

École Nationale Supérieure d'Architecture de Nancy
Université Henri Poincaré

**Mémoire de soutenance de Stage
du MASTER de recherche
« Modélisation et Simulation des Espaces Bâtis »**

***CONSOLIDATION DES NUAGES DE
POINTS EN LASERGRAMMETRIE
TERRESTRE***

**NOUVELLE APPROCHE BASÉE SUR LES ENTITÉS
LINÉAIRES**

Réalisé par : Majd AL SHAWA

Directeur de stage :
M. Pierre GRUSSENMEYER
Professeur des Universités

Lieu du stage :
INSA de Strasbourg
Equipe MAP-PAGE UMR 694

Octobre 2006

20 mm



Abstract:

The problem of matching 3D LASER clouds is a necessary stage which precedes any kind of architectural modeling in order to perfect the object's geometry and to control its accuracy. It has been studied extensively in many domains, like computer vision, the medical images, matching 3D data with a CAD model and the robotic vision. However there is a lack of an adaptive study of this problem in the domain of the architectural laser. Hence our study aims to summarize the existing methods of laser cloud's registration. We introduce a line-based matching method and the line extraction methods required. We compare also our method with the ICP one, which is mostly applied in the domain of the registration. Our method is intended to handle the special case where both of the point clouds were obtained by mean of as-built topographic scanner, which means that they are georeferenced and needed to be affined by the co-registration.

KEY WORDS: 3D matching, registration, ICP algorithm, lines extraction

Résumé:

Le problème de recalage 3D/3D nuages LASER est une étape indispensable qui précède toute opération de modélisation architecturale afin de parfaire la géométrie de l'objet scanné. Il a été bien étudié dans différents domaines, comme la vision par ordinateur, l'imagerie médicale, la robotique, la mise en correspondance de données 3D avec un modèle CAO, la reconnaissance des formes et celui de l'étude du mouvement d'un objet rigide dans l'espace. Cependant il manque une étude adaptative de ce problème au domaine de la lasergrammétrie architecturale. Notre étude aspire donc à récapituler les méthodes existantes pour le recalage laser. Nous introduisons une méthode de recalage dépendante des entités linéaires, et des méthodes d'extraction des droites nécessaires. Nous comparons aussi notre méthode avec celle d'ICP standard qui est appliquée le plus souvent dans le domaine du recalage. Notre méthode est destinée au cas spécial où les nuages de points sont acquis par un scanner topographique, ce qui veut dire qu'ils sont géoréférencés et qu'ils ont besoin d'être affinés par la consolidation.

MOTS-CLÉS : consolidation, recalage, l'algorithme d'ICP, extraction des lignes

Sommaire

1	Chapitre 1 : problématique, objectif	1
1.1	Introduction.....	1
1.2	La problématique	2
1.3	L'objectif de l'étude.....	5
2	Chapitre 2. Etat de l'art.....	7
2.1	Les méthodes indépendantes des entités géométriques	8
2.1.1	Les techniques de vote :	8
2.1.1.1	Le hachage géométrique	8
2.1.1.2	Le hachage géométrique des courbes	9
2.1.1.3	Le hachage géométrique par la transformée de Hough généralisée	10
2.1.2	Les méthodes de correspondance sous-jacente.....	10
2.1.2.1	La méthode du point itératif le plus proche (ICP)	10
2.1.2.2	La méthode DARCES basée sur RANSAC.....	12
2.1.2.3	La transformation de la distribution normale NDT.	14
2.2	Les méthodes basées sur les entités géométriques.....	14
2.2.1	Les méthodes basées sur les entités surfaciques :	15
2.2.1.1	Le recalage des surfaces en utilisant les images spin	15
2.2.1.2	Le recalage des surfaces par la décomposition	16
2.2.2	Le recalage par la représentation basée sur les arêtes.....	18
2.3	Les méthodes existantes pour extraire les droites.....	19
2.3.1	Algorithme Split-and-Merge (Fusionner et découper)	20
2.3.2	Algorithme de régression de ligne :	21
2.3.3	Algorithme Incrémental :	21
2.3.4	Algorithme de RANSAC :	21
2.3.5	Algorithme de la transformée de Hough :	22
2.3.6	Algorithme d'Expectation – Maximisation (EM) :	23
3	Chapitre 3 : Méthode ICL	24
3.1	L'élimination du bruit :	25
3.2	Où cherchons-nous les droites ?	26
3.3	Allure générale d'une droite dans l'espace :	27
3.4	Tracer la droite	29
3.5	La détection des droites :	30
3.5.1	Méthode de RANSAC	30
3.5.2	Méthode incrémentale.....	31
3.5.3	Transformée de Hough	32
3.5.4	Fusion des segments	34
3.5.5	Un cas particulier	35
3.6	Méthode ICP de base	35
3.7	ICL : méthode ICP adaptée aux droites	38
3.7.1	Appariement des droites	38
3.7.2	Calcul des composants de la transformation rigide	38
3.7.2.1	Forme ICP :	39
3.7.2.2	Forme alternative :	40
4	Chapitre 4 : expérimentations	43

4.1	Lycée des Pontonniers	43
4.2	Remblai ferroviaire	48
4.3	INSA de Strasbourg	53
5	Chapitre 5 Conclusion et perspectives	56
5.1	Conclusion	56
5.2	Perspectives.....	57
6	Annexe	i
6.1	Annexe A : formules de calcul utilisées	i
6.1.1	Droite perpendiculaire à un plan.....	i
6.1.2	Distance entre un point et une droite.	i
6.1.3	Distance entre deux lignes	ii
6.1.4	La projection d'un point sur une ligne.....	ii
6.1.5	Matrice de rotation dans l'espace	ii
6.1.6	Linéarisation des équations de la conjugaison des droites.....	iii
6.1.6.1	Translation (forme ICP adaptée).....	iii
6.1.6.2	Rotation (forme alternative).....	iv
6.1.6.3	Translation (forme alternative) :	vi
6.2	Annexe B : interfaces programmées	viii
7	Bibliographie.....	x
8	Terminologie.....	xii

Liste des figures :

Figure 1.1 Utilisation des sphères pour effectuer le recalage.....	3
Figure 1.2 Utilisation des cibles pour effectuer le recalage (scanner topographique).....	4
Figure 1.3 Effet de l'inclinaison des plans de l'objet sur la résolution du scan.....	5
Figure 2.1 Classification suggérée pour les méthodes existantes de recalage automatique	8
Figure 2.2 Appariement des points les plus proches appartenant à une courbe.....	10
Figure 2.3 Le point le plus proche au plan tangentiel à une surface dans un point connu	11
Figure 2.4 Un échantillonnage favorisé par rapport à l'échantillonnage uniforme (simple)	12
Figure 2.5 La recherche de la sphère virtuelle par deux distances.....	13
Figure 2.6 l'image spin pour un point p dans son voisinage.....	15
Figure 2.7 (a) un repère 3D défini, (b) la grille liée, (c) 8 cellules zoomées, (d) l'intersection avec les facettes triangulaires	16
Figure 2.8 le principe de la subdivision d'Octree.....	17
Figure 2.9 les différents types d'arêtes.....	19
Figure 2.10 les étapes de la méthode de « découper et fusionner ».....	20
Figure 2.11 : la dualité entre l'espace coordonné et l'espace modèle.....	22
Figure 3.1 organigramme général de étapes prévues.	24
Figure 3.2 le changement du nombre de côtés et du volume de l'enveloppe convexe selon l'importance du bruit.	25
Figure 3.3 les points potentiels (candidats) pour extraire les arêtes.....	27
Figure 3.4 les paramètres de l'équation d'une ligne dans l'espace.....	28
Figure 3.5 la représentation graphique d'une droite	
Figure 3.6 \mathbf{a} : un point $P_0(x_0, y_0)$, \mathbf{b} : l'ensemble des droites passant par P_0 , \mathbf{c} : la sinusoïde $S(\theta, \rho)$ rattachée à chaque point P	32
Figure 3.7 l'isolation des maximums régionaux dans l'histogramme de Hough.....	33
Figure 3.8 les résultats de la méthode incrémentale avant et après la fusion.....	35
Figure 3.9 les points appariés avant et après le filtrage.....	36
Figure 3.10 la transformation rigide entre deux segments linéaires.....	41
Figure 4.1 un vue générale de l'entrée principale du lycée.....	43
Figure 4.2 la zone de recouvrement entre les deux scans.....	44
Figure 4.3 La position de deux origines du scan	45
Figure 4.4 les points candidats et les droites extraites.....	45
Figure 4.5 La consolidation et son résultat au sein de Realworks®.....	47
Figure 4.6 les points candidats et les arêtes extraites sur les deux blocs bétons.....	49
Figure 4.7 deux scans effectués (a) pour le même escalier (b) à partir de deux stations différentes.	49
Figure 4.8 l'extraction des plans latéraux délimités des marches.....	50
Figure 4.9 les arêtes des marches extraites des deux scans.....	51
Figure 4.10 l'erreur moyenne quadratique (44 mm) liée à l'affinage de la superposition par la consolidation	52
Figure 4.11 les deux origines des scans issus de la consolidation.....	54
Figure 4.12 les points candidats et les droites extraites du nuage 2.....	54
Figure B.1 la sélection de la pose du nuage (modèle, scène) lors de l'importation.....	viii
Figure B.2 les menus principaux du programme.....	viii
Figure B.3 l'interface interactive de l'extraction des droites.....	ix

REMERCIEMENTS

Je tiens tout d'abord à remercier mon responsable de stage au sein du MAP-PAGE UMR 694, Pierre GRUSSENMEYER, pour le temps qu'il m'a consacré, la patience dont il a fait preuve et les conseils qu'il m'a prodigués. Je lui suis particulièrement reconnaissant pour son aide précieuse et sa disponibilité.

Mes vifs remerciements vont aussi à Samuel GUILLEMIN, technicien de recherche à l'INSA de Strasbourg pour tous les temps productifs que nous avons passés ensemble sur le chantier, pour sa patience, son écoute, ses suggestions et son aide inestimable.

J'adresse mes remerciements à l'équipe MAP-PAGE qui m'a permis d'évoluer dans de bonnes conditions afin d'accomplir cette recherche. Merci à toutes les personnes qui ont participé de près ou de loin à l'achèvement de cette recherche. Merci pour vos notes constructives, merci pour vos critiques.

Je remercie tout particulièrement Delphine QUESTE ma collègue de recherche, qui a toujours cru en ce sujet malgré les difficultés rencontrées et qui m'a toujours soutenu à tous les niveaux.

Merci enfin, à tous mes proches pour leurs encouragements et leur soutien, au gouvernement de la Syrie qui m'a soutenu financièrement et moralement pour poursuivre mes études supérieures en France.

1 Problématique, objectif

1.1 Introduction

‘ L’expression « espaces bâtis » s’applique à un domaine très hétérogène ; on peut en effet le considérer sous une pluralité de registres. Un de ces registres peut se qualifier de ‘naturel’, selon lequel l’espace bâti a été construit, on peut le considérer comme un ‘fait brut’, en quelque sorte dans son état d’objet existant, et la question posée est « comment est-il ? » [SCH06]. Le topographe s’intéresse en général à ce registre. Il prend le côté neutre par rapport à l’édifice. Le point de vue de l’observateur existe cependant ; son objectif est toujours de fournir la représentation la plus proche de la réalité. Il cherche à représenter le modèle 3D « tel que saisi » ; c’est la modélisation la plus fidèle à l’objet mesuré, dont la pertinence relève uniquement de la qualité des mesures et du processus d’élaboration du modèle. Il représente la construction d’un objet tel qu’il existe. Les informations nécessaires à sa construction sont uniquement issues du relevé de l’objet lui-même.

Afin d’approcher ce concept, on remarque que la plupart des techniques de mesures existantes se basent sur le découpage de l’objet et sur l’acquisition des entités géométriques les plus pertinentes. L’ambition d’acquérir l’objet en entier existe toutefois, et c’est pourquoi, la technique du scan LASER terrestre a été développée pour répondre à l’intérêt d’avoir la représentation maximale au niveau du détail.

Pour l’utilisateur, un système laser-scanner 3D est un instrument qui mesure des points 3D (XYZ) sur une zone d’un objet :

- de manière automatique et systématique,
- à une cadence de relevé importante (des centaines ou des milliers de points par seconde),
- qui permet l’obtention des coordonnées 3D en temps réel.

Un scanner peut éventuellement donner en plus et pour chaque point :

- une valeur fonction de la réflectivité de l’objet,
- des valeurs RVB obtenues via une caméra numérique.

Dans des perspectives de réalité virtuelle urbaine, les édifices et sites remarquables, qu’ils soient archéologiques, architecturaux ou autres, seront sans doute les entités de prestige des futures bases de données 3D et seront, par conséquent, modélisés avec une grande finesse tout en étant représentés de manière compacte. La photogrammétrie terrestre, technique, bien qu’adaptée à ce type de levés, est restée marginalement utilisée en raison de la lourdeur de la restitution manuelle et, par conséquent, de son coût prohibitif.

L’automatisation de la restitution photogrammétrique paraissant, à court et moyen terme, difficile (le problème des photos terrestres utilisées dans le domaine de l’architecture étant encore plus difficile qu’en aérien), nous nous donnons les moyens, par l’utilisation et la segmentation conjointe d’images et de nuages de points 3D issus d’un système LASER à balayage terrestre, d’alléger grandement l’intervention manuelle dans la modélisation « tel que construit » de l’édifice. Cependant les deux techniques sont utilisées fréquemment ensemble, car chacune possède des points complémentaires pour l’autre.

Les relevés laser-scanner sont comparables à la photogrammétrie. Chaque point scanné sur l'objet est immédiatement connu en coordonnées X, Y, Z. Du point de vue du géomètre, on note un changement de philosophie par rapport à la topographie classique : on ne relève pas les points caractéristiques de l'objet mais on mesure l'objet dans sa globalité. La précision d'un tel système est illustrée par l'écart-type sur chaque point mesuré. Cependant, la précision du résultat modélisé peut être meilleure que la précision des données brutes et cela grâce à la très haute densité d'information. Cette technologie est intéressante à partir du moment où l'on recherche une haute résolution.

Le relevé laser possède des atouts évidents pour le relevé d'objets architecturaux. En quelques minutes, il est en effet possible de relever des millions de points 3D avec précision (une numérisation dense de l'objet). L'inconvénient majeur de cette méthode reste cependant le temps nécessaire à la modélisation à partir des nuages de points 3D. Le post-traitement des données laser-scanner est une étape indispensable afin de rendre le modèle lisible et complet. L'assemblage de tous les nuages de points serait la base de tout traitement postérieur. Il faut définir un système de coordonnées commun pour les différents scans dans lequel les nuages seront référencés entre eux et bien regroupés afin d'avoir un seul modèle indépendant du scan. Les méthodes de recouvrement nuage sur nuage seront le sujet de cette étude. Ce processus est une étape de base pour tout type de modélisation architecturale suivante, dont on ne peut pas se passer si une modélisation précise et contrôlable est en jeu.

1.2 La problématique

Lors de la construction d'un modèle 3D en utilisant les scanners lasers, on est amené à scanner l'objet à partir de points de vues multiples. Les raisons possibles sont :

- La limitation du champ de visibilité du scanner,
- Les contraintes de recouvrement de l'objet,
- Les dimensions de l'objet,
- Les conditions de visibilité susceptibles de masquer des parties capitales de l'objet,
- L'intérêt de scanner les différentes parties de l'objet en adoptant une résolution qui varie en fonction de sa complexité et de ses détails.
- Il s'y ajoute les conditions de chantier et les temps nécessaires au scannage.

Le recalage 3D/3D encore appelé la mise en correspondance optimale de deux ou de plusieurs scans dont les positions dans l'espace sont aléatoires, est une étape fondamentale qui précède toutes les formes de modélisation. Cette étape est parfois considérée comme une partie de l'acquisition des données (et non du post-traitement). Dans le cas général, les données de départ sont constituées d'un semis de points 3D de coordonnées connues dans l'espace (couramment appelées le nuage modèle), et un autre nuage dont la géométrie interne est stable (désigné scène ou donné) mais sans référence spatiale convenable pour pouvoir bien se superposer au nuage modèle.

Pour simplifier ce problème, soit on néglige les erreurs systématiques liées à l'instrument lui-même, ce qui veut dire que la déformation interne du nuage n'est pas prise en compte, soit on considère que cette déformation est négligée par rapport à l'objectif de l'étude. Cette hypothèse rend « rigide » la transformation issue du recalage laser, ce qui revient à considérer que l'objet ne se déforme pas dans l'espace ; cette transformation est tout simplement définie par une rotation et une translation 3D ou par

une transformation à six paramètres. Trois points (ni colinéaires ni coplanaires) de chaque nuage nous amènent à résoudre le problème. Une solution considérant plus de points serait meilleure, et la solution idéale reviendrait à prendre en compte tous les points de l'objet ayant des informations géométriques pertinentes pour ce type de calcul.

Cette notion nous aide à comprendre la méthode des cibles consistant à « coller » sur l'objet des cibles avant le scannage. La précision de détection des cibles, leur répartition dans l'espace et leur appariement par l'opérateur déterminera la précision du recalage des nuages de points.

Pour certains scanners (SOISIC MENSI®) les cibles étaient remplacées par des sphères. Dans ce cas, le centre des sphères sera le point commun entre les nuages. La détection des centres des sphères se fait automatiquement par les logiciels spécialisés comme Realworks®, Cyclone® ou 3D Ipsos®. Ce système était largement utilisé pour la numérisation et restitution archéologique.

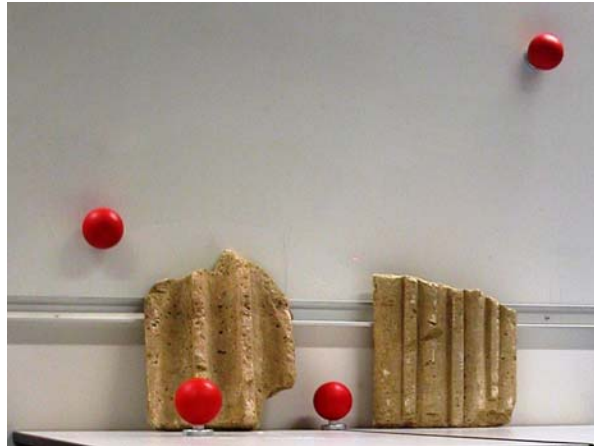


Figure 1.1 Utilisation des sphères pour effectuer le recalage

La nouvelle génération des scanners topographiques (ex. Trimble® GX 3D) permet de stationner et de s'orienter à l'aide de cibles (visée arrière ou avant en langage de géomètre) sur des points connus, ce qui engendre l'acquisition de nuages de points géoréférencés. Les différents scans se superposent automatiquement dans ce mode de saisie. Ce mode d'utilisation correspond aux méthodes utilisées habituellement en topographie, mais l'intervisibilité entre le scanner et les cibles contraint ce mode opératoire. La précision de recalage sera toujours liée à celle des opérations topographiques préalables.



Figure 1.2 Utilisation des cibles pour effectuer le recalage (scanner topographique)

Si l'objet scanné comporte une géométrie représentative, celle-ci peut jouer le rôle de cibles et servir de base au recalage automatique des nuages de points. L'objet de notre étude consistera à identifier et à extraire cette géométrie des nuages de points à recaler entre eux. D'où la formulation de ma problématique tour à tour appelée selon les auteurs : recalage ou consolidation automatique de plusieurs nuages de points, auto-occlusion (en anglais : self-occlusion), consolidation libre de cibles (en anglais : free-markers registration), ou encore recalage entièrement contraint d'un objet 3D (en anglais : Fully Constrained 3D Registration FC3DR).

Le problème de recalage entre deux nuages de points issus de scans laser (modèle-scène) est différent de celui du géoréférencement de deux groupes de points topographiques mesurés dans deux systèmes de coordonnées différents, ou des questions de corrélation de points homologues traités en photogrammétrie, ou encore du suivi du mouvement d'un objet rigide dans l'espace étudié en vision par ordinateur. Les spécificités liées au problème du recalage laser sont :

- La visibilité : il existe toujours des points visibles dans un scan et non dans l'autre en raison du positionnement du scanner lors de l'acquisition des points ; les entités scannées ne sont pas toujours identiques même si des parties communes sont indispensables ;
- Le bruit dans le nuage de points scannés notamment en milieu urbain ;
- La numérotation des points : il n'y a pas aucune correspondance prédéfinie entre les points par l'utilisateur ou par le logiciel ;
- La résolution de scan n'est pas forcément la même pour deux nuages.

La résolution linéaire est impossible à fixer avant le scannage, car pratiquement il n'existe pas de méthode pour la déterminer. Le seul paramètre qui a du sens est la résolution angulaire (le pas angulaire du rayon du scan), mais la géométrie de l'objet sera un facteur déterminant dans la répartition et la densité du nuage de points.

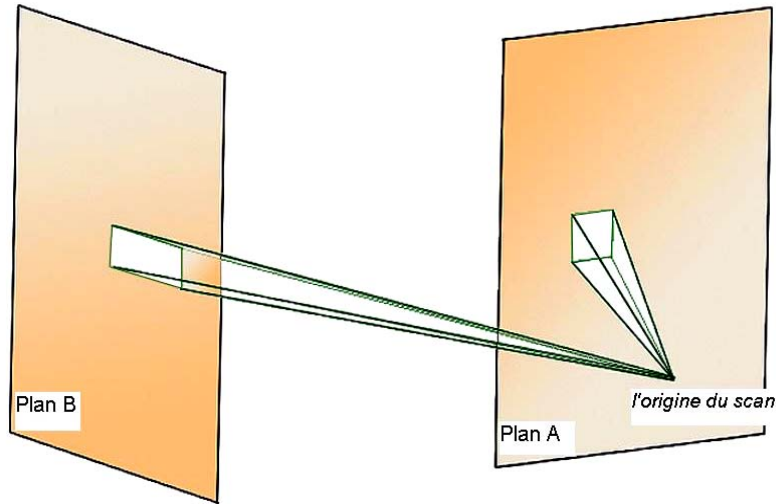


Figure 1.3 Effet de l'inclinaison des plans de l'objet sur la résolution du scan

Le recalage de deux ensembles de points passe par le calcul d'une transformation 3D de l'objet rigide. Il faudra impérativement des processus qui cherchent la correspondance entre les points des deux ensembles à recaler. Le problème se divise donc en deux phases :

La reconnaissance de l'objet scanné

La transformation géométrique

La première phase consiste à établir des correspondances ou des indices caractéristiques entre les points. Ensuite, les paramètres de la transformation seront calculés par des techniques de minimisation comme celle des moindres carrés. On comprendra mieux le problème en l'assimilant à un jeu de puzzle, la première phase est de bien regarder les pièces et d'établir une relation visuelle entre elles, la deuxième phase est de placer les pièces les unes à côté des autres. La relation visuelle sera dans notre cas une relation topologique trouvée par une méthodologie adéquate qui tente de lire la forme de l'objet à partir du scan, tandis que la deuxième phase est une transformation d'objet rigide dans l'espace dénommée souvent problème d'orientation absolue.

Pour pouvoir résoudre le problème, certaines conditions préalables doivent être remplies, comme celles du chevauchement entre les scans. Il faudra agrandir le recouvrement pour avoir de meilleurs résultats, mais le degré de complexité géométrique est un autre facteur à prendre en compte. Ainsi il n'y a aucun intérêt à prévoir une zone commune de 50% sur l'objet si ce dernier se trouve dans le même plan, par contre une superposition de 20 % serait largement suffisante pour une géométrie riche en détails et en entités. Cette condition est liée effectivement à la première phase du problème (celle de la reconnaissance géométrique). Il s'avère qu'en raison des conditions de visibilité, le recouvrement n'atteint jamais 100% même si l'on scanne l'objet deux fois à partir de deux points de vue différents.

1.3 L'objectif de l'étude

Le scanner à notre disposition (GX de Trimble®) permet d'effectuer des travaux topographiques d'exécution courante (routines), tels que les techniques d'intersection basées sur une cible s'appliquant à un point de référence connu ou au relèvement

polygonal/station, par exemple. Les techniques de mise en station peuvent être utilisées pour établir un canevas polygonal du site topographique.

Les relevés multiples sur un objet, effectués depuis des différentes stations, permettent un référencement quasi complet des données topographiques, ce qui réduit leur temps de traitement bureautique, tout en garantissant leur intégrité et leur qualité.

Ce mode d'utilisation aboutit aux coordonnées superposables puisque le scanner permet de les acquérir dans un repère commun cohérent avec les entités géographiques autour (repère monde) et en temps réel. Notre stratégie s'intéresse au cas où les nuages sont superposables, mais nécessitant un ajustement qui sera fait par la consolidation. Ce cas est une instance du problème plus général de la consolidation, cependant spécifique car on dispose dès le départ des coordonnées approchées. Le principe général de la consolidation reste le même, mais la précision des calculs requiert une attention particulière. On peut à ce stade déclarer notre objectif final : comment consolider deux nuages superposés, ce qui revient à étudier le passage « de la superposition à la consolidation ». Nous cherchons à contrôler la superposition (engendrée par les travaux topographiques préalables au scannage) en utilisant la consolidation, ou bien à chercher si on peut améliorer la superposition par la consolidation.

2 Etat de l'art

On ne peut pas nier que la modélisation architecturale à partir de données lasergrammétriques reste une problématique relativement récente. Cependant, dans les domaines particuliers de la vision par ordinateur, de l'imagerie médicale, de la robotique, de la mise en correspondance de données 3D avec un modèle CAO, de la reconnaissance des formes et celui de l'étude du mouvement d'un objet rigide dans l'espace, on trouvera différentes approches du problème de recalage laser 3D. Dans le cas particulier de l'architecture ou de l'archéologie, les utilisateurs se contentent souvent d'appliquer les méthodes proposées dans les logiciels de traitement de nuage de points comme Realworks® ou 3D Ipsos®.

Si on considère la phase de reconnaissance de l'objet scanné, on peut distinguer deux classifications principales : les techniques de vote et les méthodes de correspondance sous-jacente [NNLH05]. Si l'on aborde le problème du point de vue de la précision finale du recalage, on trouvera une autre classification chez [MBO06] : le recalage rugueux ou grossier (Coarse registration) et le recalage précis (fine registration) ; dans la plupart des références, le recalage rugueux est considéré comme une initialisation du recalage précis, on ne va donc pas adopter cette classification, mais on la mentionnera si nécessaire au fur et à mesure.

[TB95] utilise comme terminologie pour les techniques de vote le « recalage avec recherche de mises en correspondance combinatoires », et pour les méthodes de correspondance sous-jacente « le recalage avec appariements explicites itérés ».

Il existe encore une autre classification importante [CHC99] ; celle du recalage basé sur les entités (feature based registration) et du recalage indépendant des entités. Si on approfondit un peu dans toutes les méthodes existantes, on trouve qu'elles s'appuient sur les entités géométriques, des fois implicitement au cours des démarches intermédiaires de son processus (établissement des surfaces locales passant par un voisinage d'un point par exemple) et des fois explicitement avant de démarrer le processus.

On va toutefois réunir les deux classifications ([NNLH05], [CHC99]) ensemble pour mettre en place les grandes lignes de la classification. Il est clair que les techniques de vote et les méthodes de correspondance sous-jacentes (la première classification [NNLH05]) sont des méthodes indépendantes des entités du point de vue de la référence [CHC99]. Un schéma de la classification inspirée des références précédentes est illustré par la figure 2.1. On va l'adapter pour les prochaines explications.

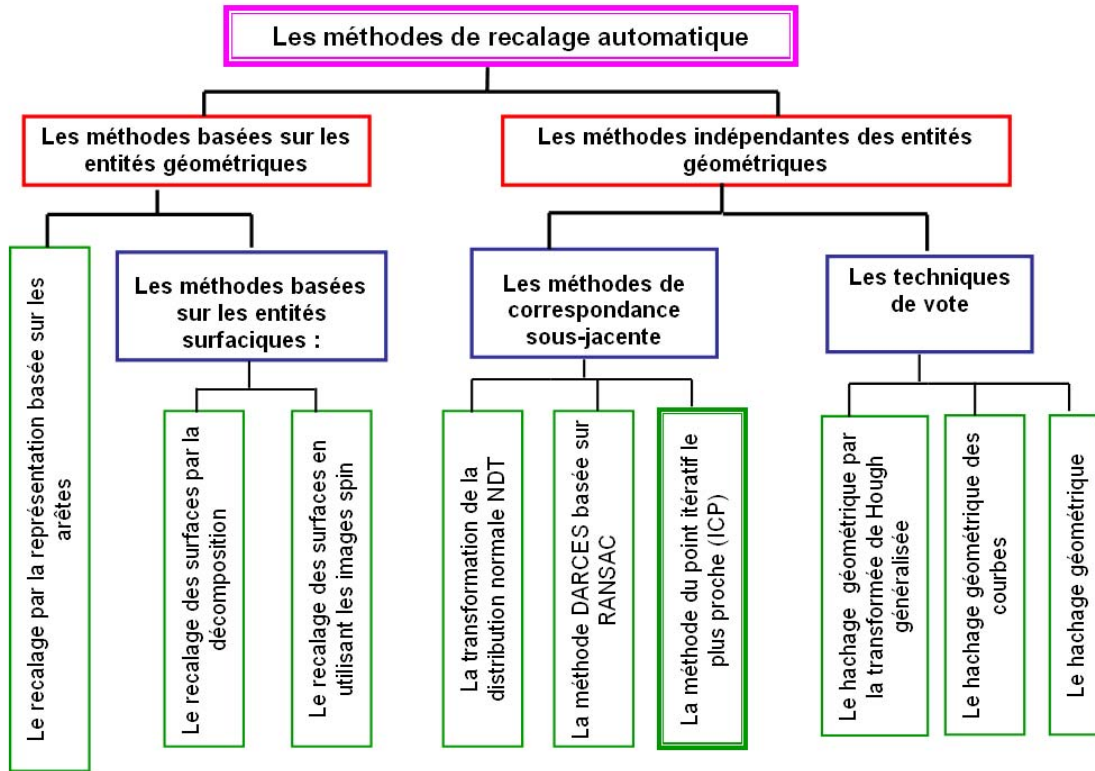


Figure 2.1 Classification suggérée pour les méthodes existantes de recalage automatique

2.1 Les méthodes indépendantes des entités géométriques

2.1.1 Les techniques de vote :

Ces méthodes bénéficient du fait que les objets rigides ne changent pas de forme lorsqu'ils passent par une transformation d'ordre inférieur. Elles cherchent toujours un nombre bas des paramètres de manière itérative. Elles choisissent leurs paramètres finaux parmi un nombre d'options fournies par la méthode elle-même et en dépendant de l'entrée de l'utilisateur.

2.1.1.1 Le hachage géométrique

Une de ces méthodes s'appelle : l'hachage géométrique [WR97] ou l'hachage géométrique de base, qui prolonge la conception de recherche par hachage au monde informatique vers l'espace géométrique. L'hachage par définition est une méthode pour créer une petite « empreinte digitale », pour n'importe quel type de données, il applique des fonctions qui découpent et mélangent les données pour créer ces empreintes digitales qui seront associées aux données.

Pour apporter cette notion sur le recalage de deux semis de points laser (modèle-scène), on définit la « base »; qui est un ensemble de données suffisantes pour achever une transformation quelconque. Par exemple, pour la similitude spatiale, la base serait un triplet : deux lignes non coplanaires, une ligne et un point non colinéaire ou bien trois points. On définit également la table d'hachage, qui possède des enregistrements dans

lesquels on peut mettre une base ainsi que le reste du nuage. Il nous faut finalement une place pour enregistrer les paramètres de la transformation. Ainsi on pourrait organiser le n-uplet de la table. L'algorithme se situe en deux étapes :

- 1) Le pré-processus : Pour chaque nuage modèle, il faut achever ce qui suit :
 - a) Extraire des points à partir des entités (droites, courbes, ou coins) et puis en former des bases.
 - b) Pour chaque base formée à partir des points extraits, on effectue les étapes suivantes :
 - i) Définir un repère local, dépendant de ce triplet.
 - ii) Calculer les paramètres de changement de repère, de celui initial (où se trouvait le reste des points) vers le nouveau repère.
 - iii) Enregistrer le couple (modèle-base) dans la table d'hachage et aussi les paramètres liés au changement de repère. Dans la forme finale de cette table on doit trouver toutes les nouvelles coordonnées associées au couple modèle-base qui les avait produites.
- 2) La reconnaissance automatique (on dispose de la scène et des points extraits en tant qu'entrée):
 - a) choisir un triplet aléatoire.
 - i) Changer le repère vers celui du triplet et recalculer pour le reste des points les paramètres quantifiés de la transformation.
 - ii) Pour chaque coordonnée, on récupère le couple modèle-base qui est enregistré dans la table d'hachage quand les coordonnées sont « proches », autrement dit donner le vote aux modèles-bases puis remettre les votes dans un histogramme.
 - b) Sélectionner toutes les cellules qui ont reçu un nombre de votes dépassant un certain seuil, elles seront alors marquées comme des accords potentiels.
 - c) Retrouver la transformation résultant par moindres carrés pour tous les points en question.
 - d) Transformer les entités selon la transformation retrouvée et les tester, et si le test est négatif, on revient à l'étape 2.a. (itérer la solution).

Il s'avère que la méthode est itérative et la condition d'arrêt est d'avoir un bon modèle de transformation. Elle échoue souvent lorsqu'il existe beaucoup de bruit dans les données, c'est pourquoi, plusieurs algorithmes ont été développés pour traiter le bruit en aval.

Pour avoir une sensibilité haute lors de la recherche dans la table il faut dégrader la tolérance d'accord entre les n-uplets, ce qui est rattrapable quand la solution passe sous les moindres carrés ; mais le temps consommé augmentera, ce qui est le principal inconvénient de la méthode.

2.1.1.2 Le hachage géométrique des courbes

Ce travail a été présenté par [SS87] comme la première initiative de l'hachage géométrique. Il résout le problème d'associer les courbes sous la condition restrictive qu'une des courbes à associer est une sous-courbe d'une autre. L'algorithme s'aide par les moindres carrés pour trouver la meilleure superposition entre les courbes pour pouvoir les joindre dans la table d'hachage. Il suppose que les courbes sont extraites à l'avant et qu'une certaine segmentation a été faite. Cet algorithme est encore appliqué

dans la reconnaissance des objets, la réparation de l'accord entre objet dans le domaine de CAO [Bar97] et l'accord des images médicales [GP97].

2.1.1.3 Le hachage géométrique par la transformée de Hough généralisée

L'idée de base [HB94] est que pour chaque nuage (modèle ou scène), on peut identifier des entités indépendantes du point de vue du scannage (appelées indices), et des autres entités qui en sont dépendantes (appelées références). Les indices nous aident à détecter des objets existant dans le nuage modèle et scène, tandis que les références aident à transformer ces objets.

D'une part les références sont des coordonnées, d'autre part les indices sont définis par rapport à la transformation voulue. Par exemple on adopte comme référence la longueur de ligne dans le cas de transformation rigide (sans changement d'échelle), la direction de tangente lors que la transformation est une translation simple, et les angles entre les côtés d'un triangle formé par trois points si la transformation est une similitude spatiale. L'algorithme doit être donc subtil en ce qui concerne la manière dont on regroupe les entités afin que la représentation (indice- référence) soit valable. On sauvegarde cette représentation dans une table d'hachage, on essaie encore de multiplier les indices pour chaque référence (plusieurs triangles ayant la même base dont la longueur est la référence).

La conversion vers l'espace (indice -référence) qui est dénotée par l'espace des paramètres selon la méthode généralisée de Hough permet d'appliquer le contenu de cette méthode. On accorde deux entités ayant le même indice (une par une), et on accumule celles ayant la même référence (on peut les détecter dans l'histogramme de Hough) ; ceci réduit le nombre des transformations possibles de façon à ce qu'on puisse les vérifier en appliquant la méthode des moindres carrés, mais il faut d'abord revenir de l'espace Hough pour pouvoir extraire les entités issue de la sélection précédente.

2.1.2 Les méthodes de correspondance sous-jacente

2.1.2.1 La méthode du point itératif le plus proche (ICP)

La méthode ICP (Iterative Closest Point) [BM92], est peut-être la méthode appliquée par la majorité des logiciels concernant le recalage automatique entre deux modèles de données géométriques quelque soit leurs natures (points, courbes de degré différent, surfaces) ; elle n'impose pas de conditions préalables et sa convergence est démontré pour la majorité des cas.

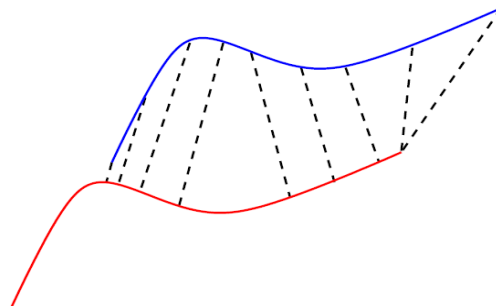


Figure 2.2 Appariement des points les plus proches appartenant à une courbe

Des préliminaires mathématiques sont étudiés [BM92] pour définir le point le plus proche (appartenant à un ensemble géométrique) à un autre point connu. Par exemple la distance entre un point et un ensemble des points est celle minimale entre ce point et chaque point de l'ensemble. Le point de l'ensemble qui rend ce minimum est celui le plus proche, ainsi on peut coupler deux ensemble de points $\{\vec{x}_i, \vec{p}_i\}$. Le nombre de ces points (N_p) est égal au nombre de points du nuage le moins dense qui se situe dans la zone de recouvrement. Supposons que R la matrice de rotation et que T le vecteur de la translation, sont les paramètres de la transformation rigide, la fonction à minimiser est :

$$f(R, T) = \frac{1}{N_p} \sum_{i=1}^{N_p} \|\vec{x}_i - R\vec{p}_i - \vec{T}\|^2. \quad (2.1)$$

Une fois ces paramètres calculés, l'ensemble p_i sera transformé, et on peut réappliquer les étapes précédentes de manière itérative jusqu'à ce que les distances ne dépassent plus un certain seuil. Pour pouvoir résoudre le système surdéterminé (2.1) par les moindres carrés, on peut linéariser les équations et poursuivre la solution traditionnelle des moindres carrés, utiliser la minimisation non linéaire de Levenberg-Marquardt [Fit01], appliquer la technique de SVD proposée par [AHB87] ou celle des quaternions [Hor87].

Les inconvénients principaux de cette méthode dans sa forme originale [BM92] est d'une part qu'elle converge vers un faux minimum spécialement lorsqu'il existe beaucoup de bruit dans les données, et d'autre part le grand nombre d'itérations nécessaires à la convergence. Plusieurs solutions ont été mises en œuvre pour surmonter ces inconvénients, comme celle [CM91] qui remplace la distance entre les points par celle entre un point et entre le plan tangentiel au deuxième plan ; en d'autres termes elle intègre les normales d'une des deux surfaces dans l'équation (2.1)

$$E = \sum \left((\vec{x}_i - R\vec{p}_i - \vec{T}) \cdot \vec{n}_i \right)^2 \quad (2.2)$$

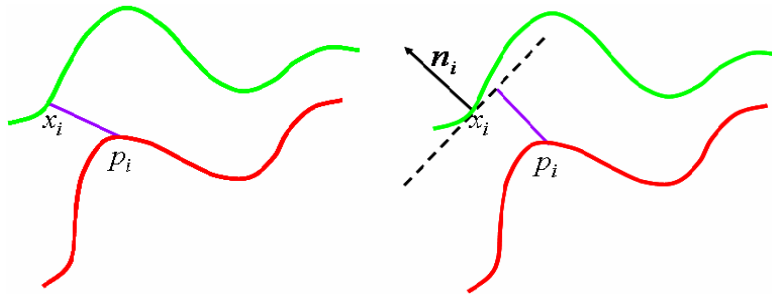


Figure 2.3 Le point le plus proche au plan tangentiel à une surface dans un point connu

[GRL03] a trouvé aussi que le fait de prendre en compte tous les points communs n'améliore pas les résultats, alors qu'un échantillonnage qui accentue la complexité géométrique accélérera la convergence et affinera l'alignement entre les deux semis.

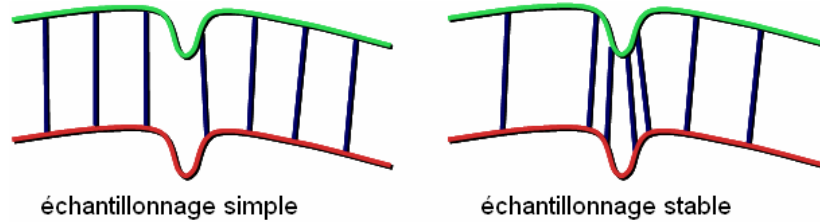


Figure 2.4 Un échantillonnage favorisé par rapport à l'échantillonnage uniforme (simple)

[Zha92] abandonne les points qui dégraderont la transformation en introduisant un test statistique à chaque itération. Ce test prend en compte la moyenne des distances et l'erreur moyenne quadratique pour éliminer les points qui ne fournissent pas de bonnes informations. Il existe encore des algorithmes qui essaient d'attribuer un type quelconque de segmentation aux données issues d'ICP. Par exemple [KSK96] a reconstruit la méthode floue en introduisant à chaque itération une fonction d'adhésion dépendante de la distance aux points modèles, cette fonction intervient dans toutes les étapes de calcul comme facteur de poids ; elle permet de classifier les points soit point objet soit bruit. La méthode fonctionne de façon idéale lorsqu'il s'agit de la reconnaissance d'un objet connu (le modèle ici) dans un nuage de points.

On trouve encore des méthodes qui adoptent le même algorithme de ICP mais en améliorant les fonctions de calculs. Pour [MGPG04] la quantité à minimiser ce n'est non pas la distance euclidienne, mais le champ de distance quadratique pour l'ensemble de points P , qui implique la transformation rigide des points du nuage scène :

$$E = \min_{\alpha} \sum_{q_i \in Q} F(\alpha(q_i), \Phi_p) \quad (2.3)$$

$\alpha(q_i)$ correspond à la transformation rigide du point q_i de l'ensemble scène Q ,

Φ_p correspond la surface approximative passant par le point p_i . La matrice de la rotation est prise dans le cas spécial où les rotations sont petites, d'où la fonction F s'écrit comme un polynôme quadratique F^+ , ce qui facilite beaucoup la linéarisation et améliore la vitesse de l'algorithme. Il a été possible d'appliquer cette restriction car deux méthodes permettant de calculer une position initiale ont été introduites pour le réaliser. La première (calcul en demande) utilise la direction des normales et les directions principales de courbatures pour calculer la pose initiale, tandis que la deuxième utilise d2Tree [LPZ03] qui divise le scan en cellules et garde pour chacune des fonctions de distance quadratique, qui aide à calculer la position initiale.

En dépit de toutes les études qui essaient de trouver automatiquement les valeurs initiales de l'algorithme ICP afin d'accélérer sa convergence, la méthode la plus répandue en milieu d'industrie informatique, reste encore de laisser la main à l'utilisateur pour sélectionner trois points communs entre les deux nuages de manière approximative, ce qui permet de calculer une transformation rigide initiale à partir de laquelle on calcule la position initiale avant de démarrer l'algorithme.

2.1.2.2 La méthode DARCES basée sur RANSAC

Cette méthode [CHC99] est destinée particulièrement aux traitements des données 3D pour lesquelles le chevauchement est partiel. DARCES (data-aligned rigidity-

constrained exhaustive search) considère un échantillon de trois points S_a, S_s, S_p (le nombre minimal pour achever une transformation rigide), elle construit en quelque sorte une sphère passant par ces trois points, puis elle cherche un triplet de points dans le nuage modèle qui appartiennent à la même sphère. Lorsque ce triplet est déterminé, on peut effectuer la transformation rigide et chercher par la suite le nombre des points S_i qui demeurent sur la surface du nuage modèle M , ce nombre est appelé le nombre de chevauchement (overlapping number). Ce sera le seul indice qui permette d'admettre la transformation lorsque celui-ci atteint son maximum au cours de la procédure itérative de l'algorithme.

DARCES impose que les point du triplet ne soient ni trop proches ni trop espacés, elle fonctionne même avec plus de trois points ce qui exige une recherche hiérarchique de plusieurs sphères topologiquement liées. L'expérience de [CHC99] a prouvé que bien que cette recherche soit longue, le nombre de sauts précoces (early jump out) est diminué. Par conséquent le temps parcouru par l'algorithme revient au même que lorsqu'il prend un triplet comme noyau de recherche. La recherche de la sphère passant par les trois points (M_p, M_s, M_q) dans le nuage modèle se fait selon certains niveaux d'organisation : le nuage est indicé selon trois niveaux de complexité (procédure connue dans le monde de l'image et de la vision par ordinateur) pour utiliser la recherche des plus grossiers aux plus fins. Normalement on fixe le point M_p à un niveau et on cherche M_s, M_q dans le niveau le plus fin (Coarse to fine scheme). La construction de la sphère est faite en utilisant deux distances (d_{ps}, d_{qa}) respectivement : Fig. 2.4

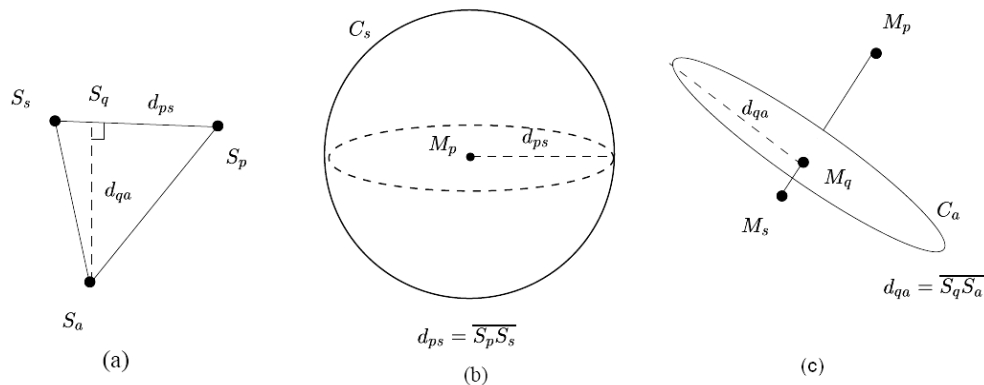


Figure 2.5 La recherche de la sphère virtuelle par deux distances

RANSAC (Random Sample Consensus) qui est un algorithme probabiliste mis par [FB81] pour trouver un modèle convenable pour des données. Parmi ces données il est possible qu'il existe un grand nombre de valeurs incohérentes. RANSAC a prouvé sa fonctionnalité lorsqu'on la combine avec DARCES notamment en ce qui concerne le choix aléatoire du triplet de points, et le nombre d'itérations suffisant pour qu'on juge que les deux nuages soient bien testés.

L'hypothèse principale de la méthode à savoir « qu'il y ait toujours des triplets de points homologues entre les deux nuages » n'est pas toujours valable notamment si l'on scanne en deux résolutions assez différentes ou lorsque les angles de scans ne sont pas de même grandeur. Pour cela on trouve que l'algorithme ne fonctionne qu'avec une large

tolérance ce qui « dégrade » ses résultats; il est même devenu pour cette raison une première approximation pour des autre algorithmes comme ICP.

2.1.2.3 La transformation de la distribution normale NDT.

Le NDT [Bip03], en théorie pur, représente le nuage de points (2D) en grille de cellules régulières, dans chacune d'elles les points sont repartis selon la distribution normale (q, σ) , la probabilité qu'un point se trouve dans une cellule est $p(x)$, la fonction « score » exprime la distribution normale (Gauss). Le comportement de cette fonction change lorsqu'on tourne les sous-ensembles des points ou lorsqu'on les déplace par rapport à un NDT fixe, et on peut observer un pic lorsque le déplacement et la rotation valent zéro. Cette propriété peut être utile pour trouver les meilleurs paramètres de consolidation si l'on applique le NDT sur un nuage et si l'on transforme l'autre de telle manière qu'on minimise la fonction score.

[RB05] propose certaines modifications sur la forme théorique du NDT pour pouvoir l'adapter au scan LIDAR, comme par exemple intégrer la stratégie de « du plus gros au plus fin » en commençant par des cellules grosses et pour les réduire au fur et à mesure, ceci attribue un degré de tolérance lors de l'initialisation des valeurs approchées. Ils proposent aussi de simplifier les données 3D en se limitant sur quelques sections parallèles au plan xy, puis d'optimiser la fonction en utilisant Levnberg-Marquardt au lieu d'utiliser celle de Gauss Newton.

2.2 Les méthodes basées sur les entités géométriques

Ces méthodes supposent une connaissance préalable des entités géométriques de l'objet ou de l'ouvrage scanné. Cette connaissance doit être assez précise car le recalage automatique en dépendra par la suite. En revanche, elles ne doivent pas être exhaustives de tout l'objet puisque leur objectif par rapport à la méthode est de démarrer le recalage et non de modéliser l'objet en 3D. Néanmoins on peut en profiter pour la modélisation. On peut constater que lorsque les entités en question sont des surfaces, les méthodes du recalage automatique seront combinées avec les méthodes concernant la construction des surfaces par le maillage ou par d'autres méthodes. On n'abordera pas ces méthodes car elles appartiennent à un autre sujet, mais on expliquera quelques notions lorsqu'il le faudra.

L'avantage principal de ces méthodes est d'accentuer les propriétés géométriques au lieu de dépendre seulement des points du scan. Au niveau technique ces méthodes n'exigent quasiment pas de valeurs approchées. Les inconvénients de ces méthodes [CHC99] sont leur handicap lors du manque de détails locaux proéminents ou saillants ; il s'y ajoute le temps considérable parcouru pour extraire les entités et pour les sauvegarder. Ces méthodes ne s'inscrivent pas dans une catégorie proprement dite, mais on peut les classer selon la classification des entités géométriques comme les méthodes qui se basent sur les surfaces ou sur les arêtes.

Il faut rappeler que toutes ces méthodes ne sont pas indépendantes de celles précédemment citées, elles les appliquent souvent après avoir extrait les entités, c'est pourquoi certaines personnes considèrent ces méthodes comme un cas spécial ou une personnalisation des méthodes mentionnées auparavant.

2.2.1 Les méthodes basées sur les entités surfaciques :

2.2.1.1 Le recalage des surfaces en utilisant les images spin

[Joh97] a introduit la notion de l'image spin et la normale d'un point. Pour un point p quelconque d'une surface, il faut premièrement trouver le meilleur plan P qui convienne dans le voisinage local de ce point ; ce plan avec sa normale n passant par p forment un repère, selon lequel on représente les points du voisinage x_i par le couple (α, β) Fig. 2.5

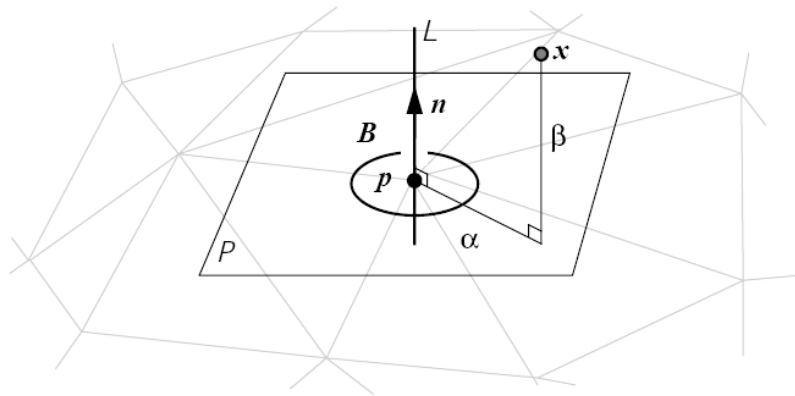


Figure 2.6 l'image spin pour un point p dans son voisinage

(α, β) seront discrétisés et accumulés dans une matrice 2D qui s'appelle l'image spin, chaque cellule (ou pixel) est appelée un « bin ». Comme le nombre de bins est dépendant du nombre de points dans le nuage, il faut dans un premier temps alléger le nuage en gardant sa capacité à représenter les entités géométriques. Pour ce faire on peut utiliser l'algorithme proposé par [JH97] qui diminue le nombre de points en essayant de garder la précision de son maillage 3D.

[Joh97] a démontré que la similarité entre deux images spins reflète la similarité entre les surfaces originales représentées par ces images. Par conséquent le problème du recalage 3D devient un problème de corrélation 2D entre deux images ce qui a été expliqué dans la même source. [CH98] suggère de sélectionner deux échantillons arbitraires (mais assez grands car la précision de l'image spin dépend de la taille du voisinage considéré), puis de créer les images spins pour ces deux échantillons. Lorsqu'une bonne corrélation a été trouvée on peut lancer la transformation entre les deux nuages. A cause de la simplification et la généralisation qui ont été faites, la transformation n'est qu'une approximation qui peut être affinée en appliquant des algorithmes comme ICP. [CH98] conseille cette méthode pour les modèles urbains complexes puisqu'elle permet de gagner un temps considérable et qu'elle amène à des résultats plus stables.

2.2.1.2 Le recalage des surfaces par la décomposition

La décomposition est une procédure commune dans la représentation surfacique des données Laser à cause de leurs volumes qui deviennent parfois impossibles à traiter par les méthodes classiques.

Pour pouvoir recalquer deux semis de points par cette méthode, l'approche de [MBO06] se base sur la décomposition tensorielle : un maillage 2D de Delaunay est construit dans le plan rétiné de la capture, puis il est extrudé dans l'espace 3D après avoir filtré les arêtes dont la longueur dépasse un certain seuil. Ce maillage doit être simplifié en utilisant un algorithme convenable qui préserve le maximum de variations géométriques de l'objet. L'étape suivante consiste à coupler les points avoisinés assez espacés dont les normales ne sont pas parallèles ; pour chaque couple valide, on sauvegarde l'angle mutuel entre les deux normales θ_d , on peut définir ainsi un repère 3D ayant l'origine au point milieu et l'axe Z suit la direction de la moyenne des normales aux deux points. Ce repère sera utilisé par la suite pour définir une grille de même origine.

[MBO06] a défini une grille de $10 \times 10 \times 10$ et de taille liée à la résolution du maillage, un tenseur $10 \times 10 \times 10$ a pour élément l'aire de l'intersection entre la cellule de la grille et le maillage qui se situe dans cette cellule (Fig. 2.6). Les tenseurs de chaque maillage seront indicés dans une table 1D puis seront regroupés selon leur θ_d .

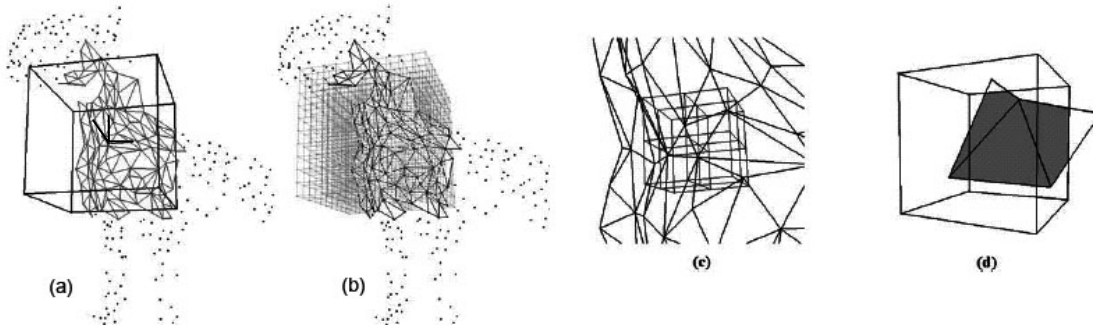


Figure 2.7 (a) un repère 3D défini, (b) la grille liée, (c) 8 cellules zoomées, (d) l'intersection avec les facettes triangulaires

Etant donné que les tenseurs sont des descripteurs des surfaces locales, il serait possible de les accorder en fonction de leur similarité, on choisit seulement les tenseurs ayant une petite différence $\Delta\theta$, et une proportion de chevauchement R_0 qui dépasse un certain seuil.

$$R_0 = \frac{n_q}{n_m + n_s - n_q} \quad (2.4)$$

Où n_q est le nombre des éléments non zéro égaux qui ont le même indice spatial dans les deux tenseurs (modèle, scène T_m, T_s) et n_m, n_s sont les nombres totaux des éléments non zéro dans chaque tenseur. Finalement, pour accorder les tenseurs remplissant les critères précédents, on utilise le facteur de la corrélation linéaire suivante :

$$C_c = \frac{n_q \sum p_i q_i - \sum p_i \sum q_i}{\sqrt{n_q \sum p^2 - (\sum p)^2} \sqrt{n_q \sum q_i^2 - (\sum q_i)^2}} \quad (2.5)$$

Où p_i, q_i sont les éléments des deux tenseurs. Quand on récupère les coordonnées des points concernés par la correspondance précédente, on peut établir un recalage rugueux qui peut être affiné en appliquant des algorithmes comme ICP.

Il sera utile dans ce contexte de rappeler le travail de [MRMT05] qui est une approche pour investir le scan laser dans le domaine où l'on peut observer des phénomènes variant au cours du temps, comme le changement d'un site de chantier ou le déplacement des équipements intérieurs. Dans ce domaine le laser a des avantages car c'est une technique rapide et simple à mettre en place.

Deux scans ont été faits en deux temps différents. Puisque les méthodes de consolidation automatique peuvent aboutir à des résultats erronés lorsque le changement dépasse 10% de l'ensemble d'objet scanné, le recalage a été achevé par les méthodes topographiques ou par la méthode des cibles. Cependant on suggère l'algorithme basé sur la division d'Octree comme un algorithme de vitrification, ce qui n'est pas son objectif dans le document de source [MRMT05].

La division d'Octree est une structure dichotomique qui sépare l'espace 3D en plusieurs cubes de manière récursive, elle part du cube enveloppant le nuage, le divise en 8 cubes, indice chacun par une série de nombres indiquant sa position et son niveau de sous-division et itère la procédure jusqu'à ce qu'on arrive à un cube vide ou au niveau de décomposition voulu (Fig.2.7), ce niveau est déterminé en fonction de la capacité des matériels informatiques.

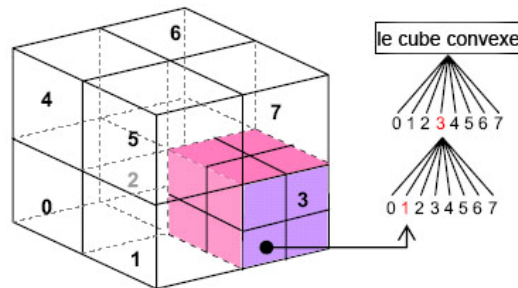


Figure 2.8 le principe de la subdivision d'Octree

Si l'on applique le même degré d'Octree sur tous les scans du même objet, on obtiendra des cellules (S, S' par exemple) comparables ou bien `homologues`, on parle toujours des cellules comparables à un certain niveau l de subdivision. Il existe trois stratégies pour comparer les points compris dans deux cellules homologues :

1. la moyenne des distances : pour chaque point de la cellule S au point le plus proche trouvé dans S' , les meilleurs résultats sont obtenus lorsqu'on applique ce calcul à chaque niveau au moment d'itérer la structure d'Octree.
2. la différence d'orientation des plans moyens : on utilise les moindres carrés pour trouver le plan moyen pour chaque cellule puis on compare l'orientation entre eux. Les résultats ne sont pas rigoureusement précis mais ils s'améliorent en itérant la structure.

3. distance d'Hausdorff : on doit trouver dans S' le point p le plus proche de l'ensemble de S , puis calculer les distances individuelles de chaque point de S à p . Cette méthode est plus précise mais elle consomme plus de temps. Si une modélisation convenable de la surface autour de p est faite, on peut réduire la sensibilité au changement de localisation des points dans les deux scans.

Pour [MRMT05] le challenge était d'isoler les cellules dans lesquelles le changement a eu lieu, notamment lorsque ce changement fait apparaître des parties qui étaient cachées, la solution a été obtenue en s'appuyant sur les images de profondeur et les notions liées au rayon du scan.

On peut juger un scan bon si la distance d'Hausdorff n'excède pas un certain seuil lié à la résolution du scan et à la nature de l'objet scanné.

2.2.2 Le recalage par la représentation basée sur les arêtes

Bien que son nom montre les démarches de cette méthode, il n'est pas assez précis, car les arêtes ici, signifient la meilleure modélisation pour les points de discontinuité. Pour [SRD01] cette modélisation se fait en utilisant des courbes quadratiques, ils la conseillent pour les scans effectués en milieu industriel.

Selon [SRD01] les points de discontinuité conduisent à détecter les arêtes qui sont, soit des arêtes de pli (crease edges) entre deux surfaces d'orientation différente, soit des arêtes de saut (jump edges) lorsqu'une rupture a lieu dans la surface elle-même. On trouve une autre classification des arêtes chez [MG06], qui considère que la variété des arêtes peut être classifiée comme ceci :

- Les arêtes de pas (step edges) : elles apparaissent à cause de la limitation du scanner, car les scanners optiques peuvent détecter seulement ce qui est visible à partir de la position du scanner lui-même. Les trous et les fenêtres sont représentés par quelques points aux bords puis les arêtes de pas sont détectées.
- Les arêtes de pli (fold edges) : elles sont caractérisées par la discontinuité de la première dérivative tout au long de l'arête, elles seront générées lorsque la modélisation des surfaces séparées est faite (modélisation de CAO).
- Les arêtes de lissage (smooth edges) : elles se trouvent dans la même surface, mais à l'endroit où l'on a un changement important de niveau de courbure (ex. la directrice des surfaces coniques).

Il s'avère que les deux termes : arêtes de saut et arêtes de pas signifient la même chose, nous allons les utiliser comme des synonymes dans les étapes suivantes.

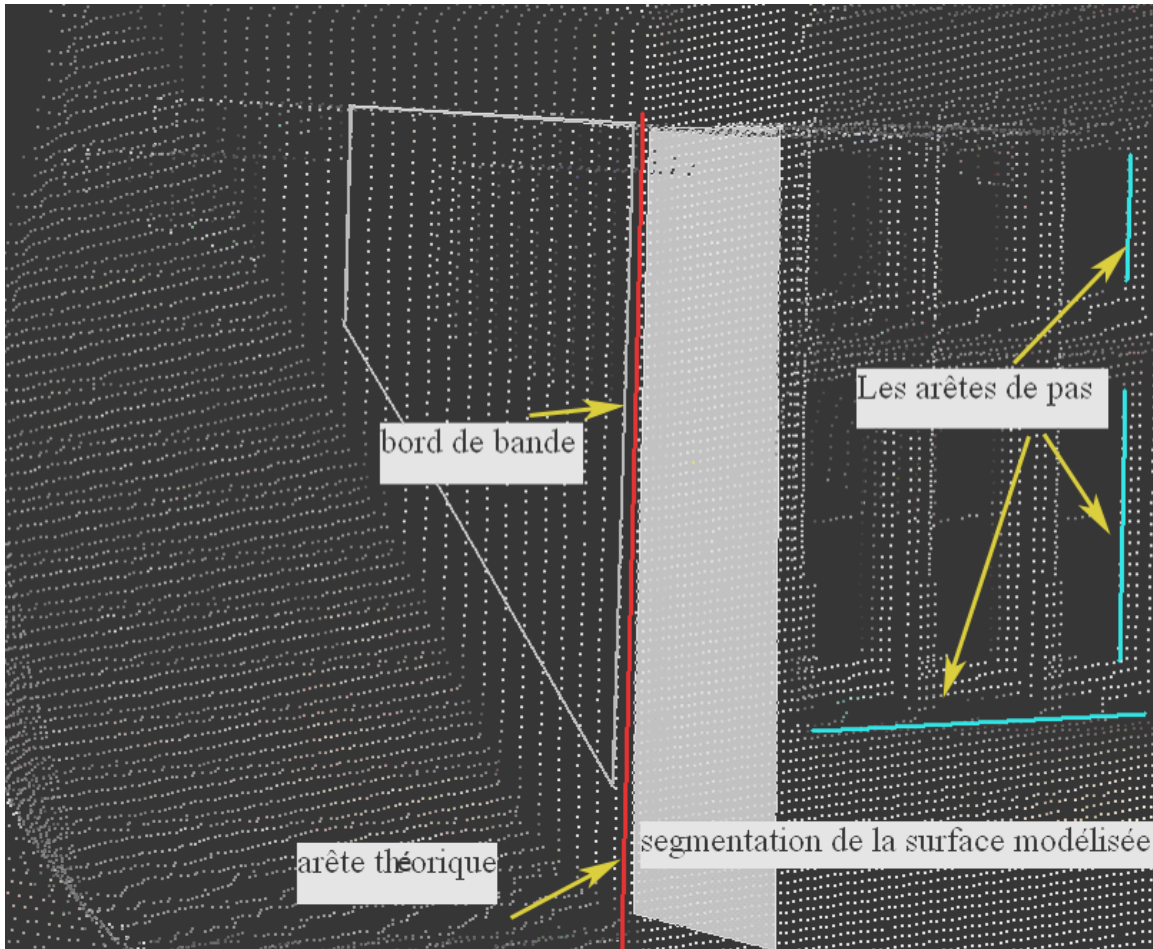


Figure 2.9 les différents types d'arêtes.

La modélisation des plis n'amène pas souvent aux résultats attendus à cause de la résolution du scan et des bandes de balayage, on peut donc représenter l'arête par quatre entités : deux bords de la bande du scan, l'arête théorique et l'arête calculée.

Une méthode efficace pour pouvoir extraire les arêtes quelque soit leur type, est celle de ligne du scan proposée par [JB99]. Elle part du principe admettant que quelque soit la surface $f(x, y, z) = 0$, elle se transforme en un profil $f(x, y_0, z) = 0$ dans le plan xz , si l'on fixe la valeur de $y_0 = y$. Ce profil s'appelle la ligne de scan. [JB99] approximent les surfaces à plusieurs courbes quadratiques, et en déduit les maximums et les minimums régionaux qui peuvent être généralisés en arêtes si l'on déplace la ligne de scan de telle manière à ce qu'elle balaye toutes les surfaces existantes.

[SRD01] ont utilisé les points des arêtes pour appliquer la méthode ICP, ils démontré que cette démarche réduit le temps CPU total par rapport à l'utilisation de l'ICP tout seul ; elle diminue aussi la sensibilité de ICP à la position initiale des nuages.

2.3 Les méthodes existantes pour extraire les droites

Comme notre travail suivant serait orienté vers le recalage qui se base sur les entités linéaires, on va mentionner les algorithmes existants qui concernent l'extraction

de droites à partir du nuage de points 3D ou les cas équivalents. On peut trouver une telle approche dans [NMTS05].

Admettons que les erreurs systématiques ont été éliminées, et le bruit du scan déjà traité. Il existe trois problématiques incontournables auxquelles il faudrait répondre :

- Combien de lignes existe-il ?
- Quel point appartient à quelle ligne ?
- Comment estimer les paramètres de ligne –modèle- au cas où la réponse du terme précédent a été trouvée.

Pour le troisième problème particulièrement, il n'existe pratiquement que la méthode des moindres carrés, en revanche les algorithmes se diversifient en résolvant les deux premiers problèmes.

Dans le contexte ci-dessous, six algorithmes ont été rapidement décrits, pour faire sortir les entités linéaires ou autrement dit les droites.

2.3.1 L'Algorithme de fusionner et découper (Split-and-Merge)

Cet algorithme est probablement le plus populaire dans le domaine d'extraction de droites, il a été obtenu de la vision par ordinateur, puis utilisé dans le domaine de traitement d'un nuage de points laser.

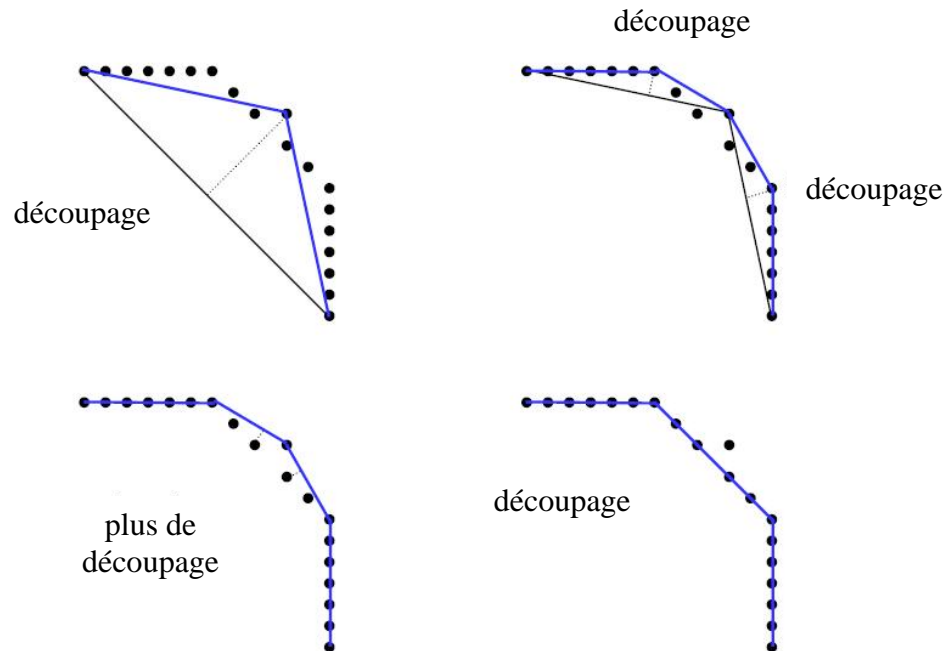


Figure 2.10 les étapes de la méthode de « découper et fusionner »

Algorithme 1 : fusionner et découper

1. Initialiser un groupe s_i comprenant N points, le mettre dans une liste L .
2. passer une ligne convenant le groupe suivant s_i de L .
3. Détecter le point P ayant la distance dp à la ligne précédente.

4. Si dp est inférieur au seuil, continuer (aller à 2).
5. Sinon, fendre s_i à P en s_{i1} et s_{i2} , Remplacer s_i par s_{i1} et s_{i2} dans L , puis continuer (aller à 2)
6. Lorsque l'ensemble du sous-groupe de L est testé, fusionner les segments colinéaires.

2.3.2 L'Algorithme de régression de ligne :

L'idée de base est inspirée de la transformée de Hough, adaptée de manière à ce qu'il transforme premièrement le problème d'extrait en problème de recherche dans l'espace modèle (domaine de paramètres de ligne), puis il utilise l'algorithme agglomératif hiérarchique (AHC) pour faire sortir les segments linéaires adjacents. Un inconvénient majeur de cet algorithme est la complexité au niveau de l'application.

Algorithme2 : régression de ligne :

1. Initialiser la taille d'une fenêtre glissante Nf .
2. modéliser une ligne qui convienne le mieux aux points de la fenêtre précédente.
3. Calculer la matrice de 'fidélité' de la ligne. Chaque élément est la somme de la distance pondérée (distance de Mahalanobis) entre trois fenêtres adjacentes.
4. Construire les segments de ligne en balayant la matrice précédente pour chercher les éléments consécutifs ayant une valeur dépassant un certain seuil (l'algorithme d'AHC).
5. Fusionner les segments superposés et recalculer les paramètres de la ligne pour chacun.

La taille de fenêtre glissante Nf est très dépendante de l'environnement du scan, elle a également un impact important sur la performance de l'algorithme.

2.3.3 L'Algorithme Incrémental :

La simplicité est l'avantage principal de cette méthode, elle est utilisée dans plusieurs applications, elle est aussi désignée par plusieurs noms comme « tracer les lignes ».

Algorithme 3: incrémental:

1. Commencer par les deux premiers points et construire une ligne.
2. Rajouter le point suivant au modèle de ligne.
3. Recalculer les paramètres de la ligne.
4. Si les conditions du modèle sont satisfaites (résidus, erreur moyenne quadratique), on continue (aller à 2)
5. Sinon, rendre le dernier point, et recalculer les paramètres de la ligne, sauvegarder la ligne.
6. Continuer avec les deux points suivants (aller à 2)

Il est possible d'accélérer l'algorithme en ajoutant N point à la fois, et réétudier cet échantillon seulement en cas d'échec (5).

2.3.4 L'Algorithme de RANSAC :

RANSAC (Random Sample Consensus) est un algorithme pour ajuster un modèle lors de la présence de données incohérentes. Il est utilisable pour plusieurs types des

données, à la condition qu'un modèle existe pour représenter les entités. Cet algorithme est largement utilisé dans le domaine de la vision par ordinateur.

Algorithme 4: RANSAC

1. Initialiser un groupe comprenant N point.
 2. Répéter :
 3. Choisir un échantillon de deux points en manière totalement aléatoire.
 4. Passer une ligne par les deux points.
 5. Calculer la distance entre les autres points et cette ligne.
 6. Construire un sous-ensemble des points proches qu'on va appeler les enclaves (inliers).
 7. S'il existe suffisamment de points «proches» de la ligne, enlever ces points du groupe initial.
 8. jusqu'à ce que le nombre d'itérations maximal soit atteint
-

2.3.5 L'Algorithme de la transformée de Hough :

Cet algorithme est utilisé initialement pour trouver les arêtes dans une image d'intensité, cependant il a été ramené sur le scan laser. La recherche s'appuie sur la transformation de l'espace coordonné (x,y) où la droite est exprimée de forme $y = ax + b$, à l'espace modèle ou l'espace paramètre (a,b), où on peut écrire l'équation de la droite sous la forme général $r = x_1 \sin \theta + y_1 \cos \theta$: (r, θ) seront les paramètres (au lieu de a,b).

Un point (x_1, y_1) de l'espace coordonné, est associé à une sinusoïde $r = x_1 \sin \theta + y_1 \cos \theta$ dans l'espace des paramètres. En revanche la droite sera représentée dans cet espace par un point (r_1, θ_1) . L'intersection de deux sinusoïdes est suffisant pour trouver les paramètres de la ligne voulue. Prenons l'exemple suivant:

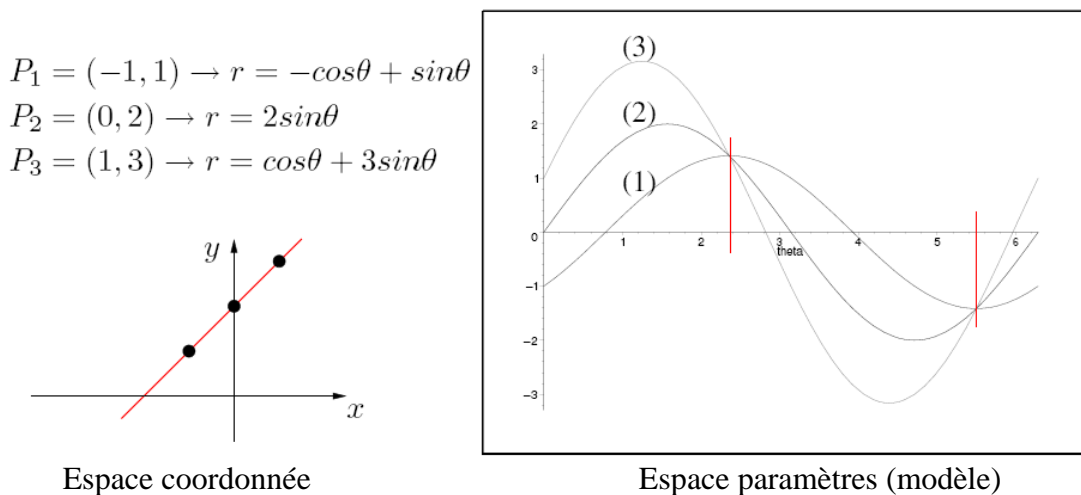


Figure 2.11 : la dualité entre l'espace coordonné et l'espace modèle

Nous pouvons trouver analytiquement les deux intersections : $(\theta = \frac{3\pi}{4}, r = \sqrt{2})$ et $(\theta = \frac{7\pi}{4}, r = -\sqrt{2})$ qui expriment la même droite.

Puisque il s'agit d'observations, on anticipe sur le faite que les intersections auront lieu dans des surfaces et pas à des points précis. La recherche de ces surfaces peut aisément s'effectuer si l'on dessine toutes les sinusoides sur une image.

Voici les étapes pour opérer l'algorithme complet :

Algorithme 5 : transformation de Hough :

1. Initialiser un groupe de N points.
 2. Initialiser la matrice accumulatrice dans l'espace modèle.
 3. Construire les valeurs de cette matrice.
 4. Sélectionner les éléments ayant des maximums V_{max} dans la matrice précédente (seuillage, maximums locaux).
 5. Si V_{max} est inférieur au seuil, terminer.
 6. Sinon, déterminer les enclaves (inliers).
 7. Ajuster une ligne qui s'apparente le mieux aux enclaves et sauvegarder cette ligne.
 8. Enlever les enclaves du groupe initial, puis aller à 2.
-

2.3.6 L'algorithme d'Expectation – Maximisation (EM) :

Cet algorithme se base sur les principes de la probabilité, il est communément utilisé pour résoudre les problèmes lors du manque de variables. Il existe quelques inconvénients pour cette méthode ; on peut citer entre autres :

- La possibilité d'être piégé par le minimum local.
- La difficulté de trouver de bonnes valeurs initiales.

Algorithme 6 : EM

1. Initialiser un group A de N point.
 2. Répéter :
 3. Générer aléatoirement des paramètres d'une ligne.
 4. Initialiser les valeurs des poids pour le reste des points.
 5. Répéter :
 6. Etape E : Calculer les poids des points en dépendant du modèle de ligne.
 7. Etape M : Recalculer les paramètres du modèle.
 8. jusqu'à ce que le nombre maximum d'itération soit atteint ou jusqu'à ce qu'il y ait convergence.
 9. jusqu'à ce que le nombre maximum d'essai soit atteint ou lorsque l'on trouve une ligne.
 10. Si une ligne, enlever les enclaves, aller à 2
 11. Sinon, terminer.
-

3 Méthode ICL

Nous allons adapter une méthode pour le recalage 3D dépendante des entités linéaires présentées par l'objet architectural. Nous pouvons justifier ce choix par les points suivants :

- L'existence quasi permanente de ces entités dans ou autour de l'objet scanné
- Le haut degré de contrainte géométrique fourni par les lignes droites (deux lignes non coplanaires sont suffisantes pour achever le recalage)
- La visibilité de la même droite de plusieurs points de vue (cas des arêtes et non des directrices de cylindres).

Cependant, notre méthode connaît ses limites pour les objets architecturaux non-standards où les entités linéaires s'absentent totalement soit sur l'objet scanné soit autour de lui.

Nous désignerons la méthode proposée par : ICL (pour Iterative Closest Line) en l'assimilant à celle de ICP (Iterative Closest Point). Les deux méthodes considèrent les mêmes grandes lignes en ce qui concerne la méthodologie et la succession des étapes.

Le choix de l'algorithme ICP comme une plateforme à développer s'interprète par le fait qu'ICP demeure la méthode la plus précise parmi les méthodes déjà mentionnées. Elle est par ailleurs une méthode générale qui n'impose aucune condition préalable sur les données de départ. Sa solution converge obligatoirement vers un minimum local et en manière monotone. Le couplage entre cette méthode et les méthodes dépendantes des entités linéaires aboutirait à des résultats satisfaisants et faciles à comparer vue que les démarches d'ICP qui sont 'applicables' sur les lignes. Nous rappelons que la méthode ICP de base [BM92] manipule les lignes à condition de les convertir dans le nuage « données » en sous-semis de points. Nous n'utiliserons pas cette approche, mais nous nous intéresserons, par contre, aux propriétés géométriques des lignes.

La figure 3.1 donne une vue globale des démarches que nous proposons. Nous essayons d'extraire les arêtes linéaires, puis de les utiliser au cours d'algorithme ICP pour pouvoir recalculer deux nuages de points.

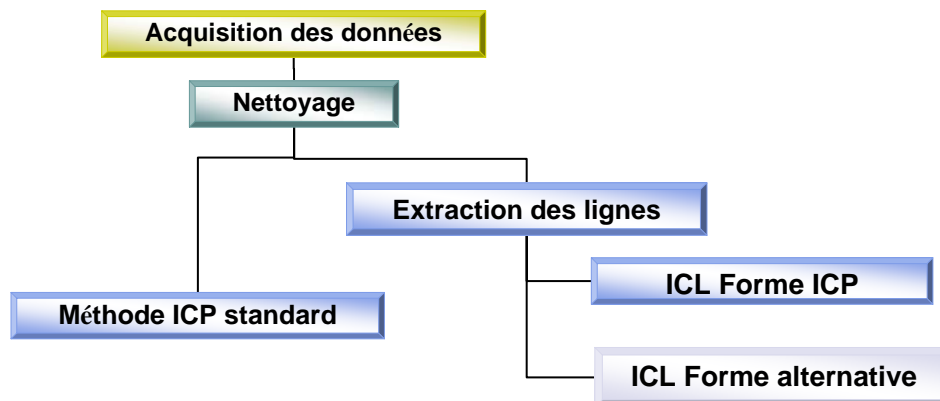


Figure 3.1 organigramme général des traitements envisagés

3.1 L'élimination du bruit :

Un bon nettoyage du nuage de points peut accélérer le temps et l'efficacité de la recherche des lignes. Bien que ce nettoyage puisse être effectué manuellement dans les logiciels du traitement du scan, nous avons suggéré un principe de nettoyage permettant de traiter les points éloignés de l'objet vers l'extérieur et non vers l'intérieur. Pour mieux illustrer la procédure, nous présentons le cas en 2D (le cas en 3D est identique).

Dans un plan, l'enveloppe convexe peut être comparée à un élastique relâché qui englobe tous les points puis qui serait contracté. Nous pouvons étendre cette idée à des points dans l'espace avec un ballon qui se dégonflerait jusqu'à être en contact avec tous les points qui sont à la surface de l'enveloppe convexe. Avec des ensembles dans des dimensions supérieures, l'enveloppe convexe d'un ensemble de points est un polyèdre convexe dont la construction n'est pas aisée et nécessite des algorithmes particuliers.

Dans la figure rattachée, nous pouvons noter que le nombre de côtés de l'enveloppe convexe augmente lorsqu'on approche de l'objet. Elle atteint un nombre de côtés maximaux quand les points de bruit seront éliminés. Nous pouvons utiliser cette constatation pour enlever une partie importante du bruit. La procédure itérative supprime à chaque fois les points les plus proches de l'enveloppe convexe. Nous comptons ensuite les côtés de l'enveloppe. Ce nombre doit diminuer au fur et à mesure. Quand nous ne constatons plus un changement important dans le nombre des côtés, la procédure doit être arrêtée.

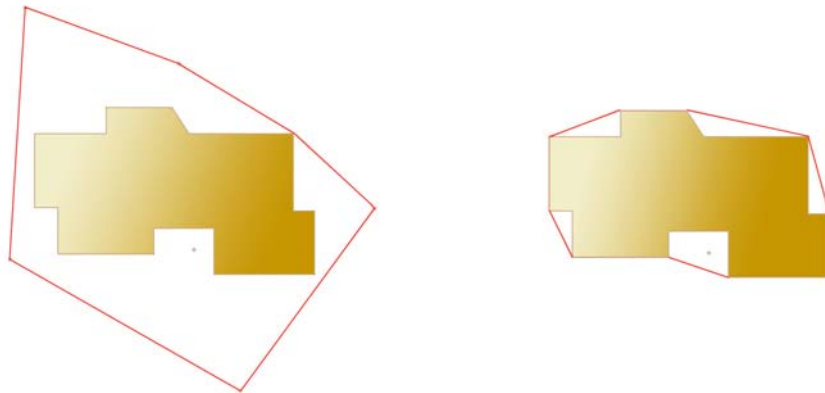


Figure 3.2 le changement du nombre de côtés et du volume de l'enveloppe convexe selon l'importance du bruit.

Nous pouvons encore calculer le volume du polyèdre convexe qui est une mesure plus signifiante que le nombre des côtés. Ce volume diminue quand l'objet se débarrasse de son bruit. Nous avons suggéré 1/50 du volume initial comme un seuil de changement pour pouvoir arrêter la procédure d'élimination itérative (la condition d'arrêt est une différence de volume de l'enveloppe convexe inférieur de 1/50 du volume initial). Nous avons laissé le choix pour l'utilisateur de changer la valeur proposée pour tout cas spécial.

Cette méthode est très efficace en ce qui concerne les points hors de l'objet, tandis qu'elle rencontre un échec quand les points se situent entre deux extensions de l'objet où quand le rayon de scan passe à travers le verre.

3.2 Où cherchons-nous les droites ?

Puisque la taille d'un nuage de point peut atteindre des millions de points, il est difficile voire impossible de le manipuler d'un seul ensemble. La nécessité de décomposer le nuage en parties est issue de cette constatation. La décomposition se fera selon les besoins de l'utilisateur.

Au niveau technique, il nous faut des parties de nuage de points, où des arêtes sont détectables ; c'est-à-dire à l'intersection de deux plans. Nous allons désormais appeler les points localisés à l'intersection « les points potentiels ». Nous allons présenter deux façons pour les détecter :

- La normale d'un point : En effet, pour chaque point, il existe un algorithme (ex. Spin image) permettant de calculer une surface locale passant par les points avoisinants. Il attribue ensuite la normale du plan à ces points. En d'autres termes, la normale du point équivaut à la normale du plan théorique adapté localement pour ce point.
- La méthode de triangulation : Cette méthode se base sur la construction de surfaces et sur le maillage automatique. L'unité de surface est alors le triangle et les normales sont étudiées pour chaque triangle. Quand une différence importante de direction est détectée, on perçoit l'existence d'une arête.

La deuxième méthode propose une construction de formes, préalable à la recherche des arêtes ; même s'il s'agit d'une construction approximative qui impose un traitement inverse à la logique traditionnelle. Le temps nécessaire pour mailler les nuages laser est assez important au niveau du processeur (comme la fonction Delaunay sur Matlab® par exemple). Pour cela des méthodes itératives ont été mises en place, comme par exemple celle qui commence par alléger le nuage aléatoirement (un point sur n point), puis qui construit la triangulation de Delaunay. Elle calcule ensuite les normales pour chaque triangle et elle détecte les points correspondant à un changement important dans les directions des normales. Enfin, elle densifie le nuage où un changement important de normales est observé. La solution se réitère jusqu'à ce qu'on ne puisse plus affiner le voisinage d'une arête.

On a utilisé la première méthode (la normale d'un point) pour pouvoir profiter des normales qui sont calculées pour chaque point importé dans le logiciel Realworks®. Il faut rappeler que les deux algorithmes précédents n'excluent pas seulement les points avoisinants des lignes, mais également ceux liés aux autres causes (courbure importante, présence de feuilles d'arbres, les lignes du début de scan ...etc.). La figure 3.3 montre les points potentiels (les points de discontinuité) pour un nuage de points.

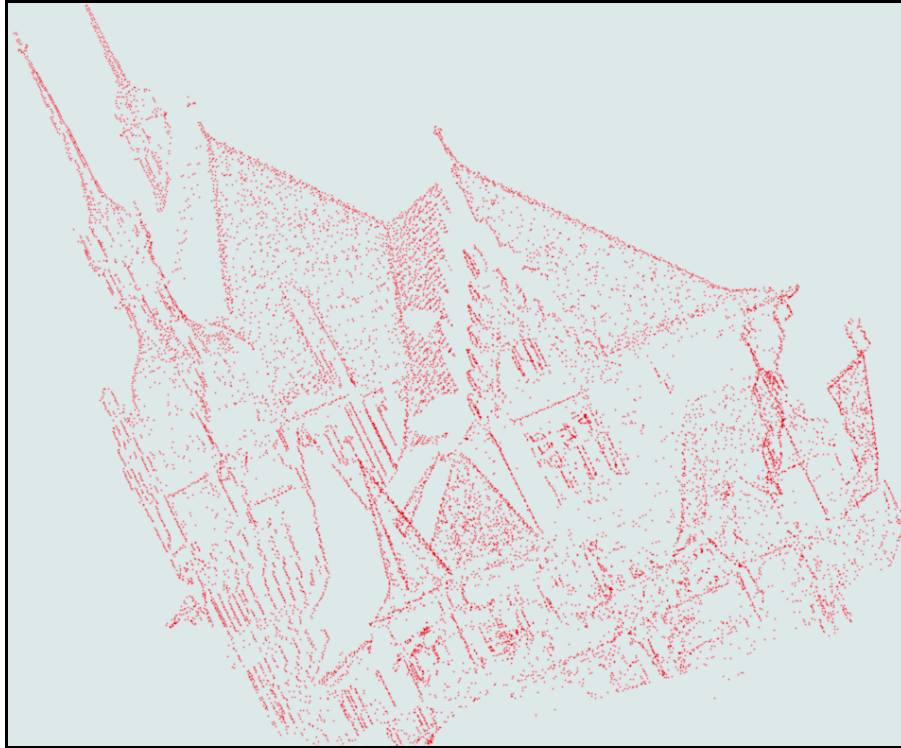


Figure 3.3 les points potentiels (candidats) pour extraire les arêtes

3.3 Allure générale d'une droite dans l'espace :

Chaque ligne dans l'espace peut être déterminée par un point qui lui appartient $Po(xo, yo, zo)$ et par une direction $v = (vx, vy, vz)$. Une fois que nous avons les six paramètres, nous pouvons écrire l'équation de la droite à l'aide d'une des deux formules suivantes :

La formule paramétrique :

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} xo \\ yo \\ zo \end{pmatrix} + t \times \begin{pmatrix} vx \\ vy \\ vz \end{pmatrix} \quad (4.1)$$

La formule symétrique :

$$\frac{x - xo}{vx} = \frac{y - yo}{vy} = \frac{z - zo}{vz} \quad (4.2)$$

Cette forme impose qu'aucun des dénominateurs ne valent zéro, il faut donc reformuler les équations en fonction des différents cas. La première formule, quant à elle, est plus maniable au niveau de programmation.

Les deux définitions imposent la connaissance des 6 paramètres, qui ne sont pas indépendants. En effet, la ligne peut être totalement définie par 4 paramètres (si elle n'est pas parallèle à un des plans xy , yz ou xz). Comme nous désirons trouver la droite la plus probable pour un ensemble de points, il faut écrire ces équations en forme minimale, autrement dite, enlever les paramètres redondants.

Puisque $Po(xo, yo, zo)$ est un point aléatoire sur la droite, nous pouvons le proposer comme étant à l'intersection avec le plan xy ($zo = 0$). Nous excluons alors le cas dans lequel les droites sont parallèles au plan xy . Selon la dernière proposition nous pouvons encore rééchantillonner les éléments du vecteur de la direction par vz , de la manière suivante : $(vx1, vy1, 1)$ où : $vx1 = vx/vz, vy1 = vy/vz$. Par conséquent, les équations paramétriques d'une droite deviennent :

$$\begin{aligned} x &= x0 + vx \cdot z \\ y &= y0 + vy \cdot z \end{aligned} \quad (4.3)$$

Rappelons qu'il est possible de récupérer le vecteur d'unité de la droite en divisant $(vx1, vy1, 1)$ par sa norme. Il est possible aussi de changer le point d'origine $Po(xo, yo, zo)$ en suggérant une valeur quelconque de z_0 , quoique celle de l'intersection avec le plan xy est plus signifiante et compréhensible

[MMB01] ont proposé une autre représentation de la ligne droite pour laquelle les cas dégénérés sont limités à deux : une droite passant par l'origine et une droite horizontale qui intersecte l'axe Z :

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \lambda \begin{pmatrix} \sin \beta_1 \sin \beta_3 - \cos \beta_1 \sin \beta_2 \cos \beta_3 \\ \cos \beta_1 \sin \beta_3 + \sin \beta_1 \sin \beta_2 \cos \beta_3 \\ \cos \beta_2 \cos \beta_3 \end{pmatrix} + q \begin{pmatrix} \cos \beta_1 \cos \beta_2 \\ -\sin \beta_1 \cos \beta_2 \\ \sin \beta_2 \end{pmatrix} \quad (4.4)$$

Les paramètres sont illustrés par la figure 3.4

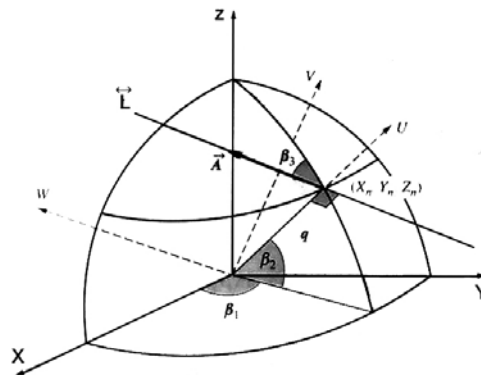


Figure 3.4 les paramètres de l'équation d'une ligne dans l'espace

On doit préparer les équations sous forme matricielle ($A \cdot x = b$) pour passer, par la suite, à solution itérative des moindres carrés. Si l'on considère les équations (4.3), on peut écrire pour un ensemble de points (p_1, p_2, \dots, p_n) :

$$\begin{pmatrix} 1 & 0 & z_1 & 0 \\ 0 & 1 & 0 & z_1 \\ 1 & 0 & z_2 & 0 \\ 0 & 1 & 0 & z_2 \\ \cdot & \cdot & \cdot & \cdot \\ 1 & 0 & z_n & 0 \\ 0 & 1 & 0 & z_n \end{pmatrix}_{2n \times 4} \begin{pmatrix} x_0 \\ y_0 \\ vx \\ vy \end{pmatrix}_{4 \times 1} = \begin{pmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ \cdot \\ x_n \\ y_n \end{pmatrix}_{2n \times 1} \quad (4.5)$$

La méthode par les moindres carrés aboutie à la détermination de l'erreur moyenne quadratique totale et des résidus aussi. Une fonction a été développée de façon à ce que les coordonnées du point d'origine, le vecteur de direction, l'erreur moyenne quadratique et le résidu maximal, soient sauvegardés dans un vecteur de 8 éléments qui seront utilisés plus tard. Les cas spéciaux (les droites quasi horizontales ou horizontales) peuvent être distingués préalablement au calcul si tous les points ont les « même » valeurs de coordonnée verticale z , ceci peut être découvert si leur écart-type est très faible. Il faudra alors reformuler les équations de manière à transférer le terme « z » vers le côté gauche. Nous avons évité l'utilisation de la formule (4.4) car elle exige une linéarisation lors de l'utilisation de la méthode des moindres carrés. En effet, elle nécessite plus de temps pour converger. Le temps d'exécution est un impact important quand il s'agit de la manipulation de nuages de milliers de points.

3.4 Tracer la droite

Pour représenter une droite graphiquement il faut tracer le segment linéaire correspondant au semis de points issus de la modélisation. La meilleure représentation est atteinte si nous trouvons les deux points les plus espacés du semis de points puis leurs projetés sur la droite dont l'équation est déjà déterminée. Il suffit de tracer un segment entre ces deux points.

Des cas spéciaux peuvent être traités séparément ; pour les droites verticales il est possible de trouver les deux points les plus espacés, garder leurs valeurs de z puis substituer leurs valeurs de x, y par celles x_0, y_0 des points de l'intersection de la droite avec le plan horizontal (paramètres de la droite).

Dans tous les cas, il serait utile de calculer la longueur du segment qui n'est que la distance entre les deux points à partir desquels nous traçons notre droite.

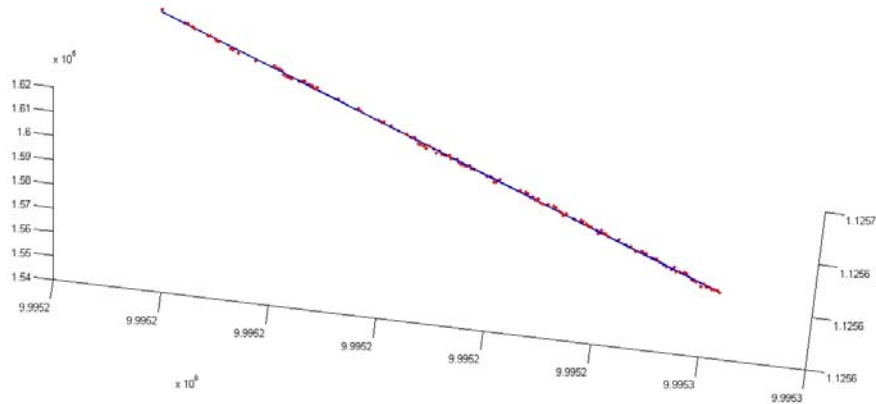


Figure 3.5 la représentation graphique d'une droite

3.5 La détection des droites :

Avant commencer à expliquer les algorithmes mis en place pour extraire les lignes, il faut rappeler que si nous réussissions à détecter la direction et la longueur des segments linéaire dans un nuage, la solution mathématique du problème serait alors bien plus simple si nous détectons seulement leur direction. Sinon un effort important sera fait pour savoir quelle portion est commune entre deux ligne ou pour limiter la recherche sur la direction. Des détails supplémentaires sur l'accord entre les segments de longueur finie et infinie peuvent être trouvés dans [Kam01]. Nous allons ouvrir le problème en raison de l'existence des lignes longues ayant une portion dans un nuage et une portion dans un autre, bien que la détection de certaines droites connues au niveau de leur longueur (dans la zone du recouvrement) facilite la solution.

Théoriquement, pour un semis de n points, il nous faudrait un nombre d'essais valant la combinaison C_n^2 pour y trouver toutes les arêtes. Il existe beaucoup d'algorithmes pour réduire ce nombre d'essais qui s'intitule : l'expulsion de complexité. Dans ce qui suit nous expliquerons trois algorithmes pour extraire les lignes. Nous essayerons également de comparer leur vitesse et leurs avantages et inconvénients.

3.5.1 La méthode de RANSAC

Plusieurs modifications doivent être appliquées sur l'algorithme de RANSAC concernant sa forme de base [FB81]. L'entrée de cette forme est un ensemble de valeurs de données (un semis de points), un modèle pour juger l'accord entre les données (les paramètres de l'équation d'une ligne dans l'espace) et quelques paramètres de confiance. Sa sortie est une seule détermination du modèle à juger qui représente la majorité disciplinée des données. Dans notre cas la détermination de ce modèle n'est pas unique, car plusieurs lignes existent dans le nuage de points. Ceci impose des modifications (expliquées ci-dessous) sur la forme de base de RANSAC.

RANSAC achève son but en choisissant itérativement un couple aléatoire des points. Pour ce couple, une ligne est calculée, c-à-d que tous les paramètres libres du modèle sont reconstruits pour cette ligne. Tous les autres points sont examinés par rapport à la ligne adaptée. Le test est le suivant : la distance dans l'espace entre la ligne proposée et un point du nuage doit être inférieur à un seuil donné S . Quand le test produit un nombre n de points supérieur à un autre seuil N donné, la ligne est acceptée comme

sous-modèle potentiel (un cas de réussite). Le sous-ensemble des points admis est modélisé par les moindres carrés pour avoir une droite qui s'approche le plus de tout le sous-ensemble et non seulement de l'échantillon pris. Le sous ensemble est exclu en cas de réussite et le processus est redémarré pour en trouver un autre.

La sélection des seuils précédents est très importante pour la fonctionnalité de RANSAC. Il nous faudra un choix garantissant le nombre le plus important de lignes, sans tomber dans le piège de perdre la précision de la modélisation, Nous avons trouvé que le choix d'une distance S valant deux fois la résolution linéaire du scan apporte des bons résultats. Pour un scan de résolution de 2 cm sur l'objet, nous trouvons qu'un choix de nombre minimal de points N identifié à 30-40 points pour admettre l'existence d'une droite apporte des résultats significatifs aussi.

RANSAC est encore sensible à un autre paramètre qui est le nombre d'itérations suffisantes pour juger que les données ont été raisonnablement testées. Soit p la probabilité de deux échecs consécutifs, $s = 2$ le nombre minimum de points nécessaire pour l'estimation des paramètres et r le pourcentage prévu de trouver des points modélisable par des lignes droites. Le nombre m de tirages aléatoires nécessaire pour avoir les bons paramètres est :

$$m = \frac{\log(1-p)}{\log(1-r^s)} \quad (4.6)$$

Nous avons attribué les valeurs $p = 0.99, r = 0.02$ afin d'avoir suffisamment de lignes.

Comme l'algorithme supprime un sous-ensemble du nuage chaque fois qu'une droite est trouvée, une diminution de la taille initiale du nuage apparait également. Cette propriété peut être utile pour défendre l'extra itération. Nous suggérons une double condition pour cesser l'itération de RANSAC : Soit quand le nombre d'itérations atteint le plafond m soit quand la taille du nuage diminue de moitié.

Malgré la faible probabilité de tomber sur la même couple de points deux fois, nous avons rejeté ce cas spécial en construisant une liste de couples déjà pris, comme une dernière modification sur l'algorithme de RANSAC.

3.5.2 La méthode incrémentale

La philosophie de cette méthode est très simple : Nous commençons par les deux premiers points du nuage, nous passons une droite par ces deux points, nous ajoutons un point encore en gardant l'œil sur l'erreur de la modélisation. Si cette erreur n'excède pas un certain seuil, nous rajoutons encore un point et ainsi de suite. Quand l'erreur de modélisation dépasse le seuil proposé nous percevons un saut d'un modèle à l'autre. Nous revenons sur les points précédents ; si leur nombre est supérieur à un autre seuil N proposé. Nous les modélisons par les moindres carrés. La considération du maximum des résidus comme l'erreur de modélisation est plus signifiant que prendre l'erreur moyenne quadratique en compte lors de la modélisation par les moindres carrés.

Il s'avère qu'un tel algorithme fonctionne bien si le nuage est trié. Pour nos points potentiels ce n'est pas le cas. Nous pouvons tout simplement trier les points selon leurs distances croissantes de premier point ou de point du scan. Nous pouvons encore améliorer la solution si le tri du nuage est effectué à chaque itération par rapport au dernier point pris.

Il se peut que deux points successifs soient éloignés de telle manière qu'il n'ait pas de droite qui passe par eux. Nous avons introduit une contrainte supplémentaire pour basculer d'un modèle vers un autre. Si la distance entre deux points successifs dépasse deux fois le seuil de distance proposé, nous percevons un changement de modèle.

Puisque la méthode est incrémentale dans une seule direction (marche avant), il est possible d'éliminer les points testés si nous avons un échec ou une réussite. L'algorithme s'itère jusqu'à ce que tous les points soient testés.

3.5.3 La transformée de Hough

Comme expliqué précédemment, la transformée de Hough (Standard) est une méthode classique de détection de lignes droites dans une image. Il est possible de l'appliquer pour les semis de points 2D sans aucune modification structurelle, néanmoins l'application en 3D impose une restructuration complète pour la méthode, de fait qu'une ligne dans l'espace est définie par deux équations.

Afin de pouvoir adapter la transformée standard avec nos données 3D, nous avons choisi la projection de ces points sur un des plans XY , YZ , XZ , ou encore l'utilisation de la projection perspective du point de vue donné par l'utilisateur. L'achèvement de cette dernière projection se fait à l'aide des coordonnées et de la position de la caméra qui sont récupérables à partir du visualiseur de Matlab. Une simple application des équations de colinéarité peut nous amener à la projection voulue. Nous allons chercher les droites dans l'espace 2D en sachant que :

- La projection d'une droite 3D est aussi une droite (selon les types de projections choisis)
- La conversion 2D/3D est toujours possible sans perdre de précision.

Partant de la dualité point-sinusoïde et point-droite, la transformée propose un détecteur de ligne droite dans un espace 2D. Nous avons théoriquement besoin d'un histogramme, sur lequel toutes les lignes passant par chaque point seront représentées. En pratique, nous sommes obligés d'adopter un certain degré de discrétisation, d'où le premier seuil qui est celui de la discrétisation angulaire entre toutes les lignes passant par chaque point. A cet égard, le choix de 0.5-1 degré donne de bons résultats.

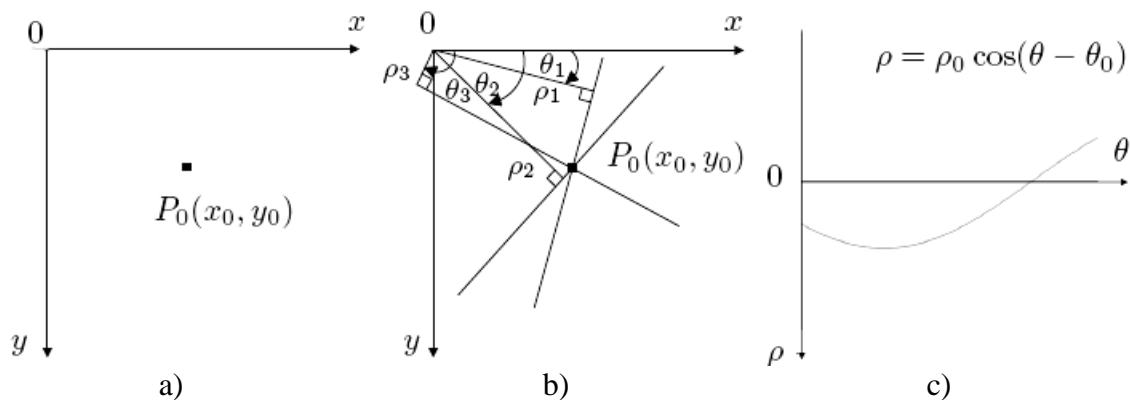


Figure 3.6 **a** : un point $P_0(x_0, y_0)$, **b** : l'ensemble des droites passant par P_0 , **c** : la sinusoïde $S(\theta, \rho)$ rattachée à chaque point P

Nous devons aussi choisir une plage de valeurs pour (θ, ρ) de telle manière que la taille du pixel sera significative par rapport à la précision voulue. Après avoir établi toutes les sinusoides, nous pouvons les tracer en formant un type d'image accumulative. Quand nous échantillons cette image sur une gamme de couleurs convenables (256 valeurs par exemple), nous pouvons l'appeler histogramme de Hough.

L'étape suivante serait la détection des maximums régionaux dans l'histogramme de Hough. Il est clair que les régions, où les maximums se trouvent, ont des propriétés de symétrie (forme de papillon). Nous obtenons les maximums régionaux, si nous regroupons toutes les régions supérieures à un certain seuil (0.7 fois la moyenne arithmétique des valeurs des pixels par exemple), puis si nous trouvons le centroïde de chaque région. C'est la méthode la plus simple pour cerner les points ayant le nombre maximum de sinusoides passant par eux, ou autrement dit les points ayant le nombre maximum de « votes ». Ces points (θ_i, ρ_i) se transforment par la suite en droites en appliquant l'équation :

$$x \cos \theta_i + y \sin \theta_i = \rho_i$$

ce qui aboutit à un fuseau de lignes 2D dont seule la direction est connue.

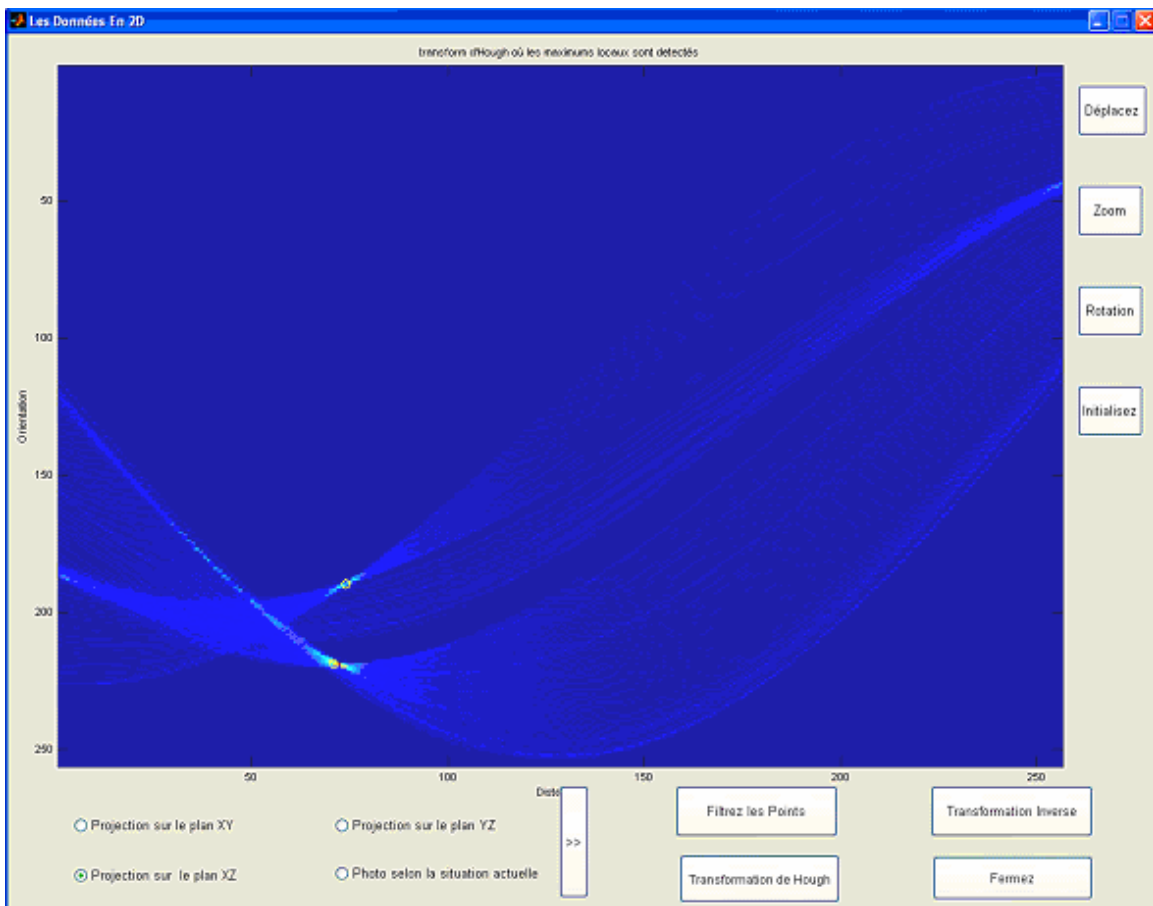


Figure 3.7 l'isolation des maximums régionaux dans l'histogramme de Hough

Il ne reste qu'à détecter les points du nuage projeté qui ne s'éloignent pas de chaque ligne d'une distance inférieure à un certain seuil. Si le nombre de ces points est supérieur à un autre seuil nous les isolons en un sous nuage modélisable par une droite dans l'espace 2D.

La projection inverse dans l'espace 3D produit les mêmes sous nuages mais en 3D. Une simple modélisation par les moindres carrés est suffisante pour obtenir les droites 3D finales.

3.5.4 La fusion des segments

Les procédures décrites ci-dessus tendent généralement à produire des segments linéaires qui appartiennent réellement à la même droite. Ce phénomène s'appelle habituellement la sur-segmentation (over segmentation). Pour surmonter cet obstacle, il serait nécessaire d'éliminer les segments fallacieux en fusionnant les segments contigus. La fusion se base sur les deux conditions suivantes :

- La distance entre les deux segments est inférieure à un seuil donné.
- La droite représentant les deux segments (si on considère les sous-nuages ayant été modélisés par les deux segments) est d'orientation similaire aux deux segments originaux.

Pour pouvoir traduire techniquement ces deux conditions, nous devons calculer la distance minimale entre les deux sous-nuages. Cette distance est la première mesure à comparer pour la fusion. La deuxième mesure advient du produit scalaire ; pour deux segments $AB, A'B'$, nous pouvons écrire :

$$\cos \theta = \frac{\overline{AB} \times \overline{A'B'}}{\|AB\| \times \|A'B'\|} \quad (4.7)$$

Où θ est l'angle entre $AB, A'B'$. Cet angle est la deuxième mesure à comparer avec un seuil donné pour pouvoir effectuer la fusion. Il faut également le comparer avec l'angle entre chacun des deux segments et la droite liant leurs centroïdes. Une autre formule basée sur les directions (v_x, v_y) calculées par l'équation (4.3), fournit une mesure plus aisée pour la programmation :

$$i = \arctan \frac{v_y}{v_x} \quad (4.8)$$

L'algorithme de fusion a comme entrée une table comprenant toutes les droites issues des étapes précédentes. Cette table se changerait au fur et à mesure puisqu'à chaque itération nous fusionnons deux segments et nous restructurons la table. L'algorithme fonctionne de manière récursive jusqu'à ce que nous puissions plus fusionner de segments. La sortie de la procédure serait, par conséquent, la même table après modifications.

Pour éviter d'introduire de nouveaux seuils, nous pouvons utiliser ceux de la distance issue de RANSAC ou de l'algorithme incrémentale. Un seuil de 3-4 fois l'erreur maximale de modélisation au niveau de distance aboutit à des résultats satisfaisants, tandis qu'un seuil angulaire de 0.1-0.2 degré améliore encore la fusion. Il faut faire attention à tolérer les seuils élevés, notamment pour la mise en correspondance postérieure qui est très sensible à la précision de la direction des droites.

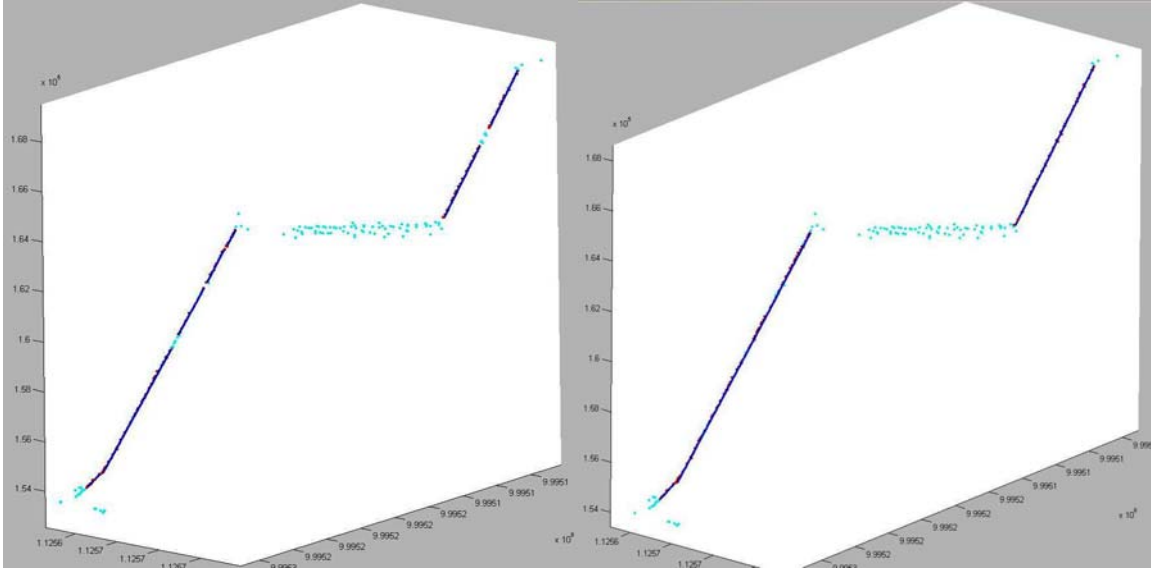


Figure 3.8 les résultats de la méthode incrémentale avant et après la fusion.

3.5.5 Un cas particulier

Il faut finalement s'aider de la logique architecturale pour affiner la définition des quelques lignes extraites. Si le calcul d'une arête droite engendre un vecteur de direction $[v_x, v_y, v_z]$ qui vaut $[-0,00036048 -0,00051261]$ par exemple, nous percevons que cette ligne n'est qu'une ligne verticale, ce qui impose une redéfinition du vecteur calculé de telle manière qu'il devient $[0 0 1]$. Un cas similaire peut être constaté pour les droites ayant une direction « proche » de $[0 1 0]$ ou de $[1 0 0]$ qui sont horizontales. Pour notre étude l'approximation sera faite pour toutes valeurs inférieures à 0.01 (comme un seuil de direction) ce qui garantit une meilleure correspondance entre les droites dans le monde « objet » et leur représentation dans le monde « modèle ».

3.6 La méthode ICP de base

Premièrement, il sera utile de définir le point le plus proche d'un autre situé dans un semis de points. La distance Euclidienne $d(\vec{r}_1, \vec{r}_2)$ entre deux points $\vec{r}_1(x_1, y_1, z_1)$ et $\vec{r}_2(x_2, y_2, z_2)$ est égale à :

$$\|\vec{r}_1 - \vec{r}_2\| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}. \quad (4.9)$$

Soit A un ensemble de points \vec{a}_i . La distance entre un point p et l'ensemble A est égale à :

$$d(\vec{p}, A) = \min_{i \in \{1, 2, \dots, N\}} d(\vec{p}, \vec{a}_i). \quad (4.10)$$

Le point \vec{a}_j le plus proche de p est celui qui satisfait l'égalité :

$$d(p, \vec{a}_j) = d(p, A) \quad (4.11)$$

Nous allons appeler le point \vec{a}_j : la projection réelle du point p sur l'ensemble des points A .

La tâche d'appariement des points consiste à établir une bijection entre chaque point du nuage P « scène ou données » et son point le plus proche du nuage X « modèle ». L'astuce ici est de ne pas lier plus d'un point à un autre ; pour ce faire, il faudra une procédure itérative qui couple, à chaque fois, les deux points les plus proches, puis elle les exclut de la recherche qui sera effectuée à l'itération suivante. Si nous lançons cette procédure en partant des deux points ayant la distance réciproque minimale et de manière ascendante, nous pouvons nous arrêter sur un seuil au-delà duquel nous n'apparions plus aucun couple. Ce seuil réduira remarquablement le temps d'exécution et facilitera le filtrage prochain.

Pour pouvoir limiter l'appariement des points sur la masse principale des points situés dans la zone de recouvrement, nous appliquons le filtrage de [Zha92], ceci dégrade l'effet du bruit et rend la transformation calculée plus significative, ce qui accélère l'algorithme. Nous constatons que le nombre de points (N_p) intervenant dans la transformation suivante sera égal au nombre de points dans le nuage minimal après avoir retranché le nombre de points filtrés dans l'étape précédente.

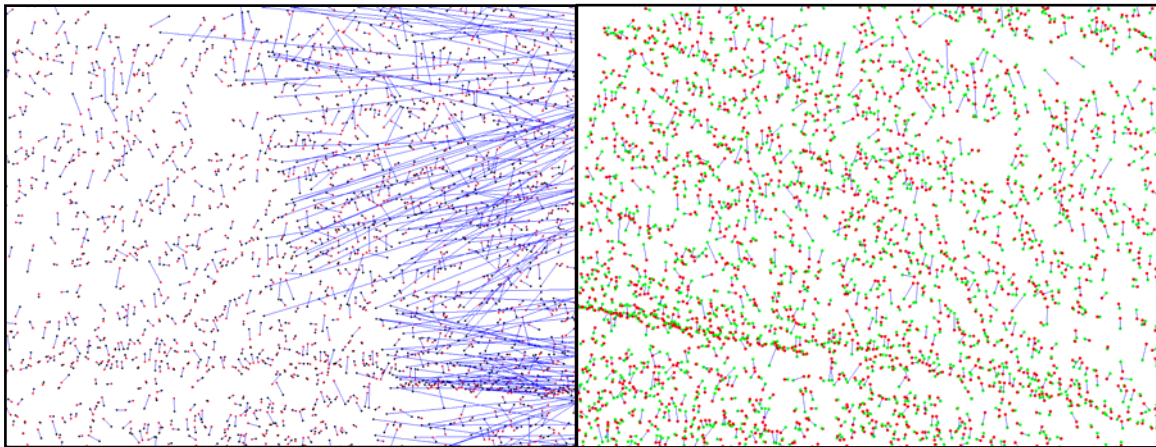


Figure 3.9 les points appariés avant et après le filtrage

Supposons que R , la matrice de rotation et T , le vecteur de la translation, sont les paramètres de la transformation rigide ; la fonction à minimiser est :

$$f(R, T) = \frac{1}{N_p} \sum_{i=1}^{N_p} \left\| \vec{x}_i - R\vec{p}_i - \vec{T} \right\|^2. \quad (4.12)$$

Une fois ces paramètres calculés, l'ensemble « donnée ou scène » p_i sera mis à jour. Nous devons appliquer les étapes précédentes de manière itérative jusqu'à ce que les distances ne dépassent plus un certain seuil. Pour pouvoir résoudre le système non linéaire surdéterminé (4.12) par les moindres carrés, [BM92] ont utilisé la forme fermée « closed-form » de la solution par les quaternions proposée par [Hor87] : Le centre de masse pour les semis P, X vaut respectivement :

$$\bar{\mu}_p = \frac{1}{N_p} \sum_{i=1}^{N_p} \bar{p}_i \quad \text{et} \quad \bar{\mu}_x = \frac{1}{N_x} \sum_{i=1}^{N_x} \bar{x}_i \quad (4.13)$$

La matrice de covariance avec retard (la covariance croisée) Σ_{px} pour les deux ensembles P, X est donnée par la formule :

$$\Sigma_{px} = \frac{1}{N_p} \sum_{i=1}^{N_p} (\bar{p}_i - \bar{\mu}_p)(\bar{x}_i - \bar{\mu}_x)^t = \frac{1}{N_p} \sum_{i=1}^{N_p} (\bar{p}_i \bar{x}_i^t - \mu_p \bar{\mu}_x^t) \quad (4.14)$$

Les composants cycliques de la matrice antisymétrique $A_{ij} = (\Sigma_{xp} - \Sigma_{xp}^T)_{ij}$ sont utilisés pour former le vecteur $\Delta = [A_{23}, A_{31}, A_{12}]$ qui sera utilisé par la suite pour former la matrice $Q(\Sigma_{px})$:

$$Q(\Sigma_{px}) = \begin{bmatrix} tr(\Sigma_{px}) & \Delta^T \\ \Delta & \Sigma_{px} - \Sigma_{px}^T - tr(\Sigma_{px})I_3 \end{bmatrix}_{4 \times 4} \quad (4.15)$$

Le vecteur de valeurs propres normalisées est sélectionné pour la rotation optimale $\bar{q}_R [q_0 \ q_1 \ q_2 \ q_3]$ représentée en quaternions. La matrice de rotation traditionnelle d'Euler $R(\bar{q}_R)$ est calculée par la formule :

$$R(\bar{q}_R) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_2 q_1 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_3 q_1 - q_0 q_2) & 2(q_3 q_2 - q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (4.16)$$

tandis que le vecteur de la translation \bar{q}_T peut s'écrire :

$$\bar{q}_T = \bar{\mu}_x - R(\bar{q}_R) \bar{\mu}_p \quad (4.17)$$

Une autre forme fermée de solution plus facile à mettre en œuvre, est celle de la décomposition en valeurs singulières SVD proposée par [AHB87] ; il part de la décomposition de matrice Σ_{px} eq (4.14) :

$$\Sigma_{px} = USV^T \quad (4.18)$$

la matrice de rotation R est égale tout simplement à :

$$R = VU^T \quad (4.19)$$

la translation T serait calculée par la même formule (4.17).

Nous avons adopté la solution de SVD pour notre étude, puisqu'elle nécessite moins de temps à s'exécuter sur Matlab®. [LEF95] ont démontré empiriquement que la méthode SVD produit des résultats légèrement plus précis que celle des quaternions, ce qui est très important dans notre cas, où on affine la superposition entre deux nuages de points.

3.7 ICL : La méthode ICP adaptée aux droites

La méthode proposée suit les mêmes étapes que celle de base, sauf que les entités de base ne sont plus des points mais des droites. Toutes les notions déjà citées doivent donc être adaptées pour les droites.

3.7.1 Appariement des droites

L'appariement des lignes se base uniquement sur la distance entre deux lignes (annexe A 6.1.3). Nous apparions, entre deux nuages, les deux droites les plus proches à chaque fois. Quand la solution se met à converger, nous pouvons introduire une condition concernant les directions des droites (cf. La fusion des segments 3.5.4), nous apparions les droites quasi parallèles et dans une étape suivante (la dernière itération par exemple) nous pouvons exclure les droites dont la distance est inférieure à un seuil proposé.

En général, nous combinons chaque droite avec celle la plus proche, puis nous effectuons la transformation rigide pour le nuage scène à partir du couplage actuel. Nous répétons le couplage en partant de la pose actuelle du nuage scène (après l'avoir transformé), ce qui nous donne les paramètres suffisants pour pouvoir effectuer une autre transformation rigide et ainsi de suite. Quand nous notons que le couplage fait est le même que celui déjà effectué, nous cessons l'itération de solution. La rotation et la translation liées à la dernière transformation accomplie sont adaptées.

Le cas spécial où nous disposons des coordonnées approchées est un environnement idéal pour injecter la condition de direction et le seuil de la distance. Dans cette étape nous affectons l'infinité comme valeur de distances entre deux droites « non parallèles » pour interdire leur appariement.

Il faut remarquer aussi qu'un bon appariement prenant en considération les conditions mentionnées, peut servir de filtre supplémentaire qui exclut les droites ayant une détermination faible au niveau du nombre de points ou de marge de la tolérance imposée. D'une façon générale, nous avons constaté que presque 50% des droites détectées ne seront pas appariées, c'est non seulement à cause de leur mauvaise détermination, mais également en raison d'existence de lignes qui ne correspondent à aucune entité géométrique (lignes de début ou coupure du scan).

3.7.2 Calcul des composants de la transformation rigide

Nous proposons ici deux formes de la solution ; la première est identique à celle d'ICP de base lorsqu'il s'agit du calcul de matrice de rotation, il suffit alors de remplacer les coordonnées par les composants du vecteur d'orientation. La deuxième forme est très différente de celle d'ICP de base si nous considérons que les solutions proposées sont itératives et non de forme fermée. Cependant les deux méthodes présentent une différence importante d'ICP de base quand il s'agit de la translation comme nous allons le voir plus tard.

3.7.2.1 Forme ICP :

Supposons que nous disposons de deux groupes G_1, G_2 de droites extraites automatiquement du premier et du deuxième nuage respectivement

$$\begin{aligned} G_1 &: (v_i, p_i), & i = 1 \dots N \\ G_2 &: (v'_i, p'_i), & i = 1 \dots N \end{aligned}$$

la matrice de rotation peut être calculée en bénéficiant seulement de l'orientation v_i, v'_i de chaque couple de droites ; ceci est donné par la fonction d'erreur à minimiser :

$$Err(R) = \sum_{i=1}^N \left\| v'_i - R v_i \right\|^2 \quad (4.20)$$

où R est la matrice de rotation. La minimisation de cette équation passe sous une forme fermée de solution si la matrice de rotation a été exprimée en tant que quaternions. Il est démontrable que deux paires de droites couplées sont suffisantes pour la solution, néanmoins la qualité de solution accroît de plus en plus quand nous rajoutons d'autres couples.

Nous allons suivre les mêmes étapes réalisées pour la formule 4.20 ; pour ce faire, nous définissons premièrement :

$$\mu_v = \frac{1}{N} \sum_{i=1}^N v \quad \text{et} \quad \mu_{v'} = \frac{1}{N} \sum_{i=1}^N v' \quad (4.21)$$

La matrice de la covariance croisée $\Sigma_{v,v'}$ pour les deux ensembles de direction est donnée par la formule :

$$\Sigma_{v,v'} = \frac{1}{N} \sum_{i=1}^n (v_i v_i'^T - \mu_v \mu_v'^T) \quad (4.22)$$

Les composants cycliques de la matrice antisymétrique $A_{ij} = (\Sigma_{v,v'} - \Sigma_{v,v'}^T)_{ij}$ sont utilisés pour former le vecteur $\Delta = [A_{23}, A_{31}, A_{12}]$ utilisé à son tour pour former la matrice $Q(\Sigma_{v,v'})$:

$$Q(\Sigma_{v,v'}) = \begin{bmatrix} tr(\Sigma_{v,v'}) & \Delta^T \\ \Delta & \Sigma_{v,v'} - \Sigma_{v,v'}^T - tr(\Sigma_{v,v'}) I_3 \end{bmatrix}_{4 \times 4} \quad (4.23)$$

encore une fois, le vecteur de valeurs propres normalisées est sélectionné pour la rotation optimale $\vec{q}_R [q_0 \ q_1 \ q_2 \ q_3]$ représentée en quaternions. La matrice de rotation traditionnelle d'Euler $R(\vec{q}_R)$ est calculée par la formule :

$$R(\vec{q}_R) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_2 q_1 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_3 q_1 - q_0 q_2) & 2(q_3 q_2 - q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (4.24)$$

Dès que nous trouvons la rotation, la détermination du vecteur de la translation $T(T_x \ T_y \ T_z)$ serait relativement aisée. Prenons deux points aléatoires (a_1^i, a_2^i) sur la i ème ligne (v_i, p_i) du nuage modèle, ces points peuvent être exprimés de la manière suivante :

$$a_1^i = p_i + t_1 v_i \qquad a_2^i = p_i + t_2 v_i \qquad (4.25)$$

Ils ont deux points homologues correspondant sur la i ème droite appariée du nuage scène :

$$a_1^{i'} = p_i' + t_1' v_i' \qquad a_2^{i'} = p_i' + t_2' v_i' \qquad (4.26)$$

La correspondance entre les deux lignes nous fournit la contrainte suivante :

$$\begin{aligned} a_1^{i'} &= R a_1^i + T & p_i' + t_1' v_i' &= R(p_i + t_1 v_i) + T \\ a_2^{i'} &= R a_2^i + T & p_i' + t_2' v_i' &= R(p_i + t_2 v_i) + T \end{aligned} \qquad (4.27)$$

Sachant que la matrice de rotation R est connue de l'étape précédente, le système de six équations (4.27) possède sept inconnus (les trois composants de la translation T et les quatre variables réelles t_1, t_2, t_1', t_2'). Il nous faut donc au moins un autre couple de droites pour accomplir le calcul. Quand nous avons deux couples (ou plus) de droites, le système des équations (4.27) sera surdéterminé et pourra être résolu par les moindres carrées.

Les étapes de la méthode traduisent clairement la différence entre le problème non-linéaire de rotation et celui linéaire de translation. Nous pouvons constater aussi qu'il n'existe aucune condition préalable à valider, l'algorithme proposé ici est donc valable pour tous les cas en général.

3.7.2.2 Forme alternative :

Cette méthode est inspirée de [HG04] qui essaye de conjuguer un modèle 3D photogrammétrique avec un autre modèle laser pour le même objet. Il utilise les droites pour calculer les paramètres de similitude spatiale qui serait une transformation rigide si nous considérons le facteur d'échelle égal à 1. Ce cas est identique au nôtre sauf qu'il suppose que les segments linéaires appariés ont la même longueur. Cette contrainte nous oblige à projeter (voir. Annexe A 6.1.4) les extrémités des segments « scène » sur les droites qui leur sont homologues du nuage « modèle » (ou le contraire) pour unifier leur longueur. Il est ensuite possible de démarrer la méthode.

Reportons nous à la figure 3.10 ; les deux points A et B représentent un segment du nuage modèle (ils n'ont pas besoin d'être transformés), 1 et 2 délimitent un segment du nuage scène. Il est clair que le but de l'opération suivante n'est pas de conjuguer les deux paires de points mais de conjuguer les segments délimités par ces derniers.

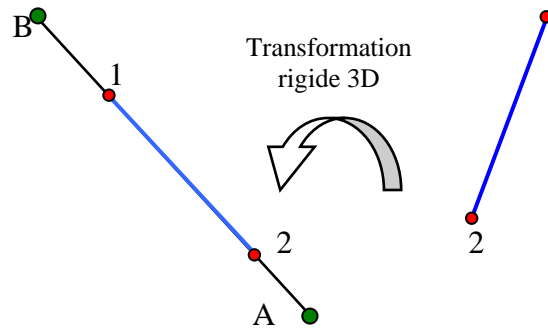


Figure 3.10 la transformation rigide entre deux segments linéaires

Nous pouvons exprimer la transformation rigide par les deux équations suivantes :

$$\begin{aligned} \begin{bmatrix} X_T \\ Y_T \\ Z_T \end{bmatrix} + R_{(\omega, \varphi, \kappa)} \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} &= \begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} + \lambda_1 \begin{bmatrix} X_B - X_A \\ Y_B - Y_A \\ Z_B - Z_A \end{bmatrix} \\ \begin{bmatrix} X_T \\ Y_T \\ Z_T \end{bmatrix} + R_{(\omega, \varphi, \kappa)} \begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \end{bmatrix} &= \begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} + \lambda_2 \begin{bmatrix} X_B - X_A \\ Y_B - Y_A \\ Z_B - Z_A \end{bmatrix} \end{aligned} \quad (4.28)$$

où $[X_T \ Y_T \ Z_T]^T$ est la translation entre l'origine du premier nuage et l'origine du deuxième, λ_1, λ_2 sont des facteurs d'échelle, $R_{(\omega, \varphi, \kappa)}$ est la matrice de la rotation exprimée par les trois angles d'Euler ω, φ, κ . (Annexe 6.1.5). Le fait d'écrire la matrice sous cette forme est à l'origine de la différence entre cette solution et la précédente (forme ICP).

La soustraction des deux équations 4.28 donne :

$$\begin{aligned} (\lambda_2 - \lambda_1) \begin{bmatrix} X_B - X_A \\ Y_B - Y_A \\ Z_B - Z_A \end{bmatrix} &= R_{(\omega, \varphi, \kappa)} \begin{bmatrix} X_2 - X_1 \\ Y_2 - Y_1 \\ Z_2 - Z_1 \end{bmatrix} \\ \begin{bmatrix} X_B - X_A \\ Y_B - Y_A \\ Z_B - Z_A \end{bmatrix} &= \lambda R_{(\omega, \varphi, \kappa)} \begin{bmatrix} X_2 - X_1 \\ Y_2 - Y_1 \\ Z_2 - Z_1 \end{bmatrix} \end{aligned} \quad (4.29)$$

L'équation 4.29 souligne le fait que les deux segments seront forcément parallèles après les avoir mis dans le même système de coordonnées. Pour éliminer le facteur λ nous divisons les premières deux lignes par la troisième :

$$\begin{aligned}\frac{X_B - X_A}{Z_B - Z_A} &= \frac{R_{11}(X_2 - X_1) + R_{12}(Y_2 - Y_1) + R_{13}(Z_2 - Z_1)}{R_{31}(X_2 - X_1) + R_{32}(Y_2 - Y_1) + R_{33}(Z_2 - Z_1)} \\ \frac{Y_B - Y_A}{Z_B - Z_A} &= \frac{R_{21}(X_2 - X_1) + R_{22}(Y_2 - Y_1) + R_{23}(Z_2 - Z_1)}{R_{31}(X_2 - X_1) + R_{32}(Y_2 - Y_1) + R_{33}(Z_2 - Z_1)}\end{aligned}\quad (4.30)$$

Pour pouvoir récupérer les composants de la translation nous récrivons l'équation (4.28) pour le point 1 du nuage scène :

$$\begin{bmatrix} X_T \\ Y_T \\ Z_T \end{bmatrix} + \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = \begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} + \lambda_1 \begin{bmatrix} X_B - X_A \\ Y_B - Y_A \\ Z_B - Z_A \end{bmatrix}\quad (4.31)$$

où $\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = R_{(\omega, \varphi, \kappa)} \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix}$ sont les coordonnées du point 1 après la rotation (voir figure 3.10).

Nous allons encore diviser les deux premières lignes par la troisième, ce qui donne l'équation suivante :

$$\begin{aligned}\frac{(X_B - X_A)}{(Z_B - Z_A)} &= \frac{(X_T + x_1 - X_A)}{(Z_T + z_1 + Z_A)} \\ \frac{(Y_B - Y_A)}{(Z_B - Z_A)} &= \frac{(Y_T + y_1 - Y_A)}{(Z_T + z_1 + Z_A)}\end{aligned}\quad (4.31)$$

De la même façon nous pouvons construire les deux équations suivantes pour le point 2 :

$$\begin{aligned}\frac{(X_B - X_A)}{(Z_B - Z_A)} &= \frac{(X_T + x_2 - X_A)}{(Z_T + z_2 + Z_A)} \\ \frac{(Y_B - Y_A)}{(Z_B - Z_A)} &= \frac{(Y_T + y_2 - Y_A)}{(Z_T + z_2 + Z_A)}\end{aligned}\quad (4.32)$$

Nous pouvons, enfin, réduire les deux paires d'équations (4.31),(4.32) en une seule :

$$\begin{aligned}\frac{(X_T + x_1 - X_A)}{(Z_T + z_1 - Z_A)} &= \frac{(X_T + x_2 - X_A)}{(Z_T + z_2 - Z_A)} \\ \frac{(Y_T + y_1 - Y_A)}{(Z_T + z_1 - Z_A)} &= \frac{(Y_T + y_2 - Y_A)}{(Z_T + z_2 - Z_A)}\end{aligned}\quad (4.33)$$

Une paire de lignes conjuguées produit deux paires d'équations qui déterminent à leur tour 4 inconnues des 6 de la transformation rigide ; il nous faut donc encore une autre paire. Plus nous disposons de paires de lignes, plus la précision de détermination augmentera. Un tel calcul exige cependant un système de moindres carrés expliqué dans l'annexe A (6.1.6.3)

4 Expérimentations

4.1 *Lycée des Pontonniers*

Fondé en 1815 à Strasbourg et situé sur un terrain de 4200 m²; le lycée des Pontonniers représente un ouvrage architectural important à Strasbourg. Le lycée comporte des salles de cours, une bibliothèque, un gymnase, les logements de la direction et du concierge. D'une façon générale, l'ensemble du lycée laisse apparaître l'aspiration à renouer avec l'architecture locale et à utiliser avec habileté les éléments venant de bâtiments depuis longtemps disparus. Par ce biais, on préserve de petits chefs-d'œuvre artistiques.



Figure 4.1 une vue générale de l'entrée principale du lycée

Un relevé lasergrammétrique exhaustif a été fait par l'équipe MAP-PAGE pour le compte de la région Alsace. Plusieurs stations disposés autour et à l'intérieur du site étaient indispensables pour relever un ouvrage d'une telle ampleur. Les coordonnées de toutes les stations ont été déterminées à l'avance grâce à une polygonation qui avait été faite préalablement, et différents types d'opérations topographiques (triangulation,

relèvement spatial, intersection...etc.) qui sont possibles avec le scanner topographique comme le nôtre (Trimble® GX) utilisé pour ce projet. Nous pouvons constater que tous les nuages sont superposables grâce au géoréférencement direct des nuages de points et la tâche à effectuer sera de transformer cette superposition en consolidation en appliquant la méthode proposée auparavant.

Pour achever la manipulation, nous avons choisi deux scans ayant une zone de recouvrement commune riche en entités linéaires. La figure 4.2 représente les deux nuages superposés (dans Realworks®) selon leurs coordonnées acquises directement par le scanner.

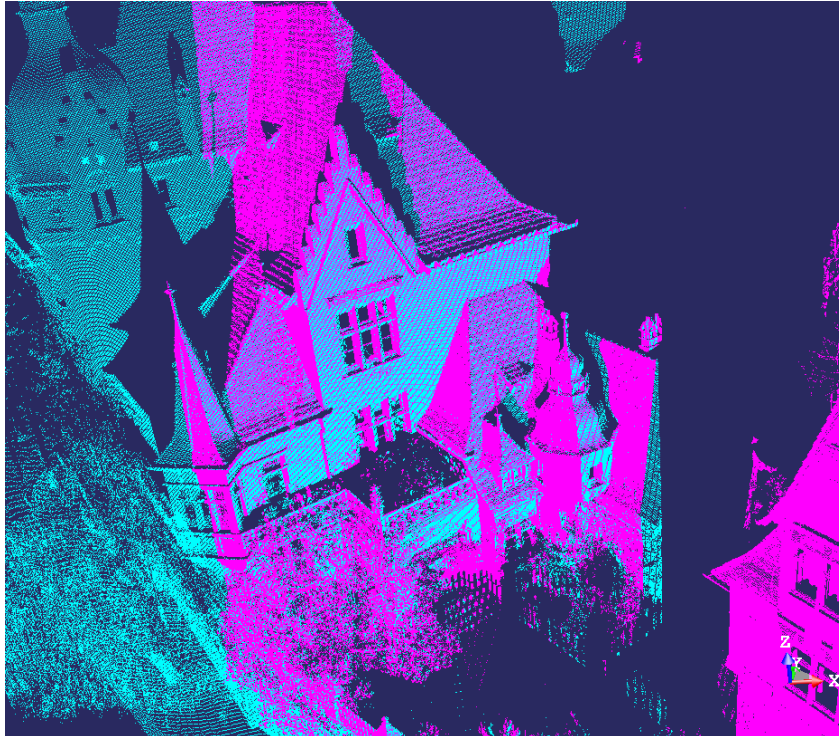


Figure 4.2 la zone de recouvrement entre les deux scans

Le premier scan a été effectué à partir de la station *R14*, la distance jusqu’au centre de l’objet scanné est approximativement égale à *46 m*. La résolution linéaire fournie était de *30 mm à 60 m*, nous pouvons donc percevoir un point tous les *2-3 cm* sur la partie principale du scan (salle des professeurs qui se situe derrière la terrasse). Le deuxième scan a le point *I13* comme origine qui s’éloigne d’environ *36 m* de la partie centrale du bâtiment scanné. La résolution fournie était de *120 mm à 100 mm* ce qui veut dire un point tous les *5-6 cm* sur la partie centrale de l’objet dans la direction verticale et horizontale. Dans la suite nous appellerons chaque nuage par le nom de son origine.

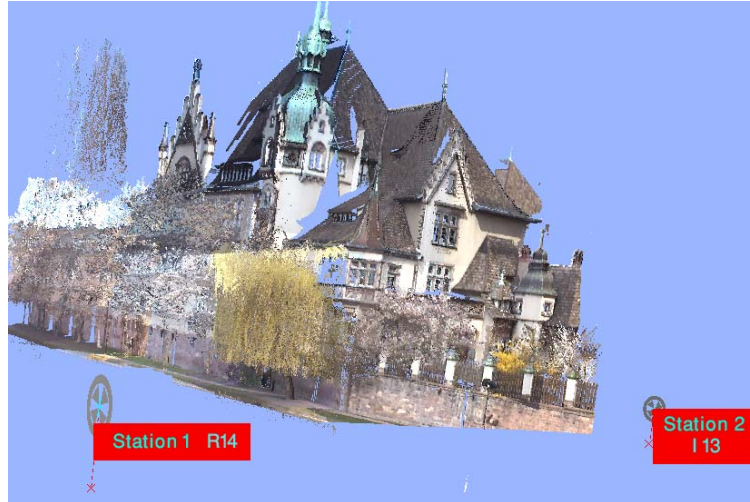


Figure 4.3 La position de deux origines du scan

Nous avons utilisé le logiciel Realworks® pour extraire les points potentiels qui sont de deux types : le pli et le pas (voir 2.2.2) [MG06]. Cette opération indispensable pour extraire les lignes dans les étapes suivantes, a permis de réduire considérablement le nombre de points dans chaque nuage.

Les résolutions précédentes sont très importantes pour décider quels seuils sont les plus appropriés pour les deux nuages. Nous avons utilisé un seuil de distance qui vaut 20 mm et un seuil de 30 point pour le nombre de points pour l'algorithme RANSAC. Une telle application engendre 58 droites pour le nuage R14. Pour le nuage I13, nous accordons 30 mm comme seuil de distance et 20 points comme seuil de nombre, ce qui a produit 33 droites.

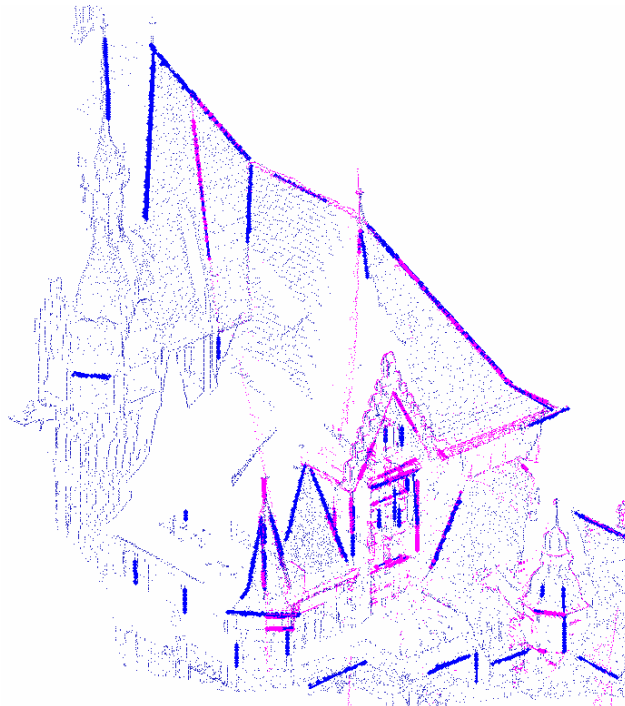


Figure 4.4 les points potentiels (candidats) et les droites extraites.

Le tableau (4.1) récapitule toutes les caractéristiques essentielles utilisées lors de l'extraction des droites

	R14	I13
coordonnées d'origine	(999519.899, 112515.435, 141.780)	(999528.744, 112523.804, 141.897)
résolution (linéaire)	30 mm à 60 m	100 mm à 120 m
nombre de points	601952	292706
nombre de points potentiels	9527	15401
seuil de nombre	30	20
seuil de distance	20 mm	35 mm
nombre de droites extraites	58	33
nombre de droites appariées	7	
seuils de couplage	60 mm (distance), 0.5 degré (direction)	

Table 4.1 les paramètres utilisés lors de l'extraction et de l'appariement des droites

La chute importante du nombre de droites appariées s'explique par le fait que le closoir des faîtières est couvert par des tuiles cylindriques (figure 4.1), c'est pourquoi la modélisation engendre une directrice du cylindre qui diffère en fonction du point de vue.

La figure (4.6) est une capture d'écran illustrant l'opération de consolidation (sans cible) dans le logiciel Realworks®. Il s'avère clairement que la précision issue de la consolidation (en utilisant tous les points des nuages), donne lieu à une précision (44mm) qui est inférieure à celle de la superposition elle-même. Cette constatation laisse supposer que : la consolidation pourrait être obsolète si la superposition était bien faite .

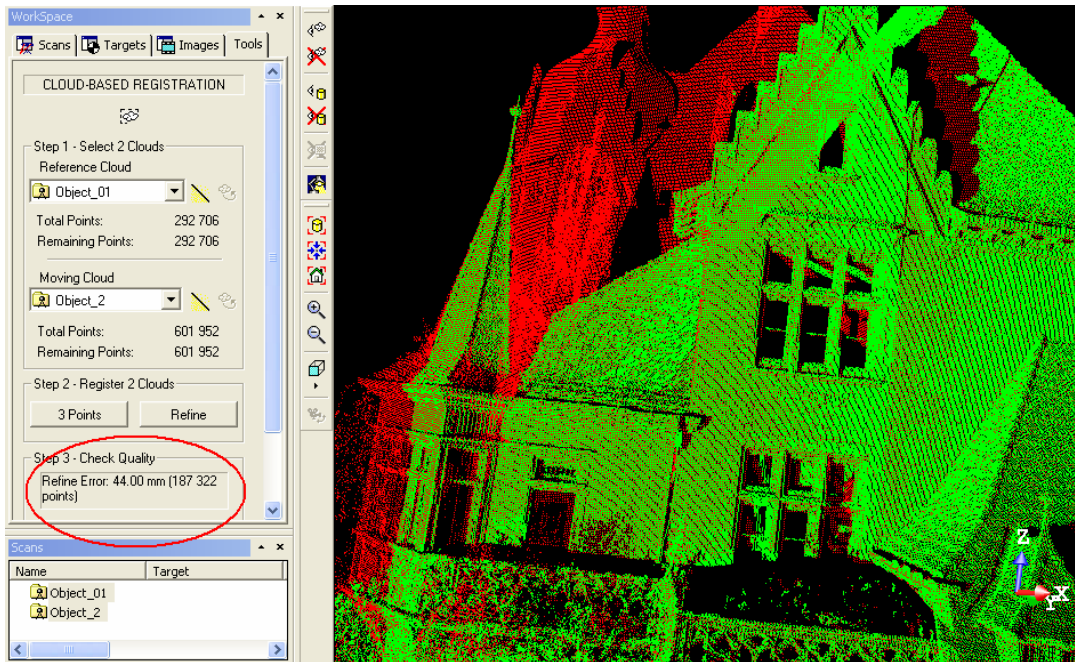


Figure 4.5 La consolidation et son résultat dans Realworks®

L'application de la méthode ICL sous les deux formes sur les 7 droites appariées (5 degré de liberté) nous amène aux résultats suivants :

méthode ICL forme ICP			
rotation (1.0e-006 gon)	ecart type (1.0e-07 gon)	translation (mm)	écart type (mm)
-0.031	0.74	2.1	0.26
-0.024	0.8	0.4	0.45
0.162	0.37	2.7	0.16
erreur moyenne quadratique sur l'unité de poids : 0.51e-07		erreur moyenne quadratique sur l'unité de poids : 0.498 mm	
méthode ICL forme alternative			
rotation (1.0e-007 gon)	écart type (1.0e-08 gon)	translation (mm)	écart type (mm)
-0.03	0.07	-10.07	2.46
-0.014	0.08	-1.62	2.46
0.715	0.29	-12.99	1.56
erreur moyenne quadratique sur l'unité de poids: 0.15-08		erreur moyenne quadratique sur l'unité de poids : 3.81 mm	

Table 4.2 les résultats de la méthode ICL (nuage modèle: R14, nuage donnée I3)

Bien que les chiffres mentionnés dans la table 4.2 soient généralement infimes, nous pouvons constater que le calcul « ICL forme ICP » tend à résoudre le problème de

consolidation par la rotation, tandis que celui que nous intitulons « ICL forme alternative » tend à utiliser la translation. Cette constatation concorde avec les hypothèses de base faites lors du développement de chaque méthode (nous avons proposé que les angles soient minimales pour la forme alternative). Il est très facile à démontrer qu'une petite rotation autour du repère d'origine lointaine (origine de système Lambert dans notre cas) est équivalente à une translation locale, où autrement dit, un petit arc de cercle de très grand rayon peut être approximé à un segment droit.

Encore une fois, nous rappelons que la faiblesse des résultats (les valeurs sont proches de zéro) s'explique par le fait que les deux nuages ont été acquis aux bonnes coordonnées (nuages de points géoréférencés). Nous pouvons dire que l'opération de consolidation (ou bien la superposition) fournit une précision centimétrique.

4.2 Remblai ferroviaire

Ce projet a été effectué pour la SNCF dans le cadre de la surveillance périodique des déformations d'un remblai ferroviaire le long d'une voie ferrée portant un rail de train censée être la raison majeure de ces déformations. La consolidation entre deux nuages acquis quasi simultanément (comme dans notre cas) est une application ordinaire qui aspire à compléter la géométrie de l'objet et à combler le problème de l'occlusion automatique (voir la problématique). En revanche une consolidation entre deux scans effectués en deux temps relativement éloignés nous amènerait à détecter les déformations possibles.

Le chantier fait 200 m de longueur et 32 m de hauteur, l'angle de la ligne d'inclinaison principale avec le plan principal est égal à 35° ce qui est équivalent à une pente de $1:\sqrt{2}$. Les conditions de sécurité nous ont obligé à stationner en bas du remblai et non à côté des rails, c'est pourquoi nous percevons un décroissement de résolution dans les parties hautes de la digue. Quatre stations ont été effectuées pour pouvoir recouvrir la totalité de l'objet. Nous prenons ici les stations 3 et 4 comme échantillon pour tester nos processus suggérés pour la consolidation.

Nous pouvons constater d'une simple prévisualisation du chantier que les entités linéaires sont regroupées dans trois secteurs distincts ; les escaliers, les deux blocs bétons et la clôture au-dessus de la digue. Au-delà, tout le chantier peut être représenté par un seul plan. Nous avons isolé toutes ces parties à l'aide de RealWorks® avant de commencer le traitement.

L'insuffisance des entités linéaires s'accumule du fait que la clôture parallèle aux rails est toujours la plus éloignée du scanner, où la résolution sera très faible ; il est donc inutile d'y chercher les droites. En conséquence les parties utilisables pour l'extraction des lignes sont les blocs bétons et les escaliers. Nous étions conscients de rescanner ces parties en adoptant une résolution supérieure à celle du reste de l'objet.

En ce qui concerne les blocs, ils ont été scannés avec une résolution de 1mm à 100 m, leur distance moyenne à la station 3 vaut environ 57m, et 61m à la station 4 ; leurs arêtes sont évidemment des arêtes de « pli ». Une simple application de la méthode incrémentale ou de RANSAC (seuil distance =20 mm, seuil nombre = 60 points) peut nous amener aux résultats attendus.

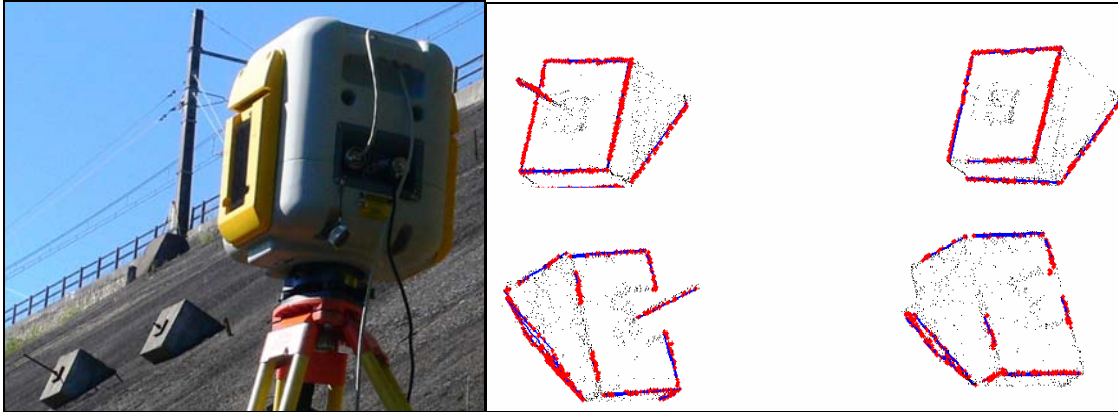
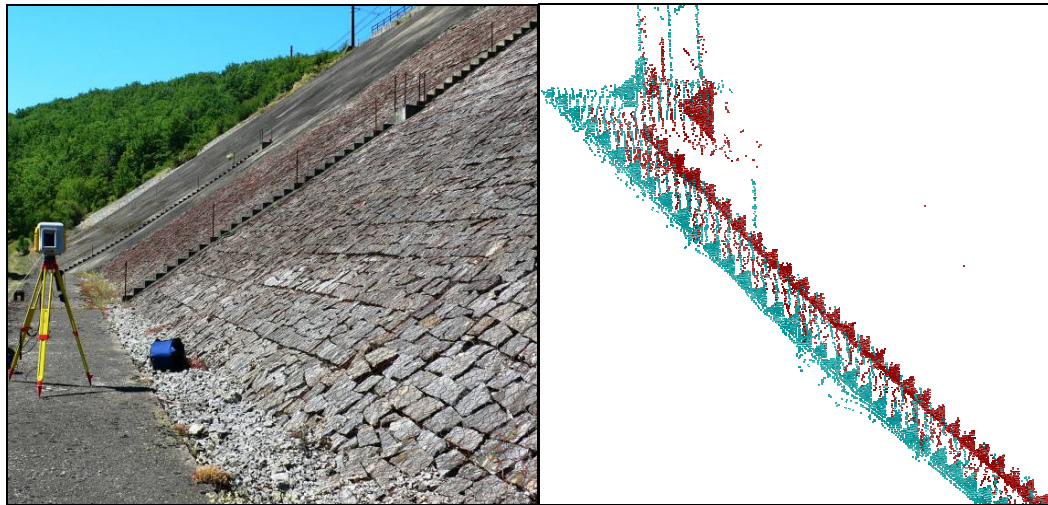


Figure 4.6 les points potentiels et les arêtes extraites sur les deux blocs bétons

Les escaliers représentent un cas spécial qui pose un autre genre de modélisation. Bien que qu'ayant veillé à scanner les escaliers avec une résolution élevée, la forme finale du scan n'était pas celle souhaitée. L'angle étroit du scan impose que le coté acquiert la plupart des points laser tandis que les facettes verticales principales des marches (les contremarches) n'acquièrent que quelques points et celles horizontales ne reçoivent aucun point. La figure 4.7a montre que dans la région du recouvrement entre deux scans différents, les surfaces scannées de haute résolution ne sont pas identiques, et même quand il s'agit de la même surface, la résolution n'est pas suffisante pour la modélisation (10-12cm entre deux lignes successives du scan).



(a)

(b)

Figure 4.7 Deux scans effectués (a) pour le même escalier (b) à partir de deux stations différentes.

Nous devons rappeler que les arêtes ici sont de type saut qui sont dues à l'incapacité du scan et non à la géométrie de l'objet lui-même. Ces arêtes (dans notre cas spécial) ne peuvent pas être modélisées en appliquant notre méthode traditionnelle car les points potentiels sont trop peu nombreux pour créer des lignes.

Face à cette situation, nous avons décidé de nous aider des côtés où le scan est suffisamment dense pour achever plusieurs types de traitement. L'idée de base est de modéliser la surface latérale de chaque marche, puis de tracer les droites perpendiculaires à ces plans qui passent par certains points. Or une droite dans l'espace peut être déterminée par un point connu et un vecteur de direction. La technique suggérée fournit la direction de la droite à partir du plan, tandis que le point connu est celui où l'angle est presque droit.

Si le problème est de chercher le plan qui approche le mieux les facettes latérales, nous ne trouvons qu'un seul plan. L'intérêt est non pas seulement de trouver ce plan, mais aussi les bords de chaque marche pour pouvoir placer les droites perpendiculaires plus tard. Nous pouvons noter facilement dans notre cas que les plans à extraire sont verticaux et que les bords des marches requis sont horizontaux.

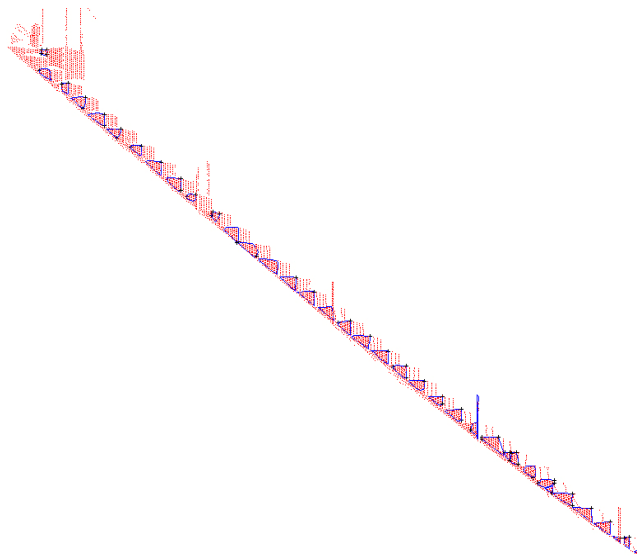


Figure 4.8 l'extraction des plans latéraux délimités des marches.

La méthode appliquée est un compromis entre RANSAC et la méthode d'agrandissement des régions « region growing ». Selon RANSAC nous prenons un échantillon de deux points pour placer un plan (non trois car les plans sont déjà supposés être verticaux). Quand nous arrivons à un consensus majeur de points qui approchent 'bien' le plan échantillon, nous adaptons ce plan. La méthode d'agrandissement des régions est une sorte de segmentation qui sépare les plans délimités ; nous prenons un point quelconque (soit en ordre incrémental ou aléatoire), puis nous testons les points situés au voisinage en considérant leur nombre et leur distance à part. Si le test est positif, nous regroupons tous les points déjà mentionnés en tant que région à agrandir en trouvant son enveloppe convexe ; chaque point de cette enveloppe sera un point de test pour les étapes prochaines. L'algorithme cesse d'itérer quand les conditions d'agrandissement ne sont plus valables.

Nous avons utilisé le même seuil de distance de RANSAC pour la méthode d'agrandissement des régions (2 cm). Une vingtaine de points étaient le nombre approprié

pour admettre un plan délimité, la présence de 2 points (ou plus) proches des limites d'une région garantit son agrandissement.

Après avoir fusionné les plans ayant une intersection entre eux et filtré les plans ayant des paramètres incohérents, l'étape suivante sera de trouver les points de l'enveloppe convexe où l'angle est presque droit. Cette détermination n'est pas forcément idéale (en raison de l'érosion des arêtes et de la fausse détection des angles droits). La seule solution de cette problématique serait donc de dessiner toutes les droites perpendiculaires aux plans latéraux et qui passent en même temps par un point de leur périphérie où l'angle est droit, puis de compter le nombre de points proches de ces droites. Si le nombre des points concernés est supérieur à un certain seuil (la moitié du nombre de points déjà utilisés par exemple), un calcul de ligne horizontale se fait à l'aide de ce groupe de points. La droite attendue est une droite perpendiculaire au plan latéral vertical, elle passe également par le point du groupe mentionné ayant la coordonnée z maximale. Pour pouvoir tracer cette ligne nous choisissons le point du bord coïncidant avec un angle droit et le point du groupe mentionné le plus éloigné.

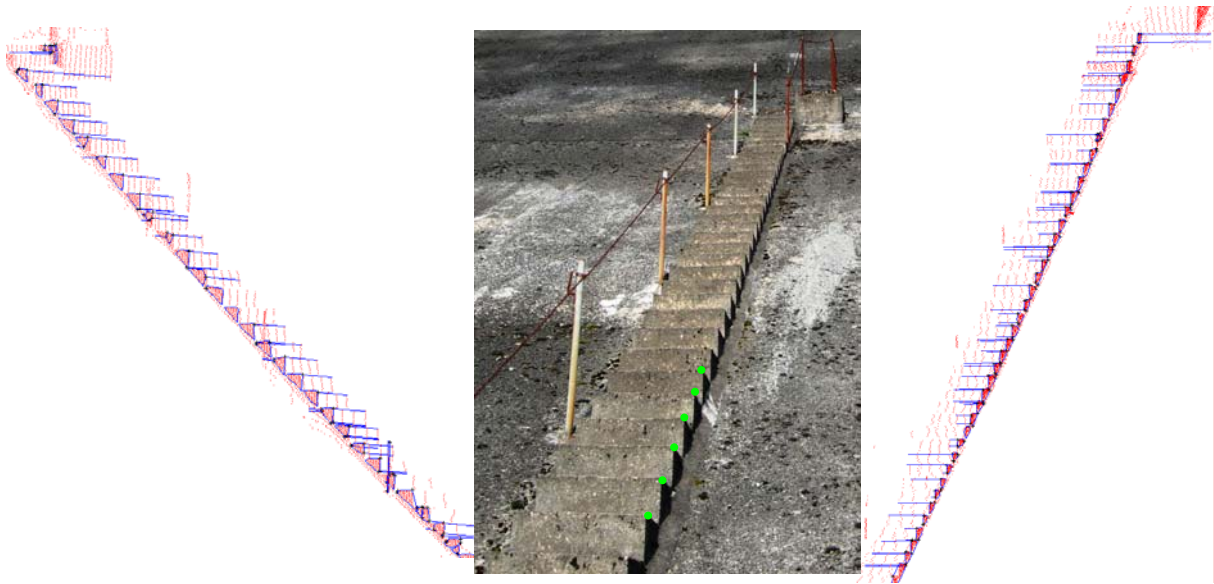


Figure 4.9 les arêtes des marches extraites des deux scans.

Nous verrons alors clairement que chaque arête dans ce cas spécial sera identifiée par :

- Un vecteur de direction perpendiculaire à la direction de la facette latérale de la marche (leur produit vectoriel vaut zéro).
- Le point de périphérie de la facette latérale pour lequel l'angle est égal à zéro.

Ce type de modélisation adopte le minimum d'informations pour trouver ses paramètres, nous ne pouvons donc ni calculer l'erreur moyenne quadratique, ni les résidus. Il serait d'ailleurs inutile d'appliquer les moindres carrés sur les cinq ou six points proches pour retrouver l'équation de l'arête car ils ne représentent pas un échantillon suffisant statistiquement. Cependant il est possible de considérer l'erreur moyenne quadratique correspondant à la modélisation des plans des parties latérales comme un indice de modélisation des arêtes.

Il est clair que les arêtes des escaliers ne seront jamais suffisantes pour effectuer la consolidation, car elles sont quasiment parallèles, ce qui pourrait avoir comme effet le glissement d'un nuage sur l'autre dans la direction horizontale. Ce sont seulement les arêtes des blocs bétons déjà extraites qui permettent de trouver une solution au problème. Bien que nous disposions de 16 arêtes des marches appariées, elles ne fournissent quasiment qu'un seul jeu d'information pour la consolidation. Cependant l'utilisation de ce nombre d'arêtes renforce son rôle (donne un poids plus important) au sein de la consolidation.

La consolidation (dépendant des points) effectuée par le logiciel Realworks® (figure 4.10) est encore moins précise que les opérations préalables (le réseau de points topographiques mesuré et contrôlé par le département de topographie de SNCF). Ceci laisse penser encore une fois que la consolidation n'améliore plus une superposition faite soigneusement. Bien que les échantillons étudiés lors de l'expérimentation soient très peu nombreux pour en déduire des règles générales, nous avons tendance à dire que la consolidation en utilisant les points de deux nuages s'arrête à une précision égale à deux fois la résolution du scan.

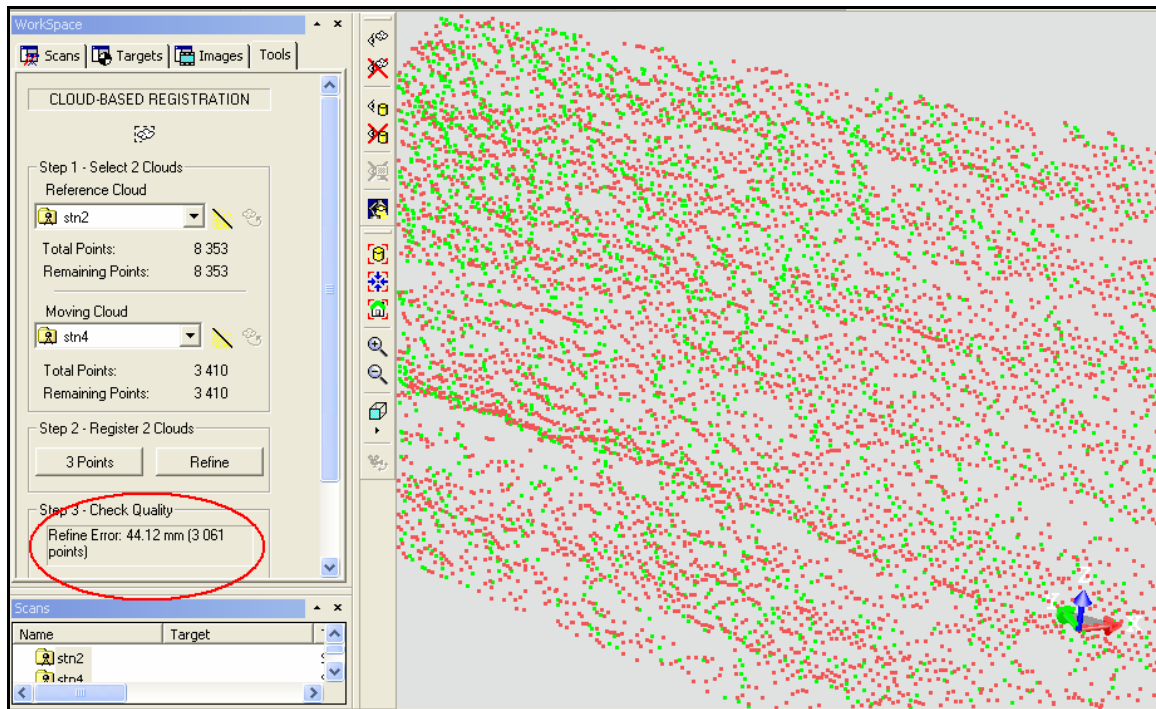


Figure 4.10 l'erreur moyenne quadratique (44 mm) liée à l'affinage de la superposition par la consolidation

L'application de la méthode ICL suggérée sur les deux nuages (n°4 comme modèle et n°2 comme scène) engendre les résultats suivants :

méthode ICL forme ICP			
rotation (1.0e-006 gon)	écart type (1.0e-07 gon)	translation (mm)	écart type (mm)
0.80	0.74	-1.16	0.25
-0.41	0.8	0.62	0.13
0.74	0.37	1.07	0.18
erreur moyenne quadratique réduite l'unité de poids 0.632e-08		erreur moyenne quadratique réduite l'unité de poids : 0.498 mm	
méthode ICL forme alternative			
rotation (1.0e-007 gon)	écart type (1.0e-08 gon)	translation (mm)	écart type (mm)
0.19	0.73	-1.67	0.6
-0.15	0.106	-9.84	1.5
0.11	0.114	-9.92	0.19
erreur moyenne quadratique réduite l'unité de poids 0.78e-08		erreur moyenne quadratique réduite à l'unité de poids : 2.64 mm	

Table 4.3 les composants de la transformation rigide entre les nuages: 3 et 4

L'application de la rotation de la méthode ICL forme ICP sur le point d'origine du scan (comme échantillon du test) engendre une translation valant (3.4, -6.2, -7.1 mm) respectivement dans les directions x, y, z . Si nous rajoutons ces valeurs aux translations calculées nous aurions (2.24, -5.58, -6.03mm). Nous constatons que les deux méthodes présentent une différence de (3-4 mm).

4.3 INSA de Strasbourg

Établissement bâti à l'époque de la reconstruction (après guerre) en 1959, avec une trame « éducation nationale », le bâtiment de l'INSA de Strasbourg représente une architecture simple qui comporte un seul bâtiment principal et plusieurs sections rattachées (section technique, bibliothèque, etc...). Le scan de l'INSA a été effectué en des temps variés lors des séances de TP d'étudiants en topographie de l'INSA.

Nous utilisons deux nuages : le premier (station1) est acquis depuis la station près de l'entrée, avec une résolution de 30 mm à 40 m qui est égale à la distance moyenne à l'objet, ce qui signifie que la résolution du scan est approximativement égale à 30 mm. Le deuxième scan est effectué depuis la station2 de résolution linéaire de 50mm à 70 m .



Figure 4.11 les deux origines des scans issus de la consolidation

Il s'avère que ce bâtiment est très riche en entités linéaires, il représente une application idéale pour toutes hypothèses émises lors du développement de la méthode ICL. Le premier nuage qui comporte 882269 points est réduit à 37380 points après avoir extrait les points potentiels, le deuxième nuage est passé de 708404 à 19507 points.

Les deux nuages partagent la plupart des caractéristiques, ce qui nous a permis d'utiliser les mêmes seuils pour extraire les droites. Une combinaison de RANSAC (50 points, 50mm) et de la méthode incrémentale (20 points, 50mm) nous ramène à extraire 121 droites du premier nuage et 82 du deuxième.

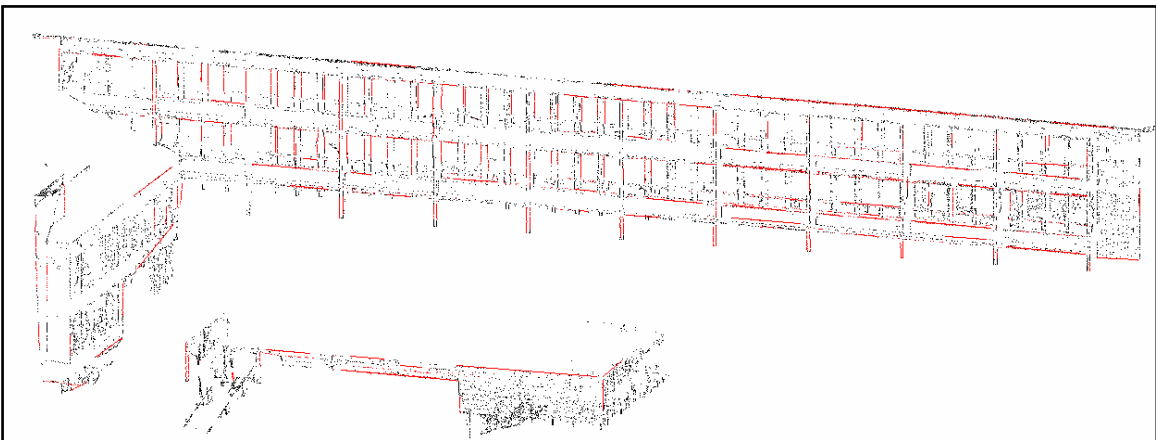


Figure 4.12 les points potentiels et les droites extraites du nuage 2

Un appariement des droites en utilisant le seuil de 40mm produit 34 couples qui participeront au calcul matriciel lors de l'application de la méthode ICL (32 degrés de liberté). La table 4.3 montre les translations et les rotations issues d'ICL en considérant le nuage 1 comme modèle et le nuage 2 comme données :

méthode ICL forme ICP			
rotation (1.0e-007 gon)	écart type (1.0e-08 gon)	translation (mm)	écart type (mm)
-0.35	0.15	-2.5	0.25
0.14	0.28	0.72	0.18
0.26	0.48	-4.8	0.11
erreur moyenne quadratique réduite à l'unité de poids 0.29e-08		erreur moyenne quadratique réduite à l'unité de poids : 0.38 mm	
méthode ICL forme alternative			
rotation (1.0e-007 gon)	écart type (1.0e-08 gon)	translation (mm)	écart type (mm)
-0.20	0.21	-3.42	0.3
0.047	0.02	-1.91	0.3
0.028	0.32	-4.53	0.01
erreur moyenne quadratique réduite à l'unité de poids 0.27e-08		erreur moyenne quadratique réduite à l'unité de poids : 0.98 mm	

Table 4.4 les composants de la transformation rigide entre les nuages: 1 et 2

La première constatation est que les deux techniques montrent une approche importante en ce qui concerne les translations. Les rotations aussi sont beaucoup plus faibles que les deux exemples déjà traités. Nous pouvons lier ces constatations à la croissance importante dans le nombre des droites appariées. En fait ce cas présente une application idéale pour la majorité des hypothèses mises en place lors du développement de la méthode ICL.

Prenons l'origine du premier nuage en tant qu'échantillon pour pouvoir traduire l'effet des rotations motionnées ci-dessus. Nous trouverons que les rotations de la première technique d'ICL donnent une translation de $(0.46, -4.10, 2.92 \text{ mm})$, tandis que les rotations de la deuxième technique donnent : $(0.05, -0.45, 1.113 \text{ mm})$. Quand nous rajoutons ces valeurs aux translations déjà calculées, nous notons que les résultats des deux techniques se rapprochent de plus en plus.

Quelque soit la différence entre les deux méthodes, les résultats de la consolidation ne dépassent pas une précision centimétrique, ce qui laisse croire que les travaux topographiques effectués avant de lancer le scan sont de même précision, et que l'amélioration apportée par la consolidation peut être considérée comme négligeable si la préparation topographique a correctement été effectuée. Nous pouvons uniquement dire que la superposition a été validée par la consolidation.

5 : Conclusion et perspectives

5.1 Conclusion

Nous pouvons remarquer que notre sujet est constitué de deux axes principaux : d'une part détecter les droites, puis, d'autre part, s'en servir pour achever la consolidation entre deux nuages de points. Les conclusions de notre recherche vont ainsi porter sur ces deux axes.

En ce qui concerne l'extraction automatique des droites, nous avons trouvé qu'il n'y a pas de solution générale qui pourrait englober tous les cas. Nous sommes encore loin de percevoir une méthode entièrement automatisée pour dessiner toutes les arêtes dans un nuage laser. Néanmoins la détermination précise de « quelques » droites peut nous permettre de répondre à notre objectif de départ (la consolidation).

Nous avons constaté que pour chaque algorithme, il existe des points forts, mais également des points faibles (table 5.1). Cependant nous ne pouvons pas privilégier un algorithme pour tous les cas ; c'est la raison pour laquelle nous avons développé une interface interactive avec l'utilisateur de manière à ce qu'il puisse choisir la méthode qui lui convient, tester les résultats et adopter les meilleurs.

L'algorithme	Incrémentale	RANSAC	Transformée de Hough
Vitesse	-	--	+
Fonctionnalité 3D/2D	+	+	-
Probabilisme	+	--	-
Nombre de paramètres à fournir	++	+	--
Besoin de fusionner les segments	--	+	+
Précision	+	+	-
Nombre de droites extraites	++	+	-

Table 5.1 : les avantages et les inconvénients des trois méthodes proposées pour extraire les droites

Nous avons trouvé que la méthode RANSAC donne des résultats plus acceptables quand il s'agit de droites longues (des centaines de points), tandis que la méthode incrémentale serait préférée quand il s'agit d'arêtes courtes, ou bien quand le scannage est effectué à une haute résolution. Il est intéressant de noter qu'une méthode dépendant de l'ordre des points (la méthode incrémentale) donnerait des résultats optimaux pour les lignes suivant le même ordre. Par exemple pour le scanner GX de Trimble®, la ligne de

scan est verticale ; la méthode incrémentale serait l'idéale pour les lignes verticales si nous utilisons les points acquis comme tels. Cependant un résultat plus général est atteint si le tri est effectué par rapport à la distance à l'appareil. La transformée de Hough apporte des résultats optimaux quand les lignes ne sont pas entachées de bruit.

Dans la majorité des cas étudiés, une utilisation séquentielle de la méthode incrémentale puis de RANSAC est largement suffisante pour détecter une majorité de droites.

Le choix des seuils est encore le facteur le plus influant pour la détection des droites. Les deux seuils à introduire (la distance et le nombre de points) sont mutuellement interactifs. Il est possible d'accepter la modélisation d'un segment avec un nombre faible de points à condition que la précision soit très élevée ; en revanche nous tolérons une précision moins élevée si le nombre de points est remarquablement important. Il est utile de déduire tous les seuils intermédiaires à partir de deux seuils : la distance et le nombre des points. Nous recommandons toutefois d'exécuter les processus d'extraction de lignes en utilisant plusieurs seuils pour obtenir le nombre maximal de droites car l'appariement des droites suivant en filtrera une grande part.

La consolidation est une opération de base qui possède des résultats difficiles à interpréter pour un utilisateur ordinaire. Cependant, l'influence directe de cette opération sur la précision de modélisation (CAO par exemple) est non négligeable. Cette précision est le critère le plus important (voire le seul) pour juger du succès ou de l'échec de la consolidation.

Il s'avère que le scanner à disposition pendant toutes les étapes du projet est un scanner topographique. Etant donné que ce scanner utilise des opérations préalables (ou simultanées) pour obtenir des nuages de points superposés, la problématique s'est transformée en une sorte de validation des opérations de terrain, or il a été démontré que rien n'empêche la méthodologie d'être applicable partout. Nous avons donc prouvé qu'un bon stationnement peut se substituer au processus de consolidation (les rotations et les translations étant très faibles). En règle générale, il est convenable que la précision finale de la consolidation ne dépasse pas la résolution linéaire du scan.

ICL est une méthode très sensible à la détection des droites, ceci est peut-être son inconvénient principal. Si la direction des droites est mal détectée, elle engendrera des rotations importantes de nuage scène lors de l'application d'ICL (la rotation résulte seulement des directions des droites). Puisque le calcul de la translation dépend de la rotation, ce calcul sera totalement erroné aussi.

5.2 Perspectives

Nous avons rencontré plusieurs difficultés au sein de notre travail, nous avons pu en surmonter quelques unes, mais il reste encore quelques obstacles qui incommode les démarches automatiques de la méthode proposée. Ces obstacles nous inspirent les perspectives suivantes:

1. L'extraction des droites en tant que contours de plans ; c'est-à-dire déterminer les plans moyens délimités qui s'adaptent au mieux aux points Laser et ensuite utiliser leurs bords comme des lignes « candidates ».
2. L'utilisation des autres entités géométriques (courbes de plusieurs degrés, arcs de cercle, plans,...etc.) pour effectuer la consolidation, celle-ci pouvant, par là même,

-
- être utilisées comme primitives initiales pour la modélisation géométrique à réaliser.
3. L'évolution de l'étude de l'extraction des points potentiels (les points du changement de normale), puisque toutes les étapes suivantes en dépendent.
 4. Le développement des méthodes fournissant des coordonnées approchées (quand le scanner utilisé n'est pas un scanner topographique). La méthode la plus répandue est de laisser l'utilisateur choisir trois points homologues de chaque nuage. Dans notre cas, il est possible d'apparier les droites ayant des caractères géométriques locaux semblables (voir le hachage géométrique). Ce processus pourrait diminuer le nombre d'itérations nécessaires pour que la méthode ICL se stabilise.
 5. Le développement d'une méthode qui réutilise les droites non appariées pour effectuer le recalage. Nous suggérons ici de s'aider de la connaissance architecturale (conditions de perpendicularité, parallélisme, intersection par un angle connu, etc....).

6 Annexe

6.1 Annexe A : formules du calcul utilisées

6.1.1 Droite perpendiculaire à un plan

Soit un plan P d'équation : $ax + by + cz = d$, $O(x_0, y_0, z_0)$ un point de ce plan ; l'équation de la ligne l perpendiculaire à P et qui passe par O est donnée par deux formes ; La formule symétrique :

$$t = \frac{x - x_0}{a} = \frac{y - y_0}{b} = \frac{z - z_0}{c} \quad (\text{A.1})$$

ou bien la forme paramétrique :

$$\begin{aligned} x &= x_0 + at \\ y &= y_0 + bt \\ z &= z_0 + ct \end{aligned} \quad (\text{A.2})$$

6.1.2 Distance entre un point et une droite.

La distance d'un point à une droite est la longueur du plus court segment joignant ce point à un point de la droite. Soit une droite passant par un point (x_1, y_1, z_1) et dont le vecteur de la direction est : (a, b, c)

$$\begin{aligned} x &= x_1 + at \\ y &= y_1 + bt \\ z &= z_1 + ct \end{aligned} \quad (\text{A.3})$$

La distance d'un point (x_0, y_0, z_0) à cette droite est égale à :

$$dis = \frac{\|(a, b, c) \otimes ((x_0, y_0, z_0) - (x_1, y_1, z_1))\|}{\|a, b, c\|} \quad (\text{A.4})$$

où \otimes désigne le produit vectoriel de deux vecteurs.

Si la droite est donnée par deux points $(x_1, y_1, z_1), (x_2, y_2, z_2)$, la distance sera donnée par la formule :

$$dis = \frac{\|((x_1, y_1, z_1) - (x_2, y_2, z_2)) \otimes ((x_0, y_0, z_0) - (x_1, y_1, z_1))\|}{\|(x_1, y_1, z_1) - (x_2, y_2, z_2)\|} \quad (\text{A.5})$$

6.1.3 Distance entre deux lignes

Soient deux lignes (l_1, l_2) données par leur forme paramétrique :

$$\begin{aligned}
 l_1 : \quad & x = x_{01} + sv_{x1} \\
 & y = y_{01} + sv_{y1} \\
 & z = z_{01} + sv_{z1} \\
 l_2 : \quad & x = x_{02} + sv_{x2} \\
 & y = y_{02} + sv_{y2} \\
 & z = z_{02} + sv_{z2}
 \end{aligned} \tag{A.6}$$

Nous allons définir les termes suivants :

$$\begin{aligned}
 p_0 &= [x_{01}, y_{01}, z_{01}] & u &= [v_{x1}, v_{y1}, v_{z1}] \\
 q_0 &= [x_{02}, y_{02}, z_{02}] & v &= [v_{x2}, v_{y2}, v_{z2}] \\
 w_0 &= p_0 - q_0 & c &= v \cdot v \\
 a &= u \cdot u & d &= u \cdot w_0 \\
 b &= u \cdot v & e &= v \cdot w_0
 \end{aligned} \tag{A.7}$$

dans ce qui a précédé, le produit est un produit scalaire. Nous pouvons finalement citer la formule de la distance :

$$dis = \left\| (p_0 - q_0) + \frac{(b \cdot e - c \cdot d)u - (a \cdot e - b \cdot d)v}{ac - b^2} \right\| \tag{A.8}$$

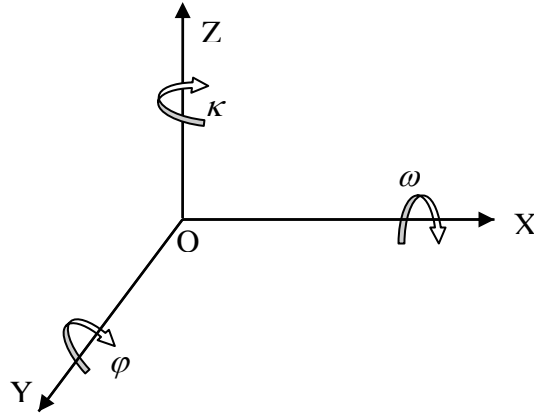
6.1.4 La projection d'un point sur une ligne

Le calcul du projeté orthogonal P' d'un point P sur une droite l_1 ayant le point d'origine $O(x_0, y_0, z_0)$ et le vecteur de direction $V(v_x, v_y, v_z)$, n'est qu'une intersection entre la droite l_1 et une droite l_2 perpendiculaire passant par P . La solution de deux équations de l_1, l_2 nous amène à déterminer la valeur de la variable t intermédiaire correspondant au projeté :

$$t = V \cdot (P - O) / V \cdot V \tag{A.9}$$

où le produit est toujours un produit scalaire. Il suffit de substituer t dans les équations paramétriques de l_1 pour obtenir les coordonnées finales du projeté P' .

6.1.5 Matrice de rotation dans l'espace :



Si l'on les 3 angles d'Euler (ω, φ, κ) pour représenter la rotation autour des 3 axes de coordonnées x, y, z respectivement, la matrice de la rotation sera donnée par la formule :

$$R_{(\omega, \varphi, \kappa)} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}$$

où

$$\begin{aligned} R_{11} &= \cos \varphi \cos \kappa \\ R_{12} &= \sin \omega \sin \varphi \cos \kappa + \cos \omega \sin \kappa \\ R_{13} &= -\cos \omega \sin \varphi \cos \kappa + \sin \omega \sin \kappa \\ R_{21} &= -\cos \varphi \sin \kappa \\ R_{22} &= -\sin \omega \sin \varphi \sin \kappa + \cos \omega \cos \kappa \\ R_{23} &= \cos \omega \sin \varphi \sin \kappa + \sin \omega \cos \kappa \\ R_{31} &= \sin \varphi \\ R_{32} &= -\sin \omega \cos \varphi \\ R_{33} &= \cos \omega \cos \varphi \end{aligned} \tag{A.10}$$

6.1.6 Linéarisation des équations de la conjugaison des droites

6.1.6.1 Translation (forme ICP adaptée)

Revenons aux équations 4.27 rédigées pour une seule paire de lignes :

$$\begin{aligned} p' + t_1 v' &= R(p + t_1 v) + T \\ p' + t_2 v' &= R(p + t_2 v) + T \end{aligned} \tag{A.11}$$

Nous pouvons développer ces équations de la manière suivante :

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} + t'_1 \begin{pmatrix} v'_x \\ v'_y \\ v'_z \end{pmatrix} = R \begin{pmatrix} x \\ y \\ z \end{pmatrix} + t_1 R \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} + \begin{pmatrix} T_x \\ T_y \\ T_z \end{pmatrix} \quad (\text{A.12})$$

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} + t'_2 \begin{pmatrix} v'_x \\ v'_y \\ v'_z \end{pmatrix} = R \begin{pmatrix} x \\ y \\ z \end{pmatrix} + t_2 R \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} + \begin{pmatrix} T_x \\ T_y \\ T_z \end{pmatrix}$$

Nous allons appeler $\begin{pmatrix} x_r \\ y_r \\ z_r \end{pmatrix} = R \begin{pmatrix} x \\ y \\ z \end{pmatrix}$ et aussi $\begin{pmatrix} v_{xr} \\ v_{yr} \\ v_{zr} \end{pmatrix} = R \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix}$ (A.13)

Le système d'équations à disposition s'écrit :

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} - \begin{pmatrix} x_r \\ y_r \\ z_r \end{pmatrix} = -t'_1 \begin{pmatrix} v'_x \\ v'_y \\ v'_z \end{pmatrix} + t_1 \begin{pmatrix} v_{xr} \\ v_{yr} \\ v_{zr} \end{pmatrix} + \begin{pmatrix} T_x \\ T_y \\ T_z \end{pmatrix} \quad (\text{A.14})$$

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} - \begin{pmatrix} x_r \\ y_r \\ z_r \end{pmatrix} = -t'_2 \begin{pmatrix} v'_x \\ v'_y \\ v'_z \end{pmatrix} + t_2 \begin{pmatrix} v_{xr} \\ v_{yr} \\ v_{zr} \end{pmatrix} + \begin{pmatrix} T_x \\ T_y \\ T_z \end{pmatrix}$$

Et sous la forme matricielle de système linéaire $Ax = b$:

$$\begin{pmatrix} x_r \\ y_r \\ z_r \\ x_r \\ y_r \\ z_r \end{pmatrix}_{6 \times 1} = \begin{pmatrix} v_{xr} & -v'_x & 0 & 0 & 1 & 0 & 0 \\ v_{yr} & -v'_y & 0 & 0 & 0 & 1 & 0 \\ v_{zr} & -v'_z & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & v_{xr} & -v'_x & 1 & 0 & 0 \\ 0 & 0 & v_{yr} & -v'_y & 0 & 1 & 0 \\ 0 & 0 & v_{zr} & -v'_z & 0 & 0 & 1 \end{pmatrix}_{6 \times 7} \cdot \begin{pmatrix} t_1 \\ t'_1 \\ t_2 \\ t'_2 \\ T_x \\ T_y \\ T_z \end{pmatrix}_{7 \times 1} \quad (\text{A.15})$$

Rappelons que ces équations sont établies pour une seule paire de droites et que nous devons les répéter pour chaque paire, de telle façon que la taille de la matrice A sera $6n \times 7$ et la taille de la matrice b sera $6n \times 1$. La solution pour ce système est :

$$x = (A^T A)^{-1} A^T b \quad (\text{A.16})$$

6.1.6.2 Rotation (forme alternative)

Revenons aux équations 4.30 ; nous pouvons les écrire sous la forme matricielle des moindres carrés qui correspond à la compensation par les observations intermédiaires : $v + B \cdot \Delta = f$ comme suit :

$$\begin{aligned}
 F_1 &= (X_B - X_A) - (Z_B - Z_A) \frac{R_{11}(X_2 - X_1) + R_{12}(Y_2 - Y_1) + R_{13}(Z_2 - Z_1)}{R_{31}(X_2 - X_1) + R_{32}(Y_2 - Y_1) + R_{33}(Z_2 - Z_1)} \\
 F_2 &= (Y_B - Y_A) - (Z_B - Z_A) \frac{R_{21}(X_2 - X_1) + R_{22}(Y_2 - Y_1) + R_{23}(Z_2 - Z_1)}{R_{31}(X_2 - X_1) + R_{32}(Y_2 - Y_1) + R_{33}(Z_2 - Z_1)}
 \end{aligned} \tag{A.17}$$

Pour simplifier le calcul, nous allons aborder uniquement le cas spécial où le nuage scène est acquis en temps réel avec ses vraies coordonnées qui ont besoin d'être affinées. Nous faisons donc l'hypothèse que les angles de rotation sont faibles ce qui donne la forme suivante à la matrice de rotation :

$$R_{(\omega, \varphi, \kappa)} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} = \begin{bmatrix} 1 & -\kappa & \varphi \\ \kappa & 1 & -\omega \\ -\varphi & \omega & 1 \end{bmatrix} \tag{A.13}^2$$

Ainsi nous pouvons substituer les valeurs de R_{ij} dans les équations 4.30 :

$$\begin{aligned}
 F_1 &= (X_B - X_A) - (Z_B - Z_A) \frac{(X_2 - X_1) - \kappa(Y_2 - Y_1) + \varphi(Z_2 - Z_1)}{-\varphi(X_2 - X_1) + \omega(Y_2 - Y_1) + (Z_2 - Z_1)} \\
 F_2 &= (Y_B - Y_A) - (Z_B - Z_A) \frac{\kappa(X_2 - X_1) + (Y_2 - Y_1) - \omega(Z_2 - Z_1)}{-\varphi(X_2 - X_1) + \omega(Y_2 - Y_1) + (Z_2 - Z_1)}
 \end{aligned} \tag{A.18}$$

$$\begin{aligned}
 F_1 &= \Delta X - \Delta Z \frac{r}{t} \\
 F_2 &= \Delta Y - \Delta Z \frac{s}{t}
 \end{aligned} \tag{A.19}$$

La théorie de base dans cette recherche est que chaque nuage est intérieurement stable et que les erreurs systématiques et accidentelles de l'appareil ne sont pas prises en compte. Cette constatation se traduit par le fait que toutes les coordonnées sont des constantes et que seulement les trois rotations ω, φ, κ sont des variables, ou autrement dit :

$$F_1, F_2, r, s, t = f(\omega, \varphi, \kappa)$$

Par conséquent nous pouvons écrire la forme générale d'une dérivée partielle :

$$\begin{aligned}
 \frac{\partial F_1}{\partial p} &= -\frac{\Delta Z}{t} \left[\frac{\partial r}{\partial p} - \frac{r}{t} \frac{\partial t}{\partial p} \right] \\
 \frac{\partial F_2}{\partial p} &= -\frac{\Delta Z}{t} \left[\frac{\partial s}{\partial p} - \frac{s}{t} \frac{\partial t}{\partial p} \right]
 \end{aligned} \tag{A.20}$$

où p désigne à la fois une des variables ω, φ, κ , nous allons citer tous les dérivées nécessaires pour substituer dans la formule A.20 :

$$\begin{aligned}
 \frac{\partial r}{\partial \omega} &= 0 & \frac{\partial t}{\partial \omega} &= Y_2 - Y_1 & \frac{\partial s}{\partial \omega} &= -(Z_2 - Z_1) \\
 \frac{\partial r}{\partial \varphi} &= (Z_2 - Z_1) & \frac{\partial t}{\partial \varphi} &= -(X_2 - X_1) & \frac{\partial s}{\partial \varphi} &= 0 \\
 \frac{\partial r}{\partial \kappa} &= -(Y_2 - Y_1) & \frac{\partial t}{\partial \kappa} &= 0 & \frac{\partial s}{\partial \kappa} &= (X_2 - X_1)
 \end{aligned} \tag{A.21}$$

Enfin, nous pouvons construire la formule matricielle : $v + B.\Delta = f$ sachant que pour chaque paire i des lignes appariées :

$$\begin{aligned}
 v_i &= \begin{bmatrix} v_x \\ v_y \end{bmatrix} \quad \text{le vecteur des résidus,} \\
 f_i &= \begin{bmatrix} -F_{2_i}^0 \\ -F_{2_i}^0 \end{bmatrix} \quad \text{les valeurs approchées des vecteurs } F_1, F_2 \\
 B_i &= \begin{bmatrix} \frac{\partial F_1}{\partial \omega} & \frac{\partial F_1}{\partial \varphi} & \frac{\partial F_1}{\partial \kappa} \\ \frac{\partial F_2}{\partial \omega} & \frac{\partial F_2}{\partial \varphi} & \frac{\partial F_2}{\partial \kappa} \end{bmatrix}
 \end{aligned} \tag{A.22}$$

Ainsi, pour n paires de lignes conjuguées le système des équations sera :

$$\begin{bmatrix} f_1 \\ f_2 \\ \cdot \\ \cdot \\ f_n \end{bmatrix}_{1 \times 2n} = \begin{bmatrix} v_1 \\ v_2 \\ \cdot \\ \cdot \\ v_n \end{bmatrix}_{1 \times 2n} + \begin{bmatrix} B_1 \\ B_2 \\ \cdot \\ \cdot \\ B_n \end{bmatrix}_{3 \times 2n} \cdot \begin{bmatrix} \Delta \omega \\ \Delta \varphi \\ \Delta \kappa \end{bmatrix}_{1 \times 3} \tag{A.23}$$

$\Delta \omega, \Delta \varphi, \Delta \kappa$ sont des corrections que nous devons ajouter aux valeurs initiales à chaque itération de la solution suivante :

$$\Delta = (B^T W B)^{-1} (B^T W f) \tag{A.24}$$

où W est la matrice des poids.

6.1.6.3 Translation (forme alternative) :

Nous allons maintenant répéter les mêmes démarches pour les équations 4.33 qui peuvent être écrites comme suit :

$$\begin{aligned}
 F_1 &= (X_T + x_1 - X_A) - (Z_T + z_1 - Z_A) \cdot \frac{(X_T + x_2 - X_A)}{(Z_T + z_2 - Z_A)} \\
 F_2 &= (Y_T + y_1 - Y_A) - (Z_T + z_1 - Z_A) \cdot \frac{(Y_T + y_2 - Y_A)}{(Z_T + z_2 - Z_A)}
 \end{aligned}
 \tag{A.25}$$

les deux équations précédentes possèdent seulement des dérivées partielles par rapport aux inconnues X_T, Y_T, Z_T qui sont :

$$\begin{aligned}
 \frac{\partial F_1}{\partial X_T} &= 1 - \frac{(Z_T + z_1 - Z_A)}{(Z_T + z_2 - Z_A)} & \frac{\partial F_2}{\partial X_T} &= 0 \\
 \frac{\partial F_1}{\partial Y_T} &= 0 & \frac{\partial F_2}{\partial Y_T} &= 1 - \frac{(Z_T + z_1 - Z_A)}{(Z_T + z_2 - Z_A)} \\
 \frac{\partial F_1}{\partial Z_T} &= \frac{(X_T + x_2 - X_A)}{(Z_T + z_2 - Z_A)} \left[\frac{(Z_T + z_1 - Z_A)}{(Z_T + z_2 - Z_A)} - 1 \right] & \frac{\partial F_2}{\partial Z_T} &= \frac{(Y_T + y_2 - Y_A)}{(Z_T + z_2 - Z_A)} \left[\frac{(Z_T + z_1 - Z_A)}{(Z_T + z_2 - Z_A)} - 1 \right]
 \end{aligned}
 \tag{A.26}$$

encore une fois, chaque paire de lignes appariées i produit l'équation matricielle $v_i + B_i \cdot \Delta = f_i$ où :

$$\begin{aligned}
 v_i &= \begin{bmatrix} v_x \\ v_y \end{bmatrix} \quad \text{le vecteur des résidus,} \\
 f_i &= \begin{bmatrix} -F_{2_i}^0 \\ -F_{2_i}^0 \end{bmatrix} \quad \text{les vecteurs } F_1, F_2 \text{ en fonction des valeurs approchées } (X_T^0, Y_T^0, Z_T^0) \text{ (sont}
 \end{aligned}$$

souvent égales à zéro)

$$B_i = \begin{bmatrix} \frac{\partial F_1}{\partial X_T} & \frac{\partial F_1}{\partial Y_T} & \frac{\partial F_1}{\partial Z_T} \\ \frac{\partial F_2}{\partial X_T} & \frac{\partial F_2}{\partial Y_T} & \frac{\partial F_2}{\partial Z_T} \end{bmatrix}
 \tag{A.27}$$

et pour un système de n équations nous aurions la formule générale : $v + B \cdot \Delta = f$ qui a la même solution qu'auparavant.

6.2 Annexe B : interfaces programmées

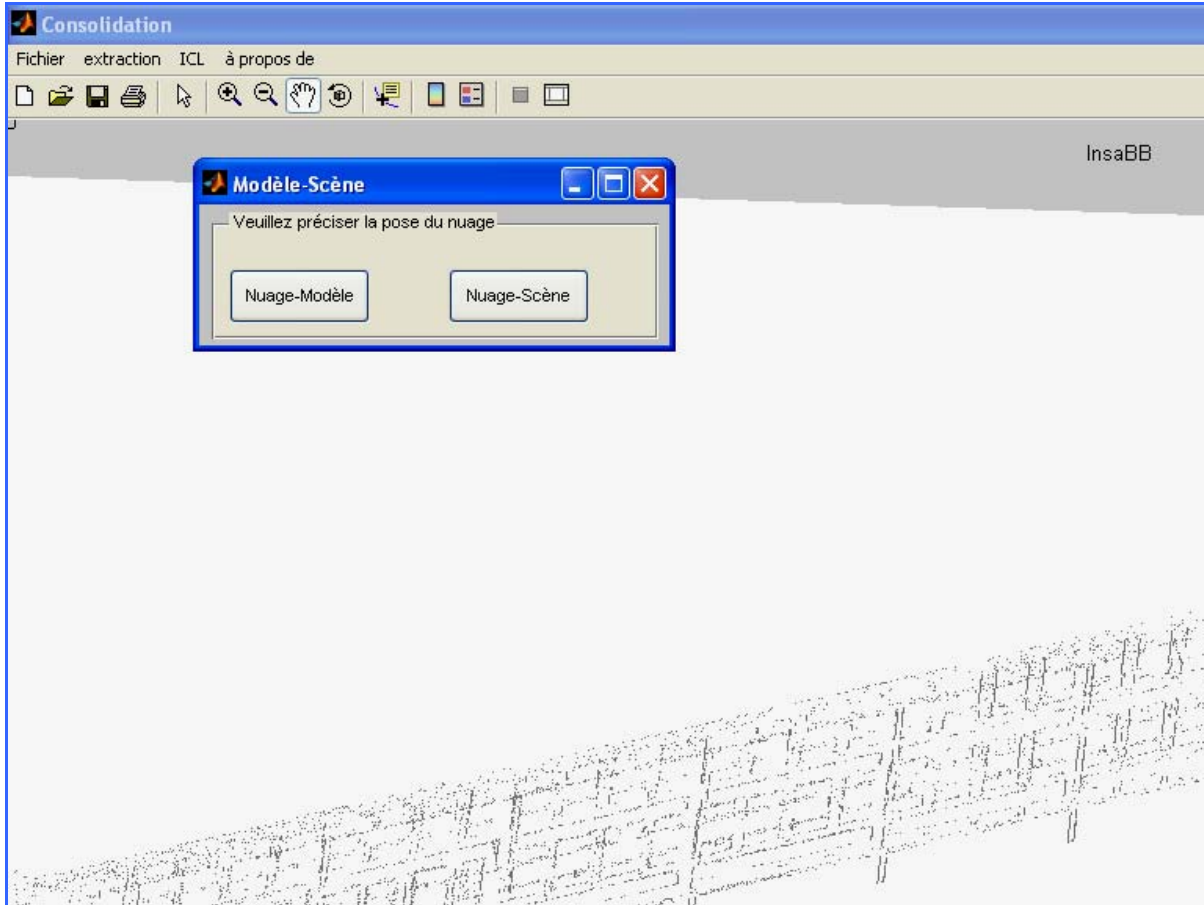


Figure B.1 la sélection de la pose du nuage (modèle, scène) lors de l'importation

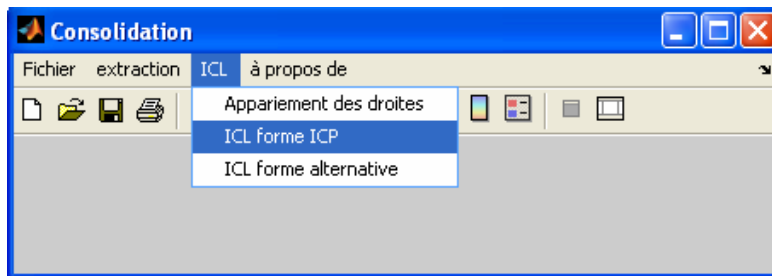


Figure B.2 les menus principaux du programme.

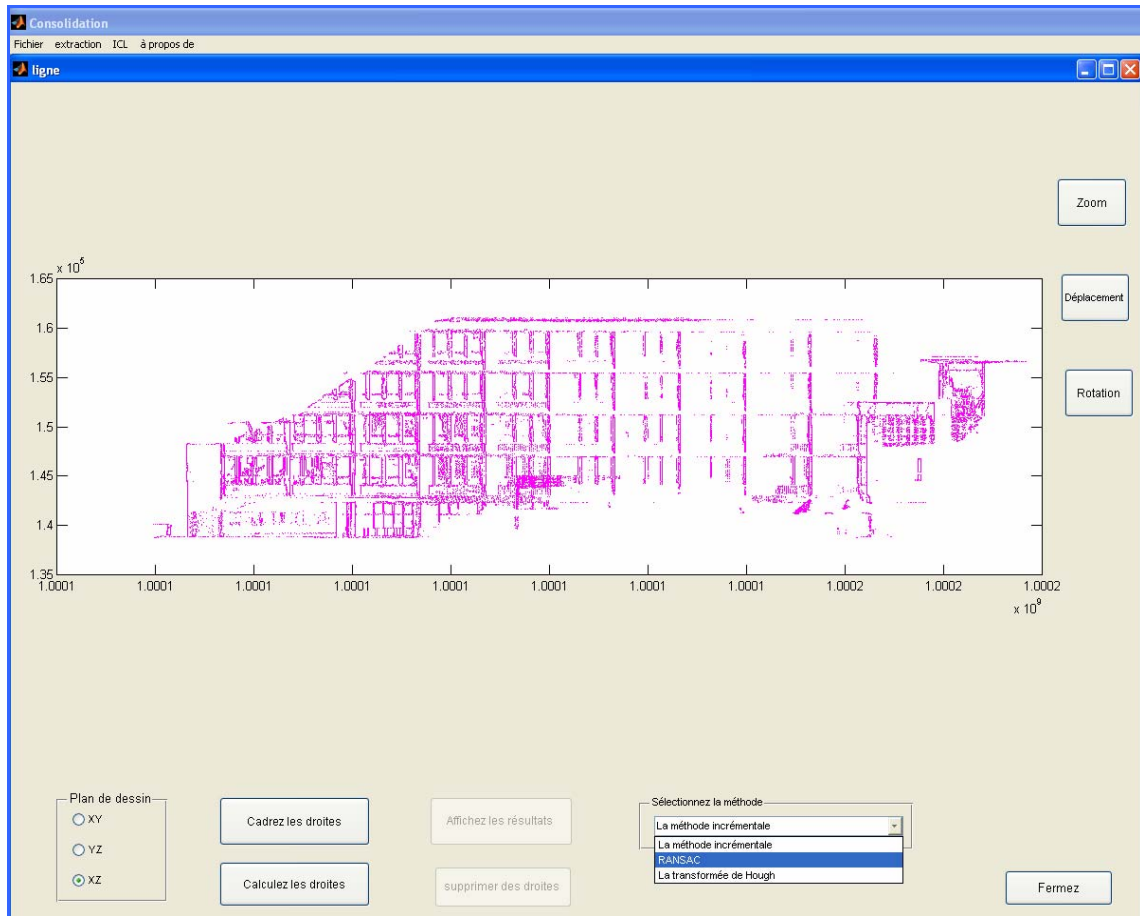


Figure B.3 l'interface interactive de l'extraction des droites

7 Bibliographie

- [AHB87] K. S. Arun, T. S. Huang, and S. D. Blostein. "Least square fitting of two 3-d point sets". IEEE Transactions on Pattern Analysis and Machine Intelligence, 9(5):698 -- 700, 1987.
- [Bar97] G.Barequet;"Using geometric hashing to repair CAD objects",Computational Science and Engineering, IEEE [see also Computing in Science & Engineering],Volume 4, Issue 4, Oct.-Dec. 1997 Page(s):22 - 28
- [Bib03] P.Biber, , 2003. "The normal distributions transform: A new approach to laser scan matching". Technical Report 3, Wilhelm Schickard Institute for Computer Interactive Systems (WSI/GRIS), University of Tübingen.
- [BM92] P.J.Besl, N.D McKay, 1992."A method for registration of 3D shapes". IEEE PAMI, 14(2), 239-256
- [CH98] O.Carmichael, M.Hebert, "Unconstrained registration of large 3D point sets for complex model building" ;Intelligent Robots and Systems, 1998. Proceedings., 1998 IEEE/RSJ International Conference
- [CHC99] C.Chen,Y.Hung, and J.Cheng, "RANSAC-based DARCES: A New Approach to Fast Automatic Registration of Partially Overlapping Range Images" (1999),IEEE Transactions on Pattern Analysis and Machine Intelligence,vol 21 n 11 (1999) p1229-1234
- [CM91] Y. Chen, G. Medioni "Object modelling by registration of multiple range images", Image and Vision Computing Volume 10 , Issue 3 (April 1992)
- [FB81] M. A. Fischler, R. C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Comm. of the ACM, Vol 24, pp 381-395, 1981.
- [Fit01] A. Fitzgibbon. "Robust registration of 2d and 3d point sets". In The British Machine Vision Conference, 2001
- [GP97] Gueziec, A.P.; Pennec, X.; Ayache, N.; " Medical image registration using geometric hashing" Computational Science and Engineering, IEEE [see also Computing in Science & Engineering],Volume 4, Issue 4, Oct.-Dec. 1997 Page(s):29 - 41
- [GRL03] N. Gelfand, S. Rusinkiewicz, and M. Levoy. "Geometrically Stable Sampling for the ICP Algorithm". In Proc. IEEE 3DIM, Canada, 2003.
- [HB94] Hecker, Y.C. Bolle, R.M. "On geometric hashing and the generalized Hough transform" Systems, Man and Cybernetics, IEEE transactions on Volume: 24, Issue: 9 Sep 1994
- [HG05] A. F. Habib, M. S. Ghanma, M. Tait: "Integration of Lidar and Photogrammetry for Close Range Applications" ISPRS Proceeding of Congress, 12-23 July 2004 Istanbul, Turkey
- [Hor87] B. K. P. Horn, "Closed-form solution of absolute orientation using unitquaternions," Journal of the Optical Society of America A, vol. 4, no. 4, pp. 629–642, April 1987.
- [JB99] X. Jiang and H. Bunke, "Edge detection in range images based on scan line approximation," CVIU, vol. 73, no. 2, pp. 183--199, February 1999.
- [JH97] A. Johnson and M. Hebert. "Control of polygonal mesh resolution for 3D computer vision." Technical Report CMURI -TR-96-20, Carnegie Mellon University, Pittsburgh, PA, April 1997
- [Joh97] A. Johnson ; "Spin-Images: A Representation for 3-D Surface Matching", doctoral dissertation, tech. report CMU-RI-TR-97-47, Robotics Institute, Carnegie Mellon University, August, 1997. p17-20

- [Kam01] Kamgar-Parsi, B. "An open problem in matching sets of 3D lines" .Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on 2001 p: I-651- I-656 vol.1
- [KSK96] B. Krebs, P. Sieverding and B. Korn," A fuzzy ICP algorithm for 3d free form object recognition", in Proceedings of the 13th International Conference on Pattern Recognition, Vienna, 1996, vol. A, pp. 539-543.
- [LEF95] A. Lorusso, D.W. Eggert, R.B. Fisher : " A Comparison of Four Algorithms for Estimating 3-D Rigid Transformations" British Machine Vision Conference (1995)
- [LPZ03] LEOPOLDSEDER S., POTTMANN H., ZHAO H.: "The d2-tree: A hierarchical representation of the squared,distance function". In Tech.Rep. 101, Institut of Geometry,Vienna University of Technology (2003).
- [MBO06] A. S. Mian , M. Bennamoun and R. A. Owens;"A Novel Representation and Feature Matching Algorithm for Automatic Pairwise Registration of Range Images", International Journal of Computer Vision ,Volume 66, Number 1 January 2006 p19 - 40
- [MG06] Ivo MILEV and Lothar GRUENDIG, "Geometrical Approximation and Segmentation of Laser Scanning Point Clouds" ,Promoting Land Administration and Good Governance 5th FIG Regional Conference for Africa,Accra, Ghana, March 8-11, 2006
- [MGPG04] Niloy J. Mitra, Natasha Gelfand, Helmut Pottmann, and Leonidas Guibas ;"Registration of point cloud data from a geometric optimization perspective", Eurographics Symposium on Geometry Processing (2004)
- [MMB01] Edward M. Mikhail, James S. Bethel, J. Chris McGlone . - Introduction to Modern Photogrammetry. -April 2001 Annexe.D
- [MRMT05] D. Girardeau-Montaut, M. Roux, R. Marc, G. Thibault; "Change detection on point cloud data acquired with a ground laser scanner" ;Proceedings of the ISPRS Workshop Laser scanning 2005 Enschede, the Netherlands, September 12-14, 2005
- [NMTS05] Viet Nguyen, Agostino Martinelli, Nicola Tomatis, Roland Siegwart : "A Comparison of Line Extraction Algorithms using 2D Laser Rangefinder for Indoor Mobile Robotics"; In Proceedings of Conference on Intelligent Robots and Systems, IROS'2005, Edmonton, Canada.
- [NNLH05] Natasha Gelfand, Niloy J. Mitra, Leonidas J. Guibas, Helmut Pottmann."Robust Global Registration" Eurographics Symposium on Geometry Processing (2005) 1-2
- [RB05] N. Ripperda, C. Brenner "marker-free registration of terrestrial laser scans using the normal distribution transform" Proceedings of the ISPRS Working Group V/4 Workshop 3D-ARCH 2005:
- [Sch05] F.SCHATZ " objets et modèles" épistémologie de la modélisation: modèle et simulation en architecture cours n°3 Master modélisation et simulation d' espaces bâtis 05/06
- [SRD01] A. Sappa, A. Restrepo-Specht, and M. Devy."Range Image Registration by using an Edge-based Representation" .In Proceedings of th 9th International Symposium on Intelligent Robotic Systems, (SIRS '01), Toulouse, France, July 2001.
- [SS87] Jacob T. Schwartz and Micha Sharir. Identification of partially obscured objects in three dimensions by matching noisy characteristic curves. International Journal of Robotics Research, 6(2):29--44, 1987.
- [TB95] Jean-Philippe Tarel, Nozha Boujemaa: Une approche floue du recalage 3D: généricité et robustesse Rapport de recherche n° 2716 Novembre 1995
- [WR97] H. WOLFSON, I.RIGOUTSOS : "Geometric hashing: an overview". In IEEE Comp. Sci. and Eng. (1997), pp. 10-21.
- [Zha92] Z.Zhang, "Iterative Point Matching for Registration of Free-Form Curves and Surfaces" Internat. J. Comput. Vision 13, 1994, 119--152

8 Terminologie

agrandissement des régions	region growing
arêtes de lissage	smooth edges
arêtes de pas	step edges
arêtes de pli	crease edges
arêtes de pli	fold edges
arêtes de saut / pas	jump edges
auto-occlusion	self occlusion
chevauchement	overlapping
convariance croisée	cross covariance
correspondance sous-jacente	underlying correspondance
DARCES	data-aligned rigidity-constrained exhaustive search
découper et fusionner	split-and-merge
du plus gros au plus fin	Coarse to fine
enclaves	inliers
FC3DR	recalage entièrement contraint d'un objet 3D
hachage géométrique	geometric hashing
ICL : la ligne itérative la plus proche	iterative closest line
ICP : le point itératif le plus proche	iterative closest point
NDT	Normal Distribution transformation
RANSAC	Random Sample Consensus
recalage 3D/3D	3D registration
recalage basé sur les entités	features based registration
recalage grossier	Coarse registration
recalage précis	fine registration
scan	scan
scannage	scanning
solution de forme fermée	closed-form solution
techniques de vote	voting methods
transformations rigide	rigid transformation