 Evaluation

3.4.1 Objectifs et métriques

L'objectif de cette évaluation est de connaître la précision atteinte par les modèles d'étalonnage présentés et de les comparer avec les modèles de la littérature. Nous avons choisi deux références : une implémentation par M. Boucher [Boucher, 2014] d'une méthode inspirée de l'étalonnage stéréo similaire à la toolbox d'étalonnage de J.-Y. Bouguet sous Matlab [Hartley et Zisserman, 2000] [Xu et Zhang, 1996] et le service d'étalonnage en ligne du KIT implémenté par A. Geiger [Geiger *et al.*, 2012]. Dans cette comparaison, nous

avons choisi pour chaque méthode le jeu d'étalonnage donnant les meilleurs résultats sur leur métrique respective présentée dans le tableau 3.1.

Méthode	Métrique optimisée	Unité
M1	$\frac{1}{m \cdot n} \cdot \sum_j \sum_i \left\| \mathbf{n}_j . \tilde{\mathbf{T}} \mathbf{p}_{Cij} - d_j \right\|^2$ avec i=1..n (nb de points) j=1..m (nb de plans) $\mathbf{n}_j$ et d_j, paramètres des plans et $\mathbf{p}_{Cij}$ un point du plan dans $\mathcal{C}_j$	m
M2	$\mathbf{e} = \mathbf{s} - \mathbf{s}*$	px
M3	$\frac{1}{n} \cdot \sum_i \left\| {}^{\mathcal{C}_2}\mathbf{m}_{2i} - w({}^{\mathcal{C}_1}\mathbf{m}_{1i}, \tilde{\mathbf{T}}) \right\|^2$ avec i=1..n	px
Méthode stéréo[1]	$\frac{1}{m \cdot n} \cdot \sum_i \sum_j \left\| {}^{\mathcal{C}_j}\mathbf{m}_{ji} - p({}^{\mathcal{D}_j}\mathbf{m}_{ji}, \tilde{\mathbf{T}}) \right\|$ avec i=1..n (nb de points) j=1..m (nb de plans) et p une fonction projective	px
Méthode de Geiger[2]	$\frac{1}{n} \cdot \sum_i \left\| \mathbf{p}_{ci} - \tilde{\mathbf{T}} \mathbf{p}_{di} \right\|^2$ avec i=1..n (nb de points) $\mathbf{p}_c$ et $\mathbf{p}_d$ respectivement centres de gravité des plans observés par la caméra couleur et 3D	px

TABLE 3.1 – Métrique d'optimisation dans les différentes méthodes comparées.

Pour rendre cette évaluation plus objective, nous choisissons de ne pas comparer nos méthodes avec celles de la littérature en utilisant les métriques utilisées dans les optimisations de nos méthodes. La métrique utilisée dans la comparaison sera donc celle utilisée dans l'étalonnage stéréo et les jeux de tests seront les jeux correspondant à cette méthode (différents des jeux d'étalonnage) :

$$\frac{1}{m \cdot n} \cdot \sum_i \sum_j \left\| {}^{\mathcal{C}_j}\mathbf{m}_{ji} - p({}^{\mathcal{D}_j}\mathbf{m}_{ji}, \tilde{\mathbf{T}}) \right\|$$

avec $i = 1..n$, le nombre de points, $j = 1..m$, le nombre de plans et p une fonction de projection des points du plan observé par la caméra 3D vers la caméra conventionnelle.

1. Implémentation basée sur [Hartley et Zisserman, 2000] [Xu et Zhang, 1996] par M. Boucher
2. Service web d'étalonnage selon [Geiger *et al.*, 2012] mis en place par A. Geiger

Dans la suite du chapitre, nous présenterons tout d'abord les résultats de chaque méthode indépendamment puis nous effectuerons une comparaison avec les méthodes de référence.

3.4.2 Protocole expérimental

Nous souhaitons évaluer trois méthodes dont les processus sont très différents. Nous avons donc trois protocoles :

— Pour la méthode M1, nous avons réalisé un jeu d'étalonnage composé des images couleur et de profondeur de 33 poses d'un plan étalon traditionnel de type damier. Les coordonnées 3D des coins des cases du damier sont extraites automatiquement à partir des algorithmes classiques d'extraction de damier (implémentation d'*OpenCV*). Les plans sont obtenus à partir d'une modélisation par RANSAC de la zone d'intérêt à partir des images de profondeur. Les résultats ont été générés en validation croisée à partir de nombreux sous-ensembles du jeu d'étalonnage.

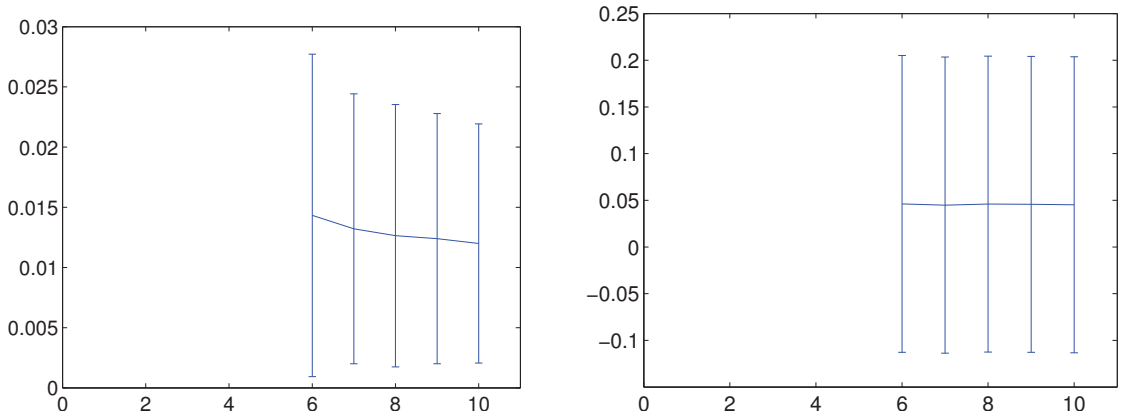
— Pour la méthode M2, nous avons réalisé 5 jeux d'étalonnage composés des images couleur et de profondeur d'une scène constituée de la mire 3D présentée en section 3.3.2. Les coordonnées 3D des centres sont extraites automatiquement à partir des images de profondeur aux contrastes réhaussés et des images en couleurs.

— Pour la méthode M3, nous avons réalisé 25 jeux de calibration avec une translation pure comme unique mouvement entre les deux poses pour étudier quantitativement l'influence de la transformation sur la précision de l'étalonnage et un jeu avec un mouvement complexe pour valider la possibilité d'extrapoler les résultats en translation pure pour n'importe quelle transformation. Les translations pures sont toutes réalisées approximativement le long de l'axe parallèle au capteur et horizontal (à l'aide d'un banc de test). Leur norme varie de 10 à 50 cm tous les 10 cm. Nous avons aussi fait varier la distance du capteur aux objets de la scène d'environ 1m, 2m et 3m. S'agissant du mouvement complexe, nous avons réalisé le jeu d'étalonnage avec un mouvement de la paire de caméras 3D et couleur à main levée.

3.4.3 Résultats

Résultats pour M1

La figure 3.9 montre l'influence du nombre de plans observés sur l'erreur d'estimation de la translation et de la rotation. On observe que l'erreur en translation diminue progressivement avec l'augmentation du nombre de plans (c'est aussi le cas pour l'écart-type de cette erreur). Au contraire pour la rotation, on n'observe pas d'amélioration notable ni pour l'erreur de rotation ni pour son écart-type en augmentant le nombre de plans. Comme nous l'avions montré dans la figure 1.11 du chapitre 1, ce type de méthode basé sur l'appartenance à un plan donne rapidement une très bonne estimation de la rotation. En effet, le résultat de l'optimisation non linéaire de la translation seule ou de la translation et de la rotation conjointement n'est pas significativement différent. Ainsi, alors que l'estimation de la translation est raffinée en augmentant le nombre de plans, l'estimation de la rotation reste stable.



(a) Erreur en translation (en m)

(b) Erreur en rotation (en 10^{-1} rad)

FIGURE 3.9 – Erreurs de la méthode M1 en fonction du nombre de plans utilisés

Comme nous pouvons le constater sur la figure 3.10 présentant le nuage de points issu de la caméra 3D en rouge et les points issus de la caméra conventionnelle (les coins du damier) et reprojetés dans le repère de la caméra 3D en bleu, il existe une superposition visuellement acceptable des deux informations à l'issue de l'étalonnage. On observe que certains points

bleus n'apparaissent pas. Ils sont en fait occultés par des points rouges issus de la caméra 3D. En effet, à cause des imprécisions de la caméra 3D, les surfaces planaires ne sont pas lisses et présentent donc une épaisseur dans laquelle se perdent certains points du damier pourtant correctement localisés.

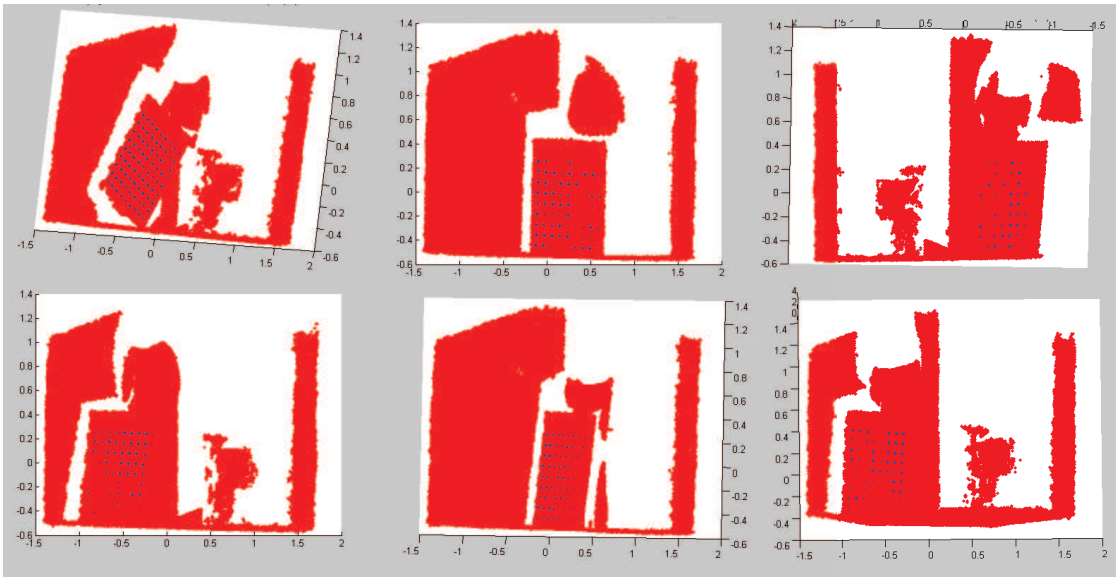
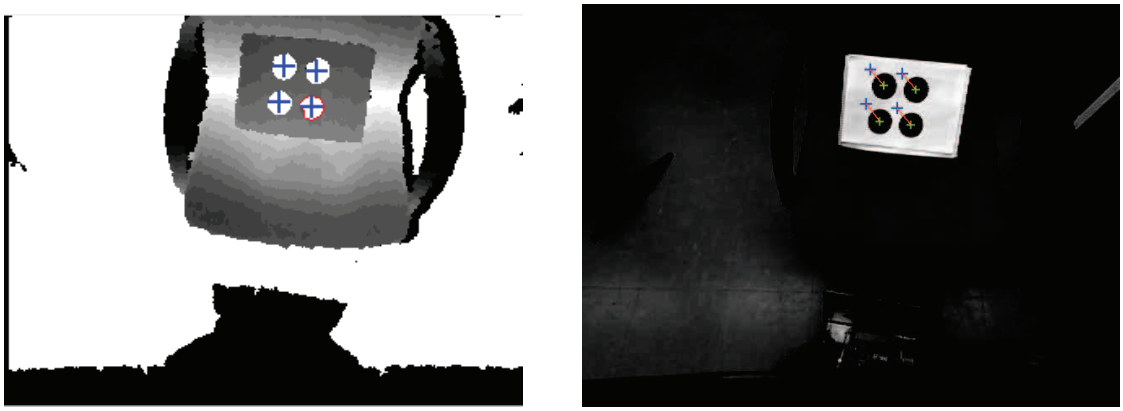


FIGURE 3.10 – Reprojection des coins du damier dans les données issues de la caméra 3D

Résultats pour M2

La figure 3.11 montre un exemple d'étalonnage par asservissement visuel virtuel entre la caméra de profondeur et la caméra conventionnelle. Les croix bleues matérialisent l'emplacement des points caractéristiques détectés dans l'image de profondeur et transformés dans le repère de la caméra conventionnelle avant étalonnage dans la figure 3.11b. Les croix vertes représentent la position des points caractéristiques détectés dans l'image conventionnelle et donc le motif désiré après convergence. Les courbes rouges montrent la trajectoire des points caractéristiques issus de l'image de profondeur dans l'image conventionnelle au cours de l'asservissement visuel. Une vue rapprochée de la mire d'étalonnage dans l'image conventionnelle 3.11b est présentée en figure 3.12. Les images ont été simplement corrigées pour réhausser les contrastes dans l'image conventionnelle et dans l'image de profondeur

pour faciliter la détection automatique des disques correspondant aux trous de la mire d'étalonnage de type *damier 3D*.



(a) Image de profondeur (b) Image conventionnelle

FIGURE 3.11 – Exemple d'asservissement visuel virtuel entre une image de profondeur et une image conventionnelle

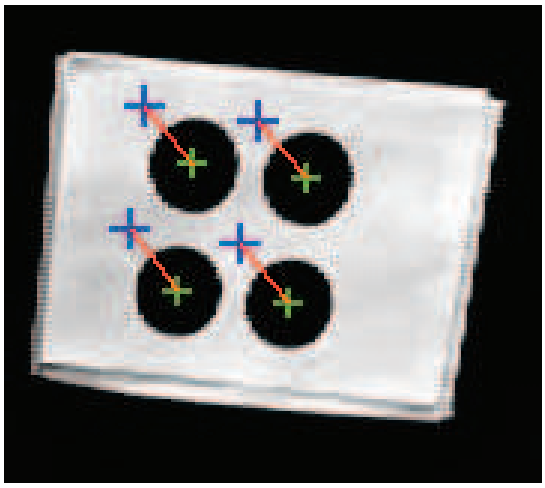


FIGURE 3.12 – Vue rapprochée de la mire d'étalonnage (caméra conventionnelle)

Le graphique 3.13 reprend la trajectoire de ces points en décrivant l'évolution de l'erreur de localisation des deux coordonnées (x, y) de chacun des 4 points caractéristiques au cours du temps. Au-dessous sont représentés les 6 éléments du vecteur vitesse de la caméra virtuelle lors de l'asservissement visuel. On observe bien une décroissance exponentielle et une convergence vers 0 à la fois de l'erreur des points caractéristiques et de la vitesse de la caméra.

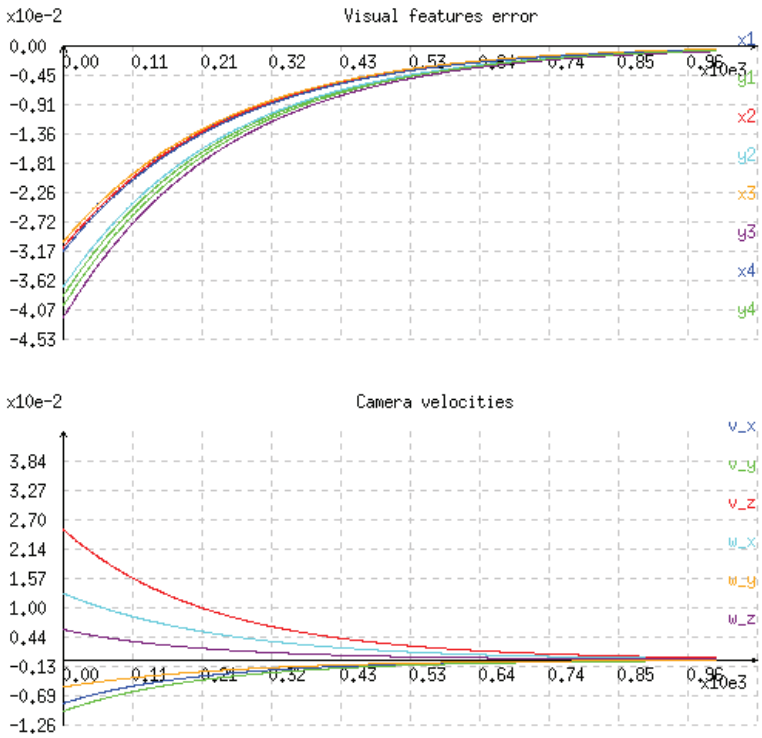


FIGURE 3.13 – Convergence des erreurs (en px) et des vitesses (en m/s et rad/s) vers 0

Résultats pour M3

Lorsque le processus d'étalonnage est terminé, nous recueillons trois types de résultats. En premier lieu nous obtenons un nouveau nuage de points 3D à partir de la fusion des deux nuages de points en entrée. Le nuage de points à la pose 1 est transformé vers le repère 2 et concaténé avec le nuage de points à la pose 2. La figure 3.14 montre un tel nuage de points final à partir de deux nuages de points captés par la caméra 3D à deux poses différentes. On observe (notamment sur le sol au premier plan à droite et à gauche du robot) une zone globalement plus claire. C'est une portion du nuage de points où la densité des points est plus importante du fait du recouvrement des deux nuages de points. Il s'agit donc du champ de vision commun aux deux images de profondeur issues de la caméra 3D aux deux poses. Nous obtenons aussi la liste des points reprojetés ainsi que les erreurs associées et l'image représentant les points d'intérêt (en blanc) et les points reprojetés (en noir) comme sur la figure 3.15. On observe que la totalité des points noirs

(points reprojetés) sont localisés très proches des points blancs (points détectés). Quelques points blancs ne sont pas associés à des points noirs notamment sur la bordure noire du panneau vertical derrière le robot (au centre à gauche). Cela est dû à l'élimination des points d'intérêt en raison d'un manque de mesure de profondeur (fréquent sur les arêtes saillantes des objets). Finalement, nous obtenons aussi les paramètres correspondant aux trois éléments du vecteur de rotation et aux trois éléments du vecteur de translation.

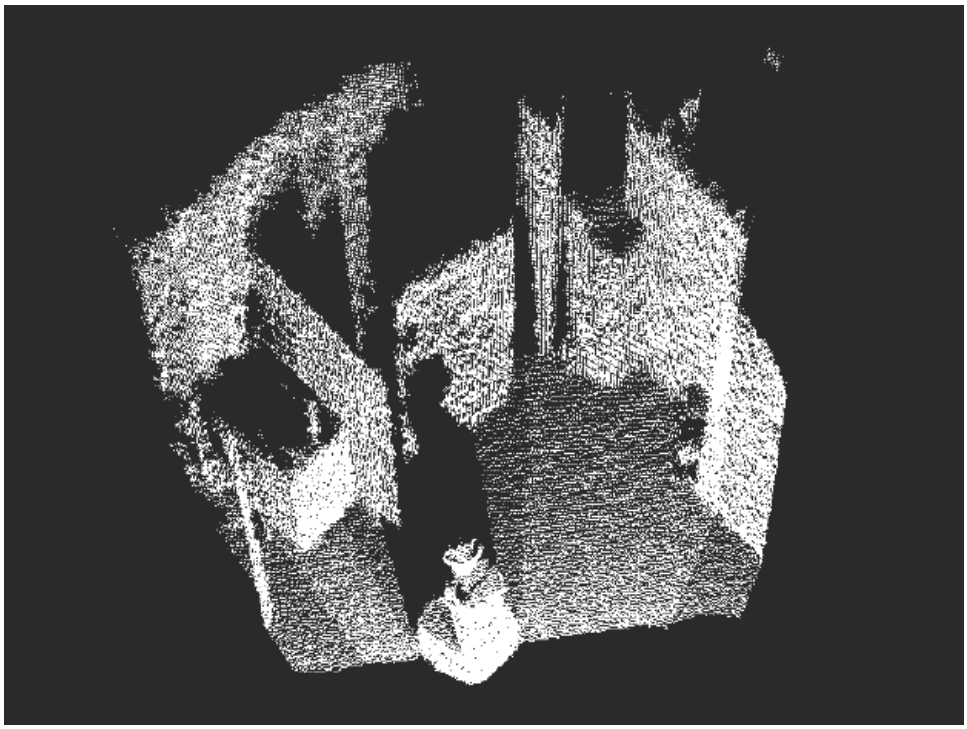


FIGURE 3.14 – Fusion des nuages de points 3D après étalonnage

Etant donné qu'il est très difficile de reproduire précisément le même mouvement avec notre robot mobile, les résultats que nous présentons ci-après ont utilisé un mouvement composé uniquement d'une translation pure. Les mouvements plus complexes ont aussi été testés mais il ne nous a pas été possible de produire des données quantitatives précises. Nous avons réalisé des étalonnages et des tests avec mouvements complexes à partir des données similaires à celles présentées dans la figure 3.7. A cause du problème de répétabilité et de l'augmentation du nombre de paramètres à étudier dans le cadre des mouvements complexes, il est difficile de produire des graphiques statistiquement significatifs sans un



FIGURE 3.15 – Reprojection des points d'intérêt après étalonnage

banc d'étalonnage de haute précision. Au contraire, nous avons pu réaliser de telles analyses dans le cadre des mouvements de translation pure comme le montrent les figures 3.16 ou 3.17. Qualitativement, nous avons pu remarquer lors des tests basés sur les mouvements complexes que dans cette situation, l'étalonnage est plus sensible à la qualité de l'estimation de la transformation entre les deux poses de la caméra 3D. Malgré cette plus grande sensibilité, l'algorithme d'alignement 3D que nous avons proposé est suffisamment précis pour permettre un étalonnage extrinsèque performant, menant à une erreur de reprojection contenue entre 1 à 2.5 px selon le jeu d'étalonnage et le jeu de données de tests.

S'agissant donc de l'étalonnage basé sur une translation pure, le premier paramètre intéressant à analyser est l'influence du nombre de points et de la qualité de la localisation des points utilisés pour minimiser la fonction de coût r. Lors de l'appariement il est possible de faire varier le seuil d'acceptation d'une paire (la distance dans l'espace du descripteur). Plus on abaisse ce seuil, plus le nombre de paires sera faible mais meilleure sera la confiance dans la qualité de l'appariement. Au contraire, plus ce seuil augmente plus le nombre de

paires appariées sera important mais plus la probabilité d'accepter de mauvais appariements augmente. Dans la figure 3.16, nous montrons l'erreur de reprojection moyenne (et son écart-type) sur une échelle logarithmique en fonction du nombre de paires utilisées dans la minimisation.

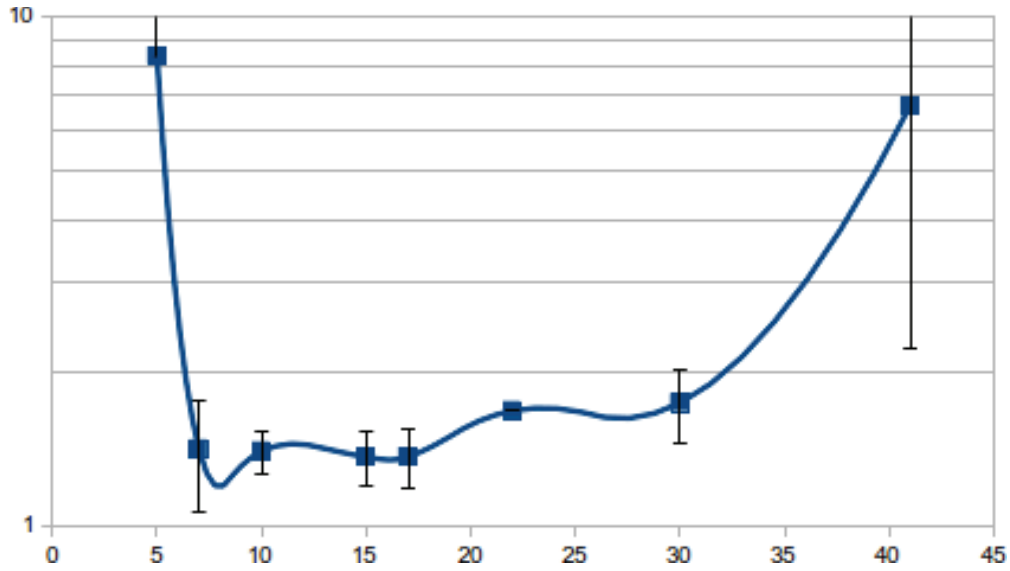


FIGURE 3.16 – Erreur moyenne de reprojection en pixels (échelle logarithmique) en fonction du nombre de paires utilisées dans la minimisation

Entre 5 à 10 paires, on observe une forte diminution de l'erreur moyenne ainsi qu'une contraction de la variance de l'erreur. Entre 10 et 17 paires, l'erreur de reprojection reste stable mais finalement toute augmentation du nombre de paires conduira à une augmentation de l'erreur moyenne de reprojection. En effet, nous avons observé que les premiers faux appariements apparaissent à partir de la 19ème paire de points et deviennent de plus en plus nombreux et augmentent de fait à la fois la moyenne et l'écart-type de l'erreur de reprojection. Lorsque le seuil utilisé pour accepter ou non l'appariement est faible seuls les appariements avec une forte confiance sont utilisés dans la fonction de minimisation. Si on augmente le nombre de paires, l'erreur de reprojection augmente. Ce phènomène est naturel puisque dans l'appariement basé sur RANSAC, on utilise aussi une minimisation dont la fonction de coût est définie dans l'espace image afin d'estimer les paramètres de mouvement contenus dans la matrice essentielle ou l'homographie. Selon nos expériences,

le minimum de la courbe de la figure 3.16 dépend de la scène observée mais la forme de la courbe conserve toujours la même apparence.

Pour analyser maintenant l'effet de l'imprécision d'estimation de déplacement de la caméra 3D (noté $\mathbf{T}'$ dans le modèle), nous représentons dans la figure 3.17 la variation de l'erreur de reprojection en pixels lorsqu'on introduit artificiellement une erreur variant de -5 à +4 cm dans la translation réellement effectuée. L'ensemble des méthodes automatiques modernes d'alignement 3D, tout comme la mesure manuelle, produit des erreurs inférieures au centimètre pour une plage de déplacement permettant le recouvrement partiel de vues entre deux poses. Il ne nous a donc pas paru nécessaire d'analyser un intervalle plus important.

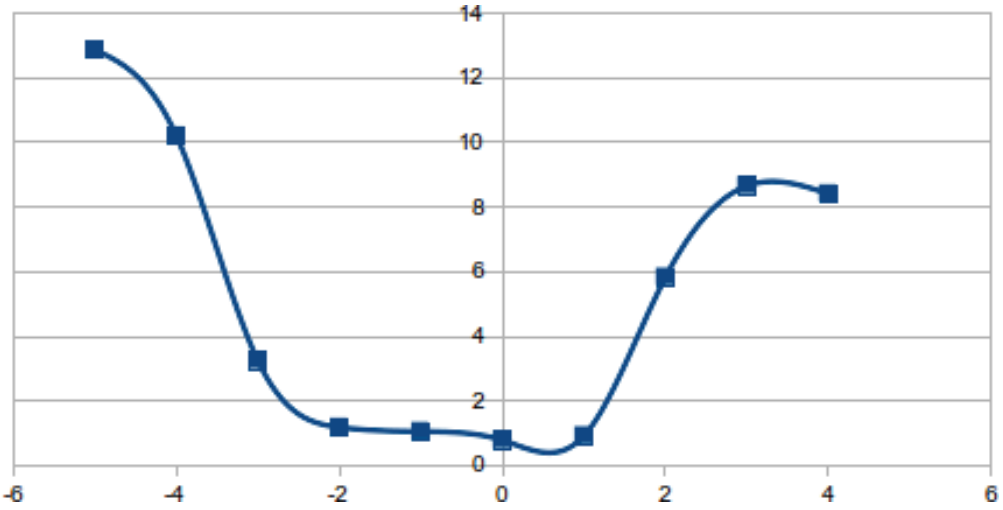


FIGURE 3.17 – Effet de l'erreur d'estimation de $\mathbf{T}'$ (en cm) sur l'erreur de reprojection (en px) en translation pure.

Comme nous pouvons nous y attendre, l'erreur de reprojection est plus faible quand l'estimation du déplacement $\mathbf{T}'$ est meilleure. Au contraire, lorsque l'erreur dépasse 1 à 2 cm, l'erreur de reprojection augmente fortement. Cette figure nous permet de conclure en soulignant l'importance d'une bonne estimation de $\mathbf{T}'$ pour obtenir un étalonnage précis. L'explosion rapide de l'erreur laisse supposer qu'un faible décalage de la transformation peut mener à des minima locaux assez éloignés de l'étalonnage optimum.

Afin de déterminer la transformation idéale entre les deux poses, il est aussi intéres-

sant de discuter l'influence de la norme de $\mathbf{T}'$ sur la précision de l'étalonnage. La figure 3.18 représente l'erreur moyenne de reprojection en fonction de la norme de la translation effectuée entre les deux poses. La droite de tendance est dessinée en orange. On observe que cette ligne est légèrement décroissante mais étant donné l'écart-type important pour la mesure à 10 cm, on ne peut conclure sur la significativité de cette tendance. De même l'écart-type semble augmenter au-delà de 20 cm. Il paraît clair qu'une translation de 10 cm n'est pas adaptée pour un étalonnage sur une scène quelconque alors que les normes de 20, 30 et 40 cm semblent convenir mieux. A partir de 50 cm, nous avons observé un grand nombre de points d'intérêt non contenus dans la zone observée commune aux deux poses. De fait, l'écart-type de l'erreur augmente (le nombre d'appariements utilisables se réduisant en augmentant la norme de la translation).

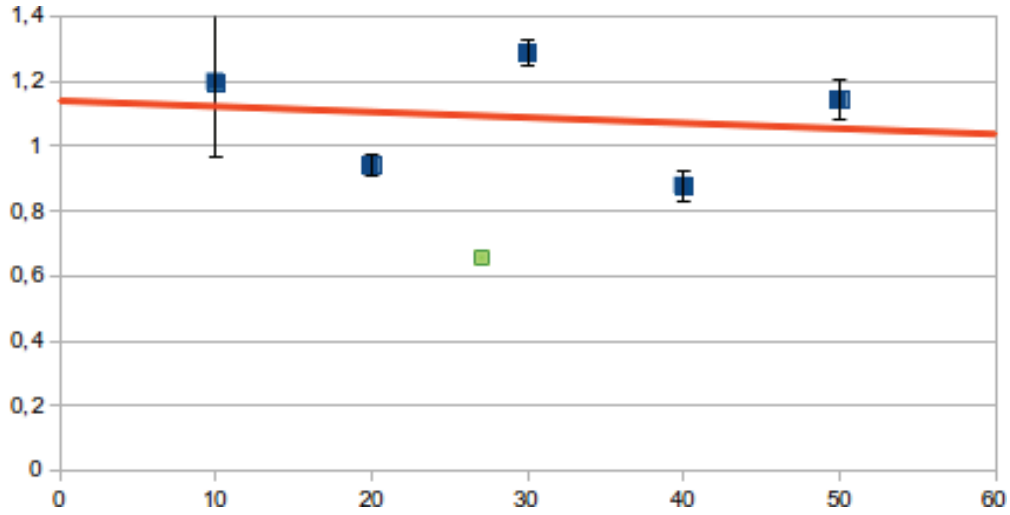


FIGURE 3.18 – Effet de la norme de la translation (en cm) sur l'erreur de reprojection (en px). La courbe orange représente la tendance affine.

La figure 3.19 montre l'histogramme des erreurs de 142 reprojections provenant de différentes scènes et diverses translations en utilisant à chaque fois 18 paires de points d'intérêt. Nous avons utilisé des classes de 0.2px de large. Par exemple, la barre 0,4 représente le nombre de paires (15) dont l'erreur de reprojection associée est située entre 0.2px et 0.4px. La courbe jaune montre le pourcentage cumulé pour chaque classe d'erreur.

La plupart des erreurs (85 %) sont supérieures à 0.2px et inférieures à 2px. La modé-

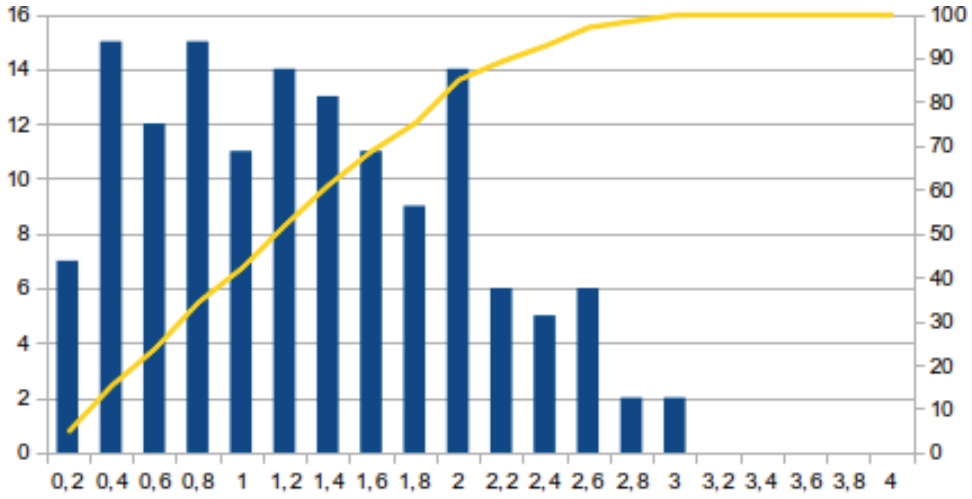


FIGURE 3.19 – Histogramme des erreurs de 142 reprojections (en bleu) et pourcentage cumulé (en jaune).

lisation par une loi normale donne un maximum situé aux alentours de la classe de 1px d'erreur.

Comparaison avec la littérature

Afin de comparer nos méthodes avec celles de la littérature, nous avons choisi une implémentation d'une méthode d'étalonnage par M. Boucher basée sur le paradigme de l'étalonnage stéréo tel que présenté dans [Hartley et Zisserman, 2000] et [Xu et Zhang, 1996] ou encore utilisé par J.-Y. Bouguet ou OpenCV dans leur bibliothèque d'étalonnage. Malgré son caractère fastidieux (nécessité d'une grande précision dans la construction de la mire 3D, processus d'étalonnage long, ...), cette méthode est la plus précise que nous ayons rencontrée. Nous avons aussi exploité la très pratique et récente toolbox de A. Geiger au travers du service disponible gratuitement en ligne auprès du KIT [Geiger *et al.*, 2012]. Pour chacune de ces méthodes nous avons choisi le jeu de données d'étalonnage présentant la plus faible erreur de reprojection dans leur métrique propre présentées dans le tableau 3.1. Nous donnons les résultats de comparaisons dans le tableau 3.2 suivant où nous établissons une évaluation relative (de 0 à 3 étoiles, respectivement pas du tout et totalement) du caractère naturel de la scène utilisée et de l'automaticité des méthodes :

Méthode	Nb de poses	Moyenne	Ecart-type	Scène naturelle	Automaticité
M1	6	2.51	0.47	*	*
M1	33	2.46	0.75	*	*
M2	1	2.49	1.32	*	***
M3	2	0.92	0.05	***	**
Stereo[1]	6	0.83	0.24	*	*
Geiger[2]	1 (6 plans)	2.12	0.14		***

TABLE 3.2 – Comparaison globale des méthodes d'étalonnage. L'erreur moyenne et l'écart-type sont présentés en pixels en utilisant la métrique de la méthode stéréo.

Globalement nous pouvons noter que la méthode M3 est aussi précise que l'approche stéréo (la différence n'est pas significative) alors qu'elle permet d'éviter l'utilisation d'un motif d'étalonnage complexe à réaliser et qu'elle est beaucoup plus automatisable. La méthode de Geiger pourtant basée sur un motif de type damier est elle significativement moins performante que notre méthode basée sur scène naturelle. La méthode M3 utilise seulement deux vues pour atteindre ce niveau de performances alors que la méthode stéréo en nécessite au minimum 6 pour être stable. De la même façon, la méthode de Geiger ne nécessite qu'une unique vue mais cette vue elle-même nécessite de disposer de 6 plans étalons ce qui ne facilite pas le processus d'étalonnage. La méthode M1 est clairement moins performante que l'ensemble des autres méthodes présentées ici que ce soit en utilisant 6 vues ou 33. On explique aisément ce manque de performance au regard de la métrique pixellique puisque la minimisation n'est pas réalisée dans l'espace image mais dans l'espace métrique et que le nombre de contraintes utilisées par vue est très inférieur à celui des autres méthodes. Enfin, la méthode M2 présente un niveau de performance moyen comparable à celui de M1 mais avec un écart-type beaucoup plus important qui provient de la grande influence de la scène d'étalonnage et de test sur l'erreur moyenne de reprojection. L'utilisation d'une unique prise de vue induit de plus une répartition irrégulière de l'erreur selon les différents degrés de liberté, répartition dépendante de l'orientation et de la position de la mire dans la scène.

1. Implémentation basée sur [Hartley et Zisserman, 2000] [Xu et Zhang, 1996] par M. Boucher
2. Service web d'étalonnage selon [Geiger *et al.*, 2012] mis en place par A. Geiger

3.5 Conclusion

Dans ce chapitre nous avons proposé trois méthodes d'étalonnage extrinsèque entre une caméra 3D et une caméra conventionnelle. Dans une première méthode, nous avons étendu la méthode d'étalonnage laser-caméra 3D présentée dans le chapitre 1 au cas de l'étalonnage caméra 3D-caméra conventionnelle. Nous avons montré que ce type d'étalonnage (pouvant être réalisé conjointement avec l'étalonnage caméra 3D-laser) n'arrive pas à la précision de reprojection de l'état de l'art et présente l'inconvénient de reposer sur un processus d'étalonnage traditionnel complexe basé sur l'observation d'un grand nombre de scènes présentant un plan étalon (damier) à diverses poses.

Nous avons alors proposé deux autres méthodes facilitant le processus d'étalonnage tout en améliorant sa qualité. La première repose sur une mire 3D dont certaines caractéristiques sont communes aux images des deux caméras 3D et conventionnelle. Cette méthode est complètement automatique et le prototype d'implémentation est quasiment temps réel alors même qu'aucune optimisation n'a été réalisée. Un étalonnage temps réel, même approximatif, peut-être nécessaire dans le cadre d'application où la caméra couleur et/ou la caméra 3D sont mobiles (sur une tourelle pan/tilt par exemple).

La dernière méthode proposée reprend le principe de simplification du processus d'étalonnage mais se concentre sur l'utilisation de scènes naturelles texturées. Nous avons montré que la précision atteinte par notre méthode rivalise avec les meilleures méthodes de l'état de l'art tout en faisant abstraction de la nécessité de disposer d'une mire étalon de type damier ou similaire. De plus, cette méthode est automatique là où les autres méthodes naturelles de la littérature nécessitent de faire réaliser l'appariement 2D/3D manuellement par l'utilisateur ou d'utiliser un capteur supplémentaire (par exemple une centrale inertielle) pour estimer le mouvement des capteurs.

Si la première méthode n'offre pas de perspective intéressante, les deux dernières méthodes présentent des intérêts différents qui peuvent être améliorés chacun. On distinguera principalement trois axes : le temps de calcul, l'extension à l'étalonnage intrinsèque et la

robustesse des points caractéristiques utilisés. Nous l'avons déjà dit, la méthode M2 est assez rapide, un petit travail d'optimisation de l'implémentation permettrait un étalonnage temps réel et l'utilisation conjointe d'une caméra 3D et d'une caméra conventionnelle mobiles dès lors que l'étalon 3D est visible pour les deux capteurs. Nos algorithmes partent de l'hypothèse que les paramètres intrinsèques de la caméra 3D (et de la caméra conventionnelle) sont connus et de bonne qualité. Nous avons montré que les caméras 3D à bas coût souffrent d'une distorsion de disparité qui a une influence importante sur les données de profondeur au delà de 2-3m. Il serait intéressant de formuler un modèle paramétrique de ces distorsions afin de pouvoir conjointement estimer les paramètres extrinsèques et les paramètres intrinsèques. Enfin, nous avons montré que nos méthodes sont sensibles à la robustesse des points caractéristiques utilisés. Que ce soit pour leur localisation pour M2 et M3 (mais c'est aussi le cas pour les méthodes de la littérature basées mire d'étalonnage) ou pour leur appariement pour M3, il est nécessaire de pouvoir compter sur des points caractéristiques robustes puisque cela influe directement sur la qualité de l'étalonnage.

Dans le chapitre suivant 4, nous verrons dans une interface graphique 3D en réalité augmentée, l'intérêt d'un tel étalonnage caméra 3D - caméra conventionnelle pour une application de téléopération d'un robot mobile en milieu intérieur. Dans cette section 4.2, nous présenterons dans un premier temps, la théorie liée à la construction de ce type d'application puis nous montrerons l'intérêt de l'ajout de la caméra 3D dans une interface centrée sur le point de vue de la caméra conventionnelle du robot. Enfin, nous montrerons dans une dernière section 4.3, l'influence de la qualité de l'étalonnage entre la caméra 3D et la caméra conventionnelle dans une application de SLAM RGB-D issue de la littérature.

Chapitre 4

Application à la robotique téléopérée ou autonome

4.1 Introduction

De nombreuses applications comme le suivi d'objet, la segmentation, la classification ont montré l'intérêt des images RGB-D. Alors que certains problèmes en vision monoculaire sont réputés difficiles, l'ajout de l'information de profondeur permet de fiabiliser et d'améliorer les performances applicatives. Généralement, ces applications se sont démocratisées avec l'apparition des capteurs RGB-D à bas coût comme la Kinect par exemple. Mais il n'est pas forcément nécessaire que le capteur 3D et la caméra conventionnelle aient été intégrés dans un même boitier. Il est possible de créer des images RGB-D en tout point comparables avec celles produites par les capteurs RGB-D à partir d'un capteur 3D (caméra 3D, télémètre 3D, radar, paire stéréo, ...) et d'une caméra conventionnelle indépendante (généralement de meilleure qualité que celles intégrées dans les capteurs RGB-D à bas coût). Il faut pour cela utiliser les paramètres intrinsèques des capteurs et un étalonnage extrinsèque comme ceux proposés au chapitre précédent. Dans ce dernier chapitre, nous nous intéresserons à deux types d'applications très différentes pour montrer l'intérêt de l'étalonnage extrinsèque entre une caméra conventionnelle et une caméra 3D (indépen-

dantes ou non).

La première application concerne la téléopération d'un robot mobile à domicile. Le robot en question doit permettre à un centre de télévigilance du SAMU, lorsqu'une alerte est émise, de pouvoir chercher la personne en détresse dans son appartement. Le robot, une fois arrivé près de la personne, établit une liaison bidirectionnelle entre la personne en détresse et le médecin régulateur du SAMU qui va émettre un pré-diagnostic permettant d'adapter le niveau de la réponse (ne rien faire en cas de fausse alarme, faire appel à un voisin/parrain, dépêcher les pompiers et/ou le SAMU en cas d'urgence). Pour ce faire, les opérateurs du centre de télévigilance doivent pouvoir contrôler les déplacements du robot via une interface de contrôle du robot et de visualisation de la situation. Initialement développée par le laboratoire IBISC en partenariat avec la start-up Droïds Company, l'interface de contrôle associée au robot Lina présentait simplement à l'opérateur le flux vidéo capté par la caméra du robot. Cette information vidéo était affichée entourée de blocs colorés en rouge, en orange ou en vert selon la proximité des obstacles latéraux (distance mesurée à partir d'un télémètre laser à balayage ou d'une ceinture de télémètres à ultrason). Si ce type de retour d'information et d'interaction avec l'utilisateur était appréciable, les expérimentations avec les opérateurs et les médecins du SAMU ont soulevé la difficulté de navigation avec le robot pour un opérateur qui n'est pas totalement concentré sur l'interface (consultation de dossier, multiples écrans et interfaces, ...) surtout dans un environnement inconnu comme peut l'être le domicile d'un patient et dans l'urgence. L'équipe a recruté en 2010-2012 un post doctorant, Bruno Mantel, qui a particulièrement travaillé sur une refonte de l'interface de téléopération avec une méthodologie dite *écologique*. Nous verrons plus précisément en section 4.2 en quoi consiste cette approche appliquée à la conception d'interface homme-machine.

La deuxième application de l'étalonnage extrinsèque consiste en une localisation et une cartographie simultanées (SLAM) basées sur une paire de caméras 3D et conventionnelle.

4.2 Application à la téléopération d'un robot en RA

4.2.1 Modèle de l'existant

L'approche écologique de la perception, initialement proposée par le psychologue James J. Gibson dans les années 70, part d'un constat simple : on ne peut expliquer et décrire les mécanismes de perception qu'en mettant de côté la question du comment pour s'attarder sur le problème du quoi. Selon Gibson, la description du fonctionnement des organes de perception ou de l'objet à percevoir (le comment) ne donne pas d'information sur la perception. Il n'existe de perception que dans une relation entre un agent et son environnement (d'où le nom d'approche écologique). Si on met de côté les questions *Comment perçoit-on ?* ou encore *Quels sont les processus sous-jacents au sein de l'organisme ?*, il reste deux questions principales liées aux notions de contenu et de forme de la perception.

— Qu'est-ce qui est perçu ?

— Quelles informations à propos de l'environnement peut-on prélever par notre perception ?

S'agissant du contenu, il convient non pas de s'intéresser aux propriétés des objets (comme par exemple la largeur d'une porte) mais à leurs perspectives fonctionnelles et aux opportunités d'action (la porte est-elle franchissable ?). Ces perspectives fonctionnelles sont intrinsèquement relatives à l'agent (une porte peut être franchissable ou non selon la corpulence, la mobilité ou l'autonomie de l'agent) et doivent donc être comprises comme des relations entre les propriétés de l'objet et les propriétés de l'agent.

La théorie de Gibson décrit ces relations avec la notion d'affordance qu'il définit comme les propriétés relationnelles du système agent-environnement. Une affordance correspond à une action (possible, efficiente ou confortable) réelle (qui concorde avec un comportement habituel observable) caractérisable (contraintes géométriques, dynamiques, ...) qu'elle soit perçue ou non. Il s'agit du contenu de l'information disponible.

Cette information disponible possède aussi une forme que Gibson appelle *structure des énergies ambiantes* et qui correspond globalement aux différentes capacités sensorielles :

— L'énergie optique

— L'énergie acoustique

— L'énergie haptique

— ...

Reprenons l'exemple de la porte. Franchir la porte est une action possible. C'est donc une affordance liant l'agent et la porte dans l'environnement. La question de savoir si l'agent peut effectivement franchir la porte ou pas n'est résolue qu'en ajoutant un critère de passabilité. Habituellement, on compare simplement la largeur de l'agent avec la largeur de la porte. Dans l'approche écologique, on propose une mesure inspirée de la perception humaine qui ne dépend que de la configuration de l'agent et sa perception visuelle (de l'énergie optique). Dans la figure 4.1, les équations montrent comment il est possible pour un agent de savoir s'il peut franchir une porte sans mesurer sa largeur ni celle de la porte. Le rapport entre la largeur de l'ouverture et celle des épaules peut s'exprimer sous la forme d'un rapport d'angles α et β issu uniquement de la perception visuelle multiplié par une constante apprise au cours du temps par la personne. Nous pouvons ainsi voir que l'information de passabilité peut s'exprimer sous une forme qui ne dépend pas des propriétés de l'ouverture (la largeur par exemple) mais uniquement de données de perception.

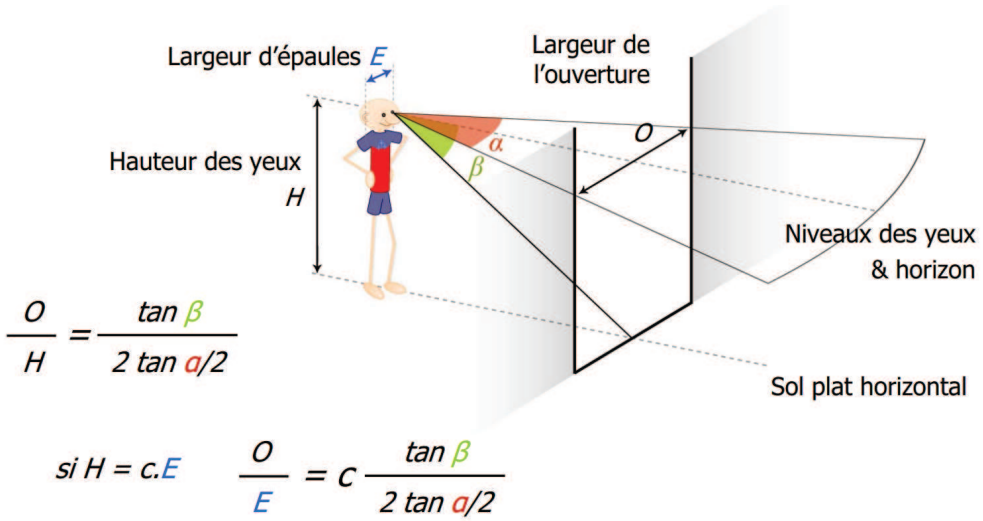


FIGURE 4.1 – Spécification de la passabilité dans l'approche écologique (B. Mantel).

Un autre exemple, le temps de pré-contact (TTC - *Time-To-Contact*) est souvent utilisé en robotique pour connaître la dangerosité d'un obstacle. Alors qu'une approche naïve nécessite de connaître la distance de l'obstacle ainsi que sa vitesse relative, l'approche écologique propose de ne s'intéresser qu'à la taille perçue des objets [Alenya *et al.*, 2009]. Sur la figure 4.2, les équations montrent qu'il est possible de calculer le temps avant impact seulement à partir de la taille perçue de l'objet t et de sa dérivée e.

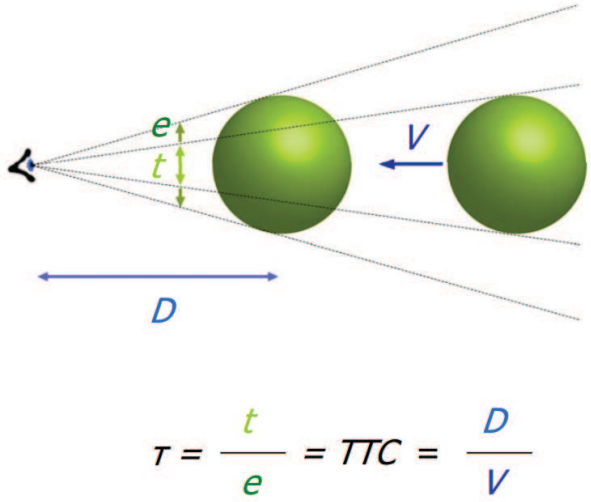


FIGURE 4.2 – Spécification du TTC dans l'approche écologique (B. Mantel).

Lorsqu'on s'intéresse aux interfaces homme-machine (IHM) et plus particulièrement aux interfaces de téléopération pour la robotique, on étudie un cas particulier de la relation entre l'agent et l'environnement. Il s'agit en fait d'un système agent-environnement dans lequel on intercale une interface et un robot. Les conséquences se résument en une altération des affordances, des règles de contrôle et de la structure de la stimulation.

Concevoir une IHM dans l'approche écologique, c'est mettre de côté les questions *Quels capteurs ? Quels périphériques ? Comment présenter les données ?* pour ne s'intéresser d'abord qu'à la perspective fonctionnelle : *Quelle est la finalité de l'interface de téléopération ?* [Flach *et al.*, 1998]

B. Mantel, dans [Mantel *et al.*, 2012], propose une démarche écologique de conception d'IHM qui peut être résumée à la figure 4.3 permettant de compenser les altérations dues à

l'insertion du système interface-robot entre l'agent et l'environnement. Les aménagements de l'interface de téléopération suivent quatre pistes : l'anthropomorphisme (concevoir l'interface et le robot de telle sorte que leurs propriétés correspondent à celles de l'agent), l'augmentation de l'information (ajouter un complément d'information généralement sous forme synthétique), l'augmentation de l'action (proposer des actions qui permettent à l'agent de récupérer de l'information) et enfin la délégation. Cette dernière consiste à laisser le robot se charger des actions pour lesquelles l'agent n'a pas assez d'information en utilisant ses propres capacités d'autonomie. L'assistance à la navigation est l'exemple le plus évident de délégation en téléopération : l'opérateur s'occupe de déterminer un déplacement et le robot se charge de planifier et de suivre une trajectoire réalisant au mieux ce déplacement tout en évitant les obstacles non prévus par l'opérateur [Devaux *et al.*, 2011].

A partir de ces principes il est possible de concevoir itérativement une interface de téléopération pour robot mobile. L'essentiel de la perception humaine reposant sur la vue, le point de départ de cette interface consiste simplement en l'affichage de l'image de la caméra conventionnelle équipant le robot (figure 4.4a).

L'ajout d'anthropomorphisme se traduit par le déplacement de la caméra virtuelle à l'emplacement où l'opérateur aurait ses yeux (figure 4.4b). Le champ de la caméra étant plus limité que le champ de vision humain, l'image de la caméra occupe un espace plus limité correspondant au rapport entre le champ de vision de la caméra et celui d'un opérateur. Le corps du robot apparaît par l'intermédiaire du cylindre rouge en bas de l'écran. L'anthropomorphisme s'accompagne ainsi d'une conscience du corps de l'avatar robotique dans lequel se projette l'opérateur.

Les figures 4.4c et 4.4d montrent l'augmentation de l'information en affichant respectivement la position du sol par l'intermédiaire de la grille bleue (qui donne une information sur les distances entre les objets et leurs dimensions relatives) et la position des obstacles donnée par le télémètre laser par l'affichage des cylindres bleus par-dessus la scène (y compris par-dessus l'image de la caméra), ce qui permet d'augmenter le champ de vision restreint de la caméra. L'augmentation de l'information utilise les principes basiques de la

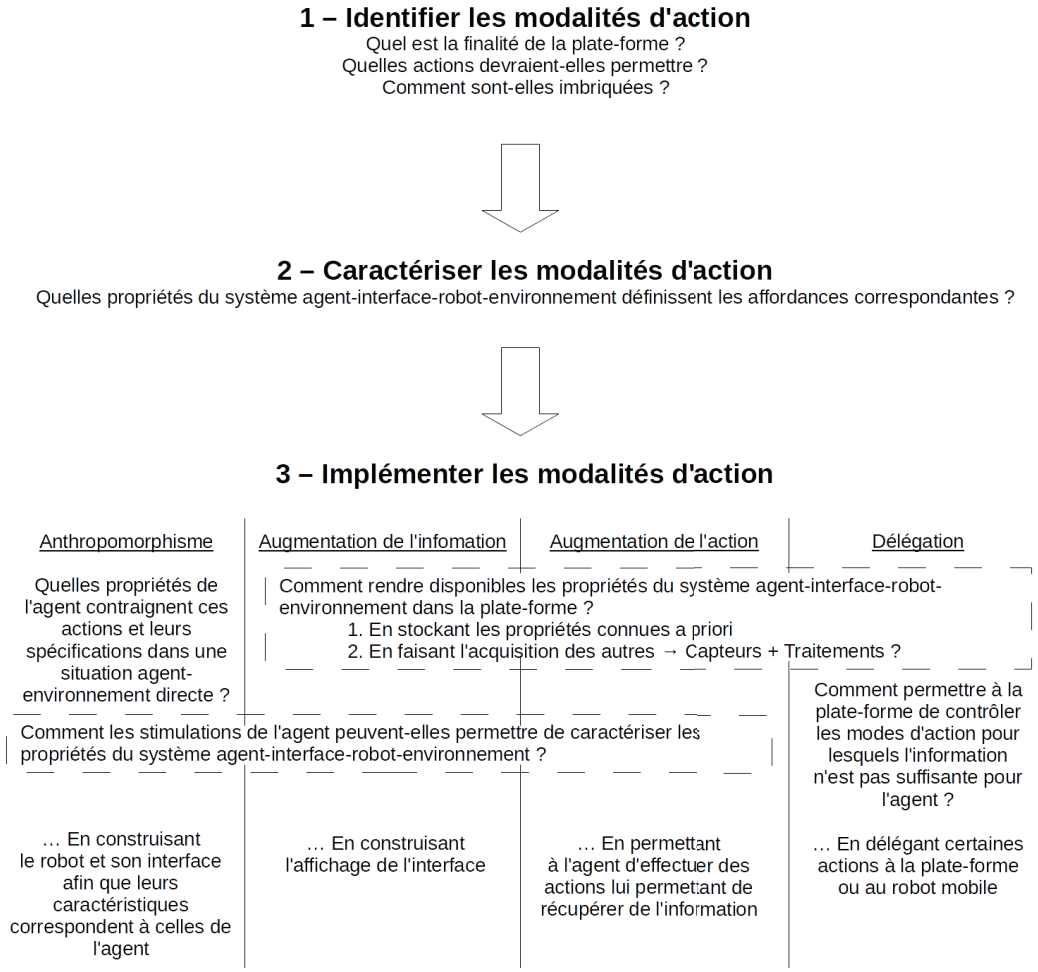


FIGURE 4.3 – Démarche de conception écologique d'une plate-forme de téléopération.

réalité augmentée, l'information pertinente pour l'opérateur étant de juger de la passabilité du robot par une ouverture ou entre les obstacles.

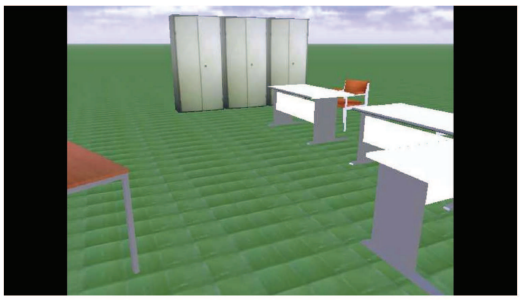
Pour faire le rendu de l'augmentation de l'action pour acquérir plus d'information, dans la figure 4.4e, on observe une modélisation de la trajectoire du robot après 1, 2, 3, 4 et 5 secondes par l'intermédiaire des disques verts placés parallèles au sol et dont la taille correspond à la circonférence du robot. L'utilisateur peut faire varier cette projection de trajectoire en déplaçant la souris. La modélisation suit les mouvements de l'opérateur. La trajectoire est calculée en supposant les ordres constants (des vitesses constantes impliquent une trajectoire circulaire). Il suffit alors de valider en cliquant pour que le robot suive la

consigne.

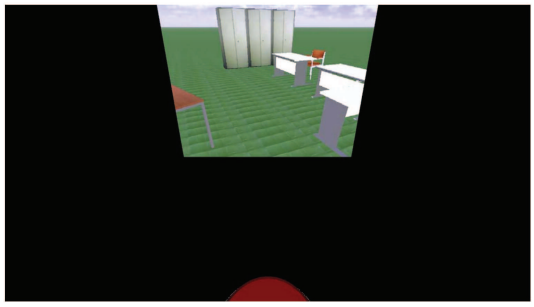
La figure 4.4f montre enfin comment l'utilisateur peut acquérir de l'information sur la passabilité en déplaçant cette trajectoire dans une zone où des obstacles sont présents. Il s'agit ici de se rendre compte de la possibilité de passer entre les bureaux. La projection de la trajectoire montre que la largeur est suffisante pour que le robot passe entre ces meubles (et que le robot sera sur place en 5 secondes, chaque disque vert représentant une position future après 1, 2, 3, 4 et 5 secondes). Enfin on observe aussi une flèche dont la blancheur donne une information sur la vitesse (plus la flèche est blanche plus le déplacement sera rapide) et dont la direction donne une information sur la trajectoire.

Finalement la succession de ces itérations de conception se termine sur une interface de téléopération qui permet à l'utilisateur d'avoir une vue de l'environnement, de percevoir les obstacles détectés par le télémètre laser, de se rendre compte de la taille de son avatar robotique, de déterminer la trajectoire et la vitesse du robot et surtour d'estimer simplement la passabilité entre les obstacles.

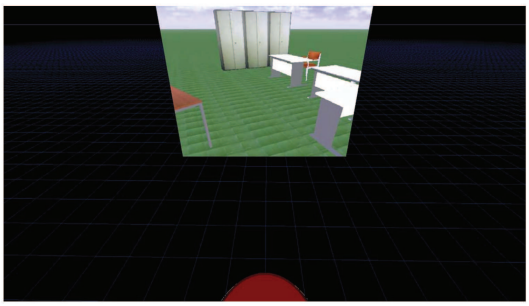
La figure 4.5 montre qu'il était effectivement possible de passer entre les bureaux. Les bureaux sortant du champ de vision de la caméra à mesure que le robot s'approche, on s'aperçoit bien de l'intérêt de l'augmentation par ajout des mesures du télémètre avec la présence des obstacles en bleu. Enfin, le passage étant étroit, l'opérateur a choisi de le franchir à vitesse réduite. Il peut consulter cette information de vitesse en observant la trajectoire verte ou la flèche blanche située près du corps du robot. Ces figures sont issues de simulations mais les avantages relevés sont les mêmes en situation réelle à ceci près que le laser seul, nous en avons déjà discuté, n'est pas toujours capable de percevoir les obstacles. En effet, en simulation le calcul des distances des obstacles utilisant la technique de la boîte englobante, les tables sont considérées comme des parallélépipèdes rectangles. En situation réelle, ce genre de mobilier n'est pas détecté par le télémètre laser seul. Pour cette dernière raison, nous allons proposer une extension de cette interface à l'usage des caméras 3D.



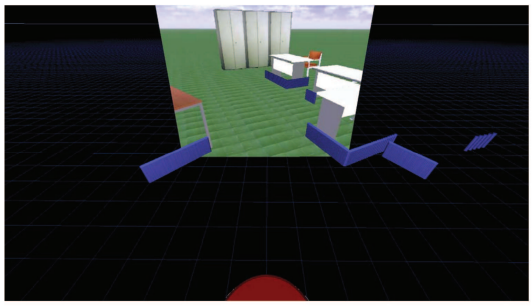
(a) Vue traditionnelle de l'image couleur



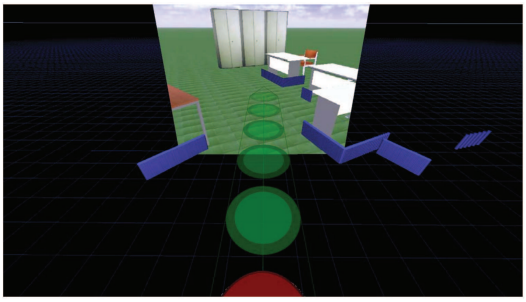
(b) + Anthropomorphisme



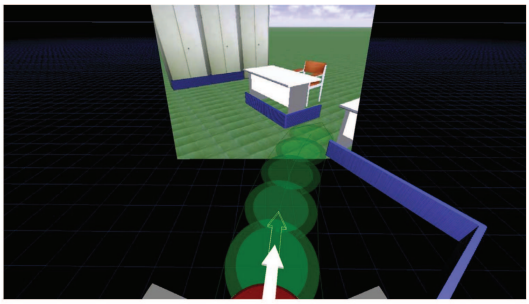
(c) + Augmentation sol



(d) + Augmentation obstacles



(e) + Augmentation trajectoire / action



(f) + Augmentation vecteur vitesse / action

FIGURE 4.4 – Conception itérative de l'interface de téléopération écologique.

4.2.2 Extension à la caméra 3D

La figure 4.6 montre une situation où l'information de passabilité est cohérente avec la réalité. On observe que les tables situées au milieu de la scène sont affublées de cartons afin de les rendre détectables par le télémètre laser. Sans cet artifice, le télémètre ne détecte pas ces obstacles et l'augmentation est incohérente avec la passabilité réelle. Nous pensons que la caméra 3D peut être utile dans un tel contexte pour fournir des données traduisibles, de façon robuste, en information de passabilité cohérente avec la scène.

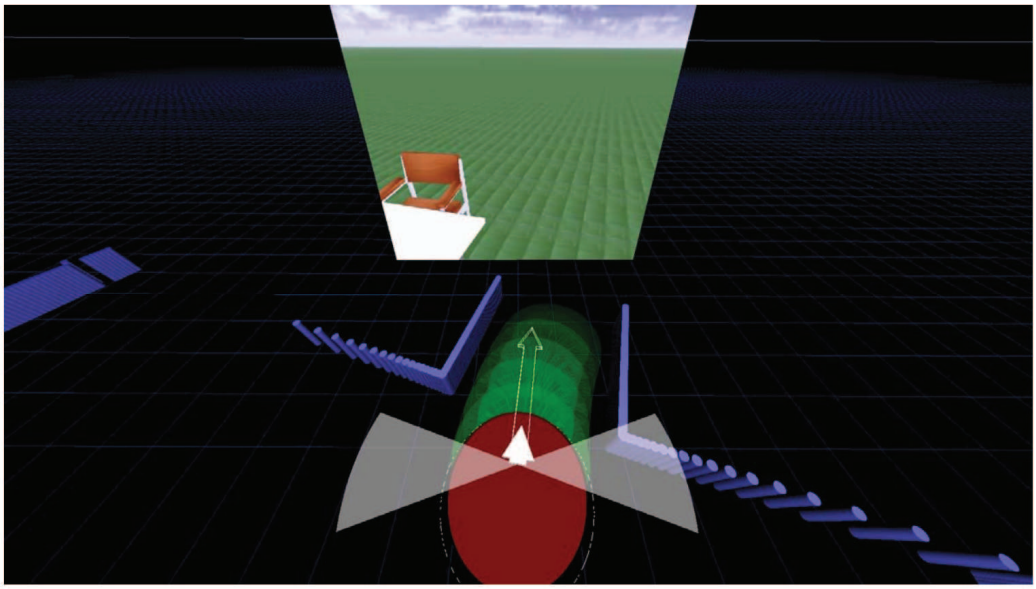


FIGURE 4.5 – Interface de téléopération en réalité augmentée.

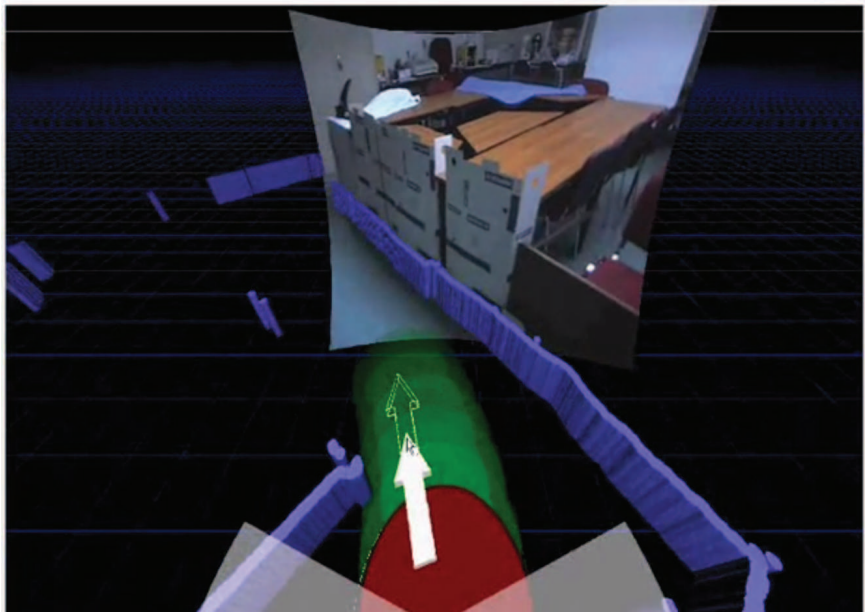


FIGURE 4.6 – Utilisation de l'interface de téléopération avec le laser seul.

L'extension de l'interface de téléopération pour un robot disposant d'une caméra 3D peut être résolue de deux façons différentes. Soit nous considérons la caméra 3D comme une extension du télémètre laser et nous utilisons la fusion proposée au chapitre 1, soit nous rendons la caméra 3D indépendante.

Cette interface de téléopération n'a pas été conçue pour un unique robot. Elle peut être adaptée facilement et rapidement à d'autres robots dont les caractéristiques peuvent être très différentes, y compris en ce qui concerne les capteurs embarqués. La base de la construction de cette IHM repose sur la caméra conventionnelle embarquée sur le robot. La première augmentation repose sur les données d'un télémètre laser à balayage mais tous les robots n'en sont pas équipés. Certains robots très largement commercialisés (comme les TurtleBot) n'en possèdent pas mais sont dotés d'une caméra 3D de type Kinect. Nous ne pouvons donc pas nous limiter au cas de l'utilisation conjointe de la caméra 3D et du télémètre laser. Nous avons donc choisi de proposer, pour la caméra 3D, une augmentation indépendante du télémètre laser.

De la même façon que pour le laser, réaliser une augmentation de la vue de la caméra conventionnelle avec les données issues de la caméra 3D nécessite de connaître la transformation entre les deux capteurs. Dans le chapitre 3 nous avons proposé trois méthodes remplissant cet objectif. La figure 4.7 montre le positionnement relatif des repères des différents capteurs réels (D pour la caméra de profondeur, C pour la caméra couleur et L pour le télémètre laser) ou de la caméra virtuelle V utilisée pour représenter la scène augmentée.

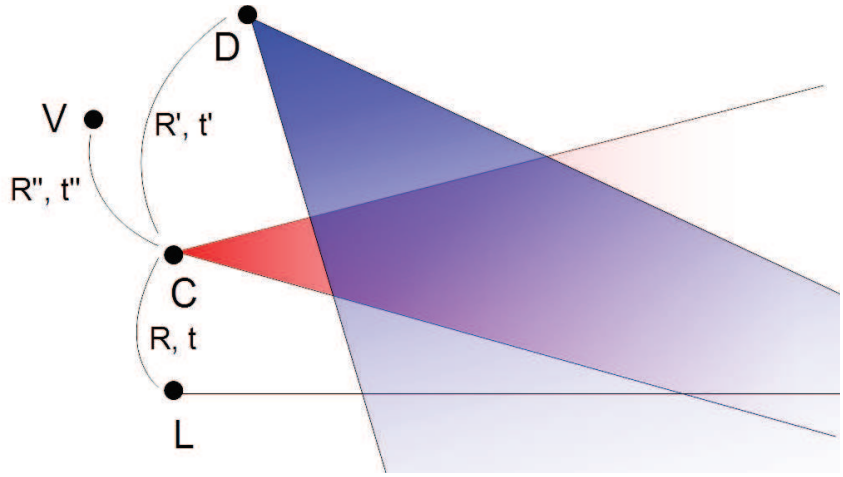


FIGURE 4.7 – Positions relatives des capteurs et de la caméra virtuelle.

Nous avons observé que l'information de passabilité était de plus en plus nécessaire à mesure que le robot se rapproche des obstacles. En effet, lorsque les obstacles sont éloignés,

l'image couleur suffit à déterminer une trajectoire approximative que l'utilisateur va corriger en utilisant les augmentations proches du robot. Nous avons donc choisi de placer la caméra 3D en hauteur avec un point de vue plongeant vers l'avant du robot afin de détecter les obstacles les plus proches, quitte à mettre hors-champ des obstacles plus éloignés qui peuvent potentiellement faire l'objet d'une détection par le télémètre laser.

L'information de passabilité donnée par le télémètre laser était représentée par tout un ensemble de petits cylindres bleus. Diverses évaluations menées par B. Mantel [Mantel *et al.*, 2012] ont montré que ce type d'affichage ne perturbe pas l'utilisateur et ne réduit pas son champ de vision dans l'image couleur tout en lui permettant une représentation de la circonférence au sol des obstacles. Nous conserverons ce paradigme pour les données de la caméra 3D mais nous les différencierons des informations transmises par le télémètre laser en utilisant la couleur rouge afin de représenter ces obstacles.

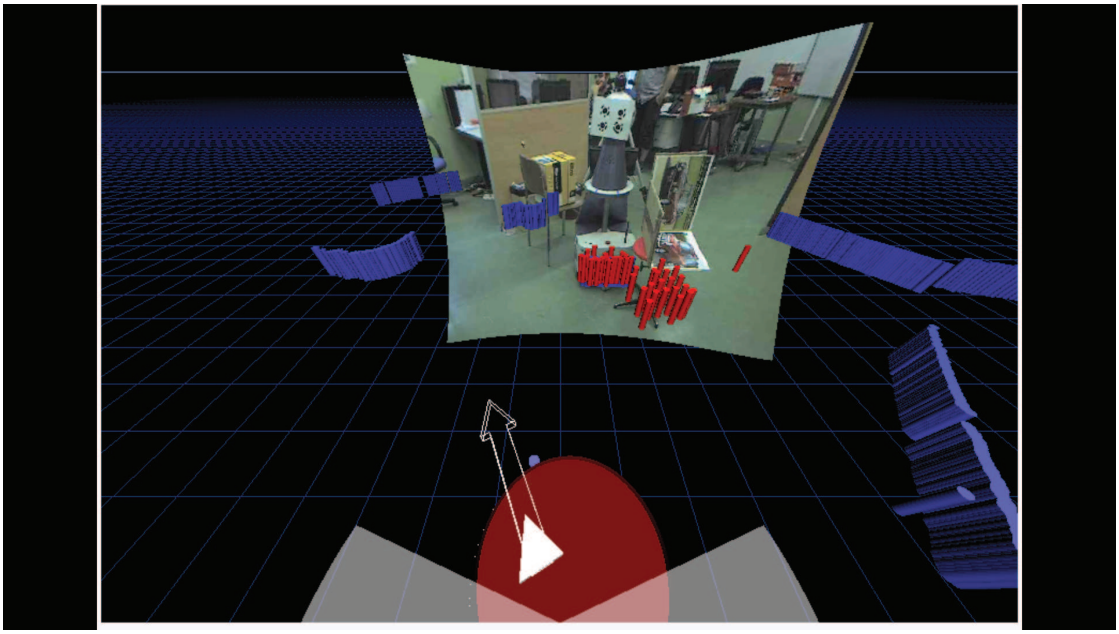
4.2.3 Evaluation

Dans cette section nous n'évaluerons pas de façon quantitative l'apport de l'interface de téléopération incluant la caméra 3D. La figure 4.6 (dans la section précédente) résume la situation d'utilisation avec le télémètre laser seul et montre que les données du laser ne peuvent pas être utilisées pour montrer la passabilité des obstacles. L'information de passabilité traduisant ces données est donc erronée et peut induire l'utilisateur en erreur.

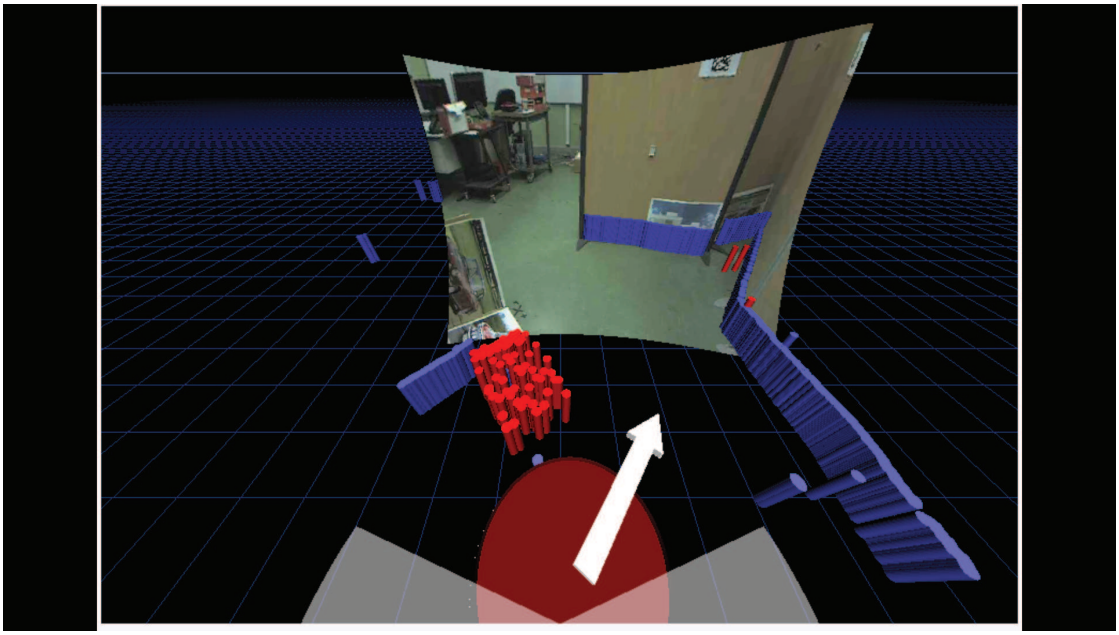
Nous montrerons donc dans cette section un ensemble de situations où l'information de passabilité (grâce aux données de la caméra 3D) est cohérente avec la scène malgré la présence d'obstacles complexes pour lesquels le télémètre laser seul ne donne pas de résultats probants.

La figure 4.8, tout d'abord, montre un groupe d'objets, au centre de l'image couleur, peu visible par le télémètre laser. Contrairement aux images de simulation montrées précédemment, l'image issue de la caméra couleur est déformée en utilisant les paramètres des distorsions de la caméra conventionnelle afin de permettre un rendu des augmentations

cohérent avec la vue en couleur de la scène.



(a) Vue d'un obstacle dans le champ de la caméra couleur



(b) Vue du même obstacle hors-champ

FIGURE 4.8 – Augmentation basée caméra 3D sur l'interface de téléopération.

Dans la figure 4.8a, on observe tout d'abord que les augmentations bleues (issues du télémètre laser) sont cohérentes avec les informations visuelles présentes dans l'image de la

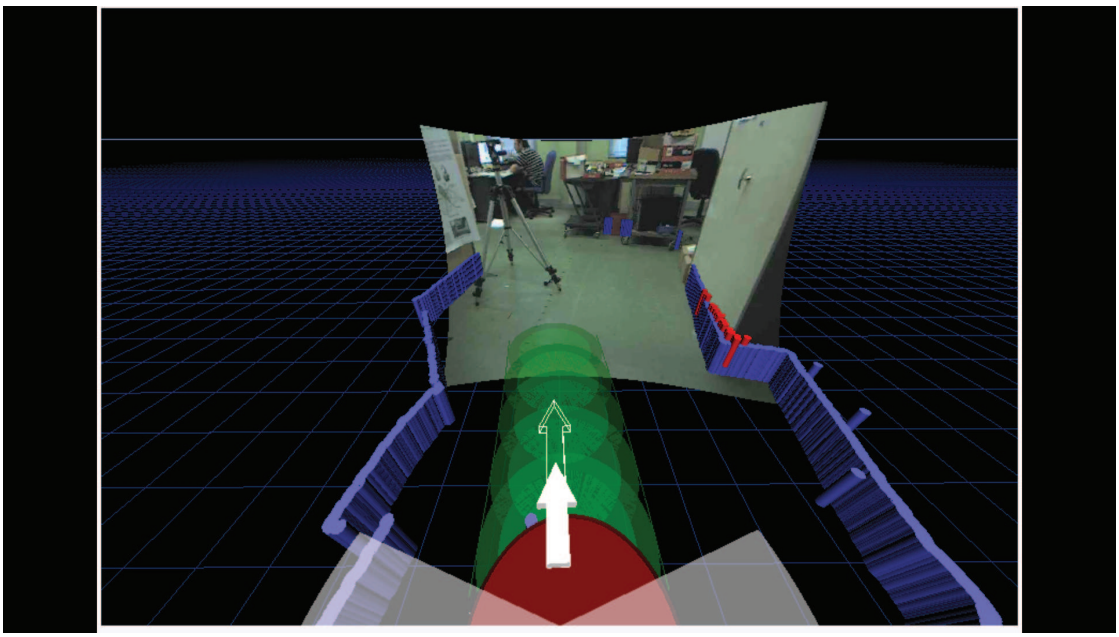
caméra conventionnelle. On remarque que le bord de la surface bleue à l'extrémité droite de l'image conventionnelle coïncide avec le bord du panneau partiellement visible. Dans le reste de la scène, non couvert par le champ de vision de la caméra conventionnelle, on observe la présence de différents obstacles difficiles à identifier mais dont la matérialisation par les augmentations bleues permet de tirer une information sur la passabilité du robot. Les augmentations rouges marquent la présence de deux objets aussi identifiables dans l'image couleur. L'un des deux objets est placé sur un tabouret et on observe que l'augmentation traduit la projection de cet objet sur le sol. Cela donne une information sur le volume global de l'objet.

Dans la figure 4.8b, on peut voir que le robot a été rapproché et déplacé légèrement à droite de l'objet placé sur le tabouret. On observe de nouveau que le panneau détecté par le laser est correctement localisé sur l'image couleur. Par contre, l'objet posé sur le tabouret est partiellement hors-champ de la caméra conventionnelle. L'opérateur peut avoir des difficultés à se rendre compte de la taille de cet objet et donc de la zone à éviter. L'augmentation rouge, basée sur la caméra 3D, permet de visualiser l'empreinte au sol de cet objet et donne donc à l'utilisateur une information sur le danger imminent lié à l'obstacle. L'opérateur devra ainsi ordonner au robot de passer entre cet obstacle et la surface située à droite, peu visible sur l'image conventionnelle mais augmentée en bleu.

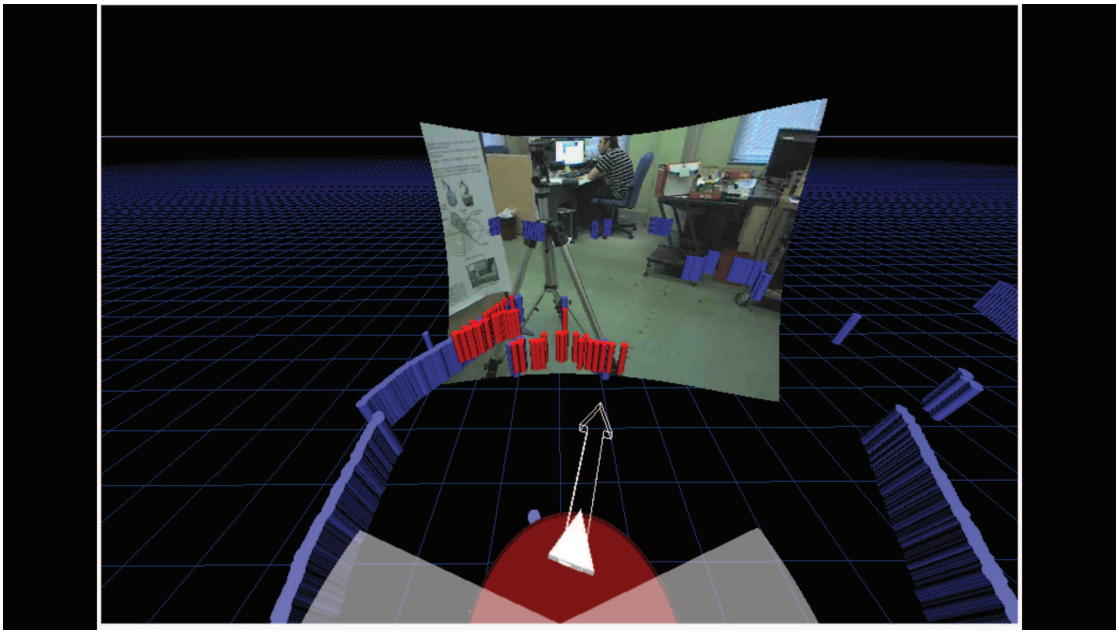
Dans la figure 4.9, la scène consiste en un passage large comme trois fois le robot dans lequel est présent un trépied.

Le trépied est un type d'obstacle très complexe à détecter pour le télémètre laser mais aussi pour la caméra 3D. En effet, il est constitué de multiples barres de dimension raisonnable qui sont très sujettes au *Mixed Pixel Problem* décrit dans le chapitre 1 et conduisent souvent à des échecs de mesure.

On observe d'ailleurs dans l'image 4.9a que le télémètre laser ne permet pas de détecter le trépied (absence d'augmentation bleue). L'obstacle étant hors-champ, la caméra 3D ne donne pas plus d'information. Dans la figure 4.9b, le robot s'est rapproché de l'obstacle suffisamment pour que le trépied entre dans le champ de vision de la caméra 3D. On ob-



(a) Vue d'un obstacle dans le champ de la caméra couleur



(b) En se rapprochant, l'obstacle rentre dans le champ de la caméra 3D

FIGURE 4.9 – Obstacle complexe de type trépied.

serve effectivement la présence d'augmentations rouges. En raison de la difficulté posée par le *Mixed Pixel Problem* y compris pour la caméra 3D, ces augmentations sont dynamiquement instables mais l'obstacle reste globalement toujours visible. On remarque en outre

que le poster à gauche de la scène est légèrement soulevé. Les augmentations rouges au sol montrent bien ce décollement de la surface du plan. Dans cette scène, grâce aux augmentations rouges et bleues qui complètent la vue conventionnelle, il ne fait aucun doute que le passage devra se faire à la droite du trépied.

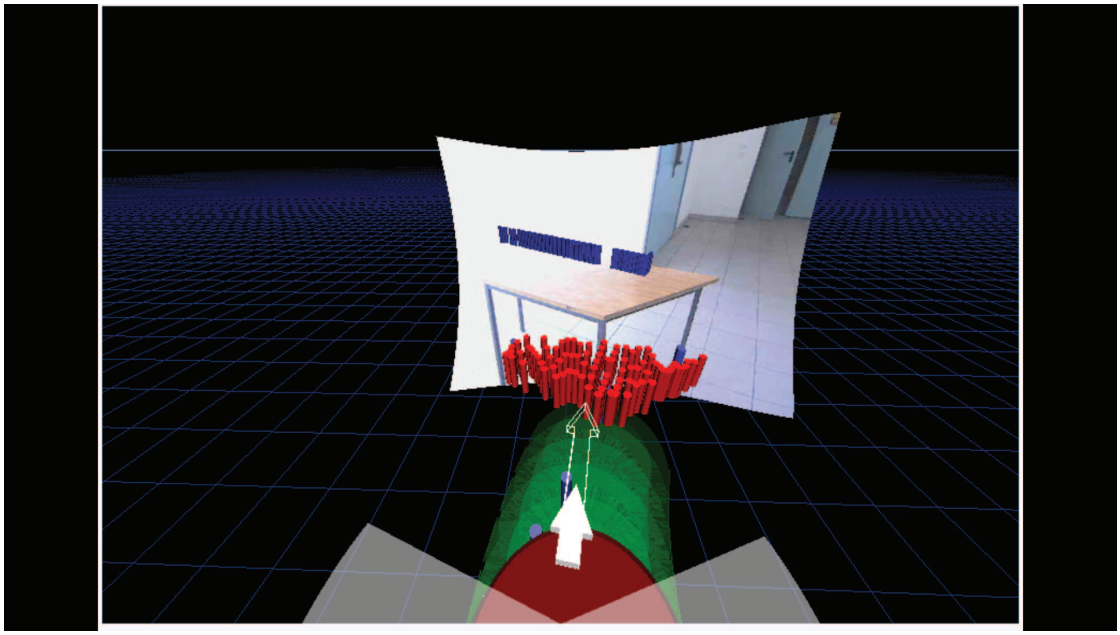


FIGURE 4.10 – Obstacle de type table

La figure 4.10 montre principalement une table et un mur. On observe que le mur est détecté par le télémètre laser alors que la table ne l'est pas du tout. Seul le pied arrière droit de la table est capté par le télémètre laser. Au contraire, la caméra 3D observe bien la totalité de l'objet (y compris la pointe à droite dépassant du pied) et les augmentations rouges au sol permettent à l'utilisateur de bien se rendre compte de l'impossibilité de continuer tout droit. On notera que ce type de situation évidente lorsqu'on dispose de l'image couleur de la table peut se révéler très difficile à gérer sans augmentation lorsque le robot se rapproche de la table et que celle-ci sort du champ de vision de la caméra conventionnelle ou lorsque l'éclairage est insuffisant pour obtenir une image couleur lisible.

Enfin, un autre obstacle difficile à gérer est représenté dans la figure 4.11. Il s'agit d'un ventilateur sur pied. On observe tout d'abord que le télémètre laser ne détecte pas

le pied et que, par conséquent, aucune augmentation bleue ne permet de se faire une idée de la passabilité dans cette situation. Au contraire, le pied du ventilateur est bien détecté par la caméra 3D comme le montrent les augmentations rouges. Si l'opérateur prend ainsi connaissance de la position du pied, cela ne permet pas a priori de déterminer l'empreinte au sol de l'objet. Néanmoins, il faut être conscient que l'opérateur possède un vécu et que ce n'est probablement pas la première fois qu'il rencontre ce type d'objet. Il aura connaissance du fait que l'empreinte au sol du pied ne se limite probablement pas à sa partie verticale. L'humain, se déplaçant près de ce ventilateur, n'aura pas besoin de regarder le sol pour éviter de marcher sur le pied du ventilateur. La position au sol de l'augmentation rouge permet de se faire une idée de la taille du ventilateur. Grâce aux caractéristiques de l'anthropomorphisme, l'opérateur peut se projeter dans la scène et se rendant compte de la taille de l'objet, va naturellement estimer une zone autour de l'augmentation rouge (représentant la partie verticale du pied) dans laquelle il ne faut pas entrer. La trajectoire ordonnée par l'opérateur va naturellement tenir compte de cette marge de sécurité et ne pas se rapprocher trop du ventilateur. On observe ainsi la trajectoire choisie par l'opérateur (en vert) qui passe plus proche de l'augmentation bleue que de l'augmentation rouge.

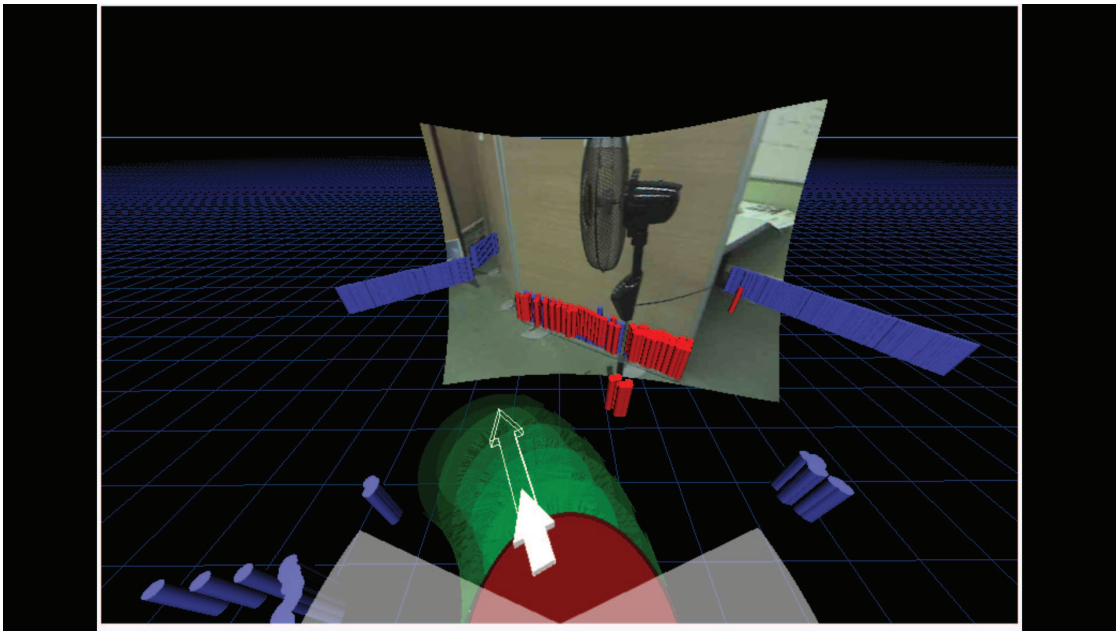


FIGURE 4.11 – Obstacle de type pied de ventilateur

Nous avons montré dans cette section l'intérêt de l'utilisation conjointe de la caméra 3D et de la caméra conventionnelle en situation de téléopération dans une interface originale en réalité augmentée conçue avec une approche écologique. Cet usage nécessite un étalonnage extrinsèque précis pour pouvoir aligner les augmentations avec l'image couleur et ainsi permettre à l'opérateur d'adapter sa trajectoire à la présence d'obstacles même hors-champ de la caméra conventionnelle. Dans la section suivante, nous verrons l'intérêt d'un tel étalonnage dans le cadre d'un algorithme de SLAM pour la robotique autonome.

4.3 Application au SLAM RGB-D

Le modèle de SLAM utilisé a été proposé par F. Endres, J. Hess, N. Engelhard, J. Sturm et D. Cremers dans [Endres *et al.*, 2014] et [Engelhard *et al.*, 2011] à partir de 2011. Cet algorithme a été choisi pour deux raisons principales. Il est reconnu par la communauté comme une référence en terme de rapport qualité/temps de calcul (aussi bien en utilisant seulement le CPU qu'en calcul basé sur GPU) pour les environnements de petite taille. Enfin, une implémentation est notamment disponible en *open source* pour une intégration facile sous ROS tout en autorisant une totale modification du code source. L'objectif de cette partie n'est pas de proposer ou d'améliorer un algorithme de SLAM pour capteur RGB-D mais de démontrer l'importance d'un étalonnage correct des paramètres extrinsèques de la paire de caméras 3D et conventionnelle. Nous nous appuierons sur les outils et les jeux de données pour l'évaluation des SLAM RGB-D de l'université technique de Munich (TUM - *Technische Universität München*) disponible librement sur internet. Dans [Sturm *et al.*, 2012], les auteurs ont notamment proposé une suite d'outils d'évaluation de l'ATE (*Absolute Trajectory Error*) que nous utiliserons dans notre évaluation ainsi que tout un ensemble de 89 jeux de données d'évaluation pour un total de plus de 50 Go d'images couleur et d'images de profondeur synchronisées.

4.3.1 Modèle du SLAM RGB-D

L'algorithme de SLAM décrit dans [Endres *et al.*, 2014] est un travail tirant parti des images denses issues de la caméra couleur et de la caméra de profondeur. Il existe d'autres approches exploitant les images RGB-D. On trouve notamment KinectFusion [Newcombe *et al.*, 2011] qui repose sur un volume discrétisé de voxels contenant la TSDF (*Truncated Signed Distance Function*) à la surface pour chaque observation. Cet algorithme a pour principal avantage de présenter une faible dérive en comparaison aux algorithmes reposant sur un alignement itératif d'images comme le RGB-D SLAM d'Endres. Au contraire, KinectFusion présente l'inconvénient d'une importante occupation mémoire et ne permet pas d'optimisation lors de la fermeture de boucle, ce qui réserve cet algorithme à des environnements de très petite taille ainsi qu'à la modélisation d'objets. Des extensions comme [Whelan *et al.*, 2013] ont résolu le problème de la taille de l'environnement en ne conservant qu'un volume limité centré sur la caméra et en triangulant les parties éliminées. Zeng et al. [Zeng *et al.*, 2013] montrent que l'on peut résoudre les problèmes de capacité mémoire avec l'encodage des voxels dans un octree. Néanmoins ce type d'extension ne résout ni le problème de l'augmentation indéfinie de la dérive à cause de l'impossibilité de traiter les fermetures de boucle, ni la nécessité de disposer d'importantes capacités de calcul GPU. Dans une approche complètement opposée, Hu et al. [Hu *et al.*, 2012] proposent un SLAM RGB-D utilisant des ajustements de faisceaux avec ou sans la profondeur rendant ainsi l'algorithme robuste aux pertes d'information sur la profondeur (en fonction des conditions d'illumination ou de la distance). Ce type de travail se rapproche plus du SLAM monoculaire ajoutant éventuellement des contraintes relatives à la profondeur des observations.

L'approche de F. Endres [Endres *et al.*, 2014] est centrée sur la position 3D des points d'intérêt issus de l'image couleur et leurs descripteurs. Le pipeline de traitement se décompose en deux parties comme le montre la figure 4.12 : le *frontend* et le *backend*.

Le *frontend* traite successivement les images couleur et de profondeur et en extrait les descripteurs des points d'intérêt détectés (dont SIFTGPU, SIFT, SURF, SURF128

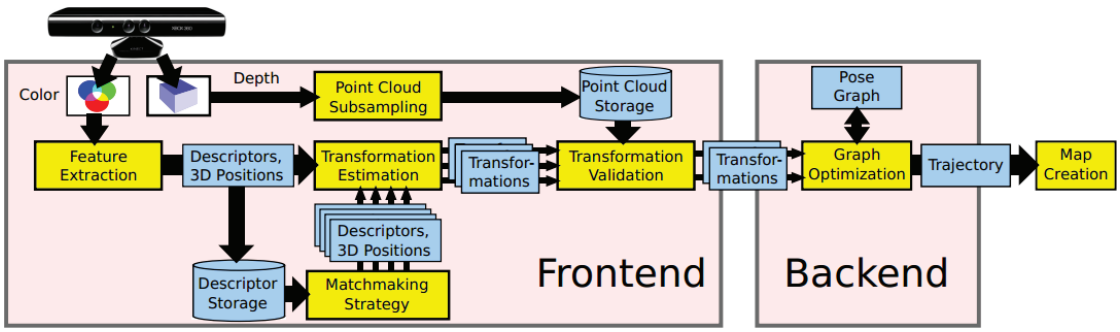


FIGURE 4.12 – Pipeline de traitement du RGB-D SLAM. ([Endres *et al.*, 2014])

et ORB) et les coordonnées 3D des points. Les descripteurs sont appariés avec la base de descripteurs retenus dans le passé puis, à partir de l'ensemble des paires formées, une transformation entre les poses du capteur est estimée à l'aide d'un algorithme ICP-RANSAC. Cette transformation est ensuite validée en calculant les *inliers* parmi les points mis en correspondance (y compris ceux ne disposant pas d'information 3D).

Dans le *backend*, la transformation ainsi calculée est ensuite utilisée pour créer un graphe composé des deux poses successives et de leurs *inliers* et optimiser les positions relatives des amers et des poses de caméra. Lorsqu'une fermeture de boucle est détectée, une optimisation est aussi effectuée sur le graphe total des amers et des poses successives de la trajectoire.

L'algorithme original de F. Endres part de l'hypothèse que les images couleur et de profondeur ont été préalablement alignées à l'aide des paramètres intrinsèques et extrinsèques issus d'un étalonnage. Les paramètres d'étalonnage extrinsèque n'interviennent donc pas dans le pipeline de traitement. Pour les besoins de notre évaluation ci-après, et avec l'aimable autorisation de l'auteur, nous avons modifié l'algorithme pour introduire explicitement ces paramètres au niveau de la construction du nuage de points ainsi qu'au niveau de l'extraction des coordonnées 3D des points d'intérêt. Lorsque nous ne modifions pas les paramètres extrinsèques d'étalonnage, les résultats obtenus avec notre algorithme sont, en tout point, comparables à ceux obtenus avec le logiciel original de F. Endres.

4.3.2 Evaluation

Dans la suite du chapitre, nous évaluerons quantitativement l'influence de l'erreur d'étalonnage sur la qualité de la trajectoire estimée par l'algorithme de SLAM. Pour ce faire, nous nous appuierons sur la métrique de l'ATE (*Absolute Trajectory Error*). Cette métrique permet d'évaluer la différence absolue entre les poses en les comparant après alignement avec la trajectoire de référence (considérée comme la vérité terrain) obtenue par un système de tracking multi-caméras à 100Hz utilisant des marqueurs réflexifs. Les résultats que nous présentons dans cette section reposent sur 11 calculs du RGB-D SLAM sur un même jeu de données appelé *fr1/360* en faisant varier indépendamment les paramètres d'étalonnage de translation et de rotation. Ce jeu de données correspond à une séquence de 28,69 s et a pour longueur 5.818 m. Il correspond aux déplacements d'un capteur Kinect au centre d'un bureau pour observer toute la pièce du sol au plafond. Les vitesses maximales du capteur sont de 0.210 m/s et 41.6 °/s. La trajectoire reste limitée à un volume d'environ 0.12 m^3. Les auteurs du SLAM annoncent un RMSE (*Root Mean Square Error*) ATE moyen de 0.059 m en utilisant le jeu de données *fr1/360* avec les points d'intérêt SIFT (GPU). Ce type de point d'intérêt offre les meilleures performances mais notre architecture n'en permet pas l'utilisation. Nous nous tournerons donc vers les points d'intérêt SURF. En utilisant les paramètres recommandés par les auteurs et les points d'intérêt SURF, le RMSE ATE moyen obtenu est de 0,062 m.

Dans les figures 4.13 et 4.14, nous présentons l'ATE moyen pour représenter l'écart moyen de la trajectoire estimée à la trajectoire réelle (carrés bleus) ainsi que leur écart-type (représenté par les barres d'erreur). La courbe bleu clair supérieure représente les valeurs maximales alors que la courbe inférieure représente les valeurs minimales d'écart à la trajectoire de référence.

La figure 4.13 présente en abscisse la norme de l'erreur sur le vecteur de translation des paramètres d'étalonnage extrinsèque. Dans le jeu de données utilisé, les images conventionnelles et de profondeur sont alignées et corrigées de telle sorte que tous les paramètres

d'étalonnage extrinsèque à partir des ces images soient nuls. Les paramètres de translation utilisés sont tous égaux et la norme du vecteur d'erreur de translation est lisible en abscisse. Pour chaque point d'abscisse x dans le graphique, nous avons donc un vecteur de translation $\left[\sqrt{3x^2}, \sqrt{3x^2}, \sqrt{3x^2}\right]^\top$ (répartissant l'erreur de translation équitablement sur chaque axe) et un vecteur de rotation nul en guise de paramètres extrinsèques. Les points du graphique correspondent successivement à une erreur de 1, 2, 5, 7.5 et 10 cm sur chaque axe de la translation.

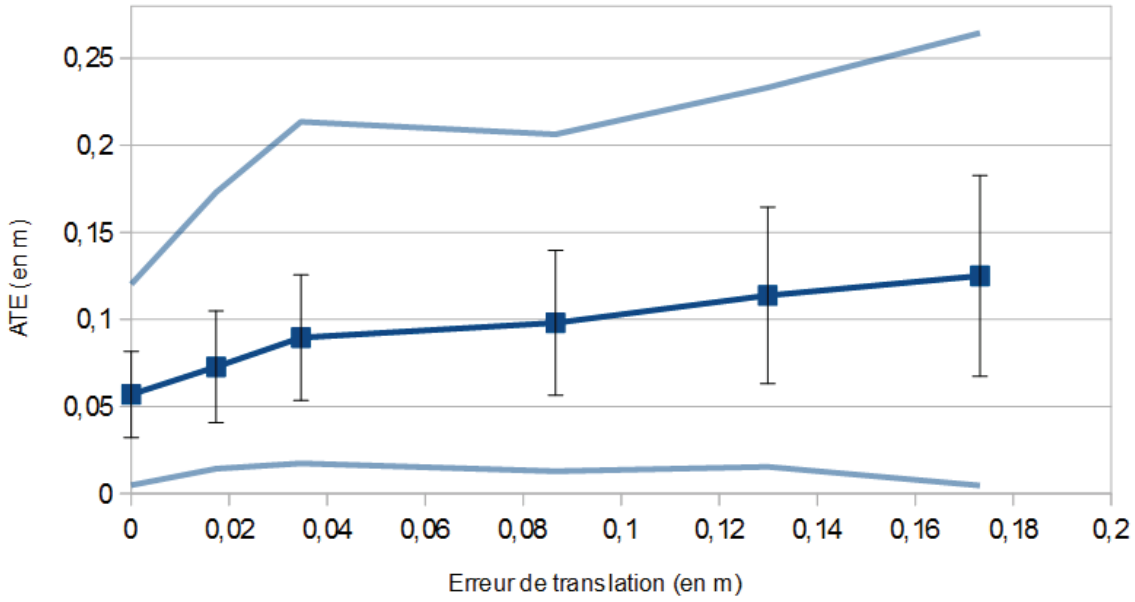


FIGURE 4.13 – Effet d'une erreur d'étalonnage en translation sur l'estimation de la trajectoire

On observe ainsi que l'erreur induite sur la trajectoire est croissante avec l'erreur de translation de l'étalonnage. On distingue sur le graphique deux parties linéaires. La première partie, entre 0 et 4 cm, présente un taux d'accroissement linéaire de l'ATE d'environ 1 cm par cm d'erreur sur la norme de la translation (équivalent à moins de 6 mm d'erreur sur chaque axe). Au contraire, au-delà de 4 cm, ce taux d'accroissement diminue à environ 3 mm par cm d'erreur. Par contre, l'écart-type de l'ATE augmente constamment pour quadrupler entre 0 et 0,17 cm d'erreur de translation.

La figure 4.14 présente en abscisse la norme de l'erreur sur le vecteur de rotation des

paramètres d'étalonnage extrinsèque. Pour chaque point d'abscisse x dans ce graphique, nous avons donc un vecteur d'erreur de rotation $\left[\sqrt{3x^2}, \sqrt{3x^2}, \sqrt{3x^2}\right]^\top$ exprimé selon le formalisme d'Euler et un vecteur de translation nul en guise de paramètres extrinsèques. Les points du graphique correspondent successivement à une erreur de 0.01, 0.035, 0.05, 0.1 et 0.125 rad sur chaque axe de rotation ou encore une norme du vecteur de rotation respectivement d'environ 1, 3.5, 5, 10 et 12.4°.

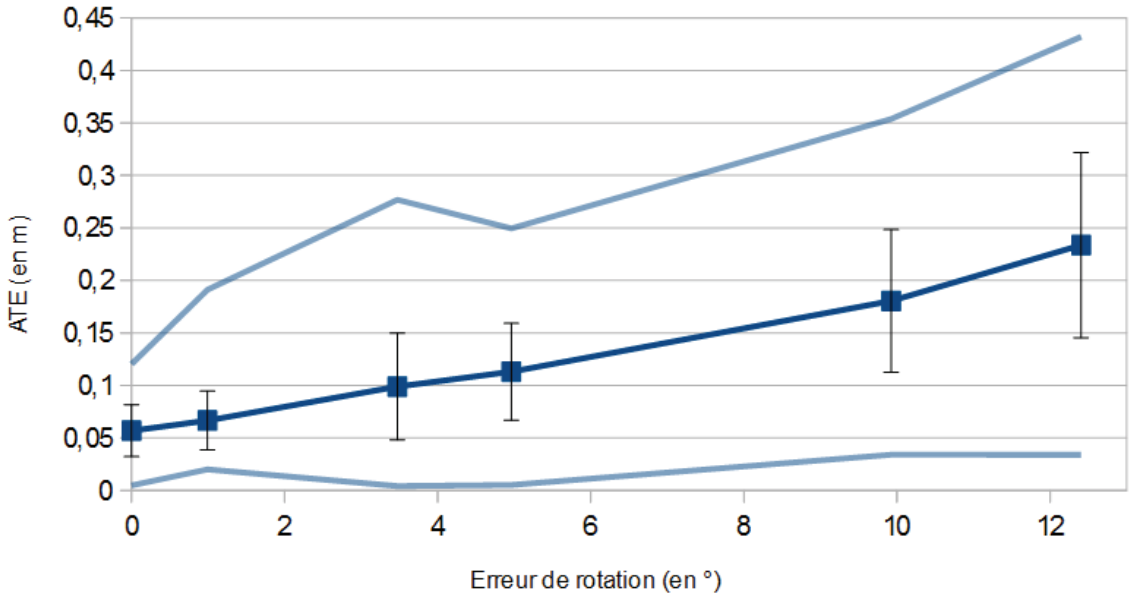


FIGURE 4.14 – Effet d'une erreur d'étalonnage en rotation sur l'estimation de la trajectoire

On observe, comme pour la translation, une tendance globalement croissante avec la norme de la rotation aussi bien en terme de moyenne que d'écart-type ou de valeurs maximales. L'ATE semble respecter une loi de croissance exponentielle (avec un coefficient de détermination $R^2 = 0.99$) ou est modélisable par une droite ($R^2 = 0.98$). Le taux d'accroissement linéaire de l'ATE en fonction de l'erreur d'étalonnage en rotation approche 1,5 cm par degré d'erreur. Au-delà du dernier point de ce graphique (au delà de 13° d'erreur), l'algorithme de SLAM n'a pas permis de trouver la trajectoire complète.

Si l'effet d'une erreur d'étalonnage est aussi visible sur la translation que sur la rotation, nous nous attendions à une dérive bien plus importante de la trajectoire estimée par le SLAM et notamment une explosion exponentielle de l'erreur (voire une non convergence du

SLAM) lorsque l'erreur d'étalonnage devient trop importante. Dans les faits, l'erreur induite reste limitée sur des trajectoires courtes car elle est corrigée par l'optimisation globale en cas de fermeture de boucle. Cette optimisation globale concerne aussi la position des points 3D. L'influence de l'étalonnage est donc atténuée ce qui explique l'utilisation possible des SLAM RGB-D même avec des étalonnages peu précis. Ceci n'est vrai que lorsque le SLAM réussit à déclencher la fermeture de boucle. Lorsque l'erreur d'étalonnage est forte et lorsque le nombre d'*inliers* est peu important, nous avons observé que l'algorithme d'alignement pouvait échouer à trouver une solution et la trajectoire ne peut plus continuer à être estimée tant qu'une boucle n'est pas réalisée. A ce moment, le SLAM reprend son estimation mais la fermeture de boucle ne déclenche pas d'optimisation (à cause de la partie manquante de la trajectoire).

4.4 Conclusion

Ce chapitre nous a permis de mettre en évidence l'importance d'une bonne estimation des paramètres extrinsèques d'étalonnage entre une caméra conventionnelle et une caméra 3D au travers de deux applications.

Tout d'abord, nous avons réalisé une extension d'une interface de téléopération en réalité augmentée qui, bien que prometteuse, souffrait d'un manque de cohérence entre l'information de passabilité issue d'un télémètre laser et la réalité dans la scène. Dans un environnement non adapté, cette interface pouvait fournir une information de passabilité incorrecte à l'opérateur et donc l'induire en erreur dans le choix de la trajectoire de son avatar robotique. Pour corriger ce défaut, nous avons réutilisé le paradigme de conception écologique afin de fournir à l'opérateur une information de passabilité plus fiable en utilisant les capacités de la caméra 3D. Nous avons montré, par différents exemples, que cette information permet de mieux déterminer une trajectoire sûre alors que l'information issue du télémètre seul n'était pas utilisable dans un environnement usuel présentant des obstacles de formes complexes.

Enfin, dans une application de SLAM pour un robot autonome, nous avons montré qu'une mauvaise estimation des paramètres extrinsèques d'étalonnage entre la caméra couleur et la caméra de profondeur avait une influence négative sur la qualité de la trajectoire estimée par l'algorithme de SLAM RGB-D. Ce résultat a une répercution directe en cas de reconstruction de la scène observée avec, par exemple, un nuage de points dense. En effet, dans ce cas, même si la forme globale de la scène est respectée grâce aux optimisations lors des fermetures de boucle, la dimension ainsi que la texture des objets ne correspondent pas à la réalité et donnent une sensation désagréable à l'utilisateur.

Par ces deux exemples, nous avons montré que l'étalonnage extrinsèque de la caméra 3D et de la caméra conventionnelle est indispensable pour permettre une utilisation correcte des outils de téléopération (interface de contrôle en réalité augmentée) et de SLAM (localisation et cartographie simultanées). Même si les caméras RGB-D du commerce sont vendues *"étalonnées"*, nous recommandons de réaliser un étalonnage spécifique pour chaque capteur comme il est d'usage pour les caméras conventionnelles. Les méthodes que nous avons proposées au chapitre 3 ne demandent pas un protocole trop exigeant. Il n'y a donc aucune raison de se contenter des paramètres d'usine. Les caméras conventionnelles intégrées à ces capteurs étant souvent de qualité médiocre, ce genre d'étalonnage est de toute façon nécessaire pour l'utilisation d'une caméra conventionnelle externe comme c'est le cas dans notre interface de téléopération par exemple.

Conclusion

Dans cette thèse, nous avons abordé la problématique de la perception multicapteur à travers deux prismes. Le premier axe se préoccupe de l'utilisation des nouveaux capteurs à bas coût de type caméra 3D par un robot mobile déjà équipé d'autres capteurs courants comme des caméras conventionnelles ou un télémètre laser à balayage. Ceci soulève un ensemble de problèmes applicatifs autour du déplacement autonome ou téléopéré d'un robot mobile. Le deuxième axe d'analyse se concentre sur la question scientifique de l'étalonnage (en particulier extrinsèque) inhérente à l'intégration d'un nouveau capteur dans un environnement multicapteur.

Dans ce manuscrit nous avons choisi de présenter ces travaux en deux parties distinctes. La première se préoccupe de l'étalonnage entre une caméra 3D et un télémètre laser à balayage et de son utilisation en navigation autonome au travers d'un évitement multicapteur robuste aux obstacles de forme complexe. La deuxième partie considère le problème de l'étalonnage entre une caméra 3D et une caméra conventionnelle. Au niveau applicatif, nous avons proposé une interface de téléopération en réalité augmentée couplant ces deux types de capteurs puis nous avons montré l'importance de ce type d'étalonnage dans une application de SLAM RGB-D très utilisée dans toute la communauté robotique.

Bilan

Caméra 3D et télémètre laser à balayage

Le premier chapitre présente une méthode d'étalonnage extrinsèque entre une caméra 3D à bas coût et un télémètre laser à balayage. Au début de ces travaux, nous n'avons pas trouvé dans la littérature de recherches portant sur l'utilisation conjointe de ce genre de capteur. Nous avons proposé d'étendre une méthode d'étalonnage connue portant sur une caméra conventionnelle et un télémètre laser. En partant du remplacement de la caméra conventionnelle par la caméra 3D, la modification de cette méthode a permis de s'affranchir de la nécessité d'utiliser un plan étalon texturé avec un motif connu (un damier) perturbant les télémètres laser à bas coût. Nous avons montré des résultats significativement plus précis lorsque le plan étalon ne présente pas l'alternance de couleurs (blanc/noir) du damier. Pour que l'étalonnage conjoint d'une caméra conventionnelle reste possible, nous avons proposé un motif d'étalonnage de type damier modifié qui tient compte de ce précédent résultat qui permet d'augmenter la précision de l'étalonnage d'environ 30% et de diminuer la dispersion de presque 40% par rapport à un motif de damier traditionnellement utilisé.

Cette méthode d'étalonnage nous a permis de proposer la création d'un télémètre virtuel remplaçant de façon transparente un télémètre laser à balayage. Ce télémètre virtuel, consolidant les données issues du télémètre laser réel et de la caméra 3D, possède deux avantages. Tout d'abord, il permet d'augmenter très significativement la sensibilité du télémètre laser réel dans la zone aussi couverte par le champ de vision de la caméra 3D tout en n'augmentant pas le taux de faux positifs. La détection de l'environnement est ainsi plus robuste aux obstacles de forme complexe pas toujours détectés par le témètre laser seul. Enfin, la création d'un tel télémètre virtuel répond à deux difficultés soulevées par la caméra 3D. L'utilisation conjointe du télémètre laser et de la caméra 3D permet d'étendre le champ de vision de cette dernière et l'expression des données sous la forme d'un télémètre à balayage virtuel permet de réduire considérablement (environ d'un facteur 300) la taille des données à transmettre (faible bande passante nécessaire) et à traiter (rapidité de

traitement).

Pour évaluer le bénéfice d'un tel télémètre virtuel et de l'étalonnage associé, nous avons proposé une application d'évitement d'obstacles composée d'un algorithme de SLAM local relatif avec oubli et d'un algorithme d'évitement réactif robuste. Dans ces travaux, nous avons ainsi montré, aux travers de quelques exemples, un ensemble de situations où l'apport de la caméra 3D par l'intermédiaire du télémètre virtuel permet d'éviter une situation dangereuse alors que le télémètre laser réel ne détecte pas d'obstacle.

Caméra 3D et caméra conventionnelle

Nous avons ensuite proposé trois méthodes d'étalonnage extrinsèque entre une caméra 3D et une caméra conventionnelle. Ces méthodes peuvent être utilisées aussi bien dans le cas de l'utilisation d'un capteur de type Kinect qui intègre les deux types de caméra dans un même boitier que dans un contexte où la caméra 3D sera indépendante de la caméra conventionnelle. Si, dans le cadre d'un capteur tout intégré, le constructeur fournit des paramètres extrinsèques d'usine, ces méthodes permettent d'affiner ces paramètres et d'obtenir de meilleurs résultats dans les applications utilisant conjointement ces deux types de caméras. Dans le cas de caméras indépendantes, les normes en rotation et en translation de la transformation rigide entre les deux capteurs seront plus importantes et la nécessité d'un étalonnage extrinsèque se fera d'autant plus ressentir.

La première méthode proposée (M1) est inspirée de la littérature de l'étalonnage pour la télémétrie laser. Cette méthode a été développée au début de cette thèse alors que la seule proposition de la communauté, à notre connaissance, était de réaliser un étalonnage entre la caméra couleur et la caméra IR. En montrant le non-alignement de l'image IR et de l'image de profondeur, nous avons établi que cet étalonnage ne pouvait convenir à la caméra 3D que si l'on se contente d'une forte approximation. Nous en avons déduit qu'il était nécessaire de réaliser l'étalonnage directement entre la caméra 3D et la caméra conventionnelle. En considérant que la caméra 3D peut se substituer à un télémètre laser, nous avons pu étendre la méthode d'étalonnage extrinsèque entre la caméra 3D et le télémètre laser à balayage

pour proposer un étalonnage utilisant un plan à motif damier traditionnel.

Les retours de la communauté RGB-D nous ont appris que l'on se contentait souvent de l'approximation proposée par les paramètres d'usine du fait de la complexité (et du caractère fastidieux) des processus d'étalonnage. La méthode par asservissement visuel virtuel (M2) que nous avons proposée est basée sur un étalon 3D simple à réaliser et dont la précision de construction importe peu. Complétement automatique et n'utilisant qu'une unique vue de cet étalon, cette démarche va complètement à contre-courant des approches de la littérature. Cette méthode automatique et rapide ouvre un champ d'applications nécessitant un étalonnage temps réel à cause de la non rigidité volontaire (utilisation de capteurs indépendants et mobiles) ou non (facteurs mécaniques) du couple caméra 3D/caméra couleur.

La dernière méthode présentée (M3) permet d'obtenir un étalonnage aussi précis que la meilleure méthode de la littérature tout en évitant l'utilisation d'un plan ou d'un motif étalon particulier. Cette approche n'a pas besoin non plus de l'intervention de l'utilisateur, ce qui fait de cet algorithme une méthode à la fois naturelle et automatique. En contrepartie, son temps de calcul est assez long, ce qui rend ce modèle inutilisable en temps réel. Pourtant cette méthode reste simple à mettre en oeuvre puisqu'elle ne nécessite que deux vues d'une scène statique suffisamment texturée. La transformation entre les deux prises de vues est calculée automatiquement en utilisant les données du capteur 3D grâce à un algorithme rapide d'alignement 3D-3D basé sur l'observation de plans structurant la scène.

A travers deux applications, nous avons voulu montrer l'intérêt de l'étalonnage extrinsèque entre une caméra 3D et une caméra conventionnelle. La première application considère une caméra 3D indépendante de la caméra conventionnelle. Ces caméras sont utilisées dans le cadre d'une application de téléopération de robot en réalité augmentée. Au travers de différentes situations rencontrées en milieu intérieur, nous avons montré que la caméra 3D apportait une information complémentaire de la caméra conventionnelle (et du télémètre laser) à propos de la passabilité de la trajectoire. Dès lors que l'obstacle sort du champ de vision de la caméra conventionnelle, il est essentiel que l'opérateur conserve

une information sur la position de l'obstacle. La caméra 3D permet d'offrir cette information et le mode de représentation en réalité augmentée libère l'opérateur d'une charge cognitive trop importante lors du contrôle du robot.

Enfin, dans la dernière application nous avons voulu voir l'influence de la précision de l'étalonnage entre une caméra 3D et une caméra conventionnelle pour l'utilisation d'un algorithme de SLAM RGB-D. Nous avons pris pour référence un algorithme de SLAM proposé par F. Endres très communément utilisé car proposé dans ROS. Nos travaux ont ainsi montré qu'une erreur d'étalonnage induit significativement une erreur sur l'estimation de la trajectoire du capteur RGB-D. Cette erreur, croissante avec l'imprécision d'étalonnage, reste relativement mesurée sur des trajectoires courtes mais prouve la nécessité d'un étalonnage rigoureux lorsque la précision du SLAM doit être importante comme dans des applications de modélisation, de reconstruction ou de métrologie ou encore lorsque les trajectoires sont de grande taille pour la navigation autonome de robot.

Perspectives

Ces travaux trouvent de nombreuses perspectives tant applicatives que techniques et scientifiques. Tout d'abord, la construction du télémètre virtuel à partir du télémètre laser et de la caméra 3D ouvre la possibilité d'utiliser ce capteur virtuel au sein d'une chaîne complète de navigation allant de la localisation et de la cartographie simultanées à la navigation en passant par la planification dynamique et le suivi de trajectoire [Devaux *et al.*, 2011]. A court terme, il serait intéressant d'évaluer l'influence du télémètre virtuel sur la qualité de la cartographie et de la localisation par rapport à l'utilisation du télémètre laser seul. Puis nous voudrions étudier l'influence de ce télémètre virtuel (et de l'éventuelle meilleure reconstruction par SLAM) sur le nombre de replanifications pendant l'exécution d'une trajectoire par un robot autonome.

S'agissant des méthodes d'étalonnage que nous avons proposées, nous voulons travailler sur le temps de calcul de la méthode M2 pour la rendre temps réel et ainsi permettre l'usage de cet algorithme dans le cadre de capteurs mobiles ne respectant pas l'hypothèse

de rigidité entre la caméra 3D et la caméra conventionnelle. L'application directe de ce type de traitement serait de pouvoir modifier l'orientation de l'un ou l'autre capteur pour une raison applicative (commande de l'opérateur, robotique holonome, ...) tout en conservant une cohérence des données. La méthode M3 doit aussi pouvoir voir son temps de calcul amélioré par une optimisation du code et l'utilisation des calculs sur GPU notamment dans la phase de génération de l'image virtuelle de profondeur.

Pour continuer, la robustesse de M3 dépend principalement de l'appariement des points d'intérêt et de la possibilité d'obtenir une information de profondeur pour chacun de ces points. Nous pensons qu'il faut travailler sur la sélection des points d'intérêt sous ce type de contrainte tout en étudiant la possibilité de compléter les images de profondeur en faisant une estimation des données manquantes éventuellement avec l'aide d'une segmentation des images conventionnelles [Boucher, 2013]. Nous voulons, de plus, étudier plus en détail le cas des mouvements complexes dans l'utilisation de M3 en se basant sur une vérité terrain obtenue à partir de caméras de tracking.

Enfin, nous pensons que l'algorithme d'alignement 3D-3D proposé peut être utilisé pour effectuer l'étalonnage extrinsèque rapide d'un couple de caméras 3D. Une telle méthode basée sur les plans structurant la scène serait alors complètement automatique et utiliserait les éléments présents dans une scène quelconque en milieu intérieur. Nous estimons que ce type d'algorithme peut permettre un étalonnage extrinsèque temps réel.

Finalement, le dernier point à améliorer à court terme se situe au niveau de la prise en compte du sol et du plafond. La suppression du sol, notamment, est réalisée à partir de seuillages basiques qui, en prenant en compte des marges d'erreur, ne permettent pas de conserver les obstacles les plus bas situés près du sol. La connaissance a priori de la forme du sol (hypothèse planaire) nous permettrait de faire une détection précise sans éliminer les obstacles les plus proches.

A moyen terme, nous voulons améliorer la méthode d'étalonnage entre le télémètre laser et la caméra 3D en se basant sur une démarche similaire à celle de [Geiger *et al.*, 2012] pour les caméras conventionnelles. En effet, plutôt que d'utiliser de multiples scènes

contenant un plan étalon particulier, nous pensons qu'il est possible d'exploiter les plans multiples naturellement contenus dans une scène standard en milieu intérieur. Ce type de méthode offrirait un double avantage : la non nécessité d'un plan étalon et la réduction du nombre de prises de vues nécessaires à un étalonnage.

Nous souhaitons aussi travailler sur l'étalonnage des paramètres intrinsèques des caméras 3D (basé ou non sur un plan étalon 3D). Nous pensons que les paramètres intrinsèques du modèle du sténopé n'ont qu'une influence mesurée sur l'application cible lorsque ces paramètres intrinsèques sont les mêmes dans l'étalonnage extrinsèque et dans l'application utilisant l'étalonnage. Pour autant, il est évident qu'un étalonnage intrinsèque est nécessaire pour obtenir les nuages de points métriques les plus précis possibles. Ainsi, nous voulons mesurer l'impact de la précision de l'étalonnage intrinsèque sur nos méthodes d'étalonnage extrinsèque. Enfin, l'utilisation conjointe de méthodes d'étalonnage intrinsèques/extrinsèques permettra peut-être de mieux expliquer la différence que nous avons mise en évidence entre les images de la caméra IR et de la caméra 3D [Devaux *et al.*, 2013].

A long terme, nous pouvons faire ressortir plusieurs axes de recherche. De nombreux travaux ont mis en évidence une distorsion (de disparité) présente sur les mesures de profondeur des caméras 3D à bas coût [Smisek *et al.*, 2013]. Les méthodes actuelles calculant un coefficient local de correction (pour chaque pixel ou groupe de pixels), nous considérons que l'état de l'art n'est pas satisfaisant à ce sujet. Le premier axe consisterait donc à proposer un modèle paramétrique (avec un faible nombre de paramètres) corrigeant ces distorsions ainsi qu'une méthode d'estimation de ces paramètres c'est-à-dire un étalonnage intrinsèque incluant les distorsions. Une première observation des résidus de modélisation d'un plan permet d'observer un phénomène de bandes ainsi que des formes approchant les composantes radiales et tangentielles des distorsions optiques.

S'agissant de l'utilisation conjointe du télémètre laser avec une caméra 3D, nous nous posons la question de l'avantage potentiel de l'utilisation de n caméras 3D. Dans [Meilland *et al.*, 2014], les auteurs proposent notamment l'utilisation de 3 paires de caméras stéréos pour créer une caméra sphérique RGB-D. La densification des données pose la question

du temps de calcul qui est un élément critique de l'utilisation de la caméra 3D dans des systèmes critiques et réactifs comme l'évitement d'obstacles. Dans cette thèse nous avons montré que le formalisme du télémètre laser offre une réduction de taille des données d'environ 300 fois par rapport à une caméra 3D à bas coût tout en conservant une description précise d'un environnement intérieur. Nous souhaitons ainsi comparer, à densité de données identique, l'apport respectif du couple caméra 3D et télémètre laser et celui de n caméras 3D utilisées conjointement. Un télémètre virtuel à 360°, basé sur n caméras 3D, devrait idéalement pouvoir corriger les distorsions intrinsèques évoquées précédemment ainsi que les distorsions *dynamiques* induites par le mouvement des capteurs et le temps d'intégration des images.

Si la réduction de la quantité d'information sans dégrader l'information conservée paraît convenir aux applications de type évitement d'obstacles, nous pensons que la compression sans perte des données est une autre piste lorsqu'une importante densité des données est nécessaire. Toutefois, ce sujet reste un défi à relever puisque les algorithmes de compression vidéo classiques sont a priori peu adaptés à ce type de flux car ils induisent généralement des pertes. L'exemple le plus frappant auquel nous avons été confrontés est la compression MJPEG qui génère des artefacts dans les images menant à une importante modification du nuage de points final. L'une des pistes actuelles est la compression H.264 sans perte mais les flux restent tout de même de taille importante [Nenci *et al.*, 2014]. Une meilleure compression du flux d'images de pronfondeur permettra de multiplier ce type de capteur sur des robots de service à bas coût.

L'interface de téléopération en réalité augmentée que nous avons proposée présente a priori de nombreux avantages par rapport à une interface de contrôle classique. Nous avons évoqué une diminution de la charge cognitive de l'opérateur et une meilleure sûreté de navigation. Nous souhaitons réaliser une évaluation complète de cette interface en la soumettant à des tests avec les utilisateurs finaux (services de télévigilance du SAMU). Nous souhaitons notamment évaluer l'impact sur la charge cognitive en soumettant les opérateurs à des exercices parallèles à la téléopération du robot.

Finalement, nous pouvons aussi nous interroger sur la relation entre le type d'application envisagé et le positionnement relatif entre les capteurs. Dans cette thèse, nous avons, par *bon sens*, choisi de placer la caméra 3D en hauteur et dirigée vers le sol pour l'application d'évitement d'obstacles. Une telle configuration ne peut être possible pour le SLAM RGB-D où l'emplacement des caméras conventionnelles et 3D doivent maximiser leur champ de vision commun. Selon l'application envisagée, on peut imaginer qu'il existe une configuration optimale pour laquelle l'usage conjoint de la caméra 3D avec le télémètre laser ou la caméra conventionnelle optimise l'information utile disponible pour améliorer l'autonomie du robot ou la perception de la situation par l'utilisateur. Enfin, cette question de la disposition des capteurs n'est pas sans influence sur l'étalonnage pour lequel il est nécessaire d'avoir des observations communes et donc des champs de vision coïncidant au moins partiellement.

Annexes

Annexe A

Communications scientifiques

Conférences internationales à comité de lecture

Fully automatic extrinsic calibration of RGB-D system using two views of natural scene
Jean-Clément Devaux, Hicham Hadj-Abdelkader et Etienne Colle
13th International Conference on Control, Automation, Robotics and Vision, ICARCV 2014, IEEE RAS, Singapore, Singapore

A multi-sensor calibration toolbox for Kinect : Application to Kinect and laser range finder fusion
Jean-Clément Devaux, Hicham Hadj-Abdelkader et Etienne Colle
16th International Conference On Advanced Robotics, ICAR 2013, IEEE RAS, Montevideo, Uruguay

High level assisted control mode based on SLAM for a remotely controlled robot
Jean-Clément Devaux, Paul Nadrag, Etienne Colle, Philippe Hoppenot
15th International Conference On Advanced Robotics, ICAR 2011, IEEE RAS, Tallinn, Estonia

Conférences nationales avec actes

Toolbox d'étalonnage pour Kinect : Application à la fusion d'une Kinect et d'un télémètre laser
Jean-Clément Devaux, Hicham Hadj-Abdelkader et Etienne Colle
19ème congrès national sur la Reconnaissance de Formes et l'Intelligence Artificielle, RFIA 2014, Rouen, France.

Apport de la robotique ambiante pour l'analyse à distance d'une situation d'urgence d'une personne en détresse à domicile
Jean-Clément Devaux, Maxime Jubert et Etienne Colle
1ères Journées d'Etude sur la TéléSanté, JETSAN 2013, IEEE EMBS France, Fontainebleau, France.

Communications nationales

Etalonnage extrinsèque Laser 2D - Caméra 3D - Caméra conventionnelle
Jean-Clément Devaux, Hicham Hadj-Abdelkader et Etienne Colle
1ère Journée d'Etude sur les Caméras 3D, JEC3D'13, Paris, France

Etalonnage extrinsèque automatique Caméra 3D - Caméra conventionnelle sur scène naturelle
Jean-Clément Devaux, Hicham Hadj-Abdelkader et Etienne Colle
Journée Analyse conjointe RGB-Depth, GDR ISIS, Paris, France

Rapports techniques

Extrinsic calibration between a conventional and a Kinect-like virtual depth cameras
Jean-Clément Devaux, Maxime Boucher, Hicham Hadj-Abdelkader, Fakhr-Eddine Ababsa, Etienne Colle et Malik Mallem
Technical report, IBISC Lab, team IRA2, University of Evry-Val-d'Essonne

Annexe B

Exemples de scènes pour le télémètre virtuel

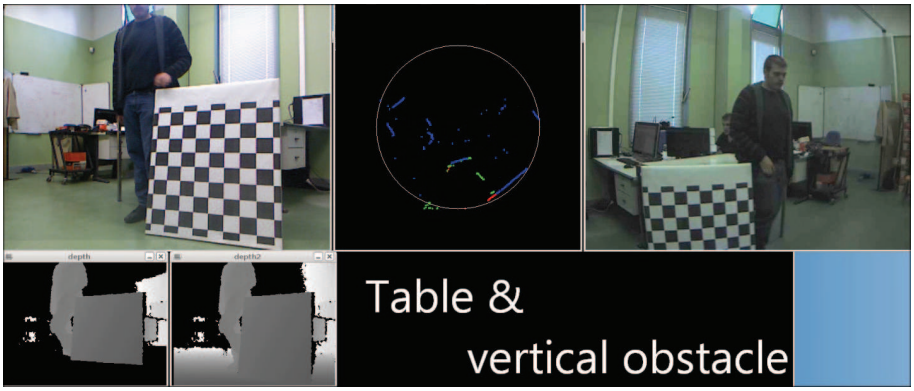


FIGURE B.1 – Scène 1 : Obstacle vertical

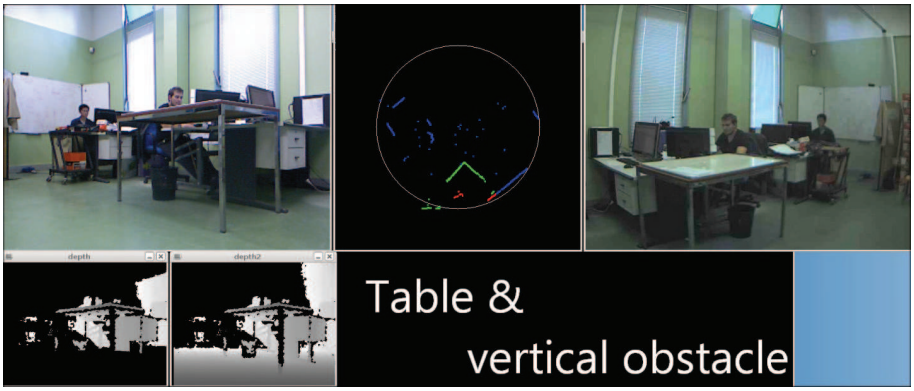


FIGURE B.2 – Scène 2 : Obstacle de type table

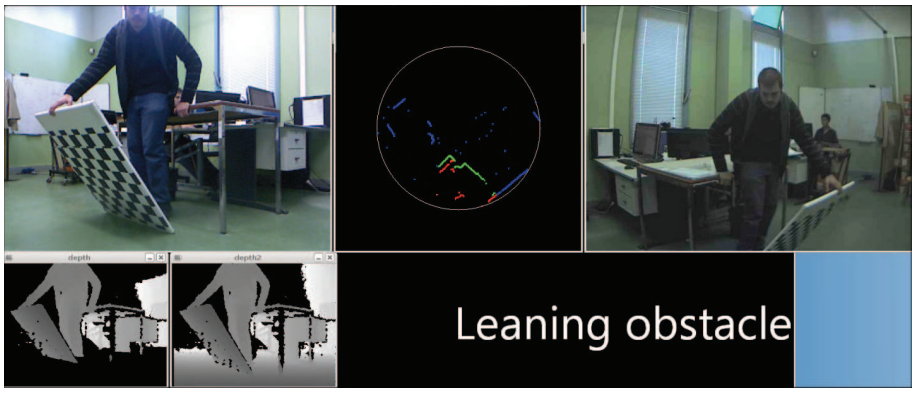


FIGURE B.3 – Scène 3 : Obstacle incliné

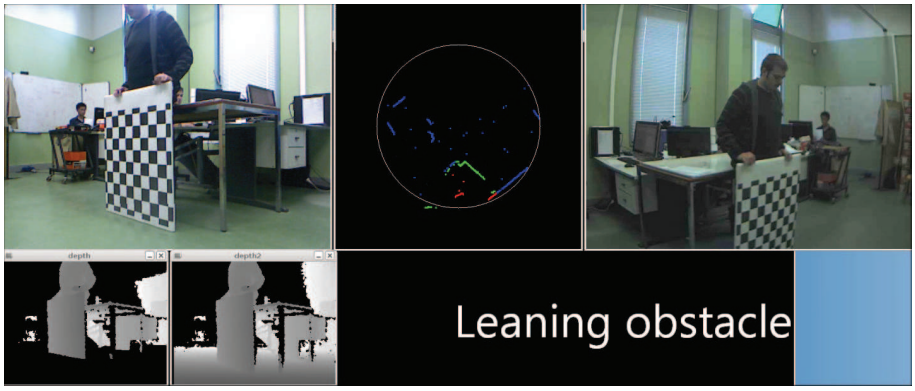


FIGURE B.4 – Scène 4 : Obstacle droit

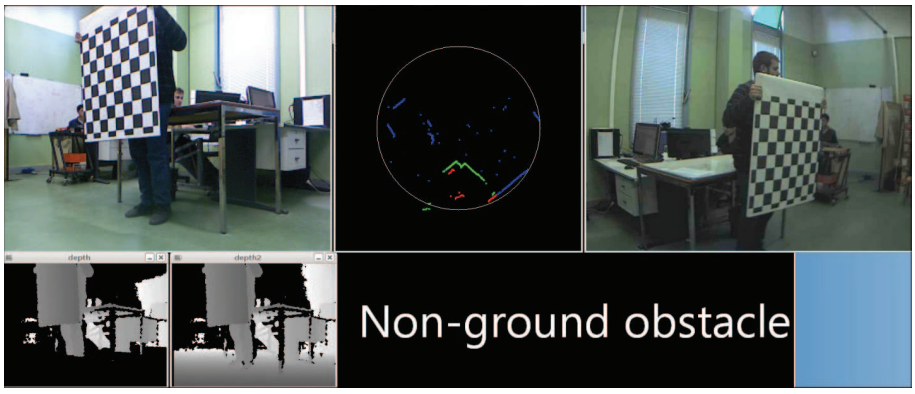


FIGURE B.5 – Scène 5 : Obstacle non posé sur le sol

Bibliographie

[Alenya *et al.*, 2009] Alenya, G., Nègre, A. et Crowley, J. L. (2009). A comparison of three methods for measure of time to contact. *In Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 4565–4570. IEEE.

[Alwan *et al.*, 2005] Alwan, M., Wagner, M. B., Wasson, G. et Sheth, P. (2005). Characterization of infrared range-finder pbs-03jn for 2-d mapping. *In Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 3936–3941. IEEE.

[Arkin, 1998] Arkin, R. C. (1998). *Behavior-based robotics*. MIT press.

[Bailey et Durrant-Whyte, 2006] Bailey, T. et Durrant-Whyte, H. (2006). Simultaneous localization and mapping (slam) : Part ii. *IEEE Robotics & Automation Magazine*, 13(3):108–117.

[Baribeau et Rioux, 1991] Baribeau, R. et Rioux, M. (1991). Influence of speckle on laser range finders. *Applied Optics*, 30(20):2873–2878.

[Bauer *et al.*, 2007] Bauer, J., Sunderhauf, N. et Protzel, P. (2007). Comparing several implementations of two recently published feature detectors. *In Proc. of the International Conference on Intelligent and Autonomous Systems, IAV, Toulouse, France*, pages 224–227.

[Bay *et al.*, 2008] Bay, H., Ess, A., Tuytelaars, T. et Van Gool, L. (2008). Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359.

[Borenstein et Koren, 1991] Borenstein, J. et Koren, Y. (1991). The vector field histogram-fast obstacle avoidance for mobile robots. *Robotics and Automation, IEEE Transactions on*, 7(3):278–288.

[Boucher, 2013] Boucher, A. (2013). Segmentation d'un orateur sur vidéo rgb-depth. Analyse conjointe des images RGB-Depth, GDR ISIS 2013.

[Boucher, 2014] Boucher, M. (2014). *Quelques contributions en localisation et cartographie simultannée multi-capteurs : application à la réalité augmentée*. Thèse de doctorat, Université d'Evry Val d'Essonne.

[Bouguet, 2003] Bouguet, J.-Y. (2003). Camera calibration toolbox for matlab. `http://www.vision.caltech.edu/bouguetj/calib_doc/`.

[Burrus, 2011] Burrus, N. (2011). Kinect calibration. `http://nicolas.burrus.name/index.php/Research/KinectCalibration`.

[Carballo *et al.*, 2008] Carballo, A., Ohya, A. et Yuta, S. (2008). Fusion of double layered multiple laser range finders for people detection from a mobile robot. *In Multisensor*

Fusion and Integration for Intelligent Systems, 2008. MFI 2008. IEEE International Conference on, pages 677–682. IEEE.

[Chaumette et Hutchinson, 2006] Chaumette, F. et Hutchinson, S. (2006). Visual servo control. i. basic approaches. *Robotics & Automation Magazine, IEEE*, 13(4):82–90.

[Costantini et Susstrunk, 2004] Costantini, R. et Susstrunk, S. (2004). Virtual sensor design. *In Electronic Imaging 2004*, pages 408–419. International Society for Optics and Photonics.

[De la Escalera et Armingol, 2010] De la Escalera, A. et Armingol, J. M. (2010). Automatic chessboard detection for intrinsic and extrinsic camera parameter calibration. *Sensors*, 10(3):2027–2044.

[Delarue *et al.*, 2010] Delarue, S., Nadrag, P., Hoppenot, P. et Colle, E. (2010). Remote control of an assistive robot : three guidelines for designing shared control modes. *AMSE*, 71(3-4):59–67.

[Devaux *et al.*, 2013] Devaux, J.-C., Boucher, M., Hadj-Abdlekader, H., Ababsa, F.-E., Colle, E. et Mallem, M. (2013). Extrinsic calibration between a conventional and a kinect-like virtual depth cameras. Rapport technique, University of Evry, IBISC Laboratory.

[Devaux *et al.*, 2011] Devaux, J.-C., Nadrag, P., Colle, E. et Hoppenot, P. (2011). High level assisted control mode based on slam for a remotely controlled robot. *In Advanced Robotics (ICAR), 2011 15th International Conference on*, pages 186–191. IEEE.

[Diosi et Kleeman, 2004] Diosi, A. et Kleeman, L. (2004). Advanced sonar and laser range finder fusion for simultaneous localization and mapping. *In Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 2, pages 1854–1859. IEEE.

[Dorsch *et al.*, 1994] Dorsch, R. G., Häusler, G. et Herrmann, J. M. (1994). Laser triangulation : fundamental uncertainty in distance measurement. *Applied Optics*, 33(7): 1306–1314.

[Dupont *et al.*, 2005] Dupont, R., Keriven, R. et Fuchs, P. (2005). An improved calibration technique for coupled single-row telemeter and ccd camera. *In Proceedings of the Fifth International Conference on 3-D Digital Imaging and Modeling*, 3DIM '05, pages 89–94, Washington, DC, USA. IEEE Computer Society.

[Endres *et al.*, 2014] Endres, F., Hess, J., Sturm, J., Cremers, D. et Burgard, W. (2014). 3D mapping with an RGB-D camera. *IEEE Transactions on Robotics*, 30(1):177–187.

[Engelhard *et al.*, 2011] Engelhard, N., Endres, F., Hess, J., Sturm, J. et Burgard, W. (2011). Real-time 3D visual SLAM with a hand-held camera. *In Proc. of the RGB-D Workshop on 3D Perception in Robotics at the European Robotics Forum*, Vasteras, Sweden.

[Estrada *et al.*, 2005] Estrada, C., Neira, J. et Tardós, J. D. (2005). Hierarchical slam : real-time accurate mapping of large environments. *Robotics, IEEE Transactions on*, 21(4):588–596.

[Faugeras, 1993] Faugeras, O. (1993). *Three dimensional computer vision : A geometric viewpoint*. the MIT Press.

[Flach *et al.*, 1998] Flach, J. M., Tanabe, F., Monta, K., Vicente, K. J. et Rasmussen, J. (1998). An ecological approach to interface design. *In Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 42, pages 295–299. SAGE Publications.

[Fox *et al.*, 1997] Fox, D., Burgard, W. et Thrun, S. (1997). The dynamic window approach to collision avoidance. *IEEE Robotics & Automation Magazine*, 4(1):23–33.

[Geiger *et al.*, 2012] Geiger, A., Moosmann, F., Car, O. et Schuster, B. (2012). Automatic camera and range sensor calibration using a single shot. *In ICRA*, pages 3936–3943. IEEE.

[Golub et Van Loan, 1996] Golub, G. H. et Van Loan, C. F. (1996). *Matrix computations (3rd ed.)*. Johns Hopkins University Press, Baltimore, MD, USA.

[Goodrich, 2002] Goodrich, M. A. (2002). Potential fields tutorial.

[Goodridge et Luo, 1994] Goodridge, S. G. et Luo, R. C. (1994). Fuzzy behavior fusion for reactive control of an autonomous mobile robot : Marge. *In Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, pages 1622–1627. IEEE.

[Hall et Llinas, 1997] Hall, D. L. et Llinas, J. (1997). An introduction to multisensor data fusion. *Proceedings of the IEEE*, 85(1):6–23.

[Hartley et Zisserman, 2000] Hartley, R. et Zisserman, A. (2000). *Multiple view geometry in computer vision*, volume 2. Cambridge Univ Press.

[Herrera *et al.*, 2012] Herrera, C., Kannala, J. *et al.* (2012). Joint depth and color camera calibration with distortion correction. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(10):2058–2064.

[Hoppenot *et al.*, 1996] Hoppenot, P., Benreguieg, M., Maaref, H., Colle, E. et Barret, C. (1996). Control of a medical aid mobile robot based on a fuzzy navigation. *In CESA'96 IMACS Multiconference : computational engineering in systems applications*, pages 388–393.

[Hu *et al.*, 2012] Hu, G., Huang, S., Zhao, L., Alempijevic, A. et Dissanayake, G. (2012). A robust rgb-d slam algorithm. *In Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 1714–1719. IEEE.

[Ilonen et Kyrki, 2011] Ilonen, J. et Kyrki, V. (2011). Robust robot-camera calibration. *In Advanced Robotics (ICAR), 2011 15th International Conference on*, pages 67–74. IEEE.

[Jokinen, 1999] Jokinen, O. (1999). Self-calibration of a light striping system by matching multiple 3-d profile maps. *In Proceedings of the 2nd international conference on 3-D digital imaging and modeling*, 3DIM'99, pages 180–190, Washington, DC, USA. IEEE Computer Society.

[Jordt *et al.*, 2009] Jordt, A., Siebel, N. T. et Sommer, G. (2009). Automatic high-precision self-calibration of camera-robot systems. *In Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 1244–1249. IEEE.

[Kabadayi *et al.*, 2006] Kabadayi, S., Pridgen, A. et Julien, C. (2006). Virtual sensors : Abstracting data from physical sensors. *In Proceedings of the 2006 International Symposium on on World of Wireless, Mobile and Multimedia Networks*, pages 587–592. IEEE Computer Society.

[Kabsch, 1976] Kabsch, W. (1976). A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A : Crystal Physics, Diffraction, Theoretical and General Crystallography*, 32(5):922–923.

[Khoshelham et Elberink, 2012] Khoshelham, K. et Elberink, S. O. (2012). Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors*, 12(2): 1437–1454.

[Kim et Kim, 2003] Kim, D.-H. et Kim, J.-H. (2003). A real-time limit-cycle navigation method for fast mobile robots and its application to robot soccer. *Robotics and Autonomous Systems*, 42(1):17–30.

[Kitt *et al.*, 2010] Kitt, B., Geiger, A. et Lategahn, H. (2010). Visual odometry based on stereo image sequences with ransac-based outlier rejection scheme. *In Intelligent Vehicles Symposium (IV), 2010 IEEE*, pages 486–492. IEEE.

[Kweon *et al.*, 1991] Kweon, I. S., Hoffman, R. et Krotkov, E. (1991). Experimental characterization of the perceptron laser rangefinder. Rapport technique, DTIC Document.

[Lee, 1990] Lee, C.-C. (1990). Fuzzy logic in control systems : fuzzy logic controller. ii. *Systems, Man and Cybernetics, IEEE Transactions on*, 20(2):419–435.

[Levenberg, 1944] Levenberg, K. (1944). A method for the solution of certain problems in least squares. *Quart. Applied Math.*, 2:164–168.

[Long *et al.*, 1999] Long, T. W., Hanzevack, E. L. et Bynum, W. L. (1999). Sensor fusion and failure detection using virtual sensors. *In American Control Conference, 1999. Proceedings of the 1999*, volume 4, pages 2417–2421. IEEE.

[Luo *et al.*, 2002] Luo, R. C., Yih, C.-C. et Su, K. L. (2002). Multisensor fusion and integration : approaches, applications, and future research directions. *Sensors Journal, IEEE*, 2(2):107–119.

[Maddern *et al.*, 2012] Maddern, W., Harrison, A. et Newman, P. (2012). Lost in translation (and rotation) : Rapid extrinsic calibration for 2d and 3d lidars. *In Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 3096–3102. IEEE.

[Mantel *et al.*, 2012] Mantel, B., Hoppenot, P. et Colle, E. (2012). Perceiving for acting with teleoperated robots : Ecological principles to human–robot interaction design. *Systems, Man and Cybernetics, Part A : Systems and Humans, IEEE Transactions on*, 42(6):1460–1475.

[Marquardt, 1963] Marquardt, D. W. (1963). An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal on Applied Mathematics*, 11(2):431–441.

[Martin *et al.*, 2006] Martin, C., Schaffernicht, E., Scheidig, A. et Gross, H.-M. (2006). Multi-modal sensor fusion using a probabilistic aggregation scheme for people detection and tracking. *Robotics and Autonomous Systems*, 54(9):721–728.

[McIvor, 1999] McIvor, A. M. (1999). Calibration of a laser stripe profiler. *In Proceedings of the 2nd international conference on 3-D digital imaging and modeling*, 3DIM'99, pages 92–98, Washington, DC, USA. IEEE Computer Society.

[Meilland *et al.*, 2014] Meilland, M., Comport, Andrew, I. et Rives, P. (2014). Dense omnidirectional RGB-D mapping of large scale outdoor environments for real-time localisation and autonomous navigation. *Journal of Field Robotics*.

[Moravec, 1988] Moravec, H. P. (1988). Sensor fusion in certainty grids for mobile robots. *AI magazine*, 9(2):61.

[Nadrag, 2011] Nadrag, P. (2011). *Faciliter la téléopération d'un robot mobile non-holonome*. Thèse de doctorat, Université d'Evry-Val d'Essonne.

[Nenci *et al.*, 2014] Nenci, F., Spinello, L. et Stachniss, C. (2014). Effective compression of range data streams for remote robot operations using h. 264. To appear, IROS'14.

[Newcombe *et al.*, 2011] Newcombe, R. A., Davison, A. J., Izadi, S., Kohli, P., Hilliges, O., Shotton, J., Molyneaux, D., Hodges, S., Kim, D. et Fitzgibbon, A. (2011). Kinectfusion : Real-time dense surface mapping and tracking. *In Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*, pages 127–136. IEEE.

[Okubo *et al.*, 2009] Okubo, Y., Ye, C. et Borenstein, J. (2009). Characterization of the hokuyo urg-04lx laser rangefinder for mobile robot obstacle negotiation. *In SPIE Defense, Security, and Sensing*, pages 733212–733212. International Society for Optics and Photonics.

[Pascoal *et al.*, 2008] Pascoal, J., Marques, L. et de Almeida, A. T. (2008). Assessment of laser range finders in risky environments. *In Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 3533–3538. IEEE.

[Pito, 1995] Pito, R. (1995). Characterization, calibration, and use of the perceptron laser range finder in a controlled environment. Rapport technique, Philadelphia, USA : University of Pennsylvania, Department of Computer and Information Science, GRASP Laboratory.

[Samson *et al.*, 1991] Samson, C., Espiau, B. et Borgne, M. L. (1991). *Robot Control : The Task Function Approach*. Oxford University Press.

[Scaramuzza *et al.*, 2007] Scaramuzza, D., Harati, A. et Siegwart, R. (2007). Extrinsic self calibration of a camera and a 3d laser range finder from natural scenes. *In Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 4164–4169. IEEE.

[Shafer *et al.*, 1986] Shafer, S., Stentz, A. et Thorpe, C. E. (1986). An architecture for sensor fusion in a mobile robot. *In Robotics and Automation. Proceedings. 1986 IEEE International Conference on*, volume 3, pages 2002–2011. IEEE.

[Smisek *et al.*, 2013] Smisek, J., Jancosek, M. et Pajdla, T. (2013). 3d with kinect. *In Consumer Depth Cameras for Computer Vision*, pages 3–25. Springer.

[Stepan *et al.*, 2005] Stepan, P., Kulich, M. et Preucil, L. (2005). Robust data fusion with occupancy grid. *Systems, Man, and Cybernetics, Part C : Applications and Reviews, IEEE Transactions on*, 35(1):106–115.

[Steux et El Hamzaoui, 2010] Steux, B. et El Hamzaoui, O. (2010). tinyslam : A slam algorithm in less than 200 lines c-language program. *In Control Automation Robotics & Vision (ICARCV), 2010 11th International Conference on*, pages 1975–1979. IEEE.

[Sturm *et al.*, 2012] Sturm, J., Engelhard, N., Endres, F., Burgard, W. et Cremers, D. (2012). A benchmark for the evaluation of rgb-d slam systems. *In Proc. of the International Conference on Intelligent Robot Systems (IROS)*.

[Teichman *et al.*, 2013] Teichman, A., Miller, S. et Thrun, S. (2013). Unsupervised intrinsic calibration of depth sensors via slam. *In Robotics : Science and Systems (RSS)*.

[Thacker et Mayhew, 1991] Thacker, N. A. et Mayhew, J. E. (1991). Optimal combination of stereo camera calibration from arbitrary stereo images. *Image and vision computing*, 9(1):27–32.

[Thrun, 1998] Thrun, S. (1998). Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 99(1):21–71.

[Thrun *et al.*, 1999] Thrun, S., Bennewitz, M., Burgard, W., Cremers, A. B., Dellaert, F., Fox, D., Hahnel, D., Rosenberg, C., Roy, N., Schulte, J. *et al.* (1999). Minerva : A second-generation museum tour-guide robot. *In Robotics and automation, 1999. Proceedings. 1999 IEEE international conference on*, volume 3. IEEE.

[Tsai, 1987] Tsai, R. Y. (1987). A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *Robotics and Automation, IEEE Journal of*, 3(4):323–344.

[Unnikrishnan, 2006] Unnikrishnan, R. (2006). Laser-camera calibration toolbox for matlab (lcct). Rapport technique, Carnegie Mellon Robotics Institute.

[Unnikrishnan et Hebert, 2005] Unnikrishnan, R. et Hebert, M. (2005). Fast extrinsic calibration of a laser rangefinder to a camera. Rapport technique, Carnegie Mellon Robotics Institute.

[Whelan *et al.*, 2013] Whelan, T., Johannsson, H., Kaess, M., Leonard, J. J. et McDonald, J. (2013). Robust real-time visual odometry for dense rgb-d mapping. *In Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 5724–5731. IEEE.

[Xu et Zhang, 1996] Xu, G. et Zhang, Z. (1996). *Epipolar geometry in stereo, motion and object recognition : a unified approach*, volume 6. Springer.

[Ye, 2008] Ye, C. (2008). Mixed pixels removal of a laser rangefinder for mobile robot 3-d terrain mapping. *In Information and Automation, 2008. ICIA 2008. International Conference on*, pages 1153–1158. IEEE.

[Ye et Borenstein, 2002] Ye, C. et Borenstein, J. (2002). Characterization of a 2d laser scanner for mobile robot obstacle negotiation. *In Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, volume 3, pages 2512–2518. IEEE.

[Zeng *et al.*, 2013] Zeng, M., Zhao, F., Zheng, J. et Liu, X. (2013). Octree-based fusion for realtime 3d reconstruction. *Graphical Models*, 75(3):126–136.

[Zhang et Pless, 2004] Zhang, Q. et Pless, R. (2004). Extrinsic calibration of a camera and laser range finder (improves camera calibration). *In Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, volume 3 de *IROS '04*, pages 2301–2306.

[Zhang, 1999] Zhang, Z. (1999). Flexible camera calibration by viewing a plane from unknown orientations. *In International Conference on Computer Vision*, volume 1, page 666, Corfu, Greece.

[Zhang, 2000] Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:1330–1334.