HAL
open science

Qualification et amélioration de la précision de systèmes de balayage laser mobiles par extraction d'arêtes

Martyna Poreba

To cite this version:

Martyna Poreba. Qualification et amélioration de la précision de systèmes de balayage laser mobiles par extraction d'arêtes. Autre [cs.OH]. Ecole Nationale Supérieure des Mines de Paris, 2014. Français. NNT : 2014ENMP0012 . pastel-01068828

HAL Id: pastel-01068828

https://pastel.hal.science/pastel-01068828v1

Submitted on 26 Sep 2014

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

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

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

MINES ParisTech

École doctorale n° 432 : Sciences des Métiers de l'Ingénieur

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

l'École nationale supérieure des mines de Paris

Spécialité « Informatique temps-réel, robotique et automatique »

présentée et soutenue publiquement par

Martyna POREBA

le 18 juin 2014

Qualification et amélioration de la précision de systèmes de balayage laser mobiles par extraction d'arêtes

~~~

Edge-based accuracy assessment and improvement of mobile laser scanning systems

Directeur de thèse : **François GOULETTE**

Jury

M. Pierre GRUSSENMEYER, Professeur, INSA de Strasbourg	Rapporteur
M. Nicolas PAPARODITIS Directeur de recherche, IGN	Rapporteur
M. Laurent TRASSOUDAINE, Professeur, Université Blaise Pascal de Clermont-Ferrand	Président du jury
M. Jean-Emmanuel DESCHAUD, Docteur, MINES ParisTech	Examinateur
M. François GOULETTE, Professeur, MINES ParisTech	Directeur de thèse

T H È S E

MINES ParisTech
CAOR, Centre de Robotique – Département Mathématiques et Systèmes
60, Boulevard Saint Michel, 75272 Paris cedex 06, France
~~~

Remerciements

Je tiens tout d'abord à remercier François Goulette, mon directeur de thèse, pour toute son aide, sa disponibilité et le suivi qu'il m'a apportés le long de ma thèse, tout en me laissant une grande autonomie.

Mes vifs remerciements vont aux membres de jury qui ont accepté de juger mes travaux. En particulier, je remercie chaleureusement M. Pierre Grussenmeyer et M. Nicolas Paparoditis d'avoir rapporté mon travail de thèse. Leurs remarques précieuses m'ont permis d'approfondir des explications sur certaines parties et d'envisager mon travail sous un autre angle. Je remercie également M. Laurent Trassoudaine de m'avoir fait l'honneur de présider mon jury de thèse. Enfin, je remercie M. Jean-Emmanuel Deschaud d'avoir participé à mon jury.

Aussi, je voudrais exprimer ma sincère gratitude envers toute l'équipe du CAOR pour la bonne ambiance et l'accueil qui m'a été réservée. Notamment, je remercie Arnaud de la Fortelle de m'avoir donné l'opportunité d'intégrer son laboratoire et de travailler sur cette thèse.

Une pensée particulière va à mes collègues doctorants et post-doctorants, pour l'encouragement et les formidables moments passés ensemble. Merci à vous tous d'avoir partagé avec moi ces moments de travail et de détente. Une spéciale dédicace à Jorge, Olivier, Martin, Fernando, Bruno, Zhuowei, Tao-Jin, Axel, Houssem, Victor, Anne-Sophie et Fatin.

Cette thèse n'aurait pu aboutir sans le soutien financier de l'Ambassade de France en Pologne, facilité par l'actuel adjoint à l'attaché pour la science et la technologie M. Nicolas Frichot-Manoury.

Résumé

Au cours de ces dernières décennies, le développement de Systèmes Mobiles de Cartographie, soutenu par un progrès technologique important, est devenu plus proéminent. Il a été stimulé par le besoin grandissant de collecte d'informations géographiques précises sur l'environnement. Nous considérons, au sein de cette thèse, des solutions pour l'acquisition des nuages de points mobiles de qualité topographique (précision centimétrique). Il s'agit, dans cette tâche, de mettre au point des méthodes de qualification des données, et d'en améliorer les erreurs systématiques par des techniques d'étalonnage et de recalage adéquates.

Nous décrivons une nouvelle démarche d'évaluation de l'exactitude et/ou de la précision des relevés laser mobiles. Celle-ci repose sur l'extraction et la comparaison des entités linéaires de la scène urbaine. La distance moyenne calculée entre les segments appariés, étant la distance modifiée de *Hausdorff*, sert à noter les nuages par rapport à des références existantes. Pour l'extraction des arêtes, nous proposons une méthode de détection d'intersections entre segments plans retrouvés via un algorithme de *RANSAC* enrichi d'une analyse de composantes connexes. Nous envisageons également une démarche de correction de relevés laser mobiles à travers un recalage rigide fondé, lui aussi, sur les éléments linéaires. Enfin, nous étudions la pertinence des arêtes pour en déduire les paramètres de l'étalonnage extrinsèque du système mobile. Nous testons nos méthodes sur des données simulées et des données réelles acquises dans le cadre du projet TerraMobilita.

Mots clés : Nuage de points, Système Mobile de Cartographie, Exactitude, Précision, Arête, Segmentation, Évaluation, RANSAC, Recalage, Étalonnage, Mise en correspondance

Abstract

Over the past few decades, the development of Mobile Mapping Systems (MMS), supported by significant technological progress, has become more prominent. It has been motivated by the growing need for precise geographic information about the environment. In this thesis, we consider approaches for the acquisition of mobile point clouds with topographic quality (centimeter-level accuracy). The aim is to develop techniques for data quality assessment and improvement. In particular, we eliminate the systematic errors inherent to an MMS data using appropriate calibration and registration methods.

We describe a novel approach to assess the accuracy and/or the precision of mobile laser point clouds. It is based on the extraction and comparison of line features detected within the urban scene. The computed average distance between corresponding pairs of line segments, taking advantage of a modified Hausdorff distance, is used to evaluate the MMS data with respect to a reference data set. For edge extraction, we propose a method which relies on the intersections between planes modelled via the RANSAC algorithm refined by an analysis of connected components. We also consider an approach to correct point clouds using a line-based rigid registration method. Finally, we study the use of edges for estimating the boresight angles of a land-based mobile mapping system. We apply our methods to synthetic data and to real data acquired as part of the TerraMobilita project.

Keywords: Point Cloud, Land-based mobile mapping system (MMS), Accuracy, Precision, Edge, Segmentation, Assessment, RANSAC, Registration, Calibration, Matching

Table des matières

1. Introduction 7
1.1 Contexte général 7
1.1.1 Projet TerraMobilita 8
1.1.2 Motivations 9
1.2 Apports de la thèse 12
1.3 Plan du mémoire 13

2. Acquisition et qualification de relevés laser 15
2.1 Systèmes mobiles terrestres voués à la numérisation d'environnement 16
2.1.1 Principes fondamentaux d'un système mobile de cartographie 18
2.1.2 Description du prototype L3D2 21
2.1.3 Autres exemples de systèmes mobiles de cartographie 23
2.2 Sources de données expérimentales 24
2.3 Qualification de données 28
2.3.1 Différents types d'erreurs en topographie 28
2.3.2 Critères servant à décrire la qualité de nuages mobiles 31
2.3.3 État de l'art sur l'évaluation de l'exactitude de relevés laser mobiles 32
2.4 Expérimentation 35

3. Erreurs relatives au système MMS 39
3.1 Analyse des erreurs liées à la cartographie mobile 40
3.1.1 Erreurs causées par le système de positionnement 40
3.1.2 Erreurs dues au capteur laser 41
3.2 Amélioration des relevés laser mobiles 43
3.2.1 Correction de la localisation en cours d'acquisition 43
3.2.2 Correction en post-traitement 46
3.3 Étalonnage extrinsèque 47
3.3.1 État de l'art sur l'étalonnage des excentricités linéaires 48
3.3.2 Étalonnage des systèmes embarqués de caméras 49
3.3.3 Étalonnage des systèmes dotés de scanners 51
3.4 Impact d'un faux étalonnage sur la géométrie du nuage 54

4. Extraction d'arêtes 57
4.1 Méthode générale - vue d'ensemble 58
4.2 État de l'art sur l'extraction d'arêtes 59
4.3 Segmentation en éléments plans - état de l'art 61
4.3.1 Segmentation par croissance de surface (régions) 62
4.3.2 Transformée de *Hough* 3D 63
4.3.3 Algorithme de *RANSAC* 65
4.4 Contribution à la détection de plans 67

4.4.1 Motivation ... 68
4.4.2 Extraction séquentielle de plans ... 69
4.4.3 Connectivité de plans basée sur la théorie des graphes ... 71
4.5 Méthode proposée d'extraction d'arêtes ... 77
4.6 Exemples de résultats ... 82
4.6.1 Détection de plans ... 82
4.6.2 Extraction d'arêtes ... 84

5. Qualification à partir d'entités linéaires ... 87
5.1 Problématique ... 87
5.1.1 Comment mesurer la distance entre deux segments dans l'espace ? ... 88
5.1.2 Contraintes ... 90
5.2 Mesure de distance proposée ... 91
5.2.1 Distance de *Hausdorff* ... 92
5.2.2 Matrice de similarité ... 94
5.2.3 Mise en correspondance ... 96
5.3 Résultats et évaluation ... 99
5.3.1 Données expérimentales ... 99
5.3.2 Évaluation de l'algorithme d'appariement ... 100
5.3.3 Exactitude et précision ... 105

6. Recalage à partir d'entités linéaires ... 107
6.1 Introduction ... 108
6.2 État de l'art du recalage rigide ... 109
6.2.1 Mise en correspondance automatique ... 109
6.2.2 Classification des méthodes existantes de recalage ... 111
6.2.3 Aperçu des méthodes basées sur les entités linéaires ... 113
6.3 Recalage basé-lignes : caractérisation des méthodes ... 115
6.4 Évaluation des méthodes présentées ... 116
6.4.1 Données pour les expérimentations ... 117
6.4.2 Algorithmes implémentés ... 118
6.4.3 Outils d'évaluation ... 118
6.4.4 Résultats – recalage des données bruitées ... 119
6.4.5 Impact du bruit sur la distance de *Hausdorff* ... 122
6.5 Approche proposée : *RLMR-FMII2* ... 124
6.5.1 Recalage grossier – principe ... 126
6.5.2 Évaluation avec données simulées ... 127
6.5.3 Tests sur données réelles ... 130

7. Perspectives et conclusion ... 133
7.1 Conclusion ... 133
7.2 Perspectives et futurs travaux ... 136
7.2.1 Étalonnage extrinsèque du système : esquisse de solution ... 136

Publications ... 139

Bibliographie ... 141

Annexe A ... 153
Annexe B ... 157
Annexe C ... 161
Annexe D ... 163
Annexe E ... 169

CHAPITRE 1

Introduction

Sommaire :

1.1 Contexte général **7**
1.1.1 Projet TerraMobilita 8
1.1.2 Motivations 9
1.2 Apports de la thèse **11**
1.3 Plan du mémoire **12**

1.1 Contexte général

Depuis plusieurs siècles, les instruments essentiels de l'arpenteur et le savoir-faire de l'homme permettaient de cartographier la surface de la Terre. La technologie employée n'a cessé d'évoluer. Durant ce temps, nous notons quelques développements remarquables qui ont révolutionné la façon de mesurer. À ce titre, nous pouvons mentionner des mesures au moyen d'une lunette et d'une planchette (XVIIe siècle), celles réalisées à l'aide des photographies (XIXe siècle), des théodolites (XXe siècle), jusqu'à l'apparition de l'électronique et l'informatique qui ont donné naissance à de nouvelles techniques de levé et de traitement des données telles que la station totale, le positionnement par satellites et le télémètre laser. Cette évolution tire ainsi son origine dans les besoins sans cesse croissants d'informations tridimensionnelles, rapides et précises sur notre environnement. Cependant, nous observons que les méthodes dites conventionnelles dénotent quelques faiblesses, lorsqu'elles sont employées pour numériser des objets de grande dimension (au niveau des villes) ou de forme allongée (canyon urbain). La cartographie mobile par *LiDAR* (*Light Detection And Ranging*), intégrant ordinairement trois sous-systèmes : le télémètre laser, le système de navigation inertielle *INS* (*Inertial Navigation System)* et le système de positionnement par satellites *GNSS* (*Global Navigation Satellite System*), tous installés à l'intérieur d'une plate-forme mobile, a relégué les levés traditionnels par la vitesse de collecte de données à une toute autre place. Elle a permis d'accéder à des endroits où la réalisation des mesures serait pratiquement impossible avec d'autres méthodes. Dès lors, il est envisageable de produire rapidement des cartes topographiques, numériques et précises du terrain. Les systèmes aériens, installés dans un avion ou hélicoptère, sont capables de produire une impressionnante quantité de points 3D géo-référencés. Néanmoins, leur

richesse en détail restant limitée, la technologie seule ne permet pas de reconstruire des modèles 3D complets de ville. Les systèmes mobiles terrestres embarqués sur un camion sont alors perçus comme un compromis entre le balayage laser fixe et celui aéroporté. Toutefois, la rapidité des acquisitions présenterait peu d'intérêt si la qualité des données collectées n'était pas garantie. Afin de s'adapter aux nouvelles réalités, les systèmes mobiles de cartographie doivent être concurrentiels au regard des techniques conventionnelles. Or, nous insistons sur le fait que l'exactitude des points laser soit équivalente aux levés traditionnels, mais avec un temps d'acquisition et de traitement nettement plus rapide.

Pour cadrer avec cet objectif, c'est-à-dire de mettre au point des systèmes mobiles de qualité topographique, il est primordial d'être en mesure de qualifier l'exactitude et la précision des données collectées afin d'améliorer celles-ci sur les erreurs systématiques. Ces systèmes étant récents, et bien que cette question soit essentielle, ces aspects ont encore été peu étudiés.

1.1.1 Projet TerraMobilita

Notre travail est effectué dans le cadre du projet TerraMobilita, dont le but est de mettre au point de nouveaux processus automatiques de création et de mises à jour des cartes 3D de voirie urbaine, avec une précision centimétrique. Pour ce faire, des données laser et images, issues d'un système mobile de cartographie développé, entre autres, au sein du Centre de Robotique *CAOR* de MINES ParisTech, sont employées. Il s'agit alors de construire des modèles 3D complets de l'espace public afin de pouvoir étudier minutieusement l'encombrement de divers éléments (stationnement, poubelles, etc.), de gérer l'entretien des voiries, de produire des Plans de Déplacement Urbain (*PDU*) pour les besoins de circulations douces (piéton, vélo, rollers, poussette, chaise roulante, etc.) et de diagnostics d'accessibilité de la voirie pour les Personnes à Mobilité Réduite (*PMR*). L'objectif à atteindre est de fournir des solutions pour l'acquisition, le traitement de données 3D et la production de cartes, plus automatisées et moins coûteuses que les méthodes actuelles. Mais, on se doit de garder à l'esprit la contrainte de relevé couvrant plusieurs km² de l'espace public en un temps court, et de volume importante de données à traiter.

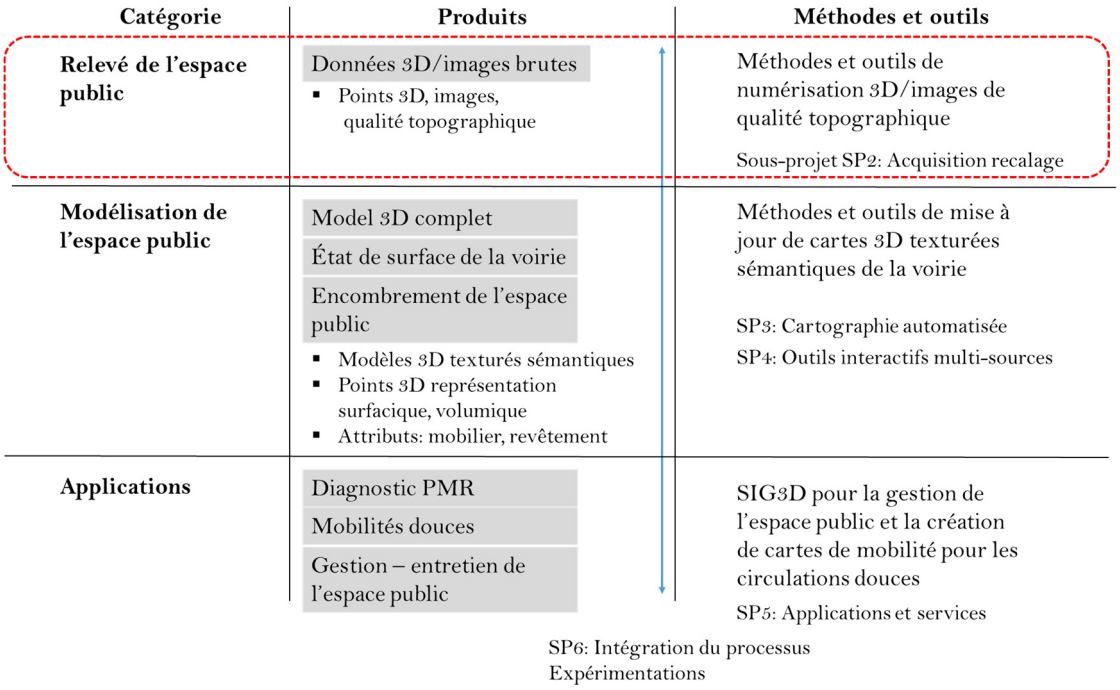

Figure 1.1 Diagramme de décomposition fonctionnelle du projet TerraMobilita.

1.1.2 Motivations

Le projet TerraMobilita intègre plusieurs sous-projets, répertoriés principalement en trois catégories dont nous présentons la décomposition fonctionnelle dans la Figure 1.1. La partie concernant nos travaux est encadrée en rouge. Il s'agit dans la tâche de ce sous-projet (SP2) de mettre au point des techniques de numérisation 3D de qualité topographique. Ainsi, le dessein de cette thèse est de qualifier des données laser brutes acquises en environnement urbain, mais aussi d'améliorer, par des méthodes adéquates de recalage et d'étalonnage, la plate-forme mobile en terme de précision.

Depuis l'avènement de la technologie laser, les géomètres sont obligés de recueillir des données de terrain selon une autre philosophie. Le relevé laser est acquis d'un point de vue surfacique et non plus par un choix de points caractéristiques comme c'est le cas avec un tachéomètre. Il est donc primordial de réfléchir, à quoi le terme « exactitude » doit se référer – à un point particulier du nuage, ou à un élément extrait à partir de nuages tel qu'une surface plane, une arête. Aussi, en dépit des gains de résolution et de précision des scanners laser récents, la nature des nuages de points (bruit, densité) peut toujours gêner son traitement (Figure 1.2). Cet état de fait devrait être pris en compte lors du choix des méthodes appropriées.

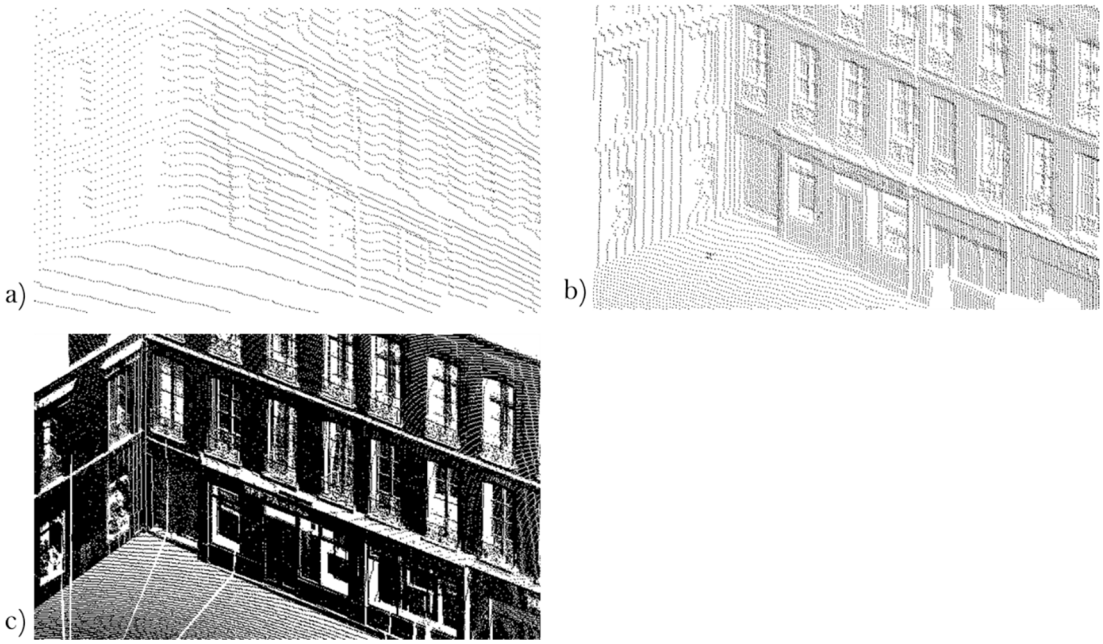

Figure 1.2 Exemple de relevés laser acquis en 2009 au sein d'une même zone de test (rue Soufflot à Paris) par : a) LARA3D ; b) Stéréopolis ; c) Scanner laser fixe (*Riegl VZ-400*)

Sachant que nous avons l'intention de fournir des solutions universelles, et ce quelle que soit la densité des données laser collectées, nous choisissons de ne pas utiliser de points de contrôle sélectionnés du nuage. Puisque l'identification précise des cibles, due aux caractéristiques des données laser, peut être ardue, voire impossible, nos solutions ont été conçues pour négliger l'influence de la densité des nuages de points sur le résultat de reconstruction des primitives. Nous avons donc posé l'hypothèse suivante : *Les entités linéaires (dont la position est connue) extraites à partir de nuages de points de scènes urbaines numérisées par un système de balayage laser mobile, permettent de quantifier et d'améliorer l'exactitude des données produites.* Dans ce travail, nous cherchons à démontrer la validité de

cette affirmation. Ainsi, nous vérifierons la pertinence des arêtes de pli, représentant les intersections entre les segments plans principaux dans le nuage de points.

Les travaux décrits au sein de ce mémoire abordent le processus d'extraction d'arêtes, par l'intermédiaire desquelles nous pourrons ensuite qualifier les données mobiles et développer des outils servant, à leur tour, à améliorer la précision de celles-ci. Plusieurs axes de recherche peuvent être imaginés, à partir desquels nous listons sur la (Figure 1.3) les questions que nous allons aborder dans ce manuscrit. Quant à l'étalonnage intrinsèque des scanners, cette tâche ne fait pas l'objet de cette thèse, car ses paramètres, étant fournis par le constructeur, nous les considérons comme un modèle valable. Néanmoins, il faut se rendre à l'évidence que des unités incorrectement étalonnées propageront aussi des erreurs entachant le nuage de points.

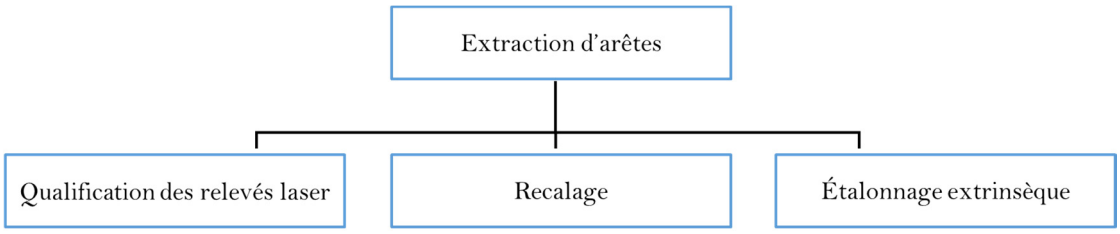

Figure 1.3 Principaux axes de recherche

Les éléments linéaires sont omniprésents dans les scènes urbaines ce qui les rend de fait comme une source riche en informations. D'après [Sappa *et al.*, 2006] et [Alshawa, 2006], les arêtes (les points de discontinuité) peuvent être regroupées en deux catégories (Figure 1.4) :

1) Les arêtes de pli (discontinuité de la dérivée première) qui apparaissent entre deux surfaces d'orientation différente. Elles peuvent être retrouvées par l'intersection des plans adjacents.
2) Les arêtes de pas (discontinuité d'ordre zéro) représentées par les points aux bords des trous et des fenêtres. Elles sont dues à la limitation du scanner laser, aux propriétés des surfaces numérisées pouvant absorber le signal émis, ou au manque de visibilité scanner – objet causé par des artefacts.

Dans le premier cas, il est préférable, afin de minimiser le bruit, que les deux sous-nuages soient acquis sous un angle direct. Ce critère n'est pas toujours aisé à satisfaire. Tout en considérant les arêtes de pas, il faut garder à l'esprit que la nature hétérogène d'un nuage de point ne permet pas de garantir que la ligne extraite corresponde fidèlement à l'arête réelle. Nous préférons alors utiliser les droites sous forme d'intersections, soit des arêtes de pli, puisqu'elles peuvent être facilement extraites et décrites, même si les nuages de points sont peu denses et assez bruités. Dans ce manuscrit, le terme « arêtes » signifiera donc une droite d'intersection (limitée) entre deux plans voisins.

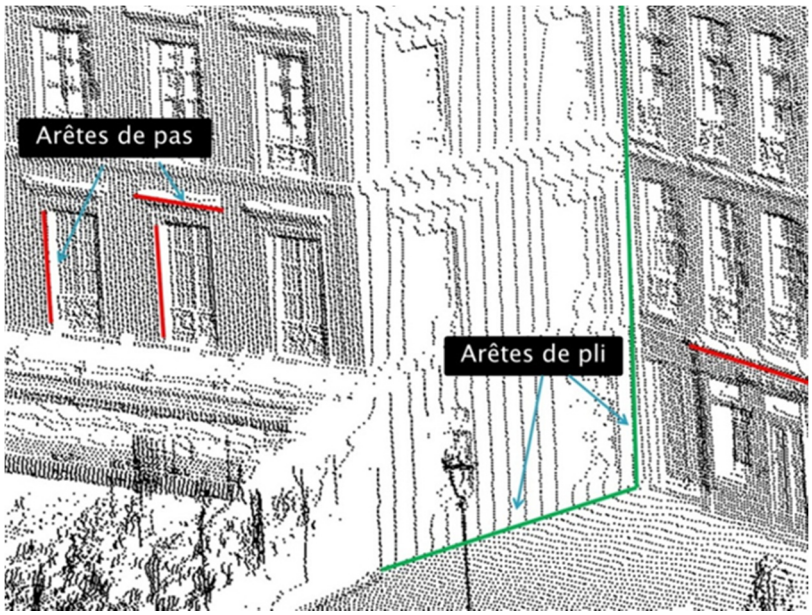

Figure 1.4 Différents types d'arêtes

1.2 Apports de la thèse

La qualité des relevés laser est un facteur clef à prendre en considération. Elle comprend plusieurs aspects, dont l'exactitude des mesures. Le principe consiste toujours à utiliser une référence dont l'exactitude est plus élevée que celle des données étudiées. La plupart des travaux actuels se réalisent, soit par une analyse de coordonnées de points de contrôle, soit par une comparaison avec d'autres types de données (nuage de points, modèle de ville 3D). Notre première contribution est de fournir une nouvelle approche de qualification de l'exactitude et de la précision, fondée sur les arêtes. S'agissant de la référence, nous pouvons imaginer des éléments linéaires mieux géopositionnés qu'ils soient 2D ou 3D, acquis lors du levé topographique, ou bien encore extraits du nuage de points statique (relevé par scanner laser fixe) ou du plan 2D (emprise de bâtis, bordure de trottoir, etc.) (Figure 1.5). L'éloignement des deux sous-ensembles de segments, exprimés en tant que la distance de *Hausdorff* adaptée aux lignes, représentera désormais une mesure de la qualité.

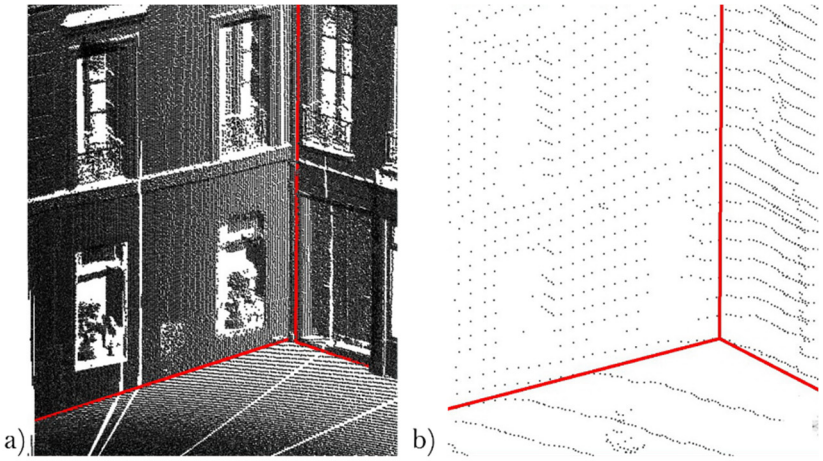

Figure 1.5 Arêtes de pli : a) Données de référence – nuage statique ; b) Relevé laser mobile

Notre deuxième contribution se veut de proposer une méthode automatique pour l'appariement des entités linéaires en s'appuyant sur leur similitude. Nous combinons, entre deux nuages, les lignes les plus semblables au sens du score attribué.

En troisième lieu, nous évaluons quelques approches existantes permettant le recalage rigide basé sur des entités linéaires, en examinant leur pertinence vis-à-vis de notre problématique. Notre but est plutôt d'affiner les différents nuages déjà géo-référencés pour corriger le décalage (*shift*) survenu lors de l'estimation de trajectoire du véhicule par un système de navigation. C'est pourquoi nous portons un regard critique sur les méthodes actuelles qui paraissent peu robustes au bruit. Au vu de ces éléments, nous développons une chaîne complète, comprenant un appariement et une estimation des paramètres de pose par des segments de droite.

La quatrième contribution réside dans la mise en évidence des arêtes afin de pouvoir en déduire l'étalonnage extrinsèque du système intégral, c'est-à-dire la matrice de rotation (*boresight matrix*) et le bras de levier (*lever-arm*) entre le scanner laser et la centrale inertielle, les deux étant embarqués sur une plate-forme terrestre. Nous démontrons que ces paramètres, ayant un impact sur les positions et orientations des lignes extraites d'une scène urbaine, sont observables, et que par conséquent, leurs valeurs numériques peuvent être estimées. Nous esquissons, un concept d'approche, devant être encore approfondie et testée, permettant un étalonnage extrinsèque du système doté d'un scanner à double balayage.

Concernant l'extraction des arêtes de pli, nous employons, dans nos recherches, une méthode amenant à déterminer des droites d'intersection entre les plans détectés auparavant à l'aide de l'algorithme de *RANSAC*. Ce dernier est enrichi d'une analyse de composantes connexes. Nous proposons une solution alternative, étant notre dernier apport, pour vérifier la connectivité des plans, celle-ci faisant appel aux éléments de la théorie des graphes.

Ces travaux ont donné lieu à plusieurs publications listées à la page 139 du manuscrit.

1.3 Plan du mémoire

Ce mémoire de thèse est structuré en trois parties principales : l'extraction d'arêtes, la qualification des relevés laser et l'amélioration par consolidation. En souhaitant s'affranchir des limitations du système mobile de cartographie et pour répondre aux exigences imposées par le projet TerraMobilita, la qualité des données laser doit être étudiée et évaluée, pour pouvoir par la suite proposer des améliorations utiles et efficientes. Chacun des chapitres, s'articulant autour de ces objectifs, est accompagné par un état de l'art permettant de s'inspirer des moyens et des méthodes mis en œuvre.

Nous caractérisons, dans un premier temps (Chapitre 2), le système mobile de cartographie. Nous commençons par décrire le principe de son fonctionnement. Nous présentons des données avec lesquelles nous travaillons, acquises dans le cadre du projet TerraMobilita ou d'une autre manière. Nous définissons les notions d'erreurs telles que liées au levé topographique, et introduisons les termes d'exactitude et de précision. En outre, nous testons quelques méthodes simples permettant d'évaluer l'exactitude relative des relevés laser mobiles.

Ensuite (Chapitre 3), nous listons non seulement les sources d'anomalies envisageables contribuant à l'erreur globale du système, mais aussi nous présentons des solutions récentes pour pallier à ces difficultés. Ainsi, différentes méthodes applicables en temps réel et en post-traitement y compris l'étalonnage extrinsèque sont discutées. Les effets du défaut d'étalonnage précis des excentricités angulaires (*boresight angles*) entre les capteurs de perception et la centrale inertielle, sur la qualité géométrique des nuages de points, sont aussi expliqués.

Notre point de départ est l'extraction des arêtes de pli à partir de relevés laser (Chapitre 4). Nous mettons en avant une procédure de détection des droites d'intersections via des segments plans retrouvés. Ensuite, nous nous penchons sur les algorithmes de reconstruction automatique des primitives planes dans un nuage, étant donné un modèle connu a priori, notamment sur l'estimateur de *RANSAC*. Nous nous imposons une contrainte de connectivité entre les points formant chaque segment plan, et résolvons cette question grâce à la théorie des graphes, plus précisément par la structuration propre du nuage et la décomposition *Dulmage-Mendelsohn*.

Ensuite (Chapitre 5), nous proposons une approche de qualification des relevés laser basée sur des entités linéaires. Celle-ci délivre une mesure de qualité en s'appuyant sur la distance de *Hausdorff* modifiée. Pour déterminer sa valeur, nous développons un algorithme de mise en correspondance visant à repérer les entités analogues ou très proches dans deux jeux de données. Nous appliquons cette méthode aux lignes 3D extraites au sein d'une zone de test, et présentons les résultats obtenus.

De surcroît (Chapitre 6), nous suggérons de corriger les données laser mobiles par un recalage rigide bénéficiant des segments de droite. Après avoir détaillé les contraintes apportées par une paire de primitives linéaires appariées, nous expliquons, et évaluons trois algorithmes de calcul des poses, par minimisation de ces contraintes entre deux ensembles de segments. Par la suite, nous avançons une solution complète aboutissant à raffiner, par consolidation, des nuages de points distincts. Nous montrons l'efficacité de notre approche, et présentons les résultats obtenus sur quelques jeux de données simulées et réelles acquises dans le cadre du projet TerraMobilita.

Pour terminer (Chapitre 7), nous soumettons une brève discussion relative aux conclusions et perspectives issues de ces recherches. Nous suggérons également une approche d'étalonnage extrinsèque, conçue pour des scanners 3D.

CHAPITRE 2

Acquisition et qualification de relevés laser

Sommaire :

2.1 Systèmes mobiles terrestres voués à la numérisation d'environnement 16
2.1.1 Principes fondamentaux d'un système mobile de cartographie 18
2.1.2 Description du prototype L3D2 21
2.1.3 Autres exemples de systèmes mobiles de cartographie 23
2.2 Sources de données expérimentales 24
2.3 Qualification de données 28
2.3.1 Différents types d'erreurs en topographie 28
2.3.2 Critères servant à décrire la qualité de nuages mobiles 31
2.3.3 État de l'art sur l'évaluation de l'exactitude de relevés laser mobiles 32
2.4 Expérimentation 35

Résumé

Dans ce premier chapitre, nous expliquons brièvement les principes fondamentaux d'un système mobile MLS positionné avec une navigation hybride GNSS/INS. Nous présentons quelques récentes plates-formes de cartographie. Puis, nous décrivons le prototype L3D2 développé au CAOR de MINES ParisTech ainsi que d'autres systèmes dont les données ont été employées pour tester des approches proposées au cours de cette thèse.

Ensuite, nous caractérisons différentes sources d'erreurs ayant un impact sur la qualité des mesures. Aussi, les notions complémentaires d'exactitude et de précision de données sont introduites afin d'expliciter la nature de ces erreurs. Notre objectif principal est de nous pencher sur les méthodes récentes traitant du problème de qualification. Notamment, l'exactitude de relevés laser mobiles, mesurée par rapport à diverses références et éléments de contrôle, est étudiée. Un aperçu de techniques existantes dévoile que la plupart d'entre elles souffraient du manque d'automatisation.

Finalement, deux approches sont implémentées puis testées pour vérifier l'exactitude relative du nuage de points mobile. Les données référence sont les suivantes : 1) un ensemble de points aux coordonnées connues et mesurées à l'aide d'un tachéomètre électronique ; 2) un nuage de points statique. Les points faibles des techniques mises en œuvre sont notés.

2.1 Systèmes mobiles terrestres voués à la numérisation d'environnement

Soutenus par un progrès constant dans le génie informatique et le traitement automatique d'informations, les systèmes mobiles basés sur *LiDAR* évoluent à un rythme rapide. Leur développement comprend des modifications technologiques importantes en termes de capteurs, d'architecture du système et de traitement de données. De nouveaux scanners laser offrent une large plage de fonctionnement. Les fabricants ont considérablement avancé sur l'ergonomie des appareils en les rendant plus légers et améliorant par conséquent leur portabilité. Alors qu'un scanner laser nécessitait auparavant une équipe de deux personnes, une seule suffit désormais pour effectuer un balayage laser fixe. D'après les études publiées en 2013 par la société de recherche et de conseil américaine *ARC Advisory Group* [Evans, 2013], le marché des scanners laser 3D va continuer à pénétrer de nouveaux secteurs doublant ainsi les ventes sur les quatre prochaines années. Avec cette variété de capteurs laser, une baisse massive des prix et une simplification de leur utilisation, le nombre de systèmes mobiles aussi bien que leurs applications possibles ne cessent de croître. Dès lors, les plates-formes mobiles aériennes *ALS* (*Airborne Laser Scanning*) et terrestres *MLS* (*Mobile Laser Scanning*) ainsi que les systèmes portatifs (*Hand-Held Mobile Mapping*) sont de plus en plus utilisés, permettant une exploitation de ce matériel dans des secteurs qui autrefois n'auraient pas envisagé leur utilisation. En outre, avec le développement des capteurs miniaturisés *LiDAR* pouvant être embarqués sur les petits drones, la prolifération des systèmes *UAV (Unmanned Aerial Vehicle)/ UAS (Unmanned Aircraft System)* est devenue une réalité.

Figure 2.1 Exemple de systèmes commerciaux : a) Lynx Mobile Mapper M1 de Chung Hsing Surveying ; b) Garmin Corporation de Taïwan ; c) Riegl VMX-250 de SGS ; d) *MMS* série X de Mitsubishi Electric

Les acquisitions par l'intermédiaire d'un système *MLS* peuvent être réalisées, selon la façon de combiner les données provenant des capteurs, soit en mode dynamique (*On Drive*) soit *Stop&Go*. Le premier mode permet de numériser l'environnement en mouvement, c'est-à-dire les informations sont intégrées en temps réel pendant le déplacement de la plate-forme. Lorsque la seconde technique est employée, le véhicule reste immobile et le scanner effectue un scan complet tournant sur deux axes. Après avoir fini les mesures, le système se déplace à une autre position.

Figure 2.2 Quelques récents et « légers » systèmes mobiles de cartographie : a) ZEB1 de 3DLaserMapping ; b) *BackPack Mobile Mapping System* [Liao *et al.*, 2013] ; c) Trekkeur de *Google Street View* ; d) Système mobile de l'Université de Floride (*FLREC Geomatics*) ; e) VERDINO2 de l'Université La Laguna de Ténérife

Dans tous les cas, une source d'alimentation, un ordinateur portable et deux modules produisent ses fonctionnalités basiques. À bord d'une plate-forme mobile, les capteurs de positionnement (p.ex. un récepteur *GNSS*, une centrale inertielle *INS*, un odomètre *DMI* (*Distance Measurement Instrument*)) et les capteurs d'imagerie (scanners laser, caméras), tous en une configuration propre à chaque constructeur, doivent trouver leur place. En revanche, du fait de la polyvalence des systèmes, ils peuvent être déployés sur les différents types de

véhicules y compris le camionnette, le véhicule 4x4, le quad, le chariot, le vélo, le bateau ou la locomotive, comme l'illustrent la Figure 2.1 et la Figure 2.2.

2.1.1 Principes fondamentaux d'un système mobile de cartographie

Puisque chaque système mobile *MMS* (*Mobile Mapping System*) intègre de nombreux capteurs, c'est une technologie de plus en plus complexe. Tout système est capable de fournir des données correctes, pour autant que les conditions suivantes soient remplies :

1) Les capteurs entre eux-mêmes doivent être précisément synchronisés ;
2) Le système est tenu d'être rigide, c'est-à-dire qu'il n'existe aucun déplacement relatif entre les composantes au cours de l'acquisition ;
3) La position et l'attitude de la plate-forme doivent être déterminées avec une précision suffisante ;
4) Les excentricités linéaires et angulaires à savoir les translations et les rotations entre les capteurs de perception et l'*IMU (Inertial Measurement Unit)* doivent être connues.

Chacun des capteurs mis en place travaille avec son propre système de référence. Le Tableau 2.1 caractérise les repères pouvant être définis lors du fonctionnement d'un système mobile, qu'il soit terrestre ou aérien.

Tableau 2.1 Repères de référence

Sigle	Nom du repère	Description
s	Repère du capteur (*Sensor Frame*)	Repère laser/caméra.
b	Repère body (*Body Frame*)	Repère représenté par les axes du système de navigation inertielle. L'origine de ce système est située au centre de l'*IMU*. Les angles de rotation associés sont nommés : roulis (*roll*), tangage (*pitch*), lacet (*yaw*).
l	Repère véhicule (*Local Level Frame*)	Ce cadre est généralement utilisé comme référence pour les angles d'orientation mesurés par le système intégré *GNSS/INS*. Autrement dit, c'est un repère local terrestre mobile. Son origine peut être définie comme l'intersection de la verticale locale, déterminée à partir de la position actuelle du capteur, avec l'ellipsoïde terrestre global.
e	Repère de l'objet (*ECEF Frame*)	Système géodésique cartésien géocentré (attaché à la Terre). L'origine de ce système est fixée au centre de la Terre. La direction de l'axe *OX* est donnée par l'intersection de l'équateur terrestre et par le méridien de Greenwich. Quant à l'axe *OY*, il se situe dans le plan de l'équateur et de l'orthogonal à *OX*. Pour l'axe *OZ*, il correspond à l'axe de rotation de la Terre. De ce fait, le résultat du calcul de la trajectoire est principalement fourni dans ce repère sous forme de coordonnées géocentriques ou de coordonnées géographiques : λ - latitude, φ - longitude, h – hauteur ellipsoïdale.
M	Repère terrain (*Mapping Frame*)	Ce repère est employé pour exprimer les coordonnées des points d'un nuage/image. Il peut être aussi représenté par une surface de référence, par projection ainsi que par le système de référence (national, reconnu dans le pays).

Afin de générer un nuage de points, dans un repère terrain, à partir de données brutes enregistrées par les capteurs, trois changements successifs de repère sont nécessaires (Figure

2.3). D'abord, le passage distance-angle (coordonnées sphériques) aux coordonnées cartésiennes dans le repère lié au scanner laser (s) est effectué. Ensuite, nous passons aux coordonnées dans le repère *IMU* (b) défini par la centrale inertielle pour réaliser la transformation dans le repère terrain (M). Le premier passage se fait en prenant également en compte les paramètres d'étalonnage intrinsèque. Quant au deuxième, il exige un étalonnage extrinsèque permettant de déterminer les excentricités linéaires $r_{b/s}^{b}$ et angulaires R_s^b entre les capteurs de perception et la centrale inertielle. Enfin, la dernière transformation est envisageable grâce aux données $R_b^M(t)$ mesurées par la centrale inertielle.

La notation employée pour décrire la transformation d'un repère à l'autre est la suivante : R symbolise la matrice de rotation, et r la translation. En outre, l'indice désigne le capteur concerné, tandis que l'exposant (indice supérieur) indique le repère dans lequel cette grandeur est mesurée. Par exemple, $r_{b/GNNS}^{b}$ correspond au bras de levier entre l'unité inertielle et le récepteur *GNSS*, exprimé dans le repère body (b).

La formulation mathématique de géo-référencement direct d'un système équipé de scanners laser peut être exprimée comme suit :

$$r_p^M = r_{b/GNSS}^M(t) + R_b^M(t) \cdot \left(r_{b/s}^b + R_s^b X_{laser}\right) \tag{2.1}$$

où :
$r_{b/GNSS}^M(t)$: position mesurée par le système intégré *GNSS/INS/DMI* exprimée dans le repère terrain (M) ;

$R_b^M(t)$: matrice de rotation du repère body (b) vers le repère terrain (M). Elle peut être représentée par trois angles : la rotation autour de l'axe X - roulis (φ), celle autour de l'axe Y – tangage (θ), et celle autour de l'axe Z – lacet (ψ). Toutes les trois varient au cours du temps (t) ;

La convention utilisée pour ces trois rotations par les centrales inertielles les plus courantes (*IXSEA* et *Applanix*) est la convention *roll-pitch-yaw* (2.2) :

$$R_b^M(t) = R_Z(\psi) \cdot R_Y(\theta) \cdot R_X(\varphi) \tag{2.2}$$

où :

$$R_X(\varphi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\varphi) & -sin(\varphi) \\ 0 & sin(\varphi) & cos(\varphi) \end{bmatrix}; \quad R_Y(\theta) = \begin{bmatrix} cos(\theta) & 0 & sin(\theta) \\ 0 & 1 & 0 \\ -sin(\theta) & 0 & cos(\theta) \end{bmatrix}; \tag{2.3}$$
$$R_Z(\psi) = \begin{bmatrix} cos(\psi) & -sin(\psi) & 0 \\ sin(\psi) & cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix};$$

X_{laser} : Coordonnées de points laser mesurés dans le repère scanner (b). Pour un scanner de profil 2D (balayage plan) *SICK LMS 221*, le passage des données brutes au repère scanner (s) se réalise selon la formule (2.4)

$$X_{laser} = \begin{bmatrix} \rho sin(i\theta + \varphi) \\ -\rho cos(i\theta + \varphi) \\ 1 \end{bmatrix} \tag{2.4}$$

où ρ :mesure de distance objet-scanner ; φ : angle de début du balayage ; θ : résolution du scanner ; i : index du faisceau laser.

$r^b_{b/s}$: excentricité linéaire (bras de levier) entre le centre du scanner et le centre de la centrale inertielle, exprimée dans le repère de l'*IMU* (b). Cette valeur est supposée constante au cours du temps (t) ;

R^b_s : Matrice de rotation (matrice *de boresight*) du repère laser (s) vers le repère body (b). Ses angles e_X, e_Y, e_Z sont constants dans le temps (t).

Nonobstant, dans le cadre des systèmes mobiles dotés de caméras, la formule (2.1) garde sa forme initiale, sauf que la composante X_{laser} doit être remplacée par μr^c_p. Le caractère μ symbolise le facteur d'échelle, tandis que r^c_p contient des coordonnées d'images du point dans le repère caméra.

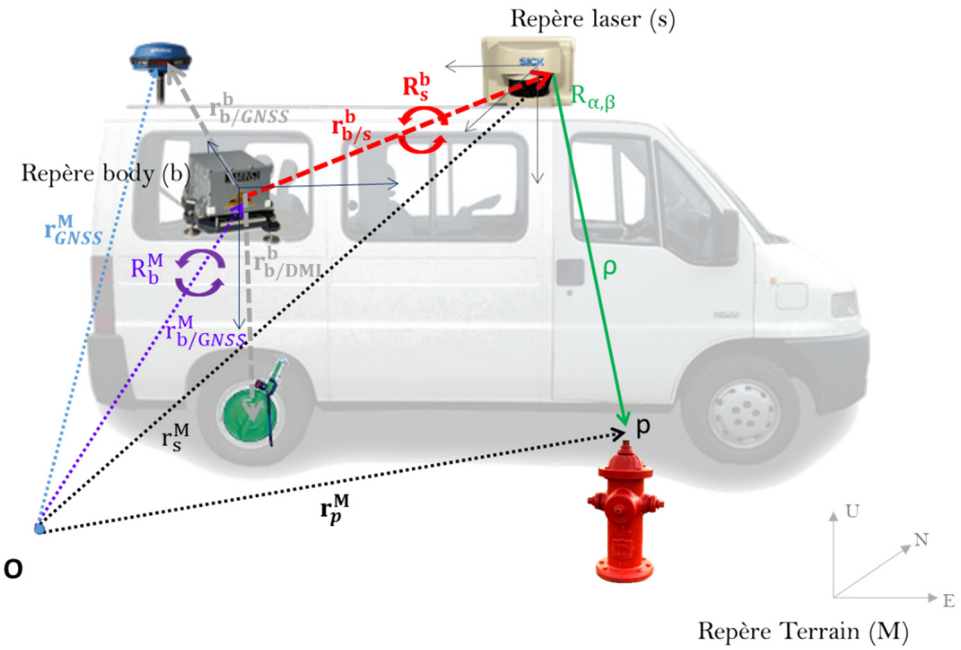

Figure 2.3 Concept du géo-référencement direct

Selon [Ellum et El-Sheimy, 2002], la qualité des nuages de points dépendra alors, dans la même mesure, de la précision de l'étalonnage du système et des erreurs de mesures laser. Ils proposent, à cet effet, d'estimer l'influence des différents composants à travers la dérivation du premier ordre de l'équation (2.1).

$$\begin{aligned}
&\delta r^M_p = \delta r^M_{b/GNSS}(t) && \text{Erreur de position } GNSS \\
&+\delta R^M_b(t)\left(r^b_{b/s} + R^b_s X_{laser}\right) && \text{Erreur d'attitude } IMU \\
&+R^M_b(t)\delta R^b_s X_{laser} && \text{Erreur sur l'excentricité angulaire (}boresight\text{)} \\
&+R^M_b(t)\delta r^b_{b/s} && \text{Erreur sur l'excentricité linéaire} \\
&+R^M_b(t)R^s_b\delta X_{laser} && \text{Erreur de mesures du scanner laser}
\end{aligned} \tag{2.5}$$

$$+\delta t\left(v(t)+\omega(t)\left(r_{b/s}^{b}+R_{s}^{b}X_{laser}\right)\right)$$ Erreur de synchronisation

Il en ressort que la multitude d'éléments associés au système mobile produit plusieurs types d'erreurs affectant les résultats d'acquisitions. Le Chapitre 3 abordera leur origine, mais il expliquera aussi la contribution de chacune des composantes au bilan global des erreurs du système.

2.1.2 Description du prototype L3D2

Un système mobile est développé au sein du *CAOR* de MINES ParisTech depuis 2002. Il s'agit du prototype *L3D2*. C'est une nouvelle plate-forme de numérisation 3D, basée sur les mêmes principes, mais plus performante que son prédécesseur appelé *LARA3D* (LA Route Automatisée). Le but de ce développement en cours est, dans le cadre du projet TerraMobilita, de rivaliser avec la précision des techniques et des méthodes topographiques conventionnelles (tachéométrie, *GPS*, nivellement direct, balayage laser terrestre).

2002 2008 2010

Figure 2.4 Évolution du prototype *LARA3D*

Figure 2.5 Prototype L3D2 (version de mai 2012)

Un tel système est configuré pour accueillir de nombreux capteurs et il a donc dû subir, depuis sa conception, plusieurs améliorations. La Figure 2.4 présente ce prototype dans la version de 2002, 2008, et de 2010. Le système le plus récent avec lequel nous sommes en train de travailler est représenté par la Figure 2.5.

Le système de positionnement de notre prototype est composé d'un récepteur *GNSS Novatel Flexpak6*, d'une centrale inertielle *PHINS* d'*IXSEA* et d'un odomètre (*DMI*). Les capteurs de perception sont constitués, optionnellement, de deux scanners laser *SICK LMS* 221 en balayage plan vertical à droite et à gauche du véhicule, situés dos à dos. Chaque télémètre laser, fonctionnant à temps de vol (le temps de parcours de l'onde entre son départ et son retour sert à déduire une mesure de distance), numérise une zone de 180 degrés. À l'heure actuelle, notre objectif principal est de remplacer ces scanners par un *Velodyne HDL-32E* (*High Definition LiDAR*), contenant trente-deux fibres inclinées dans un plan vertical tournant sur lui-même. Le laser *Velodyne* sera installé en arrière du véhicule et penché d'environ 60°. Par conséquent, environ 700 000 de points par seconde seront mesurés. Nous pouvons aussi ajouter, selon les usages, un ou plusieurs autres capteurs.

Le véhicule (Citroën Jumper) est également doté d'un ordinateur embarqué avec un logiciel robuste *RTMaps* v.3.4.10 permettant la synchronisation, en temps réel, des données datées provenant de différents capteurs. La trajectoire du véhicule est calculée grâce au filtrage de Kalman, un outil puissant servant à établir la synergie entre trois capteurs de navigation : *GNSS*, *INS* et *DMI*, puisqu'il peut tirer les avantages de ces trois systèmes de localisation distincts. Une fois le véhicule localisé, c'est-à-dire sa position et sa rotation connue dans un repère terrain, il est possible de géo-référencer toutes les données issues des capteurs de perception. Pour ce faire, la datation des données de cartographie est synchronisée avec celle de la localisation, avec si nécessaire une interpolation entre deux positions successives.

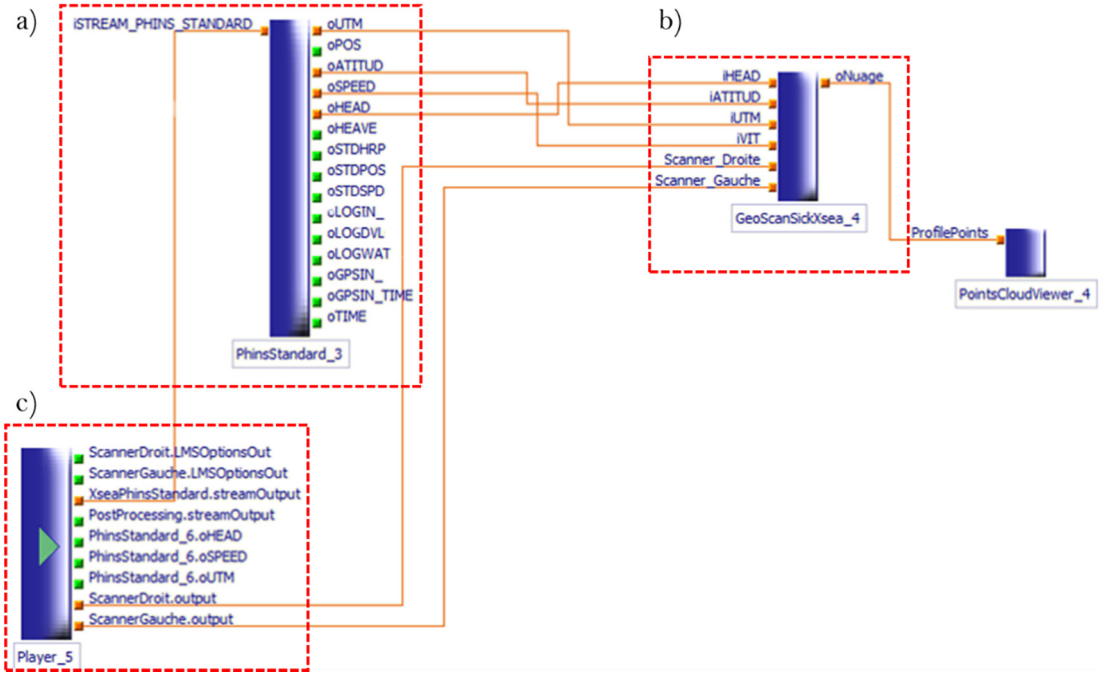

Figure 2.6 Diagramme *RTMaps* utilisé pour la génération des nuages de points : a) Module de localisation ; b) Module de géo-référencement ; c) Données de cartographie

La Figure 2.6 montre le diagramme *RTMaps*, utilisé lors de la génération des nuages de points, comprenant trois composants principaux de traitement :

1) Module de localisation (parseur de la sortie de centrale) – package PhinsStandard v.1.0

2) Module de géo-référencement (reconstruction du nuage de points) – package GeoScanSickXsea v.1.0
3) Visualiseur de nuage de points – package PointscloudViewer v.1.0

Figure 2.7 Nuage de points acquis par LARA3D – *LARA3Dv2010_Orsay*

2.1.3 Autres exemples de systèmes mobiles de cartographie

Dans le cadre de notre étude (extraction d'arêtes, détection de plans, qualification, recalage rigide), nous avons utilisé différents types de relevés laser.

Figure 2.8 Stéréopolis II, système mobile de l'*IGN* [Paparoditis *et al.*, 2012]

La Figure 2.8 montre le système de cartographie mobile développé par l'*IGN*. Son système actuel de vision comprend des capteurs de haute résolution, notamment un ensemble de dix caméras *HD* assurant une couverture horizontale complète, deux scanners laser *Riegl LMS-Q120i* avec une ouverture de 80 degrés et un Velodyne®HDL-64E. La partie concernant la navigation est constituée d'un système hybride composé de deux récepteurs *GNSS*, d'une centrale inertielle *Applanix* (*POSPac / POS LV V4*), et d'un odomètre. L'architecture du couplage de capteurs est de type lâche (*tightly coupled*). Pour plus d'informations relatives au système Stéréopolis II, le lecteur pourra se référer à [Paparoditis *et al.*, 2012].

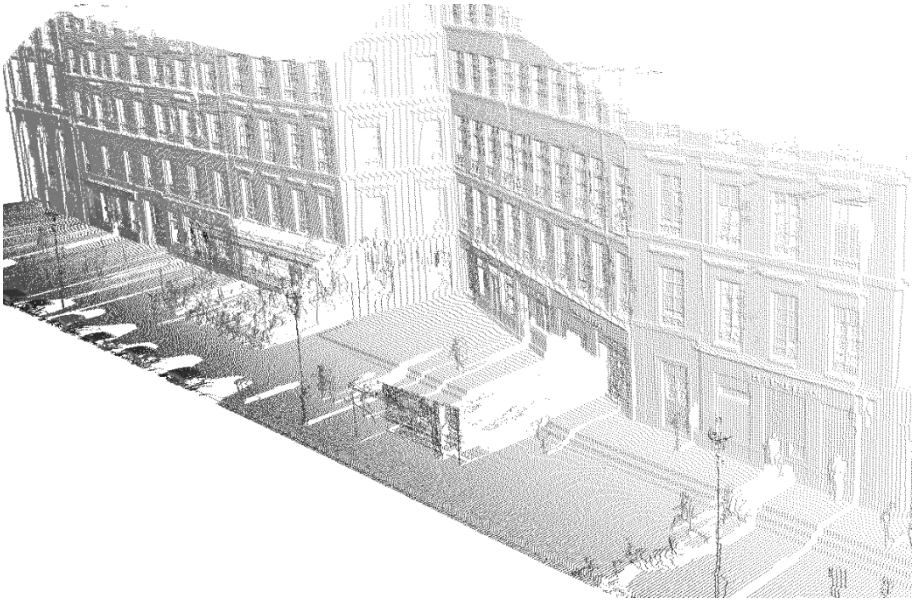

Figure 2.9 Exemple de nuages de points produits par le système Stéréopolis II – *Stereopolis2009_Soufflot*

Le second système mobile de cartographie terrestre, dont les données ont été exploitées, est un système commercial de *Riegl VMX-250*. Il est doté de deux scanners laser de *Riegl VQ-250*, de quatre caméras et de compteurs très précis de positionnement (la spécificité plus détaillée du système se retrouve sur le site web du *Riegl*).

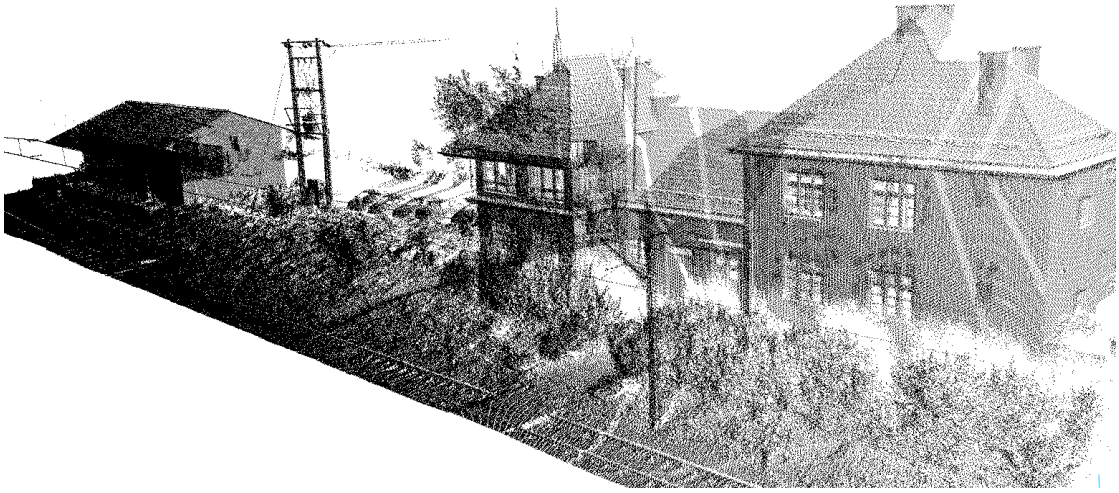

Figure 2.10 Gabarit ferroviaire numérisé par *Riegl VMX-250* – *Riegl2012_Slomniki*

2.2 Sources de données expérimentales

Relevé laser

Nous traitons, au sein de ce travail, des nuages de points produits par des systèmes terrestres fixes ou mobiles notamment :

1) Les relevés laser mobiles collectés par LARA3D (version 2010) (Figure 2.7) ;

2) Les relevés laser mobiles produits par Stéréopolis II de l'Institut Géographique National (*IGN*) (Figure 2.9) ;
3) Les relevés laser mobiles acquis par un système mobile de *Riegl VMX-250* (Figure 2.10) ;
4) Les relevés laser statiques obtenus par scanner fixe *Trimble®GX*, *Timble®VX* et Leica C10.

Les nuages de points de haute résolution (statique) servent de référence à notre étude comparative (qualification, recalage). Le premier nuage, nommé *Orsay2009_statique*, a été fourni par *Trimble 3D Scanning System* lors des travaux réalisés dans le cadre du projet TerraNumerica (projet antérieur à TerraMobilita). Les acquisitions ont été faites de façon complémentaire avec une station spatiale *Trimble®VX* présentant les fonctionnalités du scan 3D et un scanner *Trimble®GX*. Vingt-six stations, dont 10 stations de GX, ont été placées afin d'assurer une bonne couverture des deux façades du Musée d'Orsay (nord et ouest) et d'éviter au maximum les masques. Enfin, la résolution finale du nuage a été limitée à 2cm.

Le second nuage *Sulpice2013_statique* (Figure 2.11), a été collecté lors d'une acquisition de terrain que nous avons réalisée le 14 novembre 2012 en collaboration avec l'*ENSG* (École Nationale des Sciences Géographiques). La Figure 2.12 représente le schéma des emplacements de différentes stations *Leica ScanStation C10*, ainsi que des stations *GNSS* stationnées avec un *Trimble®R8*. Des scans de très haute résolution (4mm) ont été acquis et ensuite géo-référencés dans le système *Lambert-93*. Le logiciel *LGO* (*Leica Geo Office*) de *Leica* a été choisi pour traiter les données *GNSS*. Les nuages de points, grâce aux coordonnées obtenues, ont été géopositionnés à l'aide de *Leica Cyclone 7.0.2*. Au final, la précision relative des relevés laser est de l'ordre de 3mm. Concernant la précision absolue, nous estimons qu'il existe des erreurs d'environ 3cm confirmant que notre référence est fiable. En conséquence, nous disposons d'une base abondante de données ayant des caractéristiques complètement différentes (taux de bruit, densité, qualité).

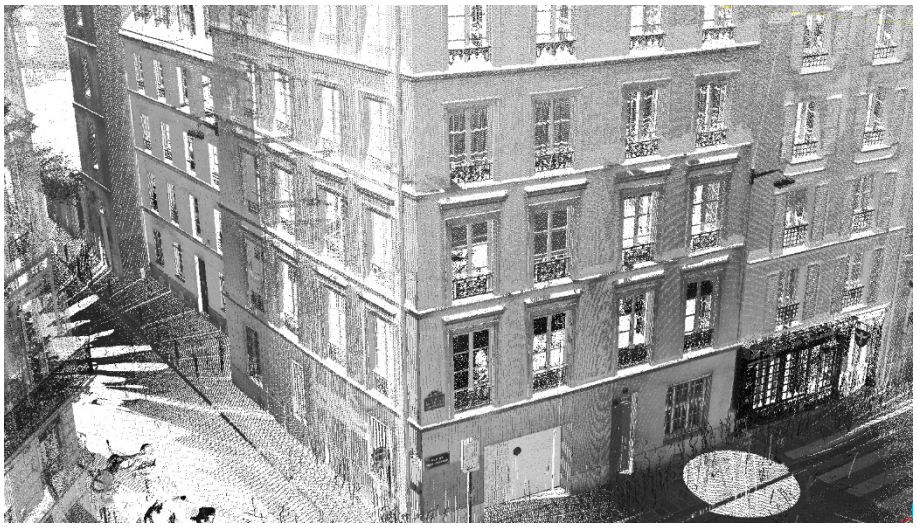

Figure 2.11 Nuage de points consolidé - *Sulpice2013_statique*

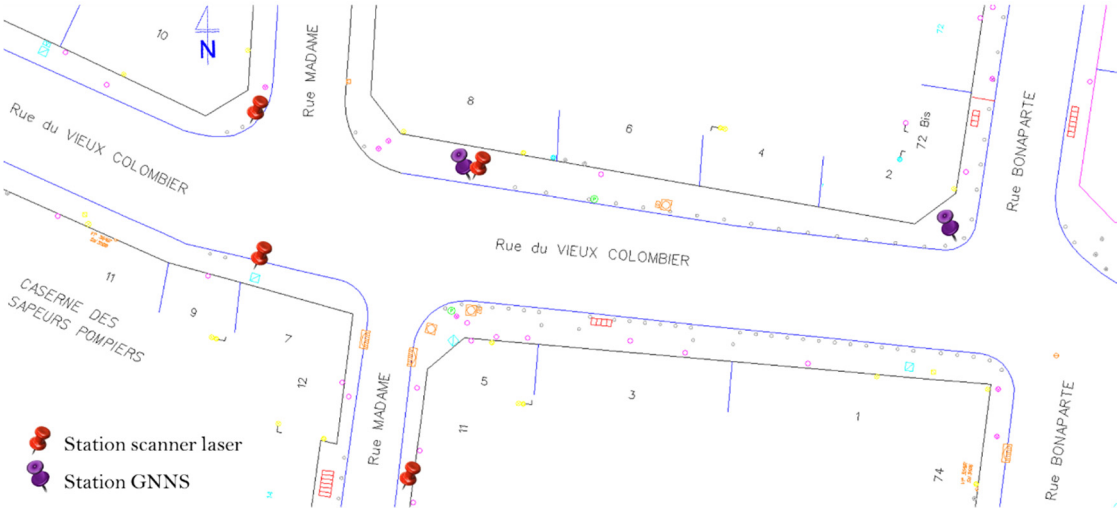

Figure 2.12 Plan de l'acquisition statique (Croisement des rues du Vieux-Colombier et Madame, Paris)

Levé topographique

Un levé topographique *Orsay2011_topo* a été réalisé le 27 février 2011 avec une *TCR 803 de Leica* (Figure 2.13b). Un cheminement ouvert comportant cinq stations intermédiaires, situées à une distance d'au maximum 30m de la façade, a été mis en place (Figure 2.13a). Le nombre et l'emplacement de points de canevas ont été planifiés de telle manière que le tachéomètre émette un rayonnement pour lequel l'angle d'incidence ne soit pas inférieur à 60° (voir l'information sur l'angle d'incidence dans la section 3.1.2). Cette contrainte a été choisie pour assurer une bonne qualité de données de référence, saisies en mode sans prisme réflecteur. D'autre part, toutes les mesures ont été géo-référencées dans un système local établi par deux stations (St4 et St5).

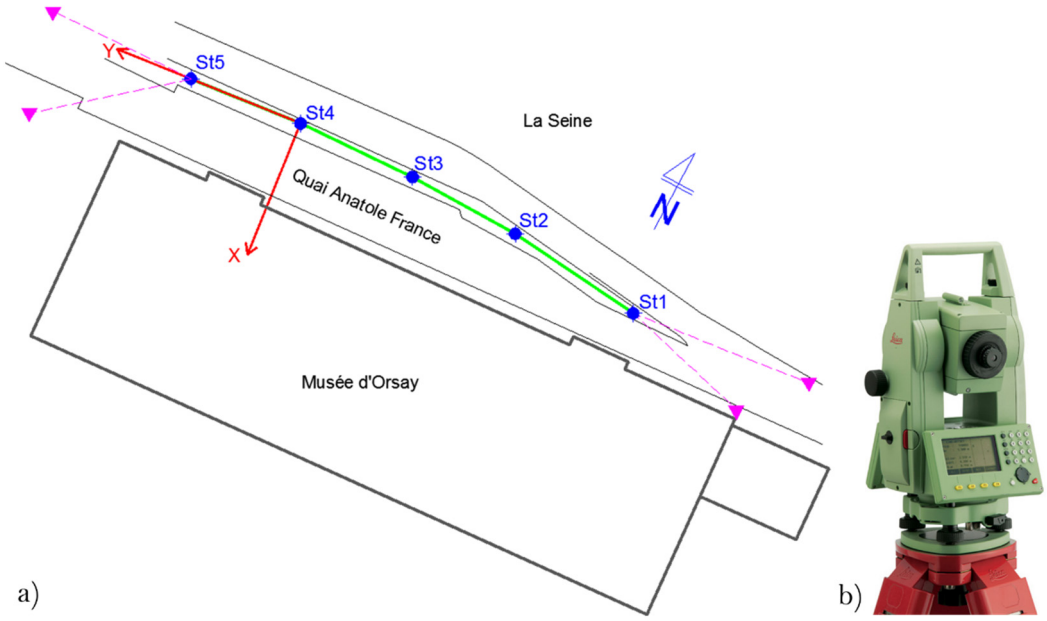

Figure 2.13 a) Cheminement réalisé lors du levé topographique ; b) *TCR 803 de Leica*

Au total, le levé opéré dans une seule position du cercle a contenu 344 points dont 297 appartenant à une coupe horizontale prise environ 7.2m au-dessus du sol (Figure 2.14). Le transfert des données depuis la station totale a été fait sous *LGO*, tandis que le traitement (décodage, calcul de coordonnées, etc.) à l'aide du logiciel *WinKalk 3.78*. Toutes ces mesures constituent une référence pour les analyses présentées dans la poursuite des travaux.

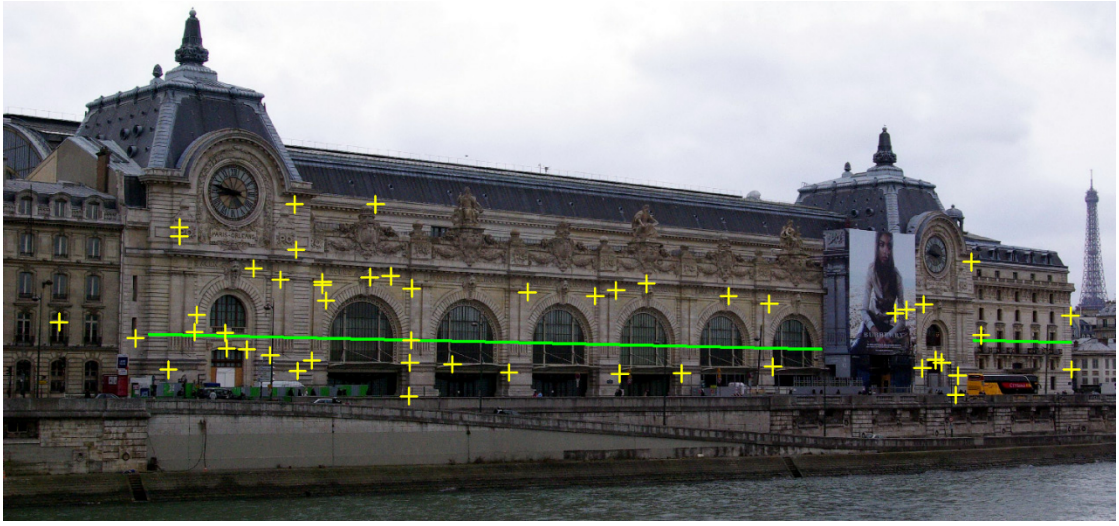

Figure 2.14 Répartition de points de contrôle (en jaune), et la coupe horizontale (en vert)

Les points de contrôle à viser ont été représentés par des détails architecturaux tels que les coins de bâtiments, les arêtes de piliers, les intersections d'éléments caractéristiques. Les piquets se trouvent alors sur un coin convexe ou concave dont la mesure à l'aide d'un télémètre laser peut poser certains problèmes. Nous obtenons, suivant les cas, un arrondi ou un congé entre deux surfaces voisines, tous deux dus à la généralisation de l'information lors de la réception du signal de retour [Klimkowska et Wrobel, 2006]. La Figure 2.15 illustre la situation typique rencontrée pendant le mesurage d'une bordure (*edge effect*) - les coordonnées obtenues ne correspondront quasiment jamais à la vraie position d'arête. Aussi, la distance mesurée entre l'emplacement actuel et une cible s'écarte de la valeur réelle.

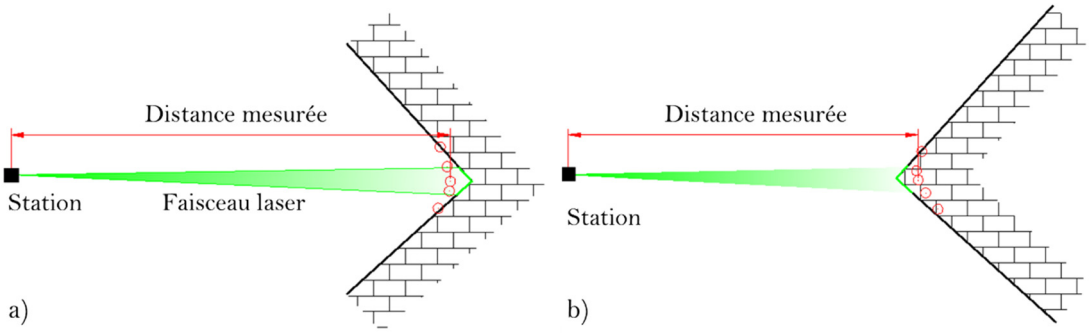

Figure 2.15 Résultats de mesure : a) Coin concave ; b) Coin convexe

Étant donné que l'empreinte laser possède une certaine taille dépendant de l'ouverture du dispositif, de la longueur de faisceau laser, mais aussi de la divergence de celui-ci, elle ne se matérialisera jamais sous la forme d'un point. En réalité, un laser produit une lumière qui va diverger de son origine. La cible balayée renvoie ainsi des retours multiples - une partie du signal proviendra de l'objet lui-même et une autre sera rétrodiffusée ; selon le cas, par une surface adjacente ou par une autre située derrière (Figure 2.16). Cependant, cette énergie ne

sera jamais retournée s'il n'y a pas d'autres éléments à portée du scanner. En fin de compte, la mesure finale de la distance sera toujours la moyenne des énergies retournées. L'erreur de mesures peut donc varier de quelques millimètres voire même de quelques centimètres. En ce qui concerne la mesure de coins concaves, le problème est moins complexe. La distance obtenue sera toujours plus petite que la valeur réelle. Quelle que soit l'origine de ces imprécisions, les cibles doivent être choisies avec soin.

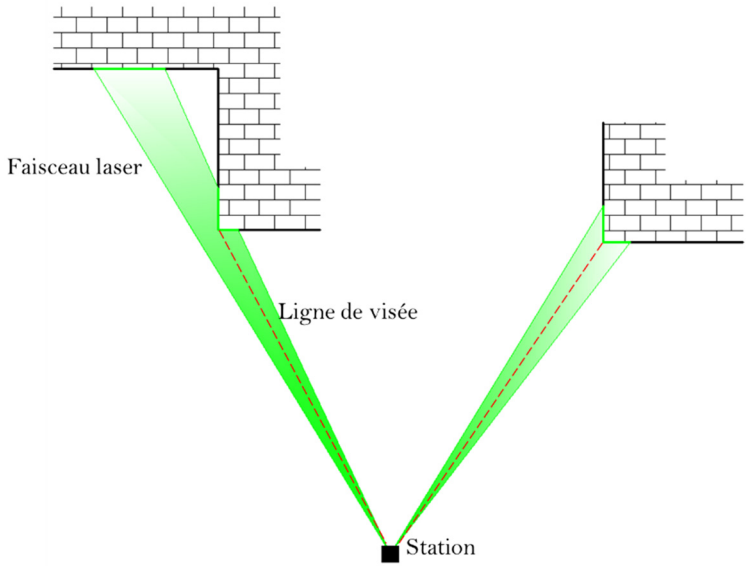

Figure 2.16 Empreinte du rayon laser sur les surfaces voisines

2.3 Qualification de données

La qualité de données quelconques est un facteur clef à prendre en compte. À cet égard, la qualification devrait se fonder sur des procédures claires et précises. Son objectif se veut de satisfaire des besoins bien déterminés : ceux des fournisseurs (entre autres les topographes) étant obligés de valider des mesures, ainsi que ceux des futurs utilisateurs devant être au fait pour exploiter au mieux ces données. La qualification n'est pas un élément négligeable, car elle atteste que les données acquises permettront de créer un produit final avec une qualité attendue. Nous observons qu'il existe deux approches traitant de cette question [Boulaassal, 2010], [Boudet, 2007]. L'une, nommée évaluation qualitative, consiste à effectuer une analyse visuelle et à noter la qualité avec un sens sémantique, en lui attribuant des coefficients (par exemple : faux, approché ou correct). Elle permet de constater la présence de fautes (erreurs grossières). La seconde, quantitative, se réalise ultérieurement avec un calcul d'indices de qualité. Elle fournit une statistique descriptive, son but étant par définition, de décrire par des statistiques (la moyenne, l'écart, etc.) des données.

Dans cette partie, nous nous intéresserons à l'évaluation de la qualité, en termes d'exactitude et de précision, des relevés laser acquis par un système mobile. Après avoir précisément défini ce qui est entendu par les notions susmentionnées (2.3.2), nous examinerons l'état de l'art des moyens actuellement disponibles pour les mesurer (2.3.3). Aussi, on se devra d'évoquer les autres critères complétant les informations sur la qualité de nuages de points.

2.3.1 Différents types d'erreurs en topographie

Divers types de levés topographiques exigent, certes, des niveaux de précision différents. Connaître l'exactitude des résultats nécessite une bonne maîtrise des appareils utilisés, la

prise en considération de cas particuliers et de situations défavorables, mais aussi, un savoir-faire pour les compenser. Les observations peuvent être entachées par :

1) des erreurs parasites ou des fautes – étant des incertitudes grossières dues à des inadvertances opératoires, un oubli ou bien encore un défaut de réglage de l'appareil. La faute est aisément repérable donc, peut et doit être supprimée. Pour déceler d'éventuelles fautes susceptibles d'être commises, des mesures de contrôles doivent être réalisées.
2) des erreurs systématiques, parfois appelées biais, étant une composante de l'erreur de mesure qui, lors des mesurages répétés, demeure constante en valeur absolue et en signe ou varie de façon prévisible. Ces erreurs sont cumulatives par voie d'addition et dues souvent aux imprécisions de l'instrument (qualité des composants, défauts d'étalonnage, temps de réponse, réglage de zéro, etc.) et aux contraintes de sa mise en œuvre. Les erreurs systématiques influencent l'exactitude (ou la justesse) d'une mesure et ne peuvent être atténuées par une augmentation du nombre de mesurages. La distribution des mesures sera, par la suite, déplacée de la valeur réelle.
3) des erreurs accidentelles (aléatoires) – les plus fréquentes, se produisant de manière aléatoire, en valeur et en signe, même quand les conditions de mesures sont identiques. Or, d'une mesure à l'autre, la valeur obtenue est surévaluée ou sous-évaluée. Les origines sont multiples : ce peut être dû à l'opérateur (par exemple : l'exactitude avec laquelle l'œil observe), à la qualité inévitablement limitée de nos instruments, mais aussi au milieu en lui-même (température ambiante, humidité, etc.). Ces erreurs subsisteront et entacheront la grandeur que l'on recherche et provoqueront une dispersion des résultats. Elles affectent essentiellement la précision des mesures.

Dès que nous commençons à envisager la qualité des données collectées, nous souhaitons l'interpréter comme la confiance accordée en vue de la performance des techniques mises en œuvre et de la capacité des appareils utilisés. Partant de cette connaissance antérieure à l'acquisition, on peut en déduire une inexactitude théorique (*a priori*) entachant les données. Une fois l'acquisition achevée, des écarts empiriques (*a posteriori*) par rapport à une valeur attendue ou bien encore des propriétés de distributions des mesurages (ex. la moyenne, l'écart-type) peuvent être calculés. Ces valeurs servent à décrire l'exactitude et la précision. Ces deux qualificatifs, par ailleurs, sont quelquefois confondus ce qui amène à les considérer improprement comme des synonymes. Pour aller plus loin dans notre propos, on se doit de clarifier ces deux notions en les définissant, puisqu'elles se réfèrent à deux propriétés bien distinctes.

La précision ou *la fidélité* (*precision*) des données est l'étroitesse de l'accord entre les valeurs mesurées $\widehat{m_i}$ obtenues par des acquisitions répétées de même grandeur. Elle est quantifiable à partir d'un seul jeu de données et permet de définir la dispersion des résultats (l'erreur apparente V_i). Selon les conditions, la précision peut être représentée par la répétabilité (*repeatability*) ou la reproductibilité *(reproducibility)* d'une mesure. *La répétabilité*, contrairement à *la reproductibilité*, est définie à travers des mesurages effectués par une même personne, dans un même lieu, avec le même appareil et dans un laps de temps très court. En revanche, dans le cas des nuages de points, il est quasiment impossible de remesurer un point plusieurs fois. En effet, une estimation de la précision des données demeure complexe.

L'exactitude (*accuracy*) représente une concordance entre une valeur mesurée et une autre considérée conventionnellement comme exacte. Elle n'est quantifiable que par rapport à une véritable référence extérieure aux mesures. Son évaluation est toujours délicate puisque l'on ne dispose jamais de valeur réelle M.

Au-delà, pour des mesures multiples, la notion de *justesse de mesure* (*trueness*) définie par un écart du barycentre de l'ensemble de mesures (moyenne arithmétique) par rapport à la valeur

de référence M, est également employée. C'est ainsi l'erreur vraie de la moyenne arithmétique.

Afin de caractériser la précision d'opérations de mesures, nous pouvons introduire l'Erreur Moyenne Arithmétique EMA correspondant à la moyenne des résidus e_i pris en valeur absolue, soit :

$$EMA = \frac{\sum_{i=1}^{N} |e_i|}{N} \tag{2.6}$$

On note alors $e_i = \widehat{m_i} - M$ l'erreur vraie d'une mesure (écart de la valeur observée d'une des mesures $\widehat{m_i}$ par rapport à la valeur vraie M), tandis que $V_i = \widehat{m_i} - \bar{M}$ l'erreur apparente d'une mesure (écart de la valeur observée $\widehat{m_i}$ par rapport à la moyenne arithmétique $\bar{M}$). Sur cette base, nous définissons le carré moyen des erreurs, soit l'Erreur Quadratique Moyenne EQM (*MSE* pour *Mean Square Error*). C'est la moyenne arithmétique des carrés des écarts e_i entre les prévisions M et les observations $\widehat{m_i}$, souvent utilisée pour comparer plusieurs estimateurs, notamment lorsque l'un d'eux est biaisé. Nous la définissons via la formule mathématique :

$$EQM = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (e_i)^2} \tag{2.7}$$

Puisque la définition (2.7) exige que les erreurs vraies soient connues ce qui n'est pratiquement jamais le cas, nous utilisons généralement à la place l'écart-type σ de la distribution des valeurs de $\widehat{m_i}$ défini par :

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N} {V_i}^2}{N-1}} \tag{2.8}$$

C'est la racine carré du quotient de la somme des carrées des erreurs V_i par le nombre N de ces erreurs ou résidus, moins 1.

La Figure 2.17 contient une représentation classique des concepts de tir sur cible, envisageant quatre configurations possibles. Elle illustre les deux termes évoqués qui seront dorénavant utilisés dans ce mémoire pour quantifier les relevés laser mobiles. Nous remarquons que les données peuvent être exactes sans être précises (Figure 2.17c) et inversement (Figure 2.17b). Or, l'exactitude et la précision d'un résultat se vérifient toujours séparément.

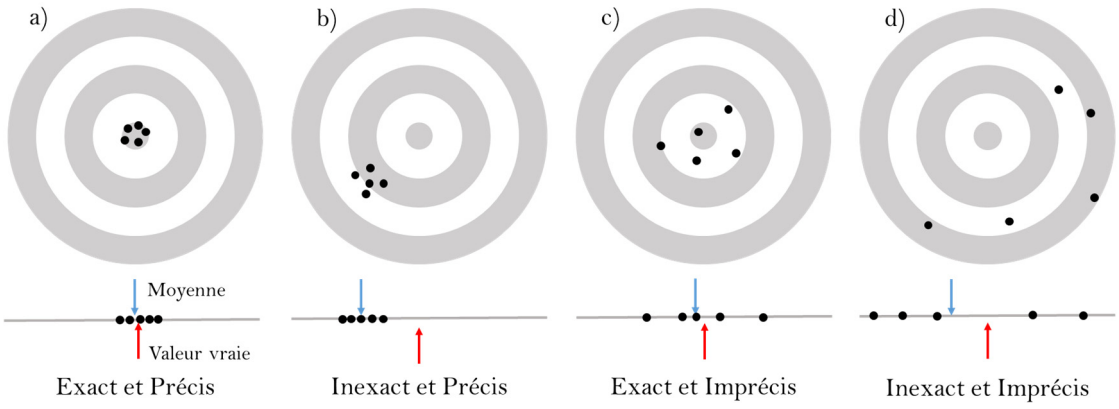

Figure 2.17 Exactitude vs. Précision d'une série de mesurages

Les définitions ci-dessus s'inspirent du *VIM* (*Vocabulaire International de Métrologie*) [JCGM 200:2012] disponible sur le site internet du Bureau International des Poids et Mesures (*BIPM*) et sont semblables à celles introduites par des éditoriaux de [Brabant, 2011] [Newby, 2011] et [Hullo, 2013].

2.3.2 Critères servant à décrire la qualité de nuages mobiles

La problématique de l'évaluation de données laser n'étant pas triviale, nous pouvons définir plusieurs critères fournissant des informations sur la qualité des données. Pour leur part [Cahalane *et al.*, 2010], affirment que la performance du système de cartographie mobile doit être étudiée en termes de résolution, d'exactitude et de répétabilité des données acquises. Quant à la résolution, elle se calcule en prenant en compte la répartition des points, à savoir l'espacement entre les profils voisins, mais aussi celui entre les points appartenant au même profil. Par le terme de répétabilité, nous exigeons que le système mobile soit capable de produire des données de même qualité lors de chaque acquisition. Mais, il faut garder à l'esprit que le fonctionnement du système mobile dépend fortement de la qualité du signal *GNSS*. Par conséquent, le balayage effectué quelque temps après pourra être exposé aux différentes configurations des satellites.

Une procédure semblable de validation est développée par [Yoo *et al.*, 2010], effectuant une analyse des relevés laser mobiles. Trois critères de qualité notamment la précision, la résolution et la complétude sont parallèlement étudiés. Toutes ces notions servent à attribuer une note unique synthétisant l'évaluation globale des nuages mobiles. La résolution permet de vérifier la répartition spatiale des points et donc le niveau de détail dans la scène numérisée. Il est évident que si les scanners fonctionnent à résolution angulaire fixe, l'emplacement entre les points varie fortement en fonction de la distance scanner-objet. Les zones très éloignées de la plate-forme seront donc acquises avec une résolution sous-densifiée et celles plus proches avec une résolution sur-densifiée. Cependant, dans une configuration optimale, les points doivent être répartis de manière homogène. Afin de bien étudier cette problématique, deux termes sont introduits : la densité locale et la densité moyenne. Tous deux représentent le nombre de points par m². La densité locale (2.9) associée à chaque point se calcule pour un voisinage local, étant incluse dans le rayon d'une sphère (r). La distribution des points dans le nuage, d'après ce critère, sera visualisée par un outil statistique appelé « boîtes à moustache ». Enfin, nous complétons ultérieurement les analyses avec la densité moyenne (2.10).

$$D_i = \frac{n_i}{\pi r^2} \tag{2.9}$$

$$\bar{D} = \frac{1}{n_T} \sum_{i=1}^{n_T} D_i \tag{2.10}$$

où :
n_i – nombre de points dans le voisinage du point i
n_T – nombre total de points dans le nuage

S'appuyant sur les calculs effectués, le Taux de Bonne Résolution (*TBR*) est aussi déterminé selon l'équation (2.11). Celui-ci représente le pourcentage de zones ayant une densité satisfaisante.

$$TBR = \frac{n}{n_T} \tag{2.11}$$

où :
n – nombre de points pour lesquels la densité est supérieure à la densité minimale souhaitée.

Le dernier critère, la complétude, interprète et mesure la surface totale couverte par les points, puisque les acquisitions mobiles souffrent de problèmes d'occultations et de manque de visibilité.

[Mano *et al.*, 2012] développent un autre processus de qualification des nuages de points mobiles. Ils regroupent quatre indicateurs décrivant : 1) la qualité du trajet dérivée à partir du nombre de satellites visibles et du coefficient *DOP* ; 2) la précision ; 3) l'exactitude absolue ; et 4) l'exactitude relative. Sachant que le système étudié est équipé de quatre scanners laser, déployés dans une configuration spécifique, la vérification de la précision des données consiste à comparer différents nuages de points captés indépendamment lors d'un seul passage. L'information redondante entre les parties se chevauchant partiellement sert à délivrer une note. Quant à l'exactitude absolue, elle peut être déterminée à l'aide des points de contrôle mesurés au préalable. Le dernier critère, l'exactitude relative, suit le même principe que l'étude de précision. La seule différence réside dans le fait que l'acquisition des données est répétée au bout d'un certain temps. En assimilant les termes introduits dans la section précédente, nous pouvons constater que cette notion correspond à la définition de reproductibilité de données.

Tous les paramètres de qualité susnommés semblent être pertinents pour les besoins d'évaluation des nuages de points. Néanmoins, comme le rapporte [Williams, 2012], le comité d'*ASPRS* (*American Society of Photogrammetry and Remote Sensing*) recommande de standardiser les procédures existantes, et de quantifier les relevés laser à l'aide seulement de deux paramètres : la résolution et l'exactitude. Ces facteurs amènent à attribuer une note commune (*rating*), tout en maintenant une distinction sur neuf niveaux de qualité (Figure 2.18). Une telle solution permettra aux utilisateurs de choisir des données satisfaisant leurs exigences. Alors, la note 3A attribuée signifiera la densité élevée par-devers l'exactitude faible, *et cetera*. Toutefois, ces données peuvent être suffisantes pour certaines applications telles que la vérification du nombre de poteaux le long de la rue. Néanmoins, l'auteur ne précise pas les valeurs numériques de résolution et d'exactitude prédéterminant les notes.

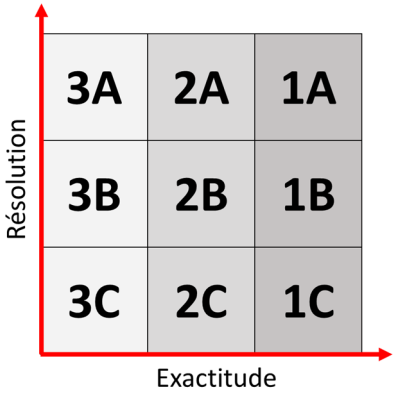

Figure 2.18 Niveaux de qualité définis ralliant l'exactitude et la résolution de données

2.3.3 État de l'art sur l'évaluation de l'exactitude de relevés laser mobiles

L'évaluation de l'exactitude et/ou de la précision de relevés lasers mobiles occupe une place particulière dans les recherches actuelles. Nombreux sont les ouvrages décrivant différents types de travaux à ce sujet. L'exactitude peut être considérée comme absolue et relative. *L'exactitude absolue* est une estimation de la qualité réalisée par rapport à la véritable position dans un cadre de référence terrestre. Elle répond à cette interrogation : Comment un nuage de point est positionné dans un référentiel. Cependant, *l'exactitude relative* estime la qualité de

la mesure d'un vecteur entre deux points. En la mesurant, nous recherchons l'exactitude des points les uns par rapport aux autres, ou bien encore par rapport à l'origine du scanner laser.

Dans toutes les approches que nous allons aborder, l'exactitude sera déterminée par rapport à des données plus précises que celles étudiées. Une référence de grande qualité, saisie par opérateur et géo-référencée, est ordinairement employée. Celle-ci peut être établie lors d'un levé topographique (tachéométrie, *GPS*), une acquisition laser statique, ou bien encore extraite de différentes bases de données telles que : des orthoimages, des modèles de ville 3D, des nuages *LiDAR* aéroporté, des bases de données topographiques ou des plans cadastraux 2D. La plupart des méthodes aboutissent à une comparaison de coordonnées de points de contrôle homologues, distinguables dans les données étudiées. Ces éléments sont fréquemment sélectionnés comme : 1) des sommets de marquages au sol [Gandofi *et al.*, 2008], [Barber *et al.*, 2008], [Guan *et al.*, 2013] ; 2) des poteaux et des lampadaires [Yoo, 2010], [Ridene, 2010] ; 3) des coins de bâtiments et de trottoirs [Kaartinen *et al.*, 2012], [Guan *et al.*, 2013] , 4) des détails architecturaux [Poreba et Goulette, 2012] ; 5) des mires 3D spécialement conçues [Feng *et al.*, 2008]. Certes, une telle approche souffre d'incertitude quant aux points sélectionnés, puisque nous ne pouvons quasiment jamais indiquer deux points homologues dans les nuages de points distincts. Ceci est causé par la nature des données laser mobiles qui sont hétérogènes et anisotropes (leur distribution varie suivant la direction). Ainsi, plus la densité des nuages de points est faible, plus la capacité de sélectionner de points appropriés est limitée. Même si des cibles sont employées, il n'existe aucune certitude que le faisceau laser balayera leurs centres. En outre, la qualité de chacun des points laser dépend fortement de la réflectance de matériaux et de l'orientation d'objet vers le scanner. La vérification de l'exactitude fait alors face à un certain nombre de défis.

Comme il a été explicité par [Barber *et al.*, 2008] et [Williams, 2012], l'évaluation de l'exactitude altimétrique est assez aisée par rapport à celle planimétrique. Ceci est dû au fait que le calcul du biais planimétrique, contrairement à celui altimétrique, exige souvent l'identification manuelle de points correspondants qui est ardue, malgré l'utilisation d'informations supplémentaires telles que l'intensité (Figure 2.19).

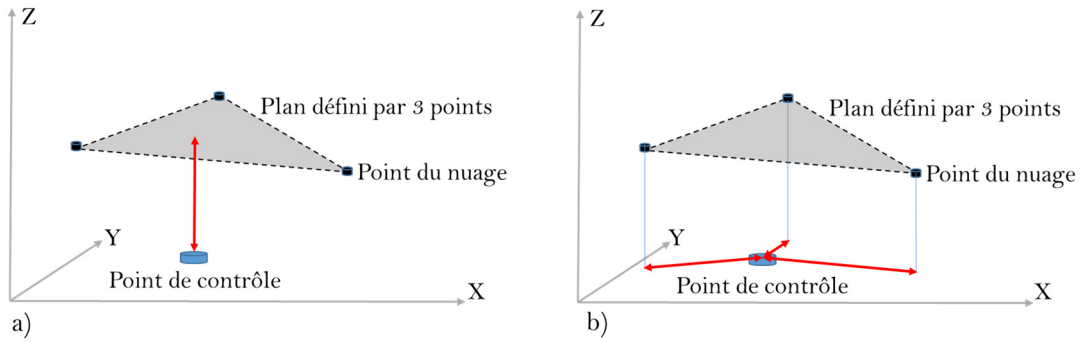

Figure 2.19 Incertitude portant sur la mesure de l'exactitude : a) estimation de l'exactitude verticale ; b) estimation de l'exactitude horizontale (Illustration reproduite à partir de [Williams, 2012])

C'est pourquoi [Ray et Graham, 2008] proposent une méthode d'analyse d'exactitude planimétrique fondée sur les primitives linéaires (des bords de marquages au sol) extraites, à la fois, depuis des données *LiDAR* et des orthoimages. Mais, une telle approche paraît être inappropriée pour évaluer les nuages de points captés par une plate-forme terrestre, puisqu'elle est censée fournir des données de précision et de résolution beaucoup plus élevée. Concernant l'erreur verticale, d'après [Kaartinen *et al.*, 2012], elle peut être calculée à l'aide d'un modèle numérique de terrain (*MNT*).

Un autre point de vue est mis en avant par [Haala *et al.*, 2008] qui choisissent comme référence des modèles de bâti 3D à partir desquels les surfaces planes seront isolées de manière semi-automatique. Il existe aussi d'autres méthodes indépendantes d'extraction de points de contrôle. Elles amènent à comparer des relevés laser entre eux. [Alshawa 2010] confrontent les données mobiles avec les nuages statiques. Dans la même veine [Mano *et al.*, 2012] appliquent un algorithme nommé *LS3D* (*Least Squares 3D*) de [Gruen et Akca, 2005] afin d'évaluer la précision et l'exactitude relative. Dès lors, le recalage rigide basé sur différentes formes extraites depuis les données laser est réalisé. À titre d'explication, nous observons que les trois dernières techniques considèrent l'écart résiduel calculé entre deux nuages de points comme une mesure de qualité.

[Feng *et al.*, 2008] élaborent une méthode comprenant l'évaluation qualitative et quantitative des données laser. Dans un premier temps, un contrôle visuel de l'angle entre les façades est effectué. En admettant que les murs adjacents d'un bâtiment forment un angle droit, la qualité est notée comme faible ou élevée en fonction de la déviation observée. Plus le nuage est déformé, plus la valeur d'angle est loin de 90°. La dernière étape vise à vérifier l'exactitude absolue au sein d'une zone de test. À cet effet, des cibles 3D conçues pour faciliter l'extraction de détails caractéristiques sont employées. L'exactitude est déterminée par comparaison de coordonnées de cibles correspondantes : celles mesurées avec une station totale avec celles extraites dans les données laser.

[Alshawa, 2010] propose une évaluation de l'exactitude par rapport à différentes références. D'abord, un profil horizontal extrait au niveau du sol du relevé laser est comparé avec l'emprise des bâtiments du plan *CAO* (plan cadastral 2D). Une autre solution envisagée consiste à rapprocher, par recalage basé sur l'*ICP* (*Iterative Closest Point*), les données laser et à calculer la distance résiduelle – l'indice de qualité. Deux approches possibles sont suggérées : la comparaison mobile-statique mais aussi celle mobile-aéroporté. Néanmoins, il est à noter que l'utilisation des données *LiDAR* comme référence soulève des préoccupations bien qu'un avion soit équipé d'une centrale inertielle de haut de gamme et qu'une réception du signal *GNSS* ne souffre pas trop de masques. Un nuage *LiDAR*, ayant souvent une densité moins élevée, est censé avoir une précision relativement supérieure que des nuages mobiles terrestres. Pourtant, en raison du faible recouvrement et d'une densité moins importante de données laser aéroportées, une comparaison basée sur le calcul de distances mutuelles est délicate à entreprendre. Afin de surmonter cet obstacle, il est possible d'ajouter des plans aux sous-nuages afin de déterminer l'écart. Au contraire, quand on utilise un nuage acquis en mode fixe, il est possible de calculer directement des écarts sous forme de distances entre les points du nuage mobile et le maillage généré.

Une méthode tout à fait différente est développée par [Williams, 2012]. Elle aboutit à une comparaison des coupes verticales 2D extraites depuis un nuage de points avec celles mesurées à l'aide d'une station totale. Un carrefour a été choisi comme zone de test et quatre profils en travers ont été disposés, un sur chaque aile du croisement. Puis des points équidistants situés le long des profils ont été mesurés. Cette référence a permis de calculer et de comparer des pentes correspondantes.

L'évaluation de l'exactitude étant fastidieuse, nous pouvons saisir pourquoi, grâce à quelques approches mises en œuvre par les chercheurs cités ci-dessous. Il est indéniable que la solution la plus répandue consiste à qualifier, sitôt après une acquisition, les relevés laser mobiles. Néanmoins, la précision théorique est aussi abordée pour déterminer, par l'intermédiaire des incertitudes, le facteur a priori affectant les nuages de points [Barber *et al.*, 2008], [Shaer, 2010], [Alshawa, 2010] et [Bitenc *et al.*, 2011]. Puisque les erreurs élémentaires de capteurs interagissent entre elles, la valeur finale suit alors la formule de propagation d'erreurs. Comme nous l'observons dans le Chapitre 3 les sources d'erreurs liées au fonctionnement d'un système mobile sont multiples. [Barber *et al.*, 2008] concluent

qu'elles proviennent principalement de la qualité du positionnement, mais aussi de la justesse des scanners laser et enfin, de manière moins importante, de la synchronisation de ces différentes données. Quant au récepteur *GNSS*, chaque point de la trajectoire reçoit une estimation de qualité, caractérisant sa contribution à une erreur totale du système. Concernant les scanners laser, à défaut de spécifications techniques suffisantes, fournies par le constructeur et décrivant les éventuelles erreurs, il nous appartient d'opérer un étalonnage. Dans le but de prévoir la précision des nuages de points *LiDAR*, un modèle de propagation d'erreur a été proposé par [Shaer, 2010]. Il y dévoile quatorze états d'erreurs, à savoir :

1) six erreurs de navigation – trois erreurs de positionnement en absolu, ainsi que trois erreurs d'orientation, toutes estimées lors de la fusion *GNSS/INS* ;
2) six erreurs d'étalonnage – erreurs résiduelles liées à l'étalonnage de l'ensemble du système dont trois décrivent les bras de levier et d'autres étant associées aux angles *boresight* ;
3) deux erreurs internes de capteur laser – la précision des mesures d'angles et de distances.

La transmission de toutes ces imprécisions suit la formule de géo-référencement direct (2.1) afin de dériver la précision finale. Comme le souligne [Shaer, 2010], les études peuvent être également complétées par une analyse d'influence de la géométrie du balayage sur la qualité des nuages de points. Pour ce faire, nous prenons en compte deux critères : l'angle d'incidence et la taille de l'empreinte laser. Étant donné que la qualité des relevés laser est une accumulation d'erreurs aléatoires et systématiques propagées par le modèle de géo-référencement et celles dues à l'analyse de géométrie, toutes ces composantes contribueront à la valeur globale d'erreur. Cette constatation nous permet d'attribuer un indicateur de qualité unique (nommé q-indicateur) à chaque point laser. La qualification des nuages de points peut être désormais effectuée automatiquement durant l'acquisition (à la volée).

[Bitenc *et al.*, 2011] cherchent, eux aussi, à vérifier la précision du système de cartographie mobile terrestre. Leur démarche se traduit par une absence de données pouvant servir de référence fiable, car les études ont été orientées vers la numérisation de dunes sans cesse mouvantes. La procédure mise en œuvre, étant similaire à celle de [Shaer, 2010], fait appel au modèle de géo-référencement direct. Nonobstant, elle se limite, telle que présentée, à l'évaluation de la précision horizontale. Pareillement, les erreurs systématiques du système ont été prises en compte ainsi que la géométrie de balayage performé.

2.4 Expérimentation

Comme nous venons de le constater, toutes les méthodes estimant l'exactitude, observées dans la section précédente, visent à acquérir une référence plus précise que les données étudiées. Cette solution prend du temps et semble être coûteuse, nécessitant aussi la recherche de cibles suffisamment bien représentatives. Nous avons relevé ce défi en décidant d'examiner la pertinence de quelques approches similaires à celles relevées dans la littérature. Pour les besoins de nos analyses, nous avons voulu éviter la mise en place de cibles et donc, nous nous sommes référés à des détails caractéristiques d'une scène conçue par l'homme.

La zone de test a été choisie à Paris à proximité du Musée d'Orsay. La qualité du nuage de points *LARA3Dv2010_Orsay* (Figure 2.7) acquis par notre plate-forme mobile pour laquelle : la vitesse approximative était de 5km/h, la résolution des scanners *SICK* était de 0.25 degré et sa fréquence de 18Hz, a été vérifiée. Comme l'affirme [Deschaud, 2010], la spécificité de ce nuage est la suivante : la distance moyenne entre deux points voisins dans un profil est de 6.7 cm, tandis que celle entre points de profils successifs est égale à 7.4 cm. Le nuage est alors hétérogène et localement anisotrope. En particulier, les distances d'un point à un autre

le plus proche varient entre 5 cm à 15 cm. Notre point de départ était l'évaluation qualitative permettant d'observer si les relevés laser subissaient des déformations. Ensuite, l'exactitude a été examinée à l'aide de deux références différentes : un levé topographique *Orsay2011_topo*, et un nuage de points acquis en mode fixe *Orsay2009_statique*. Ces données ont été collectées indépendamment après une acquisition mobile comme il a été décrit précédemment (2.2). C'est pourquoi, dans cette section, nous détaillerons toutes les démarches effectuées, ainsi que les avantages et inconvénients des techniques employées.

Comparaison avec le levé topographique

En raison du manque de géo-référencement – chaque jeu de données n'a pour référentiel qu'un système local - l'estimation de qualité s'est donc limitée à l'exactitude relative. Les levés topographiques ont été ajustés sur un nuage mobile à travers une transformation rigide calculée à partir de trois points de contrôle observables dans les deux ensembles.

Dans un premier temps, les relevés laser mobiles ont été étudiés visuellement par rapport à la coupe horizontale mesurée préalablement. Un profil correspondant, dont l'épaisseur était suffisamment large pour inclure celui de référence, a été extrait du nuage de points. Nous avons remarqué que nos données laser étaient suffisamment cohérentes même si elles étaient localement entachées par certaines déformations ; c'est-à-dire, quelquefois étendues en direction du véhicule, puis décalées (Figure 2.20). Cet état de fait est probablement dû aux imperfections liées à l'implémentation du filtre de *Kalman* à l'étape de géo-référencement, mais aussi aux capteurs de navigation de basse de gamme mise en place (notamment le récepteur *GPS* mono-fréquence calculant la position sans corrections différentielles).

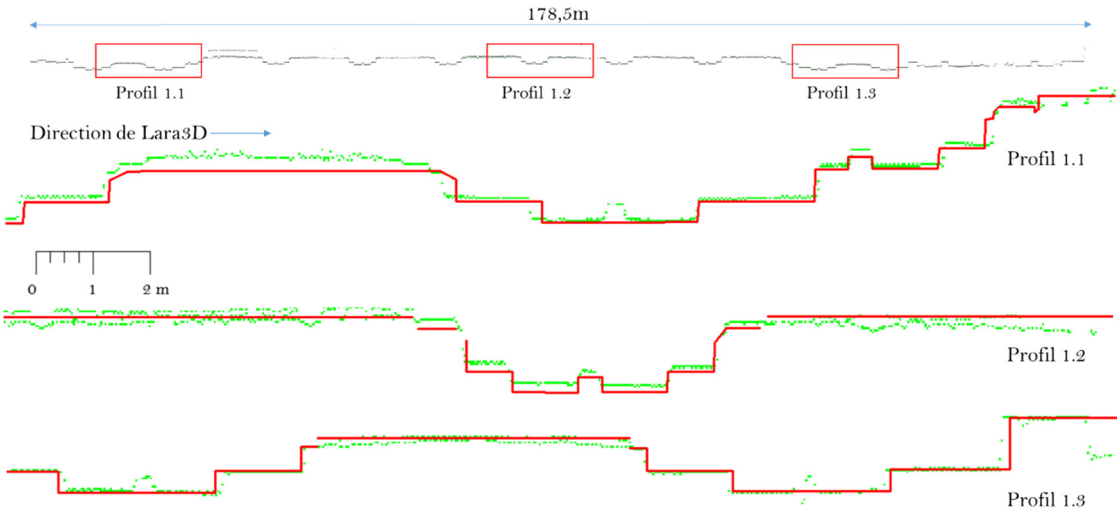

Figure 2.20 Coupe horizontale dans une superposition : levé topométrique (en rouge) – nuage de points (en vert)

Pour compléter nos analyses, une évaluation quantitative a été faite à l'aide des points de détails mesurés. Au lieu de nous appuyer sur les coordonnées, comme c'était ordinairement le cas, nous avons pris en compte les distances mutuelles. La longueur de 43 tronçons reliant les points entre eux sous différentes combinaisons était calculée. Les mêmes éléments ont été manuellement retrouvés et mesurés dans le nuage de points mobiles. La comparaison effectuée a démontré que l'exactitude de nos relevés laser mobiles, étant une moyenne des différences de segments contrôlés, était de l'ordre de quelques décimètres. Les chiffres obtenus ont révélé ainsi que les différences absolues ne dépassaient pas 20 cm pour 79% de segments.

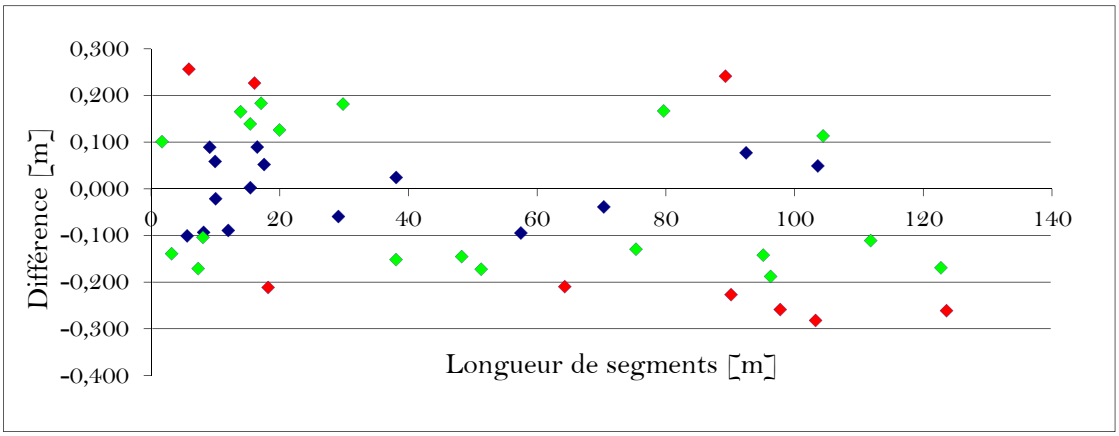

Figure 2.21 Diagramme des différences de longueur de 43 segments de contrôle

La Figure 2.21 montre les erreurs obtenues en fonction de la longueur de segments comparés. Leur grandeur a été différenciée par des couleurs évoluant avec un intervalle de 10cm. Nous pouvons alors constater que plus le tronçon est long, plus l'erreur mesurée est importante, ce qui revient à dire que le nuage mobile est légèrement étiré. En dépit de cette remarque, une différence de longueur égale ou inférieure à 10cm équivaut à une incertitude avec laquelle nous distinguons les détails dans le nuage mobile, parce que sa résolution moyenne était de 7cm. Quoi qu'il en soit, la sélection de points réalisée de manière interactive demeure le point faible de cette méthode. Cette tâche est fastidieuse dans le cas de nuages de points peu denses, et nécessite un temps de traitement conséquent. D'où aussi une source importante d'erreurs, puisque la faible résolution de nuages de points empêche une identification correcte de détails.

Recalage mobile-statique

En définitive, comme le suggère [Alshawa, 2010], un recalage rigide basé sur l'*ICP* a été réalisé. Puisque cet algorithme ne permet pas d'aligner des nuages positionnés et orientés de manière quelconque, comme c'est notre cas, nous avons d'abord recalé grossièrement les deux relevés laser. Tout le traitement a été réalisé à l'aide de Cloud Compare v.2.5.1. Étant donné que le nuage mobile étudié peut être localement déformé à cause d'une qualité temporairement faible du système de positionnement, l'évaluation a été menée de deux façons. Commençant par le recalage global de l'ensemble du nuage, nous avons obtenu une distance moyenne de 18.5cm (calculée sur 26% de points) par rapport à la référence. Pour contrôler la justesse de ce chiffre, quatre coupes transversales ont été créées à des hauteurs différentes afin d'être visuellement vérifiées. Ainsi, une importante perte de qualité du nuage, croissant avec la hauteur de plan sécant, a été notée. Des mesures prises manuellement ont révélé que la distance entre deux nuages de points variait jusqu'à 28.5cm.

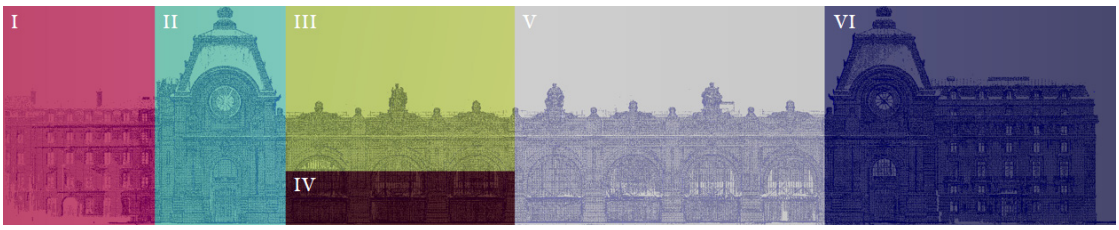

Figure 2.22 Sous-nuages extraits

De ce fait, l'estimation itérative par morceau, ce qui ramène à découper le nuage de points en six sous fragments comme l'illustre la Figure 2.22, a été effectuée. La comparaison, réalisée séparément entre ces sous-nuages, a indiqué l'exactitude finale du nuage de points égale à 9.6cm (Tableau 2.2). Vu que la partie I était considérablement déformée, nous l'avons exclue du processus de qualification.

Tableau 2.2 Distances moyennes calculées pour chaque sous-nuage

Jeu de données	**I**	**II**	**III**	**IV**	**V**	**VI**
Distance moyenne [cm] pour 75% points	17.2	7.1	11.9	9.8	11.0	9.2
Moyenne [cm]	9.6					

Pour conclure, l'utilisation du recalage permet d'évaluer rapidement l'exactitude relative des données mobiles, particulièrement lorsqu'une référence précise est déjà disponible. Néanmoins, si les nuages de points présentaient des déformations locales, dues à la dérive de la centrale et au masquage *GNSS*, une application d'*ICP* provoquerait des déplacements encore plus importants entre les données traitées. C'est pourquoi la valeur de distance calculée n'exprimera correctement ni le *shift* entre les nuages ni l'échelle de ces déformations.

CHAPITRE 3

Erreurs relatives au système MMS

Sommaire :

3.1 Analyse des erreurs liées à la cartographie mobile 40
3.1.1 Erreurs causées par le système de positionnement 40
3.1.2 Erreurs dues au capteur laser 41
3.2 Amélioration des relevés laser mobiles 43
3.2.1 Correction de la localisation en cours d'acquisition 43
3.2.2 Correction en post-traitement 46
3.3 Étalonnage extrinsèque 47
3.3.1 État de l'art sur l'étalonnage des excentricités linéaires 48
3.3.2 Étalonnage des systèmes embarqués de caméras 49
3.3.3 Étalonnage des systèmes dotés de scanners 51
3.4 Impact d'un faux étalonnage sur la géométrie du nuage 54

Résumé

Nous discutons, au sein de ce chapitre, de toutes les anomalies pouvant survenir lors du fonctionnement d'un système MMS et entacher les données laser produites. Elles sont dues notamment aux capteurs mal étalonnés, aux facteurs divers influençant la qualité de balayage laser et à l'étalonnage inexact de l'ensemble du système. Il se peut aussi que le système de positionnement ne parvienne pas à acquérir une localisation suffisante pour collecter des données précises au niveau du canyon urbain. En particulier, les masques fréquents GNSS empêchent la bonne réception des signaux. Ces perturbations entraînent une erreur absolue sur l'estimation de la trajectoire du véhicule qui n'est que partiellement compensée par les informations de la centrale inertielle. De ce fait, un décalage (une dérive) entre la trajectoire fournie par le système de positionnement et la trajectoire idéale est observé. Pour pallier cet inconvénient, il existe un certain nombre de méthodes qui s'appliquent en temps réel ou lors de l'étape de post-traitement. Elles visent à améliorer soit le trajet du véhicule, soit les données laser collectées. Néanmoins, en raison du coût relativement élevé, la majorité des solutions apportées consiste à corriger les nuages après l'acquisition, notamment par un recalage fondé sur l'ICP.

Étant donné qu'un faux étalonnage extrinsèque influencera la qualité de la localisation et par la suite les relevés laser, nous donnons un aperçu des méthodes étalonnant le système intégral. Nous en

ressortons que la plupart des approches nécessitent une zone de test adéquate, munie de cibles bien reparties et mesurées précisément. Concernant les solutions appropriées lorsque des scanners de profil 2D sont embarqués, elles ne sont pas nombreuses et permettent surtout d'étalonner des systèmes aéroportés ALS. Cependant, dans le cas des systèmes MLS dotés de scanners 2D placés latéralement, nous constatons un manque de souplesse quand il s'agit de numériser un environnement plusieurs fois. Un tel dispositif empêche le balayage multiple d'un même objet à partir de directions opposées. Par conséquent, les méthodes conçues pour les systèmes ALS ne répondent pas aux exigences imposées par les plates-formes terrestres. Pour combler cette lacune, les techniques appropriées doivent être avancées.

En dernier lieu, nous démontrons que les écarts d'angles boresight sont observables sur les arêtes extraites, et donc ces dernières pourraient, en théorie, être utilisées pour étalonner le système.

3.1 Analyse des erreurs liées à la cartographie mobile

Nous venons de voir dans le chapitre précèdent que chacun des composants du système peut contribuer à la qualité des nuages de points collectés. Parmi les erreurs les plus notables, ce sont celles provenant du système de positionnement intégrant *GNSS/INS/DMI*. Les autres sources d'anomalies, étant dans la hiérarchie sitôt après le *GNSS*, viennent du manque de précision des points laser, c'est-à-dire qu'elles sont causées par des erreurs instrumentales imputables au scanner laser lui-même, mais aussi à la nature de l'objet numérisé et à l'environnement. Ces erreurs affectent les mesures angulaires et les distances, nous les assimilons à celles connues par les tachéomètres. Elles peuvent être de types systématiques et aléatoires. Enfin, de manière moins importante, les relevés laser mobiles sont entachés par des erreurs de synchronisation de différents capteurs, et par de multiples imprécisions survenues lors de leur étalonnage intrinsèque. L'étalonnage extrinsèque du système aussi ne peut pas être négligé [Barber *et al.*, 2008].

Dans cette section, les sources d'erreurs les plus fréquentes liées au fonctionnement du système *MMS*, et par conséquent, pouvant empêcher la chaîne d'acquisition, vont être présentées. Il s'agit des erreurs provoquées par le système de positionnement (3.1.1), aussi bien que celles dues au scanner laser (3.1.2). La compréhension des relations entre l'environnement et la grandeur d'erreur survenue est nécessaire pour quantifier la précision des données.

3.1.1 Erreurs causées par le système de positionnement

Le système de positionnement est, parmi les autres capteurs, l'élément le plus prédisposé à être affecté par des erreurs [Glennie *et al.*, 2007]. Nous discuterons dans ce paragraphe des problèmes typiques liés à son fonctionnement. La fusion multi-capteurs a pour finalité d'optimiser la précision de la position calculée en profitant d'un capteur pour pallier aux défauts des autres, notamment en recalant en permanence la dérivée de l'*IMU* par l'information *GNSS*, voire de fournir une trajectoire même en l'absence de signal *GNSS*. Pour mettre en évidence les erreurs propres aux capteurs, nous observerons deux composants principaux du système : le positionnement par *GNSS* et la navigation inertielle *INS*.

Dans le cas d'un système mobile de cartographie, le positionnement à corrections différentielles *dGPS* se trouve être le plus répandu. Une telle solution permet, par défaut, de réduire plusieurs erreurs inhérentes au système telles que celles d'horloges, d'éphémérides, la propagation du signal dans l'ionosphère et la troposphère. Pourtant, la qualité de la position demeure toujours fortement liée à la configuration des satellites, c'est-à-dire à leur constellation. Des satellites uniformément répartis sont une situation préférable contrairement à celle où ils se retrouvent tous dans la même portion du ciel. La dégradation de la précision géométrique *GDOP* (*Geometric Dilution Of Precision*) est un indice quantifiant cet impact. Il faut encore mentionner les problèmes des multi-trajets et ceux des masques

GNSS, inextricablement liés à des canyons urbains. Ces derniers sont dus au blocage du signal, engendrés par les arbres, les tunnels ou les hauts bâtiments. Tandis que les premiers provoquent des erreurs dans la mesure de distance entre le satellite et le récepteur, pouvant atteindre plusieurs mètres (mesures de pseudo-distance) ou quelques centimètres (mesures de phase), les trajets multiples sont difficiles à éliminer puisque les conditions de réflexions des signaux sur des surfaces ne sont pas les mêmes d'un site à l'autre [Abuhadrous, 2005].

La centrale inertielle sert en permanence au calcul de trajectoire. La navigation inertielle étant un système autonome, son avantage est d'avoir une bonne précision à court terme, sans aucune information extérieure. En revanche, ces capteurs sont peu fiables, s'il s'agit de positionnement sur une longue période de temps. Cet état de fait est causé par trois principales sources d'anomalies : des erreurs d'alignement initial (pour les angles roulis, tangage et lacet), des erreurs de capteur inertiel (dérive de gyromètres, biais d'accéléromètres) et de calcul [Titterton et Weston, 2004]. Toutefois, la question de la précision de la centrale inertielle est bien plus complexe qu'il n'y paraît. Mais, les fabricants n'exposent guère le modèle mathématique de calcul de la mécanisation et de la compensation de l'*INS*.

En fin de compte, toutes les plates-formes mobiles se localisant grâce à des systèmes hybrides *GNSS/INS* souffrent d'un biais observable entre les nuages de points redondants. Les pertes temporaires du signal *GNSS* s'avèrent être la cause d'un décalage dans l'estimation de la trajectoire du véhicule. Pareillement, les données laser récoltées avec un système *LiDAR* aérien sont encombrées par des écarts planimétriques et altimétriques, résultant, cette fois-ci, de la dérive (*drift*) de la centrale inertielle.

3.1.2 Erreurs dues au capteur laser

Plusieurs sources d'erreurs peuvent affecter les données acquises par un scanner laser seul. De plus, elles varient d'un instrument à l'autre et dépendent de la qualité du dispositif et de son étalonnage. Ce dernier ayant pour but de corriger les erreurs systématiques inhérentes au système, l'étalonnage doit être régulièrement effectué et non seulement lors de la première utilisation. Aussi, les spécifications techniques transmises par les constructeurs de scanners doivent être vérifiées, puisqu'elles sont obtenues dans des conditions idéales de mesures et ont tendance à être surestimées. Pour pouvoir estimer l'exactitude/la précision de chaque point laser, il faut considérer l'ensemble des facteurs intervenants. [Steiger, 2005] propose, conformément au schéma affiché par la Figure 3.1, de lister toutes les caractéristiques importantes influençant la qualité des nuages de points. La discussion exhaustive, concernant le principe du balayage laser terrestre, mais aussi toutes les sources d'erreurs inhérentes au fonctionnement d'un scanner, est présentée par [Pawleta et Igielska, 2009], [Boulaassal, 2010], [Landes et Grussenmeyer, 2011] et [Landes et al., 2012].

Puisqu'un scanner laser effectue des mesures au même titre que le ferait un tachéomètre, aussi leurs modèles s'avèrent relativement semblables. Plus précisément, il est préférable de mettre en évidence deux caractéristiques – l'imprécision d'angles et de distance. Les deux angles (vertical et horizontal) souffrent des erreurs typiques d'un tachéomètre comme l'erreur de collimation horizontale, l'erreur d'index et l'inclinaison de l'axe vertical. Mais, l'incertitude de la distance est indépendante de celle des angles, car le principe des mesures de chaque grandeur est différent. Parmi ces erreurs, le facteur d'échelle et l'erreur de zéro peuvent être cités. La qualification de la précision de mesure angulaire s'effectue en mesurant de courtes distances verticales et horizontales entre les cibles déployées (p.ex. des sphères). Celles-ci sont comparées ensuite aux mesures de référence obtenues grâce à des appareils beaucoup plus précis. De son côté, l'exactitude de mesure des distances est aussi vérifiable. Étant donné que le bruit affectant le nuage de points lui donne une certaine épaisseur, l'une des façons de le mesurer consiste à balayer une cible plane orientée perpendiculairement au

scanner laser. L'exactitude sur la mesure de distance, exprimée en tant que l'écart-type, peut ainsi correspondre à la dispersion des points autour du plan moyen.

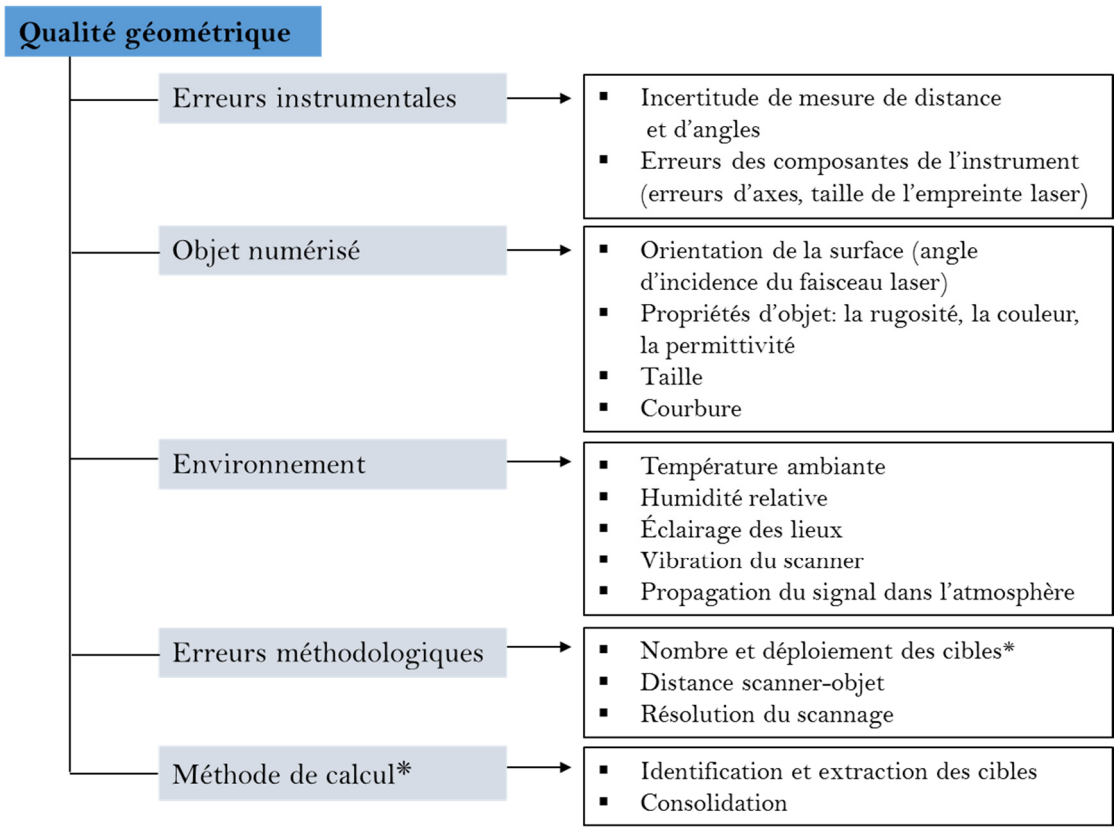

Figure 3.1 Sources d'erreurs sur des mesures avec le scanner laser terrestre
*facteur valable uniquement pour le balayage laser statique

De surcroît, d'autres facteurs ayant un impact sur les points mesurés méritent d'être également envisagés tels que la résolution spatiale, la portée (la distance minimale et maximale à laquelle le scanner laser est capable de fournir des mesures fiables), la taille de l'empreinte laser, l'impact de différentes propriétés des matériaux composant la surface, le problème d'angle d'incidence et la rencontre par faisceau laser d'une arête (échos moyennés).

Plusieurs études concernant le comportement du faisceau laser sur les surfaces composées de multiples matériaux ont été effectuées entre autres par [Boehler *et al.*, 2003], [Hanke *et al.*, 2006], [Feng *et al.*, 2008], [Hiremagalur *et al.*, 2009], [Pawleta et Igielska, 2009] et [Boulaassal, 2010], [Landes et al., 2012]. La caractéristique prise en compte était la couleur, le type de matériel (papier, verre, bois, métal, crépis, pierre), son humidité et sa rugosité. Différentes cibles ont été proposées (Figure 3.2) non seulement pour tester la performance des scanners laser existants, mais aussi pour les comparer. Les résultats obtenus montrent que l'élément influençant davantage la qualité est la réflectivité de l'objet à une longueur d'onde et l'orientation de l'objet envers le capteur laser [Pesci *et al.*, 2008]. Une source d'erreurs significatives, à laquelle il nous faut faire attention, est la manière avec laquelle un matériau numérisé renvoie la lumière laser. Tandis que le retour depuis les surfaces blanches est fort, la quantité de lumière retournée par celles noires est beaucoup plus faible. Des matériaux très réfléchissants tels que le marquage au sol, les diapositifs rétro-réfléchissants peuvent produire aussi des effets supplémentaires comme la saturation et le flou lumineux

(*bloom effect*) [Vosselman, 2010]. La saturation étant causée par un excès d'énergie renvoyée, elle apparaît sous forme de points répartis le long de la ligne de visée. Cependant, le flou lumineux se manifeste par un élargissement de l'objet [Williams, 2012]. Pour résumer, la réflectance d'un objet (albédo) joue sur la quantité de lumière reçue par l'instrument après réflexion, en lui permettant de faire la mesure. De manière générale, plus le signal rétrodiffusé est faible (que ce soit par réflectivité faible du matériau ou bien encore par angle d'incidence), plus l'observation sera bruitée. Les analyses de [Soudarissanane *et al.*, 2009] montrent que, dans un cas pratique, un angle d'incidence inférieur à 60 degrés prévaut sur la qualité des nuages de points.

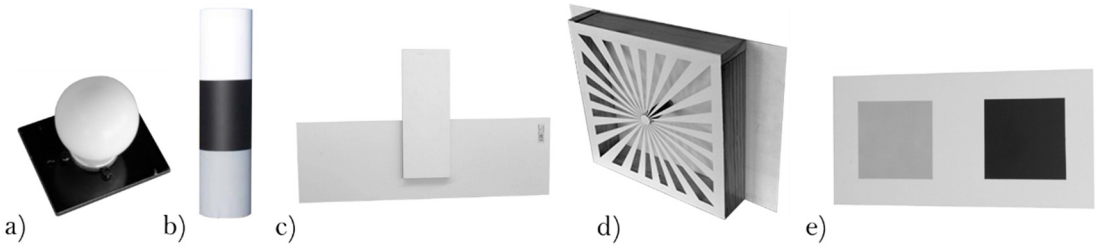

Figure 3.2 Exemple de cibles permettant de tester la performance du scanner laser terrestre :a) Précision angulaire ; b) Angle d'incidence ; c) Échos moyennés – effet de bord (*edge effect*) ; d) Résolution ; e) Réflectance [Boehler *et al.*, 2003]

3.2 Amélioration des relevés laser mobiles

C'est sur cette base que nous observons une nécessité d'améliorer les résultats produits par un système de cartographie terrestre *MLS*. Au regard des solutions apportées, conçues à cet effet, il semble raisonnable de les élaborer en deux paragraphes. La section 3.2.1 se penchera sur les approches visant à renforcer la précision du système de positionnement en temps réel. Les techniques appliquées lors de l'étape de post-traitement seront discutées dans la section 3.2.2. Cependant, la problématique d'étalonnage extrinsèque est élaborée séparément (3.3).

3.2.1 Correction de la localisation en cours d'acquisition

Pseudolites (*PL*)

Puisque la qualité des données « mobiles » est déterminée par la précision du positionnement, la performance de l'ensemble du système en dépendra aussi. Le système intégré *GNSS/INS* joue un rôle important, mais en même temps, son fonctionnement peut se voir affecté du fait des obstructions du signal *GNSS* dans un environnement urbain (atténuations, masquages, trajets multiples). Aussi, un nombre suffisant de satellites dits « visibles » ne peut pas être garanti partout. De plus, même si certains satellites, situés sur l'horizon, sont censés être suivis, nous remarquons que leurs signaux sont entachés par un bruit assez élevé. Une méthode tenant à déployer une constellation locale de satellites grâce à une augmentation des niveaux de signaux *GNSS* a été mise au point. Son but est d'améliorer la précision de la localisation, mais aussi de rendre accessible le signal *GNSS* en intérieur (localisation en indoor). Il s'agit des pseudolites (pseudo-satellites), à savoir des émetteurs terrestres de signaux dont le format est identique ou similaire à ceux émis par les satellites. La portée de ces signaux est variable - elle dépend de leur puissance et de leur utilisation. Le principe est de mettre en place, dans une zone dite de « contraintes » non couverte par le *GNSS*, des transmetteurs permettant à un récepteur d'acquérir un nombre suffisant de signaux et de calculer sa position 3D. Autrement dit, les émetteurs jouent localement le rôle des satellites et leurs signaux sont traités de la même manière que ceux des satellites. La Figure 3.3 illustre le principe de positionnement avec un réseau de pseudolites (PS_i).

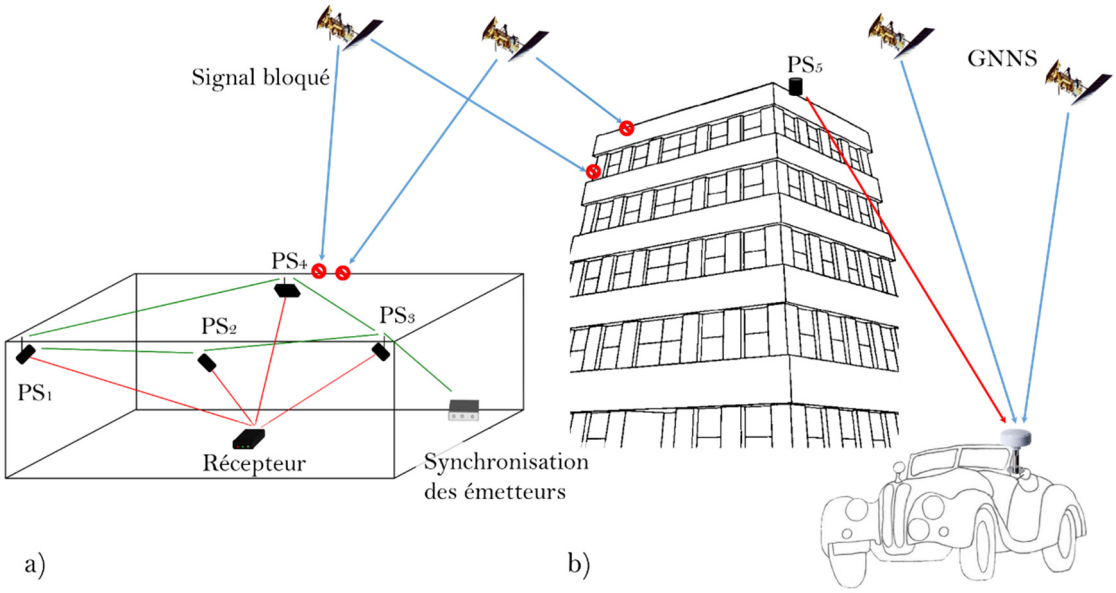

Figure 3.3 Positionnement des pseudolites en : a) indoor (Image reproduite à partir de [Vervisch-Picois, 2010]) ; b) outdoor

Plusieurs équipes à travers le monde travaillent sur la localisation avec les pseudolites. Parmi les derniers travaux concernant la localisation en indoor, nous pouvons citer ceux présentés par [Vervisch-Picois, 2010], [Selmi, 2013]. Néanmoins, l'efficacité d'une telle solution, c'est-à-dire la fusion *GNSS/PL/IMU* pour les besoins du système mobile terrestre *MMS* a été aussi testée par [Wang *et al.*, 2001], [Babu et Wang, 2009] et [Grejner-Brzezinska *et al.*, 2010]. Le concept de la mise en œuvre des pseudolites est toujours en cours de développement, mais les résultats obtenus montrent déjà leur potentiel. [Ning *et al.*, 2007] affirment que les mesures effectuées à l'aide de pseudolites ont un facteur *DOP (Dilution of precision)* réduit d'au moins 35%. En revanche, comme l'explique [Selmi, 2013], la nécessité de synchroniser les pseudolites constitue la phase la plus coûteuse et la plus complexe de cette technique.

Terrain Referenced Navigation (*TRN*)

Le Terrain Referenced Navigation System (*TRN*) sert à calculer la position de la plate-forme en mouvement (avion, *UAV*, véhicule), en comparant sa position actuelle par rapport au sol, sans recourir au *GNSS* dont le signal peut être brouillé par l'ennemi (application militaire), ou voire tout simplement indisponible. Dans ce cas, il nous faut fournir des outils de navigation alternative, lorsque le système *GNSS* n'est plus joignable. En l'absence de données *GNSS*, la localisation d'un véhicule est habituellement réalisée en utilisant une combinaison d'odométrie roue (*DMI*) et de capteurs inertiels. Par contre, une telle solution a deux limitations principales : les centrales inertielles sont sujettes, au fil du temps, à la dérive, et le *DMI* reste sensible aux dérapages et aux glissements de roues. De plus, l'odométrie roue ne donne qu'une information 2D dans le plan de la route.

Grâce au *TRN*, la position et l'attitude sont déterminées, toutes deux étant un support à la navigation inertielle. L'idée consiste donc à combiner la navigation inertielle avec les informations provenant directement du terrain. La performance d'un tel système dépendra de la qualité du capteur *INS* employé. Plus il est exact, plus précise sera la position calculée. Il existe plusieurs approches permettant d'établir la navigation *TRN* y compris le filtre de Kalman étendu (*EKF*), le filtre de Kalman inodore (*UKF*) ou le filtre particulaire [Lee *et al.*, 2013]. Des informations de la scène provenant des images/nuages de points sont employées

afin d'estimer la trajectoire du véhicule, tout en se référant à des méthodes de l'odométrie visuelle. Or, le déplacement relatif d'un capteur optique (caméra, télémètre laser) entre deux points de vue sera déterminé à travers une analyse des images successives. Les méthodes existantes visent à identifier les « *features* » (amers) sous forme de points, lignes ou surfaces pour les mettre ensuite en correspondance. Cette tâche n'étant pas triviale, il existe plusieurs techniques susceptibles d'être appliquées. Dans le cas de l'odométrie basée vision, le mouvement est estimé à partir d'une séquence d'images après la mise en correspondance robuste de points d'intérêts du type Harris [Harris et Stephens, 1988], *SIFT* (*Scale-Invariant FeatureTransform*) [Lowe, 2004], ou *SURF* (*Speeded Up RobustFeature*). Aussi, l'algorithme d'*ICP* (*Iterative Closest Point*) permettant de recaler toute sorte d'entités géométriques entre elles (2D et 3D) peut être utilisé. Pour plus d'informations concernant l'odométrie 3D basée *LiDAR* et vision, le lecteur pourra se référer à [Frémont, 2009]. Néanmoins, comme le constatent [Toth *et al.*, 2009], le rapport du *TRN* avec les applications de la cartographie mobile est double :

1) La localisation précise en position et en orientation (en temps réel) se base sur l'extraction de caractéristiques et elle dépend tout particulièrement de la capacité de la procédure d'isolement des objets mobiles parmi ceux étant statiques. En effet, dans un environnement routier, plusieurs objets sont en mouvement (piétons, automobiles, cyclistes, motos, etc.) et conjointement avec la dynamique propre du véhicule (accélération, décélération), ils constituent un challenge supplémentaire, pouvant altérer la perception.
2) L'utilisation de données déjà existantes telles que des modèles numériques de terrain (*MNT*), des données vectorielles et du *SIG (Système d'Information Géographique)* peut indubitablement soutenir les processus d'extraction de caractéristiques.

D'après [Toth *et al.*, 2009], l'utilisation du *TRN* présente plusieurs intérêts. Mais, il y a des défis à relever pour rendre cette navigation opérationnelle dans des environnements complexes de cartographie mobile. Tout d'abord, la grande quantité de données à traiter (*Big Data*) reste un objet certain de préoccupation. Ensuite, le traitement en temps réel pose un vrai problème tant en terme de capacité que de complexité d'espace numérisé contenant une variété d'objets provenant de dynamiques différentes.

Toutefois, une solution semblable est aussi proposée par [Narayana, 2011]. Il emploie, comme outil complémentaire à la localisation, des scanners laser de profil 2D. Son approche peut être assimilée à la solution conçue pour la localisation en intérieur d'un robot ou d'un véhicule autonome - connue sous le nom de cartographie et de localisation simultanées (*SLAM*). L'ensemble des démarches mises en œuvre fait appel à l'odométrie basée laser, puisque le déplacement relatif est mesuré par rapport à des propriétés invariables de l'environnement balayé par scanner. Notamment les surfaces planes fortement présentes dans les environnements aménagés par l'homme, sont utilisées pour en déduire un mouvement en 3D.

Autres solutions moins complexes

Nous venons d'observer un exemple de solutions à l'aide de localisation qui s'avèrent assez complexes lors de l'étape de l'implémentation. Alternativement [Zampa et Confortie, 2009], font appel à une technique couramment utilisée. Il s'agit d'augmenter la précision de la trajectoire à travers des stations de référence *GNSS* supplémentaires mises en place. De plus, des points de contrôle déployés tous les 50 à 80 mètres le long du trajet servent ultérieurement à corriger les relevés laser. Une approche tout à fait similaire a été aussi décrite par [Gandofi *et al.*, 2008], sauf que la position optimale des stations *GNSS* a été retrouvée grâce à des « pré-acquisitions » réalisées quelques jours auparavant. En dépit de

tous les avantages de cette solution, l'emplacement des stations de contrôle au sol s'avère très coûteux.

3.2.2 Correction en post-traitement

Le second groupe de techniques d'amélioration de performance de système mobile contient des outils permettant la correction de données en post-traitement.

Les travaux de [Tournaire *et al.*, 2006], [Ridene, 2010] et [Monnier *et al.*, 2013] s'appuient sur un recalage entre les données mobiles et les bases de données géographiques mieux géo-positionnées mais beaucoup moins détaillées. [Tournaire *et al.*, 2006] améliorent l'étape de géo-référencement du véhicule à l'aide de l'imagerie aérienne (résolution d'au moins 20cm). Les informations 3D extraites des stéréos – paires sont fusionnées avec celles des orthoimages. À cet égard, des marquages au sol, à savoir les passages piétons, ont été détectés et reconstruits à partir de ces deux types de données image. Ensuite, les coordonnées 3D des points extraits du marquage au sol ont été incorporées lors de la triangulation photogrammétrique (*Photogrammetry Bundle Adjustment*) afin de corriger la localisation absolue des images terrestres. [Ridene, 2010], quant à lui, examine des solutions permettant de mettre en cohérence différents types de données. Il propose certains scénarios de corrections de relevés laser mobiles par rapport à des données externes au système de cartographie mobile (p.ex. nuage de points statique, modèle numérique de surface). Il s'agit du recalage rigide de la famille *ICP* des données 3D. Enfin, le processus de [Monnier *et al.*, 2013] se base sur un recalage non rigide fondé, lui aussi, sur le principe d'*ICP* d'un point à un plan. L'hypothèse que l'erreur de positionnement varie lentement, au fil du temps, de façon non linéaire a été faite. Dès lors, la trajectoire de la plate-forme est déformée itérativement, morceau par morceau, afin de minimiser la distance des points du nuage mobile aux primitives planes du modèle. Ce dernier constitue une base de données géographiques (2D ou 3D) composée de primitives ponctuelles (poteaux, repères de nivellement, sommets d'objets polyédriques), linéaires (arêtes, cadastre, limites de trottoirs) ou bien encore surfaciques (faces d'objets).

Nous pouvons aussi nous confronter aux méthodes semblables à celles de l'odométrie visuelle, mais cette fois-ci, appliquées après les acquisitions. [Cazzaniga *et al.*, 2007] développent une solution s'inspirant de la photogrammétrie, particulièrement de la stéréovision. Leur approche est assez limpide. Il s'agit de comparer la position de la caméra calculée à partir des images stéréoscopiques avec celle prévue par le système de positionnement *GNSS/INS*. Plus précisément, le but est d'évaluer les éléments d'orientation externe (*EOZ*) déterminés indépendamment pour identifier et corriger les interruptions temporaires dans la réception du signal *GNSS*. De même [Shi, 2008] suggère d'effectuer la triangulation des images à partir des points de contrôle précis, extraits, cette fois-ci, à partir des données fusionnées images/laser - afin d'améliorer la position et l'orientation de la caméra.

[Yousif *et al.*, 2010] étudient la validité de théorie de l'assimilation de [Reichle, 2008] pour renforcer le positionnement du véhicule. Selon cette théorie, le processus mis en œuvre vise à s'appuyer sur la corrélation existante entre deux observations différentes d'un phénomène de manière à obtenir la meilleure estimation de l'état réel. C'est pourquoi deux nuages de points acquis lors d'un trajet aller-retour au sein de la même zone sont intégrés.

[Alshawa, 2010] envisage une méthode radicalement différente. Un récepteur *GPS* de « haute sensibilité » (*HSGPS*) est utilisé simultanément avec un *GPS* différentiel (*dGPS*). Ce premier étant moins précis, il fournit, contrairement au *dGPS*, des informations en continu sur la trajectoire. Pour pouvoir traiter une telle configuration, les trajets déterminés indépendamment par deux récepteurs sont comparés. D'abord, l'écart entre ces échantillons est réduit en appliquant, à des données moins précises, une transformation permettant de les

faire coïncider avec celles ayant la précision plus élevée. Une fois les trajectoires fusionnées, une courbe ajustant au mieux ces données, est retrouvée. La dernière étape nous amène à interpoler des points du trajet.

Autre possibilité [Narayana, 2011] présente une méthode visant à corriger, par lissage des données, la trajectoire enregistrée, puisqu'elle souffre de certaines erreurs provoquées par des limitations de capteurs de positionnement. Il en conclut qu'une telle solution étant insuffisante pour surmonter les problèmes dus aux lacunes *GNSS*, que celle-ci peut être toutefois très utile pour corriger les sauts d'altitude et les anomalies d'azimut (lacet) de la trajectoire enregistrée.

3.3 Étalonnage extrinsèque

L'étalonnage géométrique des systèmes mobiles basés sur *LiDAR* revient à estimer et à éliminer les erreurs systématiques qui affectent les nuages de points collectés. Ces erreurs sont causées par les biais de paramètres du système, notamment le biais des paramètres de montage de capteurs (relations spatiales entre les modules), ainsi que par les incertitudes des mesures d'angles et de distance inhérentes aux composantes du scanner utilisé [Habib *et al.*, 2011]. L'étalonnage dans le contexte des systèmes mobiles désigne alors deux pistes à suivre : 1) l'étalonnage de chaque scanner laser (étalonnage intrinsèque) ; 2) l'étalonnage du système intégral (étalonnage extrinsèque).

L'étalonnage extrinsèque contient des procédures nécessaires pour estimer les constantes de calcul, à savoir la position et l'orientation des capteurs de perception (caméra, laser) dans le repère lié à la centrale inertielle (b). Nous les exprimons par : trois composantes de translation $r_{b/s}^{b}$ (excentricité linéaire entre le centre de l'unité inertielle *IMU* et le centre du scanner), et trois en rotation R_s^b (*matrice de boresight*). Au-delà, nous pouvons mentionner d'autres excentricités linéaires (bras de levier) entre les capteurs de localisation, à savoir la distance $r_{b/GNNS}^{b}$ entre le centre de phase de l'antenne *GNSS* et le centre de masse de l'*IMU*, et la distance $r_{b/DMI}^{b}$ entre le centre de l'*IMU* et l'odomètre. Lors de l'étape de géo – référencement direct (formule 2.1), ces deux paramètres servent à calculer la trajectoire de la plate-forme dans le repère local (M). L'étalonnage réalisé avec beaucoup de soin, est alors indispensable pour, d'une part rendre le système opérationnel, d'autre part pour fournir des données de qualité élevée. Sinon, chaque erreur commise dans l'étalonnage du système intégral influencera la qualité de la localisation, et par suite les relevés laser.

Cette section donnera un aperçu des méthodes récentes d'étalonnage extrinsèque d'un système mobile de cartographie, et expliquera brièvement leurs principes. Puisque les translations peuvent être souvent déterminées par des mesures conventionnelles, nous présentons l'une des solutions envisageables. Concernant les composantes en rotation (e_x, e_y, e_z) formant la *matrice de boresight* R_s^b, celles-ci doivent être déterminées indirectement. Cette résultante est due au fait qu'il s'avère impossible de les mesurer en utilisant des instruments employés habituellement pour les levés topographiques et géodésiques. Il existe plusieurs approches permettant d'estimer des excentricités angulaires, mais la majorité nécessite une zone de test munie de cibles uniformément réparties et mesurées. Nous aborderons, au sein de cette section, les techniques existantes en les regroupant en deux catégories : 1) celles dédiées aux systèmes basés sur les caméras (3.3.2) ; 2) celles appropriées aux systèmes embarqués des scanners laser (3.3.3). Ensuite (3.4), nous présenterons une étude d'impact d'un étalonnage imprécis sur la géométrie des arêtes extraites. Les notations employées ont été précédemment expliquées avec la Figure 2.3, dans le Chapitre 2.

3.3.1 État de l'art sur l'étalonnage des excentricités linéaires

L'une des manières de connaître les excentricités linéaires est de les mesurer tout simplement à l'aide des techniques traditionnelles (tachéomètre, récepteur *GNSS*, etc.). Une seule condition requise est de pouvoir mener une observation directe des points caractéristiques. Les deux cibles visées doivent alors être visibles et identifiables. Autrement, l'étalonnage demeure assez compliqué à effectuer et les grandeurs ne pourront pas être déterminées avec une précision suffisante.

La Figure 3.4 illustre une procédure concevable pour mesurer le bras de levier entre l'antenne *GNSS* et la centrale inertielle. C'est une adaptation de l'approche discutée par [Shear, 2010] et utilisée, par défaut, lors de l'étalonnage des systèmes aériens *ALS*.

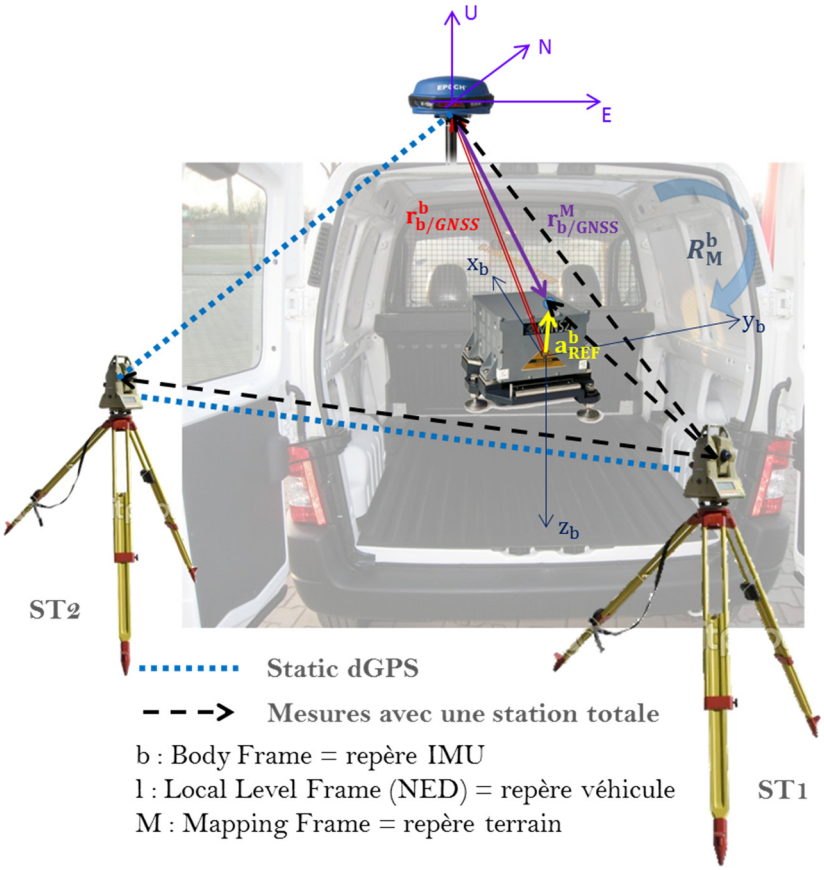

Figure 3.4 Procédure pour déterminer le bras de levier entre les capteurs de localisation

Après avoir fait des mesures *GNSS* et tachéométriques, les translations $r^b_{b/GNNS}$ se calculent comme suit :

$$r^b_{b/GNNS} = a^b_{REF} - R^b_l R^l_M r^M_{b/GNSS} \tag{3.1}$$

où :

a^b_{REF} : offset fourni par le constructeur en raison de l'impossibilité d'observer le centre de l'*IMU*

$r^M_{b/GNNS}$: position de l'antenne *GNSS* déterminée dans le repère terrain (M)

R_l^b : matrice de passage du repère lié à l'*IMU* (b) vers le repère véhicule (l), définie par trois angles (roulis, tangage et lacet)
R_M^l : matrice de rotation permettant un changement du repère véhicule (l) au repère terrain (M), calculée selon la formule $R_M^l = (R_l^e)^t R_M^e$

Également, toutes les excentricités linéaires notamment $r_{b/s}^b$ et $r_{b/DMI}^b$ peuvent être spécifiées à travers une telle procédure. Comme l'affirment [Ellum et El-Sheimy, 2002], [Shear, 2010] et [Li-Chee-Ming *et al.*, 2011], la principale limite de cette méthode est l'impossibilité à observer directement les centres de capteurs. De ce fait, les bras de levier seront connus avec une précision centimétrique. C'est pourquoi cette approche peut être parfois insuffisante et le calcul des paramètres doit se faire d'une autre manière.

3.3.2 Étalonnage des systèmes embarqués de caméras

Les premiers systèmes mobiles de cartographie étaient conçus pour recueillir des informations sur un environnement en se servant des images. Leur principe consistait à adapter des algorithmes bien connus dans le domaine de photogrammétrie. Il s'agissait alors de mettre en pratique une stéréoscopie permettant de reconstituer le relief de la scène à partir d'images prises sous différents points de vue de l'objet. En conséquence, la question de l'étalonnage des systèmes basés sur des caméras était largement discutée dans la littérature.

Excentricités angulaires

Afin d'identifier les excentricités angulaires, deux méthodes semblent être le plus souvent évoquées [El-Sheimy, 2011]. La première possibilité nommée « 1-step » amène à incorporer les trois paramètres de rotation (e_x, e_y, e_z) (3.2) comme inconnues supplémentaires, dans un ajustement par faisceaux, ou dans une compensation par moindres carrés.

$$R_s^b = R_z(e_z)R_Y(e_y)R_X(e_x) \tag{3.2}$$

Une autre solution, plus répandue, est le procédé « 2-step ». Il comprend deux étapes principales. Dans un premier temps, nous déterminons, pour chacune des images, deux matrices de rotation :

1) R_b^M du repère lié à la centrale inertielle (b) vers le repère terrain (M). Elle représente les mesures de l'*IMU* effectuées lors de l'acquisition.
2) R_s^M du cadre capteur (s) vers le repère terrain (M). Ses composants sont calculés par une terra-triangulation assistée par *GNSS*.

Ensuite, les deux matrices, c'est-à-dire, celle dérivée de l'*IMU* et celle déterminée par photogrammétrie, sont comparées selon la formule :

$$R_s^b = \left(R_b^M(t)\right)^{-1} R_s^M \tag{3.3}$$

Pour chaque image prise, il en découle une *matrice de boresight* R_s^b. La moyenne de ses éléments individuels correspond aux excentricités angulaires recherchées. Néanmoins, il faut noter qu'il apparaît impossible de prendre la moyenne directement puisque la matrice de rotation n'est probablement pas orthogonale. Elle doit être d'abord représentée par les angles eulériens ou par les quaternions afin de pouvoir reconstruire la *matrice de boresight* finale.

Bras de levier

Le classement des méthodes étalonnant les excentricités linéaires entre les caméras et le centre de navigation ressemble à celui évoqué pour les excentricités angulaires. Se référant à des techniques indirectes, ces paramètres sont calculés soit pendant l'ajustement :

$$r_{b/s}^{b} = (x_s^M, y_s^M, z_s^M)^T \tag{3.4}$$

soit, pour chaque image ; la position dérivée par système intégré *GNSS/INS/DMI* sera comparée avec celle fournie par la terra-triangulation :

$$r_{b/s}^{b} = (R_s^M)^{-1} \cdot \left(r_{b/GNSS}^{M}(t) - r_s^M\right) \tag{3.5}$$

Enfin, la moyenne de tous les paramètres obtenus décrira l'excentricité linéaire $r_{b/s}^{b}$.

Aperçu des méthodes

Le Tableau 3.1 récapitule quelques approches récentes traitant du sujet de l'étalonnage. Leur exactitude dépendra fortement de la zone de test qui doit être appropriée en vue d'utiliser simultanément le positionnement par *GNSS* et la photogrammétrie.

Pour pallier le problème de blocage du signal *GNSS*, [Hassan et El-Sheimy., 2008] suggèrent de placer sur un véhicule un prisme réflecteur et de le suivre avec une station totale en utilisant le mode continu de mesure (*tracking*). Cette information constituera un support au système de navigation inertielle travaillant de façon automne.

Tableau 3.1 Exemple de travaux étalonnant indirectement les deux types d'excentricités : caméra/*IMU*

Auteurs	Zone de test	Excentricité linéaire	Excentricités angulaires
[Li-Chee-Ming *et al.*, 2011]	Amers naturels	Procédé « 2-step »	
[Rau *et al.*, 2011a]	67 cibles	Nouveau procédé « 1-step » pour l'étalonnage des systèmes multi-caméras	
[Rau *et al.*, 2011b]	Cibles déployées sur les façades	Procédé « 2-step »	
[Chen *et al.*, 2013]	Deux zones de test, chacune munie de points de contrôle	Indoor-et-outdoor étalonnage (procédé 2-step)	

Les études comparatives entre les techniques existantes ont été effectuées par [Ellum et El-Sheimy, 2003] et [Rau *et al.*, 2011a]. Le Tableau 3.2 résume leurs observations, en juxtaposant les avantages et les inconvénients propres à chaque méthode.

Tableau 3.2 Synthèse des approches d'étalonnage

Méthode	Avantage	Inconvénient
Mesures conventionnelles (bras de levier uniquement)	Indépendant de l'ajustement	Observation ardue du centre de capteurs
« 2-step » prendre la moyenne des résultats obtenus simultanément par le système hybride *GNSS/INS* et la photogrammétrie	Conception simple, sans modifications apportées au modèle d'ajustement	Zone de test munie de cibles précises et uniformément réparties
« 1-step » insérer des paramètres d'étalonnage comme des inconnus supplémentaires dans un ajustement	Matrice de covariance contenant une information complète	La corrélation avec d'autres paramètres implique que les résultats peuvent ne pas être fiables. Modifications indispensables au niveau du modèle mathématique.

L'étalonnage du système multi-caméras est aussi abordé par [Cannelle *et al.*, 2012]. Leur objectif est d'estimer, à la fois, les excentricités entre les caméras mises en place, ainsi que celles existantes par rapport au système de positionnement *GNSS/IMU*. À cet effet, deux méthodes sont discutées et comparées. La première, nommée « *OFF-line* », consiste à étalonner séparément chaque caméra. Pour ce faire, une zone de test équipée de cibles 3D dont les coordonnées sont connues avec la précision submillimétrique, servira de référence. Ensuite, les paramètres pourront être déterminés en adaptant une méthode similaire à l'ajustement par faisceaux. L'autre démarche est l'étalonnage « *ON-line* » qui s'avère une solution entièrement automatique, ne nécessitant plus de points de contrôle. Il est fondé sur les images prises durant l'acquisition à partir desquelles les points d'intérêt sont extraits et appariés (p.ex. à l'aide de l'algorithme *SIFT*). Les tests effectués par les auteurs montrent que les deux approches fournissent des résultats assez similaires. L'étalonnage dit « *ON-line* » est exact et efficace. Mais, il peut être peu fiable lorsqu'il est employé dans un environnement empêchant une extraction de points d'intérêt (forêt, etc.).

3.3.3 Étalonnage des systèmes dotés de scanners

Les approches dédiées à l'étalonnage des systèmes dotés d'un *LiDAR* (*MLS* et *ALS*) peuvent être appropriées, en fonction du type de scanner embarqué, soit pour un scanner à double balayage (3D), soit pour un scanner de profil 2D. Cette section esquissera quelques travaux se rapportant à ce sujet.

Tandis que les techniques conçues pour les scanners 3D s'inspirent de celles développées pour les systèmes basés sur des caméras, la problématique d'étalonnage d'une plate-forme *MLS* équipée des scanners 2D est assimilée à celle d'étalonnage d'un *ALS*. C'est dans cet esprit que [Talaya *et al.*, 2004] déploient des cibles réfléchissantes dont les coordonnées sont connues. Ensuite, ils adaptent la technique similaire au procédé « 1-step » afin de déterminer les excentricités linéaires et angulaires entre le scanner laser 3D et le système de positionnement.

D'après [Chan *et al.*, 2013], les techniques récentes d'étalonnage des excentricités angulaires (e_x, e_y, e_z) peuvent être classifiées, compte tenu du principe d'estimation, en deux catégories : 1) l'estimation au sens des moindres carrés, fondée sur les primitives planaires introduisant

des contraintes lors de l'ajustement [Filin, 2003], [Skaloud et Lichti, 2006], [Atanacio-Jiménez *et al.*, 2011], [Glennie, 2012], [Chan *et al.*, 2013] ; 2) le recalage par *ICP* minimisant l'écart entre les points d'une surface (plane ou non plane) extraite à partir de données laser redondantes [Rieger, 2010], [Habib *et al.*, 2010], [Kumari, 2011]. Une autre classification, répandue au sein de la communauté *LiDAR*, est donnée par [Habib *et al.*, 2011]. Ainsi, deux procédures sont distinguées : 1) *system-driven* (étalonnage) ; 2) *data-driven* (ajustement des bandes). L'approche « *system-driven* » se réfère au modèle physique de capteurs permettant d'effectuer des liens entre les mesures des dispositifs individuels d'un système et les coordonnées terrain (M) des nuages de points générés. Vu que l'accès à des données brutes peut être limité, le développement des méthodes « *data-driven* » utilisant uniquement des coordonnées *XYZ* du nuage, est devenu nécessaire. Or, les effets des erreurs systématiques sont modélisées par une transformation arbitraire entre les bandes *LiDAR* et le repère de référence.

[Habib *et al.*, 2010] et [Habib *et al.*, 2011] proposent deux solutions alternatives pour l'étalonnage des systèmes *ALS*. Elles étaient nommées respectivement : 1) *Simplified Method* ; 2) *Quasi-rigorous Method*. La première, étant une approche dite « *data driven* », requiert des bandes parallèles *LiDAR* acquises au sein d'une zone avec un faible dénivelé. En revanche, la seconde procédure, ne nécessitant plus le strict respect des plans de vol (bandes non parallèles autorisées), exige des nuages de points horodatés, ainsi que des données de navigation (uniquement la position de la trajectoire). Cependant, les deux algorithmes minimisent les écarts entre les primitives correspondantes point/maillage triangulaire *TIN* (*Triangulated Irregular Network*) afin de déterminer les paramètres d'étalonnage.

De son côté [Kumari, 2011], suggère une approche d'ajustement des bandes *LiDAR* adjacentes, appropriée à des big data (données volumineuses) où les structures créées par l'homme ne sont pas présentes. L'algorithme a besoin de données d'entrée (nuage de points et trajectoire) et fournit des paramètres de transformation rigide en sortie. Pour ce faire, un échantillon de points d'une bande *LiDAR* est mis en correspondance avec la surface de l'autre la plus proche. L'ajustement des scans consécutifs se réalise à l'aide d'une méthode similaire à l'algorithme *ICP* point à plan.

L'étalonnage d'un *ALS* est aussi une tâche entreprise par [Skaloud et Lichti, 2006]. Une méthode rigoureuse basée sur des surfaces planes telles qu'un terrain de football, des toits inclinés de bâtiments, est discutée. L'idée est d'estimer simultanément des excentricités angulaires et des paramètres de plans à partir des primitives choisies, en ayant seulement une connaissance a priori de leur forme (aucune information ni sur leur position, ni orientation n'est demandée).

[Rieger *et al.*, 2010] de Riegl GmbH présentent, pour étalonner un *MLS*, une procédure basée sur l'algorithme d'*ICP* appliqué sur des primitives planaires extraites automatiquement de nuages distincts. Il s'agit alors de choisir une zone de test contenant un nombre important de surfaces planes (comme les façades de bâtiments) orientées diversement les unes par rapport aux autres. Elles seront balayées plusieurs fois par un scanner à double balayage (*Riegl VZ-400*). La direction de déplacement du véhicule aussi bien que l'angle de balayage laser changent après chaque passage. L'extraction des angles (e_x, e_y, e_z) s'effectue avec la procédure appelée « *boresight alignement and scan data adjustment* », connue pour les applications d'un *ALS*. Son principe repose sur le fait que les nuages collectés doivent coïncider lors des acquisitions répétées d'un même objet, si les deux étalonnages extrinsèque et intrinsèque sont justes. En supposant que les erreurs dues au scanner sont compensées, nous constatons que plus la *matrice de boresight* est exacte, plus la différence entre ces nuages est petite. Un faux étalonnage extrinsèque devrait donc entraîner un décalage entre les surfaces balayées depuis les directions opposées de la plate-forme terrestre. Les tests réalisés dévoilent que la méthode proposée requiert des nuages collectés sous de bonnes conditions

de réception *GNSS*, car le résultat de l'estimation en dépendra. Par contre, l'applicabilité de la méthode au regard des systèmes dotés d'un scanner 2D n'a pas encore été publiée. En choisissant cette approche, il nous faudra aussi trouver une solution pour déterminer le bras de levier $r_{b/s}^{b}$.

Les démarches de [Chan *et al.*, 2013] consistent à employer différentes primitives pour les besoins de l'auto-étalonnage (*self-calibration*) d'un système *MLS* équipé de quatre scanners de profil 2D (*Riegl LMS-Q120i*). Pour améliorer la procédure de l'estimation, ils suggèrent de prendre en compte, en plus des plans, le modèle 3D de courbe caténaire. Ainsi, la formule classique de géo-référencement et les modèles géométriques respectifs élaborent un modèle fonctionnel d'étalonnage. Les trois composantes en rotation (e_x, e_y, e_z), trois en translation $(\Delta x, \Delta y, \Delta z)$, aussi bien que les paramètres de modèles (quatre pour un plan, et trois pour une caténaire) participent à l'ajustement fondé sur le modèle de *Gauss-Helmert*.

Cependant, il existe plusieurs travaux portant sur l'étalonnage d'une plate-forme équipée d'un Velodyne [Atanacio-Jiménez *et al.*, 2011], [Levinson, 2011], [Glennie, 2012], [Chan et Lichti, 2013], [Zhu et Liu, 2013], [Wiart, 2013], [Elsberg *et al.*, 2013]. Le Velodyne est un scanner qui, selon le modèle, possède 32 ou 64 fibres laser alignées le long d'un axe. Un tel dispositif positionné sur un système mobile permet une acquisition à 360° de l'espace. [Atanacio-Jiménez *et al.*, 2011] et [Glennie, 2012] apportent des méthodes d'étalonnage extrinsèque et intrinsèque du Velodyne HDL-64E. L'ajustement, par la méthode des moindres carrés, des primitives planaires fournit alors les valeurs recherchées.

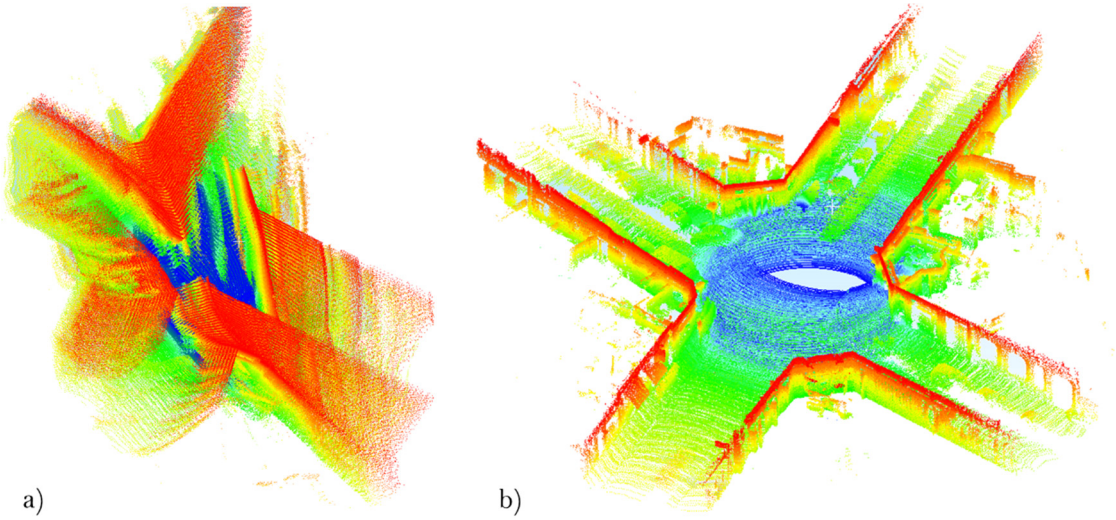

Figure 3.5 a) Nuage de points captés par un Velodyne HDL-32E : a) faux étalonnage extrinsèque ; b) après l'étalonnage extrinsèque [Wiart, 2013]

Dans la même veine [Wiart, 2013] discute des solutions appropriées lorsqu'un *Velodyne HDL-32E* est monté sur un véhicule terrestre. Plusieurs méthodes automatiques de recherche d'étalonnage extrinsèque, toutes faites par itération multi-échelle, ont été analysées. Il s'agit de la méthode de *Levinson* [Levinson, 2011], et de deux nouvelles approches développées par l'auteur : 1) l'algorithme de planéité qui amène à vérifier la planéité locale dans un petit voisinage d'un point ; 2) l'algorithme *Local – ICP Point-to-Plan* qui s'inspire de l'*ICP*. Ce genre de techniques ne nécessite ni cibles spéciales, ni de mesures manuelles, évitant ainsi une tâche très fastidieuse. La démarche de *Levinson* utilise la redondance d'information pour calculer la compacité du nuage de points, et ensuite améliorer, par optimisation, l'étalonnage extrinsèque. Nous observons que plus l'étalonnage extrinsèque est faux, plus le désordre (entropie) au sein du nuage de points est important

(Figure 3.5). L'hypothèse que le nuage de points d'une fibre forme, à condition que l'étalonnage intrinsèque soit juste, des surfaces planes sur des objets plans, est devenue le point de départ de la méthode de *Levinson*. Les tests effectués par [Wiart, 2013] ont montré l'efficacité de cette approche, mais les résultats optimaux ont été obtenus pour des nuages possédant certaines caractéristiques (p.ex. acquisition dans un virage). Deux autres méthodes, c'est-à-dire l'algorithme de planéité et l'algorithme *Local-ICP Point-to-Plan* permettent aussi d'estimer précisément les excentricités angulaires. En revanche, les translations, comme avec la méthode de *Levinson*, sont très peu observables et doivent être déjà connues. L'inconvénient majeur de ces solutions est le temps d'exécution qui empêche l'étalonnage« *ON-line* ».

Enfin [Elsberg *et al.*, 2013] mettent en avant une méthode permettant d'effectuer simultanément un étalonnage intrinsèque et extrinsèque d'un scanner à double balayage (*Riegl VZ-400*) positionné dans un *SLAM*. Leur approche n'exige pas non plus de mesures à la main des points de contrôle ni de zone de test conçue à cet effet. Au contraire, ils s'appuient sur une métrique de qualité calculée à partir de données laser, et sur la méthode de *Powell* comme l'algorithme d'optimisation. Les points du nuage sont modélisés par une fonction de répartition de probabilité $d(l)$ représentant la probabilité que l'emplacement ait été déjà mesuré. Les erreurs d'étalonnage entraînent donc des surfaces apparaissant à plusieurs positions. Étant la mesure de qualité, l'entropie de la probabilité $d(l)$, augmente avec ces erreurs et diminue considérablement lorsque le nuage est compact.

3.4 Impact d'un faux étalonnage sur la géométrie du nuage

Étant donné qu'un faux étalonnage extrinsèque introduit certaines déformations et que leur impact est observable (Figure 3.6), il paraît intéressant de déduire, par l'analyse de la structure des relevés laser, de vrais paramètres d'étalonnage. Nous examinons alors la pertinence des lignes à étalonner le système intégral basé sur des scanners, qu'ils soient 2D et 3D.

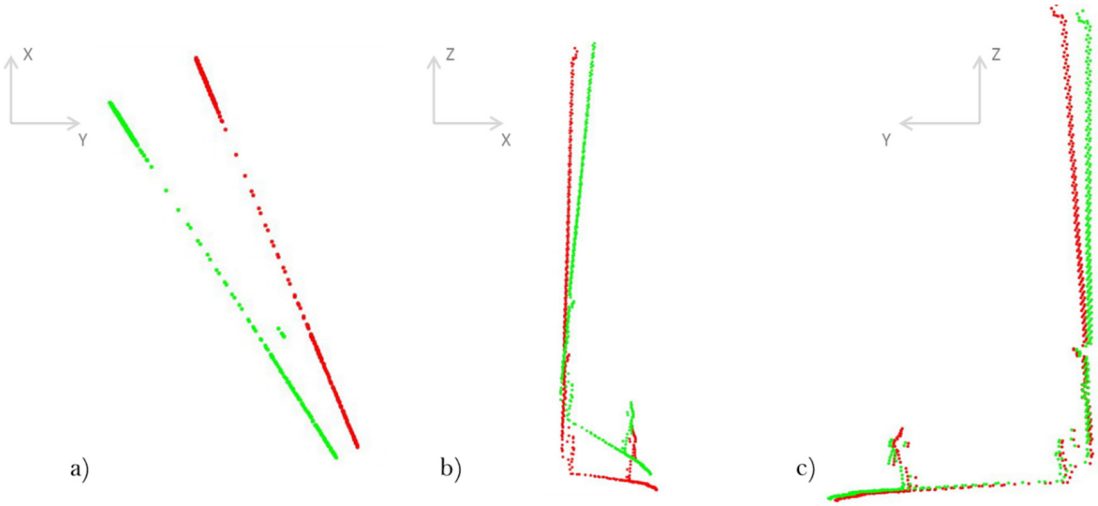

Figure 3.6 Impact de chacune des excentricités angulaires sur un profil laser : a) lacet e_z ; b) tangage e_y ; c) roulis e_x (en vert - nuage généré pour l'un des angles faux ; en rouge - nuage correct)

Pour comprendre l'impact des excentricités angulaires R_b^s entre le capteur laser et la centrale inertielle, nous régénérons des nuages acquis auparavant par notre prototype L3D2 équipé de deux scanners *SICK LMS 221*. Nous introduisons de faux paramètres d'étalonnage extrinsèque. La simulation est faite à l'aide du logiciel *RTMaps*. L'un de ses avantages

principaux est certes la possibilité d'enregistrer des données datées et synchronisées de manière à pouvoir effectuer un rejeu (*replay*) de ces dernières pendant des tests « hors lignes ». En effet, cinq nuages de points sont obtenus après avoir modifié des angles eulériens de la *matrice de boresight* selon les valeurs du Tableau 3.3. Les relevés laser ainsi que les segments extraits (à la main sous *MicroStation v8i SSE2* pour éviter des erreurs possibles), au sein d'un petit fragment de la scène balayée, sont visualisés par la Figure 3.7. Nous remarquons que les excentricités angulaires erronées ont une grande influence sur la géométrie du nuage, et donc sur la qualité. De plus, si un seul angle *boresight* est incorrect, l'écart de cet angle est aisé à observer. Une analyse plus approfondie nous permet de retrouver ces valeurs numériques, en examinant des angles entre les arêtes correspondantes (Figure 3.8).

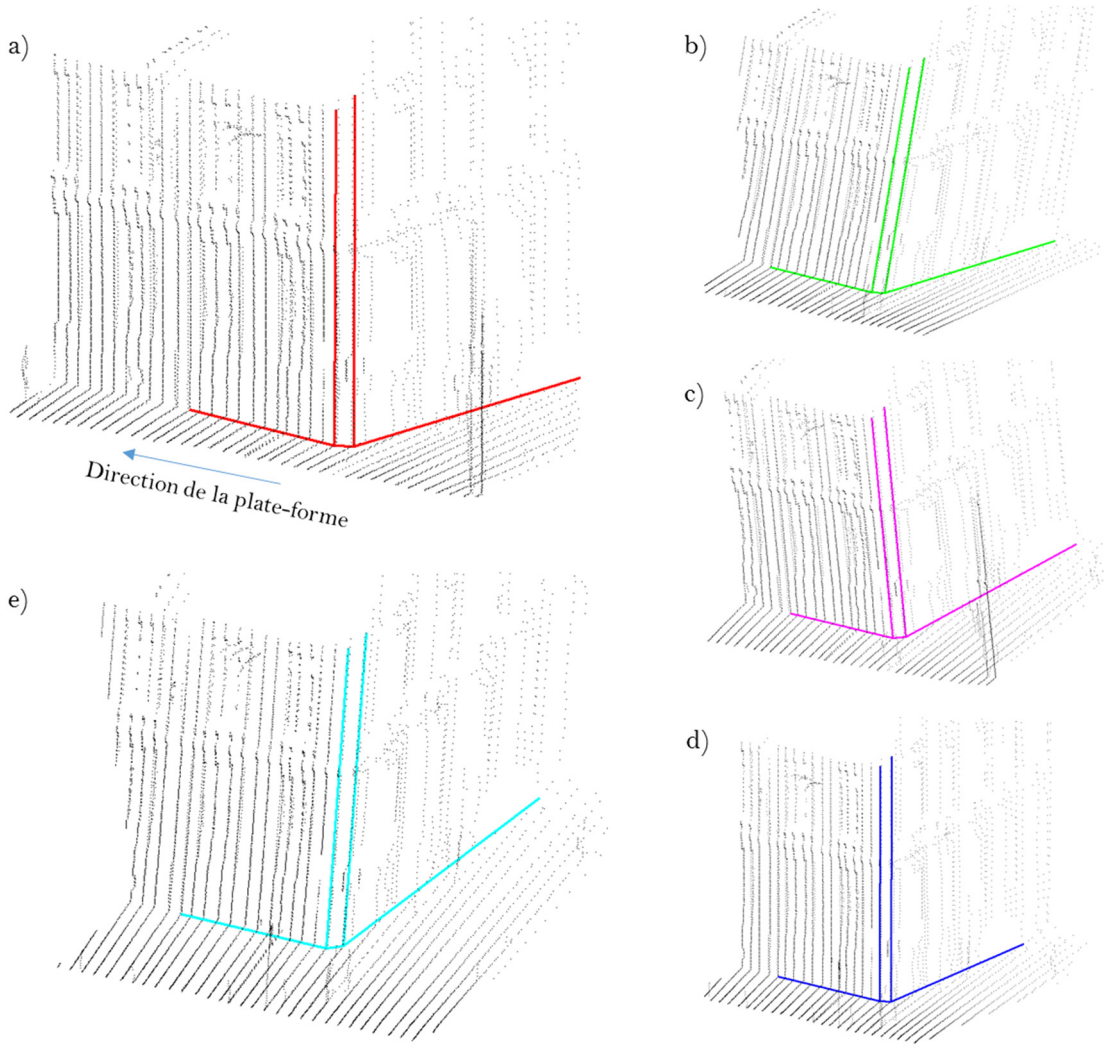

Figure 3.7 Impact de faux étalonnage extrinsèque sur le nuage : a) Nuage correct ; b)c)d)e) Nuages respectivement N1,N2,N3,N4 obtenus en prenant en compte les valeurs du Tableau 3.3

Tableau 3.3 Angles d'Euler formant la *matrice de boresight*, prise en compte pour générer des nuages de points

Nuage	e_z (lacet) [°]	e_y (tangage) [°]	e_x (roulis) [°]
Réf*	-90.0000	0.0000	0.0000
N1	-90.0000	0.0000	10.0000
N2	-90.0000	10.0000	0.0000
N3	-80.0000	0.0000	0.0000
N4	-80.0000	10.0000	10.0000

*Nuage correct

En se basant sur les études effectuées, nous concluons que :

1) L'écart de lacet est visible sur les arêtes horizontales et à la fois perpendiculaires à la direction de déplacement du système ;
2) L'écart de tangage est notable non seulement sur les segments verticaux, mais aussi sur ceux horizontaux et perpendiculaires à la direction de la plate-forme ;
3) L'écart de roulis est identifiable dans les lignes verticales.

En revanche, si tous les angles sont biaisés, il faut trouver une façon de quantifier l'écart de chacun des angles (e_x, e_y, e_z).

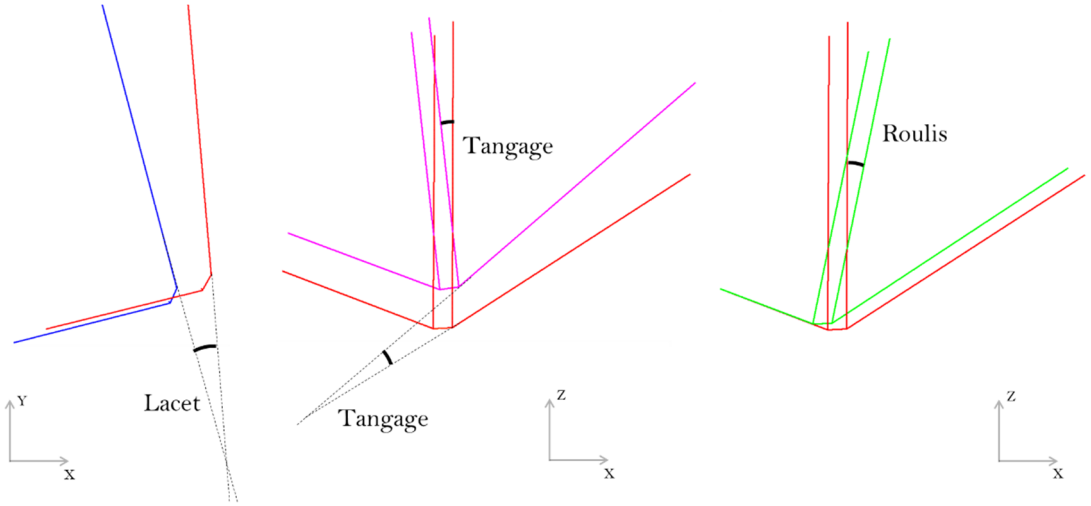

Figure 3.8 Étude de l'impact des écarts d'excentricités angulaires sur la géométrie des lignes

En conclusion, l'utilisation d'arêtes extraites dans les nuages de points devrait permettre l'étalonnage du système.

CHAPITRE 4

Extraction d'arêtes

Sommaire :

4.1 Méthode générale - vue d'ensemble 58
4.2 État de l'art sur l'extraction d'arêtes 59
4.3 Segmentation en éléments plans - état de l'art 61
4.3.1 Segmentation par croissance de surface (régions) 62
4.3.2 Transformée de *Hough* 3D 63
4.3.3 Algorithme de *RANSAC* 65
4.4 Contribution à la détection de plans 67
4.4.1 Motivation 68
4.4.2 Extraction séquentielle de plans 69
4.4.3 Connectivité de plans basée sur la théorie des graphes 71
4.5 Méthode proposée d'extraction d'arêtes 77
4.6 Exemples de résultats 82
4.6.1 Détection de plans 82
4.6.2 Extraction d'arêtes 84

Résumé

Dans ce chapitre un état de l'art des algorithmes récents d'extraction d'arêtes et de détection de plans dans le nuage de points est présenté. Pour satisfaire à nos exigences, à savoir, celles de recueillir des éléments linéaires (arêtes de bâtiments) pouvant remplacer les points de contrôle employés ordinairement, une méthode de détection des plis est proposée. Son idée repose sur l'intersection des plans principaux extraits dans le nuage. La recherche des plans sécants consiste avant tout à analyser l'angle d'intersection, ainsi que la distance réciproque entre toutes les combinaisons possibles de segments plans. Enfin, seules les entités linéaires dont la longueur est supérieure à un seuil prédéfini sont retenues.

Concernant la segmentation en éléments plans, qui est une phase déterminante, nous utilisons l'estimateur de RANSAC pour détecter précisément les plans principaux de la scène. Mais, malgré tous ses avantages, l'algorithme de RANSAC seul peut parfois donner des résultats erronés. Plus précisément, il cherche « le meilleur plan » sans tenir compte de la particularité de l'objet. Par

conséquent, les relations topologiques sont omises et les segments plans détectés ne coïncident pas toujours avec la scène réelle du point de vue sémantique. C'est pourquoi nous avons renforcé l'algorithme de RANSAC par une analyse des composantes connexes. Pour résoudre le problème de connectivité de plan, une approche innovante basée sur la théorie des graphes est proposée. Nous suggérons de systématiser, au sein du nuage, les relations entre les points en tenant compte de l'information sur leur voisinage local. Le nuage est désormais considéré comme un graphe non orienté défini par un ensemble de sommets V et d'arêtes E. Ensuite, la recherche des composantes connexes se résume à déterminer tous ses sous-graphes connexes. Nous nous référons, au lieu d'employer l'un des algorithmes récursifs d'exploration du graphe, à une décomposition de Dulmage-Mendelsohn. Enfin, des résultats d'extraction d'arêtes sur quelques jeux de données sont illustrés.

4.1 Méthode générale - vue d'ensemble

En vision par ordinateur et tout particulièrement en traitement d'image, la détection de différentes caractéristiques y compris des contours demeure un champ de recherche souvent abordé. La représentation d'une image par des contours peut significativement réduire la quantité des données et permet d'éliminer les informations jugées moins pertinentes, tout en préservant les propriétés structurelles de l'objet. Il existe un grand nombre d'algorithmes de détection, applicables à des images. Cependant, lorsqu' il s'agit des nuages de points 3D, les travaux relatifs à ce sujet ne sont pas nombreux, même si l'extraction des arêtes reste une étape primordiale dans la modélisation tridimensionnelle.

Concernant l'extraction des arêtes, nous proposons une méthode de détection de plis, conçue afin de diminuer l'influence de la densité de nuages de points sur le résultat final. Les intersections réalisées entre les plans principaux, à savoir, entre les façades elles-mêmes, mais aussi entre les façades et le sol, devraient répondre à nos besoins – fournir des entités linéaires précises, servant, à leur tour, à la qualification des données laser, au recalage ou bien encore à l'étalonnage. Du fait du manque, à notre connaissance, de description exhaustive de ce genre d'algorithmes, nous expliquons, dans ce chapitre, toutes les démarches effectuées, permettant d'extraire des droites d'intersection à partir de nuage de points.

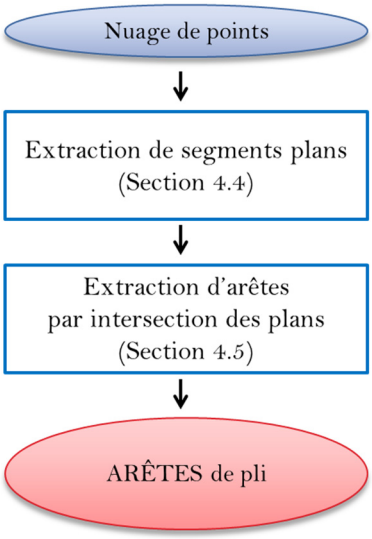

Figure 9.1 Organigramme de l'approche d'extraction d'arêtes proposée

L'approche proposée comporte deux étapes cruciales, comme illustré par la Figure 9.1. Dans un premier temps, le nuage est segmenté en éléments plans avec l'algorithme de *RANSAC*. Nous procédons ensuite à une analyse des composantes connexes pour vérifier, de façon robuste et précise, la connectivité et la sémantique des points formant chaque segment plan.

Nous obtenons alors un ensemble de segments plans $\prod$ accompagnés par les paramètres du plan ajustant au mieux ces points. Nous entendons par le terme « segment plan » un ensemble de points coplanaires à une tolérance près, et avec des limites bien définies. Autrement dit, c'est un plan limité possédant une certaine épaisseur. Nous étudions ensuite les relations de voisinage entre tous les segments plans détectés. Leur analyse va nous amener à la reconstruction des arêtes ayant une longueur supérieure à une tolérance admise T_{seg}.

Nos travaux se concentrent, dans un premier temps, sur la détection de plans indispensables dans le processus d'extraction d'arêtes. Ce chapitre est structuré de la manière suivante. Nous commençons par présenter l'état de l'art traitant de l'extraction d'arêtes (4.2) et de la reconnaissance de plans (4.3) dans les nuages de points. Nous caractériserons quelques méthodes, les plus souvent utilisées pour la segmentation des nuages 3D. En second lieu (4.4), nous détaillerons notre approche de segmentation basée sur l'algorithme de *RANSAC*. Comme la réalité n'est pas toujours compatible avec le modèle mathématique, nous proposons d'intégrer une contrainte de connectivité entre les points constituant chaque sous-nuage (segment plan) extrait. Cette problématique étant déjà connue, nous développons une solution alternative ne nécessitant plus de conversion d'un nuage 3D en une image 2D. Ainsi, la perte possible d'information lors de la transformation du nuage, causée habituellement par le nombre impropre de pixels par unité de longueur (résolution pix/m), ne se produira jamais. De cette façon, les erreurs apparaissant quand différents objets architecturaux de la scène se trouvent sur le même plan mathématique peuvent être corrigées et les plans détectés correspondront aux façades réelles. Ensuite (4.5), nous décrivons l'étape de recherche des intersections entre les segments plans. Par la suite (4.6), nous montrerons l'efficacité de notre algorithme en illustrant les résultats obtenus sur quelques nuages de points mobiles. Nous finirons par une brève discussion concernant les points forts et faibles d'une telle approche.

4.2 État de l'art sur l'extraction d'arêtes

La majorité des méthodes visant à indiquer les zones contenant « des points candidats », c'est-à-dire les points situés à l'intersection de deux surfaces, consiste à calculer la normale au point. Alternativement, il est possible de construire un maillage pour ensuite étudier la normale de chaque triangle. Dès qu'un changement important dans la direction des normales s'opère, l'existence d'une arête est perçue [Alshawa, 2006], [Demarsin, 2007].

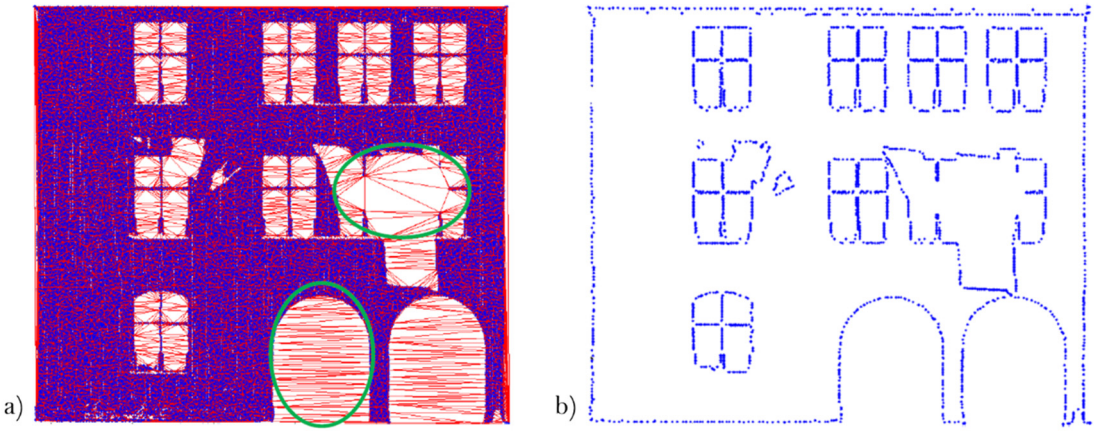

Figure 9.2 Exemple de résultats intermédiaires d'extraction d'arêtes :a) façade triangulée ; b) points de contours [Boulaassal, 2010]

Les travaux de [Boulaassal, 2010] visent à extraire des points de contour représentant les deux types d'arêtes (Figure 9.2b). Dans son approche, quatre étapes essentielles se distinguent : 1) la détection des plans ; 2) la triangulation de *Delaunay* ; 3) l'analyse de la longueur des côtés des triangles ; 4) la vectorisation des contours. Les sommets, reliant les côtés longs des triangles (marqués par un cercle vert dans la Figure 9.2a) dont la longueur change brusquement, et est à la fois supérieure à un seuil fixé, seront considérés comme des points de contour.

L'algorithme de [Wang *et al.*, 2011] permet de détecter automatiquement, dans le nuage de point mobile, des ouvertures sous forme de fenêtre et de porte. Il s'appuie, lui aussi, sur la détection préalable des façades de bâtiments. Ensuite, le voisinage local de chaque point du nuage est examiné suivant deux directions : verticale et horizontale. Pour faciliter cette tâche, un paramètre connu a priori, notamment celui de l'écartement entre deux fenêtres voisines, est pris en compte.

Dans le même contexte [Tuttas et Stilla, 2011] détectent des fenêtres dans les nuages acquis par un système mobile aéroporté *ALS*. Ces données d'entrée se caractérisent par une faible densité (seulement 5 points par m²). Leur approche est basée sur deux hypothèses :

1) Le faisceau laser passe toujours à travers les vitres des fenêtres. C'est pourquoi la présence des fenêtres peut être assimilée à des points se trouvant derrière la façade principale ;
2) Toutes les fenêtres sont disposées de manière régulière.

Afin d'extraire des points notablement éloignés et situés derrière la façade, les distances entre tous les points du nuage et le plan approximant sont calculées. La dernière phase consistera à faire une corrélation entre deux images : celle créée à partir des points détectés lors de l'étape précédente et un marqueur (motif linéaire) horizontal ou vertical.

Cependant [Soria-Medina *et al.*, 2011] cherchent à extraire des éléments d'une façade en s'appuyant non seulement sur l'information géométrique, mais aussi sur l'intensité. Les démarches proposées font référence à des outils dédiés au traitement d'image 2D. Ils proposent alors une solution intégrant l'algorithme des K-moyennes (*K-means*) et les opérateurs de la morphologie mathématique afin d'extraire des contours.

Un autre point de vue est envisagé par [Bienert, 2008]. Une méthode de détection des arêtes dans un nuage de points dense est proposée. Son idée repose sur le lissage du nuage avec un élément structurant. En effet, la position de chaque point est recalculée à partir de son voisinage. Ensuite, tous les points englobés par cet élément structurant participeront à l'ajustement d'une droite.

Nous venons de voir que la plupart des algorithmes d'extraction de segments, dédiés au nuage de points, permettent de détecter des points de contour appartenant potentiellement à l'arête. Néanmoins, il faut souligner qu'une telle entité linéaire peut être loin de la représentation de l'arête réelle. Les points du nuage seront d'autant plus proches de l'arête que la résolution du balayage effectué est petite. Mais, les tests des télémètres laser démontrent qu'il est très difficile, voire impossible, de mesurer précisément des bordures d'un objet, en raison de la diffraction subie par le faisceau laser à cet endroit [Kilmkowska et Wrobel, 2006]. L'une des pistes d'amélioration du processus d'extraction s'oriente vers la fusion des informations liées aux points 3D avec celles des images. L'utilisation conjointe de ces deux types de données est examinée entre autres par [Yong et Huayi, 2008], [Wang *et al.*, 2013] et [Wang *et al.*, 2014]. Les résultats obtenus mettent en évidence les avantages de cette solution.

4.3 Segmentation en éléments plans - état de l'art

Le traitement des relevés laser en vue de leur reconstruction fait l'objet de très nombreux travaux. La reconnaissance de différentes surfaces est considérée comme un problème de segmentation. Son objet est de séparer des formes de l'environnement en fonction de leur homogénéité, étant vérifiée au regard de la position des points, de la courbure, de la planéité ou de la direction de normale. Elle constitue également l'une des premières étapes de la modélisation tridimensionnelle puisque son but est de faciliter l'interprétation ultérieure des données.

Les méthodes existantes de la segmentation des nuages 3D s'inspirent, en général, des approches largement connues en vision par ordinateur. Le problème se réduit souvent à l'analyse de l'image de profondeur obtenue par la projection de nuage de points sur un plan image, où la valeur du pixel correspond à la distance du point au plan [Hernandez et Marcotegui, 2008], [Serna et Marcotegui, 2013]. D'après [Boulaassal, 2010], les algorithmes de segmentation dédiés aux relevés laser peuvent être regroupés en deux catégories. La première famille, bénéficiant des relations de proximités et de similitude, comprend des approches fondées sur la fusion notamment la segmentation par croissance de surface, la segmentation par division-fusion (*split-and-merge*), ou bien encore la segmentation par profils de balayage. Le second groupe adapte les techniques de reconnaissance automatique de formes. Celles-ci étant plus robustes, elles ne peuvent être employées qu'afin de subdiviser le nuage en primitives géométriques définies par l'ensemble de paramètres. Il s'agit des modèles 3D a priori tels que la sphère, le plan, le cône, ou le tore. Les techniques les plus utilisées font appel à la transformée de *Hough* [Hough, 1962] ou à l'algorithme de *RANSAC* (*RANdom SAmple Consensus*) de [Fischler et Bolles, 1981]. Les deux approches s'avèrent particulièrement robustes au bruit, entachant les nuages de points.

La détection de plans ou l'approximation par un plan s'avère un problème spécifique de la segmentation de données 3D permettant de séparer une scène complexe en deux parties : les zones planes et non planes. Durant ces dernières années, plusieurs méthodes d'extraction de plans ont été proposées. Un aperçu de quelques techniques est donné par [Vosselman *et al.*, 2004].

Les travaux de [Bauer, 2005] et [Boulaassal, 2010] s'appuient sur l'utilisation de l'algorithme de *RANSAC* dans le but de détecter les façades de bâtiments à partir du nuage de points dense. Cependant, plusieurs améliorations de cet estimateur ont été apportées afin de l'optimiser et de le rendre encore plus efficace et fiable. [Delmerico *et al.*, 2011] et [Awwad *et al.*, 2011] proposent de prendre en compte des normales en chaque point. De même [Bretar et Roux, 2005] détectent les plans de toits dans les nuages de points aériens à l'aide de l'algorithme *ND-RANSAC* (*Normal Driven RANSAC*). À cet effet, ils calculent les normales au point. Puis, le tirage aléatoire d'un échantillon minimal composé de trois points ayant les vecteurs normaux d'orientation identique est effectué. L'approche de [Yang *et al.*, 2010] consiste à diviser un nuage de points en petits blocs. Ensuite, une méthode combinant l'algorithme de *RANSAC* avec la longueur de description minimale *MDL* (*Minimum Description Length*) est exécutée au sein de chaque sous-nuage de points. La tâche de ce dernier est de vérifier le nombre de plans retrouvés, étant entendu que ce nombre varie, d'un bloc à l'autre, de zéro à trois plans. En même temps [Gallo *et al.*, 2010] développent un algorithme appelé *CC-RANSAC* dont l'idée repose sur la recherche de la composante connexe la plus grande dans l'image de profondeur.

De leur côté [Tasha-Kurdi *et al.*, 2007], [Borrmann *et al.*, 2011] et [Borowiec, 2012] emploient la transformée de *Hough* 3D comme un outil de détection des plans.

Les méthodes de [Deschaud, 2010], [Jarzabek-Rychard et Borkowski, 2010] et [Tuttas et Stilla, 2010] sont fondées sur un algorithme de croissance de surface. L'idée consiste toujours à identifier tout d'abord des régions graines et de les faire croître selon certaines

conditions comme nous l'expliquerons dans le paragraphe 4.3.1. Son avantage principal est d'être rapide lorsqu'il y a de nombreux petits plans à extraire. Sinon, lors de détection des plans grands (par exemple les plans de taille d'une façade) l'algorithme tend à être plus en plus lent [Deschaud, 2010].

Parmi les travaux portant sur la détection des toits de bâtiment dans les données laser aéroportées, nous pouvons remarquer un certain nombre d'études comparatives. [Jarzabek-Rychard et Borkowski, 2010] analysent les résultats obtenus avec la croissance de surface et l'algorithme de *RANSAC*. Ils en concluent que ce dernier est plus efficace pour la segmentation de modèles peu compliqués. En revanche, il est plus susceptible d'unir des points situés sur le même plan mathématique, mais appartenant, en réalité, à différentes structures architecturales. Dans ce même contexte [Tarsha-Kurdi, 2008] améliore l'algorithme de *RANSAC* pour le comparer, par la suite, avec la transformée de *Hough*. Il constate que l'algorithme de *RANSAC* est plus adapté aux données 3D, tant en termes de traitement que d'insensibilité au bruit.

Pour tirer profit des deux méthodes de segmentation, [Tuttas et Stilla, 2013] proposent une approche de détection de façades en deux étapes : le traitement grossier et fin. Dans un premier temps, la segmentation par croissance de surface basée sur deux critères (seuil de normale et de distance) est effectuée. Ensuite, chaque plan est analysé afin de vérifier s'il doit être subdivisé en plusieurs segments plans. Le plan final est retrouvé avec l'algorithme de *RANSAC*. Pareillement [Oehler *et al.*, 2011], emploient la transformée de *Hough* combiné avec l'algorithme de *RANSAC*. D'abord, les normales sont calculées et la pré-segmentation est réalisée avec la transformée de *Hough*. Puis, l'algorithme de *RANSAC* est directement employé sur les segments plans détectés auparavant.

Finalement [Borowiec, 2008], s'appuie, afin de détecter de toits, sur une approche basée sur l'algorithme de division-fusion.

L'étude des travaux actuels révèle que la majorité des approches optent pour la détection de plans dans le nuage de points à l'aide de l'algorithme de *RANSAC*, de la transformée de *Hough*, ou éventuellement de la croissance de surface. C'est pourquoi nous détaillerons dans cette section les principes de fonctionnement propres aux techniques susmentionnées.

4.3.1 Segmentation par croissance de surface (régions)

La segmentation par croissance de surface est conceptuellement une adaptation, dans l'espace $\mathbb{R}^3$, d'une méthode de croissance de régions. Son objectif est d'isoler des zones les plus homogènes possibles, tout en respectant des conditions arbitraires imposées. Nous distinguons deux étapes principales de l'algorithme :

1) Trouver les points de départ – régions graines (*seed*) ;
2) Rassembler des points/pixels voisins entre eux suivant les critères prédéfinis

Or, cette méthode consiste à faire progressivement grossir les régions autour de leurs points de départ. Les zones grandissent alors par incorporation des points les plus similaires. L'agglomération des points n'exploite aucune connaissance a priori du nuage ou du bruit qui le dégrade. La décision d'intégrer à la région un point voisin repose sur un seul voire deux critères tels que la proximité des points (seulement les points proches peuvent être ajoutés), la planéité locale (la distance orthogonale du point au plan doit être inférieure à un seuil fixé), le champ de vecteurs normaux (les angles entre la normale du plan et les normales de tous les points voisins appartenant au plan doivent être inférieurs à un seuil choisi) [Vosselman *et al.*, 2004]. Par conséquent, les variantes se distinguent surtout par la manière de sélectionner des plans initiaux et par les critères de croissance définis. Le choix des régions graines est la phase la plus critique de l'algorithme puisqu'elle joue sur la qualité des résultats obtenus et sur le temps de calcul.

4.3.2 Transformée de *Hough* 3D

La transformée de *Hough* [Hough, 1962] est un outil de reconnaissance de formes basé sur une connaissance a priori. Cette méthode fait partie de la famille des algorithmes de votes. Elle est souvent utilisée dans la détection de lignes 2D (en analyse d'images). Mais, l'extension à d'autres primitives géométriques représentées par une fonction mathématique simple (par exemple plan, cercle, ellipse) ou à des espaces de dimension quelconque est aussi possible.

Cas 2D - ligne

Afin d'expliquer le principe de la transformée de *Hough*, nous commençons par un exemple typique – une ligne 2D.

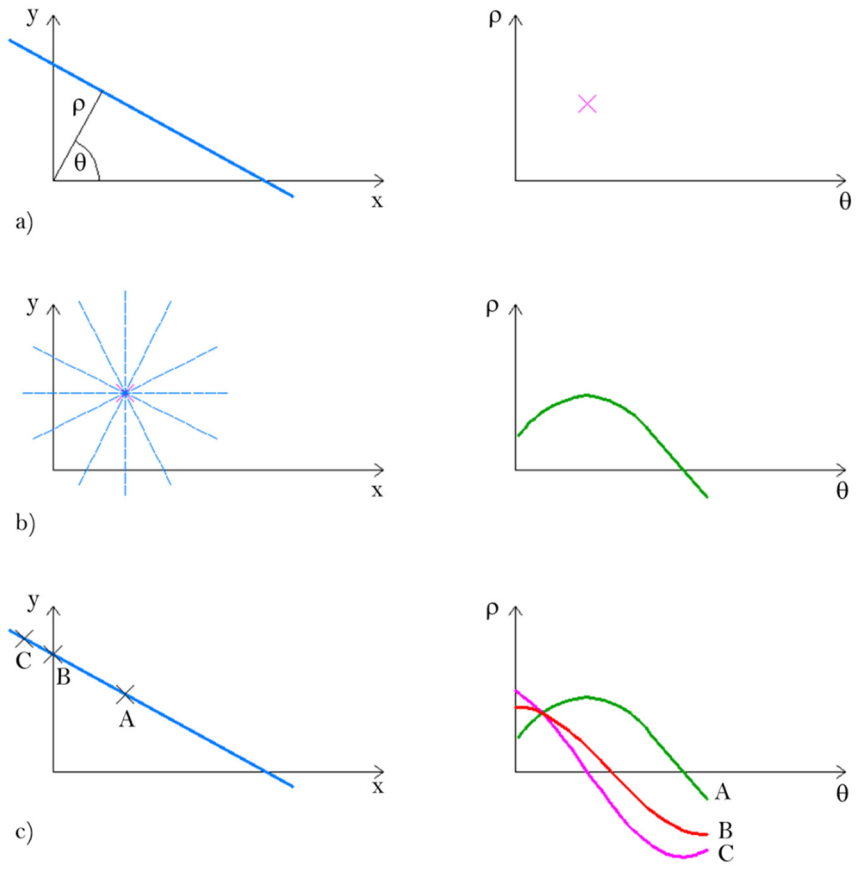

Figure 9.3 Correspondance géométrique entre l'espace cartésien (à gauche) et l'espace paramétrique de *Hough* (à droite) (Illustration reproduite à partir de [Poncelet et Cornet, 2010])

Étant donné une droite notée $y = mx + b$ (où m : coefficient directeur et b : ordonnée à l'origine) dans l'espace *Euclidien*, il nous faut, conformément à l'idée de *Hough*, changer l'espace de représentation de cette droite et l'exprimer désormais par $b = -mx + y$. Nous obtenons l'équation d'une droite passant par un point (m, p) dans l'espace de paramètres (l'espace de *Hough*). Cette forme la plus classique, servant à représenter la droite, n'est cependant pas sans problème, car les droites verticales et horizontales auront des valeurs infinies pour m et b. Pour pallier cet inconvénient, il est donc préférable d'exploiter les coordonnées polaires (l'équation normale) d'une droite au lieu de sa formulation cartésienne (9.1) [Duda et Hart, 1971].

$$\rho = x\cos\theta + y\sin\theta \tag{9.1}$$

où :

ρ : distance de la ligne à l'origine

θ : angle (l'orientation du vecteur normal de la droite depuis l'origine)

Cas 3D - plan

Le même principe s'applique à la détection de plans. Chaque plan de l'espace Euclidien est défini par la formulation normale donnée par :

$$\rho = x\cos\theta\sin\varphi + y\sin\theta\cos\varphi + z\sin\varphi \tag{9.2}$$

La Figure 9.4 explique la notation employée dans l'équation (9.2).

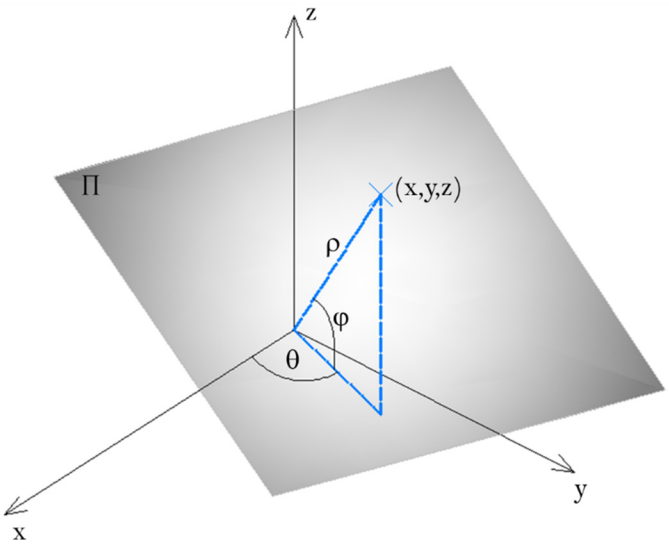

Figure 9.4 Formulation normale d'une droite

De manière analogue, les trois paramètres (θ, φ, ρ) définissent l'espace de paramètres. Chaque point (θ, φ, ρ) de cet espace correspondra à un plan de l'espace cartésien et vice versa. Étant donné un point quelconque du plan $\prod$, tous les plans auxquels ce point appartient doivent être trouvés.

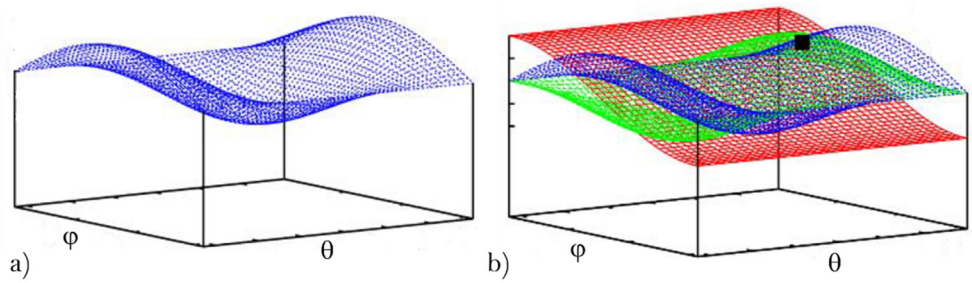

Figure 9.5 Espace de paramètres : a) Sinusoïde 3D calculée à partir d'un point de l'espace cartésien ; b) Transformée de *Hough* de trois points et le point d'intersection (carré noir) [Borrmann *et al.*, 2011]

En traçant ces points dans l'espace de *Hough*, une courbe sous forme d'une sinusoïde 3D est obtenue (Figure 9.5a). Par conséquent, un plan passant par N points est, dans l'espace de paramètres, le point d'intersection de N sinusoïdes (Figure 9.5b). Finalement, le pic de l'accumulateur correspond à un maximum de points dans un même alignement. Plus des courbes se coupent en un même point de l'espace de paramètres, plus des points appartiennent au plan détecté. La transformée de *Hough* cherche à trouver le meilleur plan qui est perçu comme celui contenant un nombre maximal de points. De ce fait, le plan optimal n'est pas calculé au sens des moindres carrés.

4.3.3 Algorithme de *RANSAC*

L'algorithme de *RANSAC* [Fischler et Bolles, 1981] est très largement utilisé dans le domaine de la vision par ordinateur pour l'estimation des paramètres. Son avantage est d'être robuste et rapide, même en présence d'observations aberrantes. L'une de ses applications est la détection, de manière interactive, de plans. [Zuliani, 2012] rapporte que cet estimateur résiste très bien à la présence de données non valables (*outliers*) en supportant jusqu'à 50% d'erreurs grossières. Cette marge rend, de ce fait, la méthode particulièrement intéressante pour traiter les nuages de points contenant pourtant des mesures bruitées.

Son principe est simple : il s'agit toujours de sélectionner au hasard un nombre minimum de données afin de calculer les paramètres d'un modèle mathématique, puis d'identifier parmi toutes les données, celles ayant une chance raisonnable d'appartenir à ce modèle. Le processus étant itératif, il s'arrête lorsque l'on trouve un sous-ensemble de points le plus nombreux.

Deux phases essentielles de l'algorithme de *RANSAC* peuvent être distinguées. La première nommée « *Initialisation* » consiste à tirer aléatoirement un nombre minimum de points (un quorum) S parmi N points de l'ensemble. Un plan est défini par quatre paramètres, dont trois $(\theta_1, \theta_2, \theta_3)$ étant indépendants, sont les composantes de la normale au plan et θ_4 est la distance du plan à l'origine. Alors, l'échantillon minimal contiendra trois points $S = \{p_1, p_2, p_3\}$ à partir desquels tous ces indices seront déterminés. Nous désignons par $M(\theta)$ le modèle mathématique du plan défini de la manière suivante :

$$M(\theta) = \{\theta_{1 \leq i \leq 4} \in \mathbb{R} : \theta_1 x + \theta_2 y + \theta_3 z + \theta_4 = 0\} \tag{9.3}$$

La seconde étape « *Test* », exécutée après le calcul du modèle, vérifie un score, perçu généralement comme le nombre de points ajustant au mieux le plan. Nous considérons qu'un point quelconque appartient au plan, si et seulement si, il est situé à une distance inférieure à une tolérance prédéfinie δ.

Nous introduisons les termes inhérents à l'utilisation de *RANSAC*. On appelle désormais *EC* « *Ensemble de Consensus* » (*Consensus Set*), l'ensemble de points satisfaisant au modèle prédéfini auparavant par un quorum de points. On note *Card(EC)* le nombre de points valables (*inliers*).

$$Card(EC) = \{p_i \in N : e_M(p_i, M(\theta)) \leq \delta\} \tag{9.4}$$

Dans la formule (9.4) l'écart entre le point p et le modèle $M(\theta)$ se calcule de la manière suivante :

$$e_M(p_i, M(\theta)) \stackrel{\text{def}}{=} dist(p_i, p_i^*) \tag{9.5}$$

où p_i^* symbolise la projection orthogonale du point p sur le modèle $M(\theta)$.

Les deux phases sont répétées N_{iter} fois. À ce moment, le plan est retenu s'il contient suffisamment de points, c'est-à-dire si le nombre de points *Card(EC)* appartenant au « meilleur modèle » est le plus grand (Figure 9.6).

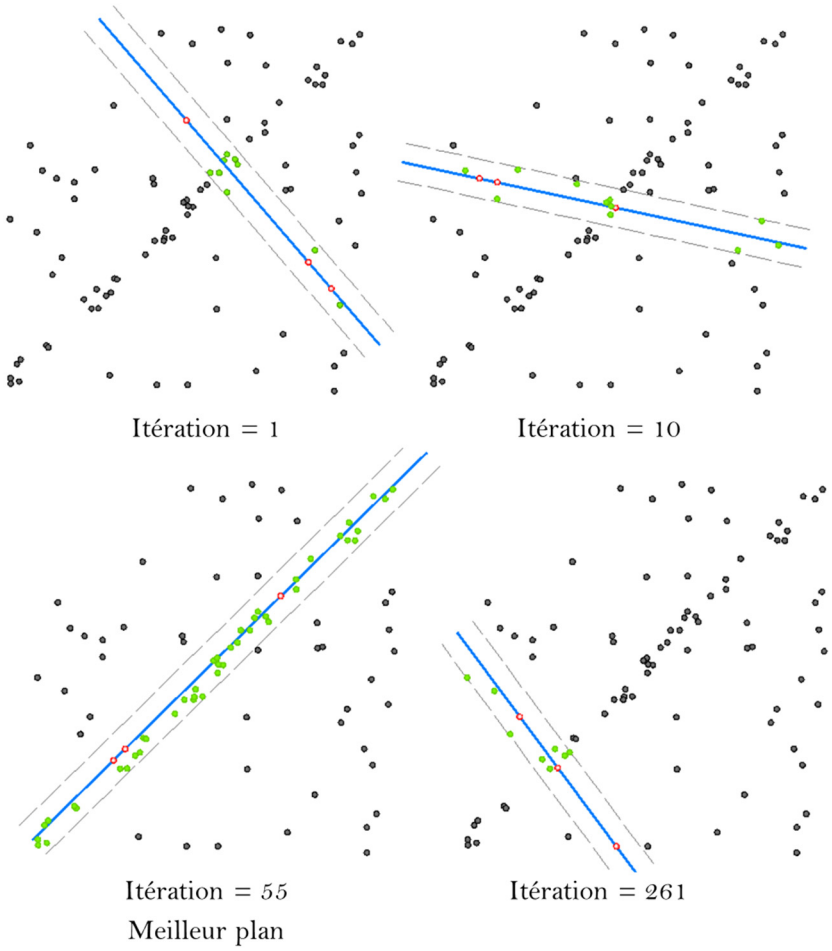

Figure 9.6 Vue de profil de la zone d'épaisseur δ contenant les points valables (vert) satisfaisants au modèle défini par un quorum de points (rouge)

L'estimation du nombre d'itérations qui permettrait d'interrompre l'algorithme reste une question délicate. L'algorithme de *RANSAC*, dans sa version de base, vise à garantir un nombre optimal de tirages indispensables pour estimer le meilleur plan avec une probabilité P. En admettant que W désigne la probabilité de choisir, à chaque itération, un point valable, la probabilité que les trois points du quorum appartiennent tous au plan final est de W^3. De ce fait, il y a une probabilité égale à $(1 - W^3)$ qu'au moins un de ces points ne fasse pas partie des points valables. Nous pouvons relier les probabilités P et W avec le nombre d'itérations N_{iter} par l'équation suivante :

$$(1 - P) = (1 - W^3)^{N_{iter}} \Leftrightarrow N_{iter} = \frac{log(1 - P)}{log(1 - W^3)} \tag{9.6}$$

La formule (9.6) démontre qu'afin de déterminer le nombre d'itérations, la connaissance de P et W est primordiale. Le choix de ces valeurs se fait de manière empirique. La probabilité P est choisie de préférence élevée ($P \geq 0.95$). Quant à la valeur W, elle doit être estimée en fonction des données. Cette probabilité, pour un seul point, est exprimée par :

$$W = \frac{Card(EC)}{Card(N)} \tag{9.7}$$

Il est difficile de choisir dûment la probabilité W car souvent nous ne connaissons pas le nombre de points valables restant dans le nuage de points. L'algorithme est ainsi initialisé en utilisant une estimation grossière de W, ce qui augmente le risque d'accepter un nombre inapproprié d'itérations.

Pour surmonter cet inconvénient, une version adaptative de *RANSAC* a été développée par [Hartley et Zisserman, 2003]. L'idée consiste à sélectionner dynamiquement le nombre de tirages à réaliser, en adaptant la valeur W grâce au cardinal de l'ensemble de consensus trouvé. En d'autres termes, le nombre maximal d'itérations sera mis à jour avant le démarrage de la prochaine sélection d'un quorum de points. L'avantage principal d'un tel raisonnement est d'être plus efficace et de réduire considérablement le temps de calcul.

4.4 Contribution à la détection de plans

Pour notre application, nous souhaitons pouvoir détecter de grands plans (plans principaux) tels que les façades des bâtiments, le sol, tout en étant suffisamment rapide. L'analyse de l'état de l'art, nous a encouragé de choisir l'estimateur de *RANSAC* comme l'algorithme de base puisqu'il paraît le plus robuste pour traiter de nombreux points. Nous implémentons sa version adaptative décrite dans la section précédente et caractérisée par [Zuliani, 2012]. La segmentation en segments plans s'effectue de manière séquentielle. Une fois un plan détecté, les points qui lui appartiennent sont retirés du nuage. L'algorithme s'interrompt lorsqu'il n'y a plus de données à classer, ou lorsque les points restants ne permettent pas d'obtenir de segment plan contenant un nombre suffisant de points, ou bien encore quand le nombre de plans désirés N_{pl} a été retrouvé.

Sachant que les segments plans obtenus à l'aide de l'algorithme de *RANSAC* ne peuvent pas toujours coïncider avec la scène réelle du point de vue sémantique, il nous faut aussi prendre en compte les relations topologiques entre les points formant un plan. Nous avons donc enrichi l'algorithme de *RANSAC* d'une analyse des composantes connexes. La problématique de connectivité a été largement étudiée en traitement d'images 2D. Elle vise à vérifier si deux pixels sont voisins et s'ils respectent, en même temps, une condition arbitraire par exemple l'égalité des niveaux de gris. En conséquence, nous pouvons nous en inspirer pour incorporer une contrainte de connectivité entre les points formant un segment plan. Dans cet esprit [Boulaassal, 2010] et [Gallo *et al.*, 2010], transforment un plan 3D détecté par l'algorithme de *RANSAC* en une image 2D. Le passage est réalisé par la projection des points du nuage sur le plan moyen. Cette étape nécessite de choisir soigneusement la résolution de l'image créée (le nombre de pixels par unité de longueur) afin d'éviter les problèmes d'échantillonnage. Des valeurs trop larges ou trop petites ne permettront pas d'étudier la connectivité de pixel. La séparation ultérieure des différentes régions consiste à calculer la différence entre les valeurs de pixels voisins.

Étant donné que la conversion en image peut être une source importante d'erreurs, nous avons proposé un algorithme fonctionnant directement sur des nuages de points. Il fait appel aux éléments de la théorie des graphes et attend seulement quelques paramètres initiaux pour s'exécuter. Ainsi, nous structurons les nuages, et effectuons une décomposition de *Dulmage–Mendelsohn* [Dulmage et Mendelsohn, 1958] aboutissant à décomposer de façon unique un graphe biparti non orienté en composantes connexes. Ce premier est défini grâce à l'analyse du voisinage de chaque point du nuage. Nous systématisons les points du segment plan, étant naturellement désordonnés, en les représentant par un ensemble de sommets (points) V et d'arêtes E établissant les relations topologiques entre ces sommets.

L'extraction de segments plans est menée en deux étapes comme l'illustre la Figure 9.7. Nous présenterons dans le paragraphe 4.4.2 l'algorithme de *RANSAC* tel qu'il est implémenté. Ensuite, nous discuterons des principes de notre méthode de détection de composantes connexes basée sur la théorie des graphes (paragraphe 4.4.3).

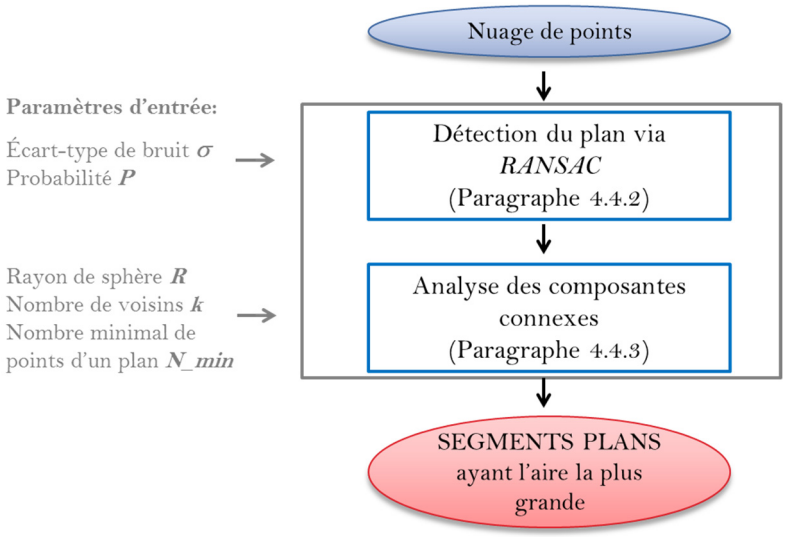

Figure 9.7 Schéma global de l'algorithme d'extraction de segments plans implémenté

4.4.1 Motivation

Un plan est une entité fondamentale à deux dimensions et nous l'imaginons comme une feuille d'épaisseur nulle qui s'étend à l'infini. Cependant, les plans extraits à partir du nuage de points n'ont pas forcément ces deux caractéristiques. D'une part, nous devons prendre en compte la troisième dimension – l'épaisseur provenant principalement du comportement du faisceau laser sur différentes surfaces. D'autre part, un plan quelconque approximant une façade, un trottoir ou une route a des limites bien définies. Mais, l'estimateur de *RANSAC* fournit un plan illimité, rassemblant tous les points qui appartiennent à un même plan mathématique.

Figure 9.8 Détection de trois plans successifs par *RANSAC*. Une couleur correspond à un plan

La Figure 9.8 montre trois plans successifs extraits à partir du nuage mobile *Stereopolis2009_Soufflot* avec cet algorithme. Les deux façades (marquées en orange), étant en réalité séparées, ont été classifiées comme une seule entité planaire. De plus, nous observons de nombreuses régions éparpillées autour des régions principales qui ne devraient pas être intégrées au modèle final.

Nous visons à supprimer ces petits plans pouvant fausser les résultats du traitement ultérieur, notamment l'extraction d'arêtes basée, par défaut, sur les plans. Étant donné que nous cherchons à conserver les plans les plus précis possibles et reconstruisant correctement la scène réelle, il nous semble primordial d'effectuer une analyse du nombre d'entités constituant le segment plan et d'en conserver une seule partie – celle ayant l'aire la plus grande. La Figure 9.9 illustre le plan détecté via l'algorithme de *RANSAC* et celui que l'on souhaiterait extraire. C'est pourquoi nous proposons une extraction innovante des composantes connexes dédiée au nuage. Il s'agit d'étudier le voisinage de points pour s'en servir lors de l'extraction des composantes connexes.

Figure 9.9 Exemple d'erreurs de connectivité : plan détecté (à gauche) ; plan recherché (à droite)

4.4.2 Extraction séquentielle de plans

L'inconvénient majeur de l'algorithme de *RANSAC* classique est la détection d'une forme unique. Pour y remédier, l'une des stratégies à envisager consiste à l'implémenter dans un mode séquentiel. L'idée de cette approche nous amène à retirer du nuage les points valables appartenant au segment plan détecté au préalable. Cette opération est de nouveau appliquée sur les points restants jusqu'à ce que le nombre prédéfini de plans N_{pl} ait été extrait ou jusqu'à ce qu'il n'y ait plus de points permettant une détection de surfaces désirées.

Du fait de la présence de bruit σ provenant de l'état d'objet balayé par scanner, sa couleur, nature et rugosité (voir le Chapitre 3), les nuages de points ont habituellement une épaisseur non négligeable. C'est pourquoi l'ajustement de plans s'effectue avec une certaine souplesse. La sélection du seuil δ – la zone tampon autour du plan moyen, joue un rôle important au niveau de la stabilité de l'algorithme de *RANSAC* puisqu'il peut provoquer des problèmes de sous-segmentation et de sur-segmentation (Figure 9.10). La valeur de ce seuil δ doit être fixée en tenant compte des caractéristiques du nuage comme sa densité et son degré de

bruitage. Ainsi, elle a tendance à être égale à l'épaisseur du nuage de points brut – sans filtrage préalable.

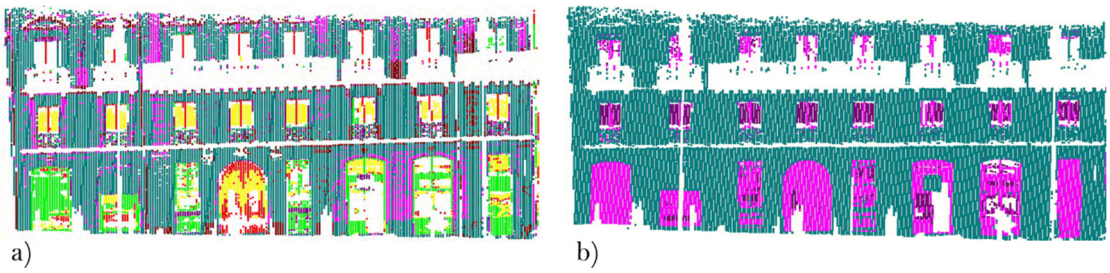

Figure 9.10 Impact du choix du seuil δ sur les résultats de détection des plans : a) Segmentation avec un seuil faible (0.03m) ; b) Segmentation avec un seuil élevé (0.15m)

D'après [Hartley et Zisserman, 2003] suivis par [Zuliani, 2012], le seuil δ peut se calculer en supposant que tous les points suivent une distribution normale autour du plan moyen. Nous faisons l'hypothèse que les nuages sont entachés par un bruit gaussien $\eta \sim N(0, \sigma I)$ pour $\eta = p - p^*$. La valeur δ relie désormais, avec une probabilité P, l'erreur générée par les points valables affectés par le bruit σ. L'épaisseur de la zone de tampon δ se calcule de la manière suivante :

$$\delta = \sigma_\eta \sqrt{F_{\chi_2^2}^{-1}(P)} \tag{9.8}$$

où :

P : probabilité considérée ($P \geq 0.95$)

$F_{\chi_2^2}^{-1}$: valeur de la fonction de répartition de la loi chi-2 (χ_m^2) à $\mathrm{m} = 2$ degrés de liberté (Tableau 9.1)

σ : degré de bruitage

Tableau 9.1 Table de distribution χ_2^2 selon la valeur de probabilité P

Degré de liberté	P=0.900	P=0.950	P=0.975	P=0.990	P=0.999
2	4.605	5.991	7.378	9.210	13.861

Si nous souhaitons récupérer 99% des points d'un plan, nous avons P=0.99, et donc $F_{\chi_2^2}^{-1}(0.99) \approx 9.21$. L'analyse du Tableau 9.1 permet de constater que plus la valeur de la probabilité P est proche de l'unité, plus le seuil δ est important et, que l'épaisseur de la zone tampon augmente. Le modèle risque, en réalité, de contenir beaucoup de points aberrants. En outre, une probabilité P trop petite génère un seuil insuffisant, et les points potentiellement valables sont susceptibles d'être exclus du modèle.

Dans notre approche, nous avons employé la formule (9.8), au lieu de définir la tolérance de distance maximale, pour tester l'appartenance des points au plan. Un point est considéré comme valable si l'écart au carré entre ce point et le plan est inférieur à δ^2. Plus précisément, le test statistique se fait de la manière suivante :

$$\begin{cases} Si \quad e_M\big(p, M(\theta)\big)^2 \leq \delta^2 & \textit{point valable situé dans la zone tampon} \\ Si \quad e_M\big(p, M(\theta)\big)^2 \geq \delta^2 & \textit{point aberrant (hors tolérance admise)} \end{cases} \tag{9.9}$$

Cette solution avec un degré de bruit σ associé au nuage est censée estimer une valeur optimale du seuil δ. Nous avons implémenté l'algorithme de *RANSAC* dans la version ci-décrite (voir aussi le pseudo-code 4.1 dans l'Annexe E). Pour la détection de plans, nous avons besoin en entrée d'un nuage de points N relativement homogène. Néanmoins, il faut noter que l'utilisation de cet estimateur implique un réglage de trois paramètres : P, σ et éventuellement un nombre de plans désiré N_{pl}. Le symbole P correspond à la probabilité d'avoir les points correspondants au plan, tandis que la valeur σ est l'écart-type des distances calculées par rapport au plan. Ces deux indices sont d'ores et déjà nécessaires pour déterminer la valeur du seuil δ. Elle est alors assimilée à la distance maximale point – modèle, et elle représente l'épaisseur du segment plan à extraire. En sortie, nous obtenons des sous-ensembles de points formant des plans $\prod$ et leurs paramètres associés $(\theta_1, \theta_2, \theta_3, \theta_4)$.

4.4.3 Connectivité de plans basée sur la théorie des graphes

Une fois le segment plan détecté, nous procédons à l'analyse des composantes connexes CC, c'est-à-dire de toutes les parties surfaciques constituant cette entité. Le terme connectivité dérive de la notion de voisinage. On dit que deux points de l'espace sont connectés s'ils appartiennent à la même composante. Quant au critère de décision permettant de distinguer les points voisins, il s'appuie sur le calcul de distance qui sera ensuite confronté avec un seuil prédéfini ρ (distance maximale). En effet, il s'agit de vérifier si un plan se constitue en un seul « morceau ». Nous entendons par « connectivité » du plan sa continuité spatiale – le segment plan doit constituer une entité contigüe et non pas un ensemble de surfaces séparées par des vides comme le montre la Figure 9.9. Sinon, chacun de ses éléments est l'une des composantes connexes.

La Figure 9.11 illustre notre raisonnement. D'abord, un segment plan est extrait via l'algorithme de *RANSAC* tel qu'il est présenté dans la section 4.4.2. Ensuite, nous étudions les relations de voisinage entre chaque point du sous-nuage. Dès que toutes les composantes connexes sont déterminées, leurs surfaces sont calculées et comparées les unes par rapport aux autres. Puis, la partie dont l'aire est la plus importante sera retenue, tandis que les points des autres régions seront réintégrés dans le nuage pour participer à la recherche ultérieure de meilleurs plans.

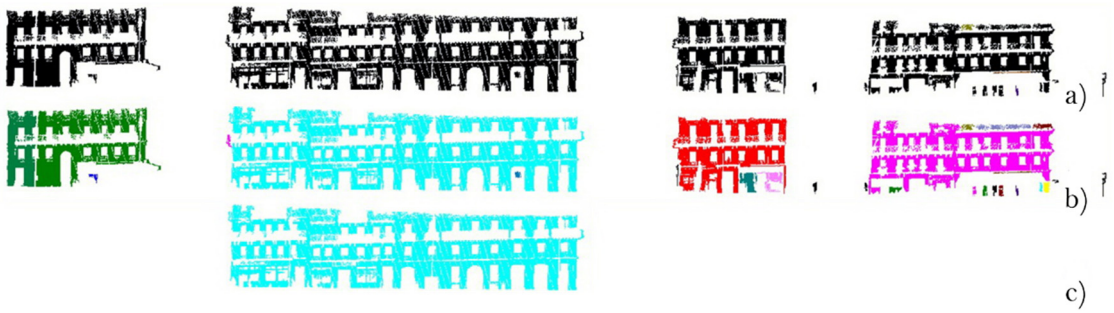

Figure 9.11 Correction d'extraction du plan : a) Premier plan détecté via l'algorithme de *RANSAC* ; b) Étiquetage des composantes connexes ; c) Segment plan retenu après intégration du critère de connectivité

Afin de résoudre le problème de connectivité, le nuage de points N est traité comme un graphe non orienté pour lequel tous ses sous-graphes G' seront recherchés. À ce stade, il nous faut rappeler que nous travaillons directement sur les données laser 3D. Nous constatons que les points appartiennent à la même composante connexe, si est seulement si, il existe au moins un chemin reliant n'importe quel couple de points de cette composante.

Nous distinguons deux phases dans notre approche de détection de composantes connexes d'un segment plan (Figure 9.12) : 1) la recherche du voisinage de chaque point du nuage en respectant les seuils renseignés avec la construction d'une liste de paires de points (sommets) de telle sorte que chaque paire correspondra à une arête au sens de la théorie des graphes ; 2) l'étiquetage des composantes connexes.

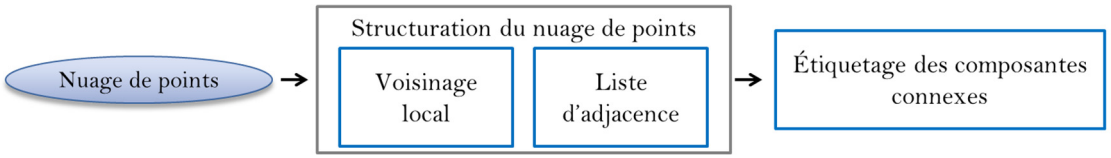

Figure 9.12 Organigramme de l'algorithme d'extraction de composantes connexes

Structuration du nuage de points

Concernant la théorie des graphes, un graphe simple non orienté $G = (V, E)$ est défini par la donnée d'un ensemble de sommets V et d'un ensemble d'arêtes $E \subseteq \{(u, v): u, v \in V, u \neq v\}$. En d'autres termes, un graphe est constitué de sommets V, dont certains sont reliés par des arêtes, si est seulement si, ces sommets sont adjacents les uns par rapport aux autres (Figure 9.13ab). Si un graphe est non orienté, dans ce cas-là, la relation binaire entre les sommets est symétrique (si $u \rightarrow v$ alors $v \rightarrow u$). On utilise la notion « ordre du graphe » pour décrire le nombre de sommets de ce graphe. Un sous-graphe d'un graphe G est défini comme un graphe G' composé de certains sommets de G, ainsi que de toutes les arêtes reliant ces sommets dans G :

$$G' \subseteq G: G' graphe(V', E') \; avec \; V' \in V \; et \; E' \in E \tag{9.10}$$

D'un point de vue pratique, le processus de recherche des composantes connexes d'un graphe non orienté quelconque aboutit à trouver ses sous-graphes connexes maximaux. Dans ce contexte, « connexe » signifie qu'il existe une suite d'arêtes permettant d'atteindre v à partir de u.

Pour systématiser les relations entre les points d'un nuage, il est possible de les représenter sous forme d'une matrice d'adjacence servant ordinairement à caractériser des graphes non orientés. Il s'agit de la matrice symétrique $A(G)$ de dimension $N \times N$ (N – nombre de points du nuage) de telle sorte que $A(G) = (a_{ij}): 1 \leq i, j \leq N$. S'il existe une arête (en terme des graphes) entre deux points i et j, alors a_{ij} est égal à l'unité, sinon on lui attribue une valeur égale à zéro. L'inconvénient majeur d'une telle représentation des nuages de points est que l'on a besoin d'une énorme quantité de mémoire pour son stockage. Cependant, tous les graphes peuvent être classés en deux catégories : les graphes denses et creux (*sparse*). Ces derniers contiennent beaucoup moins d'arêtes que le nombre de sommets au carré ($|E| \leq |V|^2$). Or, la construction de matrice d'adjacence est superflue pour ce type de graphes. Elle ne peut pas être aussi applicable à des nuages de points.

L'idée de notre approche consiste à créer plutôt une liste d'adjacence (liste de successeurs) puisque c'est une représentation alternative de la matrice d'adjacence. En fait, elle comprend, pour chaque point (sommet), une liste de sommets voisins à celui étant examiné (Figure 9.13c). Pour ce faire, il faut mettre en œuvre un algorithme de recherche de voisinage local comme nous l'expliquerons plus loin. Son choix va imposer un mode de création des connexions entre les points du nuage et donc, un nombre d'arêtes obtenues en dépendra. Tous les points relativement proches de celui analysé seront reliés par une arête adéquate.

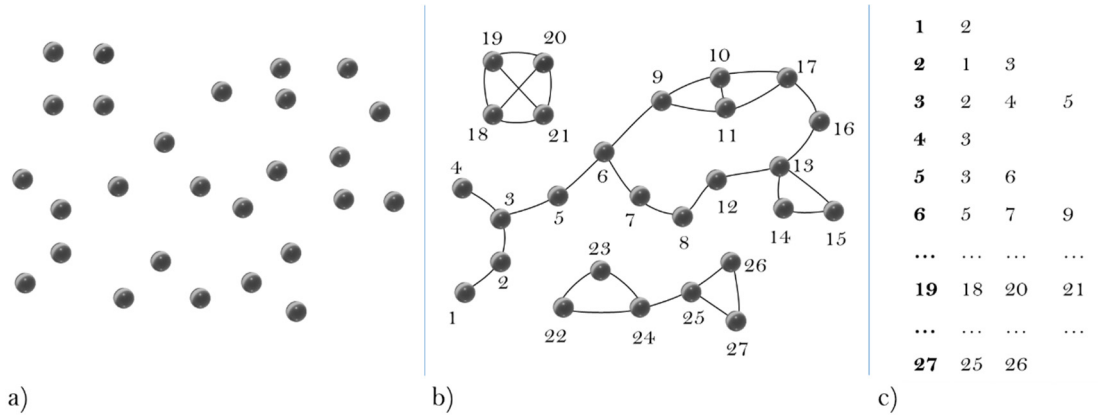

Figure 9.13 Représentation d'un nuage de points : a) Nuage de points de départ ; b) Graphe non orienté ; c) Liste d'adjacence

Voisinage local

Au regard de récentes publications, deux définitions de voisinage local s'avèrent plus répandues [Rabbani *et al.*, 2006]. La première, utilisée par [Castillo et Zhao, 2009], amène à prendre un échantillon, autour d'un point, composé des k voisins les plus proches *k-NN* (*k-Nearest Neighbors*) selon une distance à définir. La métrique peut être donnée par la distance *Euclidienne*, de *Manhattan* ou autre. En outre, la recherche des *k-NN* est souvent optimisée à l'aide de différentes stratégies de partitionnement de l'espace comme *l'arbre kd* (*kd-tree*) ou un *octree*. L'inconvénient majeur de cette technique est de dépendre de la densité du nuage. En admettant un nuage peu dense, le *k-ème* point peut être très éloigné du point envisagé et ne devrait pas être considéré comme voisin. En revanche, le fait d'être indépendant de la distance rend cette méthode plus avantageuse pour certaines applications. La seconde approche, employée entre autres dans les travaux de [Deschaud, 2010], mieux adaptée aux données de densité élevée, consiste à percevoir comme voisins tous les points inclus dans la sphère de rayon ρ. Un tel raisonnement permet d'être partiellement indépendant de la répartition des points. Par contre, cette méthode nécessite que la densité du nuage reste assez homogène. Sinon, le voisinage va inclure trop de points dans les régions denses et pas suffisamment de points dans celles ayant une densité faible. Dans le même esprit [Lari *et al.*, 2011] proposent un algorithme similaire dont l'idée consiste à établir le voisinage de chaque point par l'intermédiaire d'un cylindre adaptif, tout en prenant en compte la densité locale et la tendance de surface.

L'analyse de nos relevés laser mobile, en terme de répartition de point, confirme que les points au sein du nuage ne sont pas uniformément répartis sur la surface balayée. En effet, l'espacement entre les points n'est pas constant – il augmente avec la distance croissante objet-scanner. De plus, nous observons qu'il varie suivant deux directions. La distance entre deux points voisins en profil laser est plus petite que celle mesurée dans le sens de la direction de la plate-forme. Cette première va dépendre de la résolution angulaire du scanner laser, tandis que l'emplacement entre les profils laser change du fait de la vitesse du véhicule et de la fréquence du scanner.

En fin de compte, nous avons décidé d'unir les deux premières définitions et de déterminer le voisinage local comme un échantillon inclus dans la sphère de rayon ρ, mais en même temps de limiter le nombre de points autorisés à k. Dans ce cas, nous pouvons tirer profit des deux techniques. Le voisinage établi au sein des régions sur-densifiées ne sera jamais trop chargé, et celui créé pour les zones sous-densifiées ne contiendra pas de voisins inexacts.

Étiquetage des composantes connexes

Une fois le voisinage de chaque point du nuage établi, la matrice creuse A contenant une liste de toutes les paires est créée. L'étape précédente étant primordiale, nous pouvons procéder à l'extraction des composantes connexes à partir d'un graphe G.

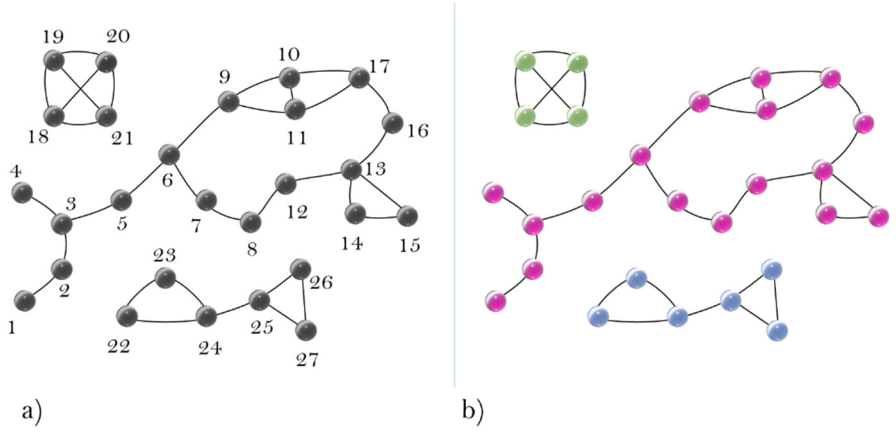

Figure 9.14 a) Graphe non orienté ; b) Trois composantes connexes détectées

L'une des façons de faire consiste à appliquer l'algorithme de parcours en profondeur *DFS* (*Depth First Search*) [Sarni et Lemarchand, 2011], [Corman *et al.*, 2001]. C'est un algorithme de recherche explorant un chemin « à fond », sommet par sommet, jusqu'à ce que l'on ne puisse plus avancer dans le graphe. Nous associons donc à un point de départ et à chacun des sommets visités lors de l'exploration, un même numéro de composante connexe. Après cette recherche deux cas peuvent se présenter :

1) Tous les sommets du graphe ont été atteints. Alors, le graphe est connexe et nous obtenons une seule composante connexe ;
2) Certains sommets n'ont pas été visités et, par conséquent, nous relançons l'algorithme à partir de l'un d'entre eux, en incrémentant l'indice de composante connexe.

Nous continuons ainsi tant qu'il subsiste des sommets non explorés. La Figure 9.14 illustre un exemple de résultats obtenus grâce à cet algorithme. Alternativement, il est possible d'employer l'algorithme de parcours en largeur *BFS* (*Breadth First Search*) qui réalise le parcours itératif d'un graphe en utilisant un fil. Il diffère du *DFS* par le fait qu'il liste d'abord les sommets voisins du point examiné pour ensuite les explorer un par un.

Ces deux solutions paraissant être coûteuses en temps de calcul, nous proposons de nous référer à une méthode plus efficace, rapide et facile à implémenter. Il s'agit d'appliquer l'un des algorithmes classiques de la théorie des graphes notamment la décomposition de *Dulmage-Mendelsohn*, notée la *DM-Décomposition* [Dulmage et Mendelsohn, 1958] de la matrice non carrée A formée au préalable. Celle-ci consiste à permuter les colonnes Q et les lignes P de la matrice creuse A (beaucoup plus grande que dense) représentant le graphe afin d'obtenir la forme bloc triangulaire inférieur (9.11).

$$A(P,Q) = \begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ 0 & 0 & A_{23} & A_{24} \\ 0 & 0 & 0 & A_{34} \\ 0 & 0 & 0 & A_{44} \end{bmatrix} \tag{9.11}$$

où : les sous-matrices A_{12}, A_{23}, A_{34} sont carrées avec une diagonale entièrement non nulle.

En d'autres termes, nous allons partitionner les lignes L (respectivement, les colonnes C) en trois sous-ensembles ce qui revient à partitionner la matrice A en neuf blocs, dont trois vont s'avérer être nuls (Figure 9.15). Pour rappel, chaque entrée non-nulle de la matrice creuse A représente une arête du graphe. Une telle matrice rectangulaire décrit naturellement un graphe biparti non-orienté où il y a deux types de nœuds, les sommets « lignes » L et les sommets « colonnes » C. Nous effectuons, pour mettre sous forme triangulaire par bloc la matrice d'adjacence A, la décomposition de *Dulmage – Mendelsohn*. Son idée consiste à calculer un couplage maximum (*matching*) M dans le graphe, c'est-à-dire, un sous-ensemble des arêtes le plus grand possible tel qu'elles n'aient aucun sommet en commun [Cormen *et al.*, 2001].

	C_h	C_s	C_v
L_h	M_h	A	B
L_s	0	M_s	C
L_v	0	0	M_v

Figure 9.15 Décomposition canonique de la matrice d'adjacence A

Nous introduisons, conformément à [Pothen et Fan, 1990], la terminologie suivante. Étant donné un couplage M, un sommet insaturé n'appartient à aucune arête du couplage. Une chaîne alternée est un chemin dans le graphe qui ne repasse pas deux fois par le même sommet, et dont une arête sur deux appartient au couplage. Une chaîne alternée est dite augmentante si elle relie deux sommets insaturés. Sur cette base, nous constatons qu'un couplage est maximal, si est seulement si, il n'existe pas de chaîne alternée augmentante relativement à M.

Une fois le couplage maximum calculé, le graphe G est alors partitionné comme suit :
L_v : sommets de L accessibles par une chaîne alternée depuis un sommet insaturé de L ;
L_h : sommets de L accessibles par une chaîne alternée depuis un sommet insaturé de C ;
C_v : sommets de C accessibles par une chaîne alternée depuis un sommet insaturé de L ;
C_h : sommets de C accessibles par une chaîne alternée depuis un sommet insaturé de C ;
$L_s = L - (L_v \cup L_h)$
$C_s = C - (C_v \cup C_h)$

Comme l'affirme [Bouillaguet, 2011], il est aussi démontrable que :

1) Tous ces ensembles sont deux-à-deux disjoints ;
2) Le couplage relie les lignes de L_v à des colonnes de C_v et respectivement les lignes de L_h à des colonnes de C_h. Par conséquent, $|L_v| > |C_v|$ et $|L_h| < |C_h|$;
3) Le couplage relie parfaitement les lignes de L_s aux colonnes de C_s. Il en découle que $|L_s| = |C_s|$.
4) Dans le graphe de départ il n'y a pas d'arêtes entre C_h et L_s, ni entre C_h et L_v, ni entre C_s et L_v.

Pour mieux illustrer la *DM*-Décomposition d'une matrice A, nous présentons le résultat obtenu avec la fonction *dmperm* (accessible sous Matlab et implémentée aussi dans notre algorithme) sur le nuage de la Figure 9.9a. Ainsi, la permutation des lignes P et des colonnes Q transforme la matrice d'adjacence A en forme illustrée par la Figure 9.16.

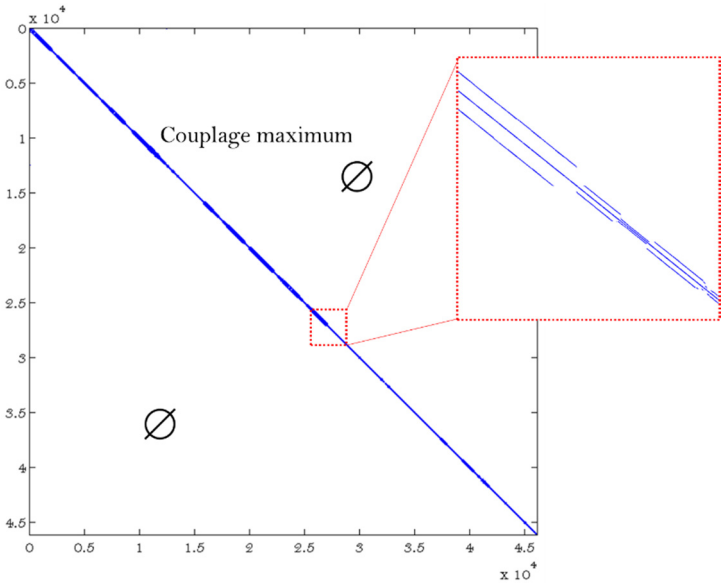

Figure 9.16 *DM*-Décomposition sur données réelles

Ensuite, les indices de points IDX du nuage N appartenant à la même composante connexe sont déterminés de la manière suivante :

$$IDX = P(r(i): r(i+1) - 1) \tag{9.12}$$

où :
P : liste permutée de colonnes
r : vecteur indiquant les limites de blocs pour la décomposition
$\|r\| - 1$: nombre de composantes connexes retrouvées

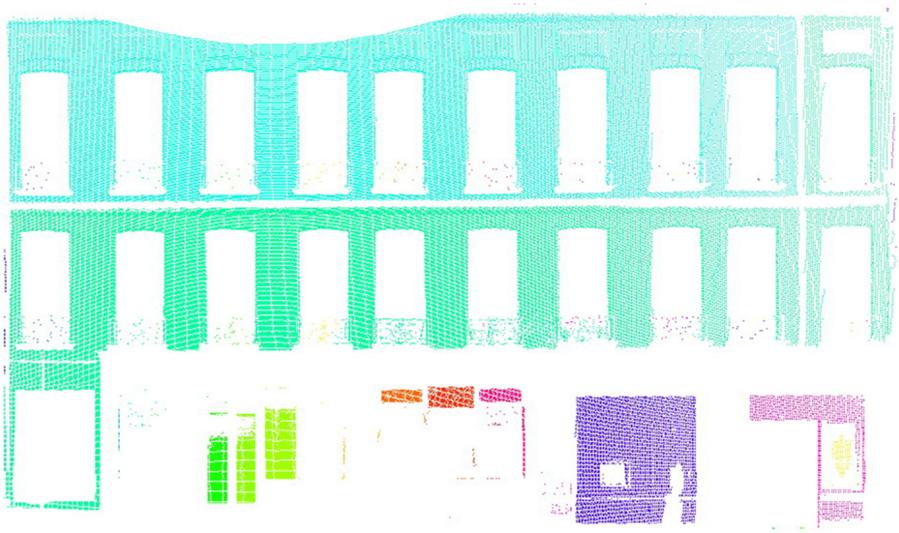

Figure 9.17 Composantes connexes extraites à l'aide de l'algorithme proposé. Paramètres fixés : R=0.10m, k=25, N_{min}=0. Une couleur correspond à une composante.

L'étiquetage des composantes connexes se fait de façon itérative, en incrémentant la valeur i jusqu'à ce que l'on atteigne le nombre de régions recherchées. À ce stade, nous pouvons introduire une contrainte concernant le nombre minimal de points N_{min} formant la

composante connexe. Tous les sous-ensembles ne satisfaisant pas aux exigences sont immédiatement rejetés. La Figure 9.17 présente les résultats d'étiquetage de composantes connexes en tenant compte de la décomposition effectuée (Figure 9.16).

Surface d'un sous-ensemble de points

Dès que toutes les composantes connexes sont extraites, nous calculons leurs surfaces. Nous entendons par ce terme l'aire de l'enveloppe convexe 2D (*convex hull*) englobant l'ensemble fini de points d'une composante. Nous cherchons ainsi le contour du plus petit polygone fermé et convexe délimité par tous les points (Figure 9.18).

Le calcul de l'enveloppe convexe se fait à l'aide du plan moyen Π sur lequel les points de la région sont projetés. Plusieurs algorithmes de complexité diverse ont été développés pour résoudre un tel problème notamment : la marche de *Jarvis* (*Gift wrapping algorithm*) [Jarvis, 1973], le parcours de *Graham* (*Graham's scan*) [Graham, 1972], l'heuristique *de Akl-Toussaint*, ou le diagramme de *Voronoï*. Nous avons employé la bibliothèque *CGAL* (*Computational Geometry Algorithms Library*) mettant à disposition l'algorithme *Quickhull* de [Bykat, 1978]. Finalement, l'aire de l'enveloppe convexe représente la taille de la composante connexe. Nous conservons uniquement, cette composante dont la surface est la plus importante. Les autres points reviennent au nuage pour participer de nouveau à la recherche d'un segment plan par l'algorithme de *RANSAC*.

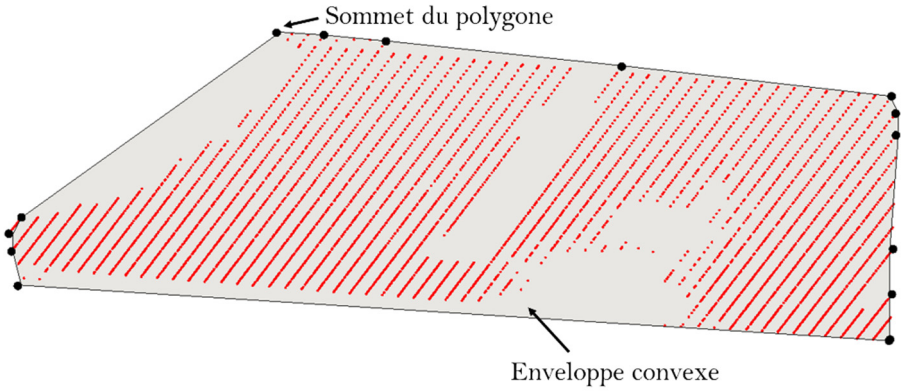

Figure 9.18 Aire du segment plan

Le pseudo-code 4.2 (Annexe E) présente notre approche d'analyse de composantes connexes telle qu'elle est décrite dans cette section. Trois paramètres sont attendus en entrée notamment le rayon de sphère ρ et la valeur k afin d'établir le voisinage du chaque point du nuage. Optionnellement, le nombre minimal de points N_{min}, peut être pris en compte afin de définir la taille de la plus petite composante connexe autorisée.

4.5 Méthode proposée d'extraction d'arêtes

Une fois tous les plans principaux détectés, nous procédons à extraire les segments de ligne. Nous cherchons à retenir uniquement les arêtes réelles figurant dans la scène. Étant donné que lors du processus de segmentation des surfaces planes décrit ci-dessus nous bénéficions non seulement d'un ensemble de modèles de plans $\prod$, mais aussi des sous-ensembles de points - segments plans (SP_i), le processus d'identification d'arêtes est assez aisé (Figure 9.19). Quant à l'ensemble $\prod$, il est défini de telle sorte que chacun de ces éléments soit caractérisé par un vecteur directeur n_i et un point A_i (par lequel le plan $\prod_i$ passe). Par conséquent, la droite d'intersection I entre deux plans $\prod_j$ et $\prod_k$ peut être facilement

retrouvée. Le vecteur directeur $\vec{u}$ de cette ligne est le produit vectoriel de deux vecteurs normaux :

$$\vec{u} = \vec{n_j} \times \vec{n_k} \tag{9.13}$$

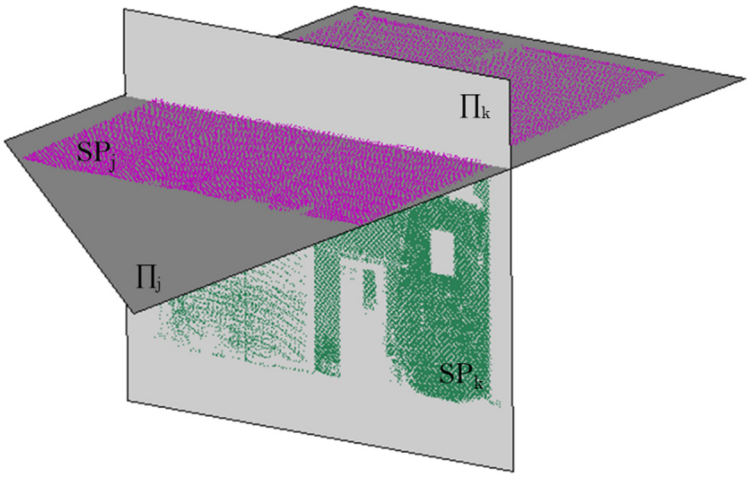

Figure 9.19 Segments plans extraits par *RANSAC*

Néanmoins, à cette étape deux questions doivent se poser : 1) Quels segments plans de l'ensemble $\prod$ sont sécants ?; 2) Comment définir les extrémités du segment de ligne résultant de cette intersection ? Pour répondre à ces interrogations, nous proposons une méthode qui s'attache à analyser toutes les combinaisons possibles des plans afin de dissocier ceux étant voisins. La Figure 9.20 contient le schéma complet de l'approche discutée.

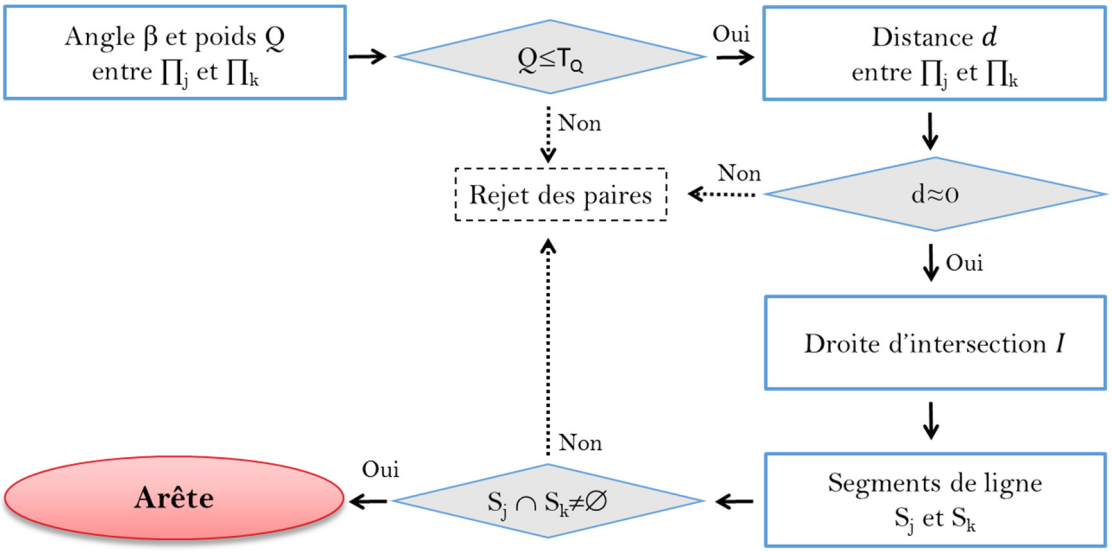

Figure 9.20 Procédure proposée d'extraction d'arêtes

Premier rejet des paires de plans – analyse de l'angle

Dans un premier temps, l'angle β entre toute paire candidate de plans est calculé, ainsi que le poids Q – critère de la qualité de cette intersection. Les deux valeurs peuvent être quantifiées en considérant les vecteurs normaux n_j et n_k des deux plans étudiés.

Du point de vue géométrique, la norme de leur produit vectoriel se définit comme :

$$\|\vec{u}\| = \|\vec{n_j}\| \cdot \|\vec{n_k}\| \cdot |sin\beta| \tag{9.14}$$

Puisque les deux vecteurs sont unitaires (la norme est égale à l'unité), le poids Q peut se calculer de la manière suivante :

$$Q = \|\vec{u}\|^2 = sin^2\beta \tag{9.15}$$

Il est évident que l'intersection est la meilleure si l'angle β est droit. Or, le poids Q associé doit être égal ou proche de l'unité. Si Q est petit, alors la longueur du vecteur $\vec{u}$ est faible et, par conséquent, la droite d'intersection est mal définie. De même, l'intersection n'existe pas, si deux plans sont parallèles ($Q = 0$). L'analyse des poids Q nous permet de réduire la taille de l'ensemble contenant des paires « potentielles » de plans. Dans ces conditions, lors du premier rejet, nous ne conservons que les paires dont le facteur Q est supérieur à T_Q. La valeur T_Q a été fixée à 0.5, ce qui revient à dire que nous ne prenons en considération que les paires de plans se coupant sous l'angle de 45° à 135°. Toutefois, rien n'empêche d'accroître les exigences envers les intersections recherchées et d'augmenter la limite basse T_Q.

Deuxième rejet des paires de plans – analyse de la distance

L'étape suivante nous amène à vérifier la distance d entre les paires restantes et à garder uniquement celles composées par les plans voisins. Considérant deux segments plans chacun doté d'un sous-ensemble de points valables SP_j et SP_k isolé lors de la segmentation, mais aussi l'équation du plan approximant $\prod_j$ et $\prod_k$ passant par ce nuage (Figure 9.19), nous pouvons vérifier leur distance réciproque. Nous calculons simplement les distances *Euclidiennes* les plus courtes des points du nuage SP_j au plan $\prod_K$ et vice versa (9.16). Il en résulte que les distances minimales d_1 et d_2, sont requises afin de pouvoir juger si les plans sont suffisamment proches l'un de l'autre. Les segments plans sont considérés comme adjacents, si est seulement si, la distance d définie comme $d = min(d_1, d_2)$ est proche de zéro, avec une certaine tolérance.

$$d_1 = \min_{j \in \aleph}\{d(SP_j, \prod_k) : j \in (1, N)\} \qquad d_2 = \min_{k \in \aleph}\left\{d\left(SP_k, \prod_j\right) : k \in (1, M)\right\} \tag{9.16}$$

pour N, M : nombre de points respectivement dans les sous-nuages SP_j et SP_k.

Cette condition de proximité étant primordiale, elle peut être toutefois insuffisante dans certains cas. La Figure 9.21 illustre un exemple particulier où les deux segments plans SP_j et SP_k (violet et vert) sont loin d'être sécants, alors que la distance d est égale à zéro. De ce fait, ils sont censés aboutir à une droite d'intersection. Nous souhaitons reconstruire fidèlement les arêtes existantes dans le nuage et éviter de créer ce genre de faux segments.

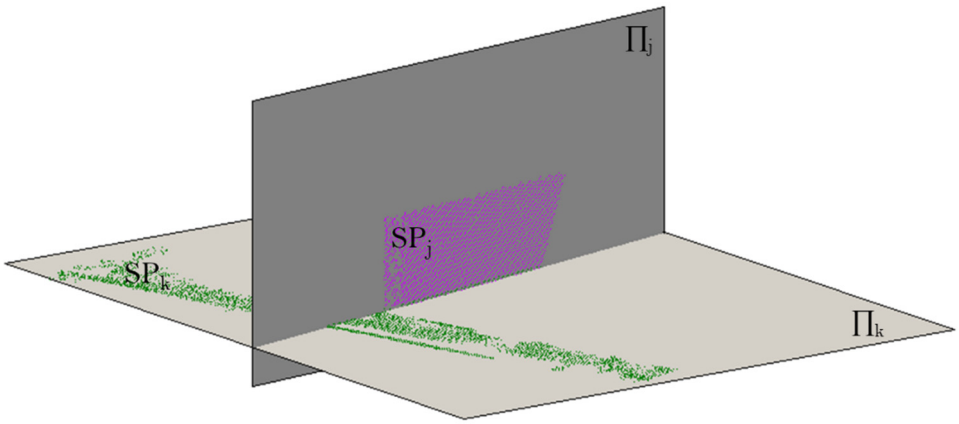

Figure 9.21 Exemple d'appariement devant être rejeté

Troisième rejet – analyse des extrémités des segments

Pour remédier aux faux appariements, nous examinons la longueur et les extrémités des segments obtenus par cette intersection. Sachant que les paires « candidates » parallèles ont déjà été rejetées, nous déterminons la droite I entre deux plans sécants, de sorte que $\prod_j \cap \prod_k = I$, à l'aide de la formule (9.13).

Ensuite, les points SP_j et SP_k situés à une distance inférieure à une tolérance prédéfinie T_d sont projetés sur cette droite. Plus précisément, les points se trouvant autour de leur ligne d'intersection servent à identifier les extrémités d'arête de pli. De cette façon, deux segments de ligne S_j et S_k sont retrouvés puisque chaque nuage a des limites différentes (Figure 9.22).

La dernière étape – l'analyse de la position spatiale des extrémités appartenant aux segments d'intersection créés S_j et S_k, permet de mettre de côté les paires de plans indûment classées et d'établir la longueur de l'arête finale. Nous ne conservons que les paires de plans dont l'intersection aboutit à deux segments se chevauchant au moins partiellement. L'étude de la position relative de ces segments colinéaires est menée en analysant la distance entre leurs milieux (M_j, M_k), ainsi que leurs longueurs (L_j, L_k). Trois cas peuvent se présenter :

1) L'un des segments est complètement inclus dans l'autre (Figure 9.23a) : $\|M_jM_k\| \leq |L_j/2 - L_k/2|$. En conséquence, la longueur de l'arête créée correspond au plus court segment ;
2) Les segments se chevauchent partiellement (Figure 9.23b) : $|L_j/2 - L_k/2| < \|M_jM_k\| < |L_j/2 + L_k/2|$ et donc la longueur de l'arête retenue est exactement la même que celle de la zone de chevauchement ;
3) Les segments sont disjoints (Figure 9.23c) : $\|M_jM_k\| \geq |L_j/2 + L_k/2|$ alors l'arête n'existe pas et la paire de plans est supprimée.

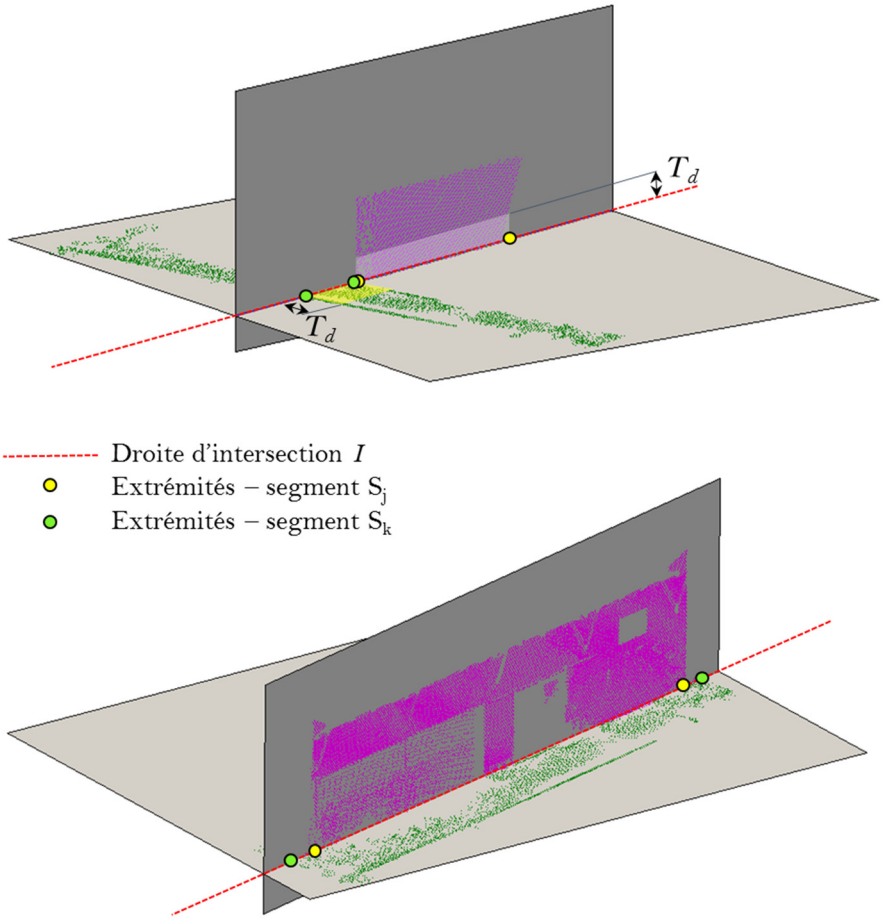

Figure 9.22 Droite d'intersection entre les plans $\prod_j$ et $\prod_k$ et les extrémités de segments S_j et S_k

Sachant que nous cherchons à extraire les segments les plus longs possible, nous pouvons, optionnellement, prendre en compte une contrainte T_{seg}, c'est-à-dire la longueur minimale d'une arête recherchée. Dès lors les arêtes trop petites seront exclues des résultats.

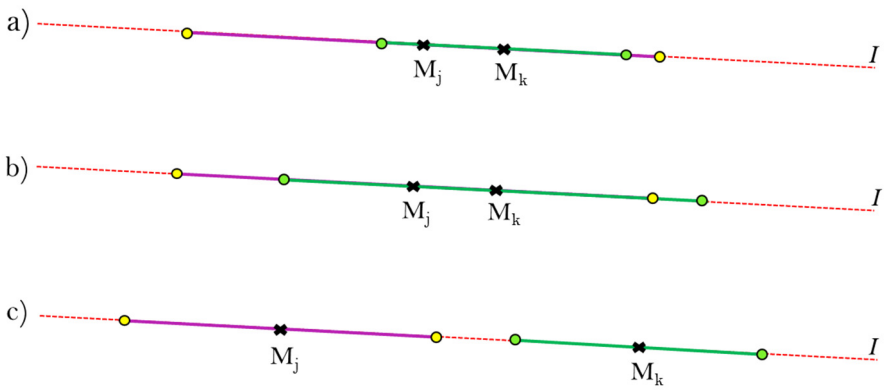

Figure 9.23 Relations spatiales entre les extrémités de segments S_j et S_k : a)b) arête retenue ; c) arête rejetée

Le pseudo-code 4.3 (Annexe E) détaille notre algorithme d'extraction d'arêtes de pli. Nous avons besoin en entrée de segments plans (SP_i) isolés lors de la segmentation ainsi que les paramètres de plan ($\prod_i$) approximant chaque sous-ensemble de points. Nous obtenons, à la sortie, un ensemble d'arêtes SD de longueur supérieure à T_{seg}.

4.6 Exemples de résultats

L'algorithme d'extraction d'arêtes discuté dans ce chapitre a été testé avec plusieurs nuages de points, qu'ils soient mobiles ou statiques. Nous présentons dans cette partie quelques résultats finaux (4.6.2), ainsi que ceux intermédiaires concernant la détection de segments plans (4.6.1). Sachant que notre algorithme a été développé entièrement sous *Matlab*, inadapté pour traiter de gros nuages de points, nous ne renseignons pas le temps de calcul (étant assez conséquent en *Matlab*). Mais, on parle d'environ 50 minutes (sous Intel Core x2 T7600 2.33GHz/4 Go Ram/Win7 64bits) en ce qui concerne les résultats illustrés par la Figure 9.25 (au total 265 266 points ont été traités).

4.6.1 Détection de plans

La Figure 9.24 montre soixante plans successifs extraits pour un fragment du nuage de points *Stereopolis2009_Soufflot*. Les valeurs de tous les paramètres nécessaires pour faire fonctionner notre algorithme sont listées au-dessous de l'image. Nous remarquons que le plan perpendiculaire au sens du déplacement du véhicule, couvrant une superficie assez grande, n'a pas été retrouvé. Cet état de fait est dû à l'angle d'incidence du faisceau laser qui était important, et influençait le taux de bruit entachant les données élevées. En outre, la position de la façade vis-à-vis de la tête du scanner fait que peu de profils laser l'interceptent. La distance moyenne entre les profils consécutifs est donc faible. Nous dénombrons environ 15 lignes sur 5m linéaires de mur, ce qui donne localement une ligne sur 30cm. Nous pourrions *ipso facto* mettre en doute le fait que ces points forment un plan satisfaisant aux exigences imposées. Quant aux autres plans, nous observons le phénomène de sur-segmentation se dévoilant quand les surfaces uniformes en réalité sont segmentées en au moins deux plans distincts.

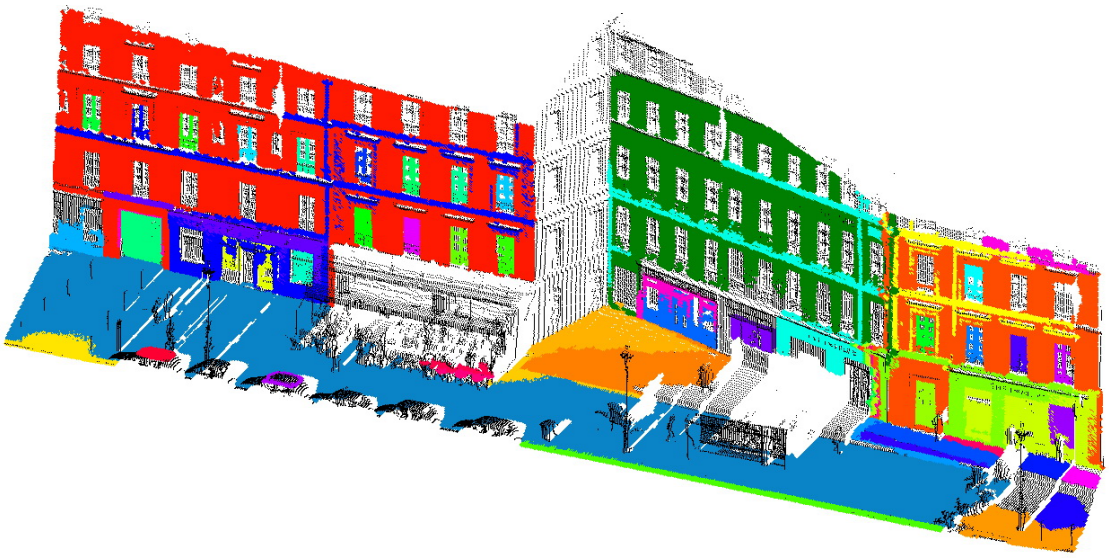

Figure 9.24 Détection de plans - *Stereopolis2009_Soufflot*. Paramètres fixés : σ=0.015m, P=0.99, N_{pl}=60, R=0.30m, k=25, N_{min}=100. Une couleur correspond à un plan (points non segmentés en noir)

La Figure 9.25 illustre les plans extraits à partir du nuage *Riegl2012_Slomniki* se caractérisant d'un faible taux de bruit. Les résultats sont corrects, mais le problème de sur-segmentation apparaît toujours, surtout lors de la segmentation du sol.

Figure 9.25 Détection de plans – *Riegl2012_Slomniki*. Paramètres fixés : σ=0.01m, P=0.99, N_{pl}=16, R=0.200m, k=25, N_{min}=200. Une couleur correspond à un plan (points non segmentés en noir)

D'autres types de problèmes rencontrés lors de l'utilisation de l'algorithme de *RANSAC* sont montrés par la Figure 9.26. Elle illustre la recherche du premier plan dans le nuage de points mobile. Les études étaient menées sur les données brutes. Le plan mis en évidence était pour l'estimateur la meilleure solution en raison de la forte accumulation des points acquis durant l'arrêt forcé de la plate-forme. Pour pallier ces fausses détections, l'étape de prétraitement des données laser paraît indispensable – le débruitage, la décimation ou bien encore la contrainte de distance minimale entre les points voisins.

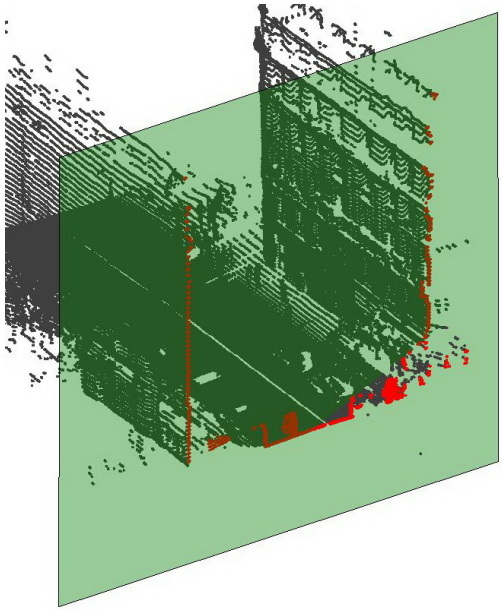

Figure 9.26 Exemple de problèmes survenus lors de la détection des plans via *RANSAC* (en rouge les points appartenant au premier plan détecté). Nuage de points du système L3D2

Nous pouvons constater que l'algorithme de *RANSAC* est sensible à la densité des nuages de points. Par contre, il fournit de très bons résultats, en peu de temps, si le nuage est homogène et sa densité élevée. L'approche proposée d'analyse de composantes connexes vérifie correctement la connectivité du segment plan. Néanmoins, elle exige un temps considérable afin de trouver le voisinage local de chaque point, ce qui est son seul point faible. L'accélération de recherche des voisins faite par un *arbre kd* ou un *octree*, au lieu d'un parcours linéaire, serait appréciée. L'un des futurs travaux sera de tester sa robustesse par rapport aux autres méthodes d'extraction des composantes connexes, notamment les algorithmes récursifs d'exploration de graphe (*DFS*, *BFS*), ou bien encore la solution d'analyse du voisinage à l'aide d'image 2D.

4.6.2 Extraction d'arêtes

La Figure 9.27 illustre des résultats d'extraction d'arêtes dans le nuage de points acquis par les systèmes mobiles. Tandis que la Figure 9.28 montre les segments extraits à partir du nuage *Sulpice2013_statique*. Notre algorithme arrive à détecter correctement toutes les arêtes, mais leur précision et le nombre dépendra fortement des segments plans pris en compte. Dès lors, l'algorithme choisi de segmentation prédéterminera la complétude et la qualité de la solution.

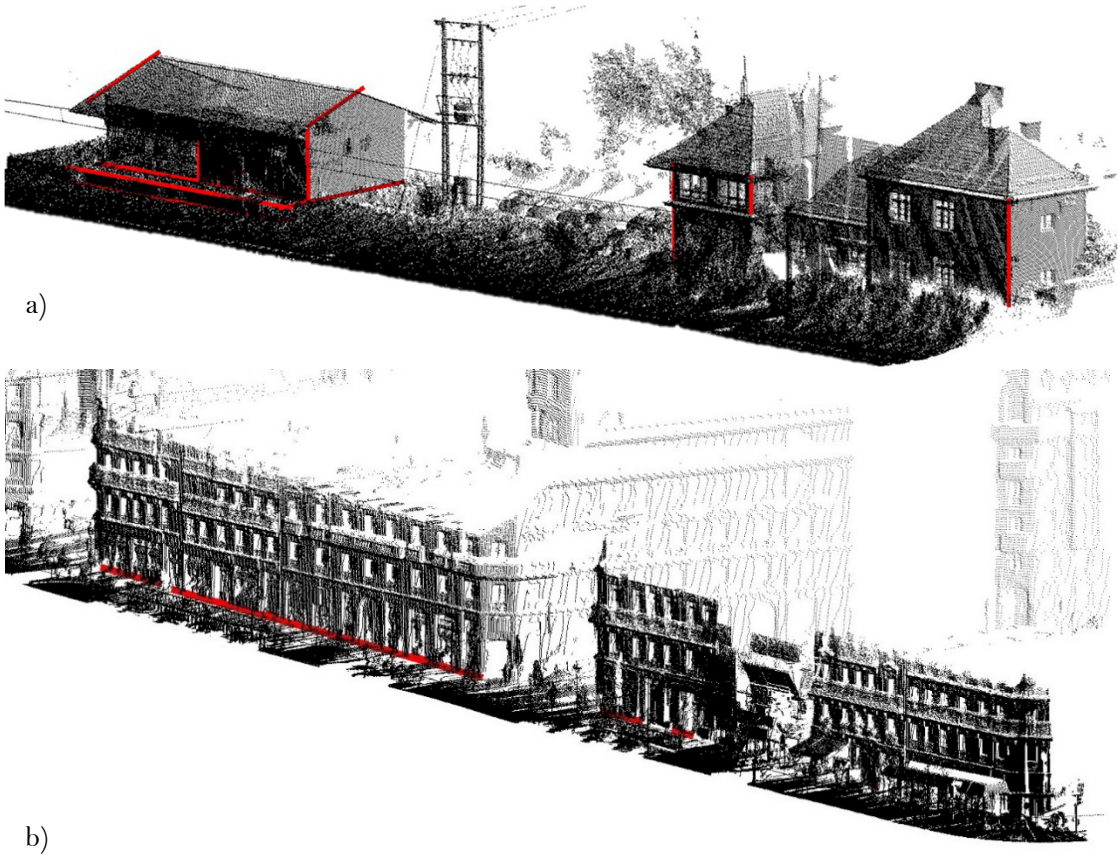

Figure 9.27 Arêtes extraites à l'aide de l'algorithme proposé (T_{seg}=2.000m).
a) *Riegl2012_Slomniki* ; b) *Stereopolis2009_Soufflot*

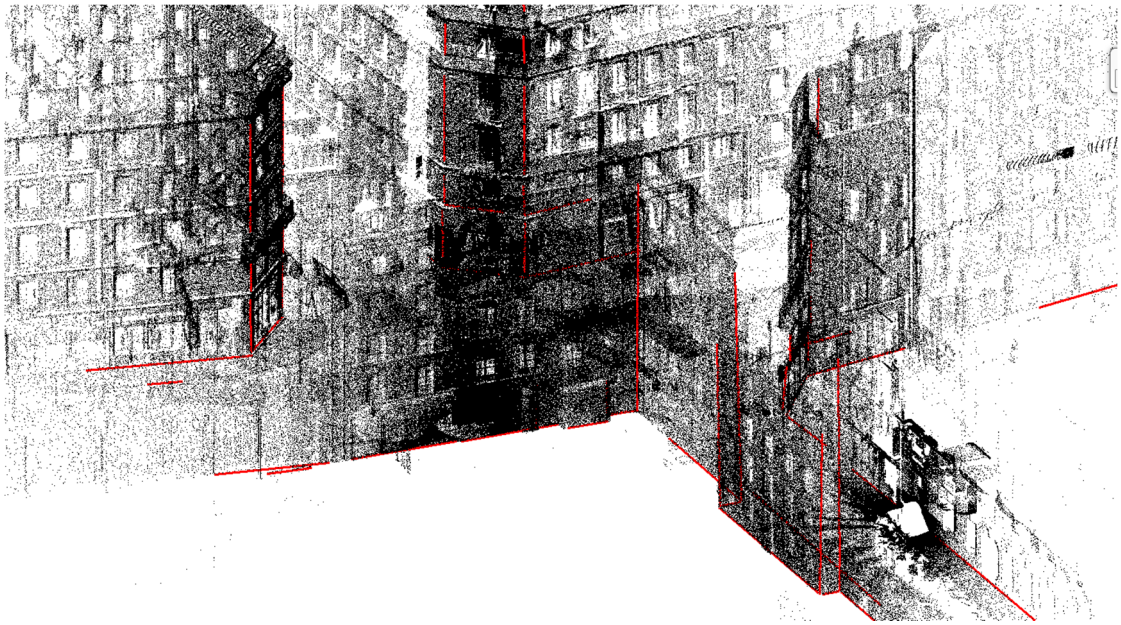

Figure 9.28 Arêtes extraites à l'aide de l'algorithme proposée (T_{seg}=1.000m) : *Sulpice2013_statique*

CHAPITRE 5

Qualification à partir d'entités linéaires

Sommaire :

5.1 Problématique **87**
5.1.1 Comment mesurer la distance entre deux segments dans l'espace ? 88
5.1.2 Contraintes 90
5.2 Mesure de distance proposée **91**
5.2.1 Distance de *Hausdorff* 92
5.2.2 Matrice de similarité 94
5.2.3 Mise en correspondance 96
5.3 Résultats et évaluation **99**
5.3.1 Données expérimentales 99
5.3.2 Évaluation de l'algorithme d'appariement 100
5.3.3 Exactitude et précision 105

Résumé

Dans ce chapitre, une nouvelle approche de qualification de relevés laser mobiles, son implémentation et les résultats sont présentés. Celle-ci, basée sur les arêtes, semble être efficace et tout particulièrement pour les besoins d'évaluation des nuages dont la densité faible empêche une identification précise des points de contrôle. Adressée principalement à des zones urbaines, riches en primitives linéaires, notre méthode délivre une mesure de qualité en s'appuyant sur la distance modifiée de Hausdorff (LHD). La mise en correspondance de segments homologues, menée simultanément, joue un rôle crucial et, par conséquent, elle doit être fiable. Nous proposons alors un algorithme entièrement automatique, consistant à analyser la similitude entre les arêtes extraites de données mobiles et celles de référence. Sa valeur, recueillie par la matrice de similarité, est déterminée en tenant compte de la distance et de l'angle. Enfin, nous testons la robustesse d'un tel algorithme d'appariement, et qualifions les relevés laser mobiles en termes d'exactitude et de répétabilité.

5.1 Problématique

Dans le dessein de développer une méthode de qualification de relevés laser mobiles acquis dans un environnement urbain, nous proposons une approche basée sur les entités linéaires. Celles-ci, représentant des intersections entre des plans principaux modelés dans le nuage,

peuvent être extraites à l'aide de l'algorithme présenté dans le Chapitre 4. Ensuite, la technique mise en œuvre se résume à une comparaison de deux ensembles de segments. Le premier ensemble *Test*, contenant des droites à qualifier, sera confronté avec le second nommé *Model*. Ce dernier, considéré désormais comme référence, se compose des segments plus précis provenant, entre autres de levés topographiques, de nuages de points collectés en mode fixe, mais aussi d'autres bases de données, qu'elles soient 2D ou 3D. Notre solution, contrairement à celles envisagées par les méthodes se référant aux points de contrôle, vise à diminuer les problèmes liés à l'identification correcte de cibles correspondantes. Toutes les démarches détaillées au fur à mesure dans ce chapitre ne nécessiteront plus l'intervention d'opérateur. L'appariement des segments s'effectue automatiquement, en examinant la similitude des droites. Grâce à la comparaison des segments entre eux, il est possible de repérer les entités linéaires *Model* (M) identiques ou très proches de celles de l'ensemble *Test* (T). Mais, cette tâche devient encore plus complexe, quand on présume que le nombre de segments peut varier d'un ensemble à un autre, et que leur longueur ne sera pas forcément la même. En outre, nous observons également des droites pour lesquelles aucun homologue n'existe dans l'ensemble de référence (Figure 5.1). Ce constat est dû au fait que le processus d'extraction des segments peut engendrer des phénomènes de sur-segmentation et de sous-segmentation.

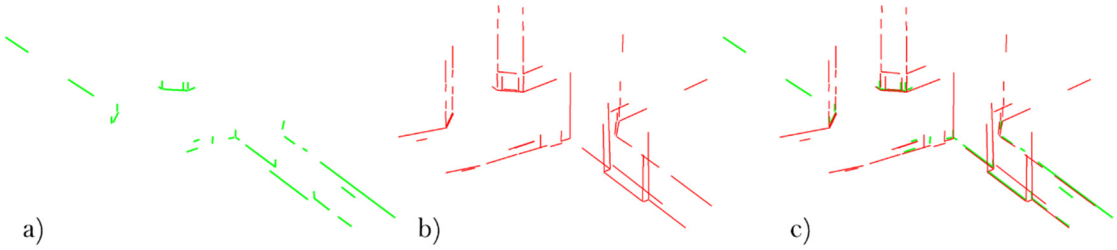

Figure 5.1 Exemple d'arêtes à comparer : a) segments $Test$; b) segments $Model$; c) les deux ensembles superposés

Dans ce chapitre, une méthode de qualification sera présentée (Section 5.2), ainsi que les résultats obtenus. Aussi, l'algorithme de mise en correspondance des segments 3D sera examiné par rapport à la vérité-terrain (Section 5.3). Pour compléter ces analyses, l'impact du décalage réciproque entre les segments étudiés sur la robustesse de l'algorithme d'appariement sera vérifié. Une fois le couplage créé, nous proposerons d'évaluer les nuages de points en termes d'exactitude et de précision (répétabilité). C'est pourquoi la distance moyenne calculée à partir de segments, étant la distance modifiée de *Hausdorff LHD* (*Line Hausdorff Distance*), servira à noter les nuages. Finalement, les avantages et les inconvénients d'une telle procédure de qualification de relevés laser seront mis en avant.

5.1.1 Comment mesurer la distance entre deux segments dans l'espace ?

L'utilisation des arêtes, au lieu des points de contrôle, pour les besoins de qualification, exige d'élaborer une métrique qui permettrait de comparer deux ensembles d'entités linéaires. Certes, nous ne pouvons plus nous référer uniquement aux extrémités, puisque les segments correspondants ne sont pas censés ni partager les mêmes coordonnées ni avoir la même longueur. L'une des façons de faire peut être de définir la moyenne des distances d calculées entre les segments homologues provenant de différentes sources. La plupart des méthodes abordant cette question s'appuient sur le calcul de la distance *Euclidienne*. Nous pouvons aboutir à différentes variantes, comme l'illustre la Figure 5.2, visant, par exemple, à déterminer la distance minimale, la distance entre les milieux de segments correspondants (M_P, M_Q), ou leurs extrémités (P_0, P_1, Q_0, Q_1).

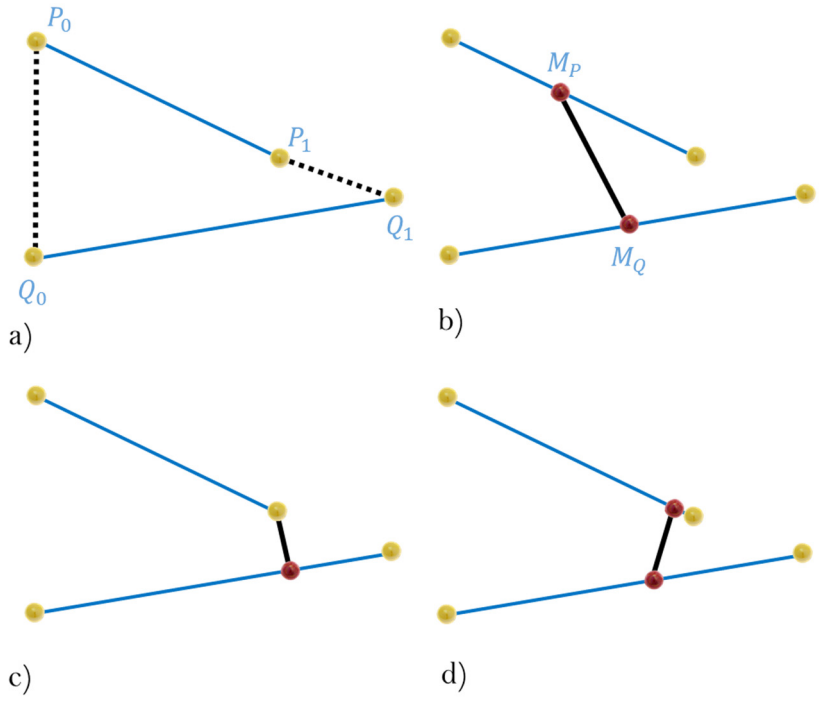

Figure 5.2 Définition de la distance entre deux segments 3D : a) Distances déterminées entre les extrémités ; b) Distance calculée entre les milieux ; c) Distance perpendiculaire d'une extrémité de segment à l'autre segment ; d) Distance minimale ;

En considérant la distance d entre les extrémités des segments (Figure 5.2a), nous calculons tout simplement les distances d'un point 3D à l'autre :

$$d(P_0, Q_0) = \sqrt{\sum_{i=1}^{3} \left(P_{0_i} - Q_{0_i}\right)^2} \qquad d(P_1, Q_1) = \sqrt{\sum_{i=1}^{3} \left(P_{1_i} - Q_{1_i}\right)^2} \tag{5.1}$$

Sachant que les coordonnées du milieu d'un segment sont les demi-sommes de chacune des coordonnées des extrémités, nous introduisons, de manière similaire (formule (5.1)), la distance d calculée, cette fois-ci, entre les milieux respectifs (Figure 5.2b).

De même, nous pouvons envisager la distance des extrémités du premier segment (P_0, P_1) par rapport à la droite L passant par le second segment, et vice versa (Figure 5.2c). Il s'agit alors de la distance minimale d'un point à une droite L, mesurée suivant la formule ci-dessous :

$$d(P_1, L) = \frac{\left\|\overrightarrow{Q_0P_1} \wedge \vec{v}\right\|}{\|\vec{v}\|} \tag{5.2}$$

pour la droite L définie par un point Q_0 et un vecteur $\vec{v}$ non nul. L'indice $\wedge$ symbolise le produit vectoriel de deux vecteurs.

Dans tous les cas mentionnés, l'étape dernière nous amène à prendre la moyenne ou le minimum des distances d obtenues.

Cependant, la formulation de la distance minimale entre deux segments n'est pas aisée (Figure 5.2d). Aussi, elle ne peut pas être confondue avec la distance minimale entre les droites traversant ces segments puisque ces valeurs sont souvent différentes (Figure 5.3). Or,

le point d'une ligne désigné comme le plus proche de l'autre ligne peut ne pas être situé sur la partie de droite délimitée par deux points fixes (extrémités).

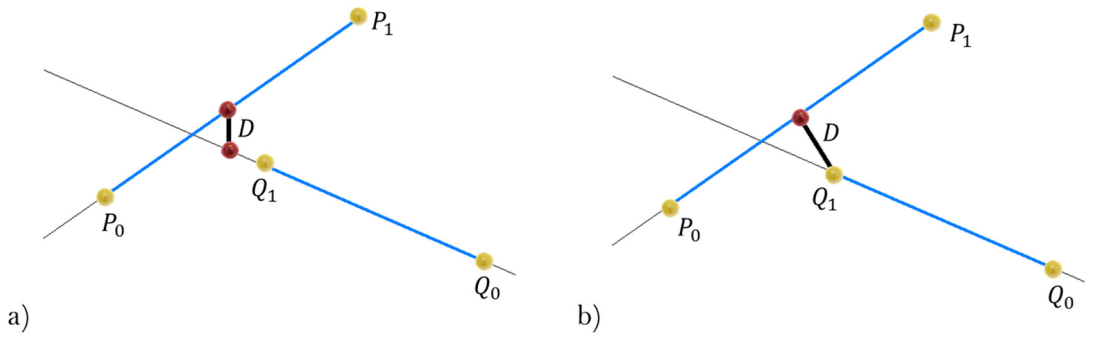

Figure 5.3 a) Distance entre deux droites; b) Distance entre deux segments de droite

[van Verth et Bishop, 2008] avancent une solution de [Sunday, 2001] permettant de résoudre cette tâche. Étant donné que la distance entre deux objets géométriques P et Q est définie comme la distance entre deux points quelconques $p \in P$ et $q \in Q$ (5.3), le problème se résume toujours à trouver ces points minimisant la distance d.

$$d(P,Q) = \min_{p \in P, q \in Q} d(p,q) \tag{5.3}$$

Quant à la distance entre deux segments, nous l'assimilons au calcul de la distance d'une ligne à l'autre. Nous représentons le segment $S_1 = [P_0, P_1]$ par l'équation paramétrique de droite $L_1: P(s) = P_0 + s(P_0 - P_1) = P_0 + s\vec{u}$ et le segment $S_2 = [Q_0, Q_1]$ comme $L_2: Q(t) = Q_0 + t(Q_0 - Q_1) = Q_0 + t\vec{v}$ pour $0 \leq s, t \leq 1$. Ainsi, nous cherchons, pour L_1 et L_2, les paramètres s_C et t_C de telle sorte qu'ils décrivent la position des points les plus proches $P(s_C)$ et $Q(t_C)$. Une seule différence réside dans le fait que la recherche de ces points est restreinte à l'intervalle défini par les extrémités de chacun des segments. Finalement, la distance minimale entre deux segments S_1 et S_2 correspond à la norme du vecteur $\overrightarrow{w_C}$ dont les coordonnées sont données par les points $P(s_C)$ et $Q(t_C)$.

$$d(S_1, S_2) = \|\overrightarrow{w_C}\| = \|P(s_C) - Q(t_C)\| \tag{5.4}$$

En minimisant la norme de $\overrightarrow{w_C}$ nous arrivons à déterminer les paramètres s_C et t_C. La description plus exhaustive de cette approche fait partie de l'Annexe A.

5.1.2 Contraintes

Afin d'être en mesure de vérifier la qualité (l'exactitude et la précision) à partir des segments de droite, la distance réciproque $d(t,m)$ entre eux doit être définie de manière à refléter leur véritable éloignement. Cependant, les approches listées auparavant ne sont pas suffisamment robustes pour nos applications puisqu'elles n'envisagent ni orientation de droites ni situation où les segments se chevauchent seulement partiellement. Nous en avons pour preuve quelques exemples présentés par la Figure 5.4. Nous y voyons clairement que, dans certains cas, les distances déterminées ne restituent pas de vraies relations mutuelles entre les segments. Ainsi, pour les droites quasiment parallèles (Figure 5.4a,b), cette valeur peut être importante, et inversement, les deux segments orientés différemment (Figure 5.4c,d,e) peuvent être classés comme relativement proches (distance faible, voire égale à zéro).

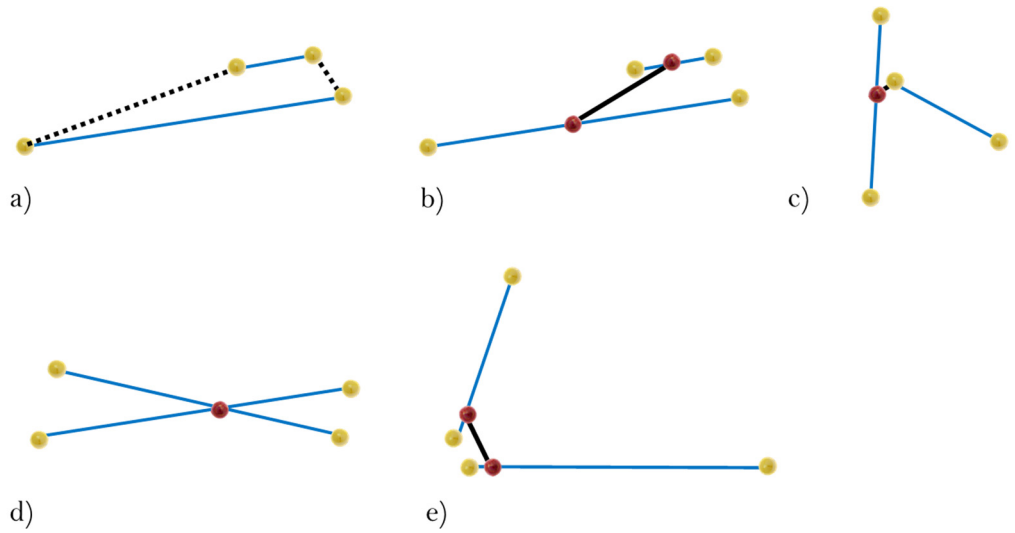

Figure 5.4 Exemples générant des calculs de distances erronés : a) Distances déterminées entre les extrémités ; b) Distance calculée entre les milieux ; c) Distance perpendiculaire d'une extrémité de segment à l'autre segment ; d)e) Distance minimale ;

Afin de surmonter ce genre d'inconvénients [Gao et Leung, 2002] suivis par [Chen *et al.*, 2003] ont proposé une approche dédiée à la comparaison des segments 2D extraits à partir des logos (images). Nous nous en sommes inspirés pour notre méthode en l'adaptant à un espace tridimensionnel. Dès lors, la distance entre les segments 3D ralliera l'écartement, mais aussi l'angle formé entre chaque paire de droites homologues. La section suivante abordera en détail ce sujet.

5.2 Mesure de distance proposée

La Figure 5.5 présente un aperçu général des étapes de calcul constituant l'approche proposée. Dans un premier temps, la similarité entre toutes les combinaisons possibles de droites, provenant de deux ensembles T ($Test$) et M ($Model$), sera notée sous forme d'un score $d(t,m)$. Celui-ci étant le coefficient de la matrice de similarité, il fera, par la suite, l'objet d'une étude particulière. Sa valeur servira à vérifier si les deux segments sont suffisamment similaires pour être appariés, et s'ils sont susceptibles d'être employés pour déterminer la distance moyenne – notre mesure de qualité. Étant donné que seuls les segments quasi-parallèles peuvent être appariés, on se devra de prendre en compte l'orientation et la position des segments pour calculer chaque score $d(t,m)$. Le paragraphe 5.2.2 se penchera sur ce sujet.

Passons à présent à la mise en correspondance des segments. Nous devons trouver le moyen permettant de conclure que les deux entités comparées sont semblables. Nous mémoriserons uniquement les paires dont les scores sont jugés négligeables, c'est-à-dire, inférieurs ou égaux à un seuil que l'on s'est fixé automatiquement en fonction de valeurs $d(t,m)$ sélectionnées. Toutes les paires retenues formeront ensuite la matrice binaire de correspondance Cor, systématisant les interactions entre les éléments linéaires de deux ensembles étudiés $Test$ et $Model$.

Une fois l'appariement retrouvé, nous déterminerons l'écartement des données traitées. Cette mesure, étant la distance modifiée de *Hausdorff LHD* (*Line Hausdorff Distance*), évaluera, d'une manière objective, l'exactitude du relevé laser en calculant sa distance par rapport à la référence. Également, la qualification de la précision deviendra possible à réaliser, sous réserve que des nuages redondants soient collectés au sein de zones d'intérêt. Comme nous

l'expliquerons plus loin, toutes les opérations seront menées dans deux sens : l'ensemble $Test$ sera comparé avec le $Model$ et vice versa. Par conséquent, deux matrices de correspondance seront obtenues, ainsi que deux distances relatives de *Hausdorff OLHD* (*Oriented Line Hausdorff Distance*).

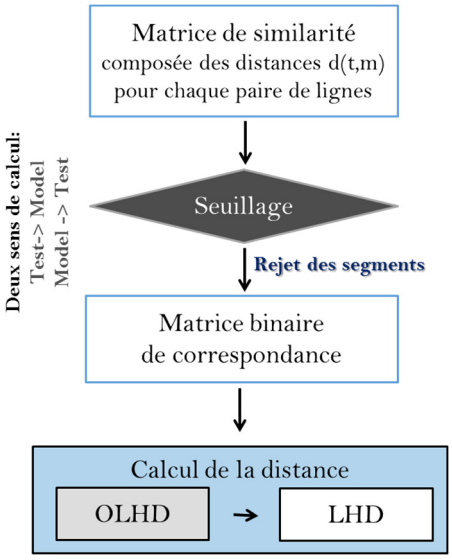

Figure 5.5 Calcul de distance entre segments : algorithme proposé

5.2.1 Distance de *Hausdorff*

En géométrie, la métrique de *Hausdorff* est un outil de détermination de la proximité de deux sous-ensembles de points dans un espace fini. Cette mesure de similarité naturelle est très utilisée pour le traitement d'image. Elle permet de qualifier les dissemblances entre les formes, de numériser une image ou bien encore de reconnaître une forme. Nombreux sont les chercheurs faisant appel à cette métrique par exemple [Huttenlocher *et al.*, 1993], [Grussenmeyer *et al.*, 1994], [Abbas *et al.*, 1995], [Choi *et al.*, 2001], [Gao et Leung, 2002], [Chen *et al.*, 2003], [Benhabiles *et al.*, 2009], [Wang et Tan, 2012], [Gao *et al.*, 2012]. Contrairement à la majorité des méthodes d'analyse de forme, la distance de *Hausdorff* ne requiert pas de mise en correspondance explicite des points $Test$ et $Model$.

Considérant deux ensembles de points $T = \{t_1, t_2, \dots, t_p\}$ et $M = \{m_1, m_2, \dots, m_q\}$, la distance de *Hausdorff* (H) est la distance maximale de deux quantités : $h(T,M)$ et $h(M,T)$ (5.5).

$$H(T,M) = max\big(h(T,M), h(M,T)\big) \tag{5.5}$$

Ces deux composants, nommés distances relatives de *Hausdorff*, n'ont quasiment jamais la même valeur (Figure 5.6). La distance de *Hausdorff* n'est pas alors symétrique ($h(T,M) \neq h(M,T)$). La première distance $h(T,M)$ est le maximum parmi toutes les distances $d(t,m)$, calculées entre chaque point d'un sous-ensemble T et un autre point le plus proche de l'ensemble M (5.6). La seconde $h(M,T)$, est définie pareillement, mais en direction de calcul opposée. Quant à $d(t,m)$, c'est une distance quelconque – généralement exprimée en tant que distance *Euclidienne*. Cependant, rien n'empêche l'utilisation des autres métriques telle que la distance de *Manhattan* par exemple.

$$h(T,M) = \max_{t \in T} \min_{m \in M} d(t,m) \tag{5.6}$$

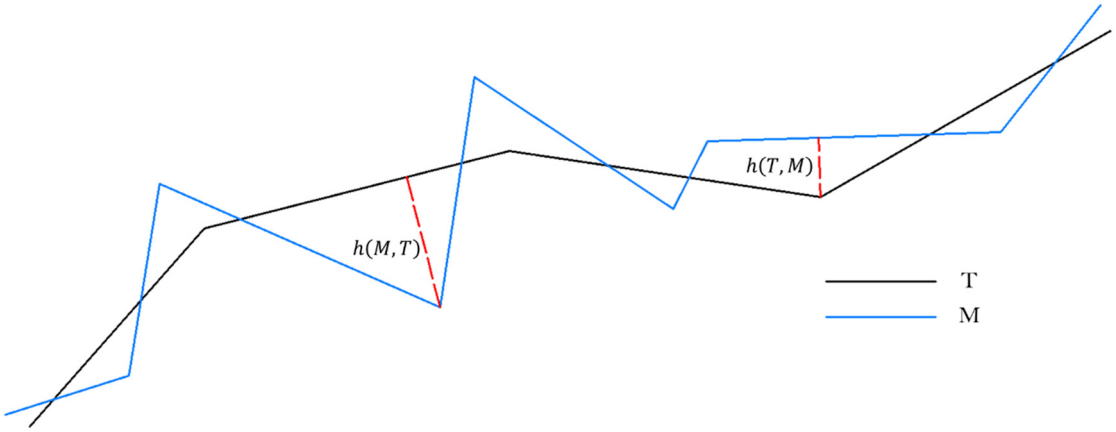

Figure 5.6 Définition de la distance de *Hausdorff* entre deux polygones

Nonobstant [Dubuisson et Jain, 1994] définissent 24 formes différentes de la distance de *Hausdorff*, dont une version modifiée *MHD* (*Modified Hausdorff Distance*). La principale différence réside dans la façon de définir la distance relative $h(T, M)$ qui sera désormais calculée comme suit :

$$h(T, M) = \frac{1}{N_t} \sum_{t \in T} d(t, M) \tag{5.7}$$

où :

N_t : nombre de points dans l'ensemble T

Il s'agit de la moyenne de toutes les distances minimales obtenues. Ainsi, chaque point de l'ensemble a un impact sur la valeur finale. Ce même constat ne peut s'observer lorsque la distance classique de *Hausdorff* est employée, même si les points sont assez proches les uns des autres. Cette définition non seulement ne néglige donc aucun des points, mais encore, selon les études complémentaires effectuées par [Benhabiles *et al.*, 2009], elle donne de meilleurs résultats que la définition de *Hausdorff* classique. En d'autres termes, la distance de *Hausdorff* définie par les équations (5.5) et (5.6) est censée être très sensible aux points aberrants. Par conséquent, quelques points aberrants, même un seul, peuvent considérablement perturber le calcul. Néanmoins, il faut souligner que sa valeur n'est pas une distance à proprement parler, car elle ne vérifie pas le principe d'inégalité triangulaire.

Line Hausdorff Distance comme mesure d'écartement entre deux jeux de segments

La définition de la distance modifiée de *Hausdorff* telle quelle n'est pas appropriée pour les besoins de comparaison des entités linéaires, puisqu'elle ne prend pas en compte l'orientation de celles-ci [Gao et Leung, 2002]. Un exemple de problèmes rencontrés est illustré par la Figure 5.7. Il en ressort très clairement que la distance renvoyée par la Figure 5.7a peut être identique à celle calculée entre les segments 3 et 4 (Figure 5.7b). Cependant, les droites sur la Figure 5.7a, dans la perception humaine, sont plus proches que ces deux dernières. De même, en analysant à l'aide de la distance de *Hausdorff* les segments présentés par la Figure 5.7c, nous comparerons l'extrémité de la partie la plus haute de 7 avec la ligne 5, tandis que celle la plus basse avec la ligne 6. Cette démarche n'est pas correcte puisque la droite 7 doit être intuitivement perçue comme une seule entité, et sa distance mesurée par rapport, soit au segment 5, soit au segment 6.

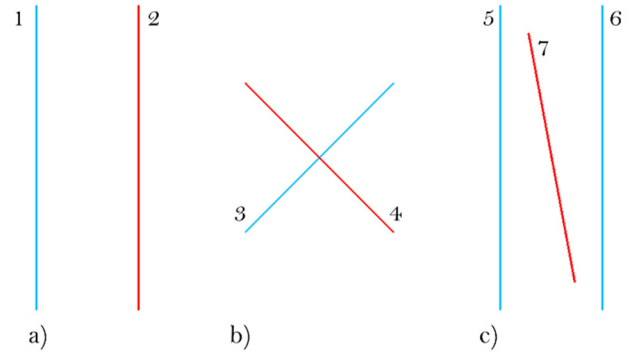

Figure 5.7 Exemple de problèmes liés à la distance modifiée de *Hausdorff* appliquée à des lignes (Image reproduite à partir de [Gao et Leung, 2002])

Afin de faire face à toutes ces contraintes [Gao et Leung, 2002] ont proposé, pour l'imagerie 2D, une extension de la distance introduite par [Dubuisson et Jain, 1994] applicable, cette fois-ci, à des droites. La distance directe $OLHD$ (*Oriented Line Hausdorff Distance*) se calcule de la manière suivante :

$$OLHD(T,M) = \frac{1}{\sum_{m_j \in M} L_{m_j}} \sum_{m_j \in M} L_{m_j} d(t_i, m_j) \tag{5.8}$$

où :

L_{m_j} : longueur du segment de l'ensemble *Model* (référence)

$d(t_i, m_j)$: distance entre deux segments

Une telle solution, fondée sur la moyenne pondérée, semble plus robuste lorsque les segments plus longs sont privilégiés.

Cette définition est devenue le point de départ pour notre approche de qualification basée sur les entités linéaires. Afin qu'elle soit capable de décrire l'exactitude et la précision de données étudiées, elle doit être modifiée. Ceci est dû au fait que nous voulons insister sur le fait que la distance de *Hausdorff* représente un chiffre aisé à interpréter, nous permettant de connaître l'écartement réel entre les segments examinés. Or, les distances directes $OLHD(T,M)$ et $OLHD(M,T)$ seront déterminées, selon la formule (5.9), à partir de segments couplés et ordonnés par la matrice de correspondance *Cor* que nous détaillons plus loin (paragraphe 5.2.3)

$$OLHD(T,M) = \frac{1}{\sum_{m_j \in M} L_{m_j}} \sum_{(i,j) \in Cor} L_{m_j} d(t_i, m_j) \tag{5.9}$$

5.2.2 Matrice de similarité

L'étape cruciale de l'algorithme de qualification proposé consiste à créer une matrice de similarité, dont chaque coefficient (i,j) indiquera un score (une distance) $d(t,m)$ attribué à une paire étudiée. L'objectif consiste à analyser toutes les combinaisons possibles de segments entre deux jeux. La matrice de similarité de taille (p,q), déterminée par le nombre d'éléments dans les ensembles respectifs $Test$ et $Model$, est obtenue selon la formule ci-dessous :

$$d(t_i, m_j) = \sqrt{W \cdot \left(d^{\alpha}(t_i, m_j)\right)^2 + \left(d^{II}(t_i, m_j)\right)^2 + \left(d^{\perp}(t_i, m_j)\right)^2} \tag{5.10}$$

Le score $d(t_i, m_j)$ étant la mesure de similitude, représente, en pratique, « l'effort » nécessaire pour superposer deux segments quelconques : l'un provenant de l'ensemble T et l'autre de M. Sa valeur intègre : la distance angulaire $d^{\alpha}(t, m)$, la distance parallèle $d^{II}(t, m)$ et la distance perpendiculaire $d^{\perp}(t, m)$. La constante W, étant choisie empiriquement, permet de mieux mettre en évidence l'éventuelle divergence d'orientation des droites. Plusieurs tests ont été réalisés pour choisir la valeur optimale de ce poids. Nous en ressortons que le poids W égale à 10 donne le meilleur résultat, c'est-à-dire, il facilite nettement l'analyse ultérieure de la matrice de similarité notamment lors de la recherche du seuil approprié. De plus, il n'affecte pas de manière importante la distance angulaire, puisque sa valeur, pour les droites quasiment parallèles, reste toujours pratiquement nulle.

La Figure 5.8 présente toutes les distances que nous définissons pour mesurer la dissemblance entre les segments 3D. La distance angulaire $d^{\alpha}(t, m)$ se calcule de la manière suivante :

$$d^{\alpha}(t_i, m_j) = min\left(||L_{t_i}||, ||L_{m_j}||\right) \cdot sin\alpha(t_i, m_j) \tag{5.11}$$

où

α : angle d'intersection de deux droites t_i et m_j ;

L_{t_i},L_{m_j} : longueur respective des segments t_i et m_j.

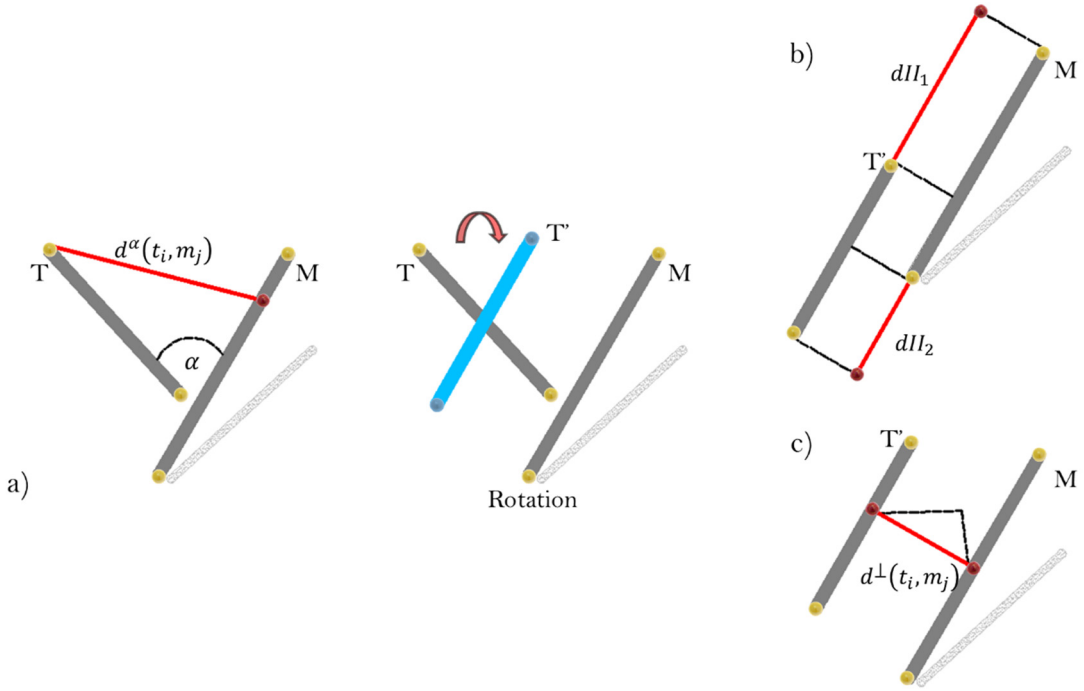

Figure 5.8 Mesure de déplacement de la droite : a) Distance angulaire ; b) Distance parallèle ; c) Distance perpendiculaire

Puisqu'en réalité, aucune paire de segments n'est jamais totalement parallèle, nous effectuons une rotation de chaque droite $Test$ autour de son milieu. Cette étape est indispensable pour obtenir une orientation cohérente des entités linéaires et pour calculer, par la suite, les deux distances manquantes : parallèle et perpendiculaire. Le calcul de la distance parallèle (Figure

5.8b) exige une étude des extrémités de deux segments. Nous assimilons ce problème à l'analyse des positions relatives de deux cercles circonscrits à ces segments (comme dans le chapitre précédent – voir la Figure 4.22). Leurs centres respectifs M_t et M_m (milieux de segments) ainsi que les rayons $R_t = L_t/2$ et $R_m = L_m/2$ seront étudiés. Plus précisément, il faut chercher si les deux cercles se croisent. Pour ce faire, le segment $Test$ est projeté sur la droite $Model$. Il n'y aura donc aucun point, ou alors un ou deux points d'intersection de cercles qui permettent de distinguer deux cas :

1) L'un des segments est complètement inclus dans l'autre, ou bien ils sont tangents intérieurement, alors $\|M_tM_m\| \leq |R_t - R_m|$. En conséquence, la distance parallèle $d^{II}(t_i, m_j)$ est nulle ;
2) Les segments se chevauchent partiellement ($|R_t - R_m < \|M_tM_m\| < R_t + R_m|$), ou ils sont disjoints ($|M_tM_m| > R_t + R_m$), ou bien encore ils sont tangents extérieurement ($|M_tM_m| = R_t + R_m$). Alors la distance parallèle $d^{II}(t_i, m_j) = min(dII_1, dII_2)$ (Figure 5.8c). Autrement dit, sa valeur correspondra au déplacement minimal nécessaire pour superposer soit les extrémités droites, soit les extrémités gauches de deux segments appariés.

En résumé, la distance parallèle se calcule selon la formule ci-dessous :

$$d^{II}(t_i, m_j) = \begin{cases} 0 & si\ t_i \subset m_j\ ou\ m_j \subset t_i \\ min(dII_1, dII_2) & autrement \end{cases} \tag{5.12}$$

Finalement, la distance perpendiculaire $d^{\perp}(t_i, m_j)$ est tout simplement la distance minimale entre deux droites parallèles D_t et D_m. Soit la droite D_t passant par M_t et de direction de vecteur $\overrightarrow{V_t}$, et la droite D_m passant par M_m et de direction $\overrightarrow{V_m}$. La distance est alors égale à :

$$d^{\perp}(t_i, m_j) = \frac{k}{\|\overrightarrow{V_t} \wedge \overrightarrow{V_m}\|} \tag{5.13}$$

où k est le produit mixte de trois vecteurs $[\overrightarrow{M_tM_m}, \overrightarrow{V_t}, \overrightarrow{V_m}]$ (déterminant de la matrice formée par les trois vecteurs).

5.2.3 Mise en correspondance

Dès que la matrice de similarité est créée (Figure 5.9a), nous procédons à l'analyse de celle-ci pour fixer un seuil δ, nécessaire dans le processus de rejet des paires erronées et de mise en correspondance des segments. Bien entendu, les correspondances recherchées ne seront quasiment jamais de 1 à 1, mais plutôt de 1 à plusieurs, de plusieurs à 1, voire de 1 à nul. Afin d'établir la valeur appropriée du seuil, nous commençons par un appariement préliminaire de segments S. Cette étape nous amène à apparier, à chaque droite de l'ensemble T, un homologue le plus semblable de M, c'est-à-dire, le segment le plus proche au sens de la distance $d(t, m)$. La tâche est aisée à remplir, puisque l'on peut toujours relier deux segments entre eux, même si leur score est significatif et même si la paire formée de cette manière n'existe pas en réalité (Figure 5.9b).

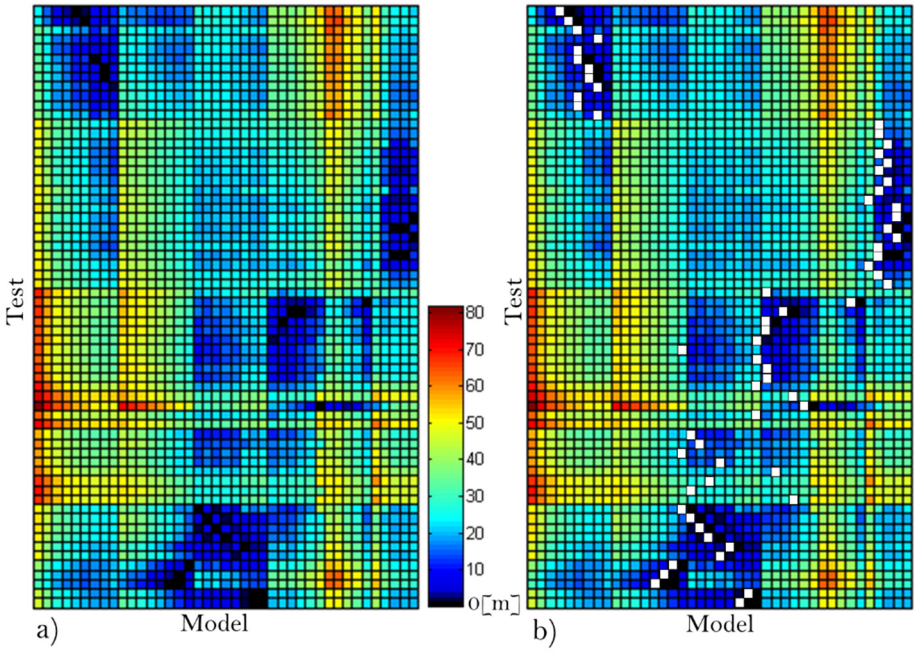

Figure 5.9 Estimation des correspondances : a) Matrice de similarité ; b) Appariement préliminaire S (carrés blancs)

Se fondant sur cet appariement grossier S, la recherche d'un seuil optimal se limite à l'analyse des différences entre les distances associées aux paires sélectionnées.

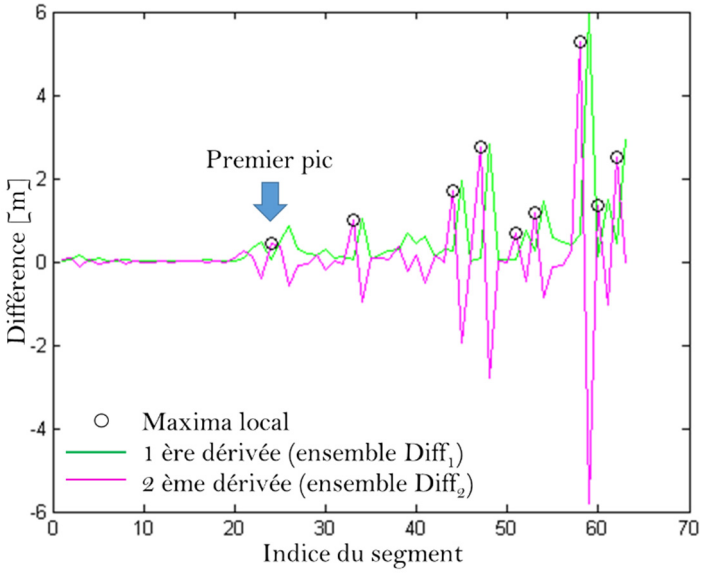

Figure 5.10 Recherche du seuil δ

La soustraction membre à membre des valeurs $d(t,m)$ d'une telle suite S est exécutée deux fois. Nous partons d'abord des distances classées selon un ordre ascendant ce qui amène à créer une nouvelle liste des valeurs Diff_1 - première dérivée (courbe verte sur la Figure 5.10). L'opération est effectuée encore une fois, mais cette fois-ci, pour les termes du suite Diff_1. Comme résultat, nous obtenons une courbe violette (Figure 5.10). Le premier pic (parmi les

valeurs $Diff_2$ – deuxième dérivée) ayant la hauteur supérieure à la sensibilité admise (U=0.5m) et détecté à l'aide d'une fonction localisant des maxima locaux, désigne l'indice du couple de segments dont le score sera admis comme le seuil de couplage δ. Cette procédure étant conçue pour les échantillons importants (nombre de paires N supérieur à 10), nous envisageons aussi une solution pour les cas ne remplissant pas cette condition. L'équation (5.14) résume la façon de calculer le seuil δ, quel que soit le nombre de paires traitées.

$$\delta = \begin{cases} max(S) & Card(Diff_1 < U) = N - 1 \\ médiane(S) + 2\sigma & si\ le\ maximum\ local\ n'existe\ pas \wedge N \leq 10 \\ le\ premier\ maximum\ local(Diff_2) \geq U & autrement \end{cases} \tag{5.14}$$

Dans le cas des échantillons petits, mais aussi ceux pour lesquels aucun maximum local n'existe, deux résolutions du problème sont envisagées :

1) Toutes les paires de l'appariement préliminaire S sont correctes et donc les différences successives $Diff_1$ sont négligeables (inférieures à U). Le seuil δ sera la valeur maximale parmi toutes les distances $d(t,m)$ ordonnées par la suite S ;
2) L'existence de paires aberrantes au sein de l'ensemble S est constatée et le seuil δ est fixé comme la médiane de celui-ci, agrandie ensuite par une tolérance de 2σ. Ainsi, toutes les valeurs inférieures ou égales à la médiane sont mises de côté, pour participer au calcul du σ représentant l'écart –type de leur moyenne.

Une fois le seuil retrouvé, la matrice de correspondance Cor est générée comme suit :

$$Cor = \{(i,j) \in \aleph^2 : \forall(t_i \in T, m_j \in M), d(t_i, m_j) \leq \delta\} \tag{5.15}$$

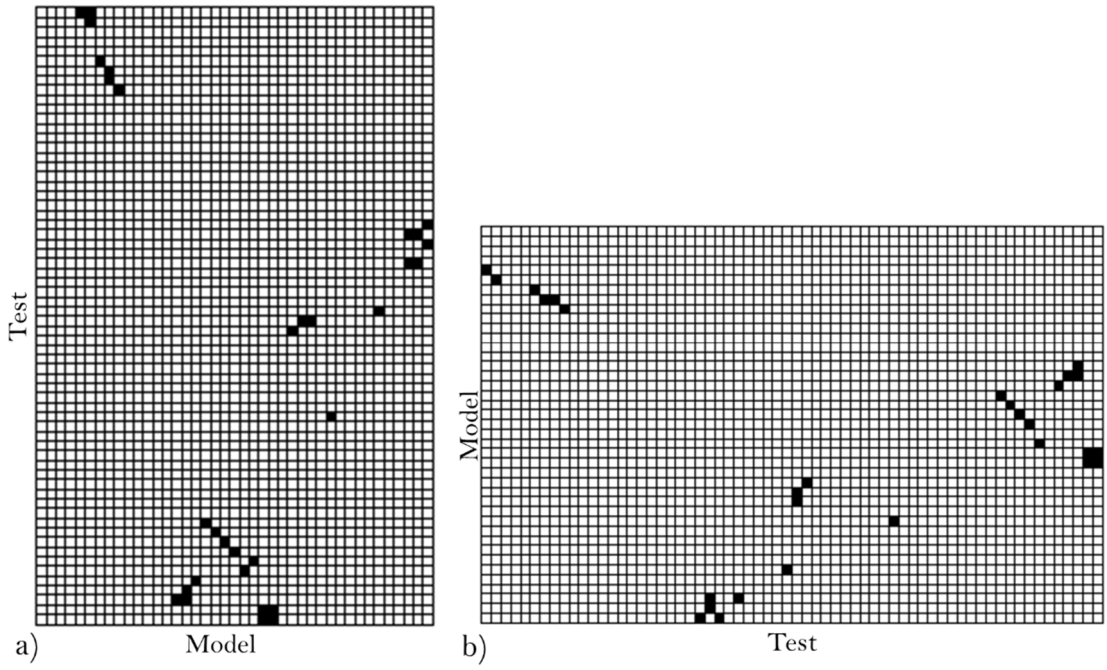

Figure 5.11 Matrice de correspondance : a) $Test \rightarrow Model$; b) $Model \rightarrow Test$

En d'autres termes, le seuillage de la matrice de similarité fournit un appariement définitif. Il convient de rappeler que la procédure décrite ci-dessus est menée dans deux sens, surtout pour des raisons de contrôle, mais aussi pour remplir les conditions imposées par la

définition de la métrique de *Hausdorff*. En effet, deux matrices de correspondance sont indépendamment déterminées, en supposant pour la seconde fois que l'ensemble $Test$ est la référence (Figure 5.11). Dès lors, les distances directes $OLHD(T,M)$ et $OLHD(M,T)$ peuvent être calculées entre les segments grâce à la formule (5.9). Quant à la distance indirecte (*LHD*), elle est obtenue à l'aide de l'équation (5.5).

5.3 Résultats et évaluation

Dans cette partie, nous validerons notre méthode de qualification des relevés laser mobile. Nous commencerons par vérifier l'algorithme de mise en correspondance des segments, puisqu'il influencera les résultats finaux (Paragraphe 5.3.2). Pour ce faire, les données réelles seront employées. Trois critères : l'exactitude, la sensibilité et la spécificité nous permettront de noter l'algorithme développé. L'appariement créé sera confronté avec la vérité-terrain. Ensuite nous compléterons les analyses effectuées par une étude de l'impact d'éloignement réciproque des nuages sur l'appariement des segments. En dernier lieu (Paragraphe 5.3.3), nous qualifierons les relevés laser, en termes d'exactitude absolue et de précision (répétabilité).

5.3.1 Données expérimentales

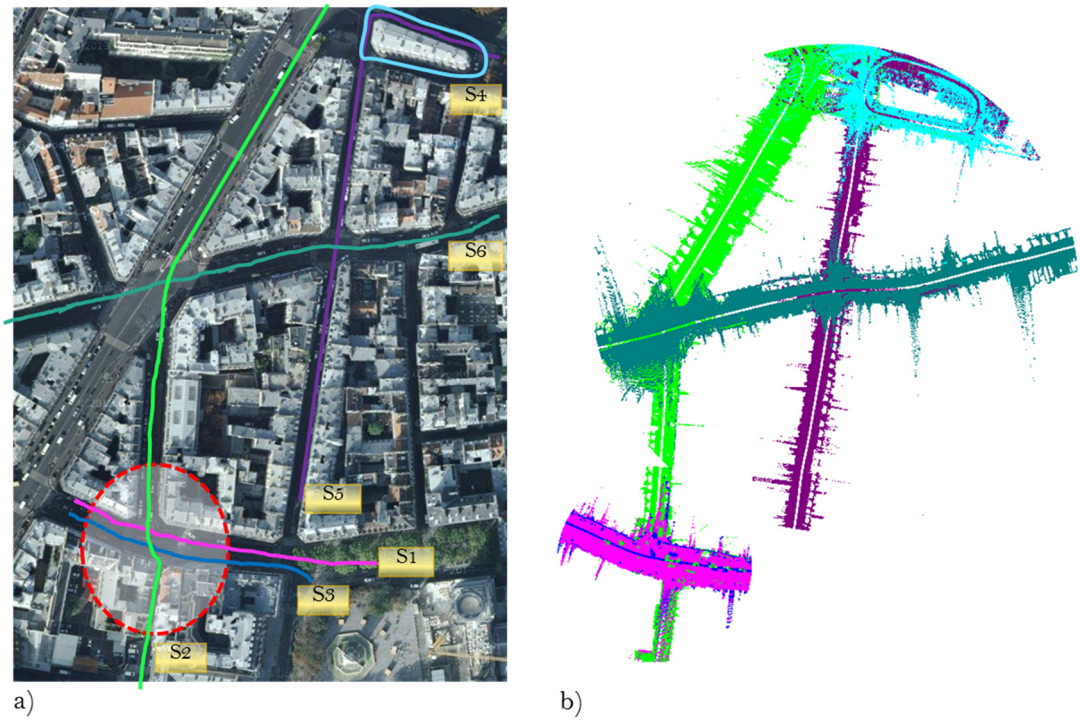

Figure 5.12 Zone de test : a) Prise de vue de la zone avec les trajets de la plate-forme superposés ; b) Vue de dessus des relevés laser mobiles acquis

La zone de test est située à Paris, à proximité du croisement des rues du Vieux-Colombier et Madame. Le choix de ce carrefour s'explique non seulement par la forte présence de façades orientées différemment (la multitude d'arêtes), mais aussi par sa diversité. Il contient des espaces « dégagés », ainsi que ceux considérés comme très problématiques du point de vue des acquisitions mobiles. La géométrie des rues (canyons urbains souvent à sens unique) et les bâtiments de hauteur importante rendent les conditions de réception du signal *GNSS* particulièrement difficile. Ceci nous a permis d'acquérir des données mobiles de qualité différente. Des relevés laser y ont été collectés par le système mobile Stéréopolis II d'*IGN*

dans la configuration décrite dans la section 2.1.3. Quelques passages de la plate-forme ont été réalisés dans ce périmètre comme on peut l'observer dans la Figure 5.12. En revanche, pour les besoins de l'analyse, seulement trois nuages de points S1, S2 et S3 ont été employés.

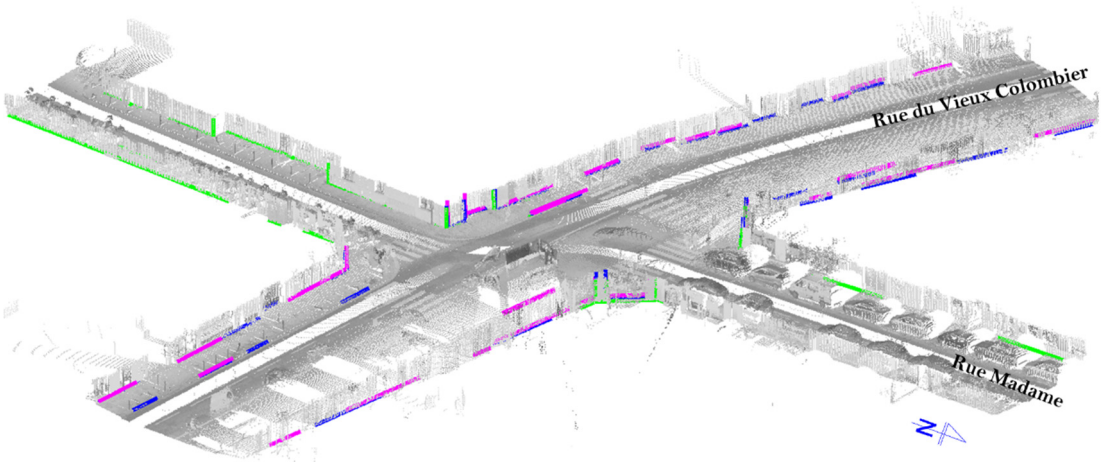

Figure 5.13 Segments extraits à partir des nuages mobiles (les couleurs de segments correspondent aux différents passages de la plate-forme - Figure 5.12

Les intersections entre les plans principaux (arêtes de pli) ont été retrouvées, constituant une base exhaustive à tester. Dans le cas des nuages mobiles, la géométrie de balayage (seulement le sol et le rez-de-chaussée de bâtiments constitué, en général, des devantures de magasins, ont été balayés) nous a obligés à extraire manuellement les arêtes (Figure 5.13). La détection précise de plans verticaux était impossible due au taux de bruit important. Quant à la référence, nous employons le nuage *Sulpice2013_statique* (voir la Figure 2.21 dans la Section 2.2) à partir duquel les arêtes (F) ont été extraites avec l'algorithme proposé au sein du Chapitre 4 (Figure 4.27).

5.3.2 Évaluation de l'algorithme d'appariement

Appariement des segments

Au premier abord, la mise en correspondance réalisée à travers l'approche développée a été comparée avec la vérité-terrain. De ce fait, la capacité de l'algorithme à prédire si une paire existe a été mesurée. À chaque fois, une matrice de confusion a été créée afin d'observer le nombre de paires de segments : 1) correctement appariées VP (Vrais Positifs) ; 2) correctement rejetées VN (Vrais Négatifs) ; 3) mal identifiées FP (Faux Positifs) ; 4) omises FN (Faux Négatifs). La Figure 5.14 visualise un exemple de résultats de la mise en correspondance des segments réalisée entre le nuage mobile (S3) et celui statique (F).

Sur cette base, trois critères servant à noter notre algorithme d'appariement, largement employés en statistique, ont été définis de la manière suivante :

$$Exactitude = (VP + VN)/(p \cdot q) \tag{5.16}$$

$$Sensibilité = VP/(VP + FN) \tag{5.17}$$

$$Spécificité = VN/(VN + FP) \tag{5.18}$$

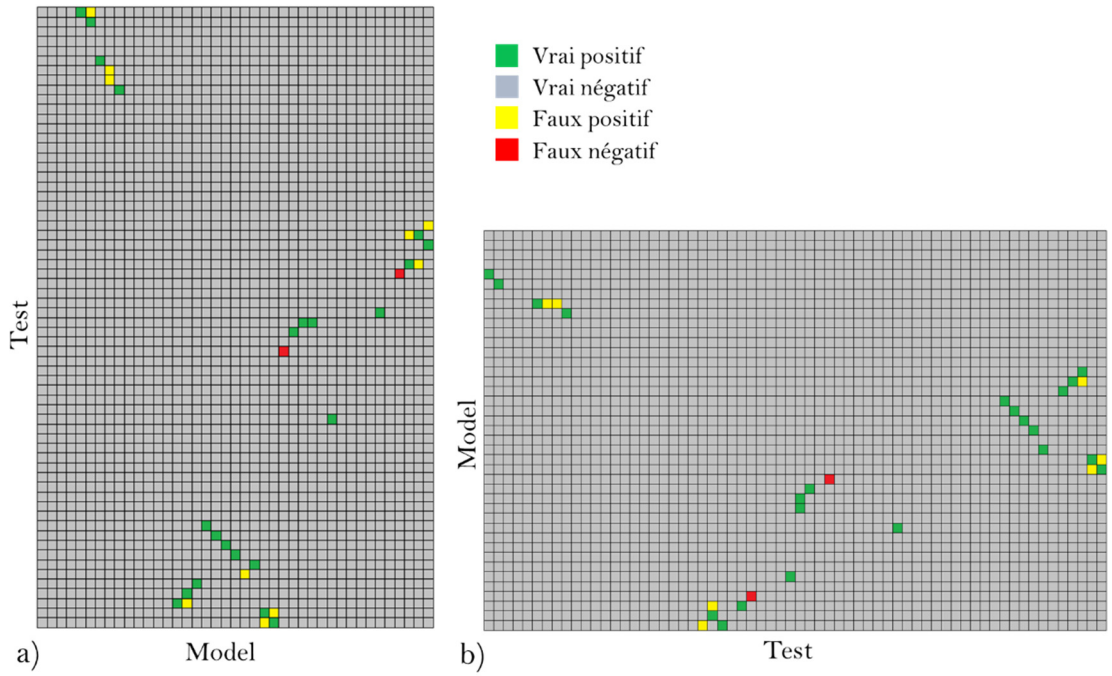

Figure 5.14 Comparaison avec la vérité-terrain : a) *Test* → *Model* ; b) *Model* → *Test*

Le Tableau 5.1 résume les valeurs obtenues pour les données réelles, séparément pour chaque direction de calcul, et accompagnées par la distance *OLHD*. Il en ressort que l'appariement réalisé entre les segments est suffisamment exact (les valeurs de tous les trois critères définis sont élevées).

Tableau 5.1 Validation de l'algorithme de mise en correspondance (T-*Test*, M-*Model*)

	Nombre de segments			Exactitude [%]		Sensibilité [%]		Spécificité [%]		OLHD [m]	
	Test (p)	*Model* (q)	N	T→M	M→T	T→M	M→T	T→M	M→T	T→M	M→T
S1-S2	36	28	11	99.8	99.8	100.0	91.7	99.9	99.8	0.518	0.472
S1-S3	36	41	29	99.5	99.5	93.1	93.1	99.7	99.7	0.302	0.297
S2-S3	28	41	12	99.7	99.7	91.7	91.7	99.8	99.8	0.481	0.560
S1-F	36	64	21	99.6	99.6	85.7	95.2	99.7	99.6	0.671	0.656
S2-F	28	64	22	99.3	98.9	90.9	90.9	99.4	99.0	1.104	1.257
S3-F	41	64	24	99.7	99.5	91.7	91.7	99.7	99.6	0.563	0.657
Moyenne				99.6		92.3		99.7			

La probabilité que l'algorithme dissocie correctement les vraies paires parmi toutes les combinaisons possibles (exactitude) est élevée et oscille autour de 99.6%. En moyenne, 92.3% des paires (voir la sensibilité) ont été retrouvées, indépendamment de l'éloignement de deux ensembles de droites, *Test* et *Model*. Quant à la probabilité de dénombrer les appariements erronés, elle peut être décrite comme faible. Puisque la spécificité atteint moyennement un niveau de 99.7%, le nombre de FP constitue 0.3% de tous les VN. Par contre, le taux de FP (détection d'une paire inexistante) paraît dépendre de la distance réciproque entre les segments de deux jeux traités. En réalité, cet impact sur l'écartement se traduit par une augmentation du nombre de droites couplées qui, dans le cas étudié, sont enrichies par 73%

de paires de plus pour une distance LHD égale à 1.257m. D'un point de vue pratique, les segments très proches de l'ensemble $Test$ seront appariés à un même élément de $Model$, car leur distance est suffisamment petite. Notamment, comme le montre la Figure 5.15a (à gauche), un grand décalage horizontal dP exerce une influence sur le seuil δ devenant plus grand. Les segments 17-18 et 15-16 seront *ipso facto* couplés avec les segments 47-48 et 5-6 respectivement, même si en réalité, leur homologue n'existe pas. Ceci est dû au fait que les distances $d(t,m)$ calculées sont inférieures au seuil fixé. Pour y remédier, nous pouvons imaginer deux solutions. La première consisterait à introduire une contrainte de couplage. Nous autorisons une correspondance unique entre deux segments. Néanmoins, le processus d'appariement nécessite que l'on affronte un phénomène de sur-segmentation. Les segments n'auront que très rarement une seule entité correspondante. Un autre défaut, une telle restriction dévoilera un accroissement du nombre de FN (non-détection d'une paire présente) et par conséquent, la probabilité d'identification correcte des paires existantes de segments (sensibilité) baissera. La seconde solution envisageable amènerait à rapprocher des segments avant de démarrer l'algorithme de mise en correspondance (voir le Chapitre 6).

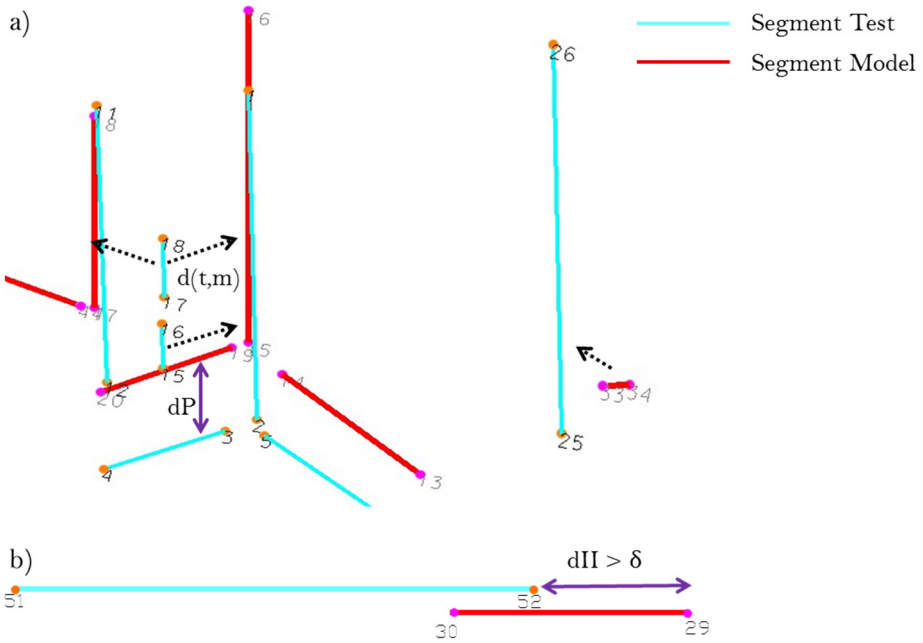

Figure 5.15 Exemple d'erreurs de mise en correspondance : a) FP ; b) FN

La Figure 5.15b illustre un autre problème noté lors du couplage d'entités linéaires. Le segment 51-52 n'a pas pu être apparié avec le segment 29-30, du fait de la distance parallèle d^{II} supérieure au seuil δ. Nous avons aussi observé que, dans certains cas particulier, le poids W introduit (voir la formule (5.10)) pour mettre en évidence l'orientation incompatible des droites peut être parfois insuffisant. Comme le présente la Figure 5.15a (à droite), le segment 33-34 a été apparié avec le segment 25-26 à cause de la distance $d^{\perp}$ très petite et de celle d^{II} étant nulle, par-devers les orientations différentes.

Impact de l'écartement entre les ensembles Test et Model

Après avoir effectué ces analyses, nous voyons que la mise en correspondance semble dépendre du décalage réciproque des segments. Plus les ensembles à comparer sont proches l'un de l'autre, plus le résultat sera précis.

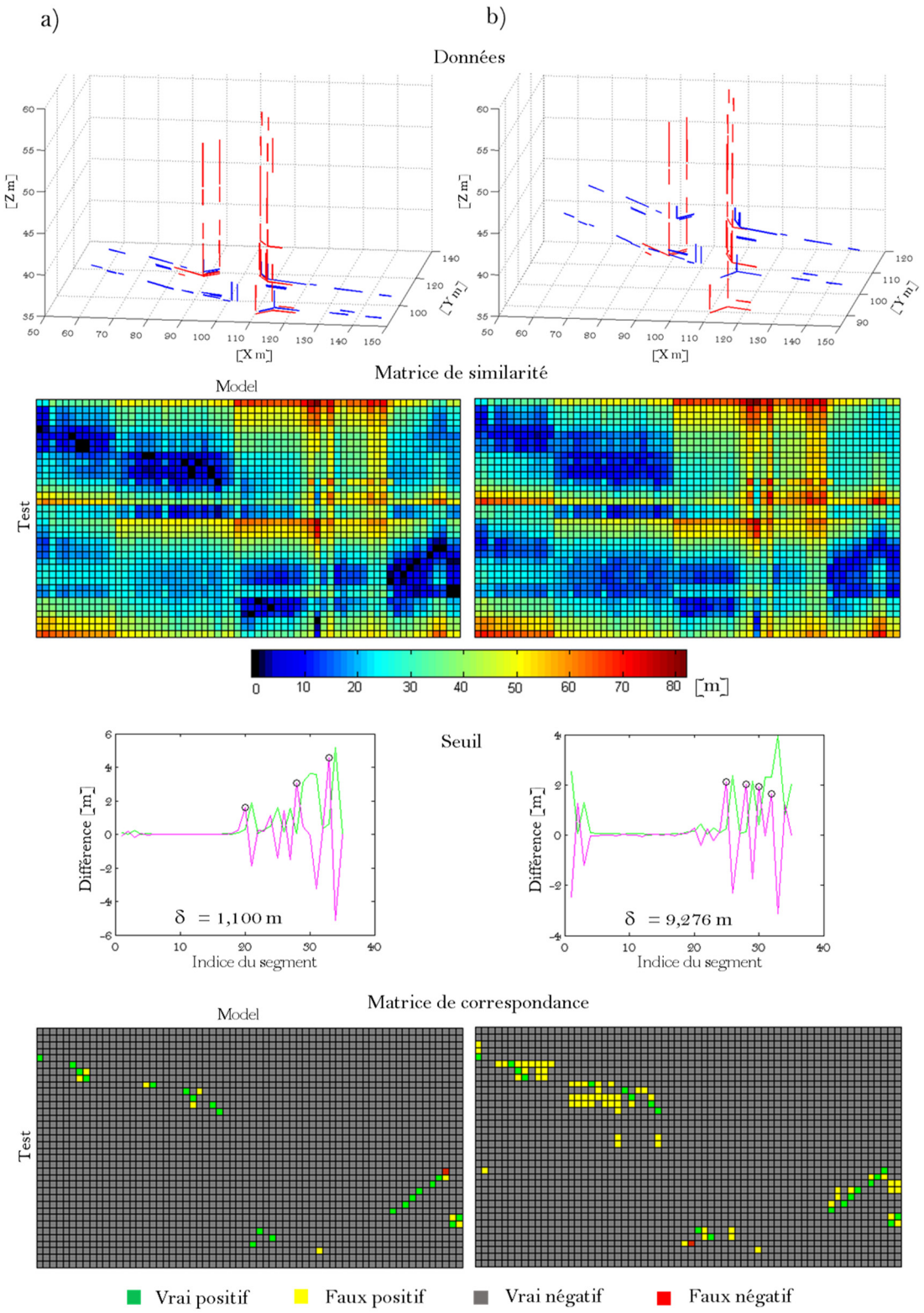

Figure 5.16 Exemple des résultats obtenus à partir de S1 et F (une seule direction de calcul) en fonction de la distance *LHD* : a) Distance faible ; b) Distance importante

Dans un premier temps, nous avons vérifié cette propriété à l'aide de données réelles S1 et F (originales, ensuite décalées). Deux cas sont illustrés : a) la distance réciproque LHD entre les ensembles est égale à 0.803m (Figure 5.16a) ; b) la distance est de 8.734m (Figure 5.16b). Nous observons, à cause de l'écartement notable, que les scores $d(t,m)$ attribués aux paires présentes sont à peu près de la même grandeur que ceux caractérisant les paires erronées. Dès lors, il est impossible d'entreprendre correctement une classification. Le seuil fixé est suffisamment important pour que les paires présentes soient détectées, par contre, il laisse également incorporer des paires erronées (FP).

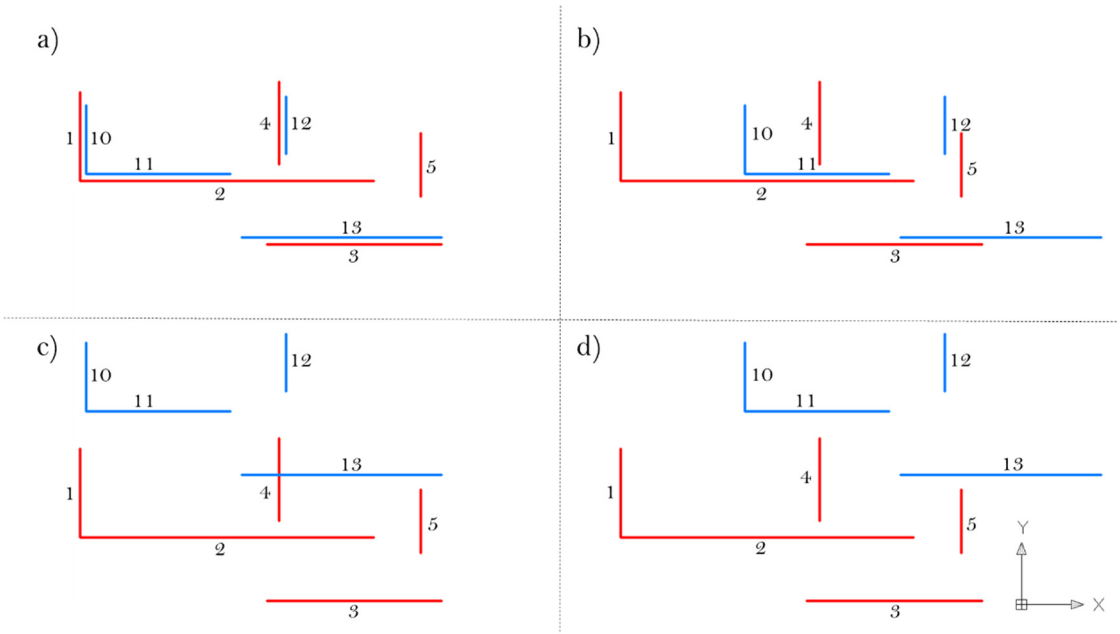

Figure 5.17 Problèmes d'appariement dus à l'écartement de segments *Test* (bleu) et *Model* (rouge) : a) Segments proches ; b) Segments *Test* déplacés T=[0.0m, 5.5m] ; c) Segments *Test* déplacés T=[5.5m 0.0m] ; d) Segments *Test* déplacés T=[5.5m, 5.5m]

Tableau 5.2 Matrices de similarité créées à partir des segments illustrés par la Figure 5.17 (sens de calcul $Test \rightarrow Model$)

a) δ=0.33m

0.28*	10.37	14.98	8.95	15.53
13.50	0.32*	8.96	13.39	15.30
9.53	8.82	10.09	0.32*	6.47
17.89	4.10	0.32*	12.72	9.98

b) δ=3,72m

	1	2	3	4	5
10	5.78	10.28	12.05	3.46*	10.05
11	15.88	3.19*	4.32	12.16	11.42
12	15.03	9.25	10.08	5.82	1.86*
13	21.98	9.04	4.34	14.96	9.45

c) δ=5.12m

	1	2	3	4	5
10	4.91*	12.60	17.55	9.97	16.82
11	13.72	5.82	12.09	13.47	15.63
12	10.91	11.68	13.88	4.83*	9.53
13	17.76	4.28*	5.82	12.46	9.98

d) δ=7.09m

	1	2	3	4	5
10	7.58	12.53	15.13	5.60*	11.94
11	16.07	5.82*	9.20	12.25	11.87
12	15.94	12.01	13.87	7.56	7.24
13	21.88	9.13	7.26	14.74	9.45

L'origine de ce problème réside dans la matrice de similarité elle-même dont l'analyse, en présence du décalage significatif, ne permet plus de restituer de vraies relations entre les segments. Par l'intermédiaire du Tableau 5.2 (matrices de similarité) associé à la Figure 5.17, nous fournissons un exemple justifiant nos conclusions. Pour simplifier notre expérience, nous n'envisageons alors que le cas 2D.

Le déploiement des segments sur la Figure 5.17a permet, sans aucun doute, de constater (visuellement) que les entités linéaires doivent être appariées de la manière suivante : 10-1, 11-2, 12-4, 13-3, nul-5. Aussi, les scores $d(t,m)$ attribués à ces paires par notre algorithme sont moindres et le seuil δ=0.33m dissocie les paires de la même manière. Nonobstant, cette certitude disparaît quand les segments sont de plus en plus éloignés (Figure 5.17b,c,d). Nous les relions désormais d'une autre façon en tenant compte de la contrainte de choisir les paires ayant la distance réciproque la plus petite, et les directions conformes. Les carrés gris du Tableau 5.2 représentent l'appariement préliminaire S, sur lequel notre algorithme d'appariement va s'appuyer afin de calculer le seuil δ. Les paires retenues après le seuillage sont marquées par un astérisque (*). Par conséquent, le résultat de mise en correspondance effectuée par notre algorithme est suivant :

1) Figure 5.17a :VP=4, VN=16, FP=0, FN=0 ;
2) Figure 5.17b :VP=1, VN=14, FP=2, FN=3 ;
3) Figure 5.17c : VP=2, VN=15, FP=1, FN=2 ;
4) Figure 5.17d : VP=1, VN=14, FP=1, FN=3 ;

Au vu de ces résultats, nous constatons que l'algorithme fournit l'appariement adéquat à l'emplacement actuel des segments. En souhaitant trouver des segments homologues parmi deux ensembles éloignés, il faut se rendre compte que cette tâche peut être irréalisable. Néanmoins, pour nos applications, c'est-à-dire l'appariement des segments lors de leur qualification, cette remarque n'est pas trop pertinente. Premièrement, les nuages de points ainsi que les données de référence sont géopositionnés dans un référentiel commun. En second lieu, l'exactitude des systèmes mobiles récents est loin d'être métrique. C'est pourquoi la contrainte de distance réciproque faible entre deux ensembles d'arêtes sera toujours satisfaite. La mise en correspondance peut ainsi être précisément effectuée. Sinon, faute de précision requise, il sera nécessaire de rapprocher des données avant d'exécuter l'algorithme de couplage.

5.3.3 Exactitude et précision

Passons à présent à la qualification des relevés laser mobiles à l'aide des arêtes extraites. Nous employons deux critères :

1) l'exactitude absolue calculée par rapport à la référence précise (données statiques) ;
2) la précision consistant à comparer plusieurs passages de la plate-forme mobile (S1-S2, S1-S3, S2-S3).

Le Tableau 5.1 récapitule les distances relatives de *Hausdorff* ($OLHD$) déterminées pour les deux sens de calcul : en comparant d'abord les entités linéaires $Test$ (S1, S2, S3) entre elles – mêmes, et ensuite avec celles de référence (F). À chaque fois, la valeur maximale LHD décrit, selon le cas, soit l'exactitude, soit la précision des données étudiées. Les tests effectués montrent que, dans les zones urbaines, la précision du système mobile examiné est de 0.30m. Cependant, une précision plus faible a été constatée lors de comparaison avec le passage S2. Or, ce nuage a été très probablement affecté par une mauvaise qualité des mesures *GNSS*. Cette hypothèse a pu être vérifiée par l'exactitude dont la valeur était égale à 1.257 m. En revanche, dans les autres cas, l'exactitude absolue est d'environ 0.70m. Le Tableau 5.3 récapitule les valeurs de distance indirecte de *Hausdorff* (LHD) retenues et décrivant la qualité des données traitées. Aussi, le temps d'exécution *CPU* sous *Matlab* est renseigné.

Même si l'algorithme exige d'être encore optimisé, le temps de calcul est de l'ordre de quelques secondes.

Tableau 5.3 Mesure de qualité :distance de *Hausdorff* retenue

	Nombre de segments		N	LHD[m]	CPU* [s]
	Test (p)	*Model* (q)			
S1-S2	36	28	11	0.518	1.39
S1-S3	36	41	29	0.302	1.99
S2-S3	28	41	12	0.560	1.54
S1-F	36	64	21	0.671	3.11
S2-F	28	64	22	1.257	2.36
S3-F	41	64	24	0.657	3.51

* Intel Core x2 T7600 2.33Ghz / 4Go Ram / Win7 64bits

Il convient aussi de souligner, que la distance moyenne évaluant la qualité des nuages à partir des segments non seulement représente l'exactitude, mais aussi l'erreur survenue lors de la procédure d'extraction d'arêtes. Cette dernière est propre à l'algorithme employé. Quant à la distance de *Hausdorff* LHD, ses autres propriétés seront encore abordées au Chapitre suivant.

Notre algorithme de qualification est décrit par le pseudo-code 5.1(Annexe E). Son but est de fournir une mesure de qualité LHD, ainsi qu'une matrice binaire de correspondance Cor entre deux ensembles de segments. Nous obtenons également deux distances relatives de *Hausdorff* : $OLHD_{TM}$ et $OLHD_{MT}$, une pour chaque direction de calcul. Nous nous en servons pour contrôler la justesse de l'algorithme d'appariement. Si leurs valeurs diffèrent significativement, il en résulte que la mise en correspondance n'est pas correcte. Il est donc recommandé de changer, dans un premier temps, la sensibilité U de la fonction de recherche des maxima locaux (par défaut $U = 0.5$) puisqu'elle décide du seuil δ. L'algorithme attend en entrée que l'on saisisse une liste de coordonnées des extrémités de segments X_{Test} et X_{Model} dans le format $[Nr, X, Y, Z]$.

CHAPITRE 6

Recalage à partir d'entités linéaires

Sommaire :

6.1 Introduction **108**
6.2 État de l'art du recalage rigide **109**
6.2.1 Mise en correspondance automatique 109
6.2.2 Classification des méthodes existantes de recalage 111
6.2.3 Aperçu des méthodes basées sur les entités linéaires 113
6.3 Recalage basé-lignes : caractérisation des méthodes **115**
6.4 Évaluation des méthodes présentées **116**
6.4.1 Données pour les expérimentations 117
6.4.2 Algorithmes implémentés 118
6.4.3 Outils d'évaluation 118
6.4.4 Résultats – recalage des données bruitées 119
6.4.5 Impact du bruit sur la distance de *Hausdorff* 122
6.5 Approche proposée : *RLMR-FMII2* **124**
6.5.1 Recalage grossier – principe 126
6.5.2 Évaluation avec données simulées 127
6.5.3 Tests sur données réelles 130

Résumé

Dans ce chapitre un état de l'art de techniques de recalage rigide basé sur les entités linéaires est effectué. Trois méthodes d'estimation sont choisies et examinées avec des données simulées, afin de vérifier leur robustesse et leur précision vis-à-vis notre problématique. Au vu des analyses effectuées, l'une des méthodes existantes est adaptée à nos besoins, puis renforcée d'un algorithme permettant parallèlement la mise en correspondance. Sachant que notre technique d'appariement impose une contrainte – la distance réciproque des segments doit être assez faible, nous procédons d'abord à rapprocher des arêtes pour faciliter et améliorer leur couplage ultérieur. Nous proposons une chaîne complète nommée $RLMR - FMII_2$ *comprenant le recalage grossier (utilisation complémentaire de l'algorithme de RANSAC et celui de* $FMII_2$*) et un recalage fin (raffinement des segments approchés par le* $FMII_2$*). Finalement, la performance d'une telle procédure est évaluée avec des données simulées et réelles.*

6.1 Introduction

La consolidation des nuages de points a pour but d'assembler des données devant servir à la reconstruction tridimensionnelle d'environnements et de base à des études géométriques. L'existence de zones d'ombre dues à la complexité des formes numérisées ainsi qu'aux artefacts (voitures, piétons ou autres obstacles) présents sur la zone durant l'acquisition se manifeste par un accroissement du nombre de stations du scanner laser fixe – les poses [Landes et al., 2011]. Lorsqu'il s'agit d'un robot mobile se mouvant, les nuages de points consécutifs contiennent toujours une même partie de l'environnement, mais avec des points de vue différents. Ces données doivent être mises en cohérence pour en déduire le déplacement relatif. Aussi, pour les zones d'intérêts numérisées deux ou plusieurs fois, au fil du temps, la question de la consolidation se pose. D'autre part, pour des plates-formes aériennes ou terrestres se localisant grâce à des systèmes hybrides *IMU/GNSS*, nous observons un biais entre les nuages de points distincts. Alors, dans le cas de la cartographie terrestre des zones de taille allongée (cartographie des corridors) les pertes du signal *GNSS* apparaissent sous la forme d'un décalage de la trajectoire estimée. Enfin, tous les relevés acquis par scanner laser aérien *ALS* sont encombrés par des écarts planimétriques et altimétriques causés, cette fois-ci, par la dérive de la centrale.

Quel que soit le moyen de collecter les données, tous les relevés laser doivent être affinés dans un système de référence commun, autrement dit consolidés (recalés). La solution au problème de calage se réduit à déterminer une bonne transformation géométrique (rigide ou non rigide), applicable aux données. Le référentiel commun est arbitrairement choisi parmi l'un des nuages de points. Par essence, le recalage rallie trois problématiques : 1) la reconstruction des primitives facilement identifiables dans la scène balayée ; 2) la mise en correspondance et la planification de la stratégie des manipulations sur des entités couplées ; 3) la résolution du système de contraintes amenant à estimer les paramètres optimaux de passage d'un repère à un autre – généralement un vecteur de translation $T = [T_x, T_y, T_z]$ et une matrice de rotation R_{3x3}.

Nous nous concentrons, dans le cadre de notre travail, sur le recalage rigide de relevés laser issus du système mobile de cartographie. Deux tâches principales sont à réaliser. D'une part, l'intégration « multidonnées » visera, à améliorer le géo-référencement absolu, ainsi qu'à incorporer d'autres types de données, externes au système d'acquisition, mais fournissant souvent des informations complémentaires. Cependant, le principe du co-recalage de relevés sera de rechercher à assembler, de manière relative, ces différents nuages acquis lors de plusieurs passages. Son inconvénient principal réside dans le fait qu'aucune information extérieure ne corrige le cumul des erreurs, si bien que l'exactitude du calcul final ne peut pas être calculée. Rappelons que les entités géométriques choisies sont des segments de droite (arêtes de pli), car les nuages de points bruts (sans traitement), avec lesquels nous travaillons, peuvent empêcher la reconnaissance précise des points de contrôle, même sous forme de cibles placées volontairement dans la scène.

Dans ce chapitre, on s'attardera, dans un premier temps (Section 6.2), sur un état de l'art des techniques de recalage et d'appariement des entités linéaires. Puis (Section 6.3), trois approches d'estimation des paramètres de pose à partir des lignes 3D seront présentées en détail notamment : *EIGEN* [Zhang et Faugeras, 1991], *ICL* (forme *ICP*) [Alshawa, 2006] et *FMII* [Kamgar-Parsi et Kamgar-Parsi, 2004]. Ces algorithmes ont été implémentés et examinés avec des données simulées afin de vérifier leur robustesse, la résistance au bruit et l'exactitude des résultats obtenus vis-à-vis de notre problématique (Section 6.4). C'est sur cette base qu'une chaîne complète, nommée $RLMR - FMII_2$ *(Robust Line Matching and Registration* – FMII_2*)* permettant un recalage qui s'appuie sur l'appariement des primitives linéaires, a été proposé (Section 6.5). La mise en correspondance s'effectue à l'aide de l'algorithme présenté dans le Chapitre 5. Néanmoins, cette méthode requiert que les deux

ensembles de segments à apparier soient relativement proches l'un de l'autre. Sinon, l'appariement n'est qu'approximatif. Nous proposons, par conséquent, de rapprocher d'abord les entités linéaires dans l'intention d'améliorer et de faciliter leur couplage ultérieur. Cette étape étant réalisée par une approche couplant *RANSAC* et $FMII_2$ amène à trouver un échantillon optimal à partir duquel les paramètres de recalage grossier seront calculés (Paragraphe 6.5.1). Ensuite, la mise en correspondance explicite des segments sert à un recalage fin, possible grâce à l'algorithme $FMII_2$ étant la version modifiée de *FMII*. Nous présenterons, à la fin du chapitre, l'analyse qualitative et quantitative des résultats obtenus avec une telle procédure.

6.2 État de l'art du recalage rigide

Dans cette partie, le premier paragraphe (6.2.1) fera un bref état de l'art de divers algorithmes d'appariements des entités linéaires. Puis, nous passerons à présenter une classification des méthodes de recalage (6.2.2), pour finalement décrire quelques techniques de minimisation des contraintes à partir des lignes, visant à déterminer les paramètres de transformation rigide (6.2.3).

6.2.1 Mise en correspondance automatique

La raison principale pour laquelle le recalage est difficile vient du fait que les appariements, c'est-à-dire les vraies associations entre les primitives extraites, sont inconnus. La mise en correspondance de plusieurs éléments, reconstruits depuis des nuages de points distincts, est une tâche cruciale puisqu'elle établit un système de contraintes à minimiser lors de l'estimation des paramètres de pose (Figure 6.1). La convergence de l'algorithme de recalage dépendra alors directement de ces associations. En d'autres termes, l'exactitude des appariements créés, se manifestant par un nombre de paires erronées ou manquées, influencera la qualité des paramètres déterminés.

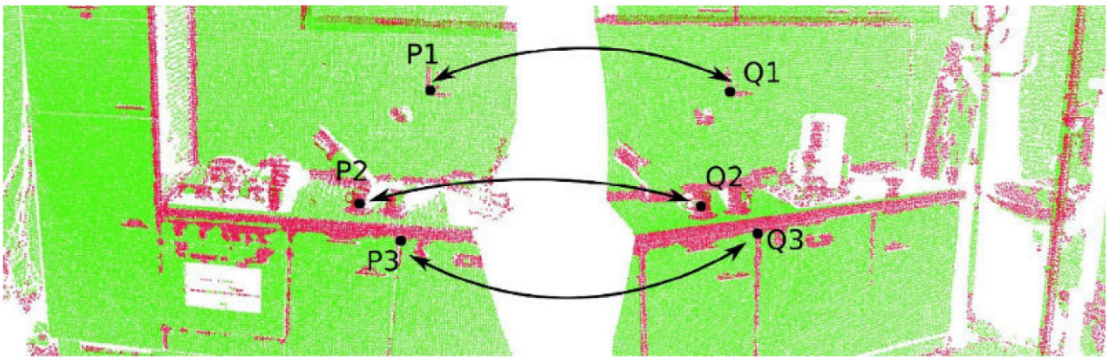

Figure 6.1 Exemple de mise en correspondance entre deux nuages de points à partir de descripteurs [Latulippe, 2013]

Considérant une technique itérative avec une présence de mauvais appariements, dans le meilleur des cas, on ralentit sa convergence, et dans le pire de cas, on cause sa divergence. Par conséquent, avec un algorithme comme *ICP (Iterative Closest Point)* (expliqué en 6.2.2), on n'est jamais certain d'arriver à une solution optimale, car ces associations sont seulement approximatives. Ainsi, lorsque l'écartement entre deux nuages est grand au départ, cet algorithme reste facilement bloqué dans un minimum local. Pour remédier à ce problème, l'une des idées serait de développer des méthodes permettant d'estimer des correspondances avant de réaliser le recalage global. Ce résultat peut être atteint, comme expliqué par [Latulippe, 2013], à l'aide d'un descripteur, à savoir un ensemble de valeurs caractéristiques, attribué à un point et décrivant la géométrie locale autour de lui. Il existe de nombreuses approches qui incorporent une étape d'extraction de points d'intérêt, auxquels sont ensuite

attribuées les valeurs de description. Ces dernières sont calculées à partir des coordonnées (X,Y,Z) de ce point et de ses voisins. Nous pouvons citer, par ordre chronologique, quelques types de descripteurs, applicables aux nuages de points par exemple : la variation de la courbure ; les moments invariants ; les spin images ; les points fingerprints ; les shape contexts 3D ; le *SIFT* (*Scale-Invariant Feature Transform*) ; les *PFH* (*Point Feature Histograms*) ; les *FPFH* (*Fast Point Feature Histograms*) ; les SHOT (*Signature of Histograms and Orientations*), etc. [Tangelder et Veltkamp, 2008] et [Latulippe, 2013].

Mise en correspondance des segments de ligne

Nous nous intéressons à réaliser le recalage par l'intermédiaire des entités linéaires 3D appariées au préalable. Néanmoins, les travaux traitant de ce problème n'exposent presque jamais la question de la mise correspondance, en supposant que celle-ci soit déjà établie. Pour satisfaire ces besoins, un nombre de techniques d'appariement a été développé. La plupart des revues liées à ce sujet dévoilent que les algorithmes utilisés se heurtent aux difficultés suivantes : la sur-segmentation des segments et le bruit. Nous détaillerons dans cette section quelques méthodes récentes d'estimation de correspondances. Une étude montre que la distance *Euclidienne* « classique » peut toutefois ne pas être suffisante pour créer un appariement relativement correct [Douadi, 2006]. Par conséquent, [Wang *et al.*, 2012] suggèrent de prendre en compte d'autres attributs de segments tels que la longueur, la direction, la position ou le taux de chevauchement.

[Yao *et al.*, 2010] proposent un algorithme bénéficiant de rigidité des droites pendant la transformation, ce qui veut dire que les écartements et les angles entre les entités restent toujours les mêmes. Pour chaque paire de segments de l'ensemble A, des paires homologues sont recherchées dans l'ensemble B qui satisfont à deux conditions : les valeurs absolues de différence de distances et celles d'angles sont inférieures à des seuils fixés. Afin de réduire le temps de recherche, seulement les k meilleures paires de B sont retenues.

De la même manière, [Li *et al.*, 2012] appliquent la même hypothèse afin d'établir de vraies associations entre les segments 3D. La solution proposée consiste à attribuer, à toutes les entités linéaires de l'ensemble, deux vecteurs, chacun comprenant quatre composantes décrivant leur relation angulaire et l'éloignement vis-à-vis des autres éléments de l'ensemble. Il s'agit de former le vecteur contenant : le maximum, le minimum, la moyenne et l'écart-type de mesures calculées. Ensuite, on détermine la distance entre *i-ème* vecteur de l'ensemble A et tous les vecteurs de B. Les segments sont appariés à condition que les deux normes (concernant l'angle et la distance) soient minimales.

[Jaw et Chuang, 2008] développent une approche s'exécutant en deux étapes. Dans un premier temps, les angles spatiaux (θ_{ij}, θ_{ik} ...) entre un segment et ses voisins successifs dans chaque ensemble sont déterminés en utilisant, à cet effet, les extrémités (Figure 6.2a et Figure 6.2b). Deux segments de droites sont appariés, si et seulement si, la différence angulaire ne dépasse pas un certain seuil. Ces associations préalables permettent ensuite d'estimer les paramètres de transformation censés rapprocher les nuages de points (Figure 6.2c). La dernière phase nous amène à calculer la distance spatiale D équivalente à la moyenne des distances (D_1 et D_2) servant à retrouver des lignes homologues (Figure 6.2d).

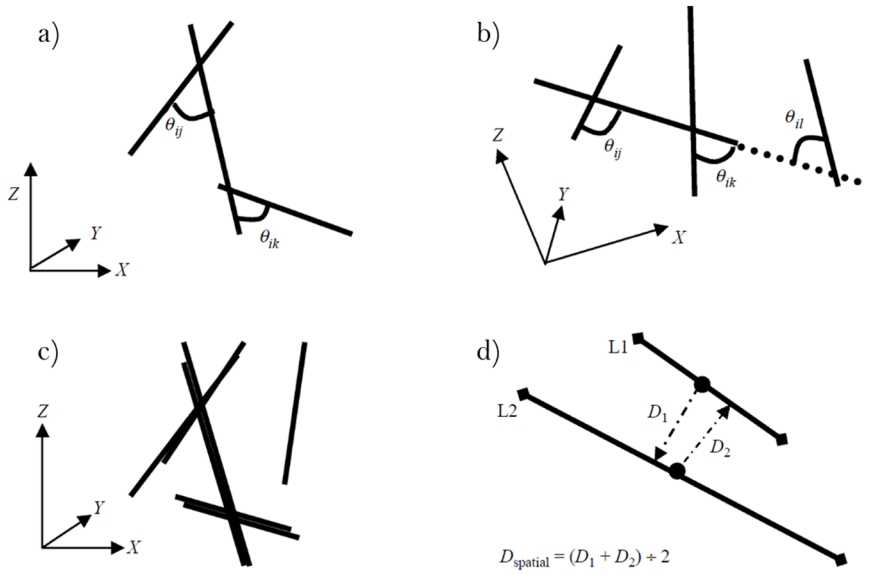

Figure 6.2 Estimation de correspondances selon [Jaw et Chuang., 2008] : a) b) Calcul des angles spatiaux pour l'ensemble A et B respectivement ; c) Deux ensembles d'entités après la transformation ; d) Définition de distance spatiale

Un regard tout à fait différent est énoncé avec des approches cherchant à apparier des points au lieu des lignes. L'une des possibilités est de discrétiser chaque droite en un certain nombre de points et de s'en servir pendant le couplage réalisé souvent à l'aide d'*ICP*. Ainsi [Belton *et al.*, 2011] proposent d'extraire des vertex représentant des intersections entre les lignes 2D. La technique mise en œuvre consiste à étudier, au sein de chaque ensemble, les distances *Euclidiennes* entre toutes les combinaisons possibles de points. Pour ce faire, une méthode de vote est employée dont l'objectif est de trouver, sous une certaine tolérance, des distances semblables. Pendant ce processus une matrice de correspondance est créée, de telle manière que ses coefficients décrivent la probabilité que le point p_i de l'ensemble A appartienne au point p_j de l'ensemble B.

De surcroît [Huh *et al.*, 2013] cherchent à apparier des polygones. Cette problématique est similaire à la nôtre puisqu'un polygone est considéré comme la réunion de segments appelés côtés. D'abord, les calculs des régions de confiance des segments sont effectués à partir des erreurs de position de leurs extrémités. Puis, des sommets ainsi que des « vertex virtuels » étant l'intersection des droites dérivées depuis les segments, sont extraits. Enfin, un algorithme de recherche de sous-chaîne (*string matching algorithm*) est employé afin d'estimer des paires de points correspondants.

6.2.2 Classification des méthodes existantes de recalage

Il existe plusieurs techniques de calage de données laser. Nous pouvons l'effectuer par consolidation (*best fit/ICP*) en nous appuyant sur la géométrie des objets numérisés, par un relèvement sur les cibles disposées (Figure 6.3) ou bien encore sur les « cibles naturelles » en les assimilant à des primitives géométriques (sphères, plans, cylindres, cônes, etc.) fortement présentes dans les environnements bâtis et industriels aménagés par l'homme. Quant aux cibles/primitives, elles peuvent servir également pour dissocier des entités géométriques (le centre, l'axe principal, le sommet, l'intersection, etc.), ainsi que des invariants géométriques tels que le rayon [Hullo, 2013].

Figure 6.3 Différents types de cibles propres pour les constructeurs du scanner

La classification des méthodes de recalage proposée par [Gressin *et al.*, 2013] peut être une bonne démarche introductive pour les algorithmes existants. En effet, trois familles d'approches sont considérées : 1) les méthodes basées sur la mise en correspondance de primitives géométriques remarquables dans deux nuages et invariants lors d'un mouvement rigide ; 2) les approches employant le modèle de surface (ordinairement un maillage) au lieu des nuages de points 3D ; 3) les techniques focalisées sur les points et donc ne nécessitant ni l'extraction préalable de primitives ni la phase de modélisation (p.ex. *ICP*, *RANSAC*). Néanmoins, cette classification n'est pas unique. [Alshawa, 2006] suggère de classer des algorithmes de la manière suivante : 1) les méthodes basées sur les entités géométriques, dont la connaissance préalable et précise, est primordiale ; et 2) celles indépendantes des entités d'intérêt. Parmi ces dernières deux sous-groupes sont distingués : 1) les techniques de vote ; 2) les méthodes de correspondance sous-jacente telles que la méthode *ICP* et la méthode *DARCES* (*Data-Aligned Rigidity-Constrained Exhaustive Search*) [Chen *et al.*, 1999] basée sur *RANSAC*. Il faut souligner que les algorithmes dépendants des primitives nécessitent un temps considérable pour les extraire et que leur handicap se dévoile lors du manque de détails caractéristiques. En revanche, ils n'exigent quasiment pas de valeurs initiales pour se réinitialiser en permettant de réduire le nombre de combinaisons possibles d'appariement. Quant aux calculs des inconnus de pose, ils sont effectués en assimilant des méthodes indépendantes des entités géométriques, mais à une différence près, après avoir extrait des primitives. C'est pourquoi certaines personnes considèrent ces méthodes comme un cas particulier de ces dernières.

L'*Iterative Closest Point (ICP*), proposé concurremment par [Besl et McKay, 1992] et [Chen et Medioni, 1992] est un algorithme itératif le plus souvent utilisé. Il est utilisé soit pour recaler deux vues partielles d'un même objet, soit pour corriger des relevés laser mobiles [Ridene, 2010]. L'idée est d'itérer deux étapes : celle de la mise en correspondance et celle de l'estimation des paramètres afin d'établir la meilleure transformation recalant des scans dont le recouvrement s'avère assez important. Au bout de chaque itération, une liste d'éléments appariés est de nouveau fournie, servant à son tour à déterminer une nouvelle estimation des paramètres de pose. Ces derniers seront utilisés par la suite, pour la mise à jour de la position des données. Toutes ces étapes seront répétées jusqu'à l'obtention de la convergence de l'algorithme. Celle-ci est atteinte lorsque l'erreur résiduelle ne dépasse pas un certain seuil, ou bien que le nombre maximum d'itérations allouées à l'algorithme est atteint. Cependant, l'*ICP* « classique » présente des limitations qui font, jusqu'à présent, l'objet de nombreuses études. C'est pourquoi plusieurs améliorations y ont déjà été apportées. Elles visent à intervenir à chaque étape de cet algorithme [Rusinkiewicz et Levoy, 2001], [Douadi, 2007], [Gressin, 2013]. Cela concerne la façon de choisir des entités à apparier [Masuda et Yokoya, 1995], [Gelfand *et al.*, 2003], la stratégie d'appariement employée [Pulli,1999], [Al-Durgham *et al.*, 2011], la pondération des paires appariées [Gressin *et al.*, 2012], le rejet de mauvais appariements [Zhang, 1994] et enfin le type de critère à minimiser et/ou l'algorithme de minimisation. L'objectif est toujours de minimiser un score, une distance ou une erreur entre les parties communes. Le critère le plus simple à utiliser est la somme des erreurs de distance quadratiques entre les points appariés (*ICP* Point à Point). Mais, les

méthodes fondées sur la somme des carrés des distances entre les points d'un scan et le plan de l'autre sont aussi couramment employées (*ICP* Point à Plan). L'estimation de la transformation rigide s'effectue à l'aide de plusieurs approches telles que la technique *SVD* (décomposition en valeurs singulières) [Arun *et al.*, 1987], l'utilisation des quaternions unitaires [Besl et McKay, 1992] ou l'algorithme d'optimisation non linéaire de *Levenberg-Marquardt* [Fitzgibbon, 2001]. L'inconvénient majeur de l'*ICP* dans sa forme originale vient du fait qu'il peut converger vers un faux minimum lorsque les données sont très bruitées et qu'il a besoin d'un nombre important d'itérations. Pour atteindre le minimum global et accélérer la convergence, une initialisation préalable fournissant de bonnes valeurs initiales est requise. En dépit de toutes les études, une intervention de l'opérateur est souvent obligatoire puisque cette opération n'est généralement pas résolue automatiquement.

6.2.3 Aperçu des méthodes basées sur les entités linéaires

L'utilisation des entités linéaires au lieu des points, pour les besoins de consolidation, constitue d'un point de vue algorithmique un défi plus important. Par contre, il a été démontré que seules deux paires de droites non colinéaires permettent de dériver la solution [Alshawa, 2006], [Yao *et al.*, 2010]. Toutefois, compte tenu des bruits qui entachent les données, il est conseillé d'employer un nombre de paires N supérieur à deux et de résoudre un système surdéterminé de N équations afin de trouver la meilleure estimation.

Les méthodes de recalage conçues pour les entités linéaires ne sont pas nombreuses, nous en citerons quelques – unes. Dans ces travaux [Zhang et Faugeras, 1991] analysent plusieurs algorithmes de minimisation, y compris le *Filtre de Kalman Étendu (EKF)*, la minimisation par la méthode de moindres carrés et la décomposition en valeurs singulières, pour étudier le déplacement à partir des segments 3D. Cette problématique étant similaire à la nôtre, elle permet d'estimer les paramètres optimaux d'une transformation rigide. Différents scénarios ont été vérifiés, en supposant que l'appariement des entités est connu, afin de juger la façon optimale de représenter la matrice de rotation et de calculer des estimées. Finalement, la représentation par un axe et un angle (*axis-angle representation*) avec l'utilisation simultanée du *Filtre de Kalman* est la solution la plus précise. En revanche, lorsque l'on considère exclusivement le temps de calcul, c'est l'approche *EIGEN* basée sur la rotation exprimée par des quaternions et estimée par la *SVD* qui paraît la plus prometteuse.

Dans le même contexte [Alshawa, 2006] propose deux variantes de méthode *ICL* (*Iterative Closest Line*) servant à déterminer des paramètres de transformation rigide à partir de droites : 1) *ICL* (forme *ICP*) ; 2) *ICL* forme alternative. La première est identique à l'algorithme *ICP* de base si l'on envisage uniquement la rotation. Mais, elle présente un aspect différent sur la façon de calculer le vecteur de translation, comme nous pourrons l'observer ultérieurement. La seconde solution inspirée de [Habib et al., 2004] est une approche itérative et non de solution analytique (*closed-form*). Son principe consiste à procéder à une consolidation en employant la transformation de similitude 3D (transformation conforme 3D) de sept paramètres dont: R - rotation, T - translation et s – facteur d'échelle, tous utilisés afin d'écrire la relation entre deux points homologues d'une paire de segments :

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = sR \begin{bmatrix} x \\ y \\ z \end{bmatrix} + T \tag{6.1}$$

Il faut noter que le système d'équations obtenu à partir de la formule (6.1) est non-linéaire, tandis que l'ajustement par la méthode des moindres carrés requiert un processus de linéarisation et une bonne approximation initiale des paramètres recherchés. Quant à la valeur du facteur d'échelle s, elle est égale à l'unité puisque les scanners laser produisent les données à l'échelle réelle. Néanmoins, cette procédure suppose que les extrémités des

segments se conjuguent avec un autre, ce qui peut ne pas avoir lieu en réalité. Les deux versions d'algorithmes implémentées par [Alshawa, 2006] permettent de coupler itérativement des entités linéaires quasi-parallèles. Au cours de chaque itération, les deux lignes les plus proches sont appariées, sous condition que leur distance réciproque soit inférieure à un certain seuil. La procédure est répétée plusieurs fois. Nous partons de la pose corrigée jusqu'à ce que l'appariement obtenu soit le même que celui de l'étape précédente.

Étant donné que les segments correspondants, extraits lors d'un processus automatique, se caractérisent souvent par des longueurs différentes (sur-segmentation due au problème des masques) une amélioration de la méthode de [Habib *et al.*, 2004] a été suggérée par [Renaudin *et al.*, 2011]. Or, un vecteur des écarts $d\vec{X}$ entre les extrémités des segments correspondants est déterminé le long d'une entité linéaire (Figure 6.4) et rajouté au modèle mathématique (6.1). Ensuite, la technique traditionnelle des moindres carrés a été modifiée afin d'éliminer cette inconnue du processus d'estimation. Cet objectif a été atteint par une nouvelle définition des poids associés au vecteur d'erreurs de mesures (bruit blanc). Cependant, aucune des modifications effectuées n'a d'impact sur le calcul. L'algorithme de minimisation de Gauss-Newton peut toujours être employé pour trouver la solution : les paramètres de similitude spatiale entre deux ensembles de segments. Ce genre d'algorithmes nécessite une transformation initiale et peut offrir une mauvaise estimation lorsque cette dernière est loin de la réalité, ce qui est son inconvénient majeur. L'évaluation qualitative (visuelle) et quantitative du recalage réalisé grâce à un tel algorithme a été effectuée par [Canaz et Habib, 2013]. À cet effet, les paramètres de pose obtenus ont été comparés avec ceux fournis par d'autres méthodes, notamment par le recalage basé sur les plans [Ghanma, 2006] et l'algorithme *ICPP* (*Iterative Closest Projected Point*) de [Al-Durgham *et al.*, 2011] étant l'une des variantes d'*ICP*. Son idée consiste à minimiser la distance entre chaque point d'un scan et sa projection sur trois points de l'autre. Quant à la technique examinée, certains problèmes au niveau d'alignement vertical ont été observés. L'une des causes de cette imprécision vient de la nature des lignes elles-mêmes. Par conséquent, la droite extraite à partir des images et associée à l'arête reconstruite du nuage de points peut être sensiblement différente.

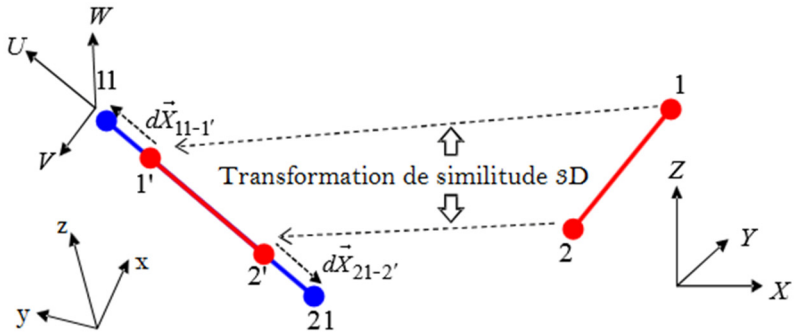

Figure 6.4 Principe du recalage se référant à la transformation de similitude 3D (Illustration reproduite à partir de [Renaudin et al., 2011])

Sous un autre point de vue [Guerra et Pascucci, 1999] discutent de différentes méthodes de recalage d'entités linéaires pour lesquelles l'appariement n'est pas connu. Ils proposent d'utiliser un triplet de paires de segments afin de trouver des paramètres de transformation rigide. Le but à atteindre est celui de tester un très large nombre de transformations, toutes calculées à partir des échantillons tirés aléatoirement. La distance déterminée, à chaque fois, entre les lignes permet d'indiquer le meilleur couplage. L'inconvénient de cette méthode est d'avoir une complexité de calcul relativement élevée. De plus, elle suppose implicitement que les milieux des segments doivent être des points correspondants.

[Kamgar-Parsi et Kamgar-Parsi, 2004] ont développé des algorithmes applicables, quelles que soient les lignes : finies ou infinies. La solution adaptée à des segments de ligne de même longueur est présentée par [Kamgar-Parsi et Kamgar-Parsi, 1997]. Pour des segments de longueur différente, trois cas sont envisagés. En désignant des entités linéaires à déplacer par *Image* (*I*), tandis que celles de référence comme *Model* (*M*), nous distinguons des approches : 1) *FMFI (Finite Model, Finite Image)* conçu pour dériver la transformation à partir de deux ensembles de segments ; 2) *IMII (Infinite Model, Infinite Image*) destiné aux lignes infinies et 3) *FMII (Finite Model, Infinite Image)* approprié à des cas mixtes. En revanche, tous les calculs associés s'appuient sur le schéma développé pour les segments égaux. La fonction de coût minimise la distance entre les points correspondants de chaque paire de droites. Ces derniers sont établis par le paramètre *shift* $\{s_i\}$. Finalement, l'estimation de la meilleure transformation se fait itérativement, par la résolution d'un système non linéaire de $(6 + N)$ équations, en partant de valeurs initiales de $\{s_i\}$, actualisées au bout de chaque boucle. Les analyses effectuées par les auteurs montrent que l'algorithme converge presque toujours vers un minimum, et indépendamment des paramètres initiaux choisis. Le temps de calcul et le nombre d'itérations dépendent de la précision souhaitée.

Deux des algorithmes mentionnés auparavant sont devenus l'élément de base des autres méthodes. [Yao *et al.*, 2010] ont cherché à combler le manque d'informations sur l'appariement des entités linéaires à travers une approche s'effectuant en deux étapes. Dans un premier temps, une transformation initiale est calculée à l'aide d'algorithme de [Zhang et Faugeras, 1991], afin d'être ensuite affinée avec l'une des solutions de [Kamgar-Parsi et Kamgar-Parsi, 2004] modifiée de telle façon qu'elle soit moins susceptible de s'appuyer sur des segments de qualité différente. Il se peut que l'incertitude associée à chaque ligne ne soit pas la même, car elle résulte de la densité du nuage diminuant avec l'augmentation de la distance objet-scanner. À chaque phase, le calcul est mené à partir d'un nombre minimal de paires ($N \geq 2$), excepté que les résultats pour $N = 3$ sont plus robustes et fiables que ceux pour $N = 2$.

6.3 Recalage basé-lignes : caractérisation des méthodes

Plusieurs approches traitant du problème de recalage rigide basé sur les entités linéaires 3D ont été proposées dans la littérature, comme il a été montré dans la section précédente. Ce paragraphe va aborder en détail trois méthodes méritant notre attention, puisqu'elles donnent un autre regard sur la question mise en avant. Dans un premier temps, la solution nommée *EIGEN* de [Zhang et Faugeras, 1991] ayant pour but d'estimer le mouvement à partir des segments 3D sera détaillée. Ensuite, l'algorithme *ICL* de [Alshawa, 2006] conceptuellement inspiré de l'algorithme *ICP* sera présenté – *ICL*(forme *ICP*). Enfin, l'essentiel de l'approche *FMII* de [Kamgar-Parsi et Kamgar-Parsi, 2004] sera discuté. Chacune des méthodes mentionnées auparavant profite d'une autre représentation du segment dans un espace tridimensionnel et s'appuie sur une autre notation mathématique décrivant la rotation d'objet telle que les quaternions, les angles d'*Euler* ou bien encore la matrice traditionnelle de rotation.

Ces méthodes sont spécifiées respectivement à l'Annexe B, Annexe C et Annexe D. Les signes apparaissant dans toutes les équations se présentent de la manière suivante : $\|v\|$ symbolise la longueur (norme) du vecteur v, t indique la transposition, l'inverse d'une matrice est marqué par l'indice ($^{-1}$), le produit vectoriel est signalé par « ∧ » et bien entendu chaque vecteur s'exprime verticalement.

Le Tableau 6.1 résume les trois méthodes d'estimation de la transformation présentées dans la section précédente. Les approches *ICL* (forme *ICP*) et *EIGEN* sont assez similaires, si nous envisageons seulement les démarches nécessaires pour calculer la matrice de rotation – elles s'appuient toutes deux sur l'orientation de lignes. Aussi, la *SVD* est employée afin de dériver la rotation optimale. Néanmoins, elles présentent un regard diffèrent quand il s'agit

de déterminer la translation, résultant principalement de la représentation différente de ligne 3D. Contrairement à *EIGEN*, l'approche *ICL*(forme *ICP*) permet simultanément d'apparier les entités linéaires ce qui devient son point fort. Ce processus se fait de façon itérative et implique le temps de calcul adéquat. Si jamais l'appariement explicite est connu, la solution peut être immédiatement trouvée (non itérativement).

Quant à la méthode *EIGEN*, elle se réfère à des points correspondants et non plus des lignes. Leurs coordonnées sont déterminées grâce au paramètre *shift* $\{s_i\}$, étant calculé de manière itérative. Nous devons alors introduire ses valeurs initiales, proches de celles réelles. Sinon, l'algorithme peut donner de mauvais résultats.

Chacune des méthodes aboutit, quelles que soient les démarches proposées, à estimer les paramètres de transformation en deux étapes : d'abord la rotation R suivie ensuite par la translation T.

Tableau 6.1 Récapitulatif de méthodes présentées

Auteurs	[Zhang et Faugeras, 1991]	[Alshawa, 2006]	[Kamgar–Parsi et Kamgar–Parsi 2004]
Nom	**EIGEN**	**ICL (forme ICP)**	**FMII**
Appariement connu	Oui	Non	Oui
Représentation du segment	Deux vecteurs perpendiculaires : vecteur directeur normalisé et vecteur parallèle à la normale d'un plan passant par cette ligne et l'origine	Deux points (extrémités) et vecteur directeur	Ligne finie : (milieu, vecteur directeur unitaire, longueur) Ligne infinie : (point de la ligne le plus proche de l'origine, vecteur directeur unitaire)
Représentation de la matrice de rotation	Quaternion	Matrice traditionnelle de rotation	Quaternion
Fonction à minimiser	Orientation de lignes	Orientation de lignes	Distances entre points correspondants
Approche	SVD	SVD	Méthode d'optimisation non-linéaire (valeurs initiales du paramètre *shift* demandées)
Besoin d'itérations ?	Non	Oui (si appariement à faire)	Oui

6.4 Évaluation des méthodes présentées

Dans cette partie, nous contrôlerons la robustesse des trois algorithmes détaillés ci-dessus et récapitulés dans le Tableau 6.1. Pour ce faire, plusieurs tests ont été proposés, puis effectués sur des données simulées pour lesquelles l'appariement était déjà connu. Les segments ont été obtenus par modification des arêtes extraites à partir du nuage *Sulpice2013_statique*, acquis au sein d'une zone test (description détaillée se trouve en section paragraphe 2.2). Nous avons évalué la précision des paramètres estimés par chacune des méthodes en admettons le taux de bruit. À chaque fois, les erreurs e_R *et* e_T des paramètres de transformation estimés $\hat{R}$ *et* $\hat{T}$ ont été calculées. Aussi, la distance de *Hausdorff* orientée

(*OLHD*) entre deux ensembles de segments, avant et après la transformation, a été déterminée comme cela est expliqué dans le Chapitre 5.

6.4.1 Données pour les expérimentations

Quant aux données simulées, elles étaient générées à partir de 64 segments (F) nommés désormais *Model*, en appliquant les paramètres prédéfinis de transformation. Deux scénarios ont été examinés : 1) les extrémités des segments originaux, en système Lambert93, ont été dégénérées en leur ajoutant un bruit blanc gaussien centré et de variance connue σ ; 2) de plus, la position des points a été modifiée en appliquant les paramètres prédéfinis de transformation rigide R et T.

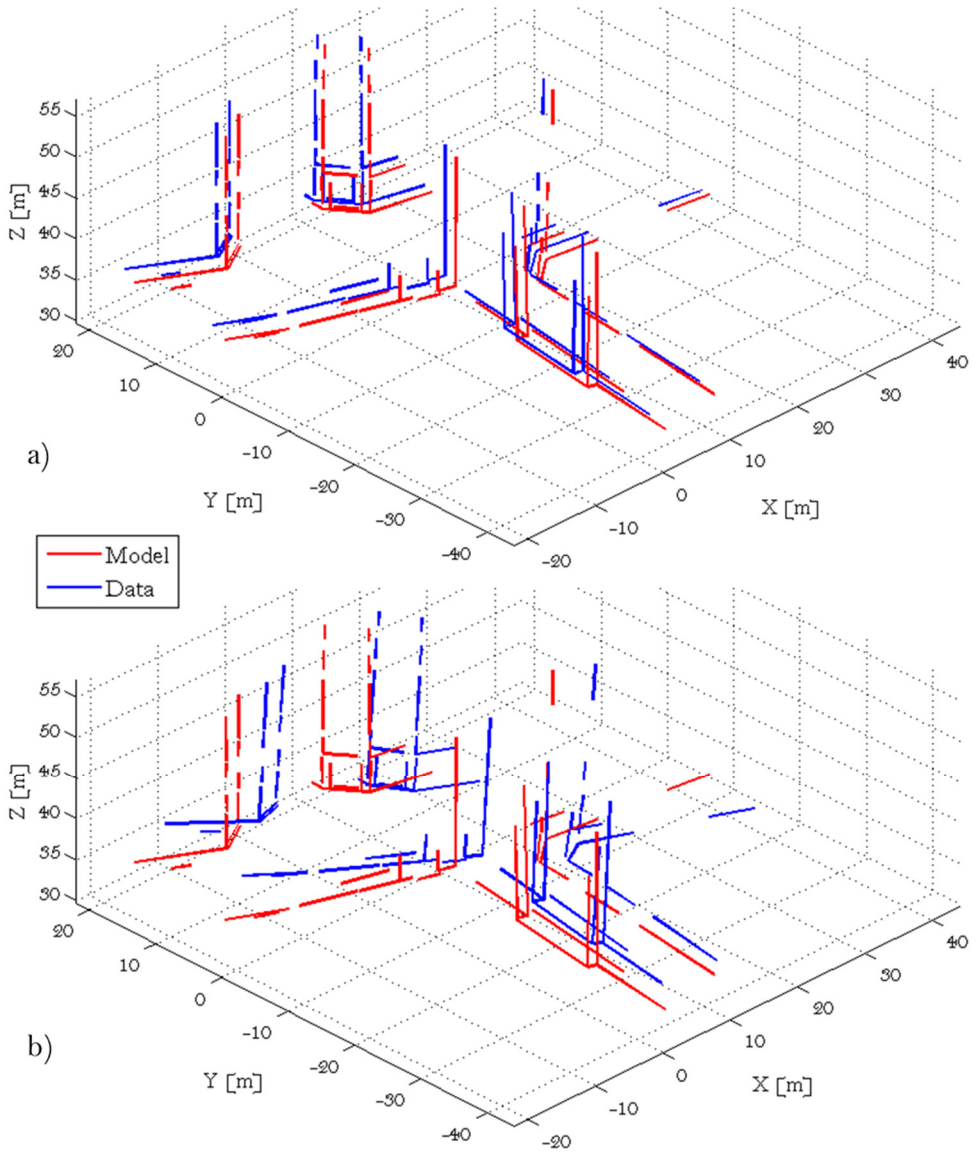

Figure 6.5 Données expérimentales à recaler : a) *Test2a* ; b) *Test2b*

Dans un premier temps, de petites rotations dans l'espace 3D, exprimées par les trois angles d'Euler (1.0000° ; -1.0000° ; 1.0000°), ont été introduites. Le vecteur de translation T = [-1.000m ; 0.500m ; 1.000m] y associé a été aussi pris en compte. De surcroît, les coordonnées de ces entités linéaires ont été bruitées par tirage aléatoire suivant une loi

normale centrée d'écart-type σ. Les tests ont été menés pour différentes valeurs de bruit, en partant de σ =0.000m et en finissant sur σ =0.050m, avec l'intervalle constant de 0.001m. En résulte, des centaines de nouveaux ensembles de données *Test*, chacun contenant 64 segments, ont été obtenus. Ils constituent d'ores et déjà les données nommées *Test2a* (Figure 6.5a). Les données *Test2b* (Figure 6.5b) étaient obtenues de façon identique, mais cette fois-ci, les angles d'Euler ainsi que la translation introduits étaient plus importants et égaux respectivement (3.0000° ;-7.0000° ; 6.0000°) et (3.000m ; 1.000m ; -2.000m). La distance *LHD* mesurée entre les données *Test2a* était égale environ à 2.7m, tandis que celle entre les entités *Test2b* est de 27.5m, en ignorant le bruit ajouté.

Nous avons généré un autre jeu de données (*Test3*), pour lequel la transformation T fixée en entrée était nulle. Nous avons uniquement simulé le bruit gaussien de variance σ pour obtenir l'ensemble *Test* à partir de *Model*.

6.4.2 Algorithmes implémentés

Le travail réalisé avec l'implémentation sous *Matlab*® R2010a a permis d'évaluer chaque approche en termes de sensibilité au bruit, de précision de résultats et de temps d'exécution. Les méthodes mises en œuvre suivent le raisonnement présenté dans la section précédente. Parmi les approches proposées par [Kamgar – Parsi et Kamgar-Parsi, 2004] nous avons décidé d'examiner l'algorithme *FMII*, en dépit du fait qu'il est dédié à recaler des lignes infinies avec des lignes finies. Notre choix s'explique, conformément à la suggestion des auteurs, par le fait que le *FMFI*, dédié aux segments de ligne, peut échouer si le segment plus court d'une paire n'est pas complètement inclus dans son homologue. Nous avons légèrement modifié l'algorithme FMII pour le rendre mieux adapté à nos besoins. Nous lui accordons le nom $FMII_2$ pour le différencier de la version de base. D'abord, la représentation d'entités à déplacer (*Image*) a été remplacée par celle utilisée pour les droites finies, à une différence près, nous ne prenons pas toujours en compte les longueurs de segments. Chaque élément *Image* $X_i = (x_i, v_i')$ est alors représenté par x_i étant son milieu (et non plus le point le plus proche de l'origine du système de coordonnées) et v_i' son vecteur directeur. Cette solution paraît plus adéquate et accélère la recherche ultérieure des points correspondants qui seront désormais déterminés par rapport aux milieux des segments. En conséquence nous avons dû changer la façon de calculer le paramètre *shift* $\{s_i\}$ en s'inspirant aussi de l'algorithme *FMFI*. Celui-ci est obtenu par la résolution de l'équation $\partial M/\partial s_i = 0$, où la distance *M(A,TX)*, introduite pour le *FMFI* (6.2), se calcule en fonction de la longueur des segments.

$$M(A_i, \mathrm{T}X_i) = \begin{cases} l_i \|a_i + s_i v_i - T - Rx_i\|^2 + l_i^3 (1 - v_i^t R v_i^{'})/6 & si\ l_i \le L_i \\ L_i \left\|a_i - T - R(x_i + s_i v_i^{'})\right\|^2 + L_i^3 (1 - v_i^t R v_i^{'})/6 & si\ l_i > L_i \end{cases} \tag{6.2}$$

où :

L_i : longueur du segment *Model*

l_i : longueur du segment *Image*

Sachant que nous considérons les entités linéaires *Image* comme infinies, leur longueur (l_i) sera toujours plus grande que celles *Model*. De ce fait, la formule (D.10) est substituée en :

$$s_i = (a_i - T - Rx_i)^t R v_i' \tag{6.3}$$

6.4.3 Outils d'évaluation

La justesse des paramètres de transformation estimés à l'aide de chacune des méthodes a été validée selon les formules :

$$e_R = 100\% \cdot \|r - \hat{r}\| / \|r\| \tag{6.4}$$

$$e_T = 100\% \cdot \|T - \hat{T}\| / \|T\| \tag{6.5}$$

où :
r : rotation réelle (paramètre fixé en entrée)
$\hat{r}$: rotation estimée
T : vecteur tridimensionnel de translation (paramètre fixé en entrée)
$\hat{T}$: vecteur de translation estimée

L'équation (6.4) est exacte pour la rotation exprimée à l'aide de quatre paramètres (forme axe-angle de rotation) : un angle θ et un axe dirigé par un vecteur unitaire $r = [e_x, e_y, e_z]^t$ (alors avec $e_x^2 + e_y^2 + e_z^2 = 1$). Il est donc nécessaire de remplacer la représentation classique de rotation R par r de la manière suivante :

$$\theta = arccos\left(\frac{trace(R) - 1}{2}\right) \tag{6.6}$$

$$r = \frac{1}{2sin\theta}\begin{pmatrix} R_{32} - R_{23} \\ R_{13} - R_{31} \\ R_{21} - R_{12} \end{pmatrix} \tag{6.7}$$

6.4.4 Résultats – recalage des données bruitées

Ensemble Test2a

La Figure 6.6 illustre les erreurs de la rotation e_R et de la translation e_T, par rapport à ses valeurs réelles. Les tests effectués avec les données *Test2a*, montrent que les algorithmes *ICL*(forme ICP) et *EIGEN* produisent des résultats à un niveau de précision similaire, avec des estimations légèrement plus précises pour *EIGEN*.

Même si les deux approches bénéficient d'une représentation différente de segments, et que la translation est calculée dissemblablement, ceci n'a pas une réelle importance. Dans les deux cas, la fonction de coût est uniquement définie à partir des orientations de segments. Aussi, la matrice de rotation est retrouvée par une décomposition *SVD* d'une matrice de covariance croisée. Rappelons que l'utilisation de *SVD* impose de considérer trois possibilités :

1) Les segments ne sont pas colinéaires : donc $det(R) = 1$, la solution unique existe $R = VU^t$;
2) Les segments sont coplanaires, mais pas colinéaires : alors $det(R) = -1$, la solution unique existe nommée la réflexion $R^* = V^*U^t$ où V^*est créé en changeant le signe de la troisième colonne de V ;
3) Les segments sont colinéaires : il existe une infinité de solutions.

Quant à l'évaluation des approches implémentées, les deux méthodes mentionnées présentent une faible résistance au bruit. L'estimation de la rotation atteint un niveau de précision de 5% pour un écart-type du bruit σ ne dépassant pas 0.015m. L'erreur de la translation est plus importante et peut arriver, dans cet intervalle, jusqu'à 27%. De loin, le meilleur algorithme le $FMII_2$ donne, indépendamment du bruit introduit ($\sigma \in [0.000 ; 0.005]$), des résultats précis. L'erreur de la matrice de rotation e_R est inférieure à 0.5% pour un σ <0.02m et oscille entre 0.5% et 3.6% pour les autres valeurs σ, afin d'atteindre son maximum pour σ =0.047. L'erreur de translation e_T, quant à elle, est supérieure à e_R et reste dans un intervalle de 2.9% à 8.5%.

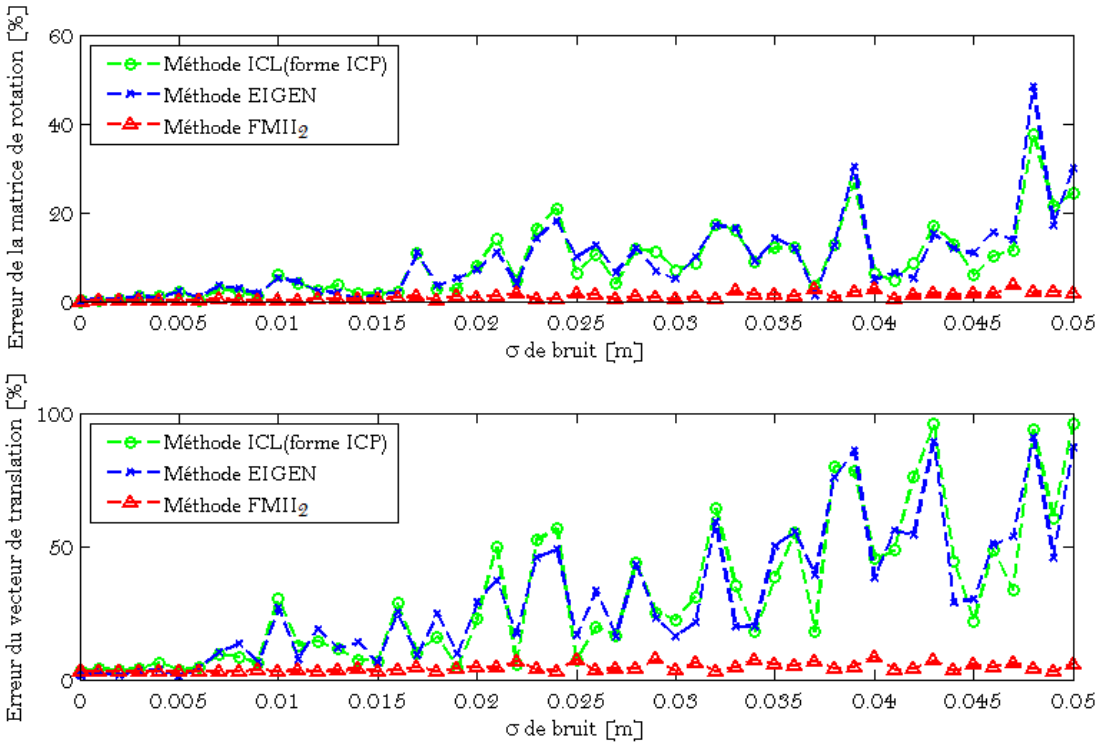

Figure 6.6 Comparaison de différentes méthodes d'estimation de transformation rigide – données *Test2a*

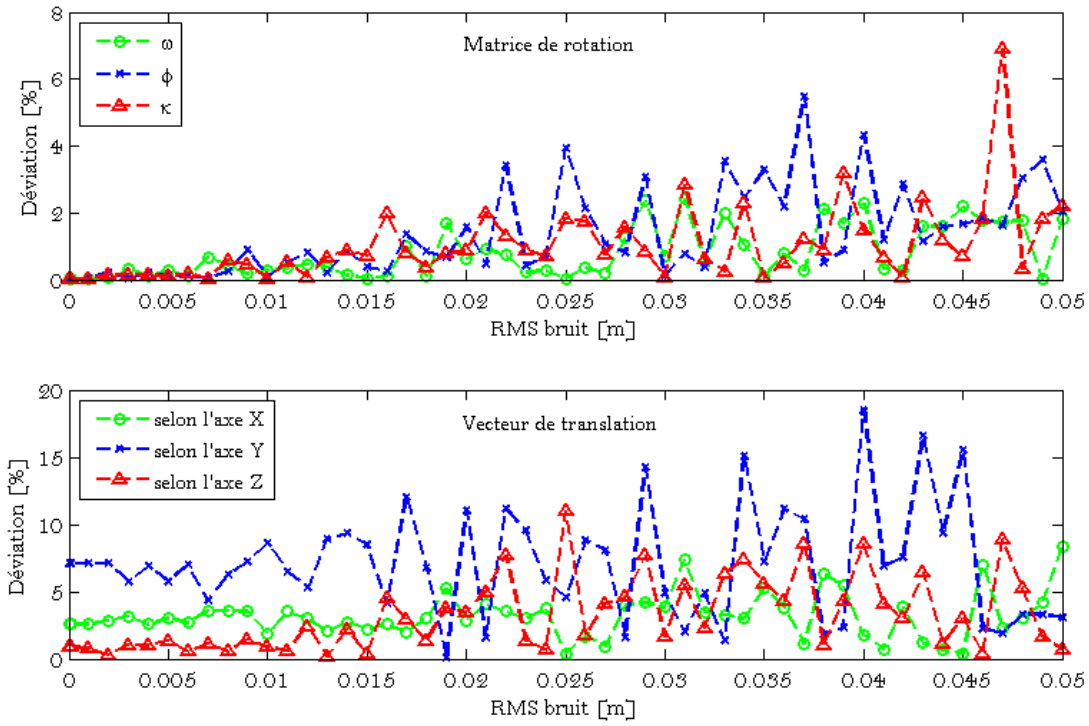

Figure 6.7 Méthode $FMII_2$: évaluation de chaque composant de la rotation et de la translation par rapport aux valeurs réelles

Afin d'obtenir une évaluation plus complète du $FMII_2$, une analyse exhaustive de chaque composant du vecteur de translation et de tous les angles d'Euler (ω,ϕ,K) a été faite. Des différences entre les paramètres estimés et réels ont été calculées et illustrées dans la Figure 6.7. Quant aux valeurs angulaires, aucune tendance n'a été constatée pouvant indiquer l'angle le moins bien estimé. Mais, en tenant compte du vecteur de translation, l'alignement selon l'axe Y est déterminé inexactement dans la plupart de cas. Pour un écart-type du bruit raisonnable, c'est-à-dire un σ inférieur à 0.02m, il est clairement visible qu'il existe une dépendance entre les erreurs relatives de déplacements. Ainsi, l'erreur d'alignement selon l'axe Z est la plus petite, suivie ensuite d'erreur selon l'axe X et Y.

Ensemble Test*2b*

Des tests tout à fait identiques ont été aussi menés pour des valeurs beaucoup plus importantes de rotation et de translation – données *Test2b*. Plusieurs combinaisons d'angles de rotation ont été examinées. La Figure 6.8 illustre les erreurs e_R et e_T obtenues pour différents écarts-types du bruit σ. Les résultats confirment que les trois méthodes sont adéquates pour recaler deux ensembles de segments à condition que les angles d'Euler décrivant une rotation soient petits (inférieurs à 2-3 degrés) et la distance relative assez faible. Cette constatation n'est pas très surprenante puisque les méthodes semblables à l'*ICP* sont censées fournir un bon recalage local. Ceci est dû à l'étape de linéarisation indispensable pour résoudre le problème non-linéaire de rotation. Ainsi, la fonction ne peut plus décrire les vraies interactions entre les éléments. Il est alors nécessaire que les deux nuages soient à peu près alignés. Une transformation grossière doit être appliquée avant de démarrer ces algorithmes. Aussi, le $FMII_2$, comme il a été démontré, exige des estimées initiales, si la rotation mutuelle des éléments à affiner est trop importante.

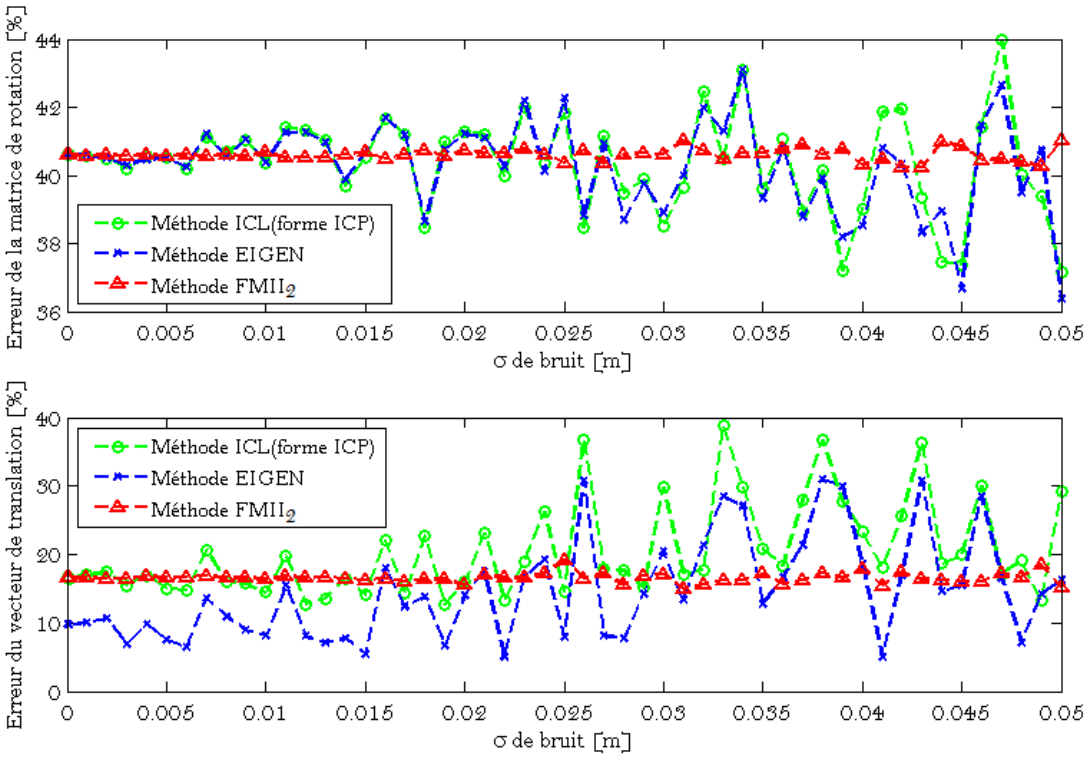

Figure 6.8 Comparaison de différentes méthodes d'estimation de transformation rigide – données *Test2b*

Pour conclure, nous constatons alors que les méthodes *ICL*(forme *ICP*) et *EIGEN*, contrairement au $FMII_2$, présentent une faible robustesse au bruit et ne peuvent être employées qu'afin de recaler des segments dont la position des extrémités est précisément connue ($\sigma < 0.005$m). De plus, l'emplacement relatif de deux ensembles d'entités linéaires joue un rôle important en rendant ces trois approches appropriées pour affiner des données proches.

Temps de calcul

En outre, une analyse complémentaire (avec les données *Test2a*) ayant pour but de mesurer le temps de calcul (*CPU time*) en fonction du bruit simulé a été effectuée. La Figure 6.9 présente la performance des trois approches, examinée sur 64 segments. La méthode *EIGEN* s'avère légèrement plus rapide (avec en moyenne 2ms de différence) que l'algorithme itératif *ICL*(forme *ICP*). Cependant, les deux techniques convergent trois fois plus rapidement que le $FMII_2$. Néanmoins, il faut noter que chaque algorithme a été appliqué sur l'appariement déjà connu et donc l'approche *ICL*(forme *ICP*) s'exécute en une seule itération.

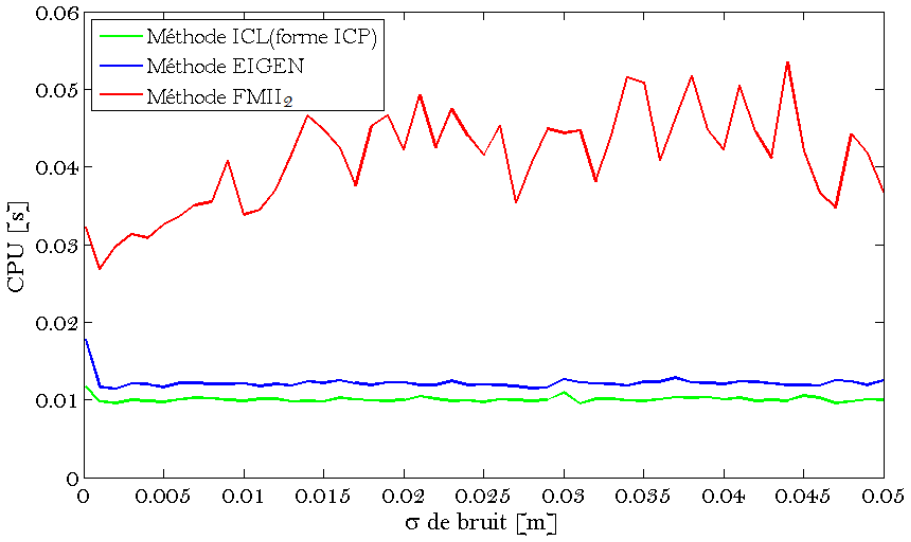

Figure 6.9 Comparaison des temps d'exécution (*Matlab*® R2010a)*
* Intel Core x2 T7600 2.33Ghz / 4Go Ram / Win7 64bits

6.4.5 Impact du bruit sur la distance de *Hausdorff*

Pour compléter les analyses présentées précédemment et pour mieux visualiser l'impact de la précision des paramètres estimés sur la qualité du recalage, la distance orientée de *Hausdorff* (*OLHD*) a été utilisée comme il a été vu et défini dans le Chapitre 5. Un autre jeu de données (*Test3*) a été employé. Nous rappelons que cette fois-ci, les paramètres fixés servant à créer les segments *Test* étaient nuls, et seulement la position des segments *Model* a été dégénérée par un bruit gaussien prédéfini.

Nous attendons que les paramètres estimés ($\hat{R}$ *et* $\hat{T}$) par chacun des algorithmes testés soient égaux, quasiment équivalents ou proches de zéro. Aussi, la position des segments transformés (*Test*) devrait rester pratiquement inchangée. Nous vérifions alors l'éloignement entre les entités linéaires, en tant que la distance *OLHD*. Les valeurs obtenues après le recalage réalisé à l'aide de la méthode *ICL*(forme *ICP*) et $FMII_2$ sont confrontées avec les écarts initiaux (Figure 6.10). L'approche *EIGEN* a été exclue de l'analyse puisqu'elle est censée fournir les paramètres de transformation ayant les erreurs similaires à l'algorithme *ICL*(forme *ICP*).

Nous observons, dans le cas d'*ICL*, une forte influence du bruit sur le résultat d'évaluation. Même si l'inexactitude des extrémités de segments est faible, cet algorithme empêche de parvenir aux paramètres optimaux. Par conséquent, la distance réciproque entre les segments consolidés augmente par rapport à celle de départ. Cela veut dire que le recalage, lui-même, est loin d'être correct et satisfaisant. Le problème mentionné n'existe pas lorsque l'algorithme $FMII_2$ est implémenté.

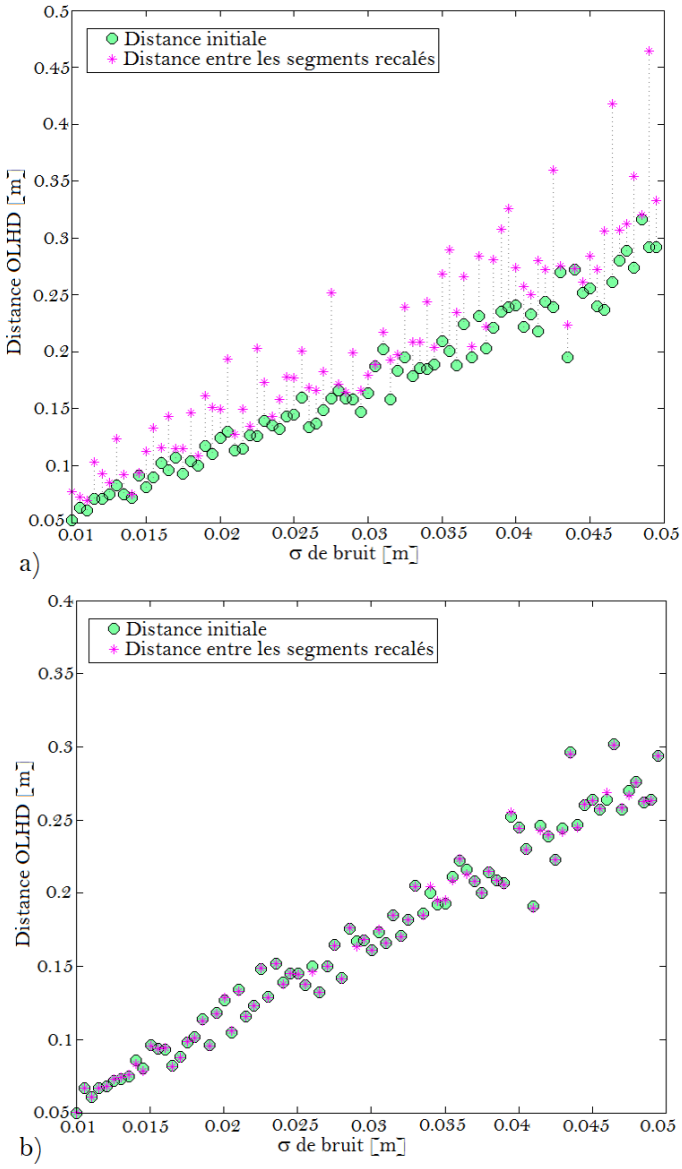

Figure 6.10 Distance orientée de *Hausdorff* (*OLHD*) : a) algorithme *ICL*(forme *ICP*) ; b) algorithme $FMII_2$

Propriété de la distance de *Hausdorff*

De surcroît, une propriété intéressante de la métrique de *Hausdorff* employée a été constatée. Il s'agit de la relation entre la distance *OLHD* et l'écart-type de bruit gaussien σ rajouté à des extrémités de segments. Nous avons démontré que plus le taux de bruit est important plus la valeur de distance *OLHD* augmente. Plusieurs échantillons ont été créés à partir de

données réelles (S1,S2,S3,F), comme celui illustré dans la Figure 6.11, pour retrouver cette constante (coefficient de proportionnalité). Finalement, l'équation d'une droite ayant une meilleure approximation (régression linéaire) notée sous la forme $y = ax + b$ a été déterminée, de sorte que le coefficient directeur de la droite (a) est égal à 5.8 ∓ 0.17 et que le nombre b est 0.000 ∓ 0.001. Pour être plus précis, il y a une proportionnalité entre la distance obtenue ($OLHD$) et l'écart-type de bruit σ, ce que l'on peut exprimer par $OLHD \propto \sigma$. C'est pourquoi, si la distance de *Hausdorff* entre deux ensembles de segments est connue, nous pouvons estimer l'écart-type de bruit σ associé aux segments et vice versa.

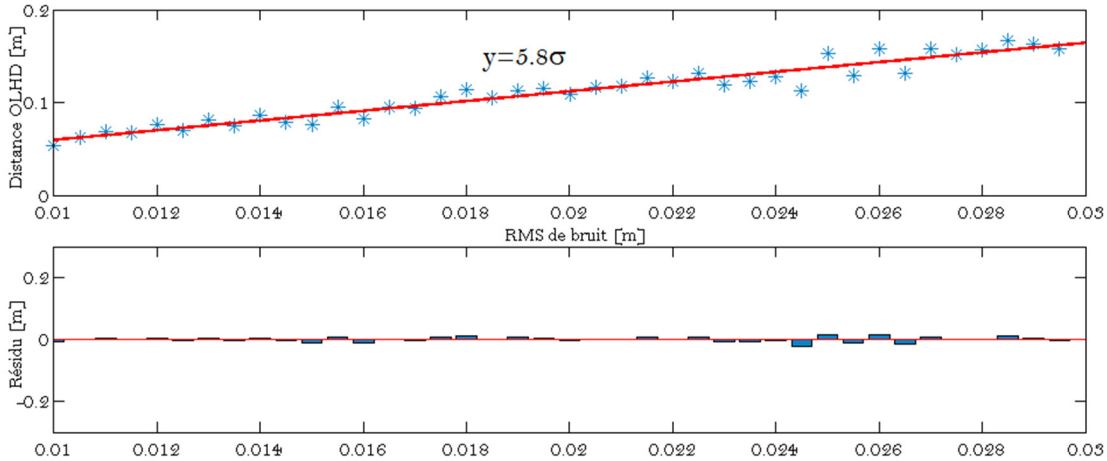

Figure 6.11 Représentation graphique d'une droite d'approximation retrouvée à partir des distances *OLHD* calculées pour les différents écarts-types de bruit σ

6.5 Approche proposée : *RLMR-FMII2*

Nous avons pu observer dans la section précédente que l'imprécision du recalage vient de l'algorithme d'estimation et voire même plus, de la particularité géométrique d'une entité linéaire. Comme le rapporte [Hullo, 2013], l'incertitude d'une ligne 3D, à la différence de celle d'un point 3D, n'est pas bien définie. En outre, elle varie d'une représentation à une autre pouvant provoquer des difficultés lors de l'optimisation des contraintes sur ces lignes. Les analyses effectuées au sein de cette thèse le confirment aussi. Les algorithmes *EIGEN* et *ICL*(forme *ICP*) basés entièrement sur des segments ont la précision en deçà du $FMII_2$ minimisant la distance entre les points correspondants. Ils ne peuvent être employés que sur des segments dont les extrémités ne sont pas bruitées.

Ce constat nous a encouragés à choisir le $FMII_2$, comme notre algorithme de base. Néanmoins, cette méthode offre des résultats précis à condition que l'appariement explicite des entités linéaires soit connu. Nous avons voulu combler ce défaut en suggérant une chaîne complète, nommée $RLMR - FMII_2$, comprenant la mise en correspondance des segments, suivie par l'estimation des paramètres de pose. Pour apparier des entités linéaires, nous proposons d'utiliser l'approche déjà discutée dans le cadre de ce manuscrit, et détaillée au chapitre précédent. Sa performance a été vérifiée à travers plusieurs tests dont les résultats ont dévoilé une dépendance forte entre l'exactitude du couplage effectuée et l'écartement de deux ensembles de segments. Ceci est dû au fait que la mise en correspondance des segments peut s'effectuer incorrectement, puisque la méthode employée se heurte à un seuil inapproprié, ou bien encore les ressemblances entre les segments ne peuvent plus être reconnues proprement. Il se peut alors qu'au bout d'une certaine distance, le taux de Faux Positif (FP) soit élevée par-devers le nombre de Vrais Positifs faible. Nous avons aussi conclu que l'un des remèdes à ce genre de problème pouvait consister à rapprocher des segments avant de démarrer l'algorithme. Nous avons fait recours à cette solution en proposant des

démarches complémentaires (recalage grossier) ayant pour but de diminuer l'éloignement entre les données traitées et faciliter leur mise en correspondance ultérieure. Notre but est donc de développer une méthode de recalage 3D résistante au bruit, aux paires aberrantes, mais aussi permettant d'apparier simultanément des segments, puisque cette tâche n'est pas triviale et surtout impossible avec le $FMII_2$ seul.

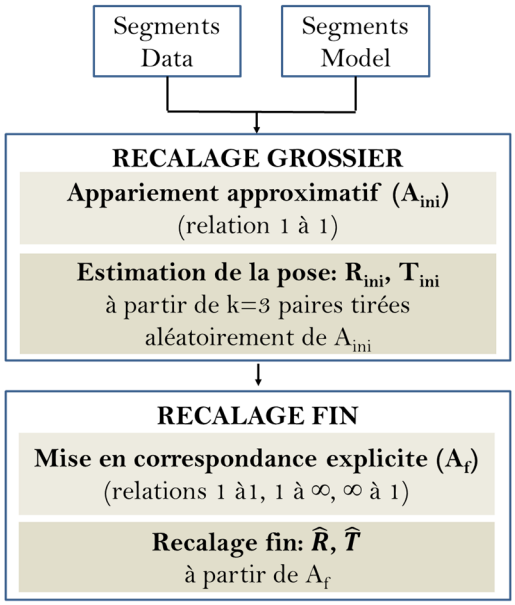

Figure 6.12 Schéma général du recalage proposé $RLMR - FMII_2$

De manière générale, nous pouvons distinguer quatre étapes principales de l'approche développée : 1) l'appariement combinatoire (approximatif) d'entités linéaires 3D ; 2) le recalage grossier à partir d'un triplet de paires ; 3) l'appariement fin; 4) le raffinement du recalage (Figure 6.12).

La mise en correspondance, qu'elle soit approximative A_{ini} ou explicite A_f, est réalisée conformément à la solution proposée pour la méthode de qualification. C'est pourquoi nous ne revenons plus sur cette procédure. Rappelons seulement qu'une fois cet algorithme exécuté, les calculs sont menés dans deux sens : en comparant l'ensemble *Test* (segments à déplacer) avec le *Model* (segments immobiles) et vice versa. Ainsi, à la sortie, deux matrices de correspondance sont indépendamment déterminées. Cette information redondante peut être utile pour contrôler la justesse de l'appariement. Afin de diminuer le nombre de paires erronées (FP) et de garantir un bon appariement approximatif A_{ini}, les segments sont initialement couplés en relation de 1 à 1 au plus. Mais, cette contrainte n'est plus valable avec l'appariement explicite A_f.

Les deux recalages – grossier et fin, sont calculés de la même façon, à savoir, à l'aide de l'approche $FMII_2$. La seule différence réside dans le fait que la consolidation initiale (R_{ini}, T_{ini}) est atteinte à partir d'un sous-ensemble minimal (k), considéré comme « le meilleur » possible. Une technique de tirage aléatoire (*RANSAC*) de k paires de l'ensemble A_{ini} amène à assurer un échantillon optimal, ne contenant plus de mauvaises associations. Or, la méthode combinée *RANSAC-FMII2* fournit une estimation initiale de la transformation en prenant en compte l'appariement préalable A_{ini}. Pourtant, la position des segments, après le recalage grossier, est suffisamment proche de l'optimum, nous pouvons créer un appariement explicite A_f servant, à son tour, au recalage fin réalisé avec le $FMII_2$ seul. En fin de compte, chacun de ces algorithmes constitue une partie intégrale de notre procédure $RLMR - FMII_2$.

6.5.1 Recalage grossier – principe

La finalité du recalage grossier cherche à rapprocher deux ensembles d'entités linéaires pour faciliter leur appariement. En présumant l'existence de mauvaises paires au sein de l'appariement A_{ini} résultante de la distance mutuelle importante entre deux ensembles de segments, nous ne pouvons pas l'utiliser tel quel lors de l'estimation des paramètres par $FMII_2$. Se référant aux tests menés par [Yao *et al.*, 2010], nous avons décidé de nous appuyer sur seulement trois paires de segments (k=3) pour trouver les estimées initiales. Ces dernières sont calculées à l'aide de la méthode $FMII_2$. Plusieurs tirages aléatoires doivent être effectués avec l'algorithme de *RANSAC*, afin de découvrir le meilleur triplet de paires parmi celles recueillies par l'appariement initial A_{ini}. Pour obtenir les bons paramètres, le nombre d'itérations (N_{iter}) est choisi dynamiquement en fonction des « inliers » présents dans la boucle précédente (équations (6.8) et (6.9)).

$$N_{iter} = \frac{\log(\varepsilon)}{\log(1-q)} \tag{6.8}$$

$$q = \frac{\binom{N_I}{k}}{\binom{N}{k}} = \frac{N_I!\,(N-k)!}{N!\,(N_I-k)!} = \prod_{i=1}^{k-1} \frac{N_I - i}{N - i} \approx \left(\frac{N_I}{N}\right)^k \tag{6.9}$$

où :
$N_I = card(EC)$: nombre de paires de segments (*inliers*) satisfaisants la transformation effectuée ;
$N = card(A_{ini})$: nombre de paires de l'appariement initial A_{ini} ;
k : nombre minimal de paires de segments pour estimer la transformation optimale, la valeur adoptée k=3 ;
ε : taux de fausses alertes (*false alarm rate*), la valeur adoptée $\varepsilon = 1e-6$;

En raison de la faible probabilité de parvenir à la même combinaison de triplet de segments deux fois, nous n'avons pas modifié l'algorithme de *RANSAC*, en exigeant la mise en place d'une liste de combinaisons d'entités linéaires déjà prises. Pour chaque échantillon, la transformation du repère est calculée (avec FMII_2), et suivie par un recalage correspondant. Dès lors, les distances $d(t_i, m_j)$ entre toutes les paires de segments listées par l'appariement approximatif A_{ini} sont recalculées (équation 5.10), afin de mesurer le nombre de valeurs aberrantes (*outliers*) définies comme $\left|\{(i,j)\epsilon A_{ini}\}: \forall(t_i \in Data, m_j \in Model), d(t_i, m_j) > \delta\right|$. Le seuil δ est fixé du fait que la distance de *Hausdorff* orientée (*OLHD*) entre deux ensembles de lignes n'est jamais inférieure à 5.8σ; comme il a été démontré préalablement (Paragraphe 6.4.5). Cette constatation nous a permis de garder « le meilleur » triplet de paires. Celui-ci est désigné à partir du nombre de segments recalés ayant la distance réciproque inférieur au seuil prédéfini δ. En conséquence, plus le nombre de segments est important, plus la transformation calculée est meilleure.

Nous avons également examiné la précision des paramètres estimés $\hat{R}$ et $\hat{T}$ lors du recalage grossier. Les données du *Test2a* ont été employées pour calculer les erreurs e_R et e_T, selon les équations (6.4) et (6.5). La Figure 6.13 présente les résultats obtenus par rapport à l'état de l'art. Il en résulte que le recalage grossier basé sur trois paires de segments est moins précis que celui réalisé par l'algorithme $FMII_2$ seul sur toutes les paires connues a priori. Cela n'est guère surprenant puisque la qualité de la solution reçue avec le $FMII_2$ augmente de plus en plus, quand nous rajoutons d'autres couples. Par contre, ce recalage garantit une bonne estimation initiale, si les deux ensembles de segments ne sont pas très éloignés, et si les angles de rotation à estimer ne sont pas trop importants.

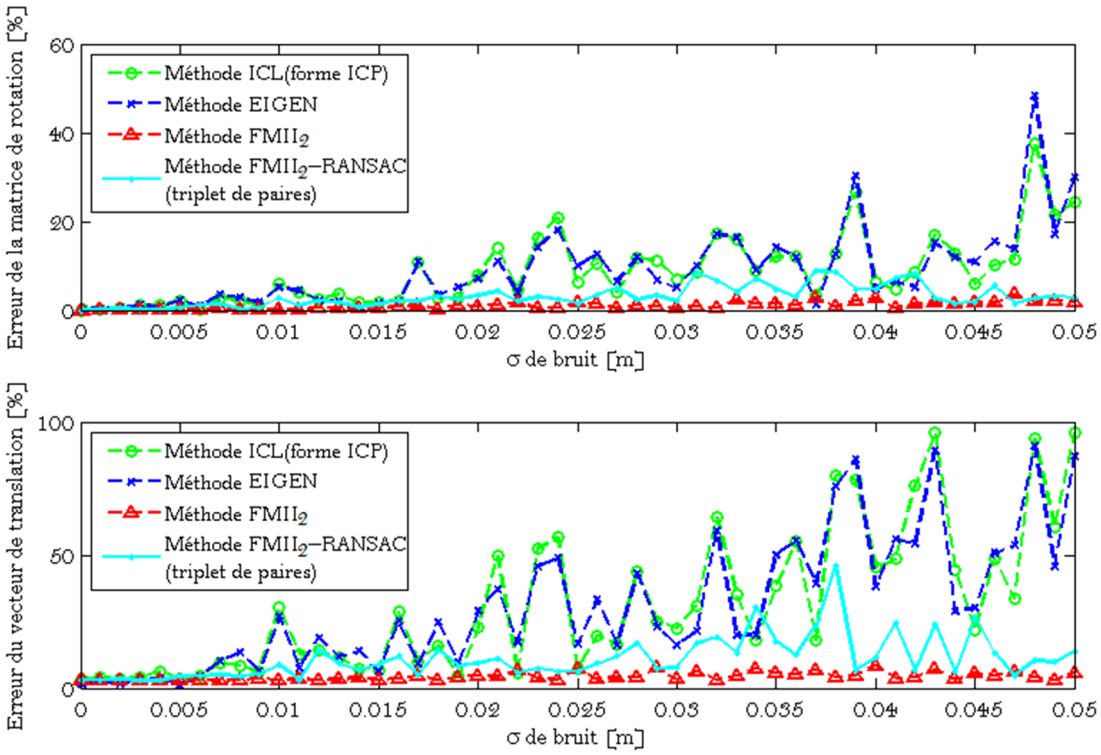

Figure 6.13 Exactitude du recalage basé sur un triplet de paires de segments (données *Test2a*)

6.5.2 Évaluation avec données simulées

La qualité des estimations obtenues par notre approche $RLMR - FMII_2$ a été vérifiée, tout d'abord, à l'aide des données simulées (*Test 2a*). La Figure 6.14 visualise le chemin détaillé de notre parcours et les résultats intermédiaires d'appariement avec leur évaluation qualitative (VP, VN, FP, FN marqués par différentes couleurs), tous les deux obtenus à partir du jeu dégénéré par le bruit gaussien dont le σ =0.03m. Conformément aux hypothèses posées, l'appariement approximatif A_{ini} contient à peine quelques Faux Positifs au détriment du nombre de Faux Négatif. Mais, il constitue une bonne base de départ pour calculer rapidement des paramètres initiaux, même si l'algorithme de *RANSAC* est généralement assez coûteux en temps de calcul.

Quant à l'appariement final A_f déterminé de nouveau à partir des segments déjà recalés grossièrement, il se caractérise par une précision beaucoup plus élevée (par exemple, dans le cas étudié Sensibilité=100%, Spécificité=99.95% alors que seulement 0.05% des combinaisons envisagées avaient été examinées de manière incorrecte). Aussi, les tests procédés sur d'autres données simulées ont confirmé que la solution proposée permet d'améliorer significativement la mise en correspondance de segments. En revanche, le résultat dépend également du taux de bruit, moyennant quoi les faux appariements peuvent être toujours présents.

Afin d'analyser l'impact de ces mauvaises paires (notamment les FP) sur la qualité des estimations, nous avons analysé les erreurs e_R et e_T calculées selon les formules (6.4) et (6.5). Ensuite, une comparaison de ces valeurs par rapport à celles obtenues par l'algorithme $FMII_2$ basé sur l'appariement précis connu a priori, a été entreprise. La Figure 6.15 présente le résultat d'une telle expérimentation. Nous pouvons constater que plus l'écart-type de bruit σ est petit, plus l'appariement est précis, et donc que les erreurs d'estimation sont du même

ordre de grandeur. Néanmoins, la méthode $RLMR - FMII_2$ paraît donner des résultats légèrement moins précis, surtout quand le σ est supérieur à 0.025m et une présence de Faux Positifs est incontestable.

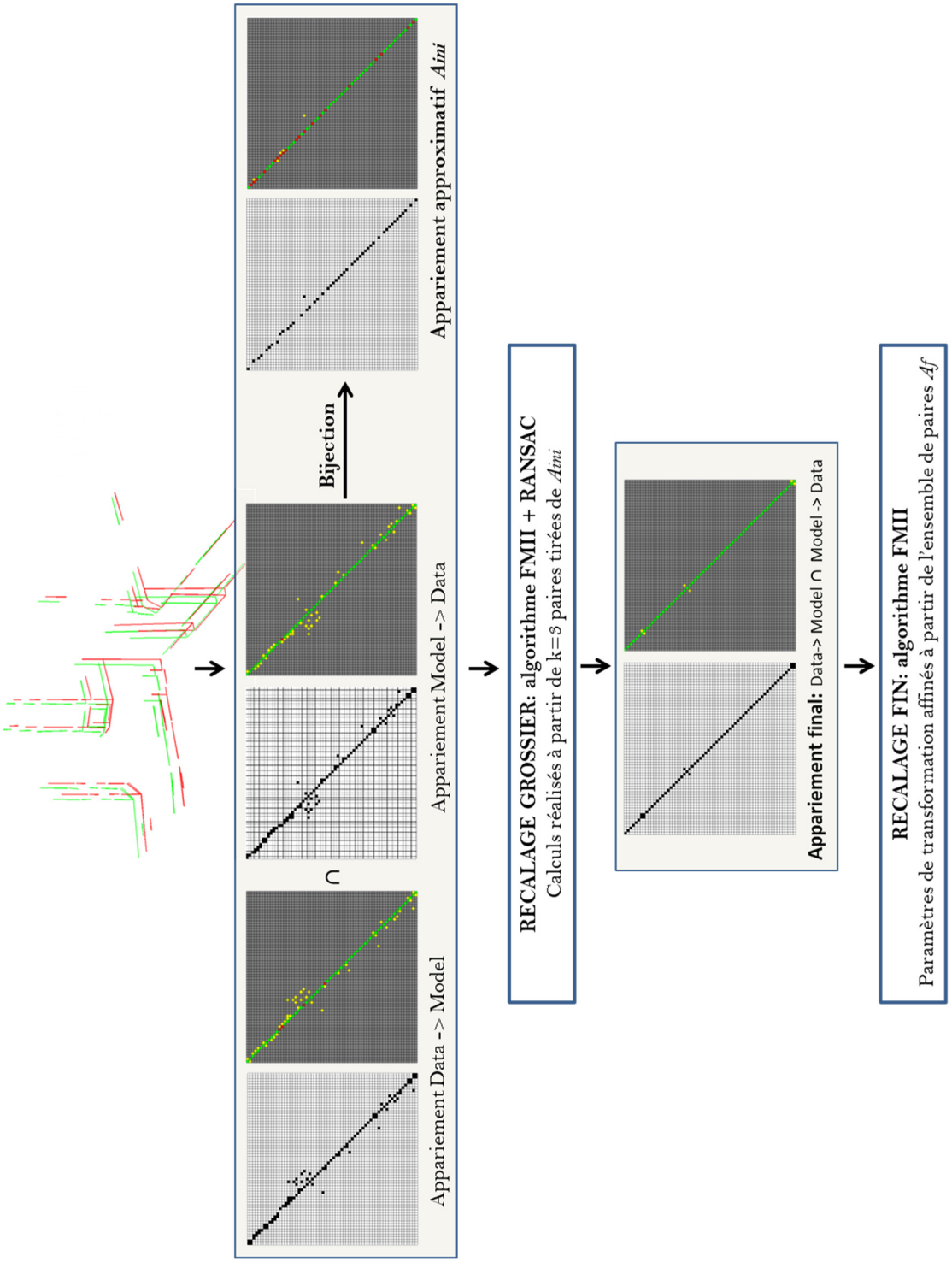

Figure 6.14 Mise en correspondance des segments *Test2a* (σ=0.03m) et le recalage rigide selon l'approche proposée $RLMR - FMII_2$. Les couleurs employées pour évaluer l'appariement : Vrai Positif(VP)-vert ; Vrai Négatif(VN) – gris ; Faux Positif(FP) – jaune ; Faux Négatif(FN) – rouge ;

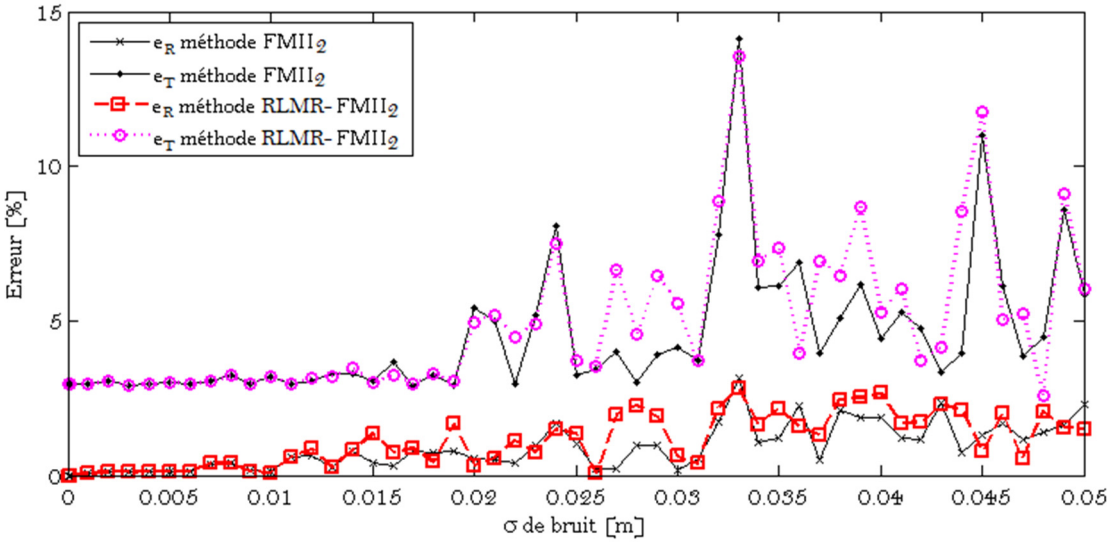

Figure 6.15 Qualité de la solution (données *Test2a*) : Algorithme $FMII_2$ vs. Approche $RLMR - FMII_2$ avec la mise en correspondance de segments effectuée simultanément

L'histogramme de répartition de différences d'erreurs d'estimation créé à partir de la Figure 6.15 montre que pour 95% des valeurs de σ, les différences de e_R sont comprises dans un intervalle de [-1.28% ; 1.28%], tandis que celles de e_T sont dans un intervalle de [-2.82% ; 2.82%] (Figure 6.16).

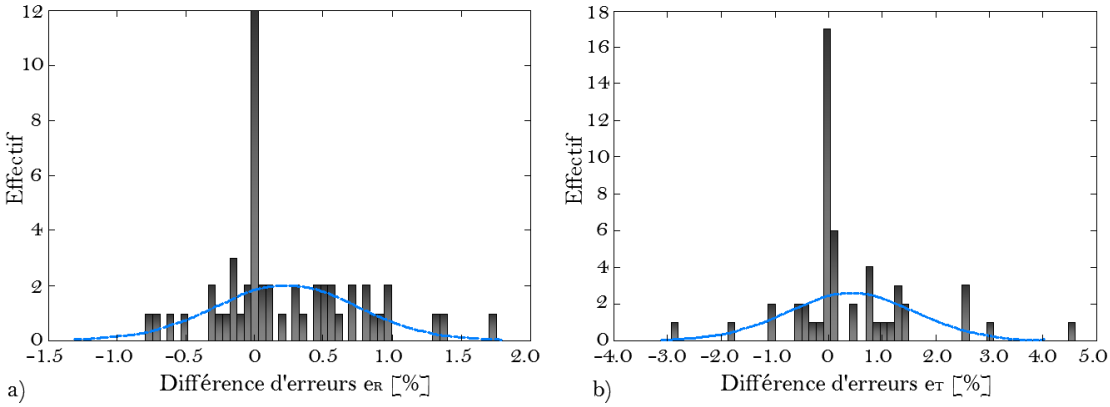

Figure 6.16 Histogramme de différences d'erreurs

Pour mieux comprendre la grandeur de cette imprécision d'estimation de paramètres sur la position relative des segments recalés, nous avons déterminé la distance *LHD*. La Figure 6.17 illustre les différences entre la distance calculée après le recalage basé sur l'algorithme $FMII_2$ et celle obtenue après avoir mis en œuvre le $RLMR - FMII_2$. Ces valeurs sont négligeables et ne dépassent pas 0.004m, même si la présence de quelques paires erronées avait été constatée. Notre algorithme, effectuant à la fois l'appariement des entités linéaires et l'estimation des paramètres de pose, donne des résultats fiables. En ce qui concerne la mise en correspondance A_f, elle se caractérise d'une précision élevée, atteinte grâce au recalage grossier convenable.

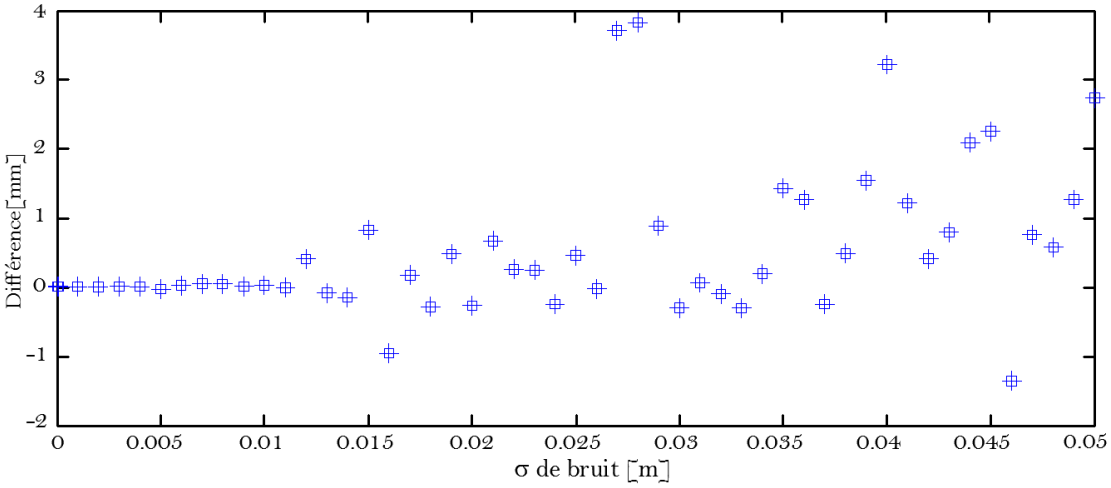

Figure 6.17 Vérification des différences de distances *LHD*

6.5.3 Tests sur données réelles

La performance de l'approche $RLMR - FMII_2$ a été aussi vérifiée à l'aide de données réelles (voir Figure 5.13). Le Tableau 6.2 représente les valeurs numériques (les paramètres de transformation estimée, l'éloignement des deux nuages de points, le temps de calcul, etc.) obtenues avec ces segments. À cet effet, les relevés laser mobiles de différents passages ont été recalés entre eux (S1_S2 ; S1_S2 ; S2_S3), ainsi qu'avec le nuage de points statique (F) dans l'intention de corriger leur position absolue. À chaque fois, une amélioration importante de l'appariement initial A_{ini} a été relevée (Figure 6.18). De plus, la vérification d'ajustement des nuages de points après les deux recalages successifs grossier et fin a été réalisée à travers la distance *OLHD*. Les valeurs obtenues confirment que la méthode proposée permet d'estimer une transformation optimale, à condition que les angles de rotation et la distance réciproque soient petits. Cette modalité est aisée à remplir puisque les nuages de points issus d'une plate-forme mobile sont déjà géo-référencés dans un système de référence commun. Dès lors, leur position relative est assez proche et l'orientation pratiquement parallèle. Par conséquent, la suppression d'erreur systématique, plus précisément d'erreur de mesure restant constante pendant l'acquisition, peut être corrigée par la méthode de recalage proposée.

Pour justifier notre démarche nécessitant la mise en correspondance des segments à deux reprises (d'abord l'appariement approximatif A_{ini} suivi par l'appariement affiné A_f), nous avons également envisagé une solution consistant à appliquer l'algorithme $FMII_2$ directement sur l'ensemble de paires A_{ini}. La dernière ligne du Tableau 6.2 contient la distance *LHD* déterminée lors d'un tel recalage. Celle-ci est plus importante que l'écart atteint avec la méthode proposée $RLMR - FMII_2$. Nous pouvons noter que l'utilisation de l'algorithme *RANSAC* habituellement très coûteux en temps de calcul a été compensée par la mise en correspondance explicite des segments. La consolidation précise de nuages de points devient possible.

Quant au temps de calcul, il est de l'ordre de dizaines de secondes (code non optimisé). Le plus important a été observé lors du recalage avec les segments S2. Mais, c'est un cas particulier car d'une côté la zone de chevauchement entre ces deux nuages est petite. De l'autre côté, la qualité du nuage S2 est faible (comme il est démontré au Chapitre 5) et il devient difficile à trouver un échantillon optimal satisfaisant aux exigences imposées (les mêmes pour tous les jeux de données). D'où un grand nombre d'itérations de *RANSAC*.

Tableau 6.2 Résultats de l'approche combinée

Jeu de données	S1-S2	S1-S3	S2-S3	S1-F	S2-F	S3-F
Distance initiale [m]	**0,516**	**0,230**	**0,560**	**0,671**	**1,257**	**0,657**
Nombre d'itérations	248	58	563	88	1640	109
Distance après le recalage grossier [m]	0,346	0,223	0,481	0,266	0,451	0,320
Distance FINALE [m]	**0,304**	**0,213**	**0,377**	**0,189**	**0,220**	**0,140**
Angle d'Euler estimés [°]:						
Ω	0,35196	-0,15530	0,68971	0,30821	-0,00763	0,03841
Φ	-0,12854	-0,00760	-0,19781	-0,17301	-0,14627	-0,08465
K	-0,43047	-0,21210	0,28428	-0,01031	0,09351	0,18708
Translation estimée [m]:						
Tx	0,950	-0,205	1,172	-0,646	-0,331	-0,205
Ty	-0,833	0,108	-1,048	0,014	-0,525	-0,461
Tz	0,189	0,177	-0,708	-0,494	-0,285	-0,507
CPU Time* [s]	18,1	10,4	50,4	7,8	183,6	16,4
Distance [m]: FMII2+A_{ini}	0,328	0,214	0,391	0,254	0,861	0,334

* Intel Core x2 T7600 2.33Ghz / 4Go Ram / Win7 64bits

Le pseudo-code 6.1 (Annexe E) est notre approche du recalage telle qu'elle est implémentée. Nous avons besoin en entrée de coordonnées de deux ensembles de segments : X_{Test} et X_{Model}. Aussi, quelques paramètres sont attendus notamment l'écart-type du bruit σ et le nombre d'un échantillon minimal (k), tous les deux indispensables pour initialiser l'estimateur de *RANSAC*. L'algorithme rend une solution optimale, c'est-à-dire les paramètres de transformation rigide R et T. Optionnellement, nous pouvons obtenir l'information sur l'appariement créé Cor, ainsi que les résidus calculés (distances) entre les segments recalés.

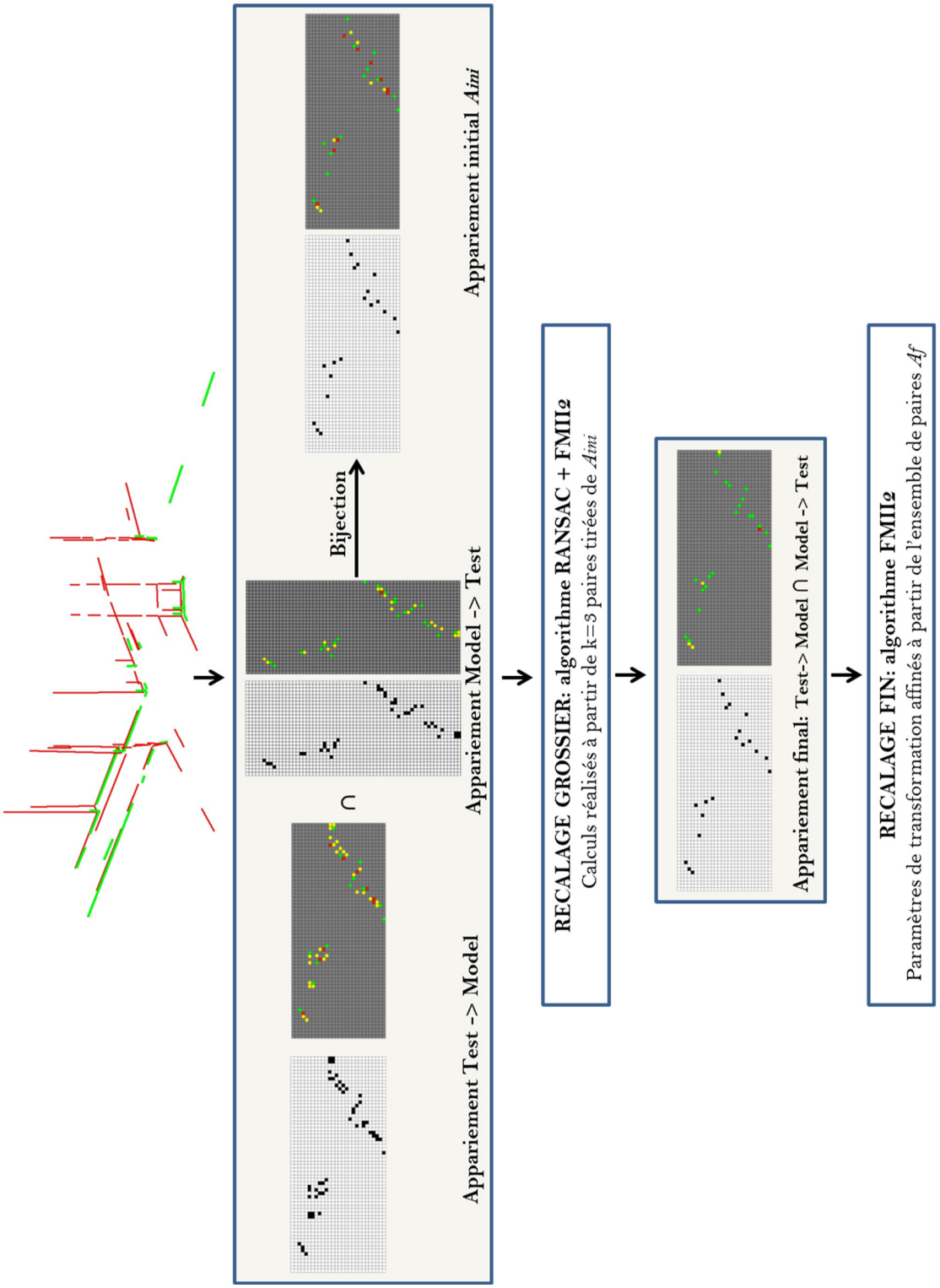

Figure 6.18 Mise en correspondance de segments réel S2 (en vert) et F(en rouge) avec l'évaluation des résultats

CHAPITRE 7

Perspectives et conclusion

Sommaire :

7.1 Conclusion **133**
7.2 Perspectives et futurs travaux **136**
7.2.1 Étalonnage extrinsèque du système : esquisse de solution 136

7.1 Conclusion

Dans cette thèse, nous nous sommes intéressés à la cartographie mobile terrestre. Notre étude se place dans le cadre du projet TerraMobilita dont l'objectif est de développer des technologies optimisées de relevé des voiries et de l'espace public avec une précision topographique élevée. Les travaux présentés proposent alors des solutions pour la qualification des nuages de points collectés, ainsi que des méthodes de recalage de données multi-sources. Nous nous sommes également penchés sur l'étalonnage du système intégral. Ce manuscrit se divise en quatre parties principales : 1) L'acquisition de nuages de points mobiles et l'étude bibliographique sur les erreurs ayant un impact sur leur qualité ; 2) L'extraction d'arêtes ; 3) La qualification à partir d'entités linéaires ; 4) Le recalage rigide basé sur des arêtes.

Nous avons commencé par présenter de manière générale les différents systèmes de numérisation 3D mobile terrestre afin de nous focaliser sur les systèmes et les données utilisées dans cette thèse. La problématique d'évaluation de l'exactitude des données mobiles et des protocoles de contrôle de la qualité géométrique sont ensuite abordés en détail. Après, de différentes anomalies entachant le fonctionnement d'un système de cartographie mobile ont été analysées. Le but a été d'aider à établir une connaissance profonde des facteurs primordiaux affectant les nuages de points. Nous pouvons ici citer à titre d'exemple la précision du système de positionnement hybride à base de *GNSS* et *INS*, l'angle d'incidence du rayon laser, la distance scanner-objet, le manque d'étalonnage juste (extrinsèque ou intrinsèque). La prise en conscience de leur nature ouvre également la porte à des améliorations et des développements possibles. Cette partie fortement théorique nous a permis de converger vers des solutions innovantes d'évaluation et de correction de données laser produites. Nous avons ainsi présenté nos travaux se rapportant à la qualification et l'amélioration du système de balayage laser

mobile. Les méthodes que nous avons adoptées sont celles basées sur les entités linéaires. D'ailleurs, toutes les parties de notre manuscrit ont été accompagnées par un état de l'art exhaustif permettant de repérer la place de notre travail par rapport à l'état actuel de la recherche. La Figure 7.1 nous montre un schéma général des tâches réalisées, avec extraction des entités linéaires, servant à leur tour à la qualification et au recalage. La problématique de l'étalonnage basé sur les segments reste toujours à poursuivre. Mais, une procédure possible à mettre en place est décrite ci-dessous.

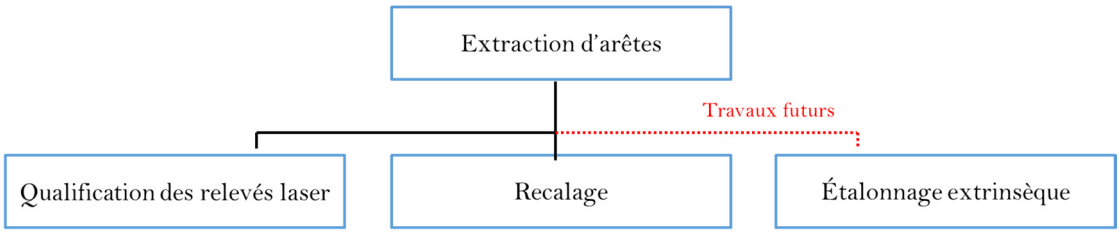

Figure 7.1 Schéma des travaux réalisés et travaux futurs

La deuxième partie de notre thèse se focalise sur l'extraction des arêtes à partir de nuage de points de densité faible. Même si la détection de lignes est une tâche primordiale dans la modélisation 3D, les travaux existants ne sont pas nombreux. Certains algorithmes permettent de détecter des ouvertures en forme de fenêtre et de porte (arêtes de pas). Une autre possibilité est d'examiner la normale au point. Le changement brusque de son orientation permet de constater la présence d'intersection de deux surfaces adjacentes. Néanmoins, l'arête reconstituée à l'aide de telles méthodes sera d'autant plus proche de l'arête réelle que la densité du nuage est élevée. Puisque notre but est de parvenir à extraire des segments précis, quelle que soit la densité du nuage, nous avons développé une méthode de détection des plis (intersections entre segments plans principaux de la scène). Celle-ci comporte deux étapes : 1) la segmentation en éléments plans via l'algorithme de *RANSAC* enrichi d'une analyse des composantes connexes ; 2) la reconstruction des arêtes. En ce qui concerne l'étude de la connectivité et de la sémantique, nous avons proposé une nouvelle méthode fondée sur la théorie des graphes. Le nuage est structuré en tenant compte du voisinage local de chaque point. Puis, la décomposition de *Dulmage-Mendelsohn* est effectuée. Les résultats obtenus avec des relevés laser mobiles confirment que cette méthode d'extraction de composantes connexes est efficace et robuste. Son étape la plus coûteuse en temps de calcul est celle de la détermination du voisinage. Concernant la performance de l'algorithme de *RANSAC* implémenté dans l'intention d'isoler les plans principaux, elle dépend de deux grandeurs : du taux de bruit et de la densité du nuage. Il se peut alors que certains plans, les plus souvent perpendiculaires à la direction de la plate-forme, ne soient pas détectés puisque ces sous-nuages se caractérisent par une densité plus faible et par un bruit important. Ce constat signifie pour nous qu'un nombre d'arêtes ne sera jamais extrait, du fait de l'incomplétude des plans fournis par l'algorithme de *RANSAC*.

Par la suite, nous avons élaboré une démarche de qualification, en termes d'exactitude et de précision, de relevés laser mobiles. Certes, l'évaluation des nuages de points pose des défis particuliers en raison de la nature de données traitées. Plusieurs méthodes portant un intérêt sur l'exactitude ont été déjà discutées, toutes nécessitant une référence externe au système d'acquisition. Les travaux aboutissent soit à une analyse de coordonnées des points de contrôle, soit à une mesure d'éloignement dérivée grâce à des approches traitant du problème de manière plus globale (algorithme de type *ICP*). Dans le cas d'une comparaison de cibles, la résolution limitée de nuage mobile peut empêcher l'identification précise de détails. Cette tâche étant fastidieuse, son automatisation est difficile à atteindre. Quant à la qualification basée sur l'*ICP*, elle ne vérifie que l'exactitude relative puisque la valeur de qualité obtenue représente la distance résiduelle entre deux ensembles de données. De plus, la mise en œuvre de l'*ICP* exige

que les données à comparer soient à peu près alignées. Étant donné que nous avons souhaité déterminer l'exactitude absolue, l'évaluation des relevés laser par recalage demeure inappropriée. Nous avons proposé une procédure de qualification basée sur des entités linéaires. Notre objectif a été de développer une solution entièrement automatique. L'éloignement de deux sous-ensembles d'arêtes, exprimé en tant que la distance de *Hausdorff* modifiée adaptée aux lignes est notre indicateur de qualité. Mais, d'abord nous apparions, entre deux nuages, les segments les plus proches au sens de la métrique définie. Nous prenons en compte la distance et l'angle entre toutes les combinaisons de segments afin de déterminer, pour chaque couple de segment, un score. Ce dernier est alors composé d'un écart angulaire, d'un indicateur de recouvrement et de la distance entre les segments parallélisés. La mise en correspondance entre les segments de référence et les segments extraits des scans se fait via le seuillage d'une matrice de similarité dont les coefficients sont un score mentionné. Les données réelles ont permis d'observer que notre algorithme d'appariement devient de plus en plus robuste lorsque l'éloignement est moindre. Nous admettons, avec une marge importante, que la distance réciproque inférieure à 2.5 m garantit un taux élevé de bons appariements. Puisque l'exactitude des données produites par des systèmes mobiles récents est loin d'être métrique, cette contrainte de distance sera toujours satisfaite. La mise en correspondance peut ainsi être précisément effectuée, permettant une qualification robuste et efficace. Quant à la métrique de *Hausdorff* introduite, elle est tout à fait adaptée pour mesurer l'écart géométrique entre segments potentiellement homologues. De plus, elle se caractérise par une propriété intéressante. Nous constatons que sa grandeur est proportionnelle à l'écart-type de bruit entachant des arêtes. Par conséquent, la connaissance de la distance entre deux ensembles de segments permet d'estimer l'incertitude associée aux extrémités et vice versa. Résumant, la méthode proposée paraît pertinente, et également transposable en photogrammétrie notamment pour l'évaluation de la qualité d'une stéréorestitution ou de modèles 3D. Notre solution peut être perçue comme un compromis entre les deux familles d'algorithmes de qualification : ceux profitants de cibles ou de détails « naturels », et ceux se référant au recalage.

Nous nous sommes ensuite intéressés aux problèmes de recalage rigide basé, lui aussi, sur les entités linéaires. Une grande partie de ce chapitre traite la problématique d'estimation des paramètres de transformation rigide à partir de lignes. Nous avons porté un regard critique sur quelques méthodes conçues à cet effet. Les analyses effectuées ont eu pour but de vérifier leur efficacité et leur précision vis-à-vis de notre problématique. Il s'agit de corriger le décalage (*shift*) survenu entre les nuages redondants acquis par une plate-forme mobile terrestre. Il en ressort que :

1) l'utilisation de primitives linéaires telles que les arêtes de plis (intersections de plans) est appropriée pour les environnements urbains et permet la réduction du nombre de cibles nécessaires ;
2) le gain de précision du recalage dépend de l'incertitude d'une ligne 3D qui, quant à elle, est due à la présence du bruit, et peut varier d'une représentation de droite à une autre ;
3) la mise en correspondance explicite des entités linéaires joue un rôle crucial dans le processus de recalage ;
4) le bruit affectant les extrémités de segments 3D peut provoquer des difficultés de convergence lors de l'optimisation de contraintes sur ces lignes ;
5) en l'absence de précision requise, le recalage basé sur les segments peut toujours servir à trouver les estimées initiales, indispensables pour initialiser certains algorithmes.

Sur cette base, nous avons proposé d'étendre l'étude à la chaîne complète comprenant une mise en correspondance suivie par une estimation des paramètres de recalage à travers de segments. Une nouvelle approche nommée $RMLR - FMII_2$ est développée, puis évaluée. Nous avons testé

sa robustesse face à différentes valeurs du bruit ajoutées aux extrémités et ayant l'effet sur l'incertitude d'une ligne.

Par ailleurs, pour diminuer les anomalies affectant la génération de données laser 3D, nous nous sommes penchés sur l'étalonnage extrinsèque. Nous avons étudié, si et dans quelle mesure, les arêtes doivent être prises en compte pour déterminer la *matrice de boresight*. Les résultats d'une analyse préliminaire étant prometteurs, cet axe de recherche mérite d'être approfondi. Avant tout, des solutions dédiées aux systèmes terrestres équipés de scanners de profil 2D doivent être développées.

En conclusion, les arêtes sont appropriées pour qualifier et recaler les relevés laser mobiles acquis en environnement urbain. Étant donné qu'avec le fort développement des *SIG*, les bases de données vectorielles sont devenues de plus en plus nombreuses (et souvent accessibles en tant que l'*Open Data* - données librement réutilisables), il serait envisageable de s'en servir lors du processus de qualification. Nous avons donc démontré notre hypothèse avancée tout au début de ce manuscrit.

7.2 Perspectives et futurs travaux

Au terme de ce travail, nous pouvons envisager plusieurs pistes de recherche qui mériteraient d'être approfondies. Tout d'abord, il pourrait être intéressant de comparer notre méthode d'extraction des composantes connexes non seulement avec d'autres algorithmes récursifs de parcours d'un graphe (*DFS*, *BFS*), mais aussi avec la méthode aboutissant à convertir un nuage de points 3D en image 2D.

Puisque la justesse et le nombre de segments extraits par notre algorithme sont liés à la qualité de plans détectés auparavant, il paraîtrait aussi nécessaire de rendre l'étape de segmentation en plans encore plus efficace. Cette remarque est surtout pertinente quand les données peu denses sont traitées. L'une des solutions possibles serait de diviser un nuage en petits blocs et d'y exécuter l'algorithme proposé.

Étant donné que la distance moyenne *LHD* évaluant la qualité des nuages à partir de deux ensembles de segments non seulement représente l'exactitude, mais aussi l'erreur survenue lors de la procédure d'extraction des arêtes, il serait judicieux de caractériser son apport à la grandeur de qualité obtenue.

Enfin, d'autres voies de recherche peuvent être favorisées, telles que l'étalonnage extrinsèque basé sur des lignes. Des solutions adéquates aux scanners laser à double balayage (3D), mais aussi celles dédiées aux scanners de profil 2D devraient être avancées.

7.2.1 Étalonnage extrinsèque du système : esquisse de solution

Concernant l'étalonnage extrinsèque du système mobile doté d'un scanner 3D, nous présentons une étude théorique, exigeant d'être encore approfondie et testée sur des données réelles. Aussi, la précision des paramètres obtenus de cette façon devrait être confrontée avec d'autres méthodes. En revanche, si le système mobile commercial est étalonné, une comparaison avec l'étalonnage renseigné par le fabricant sera appréciée. Nous cherchons une solution pour laquelle aucun déploiement antérieur de cibles n'est demandé. Une seule contrainte est de balayer un endroit riche en entités linéaires orientées différemment. La référence fiable X_{REF}^{M} peut être établie après l'acquisition. Outre, des arêtes de pli doivent être extraites à partir du nuage mobile et comparer par la suite avec celles de référence. Nous supposons que les données de trajectographie sont bonnes et le nuage de points lié au repère scanner X_{laser} est fiable (étalonnage intrinsèque connu).

Nous nous référons, pour passer du repère laser (s) aux coordonnées terrain (M), à la formule (2.1) du géo-référencement direct qui s'écrit en coordonnées homogènes :

$$X_M = \mathrm{T}_b^M \cdot \mathrm{T}_s^b \cdot X_{laser} \tag{7.1}$$

où :
$\mathrm{T}_b^M = \begin{bmatrix} R_b^M & r_{b/GNSS}^M \\ 0 & 1 \end{bmatrix}_{4x4}$: position et altitude de l'unité inertielle *IMU* dans un repère terrain (M), connues grâce à la fusion *GNSS/INS* ;
$\mathrm{T}_s^b = \begin{bmatrix} R_s^b & r_{b/s}^b \\ 0 & 1 \end{bmatrix}_{4x4}$: position et altitude du scanner laser dans le repère *IMU* (b), les paramètres d'étalonnage extrinsèque comprenant la *matrice de boresight* R_s^b et le bras de levier $r_{b/s}^b$;
X_{laser} : points laser dans le repère capteur (s) en coordonnées homogènes $[X, Y, Z, 1]^t$

Sur cette base, la formule pour passer des données laser X_{laser} au repère *IMU* (b) est suivante :

$$X_{IMU} = \widehat{\mathrm{T}_s^b} \cdot X_{laser} \tag{7.2}$$

Le nuage de points est généré, en tenant compte des valeurs initiales d'étalonnage extrinsèque $\widehat{\mathrm{T}_s^b}$ assez proche de celui que l'on cherche à déterminer. Nous visons ensuite à raffiner ces paramètres à travers une comparaison avec des données extérieures au système d'acquisition.

En admettant que nous disposions d'une référence X_{REF}^M plus précise que le nuage mobile, et dont les coordonnées sont connues dans un repère global (M), sa position dans le repère *IMU* (b) se calcule comme suit :

$$X_{REF}^b = (\mathrm{T}_b^M)^{-1} \cdot X_{REF}^M \tag{7.3}$$

Par conséquent, plus les deux nuages X_{IMU} et X_{REF}^b sont proches, plus l'étalonnage extrinsèque $\widehat{\mathrm{T}_s^b}$ est bon. Puisqu'ils représentent la même scène, nous pouvons déterminer les paramètres de transformation rigide du nuage X_{IMU} au nuage X_{REF}^b. À cet effet, nous suggérons d'effectuer un ajustement de deux ensembles de lignes à l'aide de l'algorithme de recalage proposé dans la section précédente : $RLMR - FMII_2$. Néanmoins, rien d'empêche d'utiliser un autre algorithme semblable à l'*ICP*. En sortie, nous obtenons toujours une matrice de rotation R et un vecteur de translation T permettant de rapprocher, de manière optimale, des données :

$$X_{REF}^b = \delta\mathrm{T} \cdot X_{IMU} \tag{7.4}$$

où $\delta\mathrm{T} = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}_{4x4}$.

De l'autre côté, si nous connaissions de vrais paramètres d'étalonnage T_s^b, la position des données laser dans le repère *IMU* (b) devrait coïncider à celle de référence :

$$X_{REF}^b = \mathrm{T}_s^b \cdot X_{laser} \tag{7.5}$$

En tant que les formules (7.4) et (7.5) déterminent la même grandeur, nous arrivons à formuler la relation :

$$\delta\mathrm{T} \cdot X_{IMU} = \mathrm{T}_s^b \cdot X_{laser} \tag{7.6}$$

En remplaçant, dans la formule (7.6), les coordonnées du nuage de points liées au repère *IMU* par (7.2), nous obtenons :

$$\delta \mathrm{T} \cdot \widehat{\mathrm{T}_s^b} \cdot X_{laser} = \mathrm{T}_s^b \cdot X_{laser} \tag{7.7}$$

Allons plus loin, la dépendance entre l'étalonnage extrinsèque initial et celui recherché est exprimée par :

$$\delta \mathrm{T} \cdot \widehat{\mathrm{T}_s^b} = \mathrm{T}_s^b \tag{7.8}$$

La formule (7.8) peut s'écrire également sous forme matricielle :

$$\begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}_{4x4} \cdot \begin{bmatrix} \widehat{R_s^b} & \widehat{r_{s/b}^b} \\ 0 & 1 \end{bmatrix}_{4x4} = \begin{bmatrix} R_s^b & r_{s/b}^b \\ 0 & 1 \end{bmatrix}_{4x4} \tag{7.9}$$

Nous en déduisons, par multiplication de matrices, que les valeurs initiales $\left(\widehat{R_s^b}, \widehat{r_{b/s}^b}\right)$ de l'étalonnage extrinsèque pris en compte au départ doivent être corrigées de la manière suivante :

$$R_s^b = R \cdot \widehat{R_s^b} \tag{7.10}$$

$$r_{s/b}^b = R \cdot \widehat{r_{s/b}^b} + T$$

La Figure 7.2 récapitule toutes les démarches de la procédure proposée qui mènent à déterminer l'étalonnage extrinsèque du système en partant de ses valeurs approchées. L'idée consiste alors à effectuer des scans avec un scanner laser 3D travaillant en mode statique et monté sur la plate-forme mobile. Les acquisitions depuis plusieurs points de vue ou bien encore au sein de différentes zones, sont prévues à réaliser pour améliorer la précision des résultats. Ensuite, la moyenne des paramètres calculés indépendamment lors de chaque stationnement décrit l'étalonnage extrinsèque $\left(R_s^b, r_{s/b}^b\right)$ (7.10). Conceptuellement, une telle approche peut être alors assimilée au procédé « 2-step » étalonnant les systèmes mobiles basés sur des caméras.

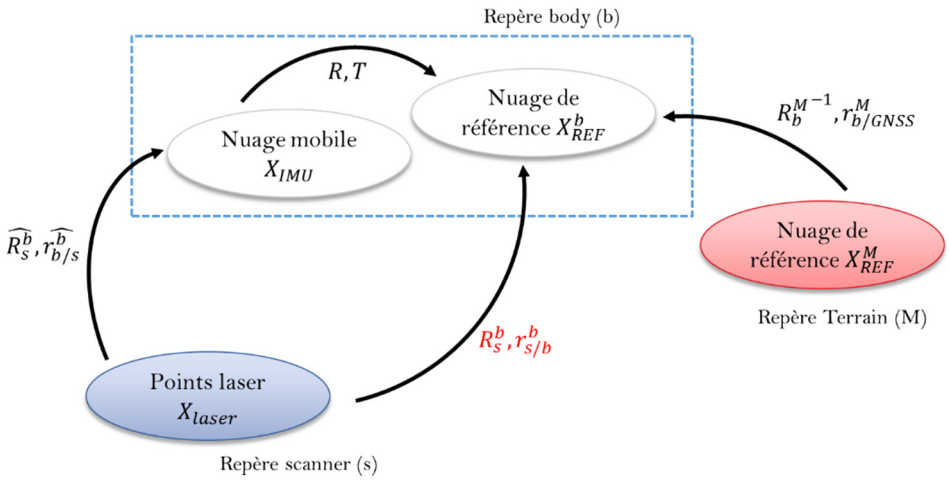

Figure 7.2 Organigramme de l'approche envisagée. Paramètres recherchés : $R_s^b, r_{s/b}^b$

Publications

Publications dans des revues avec comité de lecture et dans des conférences avec actes

- **Articles liés à la thèse :**

Poreba M., Goulette F., 2014. A linear feature-based solution for robust and accurate registration of point clouds. Sensors (en cours de soumission).

Poreba M., Goulette F., 2014. Recalage rigide de relevés laser par mise en correspondance robuste basée sur des segments. Revue Française de Photogrammétrie et de Télédétection RFPT, vol.207, (accepté).

↦ Deuxième Prix Étudiant du meilleur article soumis en 2013 à la RFPT, décerné par la **Société Française de Photogrammétrie et de Télédétection** (SFPT)

Poreba M., Goulette F., 2013. Line Segment-based Approach for Accuracy Assessment of MLS point clouds in Urban Areas. Conference Proceedings of the 8th International Symposium on Mobile Mapping Technology MMT 2013, 1-3 May, Tainan, Taïwan.

↦ Prix internationaux du **Best Student Paper Award** et du **Best Presentation Award**

Poreba M., Goulette F., 2012. RANSAC algorithm and elements of graph theory for automatic plane detection in 3D point clouds. Archives of Photogrammetry, Cartography and Remote Sensing, vol. 24, pp.301-310.

Poreba M., Goulette F., 2012. Assessing the Accuracy of Land-Based Mobile Laser Scanning Data. Geomatics and Environmental Engineering, vol. 6/3, pp.83-91.

Pyka K., Borowiec N., Poreba M., Slota M., Kundzierewicz T., 2012. Airborne laser scanning data for railway lines survey. PAK Measurement Automation and Monitoring, vol.58, pp.260-263.

- **Articles antérieurs :**

Poreba M., 2011. Use of integrated GPS and INS systems in aerial photogrammetry. Geomatics and Environmental Engineering, vol.5, no.3, pp.79-87.

Poreba M., 2009. Modern methods of earth mass volume determination. Archives of Photogrammetry, Cartography and Remote Sensing, vol.19, pp.351-361.

Séminaires sans actes

Marcotegui, B., Serna, A., Goulette, F., Poreba, M., Frauciel, L., Mittet, M.A., Rialland, S., Benjemaa, R., Hervieu, A., Soheilian, B., Paparoditis, N., 2013. TerraMobilita : urban scene analysis. 13th International Scientific and Technical Conference « From imagery to map: digital photogrammetric technologies », 23-26 September 2013, Fontainebleau, France.

Poreba, M., Goulette, F., 2013. Registration of 3D point clouds using extracted line segments. VIII Colloque national en Géomatique intitulé «Géo - informatique comme un outil intégré d'analyses spatiales », 11-13 Septembre 2013, Université de Varsovie, Varsovie, Pologne.

Poreba, M., Goulette, F., 2012. Extraction d'arêtes et comparaison avec des données de référence pour la qualification de relevés laser mobiles, Journée du GDR ISIS, Thème B – Analyse de scènes en image et vision, 8 Novembre 2012, Télécom ParisTech, Paris, France.

Bibliographie

Abbas, I., Grussenmeyer, P., Hottier, P., 1995. Contrôle de la planimétrie d'une base de données vectorielles : une nouvelle méthode basée sur la distance de Hausdorff : la méthode du contrôle linéaire. Revue de la Société Française de Photogrammétrie et de Télédétection (1995-1), n°137, ISSN 0244-6014, pp. 6-11.

Abuhadrous, I., 2005. Système embarqué temps réel de localisation et de modélisation 3D par fusion multi-capteur. Thèse de Doctorat, MINES ParisTech.

Al-Durgham, M., Detchev I., Habib, A., 2011. Analysis of Two Triangle-Based Multi-Surface Registration Algorithms of Irregular Point Clouds. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XXXVIII-5/W12, pp.61-66.

Alshawa, M., 2006. Consolidation des nuages de Points en Lasergrammétrie Terrestre. Thèse de Master, École Nationale Supérieure d'Architecture de Nancy.

Alshawa, M., 2010. Contribution à la cartographie mobile : développement et caractérisation d'un système base sur un scanner laser terrestre. Thèse de Doctorat, INSA Strasbourg.

Arun, K.S., Huang, T.S., Blostein, S.D., 1987. Least-Squares Fitting of Two 3-D Point Sets. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 9, No.5. pp. 698-700.

Atanacio-Jiménez, G., Gonzalez-Barbarosa, J-J., Hurtado-Ramos, J.B., Ornelas-Rodriguez, F.J., Jiménez-Hernandez, H., Garcia-Ramirez, T., Gonzalez –Barbarosa, R., 2011. LIDAR Velodyne HDL-64E Calibration Using Pattern Planes. International Journal of Advanced Robotic Systems, vol.8, no.5, pp.70-82.

Awwad, T. M., Zhu, Q., Du, Z., Zhang, Y., 2010. An improved segmentation approach for planar surfaces from unstructured 3D point clouds. The Photogrammetric Record, 25(129), pp. 5-23.

Babu, R., Wang, J., 2005. Ultra-Tight GPS/INS/PL integration: a system concept and performance analysis. GPS Solution, vol.13, pp.75-82.

Barber, D., Mills, J., 2008. Geometric validation of a ground-based mobile laser scanning system. ISPRS Journal of Photogrammetry & Remote Sensing, Vol. 63, pp.128-141.

Bauer, J., Karner, K., Schindler, K., Klaus, A., Zach, C., 2005. Segmentation of building from dense 3D point-clouds. Proceedings of the ISPRS Workshop Laser scanning, September 12-14, Enschede, Netherlands.

Belton, D., Mooney, B., Snow, T., Kwang-Ho, B., 2011. Automated Matching of Segmented Point Clouds to As-built Plans. Proceedings of the Surveying & Spatial Sciences Biennial Conference, 21-25 November, Wellington, New Zealand.

Benhabiles, H., Vanderborr, J-P., Lavoué, G., Daoudi, M., 2009. Une collection de modèles 3D avec vérité – terrain pour l'évaluation des algorithmes de segmentation. Proceedings of CORESA (COmpression et REpresentation des Signaux Audiovisuels), 19-20 mars 2009, Toulouse, France.

Besl, P.J., McKay, N.D., 1992. A method for registration of 3-D shapes. IEEE Transactions of Pattern Analysis and Machine Intelligence, vol. 14, no. 2, pp.239-256.

Bienert, A., 2008. Vectorization, Edge Preserving Smoothing and Dimensioning of Profiles in Laser Scanner Point Clouds. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XXXVII, part. B5, pp. 507-512.

Bitenc, M., Lindenbergh, R., Khoshelham, K., van Waarden, A.P., 2011. Evaluation of a LIDAR Land-Based Mobile Mapping System for Monitoring Sandy Coasts. Remote Sensing, vol. 3, pp. 1472-1491.

Boehler, W., Bordas, V., Marbs, A., 2003. Investigating laser scanner accuracy. Proceedings of the XIXth CIPA Symposium, 30 September – 4 October, Antalya, Turkey.

Borowiec, N., 2009. Generowanie trójwymiarowego modelu budynku na podstawie danych lidarowych. Archiwum Fotogrametrii, Kartografii i Teledetecji, vol. 20, pp.47-56.

Borowiec, N., 2012. Transformata Hough'a jako narzedzie wspomagające wykrywanie dachów budynków. Archiwum Fotogrametrii, Kartografii i Teledetekcji, vol. 25, pp.45-54.

Borrmann, D., Elseberg, J., Lingemann, K., Nüchter, A., 2011. The 3D Hough Transform for Plane Detection in Point Clouds: A Review and a new Accumulator Design. 3D Research, vol. 02, pp.1-13.

Boudet, L., 2007. Auto-qualification de données géographiques 3D par appariement multi-image et classification supervisée. Application au bâti en milieu urbain dense. Thèse de doctorat, Université Paris-EstMarne-la-Vallée.

Bouillaguet, Ch.,. 2011. Algorithmique et Programmation. Projet :permutations de matrices creuses et décompositions de graphes Cours en-ligne, École Nationale Supérieure : http://www.lifl.fr/~bouillag/teaching/ens_2011/sparse1.pdf

Boulaassal, H., 2010. Segmentation et modélisation géométrique de façades de bâtiments à partir de relevés laser terrestres. Thèse de Doctorat, INSA Strasbourg.

Brabant, M., 2011. In : Topographie opérationnelle: Mesure – Calculs – Dessins - Implémentations. Editeur Eyrolles.

Bretar, F., Roux, M., 2005. Hybrid image segmentation using lidar 3D planar primitives. Proceedings of ISPRS Workshop Laser scanning, September 12-14, Enschede, the Netherlands, 2005, pp 72-78.

Bykat, A., 1978. Convex hull of a finite set of points in two dimensions. Information Processing Letters, vol. 7, pp.296–298.

Cahalane C., McCarthy, T., McElhinney, C., 2010. Mobile mapping system performance – an initial investigation into the effect of vehicle speed on laser scan lines, RSPSoc no.2005.

Canaz, S., Habib, A., 2013. Planar and linear feature –based registration of terrestrial laser scans with minimum overlap using photogrammetric data. Proceedings of the 8th International Symposium on Mobile Mapping Technology, 1-3 May, Tainan, Taiwan.

Cannelle, B., Paparoditis, N., Pierrot-Deseilligny, M., Papelard, J-P., 2012. Off-line vs. On-line calibration of a panoramic-based mobile mapping system. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. I-3, pp. 31-36.

Castillo, E., Zhao, H.,2009. Point Cloud Segmentation via Constrained Nonlinear Least Squares Surface Normal Estimates. Computational Geometry.

Cazzaniga, N.E., Fornali, G., Roncella, R., 2007. Improving the reliability of a GPS/INS Navigation Solution for MM Vehicles By Photogrammetry. Proceedings of the 5th International Symposium on Mobile Mapping Technology, Padua, Italy.

Chan, T. O., Lichti, D. D., Feature-based self-calibration of Velodyne HDL-32E LiDAR for terrestrial mobile mapping applications. 2013. Proceedings of the 8th International Symposium on Mobile Mapping Technology, 1-3May, Tainan, Taiwan.

Chan, T-O., Lichti, D.D., Glennie, C.L., 2013. Multi-feature based boresight self-calibration of a terrestrial mobile mapping system. ISPRS Journal of Photogrammetry and Remote Sensing, vol. 82, pp.112-124.

Chen, C., Hung, Y., Cheng, J., 1999. RANSAC-based DARCES: A New Approach to Fast Automatic Registration of Partially Overlapping Range Image. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.21, no. 11, pp.1229-1234.

Chen, J., Leung, M.K., Gao, Y., 2003. Noisy logo recognition using line segment Hausdorff distance. The Journal of the Pattern Recognition Society, vol.36, pp.943-955.

Chen, Y., Medioni, G., 1992. Object modelling by registration of multiple range images. Image Vision Computing, vol. 10(3), pp.145-155.

Chen, Y-Ch., Tseng, Y-H., Wang, P-Ch., 2013. The calibration of a portable panoramic image mapping system. Proceedings of the 8th International Symposium on Mobile Mapping Technology, Mai 1-3, Tainan, Taiwan.

Choi, W-P., Lam, K-M., Siu, W-Ch., 2001. A robust Line-Feature-Based Hausdorff Distance for Shape Matching. Proceedings of Advances in Multimedia Information Processing – PCM 2001, Second IEEE Pacific Rim Conference on Multimedia, 24-26 Octobre, Bejing, Chine, pp.764-771.

Cormen, T., Leiserson, Ch., Rivest, R., Stein, C., 2001. Introduction à l'algorithmique. Cours et exercices. The Massachusetts Institute of Technology, 2e édition. pp.1-1176.

Guan, H., Li, J., Yu, Y., Wang, Ch., 2013. Geometric validation of a mobile laser scanning system for urban applications. Proceedings of the 8th International Symposium on Mobile Mapping Technology, 1-3 May, Tainan, Taiwan.

Delmerico, J.A., David, P., Corso, J.J., 2011. Building Facade Detection, Segmentation, and Parameter Estimation for Mobile Robot Localization and Guidance, Proceedings of the International Conference of Intelligent Robots and System (IROS)., pp.1632-1639.

Demarsin, K., Vanderstraeten, D., Volodine, T., Roosea, D., 2007. Detection of closed sharp edges in point clouds using normal estimation and graph theory. Computer-Aided Design, vol. 39, pp.276-283.

Deschaud, J-E., 2010. Traitement de nuages de points denses et modélisation 3D d'environnements par système mobile LiDAR/Caméra. Thèse de Doctorat, MINES ParisTech.

Douadi, L., 2006. Contribution à l'étude du recalage de données 3D/couleur. Thèse de doctorat, Université Montpellier II.

Dubuisson, M.P., Jain, A.K., 1994. A modified Hausdorff distance for object matching. Proceedings 12th International Conference on Pattern Recognition, Jerusalem, Israel, pp.566-568.

Duda, R.O., Hart, P.E., 1971. Use of the Hough Transformation to Detect Lines and Curves in Pictures. Technical Note 36, Artificial Intelligence Center, SRI International.

Feng, J., Zhong, R., Yang, Y., Zhao, W., 2008. Quality Evaluation of Spatial Point-Cloud Data Collected by Vehicle-Borne Laser Scanner. International Workshop on Education and Training & International Workshop on Geosciences and Remote Sensing.

Dulmage, A.L., Mendelsohn, N.S., 1958. Coverings of bipartite graphs. Canadian Journal of Mathematics, vol.10, pp. 517–534.

Ellum, C., El-Sheimy, N., 2002. The calibration of Image-Based Mobile Mapping Systems. Proceedings of the 6th Conference on Optical 3D Measurements Techniques, September 22-25, Zurich, Swiss.

Ellum, C., El-Sheimy, N., 2003. Investigations in boresight and lever-arm calibration.

Elsbourg, J., Borrmann, D., Nüchter, A., 2013. Automatic and Full Calibration of Mobile Laser Scanning Systems. Experimental Robotics, Springer Tracts in Advanced Robotics, Vol.88, part. XIII, pp.907-917.

El-Sheimy, N., 2011. Land-based MMT: State of the Art and Future Trends. Proceedings of the 7th International Symposium on Mobile Mapping Technology, June 13-16, Cracow, Poland.

Evans, S., 2013. 3D Laser Scanning Market Continues Trend of Robust Growth. Publication en-ligne : http://www.arcweb.com/press-center/2013-07-11/3d-laser-scanning-market-continues-trend-of-robust-growth-1.aspx (accessible le 16.04.2014).

Feng, J., Zhong, R., Yang, Y., Zhao, W., 2008. Quality Evaluation of Spatial Point-Cloud Data Collected by Vehicle-Borne Laser Scanner. Proceedings of the International Workshop on Education and Training & 2008 International Workshop on Geosciences and Remote Sensing, pp. 320-323.

Filin, S., 2003. Recovery of systematic biases in laser altimetry data using natural surfaces. Photogrammetric Engineering and Remote Sensing, vol.69(11), pp.1235-1242.

Fischler, M.A. et Bolles, R.C., 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM, vol.24(6), pp.381-395.

Fitzgibbon, A., 2001. Robust registration of 2D and 3D point sets. The British Machine Vision Conference.

Frémont, V., 2009. Odométrie 3D vision/lidar pour les véhicules intelligents. Journées Nationales de la Recherche en Robotique, Neuvy-sur-Barangeon, France.

Gallo, O., Manduchi, R., Rafii, A., 2010. CC-RANSAC: Fitting planes in the presence of multiple surfaces in range data. Pattern Recognition Letters, vol. 32, pp. 403–410.

Gandofi, S., Barbarelle, M., Ronci, E., Bruchi, A., 2008. Close photogrammetry and laser scanning using a mobile mapping system for high detailed survey of a high-density urban area. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVII, part B5, pp.909-914.

Gao, J., Sun, J., Wu, K., 2012. Image Matching Method Based on Hausdorff Distance of Neighborhood Grayscale. Journal of Information & Computational Science, vol.10, pp. 2855-2863.

Gao, Y., Leung, M.K., 2002. Line segment Hausdorff distance on face matching. The Journal of the Pattern Recognition Society, vol.35, pp.361-371.

Gelfand, N., Ikemoto, L., Rusinkiewicz, Sz., Levoy, M., 2003. Geometrically Stable Sampling for the ICP Algorithm. 2003. Proceedings of the 4th International Conference on 3D Digital Imaging and Modeling (3DIM), pp.260-267.

Ghanma, M., 2006. Integration of Photogrammetry and LIDAR. Thèse de doctorat, Université de Calagary, Canada.

Glennie, C., 2007. Rigorous 3D error analysis of kinematic scanning LIDAR systems. Journal of Applied Geodesy, vol.1, pp.147-157.

Glennie, C., 2012. Calibration and kinematic analysis of the Velodyne HDL-64E S2 LiDAR sensor. Photogrammetric Engineering and Remote Sensing, vol. 78(4), pp.339-347.

Graham, R.L., 1972. An Efficient Algorithm for Determining the Convex Hull of a Finite Planar Set. Information Processing Letters, vol. 1, pp.132-133.

Grejner-Brzezinska, D., Toth, Ch., 2010. High-Accuracy Multi-Sensor Geolocation Technology to Support Geophysical Data Collection at MEC Sites. Final Report SERDP Project MR-1564

Gressin, A., Mallet, C., David, N., 2012. Improving 3D LIDAR point cloud registration using optimal neighborhood knowledge. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. I-3, pp. 111-116.

Gressin, A., Mallet, C., Damantké, J., David, N., 2013. Towards 3D lidar point cloud registration improvement using optimal neighborhood knowledge. ISPRS Journal of Photogrammetry and Remote Sensing, vol. 79, pp. 240-251.

Gruen, A., Akca, D., 2005. Least squares 3D surface and curve matching. ISPRS Journal de Photogrammetry and Remote Sensing, vol.59(3), pp.151-174.

Grussenmeyer, P., Hottier, P., Abbas, I., 1994. Le contrôle topographique d'une carte ou d'une base de données constituées par voie photogrammétrique. Revue de l'Association Française de Topographie, XYZ 2e trim. 1994 N°59, ISSN 0290-9057, pp. 39-45.

Guerra, C., Pascucci, V., 1999. On matching Sets of 3D Segments. Proceedings SPIE, vol. 3811, pp.157-167.

Haala, N., Peter, M., Kremer, J., Hunter, G., 2008. Mobile LIDAR Mapping for 3d point cloud collection in urban areas – a performance test. Proceedings of XXIth ISPRS Congress.

Habib, A.F., Ghanma, M.S., Tait, M., 2004. Integration of LIDAR and photogrammetry for close range applications. ISPRS Proceedings of XXth Congress, July12-23, Istanbul, Turkey.

Habib, A., Bang; K., Kersting, A.P., Chow, J., 2010. Alternatives methodologies for LiDAR system calibration. Remote Sensing, vol.2, pp.874-907.

Habib, A.F., Kersting, A.P. Shaker, A., Yan, W-Y., 2011. Geometric Calibration and Radiometric Correction of LiDAR Data and Their Impact on the Quality of Derived Products. Sensors, vol.11, pp.9069-9097.

Hanke, K., Grussenmeyer, P., Grimm-Pitzinger, A., Weinold, T., 2006. First Experiences with Trimble GX Scanner. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XXXVI, part 5.

Harris, C., Stephens, M.A., 1988. A Combined Corner and Edge Detector. Proceedings of 4th Alvey Vision Conference, pp. 147-151.

Hartley, R., Zisserman, A., 2003. Multiple view geometry in computer vision. Cambrige University Press, Second edition, pp.117-121.

Hassan, T., El-Sheimy, N., 2008. System calibration of land-based mobile mapping systems. Proceedings of the International Calibration and Orientation Workshop EuroCOW 2008, 30 January-1 February, Castelldefels, Spain.

Hernandez, J., Marcotegui, B., 2008. Segmentation de Nuages de Points pour la Modélisation d'Environnement Urbains. Revue Française de Photogrammétrie et de Télédétection, vol.191, pp.28-35.

Hiremagalur, J., Yen, K.S., Lasky, T.A., Ravani, B., 2009. Testing and Performance Evaluation of Fixed Terrestrial 3D Laser Scanning Systems for Highway Applications. Transportation Research Board TBR 88th Metting CD-ROM, pp. 1-19.

Hough, P.C.V., 1962. Method and Means for Recognizing Complex Patterns. US Patent 3.069.654.

Huh, Y., Yang, S., Ga, Ch., Yu, K., Shi, W., 2013. Line segment confidence region-based string matching method for map conflation. ISPRS Journal of Photogrammetry and Remote Sensing, vol.78, pp.69-84.

Hullo, J-F., 2013. Consolidation de relevés laser d'intérieurs construits : pour une approche probabiliste initialisée par géolocalisation. Thèse de doctorat, Université de Strasbourg.

Huttenlocher, D.P., Klanderman, G.A., Kl, G.A, Rucklifge, W.J., 1993. Comparing Images Using the Hausdorff Distance. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.15, pp. 850-863.

Jarvis, R. A., 1973. On the identification of the convex hull of a finite set of points in the plane. Information Processing Letters vol.2, pp.18–21.

Jarzabek-Rychard, M., Borkowski, A., 2010. Porównanie algorytmów RANSAC oraz rosnacych płaszczyzn w procesie segmentacji danych z lotniczego skaningu laserowego. Archiwum Fotogrametrii, Kartografii i Teledetecji, vol. 21, pp.119-129.

Jaw, J-J., Chuang, T-Y., 2008. Registration of ground-based LIDAR point clouds by means of 3D line features. Journal of the Chinese Institute of Engineers, vol. 31, no. 6, pp.1031-1045.

JCGM 200 :2012. Vocabulaire international de métrologie – Concepts fondamentaux et généreux et termes associés (VIM), BIPM, pp.18-28.

Jibrini, H., 2002. Reconstruction automatique des bâtiments en modèles polyédriques 3-D à partir de donnés cadastrales vectorisées 2D et d'un copule d'images aériennes à haute résolution. Thèse de doctorat de Télécom Paris.

Kaartinen, H., Hyyppä, J., Kukko, A., Jaakkola, A., Hyyppä, H., 2012. Benchmarking the Performance of Mobile Laser Scanning Systems Using a Permanent Test Field. Sensors, vol. 12, pp. 12814-12835.

Kamgar-Parsi, B., Kamgar-Parsi, B., 1997. Matching Sets of 3D Line Segments with Application to Polygonal Arc Matching. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.19, No. 10.

Kamgar-Parsi, B., Kamgar-Parsi, B., 2004. Algorithms for Matching 3D Line Sets. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, No.5.

Klimkowska, H., Wróbel, A., 2006. Uwagi o wykorzystaniu Tachymetrów Bezlustrowych w Inwentaryzacji Architektonicznej (Some Remarks Concerning the Use of Reflector less Total Stations for Architectural Recording). Archiwum Fotogrametrii, Kartografii i Teledetekcji, vol.16, pp.297-303.

Kumari, P., Carter, W.E., Shrestha, R.L., 2011. Adjustment of systematic errors in ALS data through surface matching. Advances in Space Research, vol.47, pp. 1851-1864.

Landes, T., Grussenmeyer, P., 2011. Les principes fondamentaux de la lasergrammétrie terrestre : systèmes et caractéristiques. Revue XYZ, no. 128/3, pp.37-49.

Landes, T., Grussenmeyer, P., Boulaassal, H., 2011. Les principes fondamentaux de la lasergrammétrie terrestre : acquisition, traitement des données et applications. Revue XYZ, no. 129/4, pp.25-38.

Landes, T., Boulaassal, H., Grussenmeyer, P., 2012. Quality Assessment of geometric façade models reconstructed from TLS data. The Photogrammetric Record, no. 27(138), pp.137-154.

Lari, Z., Habib, A.F., Kwak, E., An adaptive approach for segmentation of 3D laser point clouds. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XXXVIII-5/W12, pp.103-108.

Latulippe, M., 2013. Calage robuste et accéléré de nuages de points en environnements naturels via l'apprentissage automatique. Thèse de Master, Université Laval, Québec, Canada.

Lee, D., Kim, Y., Bang, H., 2013. Vision-based Terrain Referenced Navigation for Unmanned Aerial Vehicles using Homography Relationship. Journal of Intelligent and Robotic Systems, vol.69(1-4), pp. 489-497.

Levinson, J.S., 2011. Automatic laser calibration, mapping, and localization for autonomous vehicles. Thèse de Doctorat, Stanford Artificial Intelligence Laboratory.

Li, W., Li, X., Bian, Y., Zhao, H., 2012. Multiple View Point Cloud Registration Based on 3D Lines. Proceedings of the 2012 International Conference on Image Processing, Computer Vision & Pattern Recognition, Las Vegas, USA.

Li-Chee-Ming, J., Armenakis, C., Lee, R., 2011. Mobile Stereo-Mapper: a Portable Kit for Unmanned Aerial Vehicles. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XXXVIII-1/C22, pp.1-6.

Liao, H-M., Liu, H-S., Chen, C., 2013. Develop Backpack Mobile Mapping System Based on Open Source Software and Hardware Platform. Proceedings of the 8th International Symposium on Mobile Mapping Technology, 1-3 May, Tainan, Taiwan.

Lowe, D.G., 2004. Distinctive image features from scale – invariant key points. International Journal of Computer Vision, vol.60(2), pp.91-110.

Mano, K., Ishii, K., Hirao, M., Tachibana, K., Yoshimura, M., Akca, D., Gruen, A., 2012. Empirical accuracy assessment of MMS Laser point clouds. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XXXIX-B5, pp. 495-498.

Masuda, T., Yokoya, N., 1995. A robust Method for Registration and Segmentation of Multiple Range Images. Computer Vision and Image Understanding, vol. 16, no.3, pp.295-307.

Monnier, F., Vallet, B., Paparoditis, N., Papelard, J-P., David, N., 2013. Mise en cohérence de données laser mobile sur un modèle cartographique par recalage non-rigide. Revue Française de Photogrammétrie et de Télédetection, no. 202, pp.27-41.

Narayana, K.S., 2011. Solutions for the localization of Mobile Mapping Systems in structured environments. Thèse de Doctorat, MINES ParisTech.

Newby, P. R. T., 2011. Editorial: Accuracy, precision, extraction, citation and valediction. The Photogrammetric Record, 26(134): 149-153.

Ning, F-S., Kao, S-P., Chang, Ch-Ch, Meng, X., 2007. A simulation of the effect of GPS Pseudolite observations on the obstructed sky view. Survey Review, vol. 39, pp.34-42.

Oehler, B., Stueckler, J., Welle, J., Schulz, D., Behnke S., 2011. Efficient Multi-Resolution Plane Segmentation of 3D Point Clouds. Proceedings of the 4th International Conference on Intelligent Robotics and Applications (ICIRA), Aachen, Germany.

Paparoditis, N., Papelard, J-P., Cannelle, B., Devaux, A., Soheilian, B., Nicolas, D., Houzay, E., 2012. Stéréopolis II: A multi-purpose and multi-sensor 3D mobile mapping system for street visualization and 3D metrology. Revue Française de Photogrammetry et de Télédetection, pp.69-79.

Pawleta, M., Igielska, A., 2009. Analiza dokładnosci wybranych modeli naziemnych skanerów laserowych. Thèse de Master, AGH de Cracovie.

Pesci, A., Teza, G., 2008. Effects of surface irregularities on intensity data from laser scanning: an experimental approach. Annals of Geophysics, vol. 51/1, pp.839-848.

Poncelet, N., Cornet, Y., 2010. Transformée de Hough et détection de linéaments sur images satellitaires et modèles numériques de terrain. Bulletin de la Société Géographique de Liège, vol.54, pp.145-156.

Poreba, M., Goulette, F., 2012. Assessing the Accuracy of Land-Based Mobile Laser Scanning Data. Geomatics and Environmental Engineering, vol.6, no. 3, pp.73-81.

Pothen, Al., Fan, Ch-J., 1990. Computing the Block Triangular Form of a Sparse Matrix. ACM Transactions on Mathematical Software, vol.16, no.4, pp.303-324.

Pulli, K., 1999. Multiview Registration for Large Data Sets. Proceedings of the International Conference on 3D Digital Imaging for the ICP Algorithm (3DIM), Ottawa, pp. 160-168.

Rabbani, T., van den Heuvel, F. A., Vosselman, G., 2006. Segmentation of Point Clouds using Smoothness Constraint. ISPRS Commission V Symposium on Image Engineering and Vision Metrology, pp.248-253.

Rau, J-Y., Habib, A.F., Kersting, A.P., Chiang, K-W., Bang, K-I., Tseng, Y-H., Li, Y-H., 2011a. Direct Sensor Orientation of a Land-Based Mobile Mapping System. Sensors, vol.11, pp. 7243-7261.

Rau, J-Y., Chen, L-Ch., Hsieh, Ch-Ch., Huang, T-M., 2011b. Static error budget analysis for a land-based dual-camera mobile mapping system. Journal of the Chinese Institute of Engineers, vol. 34, no. 7, pp.849-862.

Ray, J.A., Graham, L., 2008. New horizontal accuracy assessment tools and techniques for LIDAR data. Proceedings of the ASPRS Annual Conference, Portland, Oregon.

Reichle, R.H., 2008. Data assimilation Methods in the Earth sciences. Advances in Water Resources, vol.31, no.11, pp.1411-1418.

Renaudin, E., Habib, A., Kersting, A.P., 2011. Feature-Based Registration of Terrestrial Laser Scans with Minimum Overlap Using Photogrammetric Data. ETRI Journal, vol. 33, no. 4, pp.517-527.

Ridene, T., 2010. Co-recalage de données hétérogènes 3D géo-référencées: contributions à la correction de relevés laser mobiles. Thèse de doctorat, MINES ParisTech.

Rieger, P., Studnicka, N., Pfennigbauer, M., Zach, G., 2010. Boresight alignment method for mobile laser scanning systems. Journal of Applied Geodesy, vol. 4, pp. 13-21.

Rusinkiewicz, Sz., Levoy, M., 2001. Efficient Variants of the ICP Algorithms. Proceedings of the International Conference on 3D Digital Imaging and Modeling (3DIM), Québec, Canada, pp.145-152.

Shear, P., In-flight Quality Assessment and Data Processing for Airborne Laser Scanning. Thèse de Doctorat, École Polytechnique Fédérale de Lausanne.

Sappa, A., Restrepo-Specht, A., Devy, M., 2001. Range Image Registration by using an Edge-based Representation. Proceedings of the 9th International Symposium on Intelligent Robotic Systems (SIRS'01), Toulouse, France, pp.167-176.

Sarni D., Lemarchande L., 2011. Graphes, Université de Brest, cours en-ligne http://www.lisyc.univ-brest.fr/pages_perso/lemarch/Cours/polyGraphes.pdf, pp.1-63.

Selmi, I., 2013. Optimisation de de l'infrastructure d'un système de positionnement indoor à base de transmetteurs GNSS. Thèse de Doctorat, Télécom SudParis.

Serna, A., Marcotegui, B., 2013. Urban accessibility diagnosis from mobile laser scanning data, ISPRS Journal of Photogrammetry and Remote Sensing, vol.84, pp.23-32.

Shaer, P., 2010. In-Flight Quality Assessment and Data Processing for Airborne Laser Scanning. Thèse de doctorat, École Polytechnique Fédérale de Lausanne.

Shi, Y., 2008. Advanced Mobile Mapping System Development with Integration of Laser Data, Stereo Image and other Sensor Data. Publication of Tokyo City University, no.10, pp. 24-31.

Skaloud, J., Lichti, D.D., 2006. Rigorous approach to boresight self-calibration in airborne laser scanning. ISPRS Journal of Photogrammetry and Remote Sensing, vol. 61(1), pp.47-59.

Soria-Medina, A., Martinez, J., Buffara-Antunes, A.F., Arias, P., Gonzalez-Jorge, H., 2011. An approach to extracting façade features using the intensity value of terrestrial laser scanner with mathematical morphology. The 28th International Symposium on Automation and Robotics in Construction, Republic of Korea, pp.552-557.

Soudarissanane, S., Lindenbergh, R., Menenti, M., Teunissen, P.J.G., 2009. Incidence angle influence on the quality of terrestrial laser scanning points. Proceedings of ISPRS Workshop Laser Scanning; Paris, France, vol. XXXVIII, part 3/W8, pp.183-188.

Steiger, R., 2005. The Geometrical Quality of Terrestrial Laser Scanner (TLS). Proceedings of FIG Working Week, Egypt.

Sunday, D., 2001. Distance between lines and segments with their closest point of approach. Technical report, en-ligne : http://geometryalgorithms.com.

Talaya, J., Alamus, R., Bosch, E., Serra, A., Kornus, W., Baron, A., 2004. Integration of a Terrestrial Laser Scanner with GPS/IMU Orientation Sensors. Proceedings of the XX ISPRS Congress (Commission V), July 12-23, Istanbul, Turkey.

Tangelder, J.W.H., Veltkamp, R.C., 2008. A survey of content based 3D shape retrieval methods. Multimedia Tools Applications, vol.39(3), pp.441-471.

Tarsha-Kurdi, F., Landes, T., Grussenmeyer, P., 2007. Hough Transform and Extended RANSAC algorithms for automatic detection of 3D building roof planes from LIDAR data. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol.XXXVI, part 3/W52, pp.407-412.

Tarsha-Kurdi, F., 2008. Extraction et reconstruction de bâtiments en 3D à partir de relevés lidar aéroportés. Thèse de doctorat, Université de Strasbourg.

Titterton, D.H., Weston, J.L., 2004. Strapdown Inertial Navigation Technology – 2nd Edition. The institution of Electrical Engineering. 557 pages.

Toth, C., Grejner-Brzezinska, D., Oh, J.H., Markiel, J.N., 2009. Terrain-based navigation: a tool to improve navigation and feature extraction performance of mobile mapping systems. Boletim de Ciências Geodésicas – Special Issue on Mobile Mapping Technology, vol. 15, no.5, p. 807-823.

Tournaire, O., Soheilian, B., Paparoditis, N., 2006. Towards a sub-decimetric georeferencing of a ground-based mobile mapping systems in urban areas :matching ground-based and aerial-based imagery using road marks. Proceedings of the ISPRS Commission I Symposium, volume Part A, Marne-la-Vallée, France.

Tuttas, S., Stilla, U., 2011. Window Detection In Sparse Point Clouds Using Indoor Points. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. 38(3/W22), pp. 131-136.

Tuttas, S., Stilla, U., 2013. Reconstruction of façades in point clouds from multi-aspect oblique ALS. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol.II-3/W3, pp.91-96.

van Verth, J.M., Bishop, L.M., 2008. Essential Mathematics for Games & Interactive Applications. A programmer's guide. Second Edition, (674 pages).

Vervisch-Picois, A., 2010. Etude de Systèmes de Positionnement en Intérieur Utilisant des Mesures de Phase du Code ou de Phase de la Porteuse de Signaux de Navigation par Satellites. Thèse de Doctorat, Université Pierre et Marie Curie - Paris 6.

Vosselman, G., Gorte, B.G.H., Sithole, G., Rabbani, T., 2004. Recognizing structure in laser scanner point clouds. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XXXVI, pp.33-38.

Vosselman; G., Maas, H.G., 2010. Airborne and terrestrial laser scanning. Whittles Publishing, 2010 - Technology & Engineering, 318 pages.

Wang, J., Dai, L., Tsujii, T., Rizos, Ch., 2001. GPS/INS/Pseudolite Integration: Concepts, Simulation and Testing. Proceedings of the 14th International Technical Meeting of the Satellite Division of the Institute of Navigation. Salt Lake City, UT, 11-14 September, 2708-2715.

Wang, J., Tan, Y., 2012. Hausdorff Distance with k-Nearest Neighbors. Advances in Swarm Intelligence, Third International Conference, ICSI, 17-20 Juin, Shenzhen, Chine, pp.272-281.

Wang, R., Bach, J., Ferrie, F., 2011. Window Detection form Mobile LiDAR Data. IEEE Workshop on Applications of Computer Vision (WACV), 5-7 Janvier, Kona, Hawaii.

Wang, W., Lou, A., Wang, J., 2012. The research of line matching algorithm under the improved homograph matrix constraint condition. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XXXIX-B3, pp. 345-350.

Wang, Y., Ewert, D., Schilberg, D., Jeschke, S., 2013. Edge Extraction by Merging 3D Point Cloud and 2D Image Data. 10th International Conference & Expo on Emerging Technologies for a Smarter World (CEWIT), 21-22 Octobre, Melville, New York.

Wang, Y., Ewert, D., Schilberg, D., Jeschke, S., 2014. A New Approach for 3D Edge Extraction by Fusing Point Clouds and Digital Images. Applied Mechanics and Materials, vol. 457-458, pp.1012-1016.

Wiart, A., 2013. Calibration extrinsèque d'un scanner laser multi-fibre. Test sur le laser Velodyne. Rapport de stage de fin d'étude. MINES ParisTech & ENSG.

Williams, K.E., 2012. Accuracy Assessment of LIDAR Point Cloud Geo-Referencing. These de Master, Oregon State University.

Yang, M.Y., Forstner, W., 2010. Plane Detection in Point Cloud Data. Technical Report Nr.1, Department of Photogrammetry, Institute of Geodesy and Geoinformation, University of Bonn.

Yao, J., Ruggeri, M.R., Taddei, P., Sequeira, V., 2010. Robust range image registration using 3D lines. Proceedings of 2010 IEEE 17th International Conference on Image Processing, September 26-29, Hong Kong, pp.4321-4324.

Yong, L., Huayi, W., 2008. Adaptive building edge detection by combining Lidar data and aerial images. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XXXVII, pp.197-202.

Yoo, H-J., 2010. Analyse et conception de scanners laser mobiles dédiés à la cartographie 3D d'environnements urbains. Thèse de doctorat, MINES ParisTech.

Yousif, H., Li, J., Chapman, M., Shu, Y., 2010. Accuracy enhancement of terrestrial mobile LIDAR data using theory of assimilation. IAPRS International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XXXVIII, part.5, pp. 639-645.

Zampa, F., Conforti, D., 2009. Mapping with Mobile Radar. GIM International, vol. 23, no.4, pp.35-37.

Zhang, Z., Faugeras, O.D., 1991. Determining Motion from 3D Line Segment Matches: a comparative Study. Journal Image and Vision Computing, vol. 9, no.1, pp.10-19.

Zhang, Z., 1994. Iterative Point Matching for Registration of Free – Form Curves and Surfaces. International Journal of Computer Vision, vol. 13, pp.119-152.

Zhu, Z., Liu, J., 2013 Unsupervised Extrinsic Parameters Calibration for Multi-beam LIDARs, Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering, ICCSEE.

Zuliani M., 2012. RANSAC for Dummies, Technical Report. en-ligne: http://vision.ece.ucsb.edu/~zuliani/Research/RANSAC/docs/RANSAC4Dummies.pdf

Annexe A

Distance minimale entre deux segments

Nous présentons dans cette annexe la méthode de [Sanday, 2001] permettant de déterminer la distance la plus courte entre deux segments. Puisqu'elle s'inspire de la définition de distance minimale entre deux lignes, nous commençons par expliquer son calcul.

Soient L_1, et L_2 deux droites représentées respectivement par les points P_0, Q_0 et les vecteurs non nuls $\vec{u}$, $\vec{v}$ (Figure A.1) :

$$L_1 : P(s) = P_0 + s\vec{u} \qquad\qquad L_2 : Q(t) = Q_0 + t\vec{v} \tag{A.1}$$

Les paramètres s_C et t_C indiquent la localisation des deux points plus proches sur les lignes L_1, et L_2. Nous désignons par $\vec{w}_C$ le vecteur défini par les deux points $P(s_C)$ et $Q(t_C)$. Sa norme correspond à la distance minimale entre deux lignes (A.2).

$$\vec{w}_c = P(s_C) - Q(t_C) \tag{A.2}$$

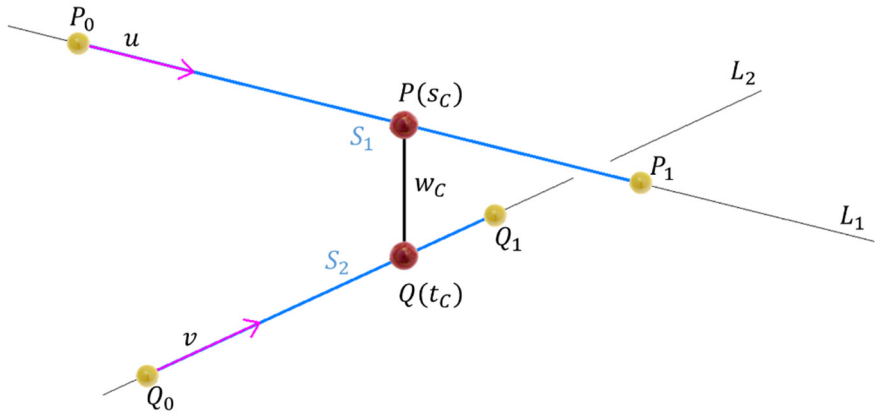

Figure A.1 Définition de la distance minimale entre deux segments de droite

En remplaçant dans la formule (A.2) les points $P(s_C)$ et $Q(t_C)$ par (A.1), nous obtenons :

$$\vec{w}_C = \overrightarrow{w_0} + s_C\vec{u} - t_C\vec{v} \tag{A.3}$$

où $\overrightarrow{w_0} = (P_0 - Q_0)$.

Par définition, le vecteur $\vec{w}_C$ doit être, à la fois, perpendiculaire à L_1 et L_2 tant qu'il représente la distance minimale entre ces deux lignes. Sachant que les deux vecteurs sont orthogonaux si est seulement si, leur produit scalaire est nul, nous notons :

$$\begin{cases} \overrightarrow{w_C} \cdot \vec{u} = 0 \\ \overrightarrow{w_C} \cdot \vec{v} = 0 \end{cases} \tag{A.4}$$

En prenant en compte de l'équation (A.3), la formule (A.4) prend forme :

$$\begin{cases} \overrightarrow{w_0} \cdot \vec{u} + s_C\vec{u} \cdot \vec{u} - t_C\vec{u} \cdot \vec{v} = 0 \\ \vec{w}_0 \cdot \vec{v} + s_C\vec{u} \cdot \vec{v} - t_C\vec{v} \cdot \vec{v} = 0 \end{cases} \tag{A.5}$$

La résolution de ce système d'équations linéaires à deux inconnues nous amène à déterminer les valeurs s_C et t_C de la manière suivante :

$$s_c = \frac{be - cd}{ac - b^2} \tag{A.6}$$

$$t_c = \frac{ae - bd}{ac - b^2} \tag{A.7}$$

pour les a, b, c, d, e s'interprètent comme le produit scalaire de deux vecteurs (A.8).

$$a = \vec{u} \cdot \vec{u} \qquad b = \vec{u} \cdot \vec{v} \qquad c = \vec{v} \cdot \vec{v} \qquad d = \vec{u} \cdot \vec{w}_0 \qquad e = \vec{v} \cdot \vec{w}_0 \tag{A.8}$$

Il faut noter que le dénominateur $ac - b^2$ de deux équations (A.6) et (A.7) doit être positif. S'il est égal à zéro, les deux lignes sont parallèles.

La recherche de distance minimale entre deux lignes se résume alors à employer les paramètres s_C et t_C obtenus pour trouver la norme du vecteur $\vec{w}_C$ défini selon la formule (A.2).

Passant à présent au cas de deux segments S_1 et S_2 représentés par les équations paramétriques :

$$S_1(s) = P_0 + s\vec{u}, s \in [0,1] \tag{A.9}$$

$$S_2(s) = Q_0 + t\vec{v}, t \in [0,1]$$

Nous commençons, de manière similaire, par calculer la distance entre deux segments en minimisant la norme du vecteur $\vec{w}_c$ $(\|\vec{w}_c\|)^2$). Ainsi, la position des points les plus proches sur les deux droites L_1 et L_2 est retrouvée grâce aux équations (A.6) et (A.7).

Ensuite, nous procédons à vérifier si les valeurs de s_C et t_C sont comprises dans l'intervalle de 0 à 1, puisque $s_C \in [0,1]$ et $t_C \in [0,1]$. Si c'est le cas, les points $P(S_C)$ et $Q(t_C)$ appartiennent

aux segments (c'est-à-dire sont situés à l'intérieur du segment délimité par ces extrémités) et nous nous en servons pour déterminer la distance minimale.

Si par contre, ils se trouvent au-delà d'un segment, il faut recalculer les paramètres s_C et t_C de façon qu'ils minimisent la norme de $\vec{w}_C$. Nous testons plusieurs scénarios (A.10) comme nous l'expliquons au fur et à mesure.

(A.10)

Valeur calculée	Valeur admise
$s < 0$	$s = 0$
$s > 1$	$s = 1$
$t < 0$	$t = 0$
$t > 1$	$t = 1$

Alors, si $s < 0$, nous changeons sa valeur qui sera désormais égale à zéro ($s = 0$). Par conséquent, l'équation (A.3) s'écrit comme :

$$\vec{w}_C = \vec{w}_0 - t_C\vec{v} \tag{A.11}$$

et la distance à minimiser :

$$\vec{w}_C \cdot \vec{w}_C = (\vec{w}_0 - t_C\vec{v}) \cdot (\vec{w}_0 - t_C\vec{v}) \tag{A.12}$$

La dérivée partielle selon t_C de l'équation (A.12) amène à la formule suivante :

$$-2\vec{v} \cdot (\vec{w}_0 - t_C\vec{v}) = 0 \tag{A.13}$$

dont la résolution donne la valeur de t_C :

$$t_C = \frac{\vec{v} \cdot \vec{w}_0}{\vec{v} \cdot \vec{v}} \tag{A.14}$$

En admettant que le $s = 0$ pour le segment S_1 (qui correspond nota bene à l'une des extrémités du segment S_1), nous obtenons la valeur de t pour le segment S_2.

Lorsque $s > 1$, on fixe $s = 1$. Puis, nous entreprenons les mêmes démarches. La distance minimisée est :

$$\vec{w}_C \cdot \vec{w}_C = (\vec{w}_0 + \vec{u} - t_C\vec{v}) \cdot (\vec{w}_0 + \vec{u} - t_C\vec{v}) \tag{A.15}$$

De même, nous calculons la dérivée partielle de (A.15) selon t_C :

$$-2\vec{v} \cdot (\vec{w}_0 + \vec{u} - t_C\vec{v}) = 0 \tag{A.16}$$

afin d'obtenir, par résolution du système linéaire, le t_C :

$$t_C = \frac{\vec{v} \cdot \vec{w}_0 + \vec{u} \cdot \vec{v}}{\vec{v} \cdot \vec{v}} \tag{A.17}$$

De cette façon, nous déterminons, sur le segment S_2, le point le plus proche du segment S_1.

Si aucun des cas envisagés ci-dessus n'aboutit pas à parvenir le paramètre t_C dont la valeur se trouve dans l'intervalle de 0 à 1, nous appliquons la même procédure pour t conformément aux variantes affichées par (A.10).

Annexe B

Méthode EIGEN

Dans leur travail [Zhang et Faugeras, 1991] discutent de différentes représentations d'une ligne L dans l'espace 3D. Celle-ci peut être intuitivement définie par deux points quelconques (représentation étant plutôt utilisée lorsque l'on s'intéresse à un segment de ligne) (Figure B.1a), ou bien encore par un point m et un vecteur directeur u (Figure B.1b). Une autre possibilité se réfère à la représentation de Plücker d'une ligne, c'est-à-dire celle par deux vecteurs perpendiculaires (u, d) de telle sorte que u soit le vecteur directeur unitaire et le vecteur d soit perpendiculaire au plan passant par cette ligne et l'origine (Figure B.1c). Le vecteur d est alors le produit vectoriel de deux vecteurs u et m (point arbitraire sur la ligne L). Cette dernière définition a été employée par les auteurs afin de dériver la solution analytique d'une transformation rigide - la combinaison d'une rotation (R) suivie d'une translation (T).

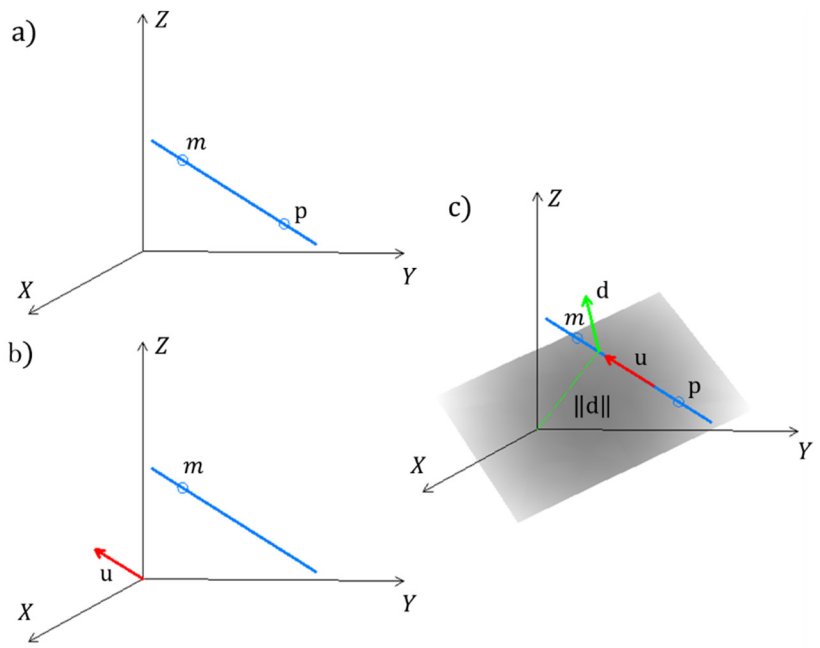

Figure B.1 Différentes représentations d'une ligne 3D

Admettons que (u, d) sont les paramètres de segments à transformer et que (u', d') ceux de segments de référence, la relation entre les entités linéaires s'écrit alors :

$$u' = Ru \tag{B.1}$$

$$d' \triangleq u' \wedge m' = Rd + u' \wedge T \tag{B.2}$$

En conséquence, nous insistons sur le fait que les segments transformés doivent être parallèles aux segments de référence. L'estimation des inconnues de pose aboutit à envisager deux sous-problèmes distincts.

Rotation

Tout d'abord, la matrice de rotation (R) est déterminée par minimisation de la fonction (B.3) qui bénéfice seulement de l'orientation de toutes les paires de segments (u_i, u_i').

$$Err(R) = Min \sum_{i=1}^{N} \|u_i' - Ru_i\|^2 \tag{B.3}$$

Évidemment, l'appariement créé se traduit implicitement par le système d'indiçage ce qui veut dire que: pour tout i, la ligne (u_i, d_i) correspond à la ligne (u_i', d_i'). Tous les calculs sont adaptés à la rotation représentée par un quaternion unitaire $\mathrm{q} = [\lambda_0, \lambda_1, \lambda_2, \lambda_3]^t$.

Étant donné que le produit Ru peut être également exprimé comme la multiplication des quaternions $(*)$ et que $\|q\| = 1$, la formule (B.3) est modifiée en (B.4) :

$$Ru = q * u * \bar{q} \tag{B.4}$$

où $\bar{q}$ symbolise le conjugué du quaternion, par définition, obtenu en conservant sa partie scalaire et en prenant l'opposé de sa partie vectorielle. La fonction à minimiser prend alors la forme suivante :

$$Err(R) = Min \sum_{i=1}^{N} \|u_i' * q - q * u_i\|^2 \tag{B.5}$$

Se référant à la définition du produit de deux quaternions l'expression $u_i' * q - q * u_i$ peut être remplacée par une fonction linéaire $A_i q$ telle que :

$$A_i = \begin{bmatrix} 0 & (u_i - u_i')^t \\ -(u_i - u_i') & (\tilde{u_i} + \tilde{u_i'}) \end{bmatrix}_{4x4} \tag{B.6}$$

où la notation $\tilde{u}$ signifie la matrice antisymétrique créée pour le vecteur $u = [x, y, z]^t$ selon la formule (B.7).

$$\tilde{u} = \begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix} \tag{B.7}$$

Avec toutes ces informations, l'équation (B.5) s'exprime finalement de la manière suivante :

$$Err(R) = Min \sum_{i=1}^{N} q^t A_i^t A_i \, q = Min(q^t A q) \tag{B.8}$$

où la matrice symétrique $A = \sum_{i=1}^{n} A_i^t A_i$ est calculée progressivement pour chaque paire de segments.

Le vecteur propre de la matrice A convenant à la plus petite valeur propre est le quaternion $q = [\lambda_0, \lambda_1, \lambda_2, \lambda_3]^t$ qui représentera la rotation optimale. En revanche, il existe toujours deux quaternions correspondant à la rotation. Ce constat n'est pas surprenant puisque la rotation autour de l'axe u et d'angle θ et la même que celle de l'axe « $-u$ » et d'angle $2\pi - \theta$. Néanmoins, dans la plupart de cas, la rotation ne dépasse presque jamais π. Nous pouvons alors présumer que le premier élément du quaternion (composante réelle) doit être positif. La matrice orthogonale correspondant à la rotation au moyen du quaternion unitaire est donnée par :

$$R = \begin{bmatrix} \lambda_0^2 + \lambda_1^2 - \lambda_2^2 - \lambda_3^2 & 2(\lambda_1\lambda_2 - \lambda_0\lambda_3) & 2(\lambda_1\lambda_3 + \lambda_0\lambda_2) \\ 2(\lambda_1\lambda_2 + \lambda_0\lambda_3) & \lambda_0^2 - \lambda_1^2 + \lambda_2^2 - \lambda_3^2 & 2(\lambda_2\lambda_3 - \lambda_0\lambda_1) \\ 2(\lambda_1\lambda_3 - \lambda_0\lambda_2) & 2(\lambda_2\lambda_3 + \lambda_0\lambda_1) & \lambda_0^2 - \lambda_1^2 - \lambda_2^2 + \lambda_3^2 \end{bmatrix} \tag{B.9}$$

Translation

Une fois la rotation calculée, la détermination de la translation T est assez aisée. Sachant que tous les paramètres de segments sont des constantes, et que seulement les trois composants de vecteur T sont des variables, le calcul de ceux-ci s'effectue par la méthode des moindres carrés, ce qui revient à minimiser :

$$Err(T) = Min \sum_{i=1}^{N} \| d_i' - R d_i - u_i' \wedge T \|^2 \tag{B.10}$$

La dérivée partielle selon T de l'équation (B.10) amène à la formule suivante :

$$\sum_{i=1}^{N} 2\left(d_i' - R d_i - \widetilde{u_i'} T\right)^t u_i' = 0 \tag{B.11}$$

dont la solution est :

$$T = \left(\sum_{i=1}^{N} \widetilde{u_i'} \left(\widetilde{u_i'}\right)^t \right)^{-1} \left(\sum_{i=1}^{N} \left(\widetilde{u_i'}\right)^t \left(d_i' - R d_i\right) \right) \tag{B.12}$$

Annexe C

Méthode ICL(forme ICP)

La méthode *ICL* (forme *ICP*) suit les mêmes étapes que l'approche *EIGEN*. Elle cherche à résoudre deux tâches : le problème non linéaire de rotation et celui linéaire de translation. Mais, nous observons une différence au niveau de représentation des segments et de calcul du vecteur de translation. Chaque droite de l'ensemble A est décrite par sa forme paramétrique (p_i, v_i) où p symbolise un point aléatoire de la droite et v son vecteur directeur unitaire. Pareillement, les segments de référence M sont caractérisés par (p'_i, v'_i).

Rotation

La matrice de rotation est calculée à partir de direction de toutes les paires de lignes (v_i, v'_i). La fonction d'erreur à minimiser est tout à fait similaire à la formule (B.3). La solution par moindres carrés d'un tel système non linéaire d'équations avait été aussi proposée par [Arun *et al.*, 1987]. Il s'agit de mettre en œuvre la décomposition en valeurs singulières (*SVD*) de la matrice de covariance croisée $\sum_{vv'}$ formée pour l'appariement composé de N paires de segments comme suit :

$$\sum\nolimits_{vv'} = \frac{1}{N}\sum_{i=1}^{N}(v_i - \bar{v})(v'_i - \bar{v}')^t \tag{C.1}$$

Sachant que les vecteurs $\bar{v}$ et $\bar{v}'$ sont les centres de masse qui valent respectivement :

$$\bar{v} \triangleq \frac{1}{N}\sum_{i=1}^{N}\vec{v_i} \quad \text{et} \quad \bar{v}' \triangleq \frac{1}{N}\sum_{i=1}^{N}\vec{v'_i} \tag{C.2}$$

nous arrivons à la formule suivante :

$$\sum\nolimits_{vv'} = \frac{1}{N}\sum_{i=1}^{N}\left(\vec{v_i}\left(\vec{v'_i}\right)^t - \bar{v}\,(\bar{v}')^t\right) \tag{C.3}$$

La décomposition de matrice carrée (3x3) $\sum_{vv'}$ s'écrit :

$$\sum\nolimits_{vv'} = USV^t \tag{C.4}$$

où la matrice S contient les valeurs singulières de la matrice $\sum_{vv'}$, et les matrices U et V comprennent chacune un ensemble de vecteurs orthonormés. Ceci nous permet de calculer la matrice de rotation d'une manière simple :

$$R = VU^t \tag{C.5}$$

Translation

Dès que la rotation R est déterminée, le vecteur de translation T peut être calculé par les moindres carrés. À cet effet, deux points quelconques (a_i, b_i) appartenant à *i-ème* droite de A, mais aussi leurs points homologues (a_i', b_i') de M, sont utilisés. Leurs positions sur les droites respectives sont déterminées grâce à l'équation paramétrique :

$$a_i = p_i + t_1 v_i \quad \text{et} \quad b_i = p_i + t_2 v_i \tag{C.6}$$

$$a_i' = p_i' + t_3 v_i' \quad \text{et} \quad b_i' = p_i' + t_4 v_i' \tag{C.7}$$

Puisque (p_i', v_i') caractérisent une ligne de référence M (immobile), la correspondance avec son homologue fournit également la contrainte :

$$\begin{cases} p_i' + t_3 v_i' = R(p_i + t_1 v_i) + T \\ p_i' + t_4 v_i' = R(p_i + t_2 v_i) + T \end{cases} \tag{C.8}$$

Allons plus loin, la formule (C.8) est substituée par :

$$\begin{cases} \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} + t_3 \begin{pmatrix} v_x' \\ v_y' \\ v_z' \end{pmatrix} = R \begin{pmatrix} x \\ y \\ z \end{pmatrix} + t_1 R \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} + \begin{pmatrix} T_x \\ T_y \\ T_z \end{pmatrix} \\ \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} + t_4 \begin{pmatrix} v_x' \\ v_y' \\ v_z' \end{pmatrix} = R \begin{pmatrix} x \\ y \\ z \end{pmatrix} + t_2 R \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} + \begin{pmatrix} T_x \\ T_y \\ T_z \end{pmatrix} \end{cases} \tag{C.9}$$

Cela permet aussi de l'écrire sous la forme matricielle d'un système linéaire $AX = L$ possédant sept inconnues, dont trois composants du vecteur de translation et quatre variables :

$$\underbrace{\begin{pmatrix} v_{xR} & -v_x' & 0 & 0 & 1 & 0 & 0 \\ v_{yR} & -v_y' & 0 & 0 & 0 & 1 & 0 \\ v_{zR} & -v_z' & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & v_{xR} & -v_x' & 1 & 0 & 0 \\ 0 & 0 & v_{yR} & -v_y' & 0 & 1 & 0 \\ 0 & 0 & v_{zR} & -v_z' & 0 & 0 & 1 \end{pmatrix}_{6nx7}}_{A} \underbrace{\begin{pmatrix} t_1 \\ t_3 \\ t_2 \\ t_4 \\ T_x \\ T_y \\ T_z \end{pmatrix}_{7x1}}_{X} = \underbrace{\begin{pmatrix} x' - x_R \\ y' - y_R \\ z' - z_R \\ x' - x_R \\ y' - y_R \\ z' - z_R \end{pmatrix}_{6x1}}_{L} \tag{C.10}$$

où :

$$\begin{pmatrix} x_R \\ y_R \\ z_R \end{pmatrix} = R \begin{pmatrix} x \\ y \\ z \end{pmatrix} \quad \text{et} \quad \begin{pmatrix} v_{xR} \\ v_{yR} \\ v_{zR} \end{pmatrix} = R \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} \tag{C.11}$$

La résolution du système revient alors à chercher les inconnues X satisfaisantes simultanément toutes les équations de la formule (C.10). Ainsi, $X = (A^t A)^{-1} A^t L$.

Annexe D

Méthode FMII

Dans nos travaux, nous nous intéressons tout particulièrement à la variante *FMII* permettant de résoudre le problème du recalage de deux jeux de lignes, dont les entités linéaires de référence (*Model*) sont finies et celles à recaler (*Image*) infinies. Ce choix peut s'expliquer, conformément à [Kamgar-Parsi et Kamgar-Parsi, 2004], par le fait que le *FMFI*, conçu pour les segments de ligne, risque d'échouer si le segment plus court d'une paire n'est pas complètement inclus dans son homologue. Dans un cas général, cette approche n'est pas exploitable puisque les segments provenant de différentes sources ne se chevauchent souvent que partiellement. L'alternative serait plutôt de remplacer des segments *Image* par des lignes et de suivre, par la suite, des démarches envisagées pour la procédure *FMII*, comme nous l'expliquerons au fur et à mesure dans cette section.

Étant donné deux ensembles d'entités linéaires, de telle sorte que $A = \{A_1, \dots, A_i, \dots, A_N\}$ comprend des segments de référence (*Model*) par rapport auxquels des lignes *Image* $X = \{X_1, \dots, X_i, \dots, X_N\}$ seront déplacés, la représentation de chacun des éléments est la suivante :

$$A_i = (a_i, v_i, L_i) \quad \text{et} \quad X_i = (x_i, v_i') \tag{D.1}$$

où :
a_i : milieu du segment *Model*
v_i et v_i' : vecteur directeur unitaire de la ligne *Model* et *Image*
x_i : point de la ligne *Image* le plus proche de l'origine ce qui implique $x_i^t v_i' = 0$
L_i : longueur du segment *Model*

Les lignes *Image* seront désormais considérées comme illimitées.

Distance entre les segments de longueur identique

Comme il a déjà été démontré (Chapitre 5), le calcul de la distance entre deux jeux de segments 3D n'est pas trivial. Pour leur part [Kamgar-Parsi et Kamgar-Parsi, 2004], la définissent comme la somme des carrés des distances obtenues entre les points correspondants appartenant à A et X. Ces derniers sont considérés comme les points ayant la même distance u par rapport à des points de départ admis. La Figure D.1 illustre deux segments de longueur identique L pour lesquels les points correspondants p et q ont été définis à l'égard de leurs milieux a et x.

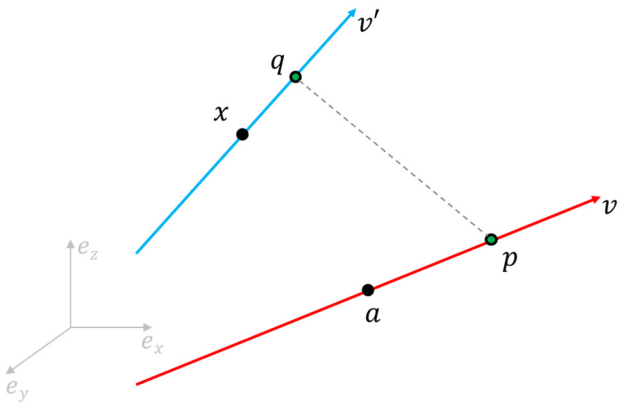

Figure D.1 Choix des points correspondants p et q par rapport aux milieux a et x (Illustration reproduite à partir de [Kamgar-Parsi et Kamgar-Parsi, 1997])

Envisageant ce cas, nous pouvons utiliser la variable u, de sorte que $-L/2 \leq u \leq L/2$, pour paramétrer les segments A et X. Nous notons les coordonnées de points correspondants : $p = a + uv$ et $q = x + uv'$. D'où, le carré de la distance *Euclidienne* entre ces deux points se calcule :

$$D^2(u) = \|(a - x) + u(v - v')\|^2 \tag{D.2}$$

ce qui amène à la formule suivante :

$$M(A_i, X_i) = \int_{-L/2}^{L/2} du D^2(u) = L_i \|a_i - x_i\|^2 + \frac{L_i^3}{6}(1 - v_i^t v_i') \tag{D.3}$$

Nous en ressortons que la distance *M(A,X)* dépend de l'orientation des lignes, mais aussi de l'écartement entre leurs milieux. Par conséquent, sa valeur est nulle, si est seulement si, les deux segments sont parallèles et leurs milieux coïncident.

Étant donné que chaque ensemble est composé de N segments correspondants, la distance mesurée entre eux s'exprime par l'équation (D.4).

$$M(A, X) = \sum_{i=1}^{N} M(A_i, X_i) = \sum_{i=1}^{N} \left[L_i \|a_i - x_i\|^2 + \frac{L_i^3}{6}(1 - v_i^t v_i') \right] \tag{D.4}$$

Admettons que les segments *Model* (A) sont fixes, nous souhaitons, dans un premier temps, tourner les segments X et ensuite les déplacer. C'est pourquoi, la nouvelle position de segment après la transformation T, consistant en rotation R suivie par la translation T, est suivante :

$$\begin{cases} x_i = T + Rx_i \\ v_i' = Rv_i' \end{cases} \tag{D.5}$$

La distance entre les segments transformés peut être remplacée par :

$$M(A, X) = \sum_{i=1}^{N} M(A_i, X_i) = \sum_{i=1}^{N} \left[L_i \|a_i - T - Rx_i\|^2 + \frac{L_i^3}{6}(1 - v_i^t v_i') \right] \tag{D.6}$$

Distance entre les segments de longueur différente

La définition introduite (D.6) a été modifiée par les auteurs afin qu'elle soit susceptible d'être employée pour les lignes – finies et infinies. Nous passons donc à présenter l'approche *FMII* dédiée au cas mixte, où les segments sont définis par la formule (D.1). Sachant que, quel que soit le point $a_i \in A_i$, il existe sur la ligne X_i son homologue $x_i + s_i v_i'$ pour $s_i \in \mathbb{R}$, la distance $D(A_i, X_i)$ entre ces deux points correspondants se calcule, conformément à la formule (D.3), comme :

$$D(A_i, X_i) = \int_{\Omega_i} du \cdot dist(a_i + uv_i \,,\, x_i + s_i v_i' + uv_i') \tag{D.7}$$

où :
u : scalaire caractérisant la droite (équation paramétrique)
Ω_i : chevauchement entre les segments
$dist$: fonction de distance employée (distance *Euclidienne*)

Nous limitons le calcul de la distance $D(A_i, X_i)$ à la partie de chevauchement entre les segments. Cet intervalle étant défini par Ω_i, nous l'attribuons la longueur L_i, puisque seulement les entités linéaires A_i sont finies.

Compte tenu de la transformation T de l'ensemble *Image*, la distance mesurée s'exprime de la manière suivante :

$$M(A, \mathrm{T}X) = \sum_{i=1}^{N} D(A_i, \mathrm{T}X_i) \tag{D.8}$$

afin d'être finalement déterminée, par la norme euclidienne *L2* selon la formule ci-dessous :

$$M(A, \mathrm{T}X) = \sum_{i=1}^{N} [L_i \, \|a_i - T - R(x_i + s_i v_i')\|^2 + L_i^3 (1 - v_i^t R v_i')/6] \tag{D.9}$$

Rotation

L'estimation de la transformation rigide dans la procédure d'alignement de segments consiste à minimiser leur distance réciproque $M(A, \mathrm{T}X)$ sur tous les T et $\{s_i\}$ possibles. La solution proposée s'avère autour du problème d'optimisation non-linéaire, résolue grâce à la technique itérative, d'un système à $(6 + N)$ variables avec six paramètres de la transformation rigide et N variables pour identifier les points correspondants. La complexité de l'algorithme est $O(N)$, alors il est considéré comme rapidement exécutable.

Lorsqu'on minimise la distance en fonction de R, T et l'ensemble $\{s_i\}$, il faut résoudre l'équation $\partial M / \partial s_i = 0$, ce qui nous amène à calculer le paramètre *shift* $s_i \in \mathbb{R}$ selon la formule (D.10).

$$s_i = (a_i - T)^t R v_i' \quad \text{en supposant que} \quad (Rx_i)^t R v_i' = x_i^t v_i' = 0 \tag{D.10}$$

En remplaçant, dans la formule (D.9), la translation T calculée selon l'équation (D.18), nous pouvons obtenir la mesure de distance se référant uniquement à la rotation R :

$$M(A, \mathrm{T}X) = \sum_{i=1}^{N} L_i \left(\|\acute{a}_i\|^2 + \|\acute{x}_i\|^2 + \frac{L_i^2}{6} \right) - 2 \sum_{i=1}^{N} L_i \left(\acute{a}_i^t R \acute{x}_i + \frac{L_i^2}{12} v_i^t R v_i' \right) \tag{D.11}$$

où :

$$\acute{a}_i = a_i - \bar{a} \quad \text{et} \quad \acute{x}_i = x_i + s_i v_i^{'} - \bar{x} \tag{D.12}$$

Le premier composant de la formule (D.11) est constant puisqu'il ne dépend pas de la matrice de rotation R. Afin de minimiser la distance *M(A,TX)*, il est nécessaire de maximaliser la seconde somme. Pour ce faire, la matrice de covariance croisée $\sum_{A,X}$ est créée entre les segments correspondants de la manière suivante :

$$\sum\nolimits_{A,X} = \sum_{i=1}^{N} [L_i(a_i - \bar{a})(x_i + s_i v_i' - \bar{x})^t + L_i^3 v_i (v_i')^t / 12] \tag{D.13}$$

où les centres de masse $\bar{a}$ et $\bar{x}$ sont donnés par la formule (D.14).

$$\bar{a} = \frac{1}{W}\sum_{i=1}^{N} L_i a_i \quad \text{et} \quad \bar{x} = \frac{1}{W}\sum_{i=1}^{N} L_i (x_i + s_i v_i^{'}) \quad \text{pour} \quad W = \sum_{i=1}^{N} L_i \tag{D.14}$$

La matrice de rotation est représentée par un quaternion unitaire ce qui nécessite, dans un premier temps, de former la matrice symétrique C_{4x4} dont les éléments sont définis à partir de $\sum_{A,X}$.

$$\begin{aligned}
&C_{11} = \sum_{11} + \sum_{22} + \sum_{33}\,; && C_{12} = C_{21} = \sum_{23} - \sum_{32}\,;\\
&C_{13} = C_{31} = \sum_{31} - \sum_{13}\,; && C_{14} = C_{41} = \sum_{12} - \sum_{21}\,;\\
&C_{22} = \sum_{11} - \sum_{22} - \sum_{33}\,; && C_{23} = C_{32} = \sum_{12} + \sum_{21}\,;\\
&C_{24} = C_{42} = \sum_{31} + \sum_{13}\,; && C_{33} = \sum_{22} - \sum_{33} - \sum_{11}\,;\\
&C_{34} = C_{43} = \sum_{23} + \sum_{32}\,; && C_{44} = \sum_{33} - \sum_{11} - \sum_{22}\,;
\end{aligned} \tag{D.15}$$

Ensuite, les quatre valeurs propres de la matrice C sont calculées en tant que solution analytique. Le vecteur propre correspondant à la plus grande valeur propre représente la rotation optimale R exprimée en quaternion unité $q = [\lambda_0, \lambda_1, \lambda_2, \lambda_3]^t$. Le passage entre le quaternion et la matrice de rotation traditionnelle est réalisé selon l'équation (B.9).

Nous pouvons aussi envisager une autre solution consistant à employer la *SVD*. Étant donné que la seconde somme de la formule (D.11) doit être maximisée pour minimiser tout terme, nous arrivons à obtenir :

$$\sum_{i=1}^{N} L_i \left(\acute{a}_i^t R \acute{x}_i + \frac{L_i^2}{12} v_i^t R v_i' \right) = trace\left(\Sigma_{A,X} R^t\right) \tag{D.16}$$

Par conséquent, la décomposition en valeurs singulières de $\sum_{A,X}$ amène à calculer la rotation R. Pour plus de détails le lecteur pourra se référer à [Kamgar-Parsi et Kamgar-Parsi, 1997].

Translation

Dès que la rotation est déterminée, le vecteur de translation T peut être retrouvé en minimisant la formule (D.9). Pour y arriver, la dérivée partielle de $M(A, TX)$ par rapport à la variable T est calculée :

$$\frac{\partial M}{\partial T_i} = -2\sum_i L_i\, \vec{e}_i^{\,t}(a_i - T - Rx_i) = 0 \tag{D.17}$$

Les équations ci-dessus étant linéaires, nous procédons alors à la résolution :

$$T = \bar{a} - R\bar{x} \tag{D.18}$$

Vu que les paramètres *shift* $\{s_i\}$ sont inconnus, les calculs des paramètres optimaux de transformation T seront effectués de manière itérative jusqu'à ce que le changement de valeurs $\{s_i\}$ ne dépasse plus un certain seuil. Le schéma récapitulatif à suivre est illustré par la Figure D.2.

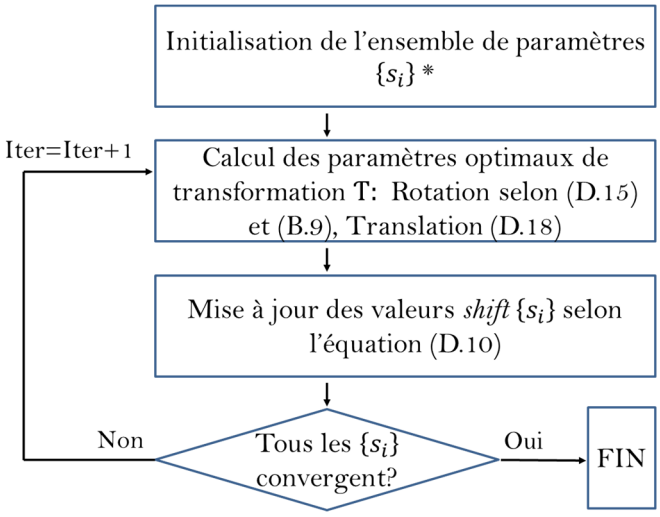

Figure D.2 Schéma général de l'approche
*Valeurs initiales recommandées pour tous les s_i sont des zéros.

Les tests effectués par [Kamgar-Parsi et Kamgar-Parsi, 2004] montrent que le nombre d'itérations *Iter* est fortement lié à la précision désirée et oscille, pour un seuil de convergence égal à 10^{-3} (valeur adimensionnelle), entre trois et vingt.

.

Annexe E

Pseudo-codes

Pseudo-code 4.1: Détection de plans via *RANSAC*

Entrée: N : nuage de points

Paramètres : P : probabilité
σ : écart-type de bruit
N_{pl} : nombre de plans

```
1  Début
2      Initialiser des variables : EC ← 0; N_iter ← ∞; i ← 1; P; σ
3      PourChaque j ← 1: N_pl Faire
4          Seuil :δ² ← σ² F⁻¹_χ²₂(P)
5          TantQue i ≤ N_iter Faire
6              S ← Choisir aléatoirement un échantillon k = 3 de N
7              Estimer le modèle: M(θ)
8              Distance au modèle: d ← e_M(p_i, M(θ))
9              Si d² ≤ δ² Alors
10             Retour EC_i
11             FinSi
12             Mettre à jour le nombre d'itérations: N_iter ← log(1 − P)/log(1 − q)
13             pour q = (Card(EC_i)/Card(N))³
14             Meilleur plan: EC = max(EC_i)
15             Incrémenter: i ← i + 1
16         FinTantQue
17         CC ← Analyser des composantes connexes d'EC (Pseudo-code 4.2)
18         Retour θ; CC
19         N ← N − CC
20     FinPour
21 Fin
```

Pseudo-code 4.2: Détection et étiquetage des composantes connexes

Entrée: N : nuage de points

Paramètres : ρ: rayon de sphère
k: nombre de points voisins
N_{min}: nombre minimal de points d'une composante connexe CC

```
1  Début
2      PourChaque point p_i De N Faire
3          Établir le voisinage V_i de p_i en respectant les seuils imposés (ρ et K)
4          Générer la matrice creuse A de M paires de points voisins
5      FinPour
6      Décomposition de Dulmage-Mendelsohn de A : dmperm(A)=[P,Q,r,s]
7      PourChaque colonne j = 1:|r| − 1 De vecteur d'indices r Faire
8          Étiqueter des composantes connexes CC
9          Si |CC_j| ≥ N_min Alors
10             Calculer la surface S_j de CC_j
11         SinonSi CC_j est trop petite Alors
12             S_j ← 0
13         FinSi
14     Conserver la plus grande composante connexe CC
15     FinPour
16 Fin
```

Pseudo-code 4.3: Extraction d'arêtes de pli

Entrée: $\prod$: paramètres associés au plan
SP : segment plan

Paramètres: T_Q : Seuil poids Q
T_{seg}:longueur minimale d'une arête
T : tolérance distance

```
1  Début
2      P ← Générer toutes les combinaisons possibles à partir de l'ensemble de plans ∏
3      {Phase 1 : Tester l'angle β entre deux plans quelconques}
4      PourChaque paire ∏j et ∏k De P Faire
5          Calculer l'angle β
6          Q = sin²β
7          Si Q ≤ T_Q Alors
8              Rejeter la paire de la liste P
9          FinSi
10     FinPour
11     {Phase2: Tester si deux plans de P sont voisins}
12     PourChaque paire SP_j et SP_k De P Faire
13         d ← Calculer la distance entre deux plans
14         Si d ≤ T Alors
15             I ← Déterminer la droite d'intersection
16             S_j ← projeter SP_j sur I
17             S_k ← projeter SP_k sur I
18             {Phase3: Tester si les segments S_j et S_k se chevauchent}
19             Si ‖M_j M_k‖ ≤ |L_j/2 − L_k/2| Alors
20                 Conserver le segment de droite (SD) de longueur L = min(L_j, L_k) si L ≥ T_seg
21             SinonSi ‖M_j M_k‖ > |L_j/2 + L_k/2| Alors
22                 Conserver SD de longueur L = L_j ∩ L_k si L ≥ T_seg
23             SinonSi
24                 Rejeter SD
25             FinSi
26         FinSi
27     FinPour
28 Fin
```

Pseudo-code 5.1: Qualification à partir d'entités linéaires

Entrée : X_{Test} : coordonnées des segments $Test$ — Paramètres: U : sensibilité

X_{Model} : coordonnées des segments $Model$

Sortie : $OLHD_{TM}; OLHD_{MT}; LHD; Cor$

```
1  Début
2      U ← 0.5
3      Test ← X_Test; Model ← X_Model;
4      PourChaque Segment i ∈ ⟨1; N⟩ De Test Faire
5          PourChaque Segment j ∈ ⟨1; M⟩ De Model Faire
6          Calculer les distances d^II; d^⊥; d^α
7          FinPour
8      FinPour
9      Créer la matrice de similarité SIM de dimension NxM
10     Créer l'appariement préliminaire S contenant N paires
11     Diff_1 ← S_{N+1} − S_N
12     Diff_2 ← Diff_{1_{k+1}} − Diff_{1_k}
13     Si Card(Diff_1 < U) = N + 1 Alors
14         δ ← max(S)
15     SinonSi
16         pic ←trouver le premier maximum local : Diff_2 ≥ U
17         Si pic = Ø Alors
18             δ ← médiane(S) + 2σ
19         SinonSi
20             δ ← S(pic + 1)
21         FinSi
22     FinSi
23     Cor_TM ← (SIM ≤ δ)
24     Calculer OLHD_TM
25     Test ← X_Model; Model ← X_Test;
26     Reprendre les démarches de 4 à 24
27     Retour Cor_MT; OLHD_MT
28     Cor ← Cor_TM ∩ Cor_MT
29     LHD ← max(OLHD_TM, OLHD_MT)
30 Fin
```

Pseudo-code 6.1: Recalage à partir d'entités linéaires

Entrée : X_{Test} : coordonnées des segments $Test$
X_{Model} : coordonnées des segments $Model$
Sortie : $R; T; Cor$

Paramètres: σ : écart-type de bruit
k : échantillon minimal

Début
Initialiser des variables $N_{I_Update} \leftarrow 0; N_{iter} \leftarrow 100; i \leftarrow 1;\ \varepsilon \leftarrow 1e-6;\ k;\ \delta \leftarrow 5.8\sigma$
$A_{ini}, N \leftarrow$ Créer l'appariement préliminaire entre X_{Test} et X_{Model} (Pseudo-code 5.1)
TantQue $i \leq N_{iter}$ **Faire**
$S \leftarrow$ Tirer aléatoirement un échantillon k de A_{ini}
$R_i, T_i \leftarrow$ Calculer les paramètres de transformation rigide par $FMII_2$ à partir de S
Recaler X_{Test} et X_{Model} : $X_{Test_Update} = X_{Test} \cdot R_i + T_i$
$d_i \leftarrow$ Calculer les distances entres les segments appariés A_{ini} (équation 5.10)
$N_I \leftarrow (d_i \leq \delta)$
Si $N_I \geq N_{I_Update}$ **Alors**
$R_{ini} \leftarrow R_i$
$T_{ini} \leftarrow T_i$
$N_{I_Update} = N_I$
FinSi
Mettre à jour le nombre d'itérations : $q \leftarrow (N_I/N)^k$; $N_{iter} \leftarrow log(\varepsilon)/log(1-q)$
Incrémenter : $i \leftarrow i + 1$
FinTantQue
$A_f \leftarrow$ Créer l'appariement fin entre X_{Test_Update} et X_{Model} (Pseudo-code 5.1)
$\hat{R}, \hat{T} \leftarrow$ Calculer les paramètres des transformation rigide par $FMII_2$ à partir de A_f
$R = R_{ini} \cdot \hat{R}$
$T = \hat{R} \cdot T_{ini} + \hat{T}$
$Cor \leftarrow A_f$
Fin

Qualification et amélioration de la précision de systèmes de balayage laser mobiles par extraction d'arêtes

RESUME : Au cours de ces dernières décennies, le développement de Systèmes Mobiles de Cartographie, soutenu par un progrès technologique important, est devenu plus proéminent apparent. Il a été stimulé par le besoin grandissant de collecte d'informations géographiques précises sur l'environnement. Nous considérons, au sein de cette thèse, des solutions pour l'acquisition des nuages de points mobiles de qualité topographique (précision centimétrique). Il s'agit, dans cette tâche, de mettre au point des méthodes de qualification des données, et d'en améliorer les erreurs systématiques par des techniques d'étalonnage et de recalage adéquates.

Nous décrivons une nouvelle démarche d'évaluation de l'exactitude et/ou de la précision des relevés laser mobiles. Celle-ci repose sur l'extraction et la comparaison des entités linéaires de la scène urbaine. La distance moyenne calculée entre les segments appariés, étant la distance modifiée de Hausdorff, sert à noter les nuages par rapport à des références existantes. Pour l'extraction des arêtes, nous proposons une méthode de détection d'intersections entre segments plans retrouvés via un algorithme de RANSAC enrichi d'une analyse de composantes connexes. Nous envisageons également une démarche de correction de relevés laser mobiles à travers un recalage rigide fondé, lui aussi, sur les éléments linéaires. Enfin, nous étudions la pertinence des arêtes pour en déduire les paramètres de l'étalonnage extrinsèque du système mobile. Nous testons nos méthodes sur des données simulées et des données réelles acquises dans le cadre du projet TerraMobilita.

Mots clés : Nuage de points, Système Mobile de Cartographie, Exactitude, Précision, Arête, Segmentation, Évaluation, RANSAC, Étalonnage, Calibrage, Mise en correspondance

Edge-based accuracy assessment and improvement of mobile laser scanning systems

ABSTRACT : Over the past few decades, the development of Mobile Mapping Systems (MMS), supported by significant technological progress, has become more prominent. It has been motivated by the growing need for precise geographic information about the environment. In this thesis, we consider approaches for the acquisition of mobile point clouds with topographic quality (centimeter-level accuracy). The aim is to develop techniques for data quality assessment and improvement. In particular, we eliminate the systematic errors inherent to an MMS data using appropriate calibration and registration methods.

We describe a novel approach to assess the accuracy and/or the precision of mobile laser point clouds. It is based on the extraction and comparison of line features detected within the urban scene. The computed average distance between corresponding pairs of line segments, taking advantage of a modified Hausdorff distance, is used to evaluate the MMS data with respect to a reference data set. For edge extraction, we propose a method which relies on the intersections between planes modelled via the RANSAC algorithm refined by an analysis of connected components. We also consider an approach to correct point clouds using a line-based rigid registration method. Finally, we study the use of edges for estimating the boresight angles of a land-based mobile mapping system. We apply our methods to synthetic data and to real data acquired as part of the TerraMobilita project.

Keywords : Point Cloud, Land-based mobile mapping system (MMS), Accuracy, Precision, Edge, Segmentation, Assessment, RANSAC, Registration, Calibration, Matching

MINES ParisTech

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY