



**HAL**  
open science

# Contribution à la reconstruction 3D de bâtiments à partir de nuage de points de scanner laser terrestre

Abdelhamid Bennis

► **To cite this version:**

Abdelhamid Bennis. Contribution à la reconstruction 3D de bâtiments à partir de nuage de points de scanner laser terrestre. Automatique / Robotique. Université de Lorraine, 2015. Français. NNT : 2015LORR0139 . tel-01275382v2

**HAL Id: tel-01275382**

**<https://hal.science/tel-01275382v2>**

Submitted on 17 Feb 2016

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

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

# Contribution à la reconstruction 3D de bâtiments à partir de nuage de points de scanner laser terrestre

## THÈSE

présentée et soutenue publiquement le 2 octobre 2015

pour l'obtention du

**Doctorat de l'Université de Lorraine**  
**En Automatique, traitement du signal et de l'image, génie informatique.**

par

Abdelhamid BENNIS

### Composition du jury

<i>Rapporteurs :</i>	Regis LENGELLE Guillaume GELLE	Professeur à l'université de technologie de Troyes Professeur à l'Université de Reims Champagne-Ardenne
<i>Examineurs :</i>	David BRIE Vincent BOMBARDIER	Professeur à l'université de Lorraine Maitre de conférence à l'université de Lorraine
<i>Invité :</i>	Stéphane OHNIMUS	Directeur du CRITTBois

Centre de recherche en automatique de Nancy, Campus sciences - BP 70239 - 54506 Vandœuvre les Nancy  
Cedex. France.



*L'existence du mot ne prouve pas l'existence du concept.*





# Table des matières

<b>Remerciements</b>	<b>vii</b>
<b>Abréviations et Notations</b>	<b>ix</b>
<b>Introduction</b>	<b>1</b>
<b>1 Acquisition et traitement des nuages de points de SLT</b>	
1.1 Numérisation d'un bâtiment par scanner LASER terrestre . . . . .	7
1.1.1 Mesures de distance . . . . .	8
1.1.2 Système de balayage et mesure d'angle . . . . .	9
1.1.3 Résolution . . . . .	11
1.1.4 Précision de mesure . . . . .	12
1.2 Recalage des nuages de points . . . . .	14
1.2.1 Recalage par points homologues . . . . .	15
1.2.2 Recalage par surfaces et primitives géométriques . . . . .	17
1.3 Reconstruction de modèle de bâtiments à partir de nuages de points . . . .	17
1.3.1 Méthodes de reconstruction par maillage . . . . .	18
1.3.2 Méthodes de reconstruction par modèle paramétriques . . . . .	21
1.4 Conclusion . . . . .	24
<b>2 Segmentation du nuage de points</b>	<b>27</b>
2.1 Algorithmes de classification (clustering) . . . . .	27
2.1.1 K-means . . . . .	28
2.1.2 RANSAC . . . . .	28
2.2 Prise en compte des données colorimétriques pour la segmentation du nuage de points. 31	
2.2.1 Segmentation colorimétrique du nuage de points . . . . .	31
2.2.2 Intégration dans l'algorithme RANSAC . . . . .	35
2.3 Évaluation des performances . . . . .	39

2.3.1	Evaluation du temps de calcul . . . . .	40
2.3.2	Évaluation des paramètres de consistance . . . . .	44
2.4	Conclusion . . . . .	48
<b>3</b>	<b>Modélisation de la densité du nuage de points et extraction des points de contours</b>	<b>49</b>
3.1	Introduction . . . . .	49
3.2	TIN based boundary point extraction . . . . .	51
3.2.1	Delaunay triangulation . . . . .	51
3.2.2	Boundary points extraction considering Delaunay triangulation edge lengths	51
3.2.3	$\alpha$ -shapes . . . . .	52
3.3	Local Surface Sampling Steps Estimation . . . . .	53
3.3.1	Surface Sampling Function Approximation . . . . .	53
3.3.2	Low resolution approximation of the surface sampling steps . . . . .	55
3.3.3	Point cloud density approximation . . . . .	56
3.4	Experimental results . . . . .	58
3.4.1	Point cloud segmentation . . . . .	59
3.4.2	Density Comparison . . . . .	59
3.4.3	Application to edge point extraction . . . . .	62
3.4.4	Discussion . . . . .	64
3.5	Conclusion . . . . .	65
<b>4</b>	<b>Classification et modélisation des zones d'intérêts</b>	<b>69</b>
4.1	Classification des zones en zones d'irrégularités et éléments architecturaux	69
4.1.1	Évolution des propriétés géométriques des régions en fonctions du paramètre de segmenta	
4.1.2	Méthode de classification des régions . . . . .	72
4.2	Modélisation des éléments de façade . . . . .	75
4.2.1	Modélisation des éléments architecturaux . . . . .	76
4.2.2	Modélisation des régions d'irrégularités . . . . .	82
4.2.3	Modélisation des régions de façades . . . . .	82
4.3	Erreur de modélisation . . . . .	86
4.3.1	Approximation de l'erreur selon les axes principaux du plan . . . . .	87
4.3.2	Intégration de l'erreur sur le modèle CAO . . . . .	88

**Conclusion et perspectives**

---

<b>Bibliographie</b>	<b>93</b>
<b>Bibliographie</b>	<b>93</b>



# Remerciements

Les travaux présentés dans cette thèse ont fait l'objet d'une convention CIFRE entre l'association CRITTBois et le centre de recherche en automatique de Nancy.

Je tiens à exprimer ma gratitude à mon co-directeur de thèse monsieur Vincent Bombardier pour son professionnalisme, ses conseils et ses directives.

Je tiens également à témoigner toute ma reconnaissance aux Professeurs Regis Lengelle Professeur à l'université de technologie de Troyes et Guillaume Gelle Professeur à l'université de Reims pour l'honneur qu'ils m'ont fait en acceptant d'être rapporteurs de ce travail de recherche et membres du Jury.

Je termine par une profonde pensée à ma famille, ma femme et mes amis pour leur soutien sans faille durant ces longues années. Enfin, je remercie Gregory Baeteman et Ouiame Najim pour leur compagnie durant ces derniers mois.



# Abréviations et Notations

## Abréviation

ALS	: Airborne <i>laser</i> Scanning.
BIM	: Building Information Modeling/Model.
CAO	: Conception Assistée par Ordinateur.
DTM	: Digital Terrain Model.
ITE	: Isolation Thermique par l'Extérieur.
LADAR	: LASER Detection and Ranging.
LASER	: Light Amplification by Stimulated Emission of Radiation.
LIDAR	: LIght Detection and Ranging.
MIT	: Module d'Isolation Thermique.
SLT	: Scanner LASER terrestre.

## Notation

$p$	: scalaire.
$\mathbf{p}$	: vecteur.
$\mathbf{P}$	: Matrice.
$\mathbb{P}$	: Ensemble de points.
$\mathcal{P}$	: critère d'évaluation / Fonction coût .
$\ \mathbf{p}\ $	: Norme 2 d'un vecteur.
$\mathbf{p}^T, \mathbf{P}^T$	: Transposé d'un vecteur, matrice.
$ \mathbb{P} $	: Cardinal.
$\langle \mathbf{v}_1, \mathbf{v}_2 \rangle$	: Produit scalaire.





# Introduction

## Contexte

Depuis plusieurs décennies, la croissance du niveau de vie et le développement des infrastructures dans de nombreuses parties du monde induisent une compétition pour l'accès aux matières premières et notamment aux ressources énergétiques fossiles, qui sont actuellement la principale source d'énergie utilisée par l'Homme. Cette compétition participe, avec des tensions géopolitiques, à une augmentation de leur coût et à une incertitude sur la disponibilité d'énergie fossile abondante dans les décennies à venir. Alors qu'une part importante des bâtiments existants a été conçue avec l'idée d'une énergie indéfiniment peu chère, on constate un développement fort de la précarité énergétique des particuliers, et plus généralement, des marges de manoeuvres financières réduites pour de nombreuses structures en raison de frais énergétiques importants.

Outre la consommation énergétique, le fonctionnement des bâtiments induit une part majeure des émissions de gaz à effet de serre (GAS) en France. Aussi, amener le parc de bâtiments à un niveau de performance bien plus élevé est l'un des moyens les plus efficaces de réduire ces émissions. Dans cette optique, la loi Grenelle 1 prévoit deux mesures pour réduire la consommation sur les bâtiments neufs. La première consiste à limiter la consommation énergétique à  $50 \text{ kwh/m}^2/\text{an}$  en fin 2012. La seconde vise la construction de bâtiments à énergie positive à l'horizon de l'année 2020. Or, le parc immobilier s'accroît sans se renouveler [Fal13] comme en témoigne son taux de reconstruction<sup>1</sup> qui ne dépasse pas 1.1% par an et le taux de renouvellement<sup>2</sup> qui ne dépasse pas 0.07%. Ainsi, la seule limitation de la consommation énergétique pour les bâtiments neufs s'avère insuffisante pour atteindre les objectifs fixés pour 2050 lors des accords de Kyoto de 1997 puisqu'à cette date le parc immobilier sera composé d'environ deux tiers de bâtiments datant d'avant l'an 2000. Devant cet impératif, un objectif de rénovation de 500.000 logement/an a été fixé [Gou].

Dans ce contexte, de nombreux spécialistes de la performance énergétique mettent en garde contre des rénovations partielles sans planification de la performance visée une décennie plus tard car elles conduisent souvent à un coût de rénovation majoré. Ainsi, il est préférable d'intervenir si possible de façon définitive sur un élément du bâtiment, afin de ne pas viser une semi-performance qui imposerait de réitérer ces travaux moins

---

1. Le taux de reconstruction représente le ratio du nombre de bâtiments neufs sur le nombre de bâtiments existants

2. Le taux de renouvellement représente le ratio du nombre de bâtiments démolis sur le nombre de bâtiment existants

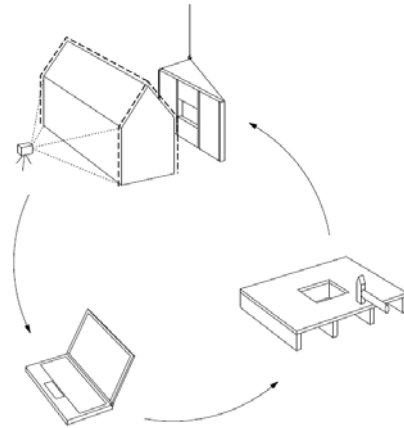


FIGURE 1 – Processus de réhabilitation par modules ossatures bois rapportées [Woo08].

de vingt ans après. Les solutions techniques d'isolation utilisant des Modules Ossatures Bois (MOB) rapportées sont employées de plus en plus fréquemment car elles impliquent généralement une augmentation de l'isolation des murs accompagnée d'un changement des menuiseries et de la ventilation. Elles passent souvent par un démontage des parois préexistantes pour les remplacer ou ajouter des nouveaux volumes dans le prolongement du bâtiment existant. Cette solution, illustrée par la figure 1, se compose de trois phases principales.

- Acquisition de la géométrie du bâtiment. Cette phase est généralement conduite par un géomètre expert. Elle consiste à prendre les dimensions réelles du bâtiment à l'aide d'une station totale, par photogrammétrie ou par scanner LASER terrestre (SLT). Les données produites par ces instruments de mesure sont utilisées pour produire un modèle CAO tel que construit du bâtiment ; cette étape de reconstruction est essentiellement manuelle.
- Conception des modules d'isolation en se fondant sur le modèle CAO tel que construit du bâtiment. Les constructeurs procèdent au dimensionnement des modules d'isolation, au choix du système de fixation et à leur positionnement, en prenant en compte les spécificités, tant géométriques que structurelles, du bâtiment.
- Montage des modules. Les modules mis au point sont montés sur les parois ou la structure du bâtiment. Généralement les constructeurs acceptent une marge d'erreur de l'ordre du centimètre entre la conception et la réalisation des modules.

Ce mode d'isolation s'avère efficace. Cependant sa généralisation se heurte à des difficultés relatives au dimensionnement et à la fixation des modules sur la façade du bâtiment. Les causes principales de ces difficultés sont liées d'une part à l'indisponibilité de passerelles logicielles entre les outils des différents intervenants de la conception : géomètres, architectes, bureaux d'études structure, fluides, incendies. Des outils de conversion ou des formats standardisés (notamment IFC) sont disponibles, mais ne sont pas ou peu utilisés. D'autre part, les technologies de mesure de types station totale ou photogrammétrie s'avèrent insuffisantes en termes d'information sur la surface pour la première et en terme de précision pour la seconde (voir figure (2)). Les scanners LASER terrestres (SLT)

représentent un compromis entre les deux méthodes d'acquisition précédemment citées. Ils délivrent un nuage de points discrétisant la géométrie des façades des bâtiments avec une précision de l'ordre du millimètre. Ils permettent, en outre, de mettre en évidence ses singularités géométriques telles que les défauts de planéité et/ou de rectitude. Cependant, cette technologie est encore peu utilisée car le processus de reconstruction d'un modèle CAO à partir du nuage de points reste manuel, long et fastidieux. Il nécessite entre 5 à 6 jours de modélisation par une seule personne qualifiée.

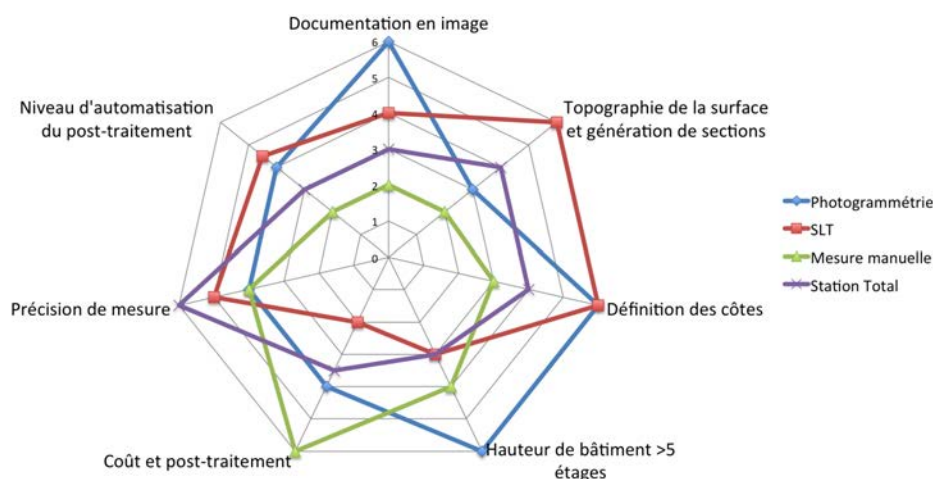


FIGURE 2 – Comparaison de méthodes d'acquisition de bâtiment pour réhabilitation thermique par l'extérieur, effectuée dans le cadre du projet Européen [Woo08]

L'objectif de ce travail de thèse est de lever les freins limitant l'utilisation de la technologie des SLT par les entreprises de construction et de rénovation en proposant une méthode de reconstruction automatique du modèle CAO de bâtiments à partir de nuages de points. Ces travaux ont été menés dans le cadre d'une thèse CIFRE en partenariat entre le Centre Régional d'Innovation et de Transferts Technologiques des industries du Bois (CRITTBois) et le Centre de Recherche en Automatique de Nancy (CRAN). Le CRITTBois est un centre de ressources pour les entreprises de la filière bois, installé depuis 1986 sur le campus de l'École Nationale Supérieure des Technologies et Industries du Bois (ENSTIB) à Epinal. A ce titre, une grande partie de ce travail a été orientée vers la rénovation énergétique de bâtiments par structures bois rapportées. Cependant, les techniques développées sont suffisamment générales pour être appliquées à d'autres techniques de rénovation et d'isolation de bâtiments.

## Problématique

Les SLT ont connu un fort développement au cours de la dernière décennie. Ils permettent la numérisation d'environnement 3D avec une précision de l'ordre du millimètre

dans des temps records (un million de points/s). Ces avantages ne vont pas sans inconvénient. Les nuages de points issus de la numérisation sont de très grande taille (généralement plusieurs millions de points), ce qui rend les coûts liés à leur exploitation (*i.e* stockage et traitement) très élevés. A ce titre, la reconstruction d'un modèle CAO à partir du nuage de points peut être vue comme une approche permettant une réduction importante du volume de données puisqu'elle consiste à représenter des sous-ensembles du nuage de points par des primitives géométriques (plans, cylindres, sphères, ...) ou des surfaces paramétriques (courbes de Bézier, Spline). Un certain nombre d'outils dédiés à la modélisation du nuage de points ont été développés. Cependant, le temps de modélisation manuelle reste fortement dépendant de la complexité géométrique du bâtiment, ce qui peut induire des coûts de mains d'oeuvre conséquents. Des méthodes de reconstruction semi-automatiques nécessitant une intervention limitée de l'utilisateur existent ([PV09], [BBH07], [FJZ05], [SA02]). Cependant, elles restent peu utilisées par l'industrie de la construction et la rénovation. D'une part ces méthodes nécessitent un nombre de paramètres conséquent et dont le niveau d'abstraction est souvent trop élevé pour l'utilisateur. D'autre part, la majorité des méthodes ne traitent pas l'aspect erreur du modèle par rapport aux dimensions réelles du bâtiment.

Les méthodes de reconstruction de modèle de bâtiments utilisent principalement deux approches pour la description de la géométrie d'un objet. La première, géométrie de construction de solides (CSG pour Constructive Solid Geometry), consiste à décrire le bâtiment par un ensemble de primitives géométriques liées entre elles par des opérateurs booléens. La seconde, représentation par bordure (B-rep pour Boundary representation), consiste à décrire le bâtiment par les frontières (bordures) des primitives géométriques, ou par les arêtes d'un maillage. Bien que ces deux modèles permettent une réduction du volume des données, ils ne mettent pas en évidence les irrégularités de la surface, information nécessaire pour le dimensionnement et le positionnement des structures rapportées.

En résumé, les principaux frein à l'utilisation de la technologie SLT dans l'industrie de la construction et de la rénovation sont relatifs à :

- une modélisation nécessitant une intervention manuelle d'opérateurs qualifiés ;
- un nombre de paramètres conséquent avec des niveaux d'abstraction élevés ;
- des modèles CAO qui ne mettent pas en évidence les irrégularités sur la façade du bâtiment.

Ce travail propose une méthode de reconstruction du modèle CAO 3D de bâtiments à partir d'un nuage de points de SLT, visant à lever au moins en partie, les limitations évoquées précédemment. Elle se compose de trois phases principales (voir le schéma bloc figure 3) pour obtenir le modèle filaire dans un fichier CAO format dxf :

1. **La segmentation du nuage de points** consiste à chercher le plan ayant le nombre de points consistant maximal. Cette étape est effectuée par une version modifiée de l'algorithme RANSAC, en intégrant les données colorimétriques. Le but étant de réduire la complexité algorithmique de RANSAC en ajoutant une étape préliminaire de classification colorimétrique du nuage de points.
2. **La modélisation du pas d'échantillonnage de la surface et l'extraction des points contours.** L'extraction des points contours est réalisée par l'algorithme  $\alpha$ -shape. La contribution de l'approche proposée consiste en l'amélioration de la

---

fiabilité de l'algorithme d'extraction par l'utilisation d'un seuil dynamique déduit à partir du pas d'échantillonnage de la surface. Pour cela une étape préliminaire de modélisation du pas d'échantillonnage a été ajoutée.

3. **La classification et la modélisation des éléments de façade.** Les points contours permettent de définir des zones qui peuvent représenter soit des éléments de la façade ou des irrégularités de celle-ci. La classification des zones se fonde sur deux critères relatifs à la région, à savoir la densité et l'aire. Les régions sont ainsi classées en trois types : région d'éléments architecturaux, région de façade et régions d'irrégularités.

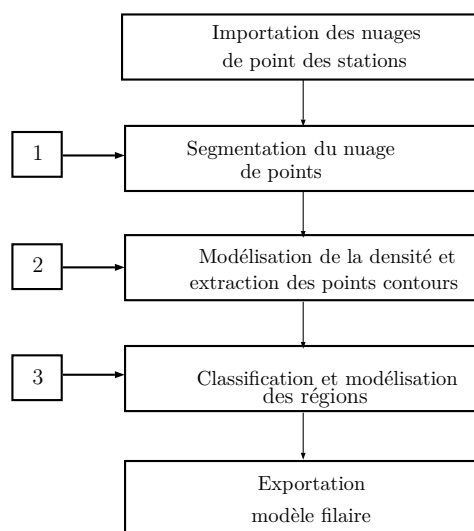


FIGURE 3 – Schéma de la méthode de reconstruction proposée.

## Organisation du document

Ce manuscrit est organisé classiquement. Le premier chapitre dresse un panorama des méthodes de reconstruction automatiques et semi-automatique de bâtiments à partir d'un nuage de points. Le second chapitre détaille la méthode de segmentation proposée. Le chapitre suivant présenté sous-forme de publication, décrit la méthode de modélisation du pas d'échantillonnage de la surface ainsi que son apport pour l'extraction des points de contours. Le dernier chapitre s'intéresse à la modélisation des différents éléments architecturaux ainsi que des régions d'irrégularités. La principale contribution à ce niveau tient dans l'approximation de l'erreur de modélisation et son intégration sur le modèle CAO. Une conclusion sur ces travaux ainsi que des perspectives sont enfin présentées.



# 1

## Acquisition et traitement des nuages de points de SLT

Dans ce chapitre, nous nous intéresserons en particulier à la méthode d'acquisition par scanner LASER terrestre (SLT). La section 1.1 présente un état de l'art des technologies de mesure. La section 1.3 présente les différentes méthodes de reconstruction de modèles à partir de nuage de points. Afin de faciliter la suite de ce paragraphe, il est nécessaire d'introduire les notations considérées pour la description des différentes données utilisées.

A l'issue du balayage de la scène, le SLT fournit un nuage de points. Chaque point est défini par ses coordonnées géométriques  $\mathbf{p}_i = [x_i, y_i, z_i]$ , l'intensité du LASER réfléchi  $il_i$ , et la couleur de la surface en ce point  $\mathbf{c}_i = [r_i, v_i, b_i]$ . L'acquisition de la couleur est faite par camera photographique montée sur le scanner. Le lecteur pourra se rapporter aux travaux de [FK04],[BF12] sur la texturation du nuage de points par des images photographiques. Dans la suite du manuscrit, on considère  $\mathbb{P}$  l'ensemble de points définis dans le référentiel géométrique et  $\mathbb{C}$  l'ensemble de points définis dans le référentiel colorimétrique.

### 1.1 Numérisation d'un bâtiment par scanner laser terrestre

Le principe de mesure 3D des SLT statiques est inspiré de celui du théodolite et la station totale [PB07b]. La mesure des coordonnées tridimensionnelles de chaque point de la scène se fait dans un système de coordonnées sphériques. Pour chaque point, le SLT mesure la distance ( $\rho$ ), l'azimuth ( $\theta$ ) et l'élévation ( $\phi$ ) (figure 1.1). On en déduit les coordonnées Euclidienne selon (1.1).

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \rho \cdot \begin{pmatrix} \cos(\phi) \cdot \sin(\theta) \\ \cos(\phi) \cdot \cos(\theta) \\ \sin(\phi) \end{pmatrix} \quad (1.1)$$

Ce système de mesure est composé de deux sous système, dont l'un mesure la distance et l'autre mesure d'angle. Dans la suite, une présentation de chaque sous-système est faite d'une manière plus explicite respectivement dans les paragraphes 1.1.1 et 1.1.2. Les



propriétés de chaque système, à savoir la résolution ainsi que la précision de mesure sont ensuite détaillées respectivement dans les paragraphes 1.1.3 et 1.1.4.

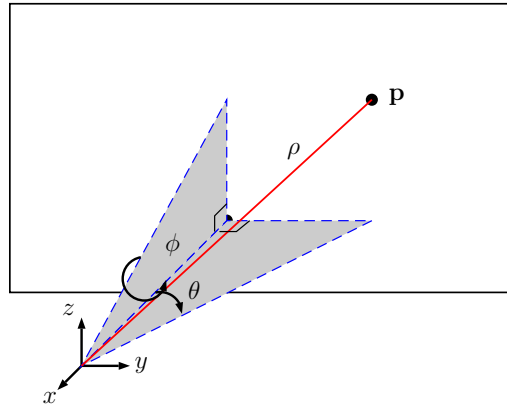


FIGURE 1.1 – Modèle géométrique d'un SLT.

### 1.1.1 Mesures de distance

Bien qu'on trouve dans la littérature plusieurs technologies de mesure de distance par optique, les SLT ne font appel qu'à trois d'entre elles, à savoir : mesure par triangulation, la mesure par différence de phase et la mesure par temps de vol.

- *Par triangulation optique*, la distance est mesurée par l'angle de l'émission  $ae$  du faisceau laser et l'angle d'observation  $ao$  (figure 1.2) par :

$$\rho = l \cdot \frac{\sin(ao)}{\sin(ae + ao)} \quad (1.2)$$

Ce principe de mesure permet une grande densité du nuage de points et une faible erreur de mesure à courte distance. Ainsi les SLT qui se basent sur ce principe de mesure sont souvent utilisés pour le contrôle de qualité et de conformité des pièces industrielles [Bla04].

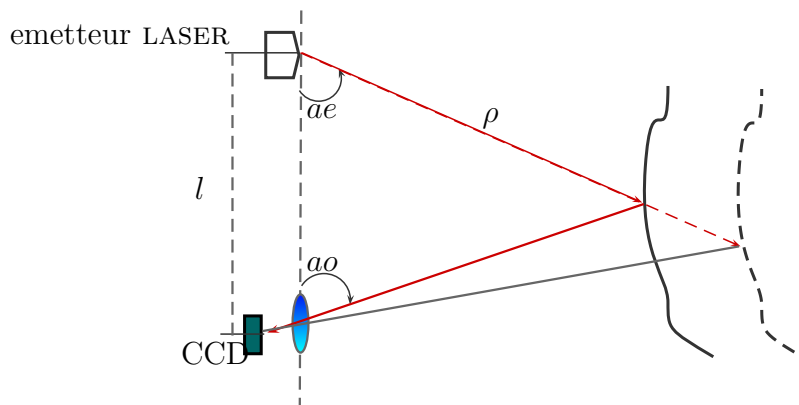


FIGURE 1.2 – Principe de mesure de distance par triangulation.

- *Par différence de phase*, la distance est obtenue par la comparaison de la phase du signal émis et celle du signal reçu [PB07b]. Le SLT envoie une onde LASER modulée en amplitude vers l'objet. L'énergie réfléchie par la surface est ensuite captée par le SLT (figure 1.3). La distance est déduite selon :

$$\rho = \frac{\lambda}{2} \left( k + \frac{\delta\varphi}{2\pi} \right) \quad (1.3)$$

où  $k$  est le nombre de périodes de la porteuse émises avant l'interception du signal réfléchi et  $\lambda$  la longueur d'onde. La précision de mesure par cette méthode dépend en grande partie de la fréquence utilisée. Plus la fréquence est grande plus la mesure est précise. Cependant, leur portée est généralement limitée à 120 m compte-tenu de la faible intensité du signal modulé [Sch07b]. Ainsi, les SLT utilisant cette méthode sont souvent réservés pour numériser des environnements industriels, monuments historiques ainsi que l'architecture de bâtiment.

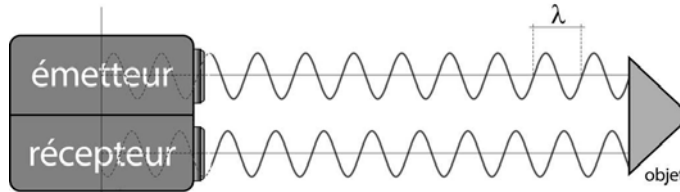


FIGURE 1.3 – Principe de mesure par différence de phase

- *Par temps de vol*, la distance est mesurée par le temps que prend un faisceau LASER pour effectuer un aller-retour entre le SLT et la surface de l'objet. Pour cela, un compteur (Timer) est déclenché lors de l'émission du faisceau, et arrêté lorsque le faisceau revient sur le capteur (figure 1.4). La distance  $\rho$  peut être déduite selon l'équation (1.4).

$$\rho = c \cdot \frac{\delta t}{2} \quad (1.4)$$

où  $c$  représente la célérité dans le milieu de propagation.

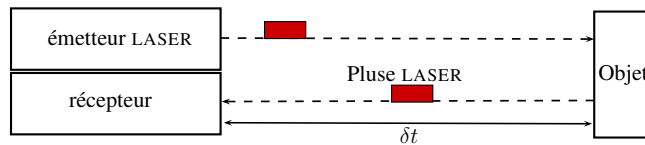


FIGURE 1.4 – Principe de mesure par temps de vol [Bou10].

### 1.1.2 Système de balayage et mesure d'angle

L'orientation du LASER est obtenue par un système de déflexion dont la rotation ou l'oscillation s'effectue selon deux axes de rotation perpendiculaires. D'après [FM04] le champ de vision/balayage du SLT dépend de la structure du système de déflexion. En se rapportant aux travaux de [Res09], ces systèmes peuvent être classés en trois catégories.

- *Déflexion par deux miroirs oscillants* où la déflexion du rayon LASER est effectuée par deux miroirs oscillants synchronisés. Le premier est dédié à la déflexion horizontale et le second à la déflexion verticale. La figure 1.5 présente une illustration de ce système. Le champ de vision du SLT est limité par les angles maximaux d'oscillation des miroirs. Généralement limité à  $60^\circ \times 60^\circ$ , ce système est utilisé dans les SLT de type caméra, majoritairement utilisés dans les SLT à longue portée.

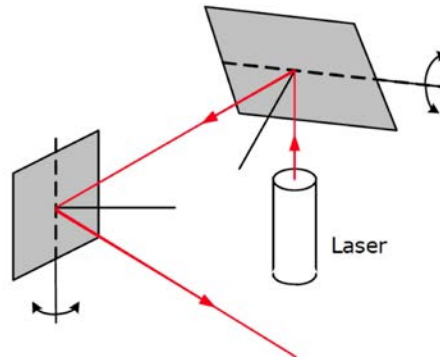


FIGURE 1.5 – Système de déflexion par deux miroirs oscillants [Res09].

- *Déflexion par un miroir oscillant* où le faisceau LASER est réfléchi par un miroir plat oscillant ou un miroir de forme polygonale en rotation selon un axe horizontal (figure 1.6). Afin de numériser toute la scène, le SLT effectue une rotation horizontale. Ainsi, le champ de vision est généralement limité à  $360^\circ \times 60^\circ$ . Ce système de balayage est dit hybride, car le champ de vision est limité en élévation par l'angle d'oscillation maximal du miroir de déflexion.

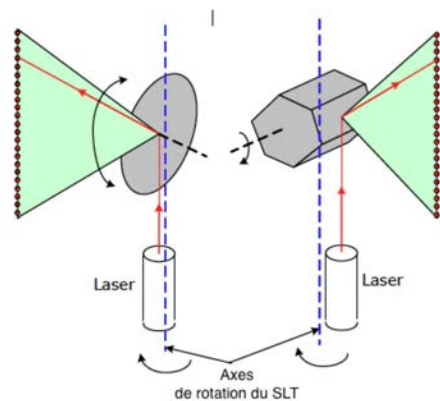


FIGURE 1.6 – Système de déflexion : à gauche un miroir oscillant à droite, un miroir polygonale en rotation [Res09].

- *Déflexion par un miroir en rotation*. Ce système est utilisé par les SLT à champ de vision panoramique (généralement entre  $360^\circ \times 180$  et  $360^\circ \times 310^\circ$ ). Comme les systèmes hybrides, le balayage de la scène se fait par un profilage vertical puis une rotation horizontale du SLT. La différence, permettant un champ de vision plus grand, réside

dans le système de déflexion. Celui-ci utilise un seul miroir plat en rotation selon un axe horizontal (figure 1.7).

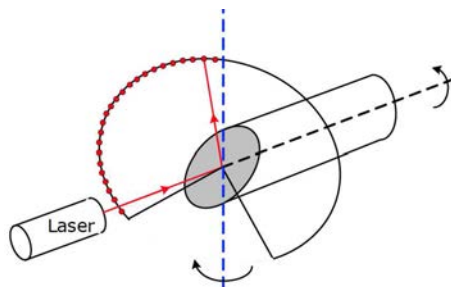


FIGURE 1.7 – Système de déflexion par miroir positionné à 45° en rotation sur un axe horizontal [Res09]

Les angles d'orientation du faisceau LASER sont mesurés par des codeurs -incrémentaux ou binaires- positionnés sur les axes de rotation du système de déflexion. Sa rotation est faite par des servomoteurs positionnés sur les axes de rotation ; le pas angulaire dépend de la résolution souhaitée par l'utilisateur.

### 1.1.3 Résolution

La résolution d'un SLT se définit par sa capacité à distinguer les fins détails sur une surface d'objet. [Bry] mesure la qualité de définition d'un détail sur la surface par :

$$Q = 1 - \frac{s}{s_d} \quad (1.5)$$

où  $s_d$  est la taille du détail à numériser, et  $s$  le pas d'échantillonnage sélectionné par l'utilisateur. Ce dernier est défini pour une surface plane et orthogonale au SLT à une distance  $\rho_n$ . Le pas angulaire peut être déduit selon :

$$\Delta\theta = \arctan\left(\frac{s}{\rho_n}\right) \quad (1.6)$$

D'après [LJ06], la résolution angulaire maximale dépend du pas angulaire minimal que les codeurs peuvent mesurer. De plus, le pas angulaire ne doit pas être inférieur à la taille de l'empreinte LASER sur la surface de l'objet, ceci afin d'éviter le chevauchement de deux empreintes LASER sur la surface de l'objet. Bien que l'équation 1.5 reflète dans une certaine mesure la qualité de détection des petits détails sur la surface d'un objet, celle-ci ne prend pas en compte la résolution de mesure de distance. [WL99] définit la résolution de mesure à temps de vol et à différence de phases respectivement par :

$$\Delta R_{puls} = \frac{c}{2} \cdot \Delta t_l \quad (1.7)$$

$$\Delta R_{df} = \frac{\lambda_{short}}{4\pi} \cdot \Delta\varphi \quad (1.8)$$

où  $\Delta t_l$  et  $\Delta \varphi$  représentent respectivement la résolution temporelle et la résolution spatiale (longueur d'onde) que les systèmes de mesure de distance peuvent distinguer. Il est à noter que la densité du nuage de points n'est pas uniforme. Celle-ci décroît en fonction de la distance ainsi que l'angle d'incidence du faisceau *laser* sur la surface. Ce point sera étudié en détail dans le troisième chapitre avec la proposition d'une méthode de modélisation du pas d'échantillonnage de la surface.

### 1.1.4 Précision de mesure

**Erreur de mesure de distance :** la précision de mesure de distance dépend de plusieurs facteurs intrinsèques (capteur du signal, bruit du capteur et système d'amplification) et extrinsèques (milieu de propagation de l'onde LASER, distance, réflectance de la surface, angle d'incidence) au SLT. Dans ce paragraphe, un intérêt particulier a été porté aux systèmes de mesure de distance, utilisés pour la numérisation d'un bâtiment, les SLT à différence de phases et temps de vol.

Pour la triangulation optique, l'erreur de mesure de distance croît proportionnellement à la distance mesurée (dégénérescence du triangle de mesure). Cependant, pour les mesures par différence de phases et temps de vol, la précision de mesure est inversement proportionnelle à l'intensité du signal réfléchi par la surface et le temps de réponse du capteur. Selon [WL99], l'erreur de mesure pour les deux technologies peut être approximée respectivement par l'équation (1.9) et l'équation (1.10).

$$\sigma_\rho = t_{rise} \cdot \frac{c}{2} \cdot \frac{\sqrt{BP_{tv}}}{Il_{pic}} \quad (1.9)$$

$$\sigma_\rho = t_{rise} \cdot \frac{\lambda_{min}}{4\pi} \cdot \frac{\sqrt{BP_{df}}}{Il_{moy}} \quad (1.10)$$

où :

- $BP_{tv}$  et  $BP_{df}$  représentent la bande passante du bruit des capteurs ;
- $Il_{pic}$  Intensité maximale du faisceau réfléchi ;
- $Il_{moy}$  Intensité moyenne de l'onde continue réfléchie.

Afin de réduire la dépendance au temps de réponse  $-t_{rise}-$  du capteur, la majorité des SLT utilisent des capteurs de signal de type Photo-Diode à effet d'avalanche, connus pour leur faible temps de réponse [Sch07b]. Cependant, l'intensité LASER reste dépendante de la réflectance de la surface de l'objet et le milieu de propagation de l'onde *laser*. Selon [Jel92] et [HP07], la relation entre l'intensité du faisceau LASER émise et l'intensité captée peut être d'écrite par l'équation (1.11).

$$Il = \frac{\pi Il_E r \cos \alpha}{(4 r^2) r_d \eta_{Atm} \eta_{Sys}} \quad (1.11)$$

où :

- $r$  et  $r_d$  représentent respectivement la réflectance de la surface et la réflectance de la surface dans une direction donnée ;
- $\alpha$  angle d'incidence ;

- $\rho$  distance entre l'objet et le capteur ;
- $I_E$  Intensité du LASER émit ;
- $\eta_{Atm}$  et  $\eta_{Sys}$  représentent l'absorption par l'atmosphère et la transmission du système ;

Dans cette équation, on suppose que l'objet est un réflecteur Lambertien parfait<sup>3</sup> et que la surface de l'objet est supérieure à la taille de l'empreinte LASER. Ainsi, l'énergie réfléchie ne dépend que de la réflectivité de la cible et de l'angle d'incidence. [CR04] ont développé des modèles de correction à partir de données issues de tests qui ont été menés sur des surfaces de différentes couleurs et matériaux positionnés à différentes distances et angles d'incidence.

**Erreur de mesure d'angle :** les causes induisant une erreur de mesure d'angle dépendent de la technologie de balayage utilisée. Ce paragraphe se limite aux causes affectant les systèmes de déflexion présentés. En se référant au travaux de [Res09], on recense les causes suivantes :

- Erreur de synchronisation entre le système de mesure de distance et le système de mesure d'angle. Cette erreur varie durant la période d'accélération du système de rotation, et devient constante une fois le scanner atteint la vitesse nominale.
- Décalage entre l'angle nul  $\ll 0^\circ \gg$  du codeur et l'angle nul de l'axe de rotation du miroir dédié à la déflexion verticale (mesure de l'élévation). Cette erreur peut être issue d'un décalage au niveau du montage du codeur ou un décalage du convertisseur analogique-numérique.
- Erreur d'échelle qui est linéairement dépendante de l'angle de mesure. Celle-ci peut être causée par une déficience du codeur ou à une erreur de la loi de commande.

L'erreur issue des deux dernières causes peut être estimée par l'équation 1.12 :

$$\Delta\phi = \phi_{scan} - \phi_{correc} = \phi_{scan} \cdot \delta\phi + \phi_0 \quad (1.12)$$

où :

- $\phi_{scan}$  angle mesuré ;
- $\phi_{correc}$  angle mesuré après correction ;
- $\phi_0$  erreur de décalage de l'angle nul ;
- $\delta\phi$  erreur d'échelle.

**Erreur d'axes :** l'assemblage des différentes composantes du SLT se fait manuellement, induisant ainsi une erreur de positionnement des axes de rotation. Celle-ci se traduit par des erreurs d'axes du modèle géométrique du SLT. En effet, dans le cas idéal, les axes de rotation -horizontale et verticale- s'intersectent avec l'axe de collimation dans l'origine du repère (illustré dans la figure 1.8). Cependant, les imperfections du montage manuel induisent des erreurs sur les axes, engendrant ainsi une excentricité de l'origine du repère.

- Erreur d'axe vertical : elle peut être due à un décalage de l'axe de rotation vertical par rapport au centre du repère  $e_v$  et/ou à une inclinaison du SLT  $\gamma_v$ . Cette dernière est corrigée par un compensateur d'inclinaison embarqué dans le SLT [MKL07].

---

3. Un réflecteur Lambertien est une surface parfaitement diffusante, où l'énergie réfléchie est également distribuée dans toutes les directions

- Erreur d'axe horizontal : en plus du décalage par rapport au centre du repère  $e_h$ , l'axe vertical peut avoir une erreur de non-orthogonalité représenté par l'axe de rotation vertical. En supposant que les deux axes soient coplanaires, l'erreur de non-orthogonalité des axes est calculée par l'angle  $\gamma_h$ , entre les deux axes dans le plan vertical.
- Erreur de collimation : l'erreur de collimation est représentée par l'angle  $\gamma_c$  entre le vecteur normal à l'axe de rotation horizontal, et l'axe de collimation du faisceau LASER dans le plan défini par la direction de l'axe horizontal et son vecteur normal.

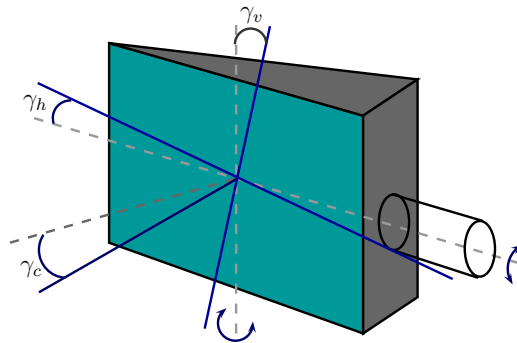


FIGURE 1.8 – Erreur de positionnement des axes de rotation et l'axe de collimation

La calibration d'un SLT consiste à mettre au point un modèle géométrique du SLT prenant en compte les erreurs instrumentales, dites aussi systématiques. Dans cet objectif, plusieurs méthodes de calibrations ont été proposées durant la dernière décennie : l'auto-calibration, la calibration par traqueur LASER, calibration par méthodes conduites par les données ([Lic07],[LSTS00]). Cependant, la calibration reste limitée aux erreurs intrinsèques du SLT et reste peut effective pour les facteurs extrinsèques tels que le milieu de propagation de l'onde LASER et la réflectance de la surface de l'objet. De plus amples informations sur ces erreurs sont présentées dans les travaux de calibration de SLT hybrides et panoramiques de [Sch07b].

## 1.2 Recalage des nuages de points

Afin de garantir une densité suffisante du nuage de points et de remédier à l'effet d'occlusion, l'objet est scanné à partir de plusieurs points de vues (ou stations ou positions). Le nuage de points de chaque position est acquis dans le référentiel propre à la station comme illustré dans la figure 1.9. Une procédure de recalage - on trouve aussi les désignations orientation et consolidation dans la littérature - permet d'exprimer l'ensemble des points dans un même référentiel.

Il existe deux types de recalage :

- absolu où les nuages de points sont rapportés à un référentiel d'une des positions d'acquisition ; ce type de recalage est utilisé dans la reconstruction de milieux industriels et la numérisation de bâtiment.
- géoréférencement : le nuage de points est geo-localisé de l'environnement scanné dans le système de coordonnées terrestres. Pour cela, les nuages de points des différentes

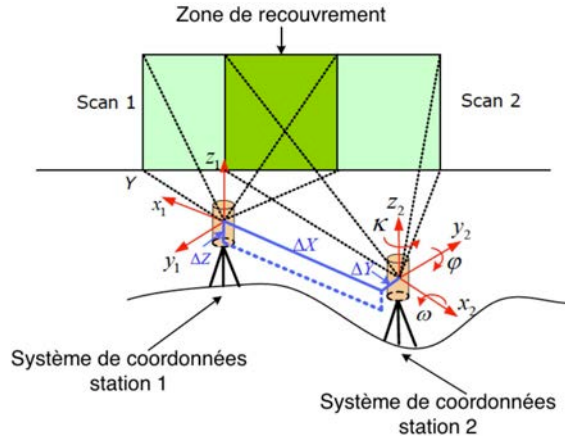


FIGURE 1.9 – Acquisition d’un objet par deux stations [Res09]

stations sont recalés dans un référentiel absolu, puis géoréférencé par recalage avec un plan cadastral, ou à l’aide de coordonnées GPS de la position du SLT. Ce type de recalage est utilisé lors de la réalisation de système d’information géographique. Le recalage dans un repère absolu nécessite de calculer les paramètres de transformation qui expriment le passage du référentiel de la station au référentiel absolu. La matrice de transformation  $\mathbf{M}$  est composée de la matrice de rotation  $\mathbf{M}_r$  et du vecteur de translation  $\mathbf{m}_t$ .  $\mathbf{p}_a$  représente les coordonnées du point  $\mathbf{p}_s$  dans le référentiel absolu après transformation.

$$\mathbf{p}_a = \mathbf{M} \cdot \mathbf{p}_s \quad (1.13)$$

$$\Leftrightarrow \begin{pmatrix} \mathbf{p}_a \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{M}_r & \mathbf{m}_t \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \mathbf{p}_s \\ 1 \end{pmatrix}$$

Dans ce paragraphe, nous nous limitons aux méthodes les plus courantes. D’autres méthodes ont été proposées permettant le recalage en utilisant des fonctions coût différentes, dépendant du domaine d’application. Pour de plus amples informations sur cet aspect, le lecteur peut se référer aux travaux de [SMFF07] et [SHA06]. Les méthodes de recalage sont présentées selon deux points de vues. Le premier concerne les méthodes de recalage par points homologues. Le second point de vue permet d’inclure de la connaissance *à priori* en structurant les données sous la forme de primitives géométriques.

### 1.2.1 Recalage par points homologues

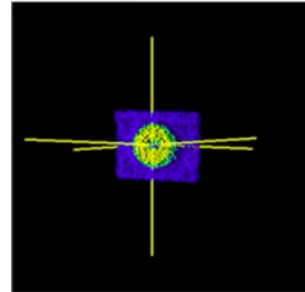
Ces méthodes consistent à mettre en correspondance des points de contrôles définis sur les deux nuages de points. La détermination des paramètres de la matrice de transformation se fait généralement par la méthode des moindres carrés. Pour cela, il faut considérer six points homologues au minimum dans les deux référentiels. Ces derniers peuvent être des centres de cibles (patchs plan, sphères) de recalage positionnées dans l’environnement numérisé [BPPU02] (figure 1.10), ou des points caractéristiques (figure 1.11) définis ma-



nuellement par l'utilisateur [Jac05] ou à l'aide d'algorithmes d'extraction des ces points tel que l'opérateur de Harris [HS88].

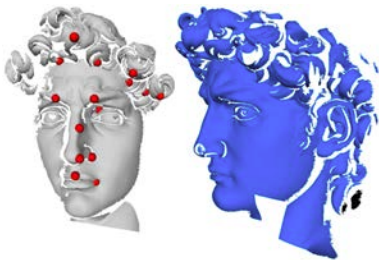


(a) Cible de recalage plate



(b) Centre estimé de la cible

FIGURE 1.10 – Exemple de cible de recalage



(a) Nuages de points dans leurs référentiels respectifs



(b) Nuages de points recalées

FIGURE 1.11 – Exemple de recalage de nuages de points par points de contrôles [MGPG04]

Pour le recalage par cibles, l'erreur du recalage dépend en grande partie de l'estimation du centre de la cible. Pour réduire cette erreur, les cibles sont scannées avec une grande densité [GL04]. La sélection des points de contrôle reste peu précise car la sélection des points homologues se fait manuellement, réduisant ainsi la probabilité de trouver une transformation optimale [BMB03]. Afin de réduire cette dépendance à l'initialisation, [BM92] propose une méthode itérative de recalage : *iterative closest point* (ICP) qui consiste à estimer les paramètres de transformation en étendant la recherche de points homologues dans le voisinage des points de contrôle initialement définis par l'utilisateur. Cette méthode reste dépendante de l'initialisation des points de contrôle. Afin de s'affranchir de ce problème, [CH99] propose une méthode de recalage se fondant sur l'algorithme RANSAC (*Random Sample Consensus*). L'algorithme consiste à sélectionner de manière aléatoire les points homologues dans les deux nuages de points, à partir desquels sont déduits les paramètres de la matrice de transformation. L'ensemble des paramètres retenu est celui maximisant le nombre de points superposés. Si cet algorithme est plus robuste que les deux précédents, la qualité du recalage reste dépendante de la taille de la zone de recouvrement.

## 1.2.2 Recalage par surfaces et primitives géométriques

Il consiste à mettre en correspondance des primitives géométriques (courbe, plan, cylindre, sphère...) ou des surfaces approximées à des sous-ensembles de points dans les deux nuages, puis à chercher les paramètres de transformation permettant de minimiser l'erreur entre l'ensemble des formes géométriques des deux nuages de points. [SRSD01] propose une méthode de recalage d'images de distance, en cherchant les paramètres de transformation maximisant l'appariement des courbes et/ou droites définies par des points caractéristiques du nuage de points. [GA05] étend cette méthode aux surfaces en proposant une méthode basée sur l'algorithme ICP. Celle-ci consiste à modéliser la surface du nuage de points de référence, puis dans un second temps, à chercher les paramètres de transformation minimisant la distance entre les points du nuage de points à transformer et la surface modélisée. [RDvdHV07] propose une solution de modélisation de milieux industriels en commençant par une segmentation du nuage de points par transformation de Hough, puis dans un second temps, le recalage des primitives géométriques ainsi définies. Ces méthodes permettent des performances supérieures au recalage par points de contrôle car l'approximation de surfaces et de droites réduit dans une certaine mesure l'erreur d'acquisition [Dev06]. Cependant, elles restent dépendantes de la diversité des formes géométriques présentes dans le nuage de points. Ces méthodes sont recommandées pour des applications de recalage des nuages de points des milieux industriels et monuments historiques [Jac05].

Dans ces travaux, il a été choisi d'utiliser la méthode de recalage fournie avec le SLT. Elle se base sur le principe de recalage par points homologues, ces derniers sont définis à partir de cibles positionnées dans l'environnement scanné.

## 1.3 Reconstruction de modèle de bâtiments à partir de nuages de points

La reconstruction de modèle géométrique de bâtiments à partir de nuage de points la base sur laquelle plusieurs informations seront imbriquées selon le domaine d'application, comme le : Building Information Modelling (BIM), le modèle de milieux urbains, la thermographie 3D, la réalité augmentée, etc. D'après [Bre05], la géométrie d'un objet reconstruit à partir de nuage de points peut être décrite par trois représentations géométriques [Jam95].

La géométrie de construction du solide (ou CSG pour Constructive Solide Geometry) dite aussi modélisation volumique, consiste à décrire la géométrie de l'objet par un arbre de primitives géométriques (plan, cylindre, sphère, cone et torus) liées entre elles par des opérateurs booléens (intersection, union et soustraction). Cette représentation est utilisée dans des applications ne faisant pas appel à des formes géométriques complexes où la précision est exigée, comme la conception de pièces mécaniques.

La modélisation surfacique (B-rep pour Boundary representation) consiste à décrire la surface de l'objet par un ensemble de facettes planes définies par leurs bordures. Cette représentation peut être optimisée par la modélisation d'un ensemble de facettes adjacentes par une surface paramétrique ou une décimation du maillage. Elle est utilisée dans

des applications de reconstruction de formes géométriques complexes telles que les monuments historiques. Elle est cependant moins performante en terme de compacité que la représentation CSG.

La décomposition spatiale consiste à représenter la géométrie de l'objet par un ensemble de cellules, voxels ou cubes sur une grille tridimensionnelle. Ce type de représentation est utilisée dans des applications en imagerie médicale.

Le choix d'une représentation dépend en majeure partie du domaine d'utilisation. Dans la littérature de reconstruction de bâtiments à partir de nuage de points, la majorité des méthodes se base sur la représentation B-rep. Selon [PV09], ce choix est du à la taille de la représentation B-rep qui est beaucoup plus petite que celle d'une représentation par décomposition spatiale. Aussi, elle permet de modéliser des formes géométriques plus complexes, contrairement à la représentation CSG, restreinte aux primitives géométriques usuelles.

La suite de cette section présente différentes méthodes de reconstruction, ayant un lien directe avec le bâtiment. Les méthodes de reconstruction peuvent être classées par rapport à différents critères comme les données utilisées pour la reconstruction, le principe de reconstruction ou le type de modèle reconstruit. Dans ce manuscrit, nous avons opté pour cette dernière classification.

### 1.3.1 Méthodes de reconstruction par maillage

Dans le cas de la reconstruction par maillage, la façade du bâtiment est représentée par un maillage composé de triangles. Il peut être obtenu par une triangulation de Delaunay pour une image de profondeur (repère 2.5D), ou par l'algorithme  $\alpha$ -shape [EM94] pour un nuage de points dans un repère 3D. Nous détaillons ces méthodes dans le troisième chapitre.

[SA02] propose une méthode de reconstruction de modèle volumique, où chaque façade du bâtiment est représentée par un maillage texturé. Le modèle volumique du bâtiment est reconstruit par extrusion du maillage des images de profondeurs des façades (voir figures 1.12-a et b). Le modèle est ensuite texturé par le recalage des images photographiques. Les paramètres du recalage sont calculés par mise en correspondance d'une part de lignes caractéristiques des images photographiques et d'autre part des images de profondeurs. Ces lignes caractéristiques pour les images photographiques sont estimées par régression orthogonale de droites sur un sous-ensemble des points de contours connexes extraits par l'algorithme de Canny [Can86]. Les lignes caractéristiques des images de profondeurs représentent les intersections de surfaces planes segmentées par croissance de région. La figure 1.12-c présente le modèle texturé résultant.

[FJZ05] introduit une méthode de reconstruction du milieu urbain à partir de nuages de points acquis par scanner LASER terrestre mobile (SLTM). La plate-forme de numérisation dispose d'un scanner LASER et une caméra photographique. La méthode de reconstruction se compose de trois étapes principales. Dans un premier temps, le nuage de points est partitionné en sous-ensembles qui correspondent à des segments de droites de la trajectoire du déplacement de la plate-forme d'acquisition. Les sous-ensembles de points sont ensuite transformés en images de profondeurs, ou images de distances. Les figures 1.13-a-b présentent respectivement un nuage de points et l'image de profondeur

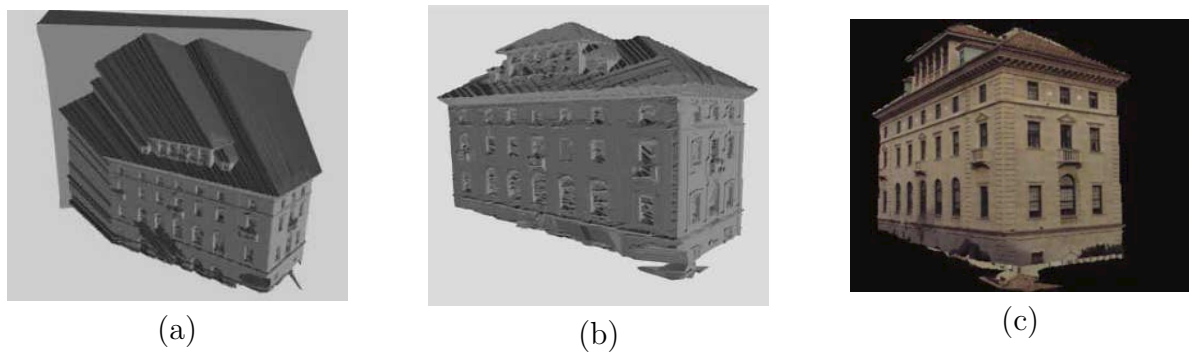


FIGURE 1.12 – Différentes étapes de la méthode de modélisation proposée dans [SA02] : (a) Extrusion d’un maillage de surface; (b) Modèle volumique dont les facettes sont représentées par maillage; (c) Modèle photo-réaliste du bâtiment

correspondante. Les points sont alors partitionnés en deux classes, les points d’objets présents dans la scène durant la numérisation (véhicules, arbre, ...) et les points de façade. Les zones de façade occultées par les objets présents dans la scène sont remplies par interpolation de leurs points de bordures respectifs. Le modèle (présenté dans la figure 1.13-c) est alors construit par maillage de Delaunay, qui est texturé par les images prises lors de la numérisation de la scène (figure 1.13-d).

Une méthode de reconstruction de modèle urbain à partir d’images stéréo géo-référencées a été introduite par [PPPD05]. Dans un premier temps, le nuage de points représentant la façade du bâtiment est déduit à partir des images de la façade qui sont transformées en images de profondeurs (voir figure 1.14-b). Ces images sont calculées par appariement des pixels de l’image référence avec les pixels des autres images de la façade. Le pixel homologue est celui qui permet une corrélation optimale avec le pixel de l’image référence. L’image de profondeur résultante est optimisée en rejetant les pixels ayant un faible score de corrélation et ceux ayant une faible texture<sup>4</sup>. Le nuage de points de la façade est alors reconstruit par orientation des images de profondeur dans un repère absolu. Finalement, le modèle par maillage est reconstruit par application de la triangulation de Delaunay (figure 1.14-c) avant sa texturation.

Les travaux de [BBH07] proposent une méthode de reconstruction à partir de nuage de points de SLT et de plan cadastral. Dans un premier temps, le nuage de points est recalé à un modèle volumique du bâtiment défini par le plan cadastral [SB05] (voir figure 1.15-a). Le nuage de points est ensuite segmenté en sous-ensembles de points correspondants aux façades du bâtiment. Une image raster pour chaque plan est alors déduite (figure 1.15-b). Celle-ci représente la distance des points par rapport au plan défini par extrusion du plan cadastral. Cette distance est alors interpolée sur le modèle volumique par la méthode de Displacement mapping. Le plan correspondant au sous-ensemble est partitionné en des cellules rectangulaires, une intrusion ou extrusion leur est alors appliquée en fonction de la déviation des pixels de l’image raster. La figure 1.15-c présente le modèle résultant de cette approche de reconstruction.

4. Un pixel est dit de faible texture si la variance de la vignette centrée sur ce pixel est inférieure à un seuil prédéfini

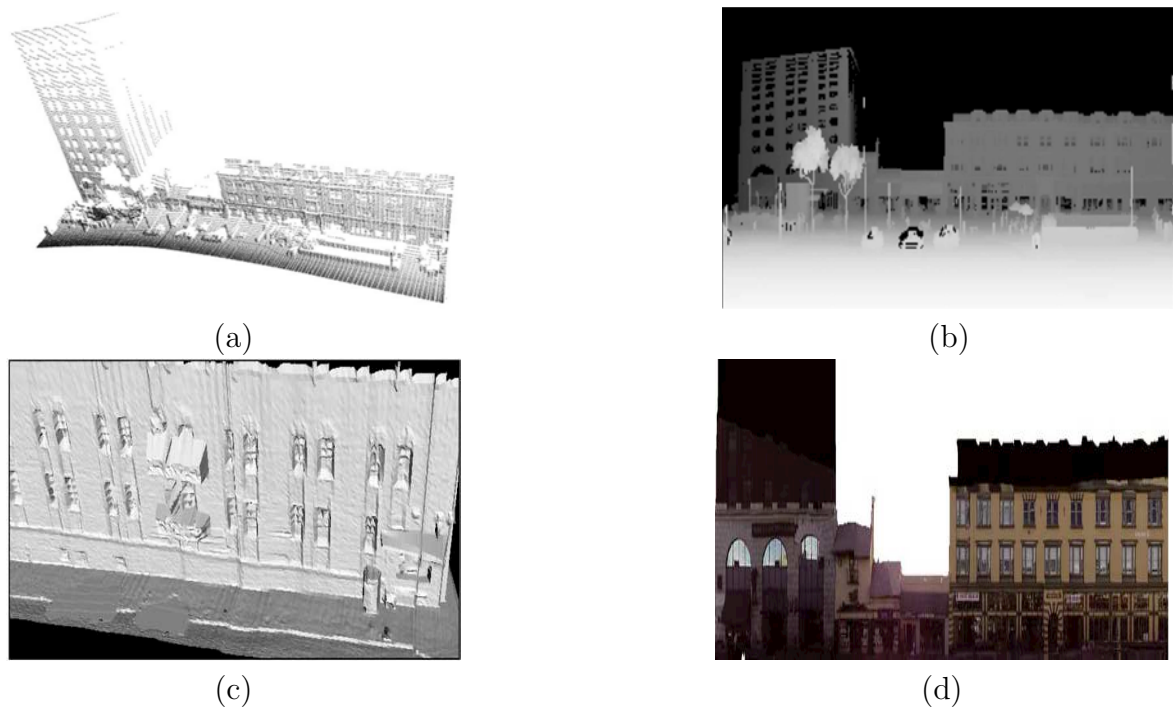


FIGURE 1.13 – Différentes étapes de la méthode de modélisation proposée dans [FJZ05] : (a) Nuage de points; (b) Image de profondeurs; (c) Modèle par maillage; (d) Modèle texturé



FIGURE 1.14 – Différentes étapes de la méthode de modélisation proposée dans [PPPD05] : (a) Image de façade; (b) Image de profondeurs; (c) Modèle par maillage; (d) Modèle texturé

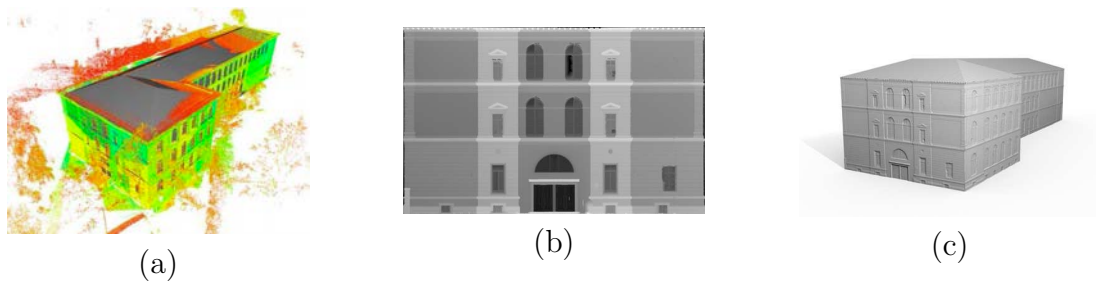


FIGURE 1.15 – Différentes étapes de modélisation [BBH07] : (a) Nuage de points et polygones recalés ; (b) Image raster, où l’intensité reflète la déviation de chaque point par rapport au plan ; (c) Modèle par maillage rectangulaire ;

Le modèle issu de ces méthodes de reconstruction permet de décrire les détails fins de la façade du bâtiment. Cependant ce type de modèle présente quelques inconvénients notamment par rapport au volume des données qui reste conséquent. Le paragraphe suivant présente des méthodes de reconstruction des modèles CAO plus compactes.

### 1.3.2 Méthodes de reconstruction par modèle paramétriques

La représentation du nuage de points par formes géométriques est beaucoup plus grossière que la représentation par maillage. Le modèle CAO est constitué d’un ensemble de formes géométriques représentant des sous-ensembles de points du nuage. Les méthodes de reconstruction basées sur ce principe sont généralement constituées de deux phases, la segmentation et la modélisation.

La segmentation vise à partitionner le nuage de points en des sous-ensembles consistants avec des formes géométriques préalablement définies. Elle se fait avec des algorithmes tels que la croissance de régions [Sch04], transformation de Hough [Hou62] et RANSAC [FB81]. Le modèle reconstruit définit l’objet par un ensemble de lignes et/ ou courbes caractéristiques qui représentent les intersections des surfaces extraites dans le nuage de points ou leurs bordures. Ce type de méthodes est généralement utilisé dans la reconstruction des milieux industriels et des bâtiments ayant une géométrie simple.

Dans la suite de ce paragraphe, nous nous sommes limités à quatre méthodes de reconstruction. D’autres méthodes de segmentation ont été proposées, mais ne vont pas jusqu’à la reconstruction complète du nuage de points.

[PV06] proposent une méthode de reconstruction de modèles 3D de bâtiments qui segmente dans un premier temps le nuage de points en différentes surfaces planes par la méthode de croissance de régions. La figure 1.16-b présente le résultat de la segmentation. Les plans issus de cette phase de segmentation sont classés selon leurs topologies en éléments architecturaux (fenêtres, toitures,...). Les points de contours sont ensuite extraits par la triangulation de Delaunay. Les points contour de chaque élément de façade sont alors assemblés en segments de droites par la transformation de Hough. Les intersections des segments résultants représentent le polygone définissant l’élément de façade. Le modèle volumique est ainsi reconstruit par projection des sommets sur le plan auquel ils sont agrégés en fonction de leurs classification sémantique.



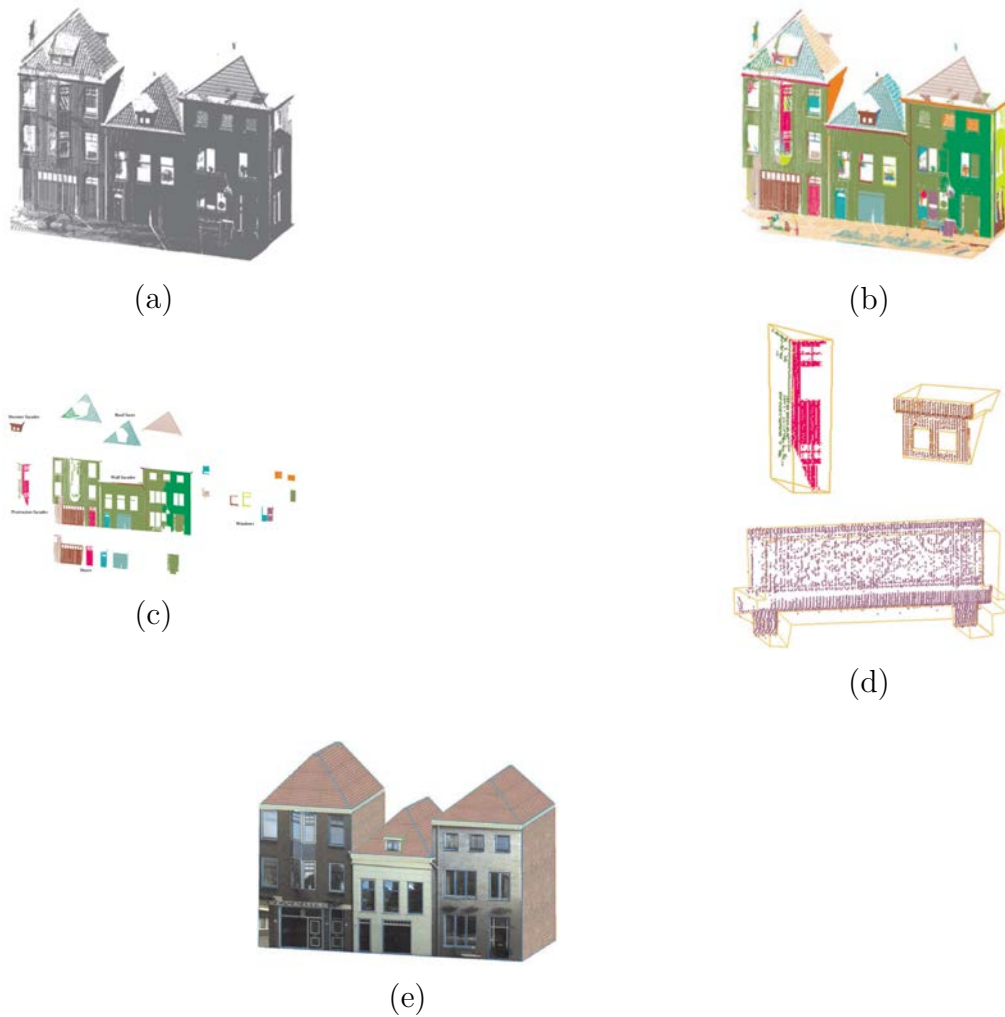


FIGURE 1.16 – Différentes étapes de la méthode de modélisation proposée dans [PV06] : (a) Nuage de points; (b) Nuage de points segmenté; (c) Classification sémantique des segments; (d) Polygones reconstruit pour les différents segments; (e) modèle CAO photo-réaliste

[HBK06] propose une méthode de reconstruction du modèle CAO à partir du nuage de points de scanner LASER terrestre et aéroporté et de plan cadastral. Dans un premier temps, un polygone représentant les façades du bâtiment est reconstruit par extrusion du plan cadastral. Les nuages de points des deux scanners -terrestre et aéroporté- sont ensuite recalés sur ce modèle (figure 1.17-a). Le nuage de points du SLT est segmenté par rapport aux facettes du polygone extrudé. Un point est dit consistant avec une facette si sa distance par rapport au plan de la facette est inférieure à un seuil préalablement défini. Les points de bordures de chaque ensemble de points consistants sont extraits en se basant sur la densité de points du voisinage. Une décomposition en cellule du plan de la façade est alors effectuée en se basant sur les points de contours. Les cellules sont ensuite classées en fonction de leur densité en cellule de façade ou cellule d'élément architectural (figure

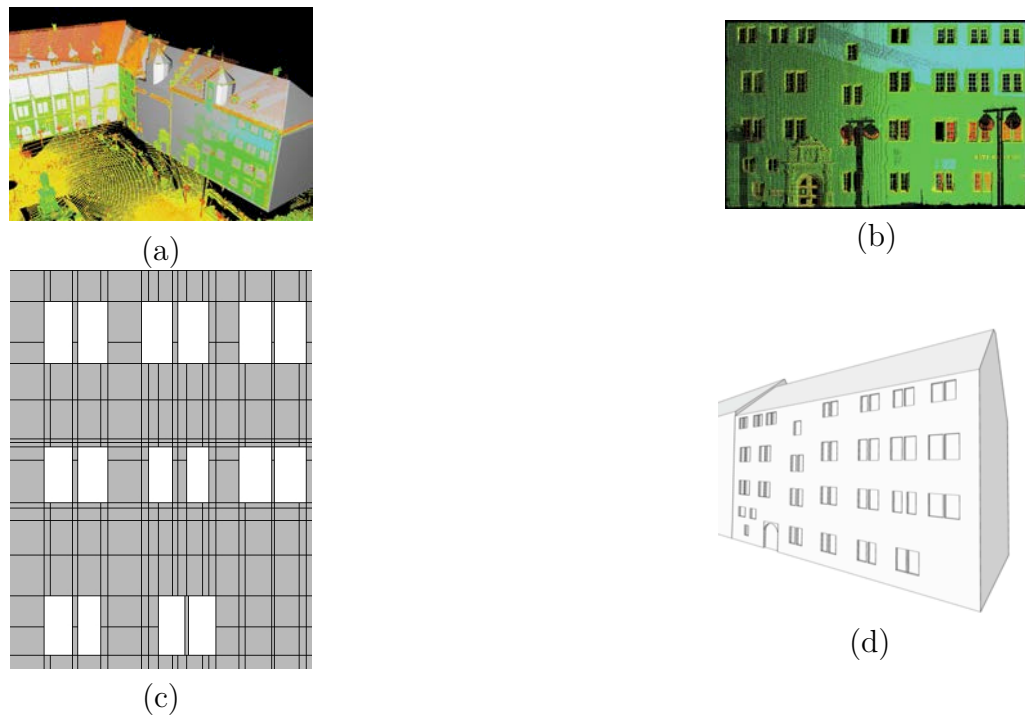


FIGURE 1.17 – Différentes étapes de la méthode de modélisation proposée dans [HBK06] : (a) Nuage de points et polygones recalés; (b) Ensemble consistant avec une façade; (c) Décomposition en cellule de polygones de la façade; (d) Modèle CAO après extrusion des cellules correspondants aux éléments de la façade.

1.17-c). Une intrusion est alors appliquée à la cellule représentant l'élément de façade. Un raffinement des fenêtres, utilisant des images photographiques est proposé dans [BH07]. La figure 1.17-d présente le modèle issu de cette méthode de reconstruction.

[Bou10] propose une méthode de reconstruction de modèle filaire de bâtiments à partir de nuage de points de SLT. La phase de segmentation est constituée de deux étapes. Dans un premier temps un plan est extrait par l'algorithme RANSAC. Le plan extrait représente une coupe dans le nuage de points (figure 1.18-a), ainsi l'ensemble des points consistants peut contenir des points appartenant à d'autres éléments architecturaux de la façade. Afin d'éviter ce défaut, une étape de validation est ajoutée. Elle consiste à définir des régions connexes dans l'ensemble des points consistants. Les régions de faibles surfaces et les points isolés sont réintégrés dans le nuage de points initial. La figure 1.18-b présente le plan segmenté. Ces deux étapes sont réitérées pour extraire les différents plans de la façade et/ou les différentes façades du bâtiment. Le résultat de cette phase est présenté dans la figure 1.18-c. La phase modélisation se compose quant à elle de trois étapes. Dans un premier temps, les points de contours de chaque plan sont extraits par triangulation de Delaunay. Ils sont ensuite segmentés en droites utilisant l'algorithme RANSAC (voir figures 1.18-d et e). Les points consistants avec chaque droite sont à leurs tours décomposés en segments dont les intersections permettent la reconstruction des polygones représentant les éléments de la façade. Le modèle filaire, présenté dans la



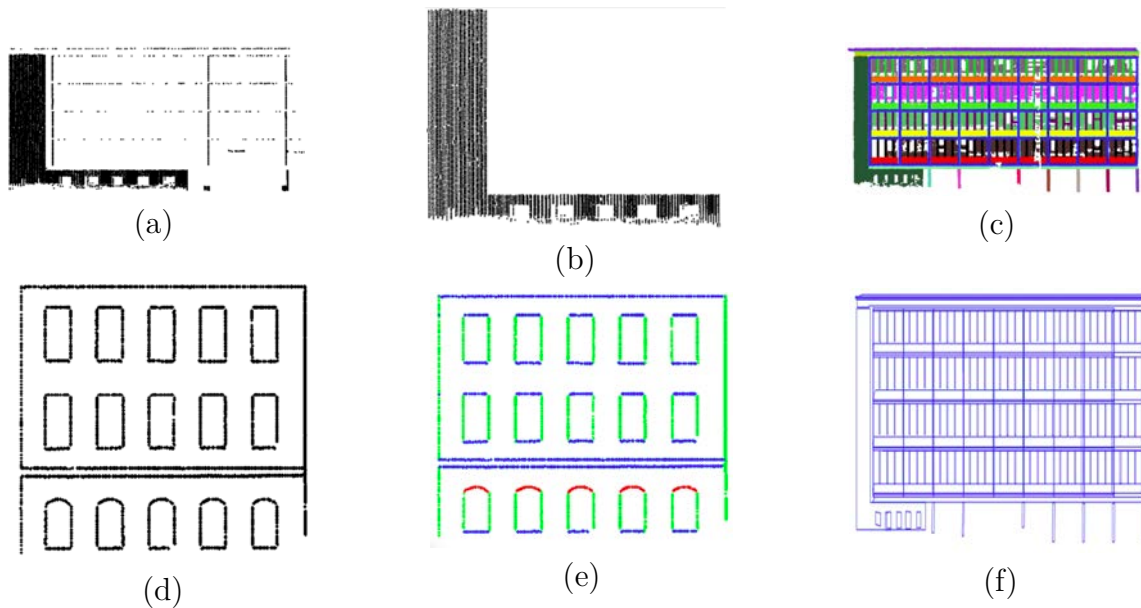


FIGURE 1.18 – Différentes étapes de la méthode de modélisation proposée dans [Bou10] : (a) Segment plan ; (b) Segment plan après raffinement ; (c) Résultat de la segmentation itérative ; (d) Points de bordures ; (e) Segment droit et arcs de bordures ; (f) modèle 3D filaire résultant de la méthode

figure 1.18-f, est reconstruit par extrusion des différents polygones dans la façade.

## 1.4 Conclusion

Ce chapitre permet d’avoir une vue d’ensemble sur le processus de reconstruction de modèle CAO de bâtiment à partir de nuage de points de scanner LASER terrestre.

La première partie de ce chapitre présente les différents principes de numérisation utilisés par les SLT ainsi que leurs domaines d’utilisation respectifs. Bien que les SLT permettent une numérisation précise, plusieurs facteurs induisent des erreurs de mesures. Ces facteurs peuvent être classés en deux types. D’une part les facteurs extrinsèques, dus aux conditions environnementales lors de l’acquisition tels que la réflectance de la surface, sa rugosité ainsi que le milieu de propagation de l’onde LASER. D’autre part les facteurs intrinsèques sont causés par des défaillances dans le SLT, tels que le bruit des capteurs, le décalage entre les axes des rotations. Afin de tenir compte de ces erreurs, plusieurs méthodes de calibration automatiques et semi-automatiques ont été développées, celle-ci permettent de définir un modèle géométrique du SLT permettant la correction de ces erreurs.

La seconde partie présente une synthèse des méthodes de recalage des nuages de points acquis par le SLT. Elles consistent à calculer une matrice de transformation permettant d’exprimer les coordonnées des points acquis dans différents référentiels propres aux sta-

tions dans un seul référentiel. Ce dernier contenant tous les nuages de points représentant l'objet numérisé. Ces méthodes se différencient dans les critères à minimiser, ainsi que les primitives géométriques (points, droites, plans ...) considérés pour le calcul de la matrice de transformation.

La dernière partie présente un panorama des méthodes de reconstruction de modèle CAO de bâtiment à partir de nuage de points de SLT. On retrouve d'une part les méthodes de reconstruction non-paramétriques dont l'avantage majeur est de mettre en avant les détails fins de la surface numérisée. Cependant, le volume de données reste conséquent ce qui rend la manipulation et l'interprétation du modèle difficile pour l'utilisateur. Un autre inconvénient, de ce type de représentation est l'impossibilité d'imbriquer des informations sémantiques sur le modèle, car aucune reconnaissance d'objets ou de formes n'est réalisée.

D'autre part, les méthodes de reconstruction paramétriques qui permettent de faire une reconstruction des éléments architecturaux par des primitives géométriques simples (droites, plans, cylindre ...) et donc de réduire le volume de données, ainsi que d'imbriquer des informations sémantiques sur le modèle CAO. Cependant, le résultat de modélisation reste généralement dépendant du paramétrage des algorithmes utilisés dans chaque phase, aussi cette reconstruction reste grossière, négligeant les fins détails de la surface.

Dans le domaine de la vision par ordinateur, le dilemme entre une reconstruction grossière et fine de la surface est toujours d'actualité [BBH07]. Pour le domaine de réhabilitation thermique de bâtiment par l'extérieur, la conception des modules d'isolation thermique se base sur les dimensions et les positions des éléments architecturaux. Ces modules sont rigides et difficilement modifiables, ainsi leur conception doit prendre en compte toutes les singularités de la surface sur laquelle ils seront fixés. Donc, le modèle CAO idéal doit représenter un compromis entre les deux approches, fines et grossières de la façade du bâtiment.



## 2

# Segmentation du nuage de points

Une façon très générale de décrire le nuage de points correspondant à un bâtiment est de le considérer comme la réunion d'ensembles de points appartenant à des plans qui vont représenter les différentes parties du bâtiment (façade, plan de toiture, terrasse, ...) [BBTB11]. Chaque ensemble de points 3D extrait appartient à un plan donné si sa distance par rapport au plan est nulle. En pratique, compte tenu des erreurs dues au bruit de mesure, aux erreurs de modèle, à la granularité de la façade, un point appartient à un plan donné si sa distance à ce plan est faible [BG Tk07]. Il s'agit donc de trouver le sous-ensemble de points de plus grande cardinalité dont la distance à un plan inconnu est inférieure à un seuil. Il s'agit en fait d'un problème d'optimisation combinatoire nécessitant  $C_N^3$  opérations où  $N$  est le nombre de points du nuage et 3 le nombre de points nécessaires pour constituer un plan. Dès que la taille du nuage de points est importante (ce qui est le cas dans notre application), la complexité de cette approche la rend totalement impossible à implanter et il devient nécessaire de recourir à des méthodes sous-optimales.

Le problème de segmentation du nuage de points peut être vu comme un problème de classification non-supervisée : l'objectif est de synthétiser les propriétés d'un sous-ensemble de points (position, orientation, couleurs, ...) par un modèle. Chaque point est affecté au modèle avec lequel il a le plus de similarités. Cette dernière est généralement mesurée par une distance entre le point et le modèle.

Dans ce chapitre, nous proposons une approche multimodale de segmentation. Celle-ci consiste à ajouter une phase de classification colorimétrique en amont de la phase segmentation par l'algorithme RANSAC. Le but étant de regrouper les points en des sous ensembles de points homogènes au niveau colorimétrique afin réduire la complexité algorithmique de RANSAC.

La première section présente les algorithmes de classification considérés.. La section 2 présente la méthode de segmentation proposée dont les résultats sont présentés dans la section 3.

## 2.1 Algorithmes de classification (clustering)

La classification consiste à partitionner l'ensemble de points en un ensemble de classes disjointes, où chaque classe contient un ensemble de points homogènes. Ainsi chaque

sous-ensemble de points peut alors être représenté par un modèle [XW<sup>+</sup>05]. Dans la littérature des méthodes de classification, on peut distinguer deux types de méthodes. La classification supervisée consiste à chercher parmi les classes déjà définies celle qui a le plus de similarité avec l'individu (point). Ces classes sont définies en amont du processus de classification par l'utilisateur ou à partir d'un processus d'apprentissage. Au contraire, la classification non-supervisée ne nécessite aucun apprentissage des classes, les différentes classes sont identifiées à partir de l'ensemble des données [Bou06].

Pour les deux types de méthodes -supervisée et non supervisée- la classification d'un point est faite en fonction de la mesure d'une distance par rapport au modèle représentant un sous-ensemble de points. Les paragraphes 2.1.1 et 2.1.2 présentent de manière générale les algorithmes de classification utilisés.

### 2.1.1 K-means

K-means est un algorithme itératif de classification, il consiste à partitionner un ensemble des points  $\mathbb{C}$  en des sous-ensembles homogènes, où chaque sous-ensemble définit une classe  $\mathbb{C}_j$  satisfaisant les contraintes suivantes :

1.  $\mathbb{C}_j \neq \emptyset$ ; avec,  $j = 1, \dots, k \leq n$ ;
2.  $\cup_{j=1}^k \mathbb{C}_j = \mathbb{C}$ ;
3.  $\mathbb{C}_j \cap \mathbb{C}_m = \emptyset, k, m = 1, \dots, k$ ; et  $l \neq m$ ;

L'algorithme K-means se compose de trois étapes principales :

- L'initialisation des classes, qui consiste à partitionner l'ensemble des points de manière aléatoire puis de calculer le centre de chaque classe par l'équation (2.1).

$$\mu_j = \frac{1}{|\mathbb{C}_j|} \sum_i^{|\mathbb{C}_j|} \mathbf{c}_i \quad (2.1)$$

- L'affectation des points, qui consiste à affecter chaque point à la classe dont le centre est le plus proche. Pour cela, l'algorithme calcul la distance de chaque point par rapport aux centres des classes. La métrique considérée dépend du domaine d'application et des caractéristiques discriminantes, ici nous considérons la distance Euclidienne (équation 2.2).

$$d_i = \|\mathbf{c}_i - \mu_j\|_2 \quad (2.2)$$

- L'actualisation des centres de classes, où chaque centre est calculé à partir des points affectés à la classe.

Les deux dernières étapes sont répétées jusqu'à ce que les classes ne changent plus, ou que le nombre d'itérations atteint par l'algorithme dépasse un nombre d'itérations fixé par l'utilisateur.

### 2.1.2 RANSAC

RANSAC est un algorithme d'estimation robuste qui a été introduit en 1981 par [FB81]. Il se compose essentiellement de trois étapes. Celles-ci seront décrites par rapport à la segmentation d'un ensemble de points  $\mathbb{P}$  en plusieurs sous-ensembles de points coplanaires.

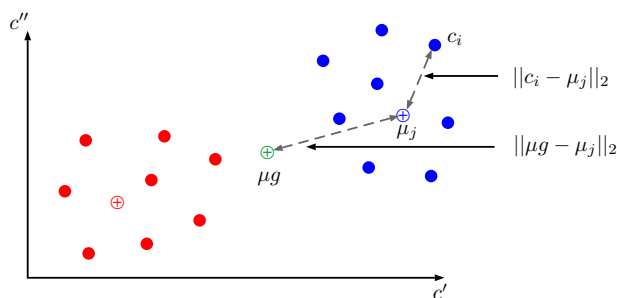


FIGURE 2.1 – Illustration des distances interclasse et intra-classe

1. **Proposition :** Dans cette étape, l'algorithme tire aléatoirement un échantillon  $\mathbb{E}$  (composé de trois points pour définir un plan) à partir de l'ensemble des données  $\mathbb{P}$ . Un modèle  $\mathbf{n}$  est alors estimé à partir de cet ensemble.
2. **Calcul de la fonction coût :** Une fonction coût est calculée, qui représente le critère que le modèle doit optimiser. En considérant que le critère représente le nombre de points consistants avec un plan, l'expression du modèle optimal peut être écrite comme suit :

$$\mathbf{n}_{opt} = \arg \max_{\mathbf{n}} \mathcal{J}(\mathbb{P}, \mathbf{n}) \quad (2.3)$$

3. **Classement du modèle :** Les modèles sont classés en fonction du nombre de points consistants avec chaque modèle. Le modèle maximisant la fonction coût  $\mathcal{J}(\mathbb{P}, \mathbf{n})$  est considéré comme le meilleur.

$$\mathcal{J}(\mathbb{P}, \mathbf{n}) = |\mathbb{P}_{in}| \quad (2.4)$$

où  $|\mathbb{P}_{in}|$  définit le cardinal de l'ensemble de points consistants.

Ces trois étapes sont itérées jusqu'à ce que la probabilité de trouver un meilleur modèle devienne faible. En général, cela revient à définir un nombre d'itérations  $\kappa$  à partir duquel on dit que le modèle le mieux classé est optimal.

Le nombre d'itérations est estimé en considérant  $q$  la probabilité de tirer un échantillon non-contaminé par des outliers (equation 2.5), et donc  $(1 - q)$  la probabilité de tirer un échantillon contaminé par au moins un outlier.

$$q = \frac{C_{N_I}^3}{C_N^3} \approx \left(\frac{N_I}{N}\right)^3 ; \quad (2.5)$$

où :

- $N_I$  proportion des inliers.
- $N$  cardinalité de l'ensemble  $\mathbb{P}$ .
- 3 cardinalité de l'ensemble  $\mathbb{E}$ .

Si on effectue  $\kappa$  tirages successifs, la probabilité que les échantillons soient tous contaminés par au moins un outlier est  $(1 - q)^\kappa$ . Ainsi, plus  $\kappa$  est grand plus cette quantité

tend vers zéro. On considère un seuil de probabilité  $\varepsilon_{th}$  à partir duquel on estime que l'échantillon minimal n'est pas contaminé :

$$(1 - q)^\kappa \leq \varepsilon_{th} \quad (2.6)$$

on est déduit le nombre d'itérations :

$$\kappa \simeq \left\lceil \frac{\log(\varepsilon_{th})}{\log(1 - q)} \right\rceil \quad (2.7)$$

Plusieurs modifications ont été apportées à cet algorithme de façon à diminuer sa complexité algorithmique. [Bou10] propose de calculer le nombre d'inliers ( $N_I$ ) de manière dynamique. Cela revient à attribuer à  $N_I$  la valeur maximale de la fonction coût (nombre de points consistants avec le plan le mieux classé). [CM08] propose d'évaluer la pertinence du modèle dans un premier test sur un sous ensemble de points défini aléatoirement, puis de calculer la fonction coût par rapport à tout l'ensemble, si le premier test est validé. Bien que cette méthode réduise considérablement la complexité algorithmique de RANSAC, elle augmente le risque de rejeter un meilleur modèle dès le premier test. [MTN<sup>+</sup>02] propose de réduire la complexité en effectuant une sélection de l'échantillon minimal de manière non uniforme. L'idée repose sur l'hypothèse que les inliers sont proches géométriquement, ainsi l'échantillon minimal est constitué à partir un ensemble de points voisins. En se basant sur ce principe, [SWK07] propose une méthode de segmentation de nuage de points où l'échantillon minimal est sélectionné à partir d'un ensemble de points dans un voxel d'une grille oc-tree. Dans cette approche, le nombre d'itérations a été considérablement réduit grâce à limitation du nombre de points présents dans le voxel  $N_{vo}$  (equation 2.8). Bien que le nombre d'itérations soit réduit par la sélection non-uniforme, le temps de calcul reste conséquent car il faut prévoir un partitionnement oc-tree en pré-traitement. De plus, le partitionnement ne garantit pas d'avoir une seule forme géométrique par voxel, ce qui augmente le risque d'une mauvaise segmentation.

$$p_{vo} = \frac{C_{N_I}^3}{C_{N_{vo}}^3} = \frac{N_I!(N_{vo} - 3)!}{N_{vo}!(N_I - 3)!}; \quad 3 < N_{vo} < N; \quad (2.8)$$

Les méthodes de classification non-supervisées se distinguent par leur robustesse d'estimation sur des données entachées de valeurs aberrantes. Cependant la complexité en terme du temps de calcul de ces algorithmes reste conséquente.

Ce chapitre présente une méthode de classification géométrique se basant sur l'algorithme RANSAC. Sa contribution majeure est la réduction de la complexité algorithmique par l'intégration des données colorimétriques du nuage de points. La solution proposée consiste à sélectionner l'échantillon minimal à partir d'une seule classe colorimétrique, qui représente un sous-ensemble de points ayant des caractéristiques colorimétriques similaires. Cette approche augmente la probabilité de sélectionner un échantillon minimal appartenant à la même forme géométrique, entraînant ainsi la diminution du temps de calcul par l'intégration des classes colorimétriques dans l'estimation du nombre d'itérations.

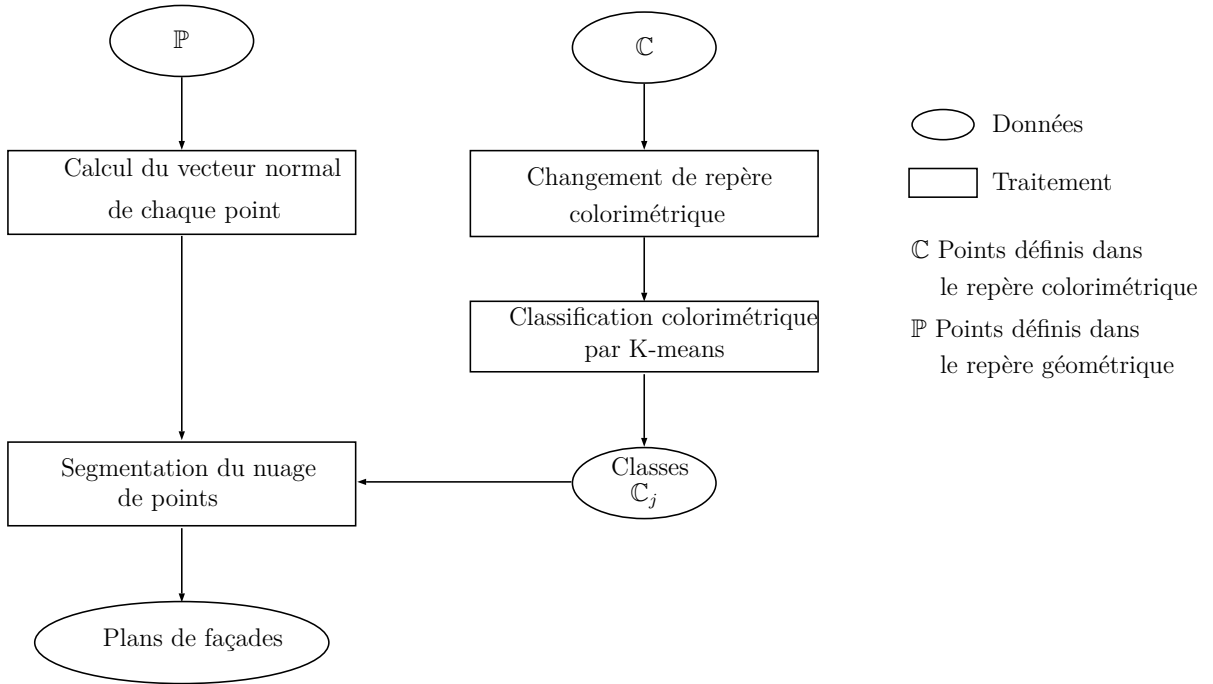


FIGURE 2.2 – Méthode de segmentation proposée.

## 2.2 Prise en compte des données colorimétriques pour la segmentation du nuage de points.

Cette section propose une méthode de segmentation géométrique du nuage de points basée sur l’algorithme RANSAC. Son originalité majeure consiste à réduire la complexité algorithmique de RANSAC sans pour autant diminuer sa robustesse. Pour cela, une classification colorimétrique en amont de la classification géométrique est introduite. Le but étant de faire une sélection efficace de l’échantillon minimal ainsi que de réduire le nombre d’itérations.

La figure 2.2 présente les différentes étapes de la méthode proposée, qui sont décrites dans le paragraphe suivant.

### 2.2.1 Segmentation colorimétrique du nuage de points

#### Choix du repère colorimétrique

L’algorithme K-means vise à minimiser l’inertie intraclasse ( $\mathcal{I}_{intra}$ ) qui est une mesure de la dispersion des points de la classe par rapport à son centre. Plus l’inertie intraclasse est faible plus la classe est compacte donc plus homogène. Un autre critère qualitatif qui peut être considéré est l’inertie interclasses ( $\mathcal{I}_{inter}$ ) qui est une mesure de la séparation entre les classes [XW<sup>+</sup>05]. Elle représente la somme des distances entre les classes et le centre de gravité du nuage de points. La figure 2.1 présente une illustration des distances



à maximiser/minimiser pour une classification optimale.

$$\mathcal{I}_{intra} = \frac{1}{n} \sum_{j=1}^k \sum_{i \in \mathbb{C}_j} \|\mathbf{c}_i - \mu_j\|_2 \quad (2.9)$$

$$\mathcal{I}_{inter} = \frac{1}{n} \sum_{j=1}^k \|\mu_j - \mu_g\|_2 \quad (2.10)$$

La qualité d'une classification dépend en grande partie des caractéristiques discriminantes utilisées, qui doivent être choisies de façon à permettre la meilleure distinction entre les classes [XW<sup>+</sup>05]. Dans cette optique, un plan d'expérience a été mené pour choisir les variables du vecteur colorimétrique  $\mathbf{c}_i = [il_i, r_i, v_i, b_i]$  minimisant l'inertie intraclasse.

La couleur attribuée à chaque point est définie dans le repère *RVB*. Cependant les composantes colorimétriques de ce repère ne sont pas indépendantes, ce qui réduit considérablement la distinction entre les éléments des classes [Kow90]. Pour ces raisons, la Commission Internationale d'Eclairage (CIE) préconise d'utiliser d'autres repères pour une comparaison de couleurs : CIEXYZ, CIELab CIELuv et CIETS. Les trois derniers repères ont la particularité d'être uniformes, autrement dit, des couleurs similaires pour un individu sont très proches dans l'espace de représentation [Sch07a].

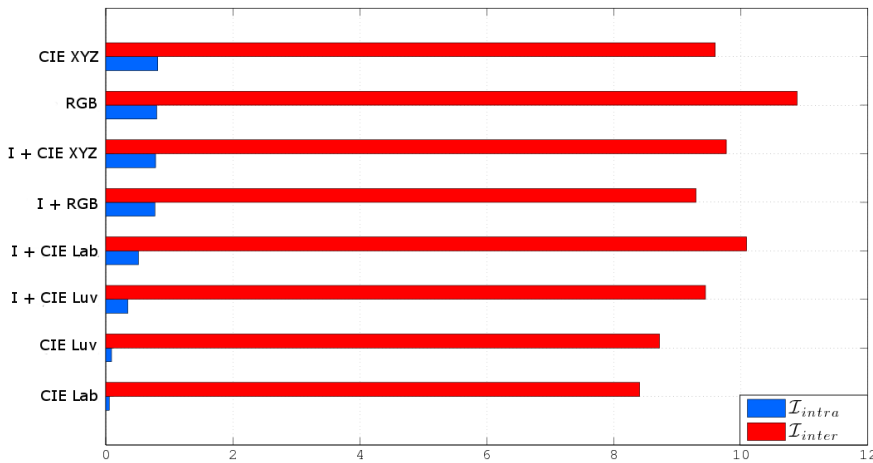


FIGURE 2.3 – Résultat montrant les inerties intra et inter classes obtenues avec différentes combinaisons de caractéristiques radio-métriques

Les tests ont été menés sur deux niveaux pour chaque repère colorimétrique. Le premier consiste à prendre en compte l'intensité du LASER réfléchi par la surface. Le second niveau se limite à la couleur comme critère de comparaison. Afin de palier à l'aspect aléatoire de l'algorithme K-means, la classification a été menée plusieurs fois sur chaque ensemble de données. Ainsi, on considère les moyennes calculées sur 10 essais pour chaque vecteur caractéristique comme indicateur de performance. La figure 2.3 présente les résultats de classification en terme d'inerties. On constate que la limitation des caractéristiques

2.2. *Prise en compte des données colorimétriques pour la segmentation du nuage de points.*

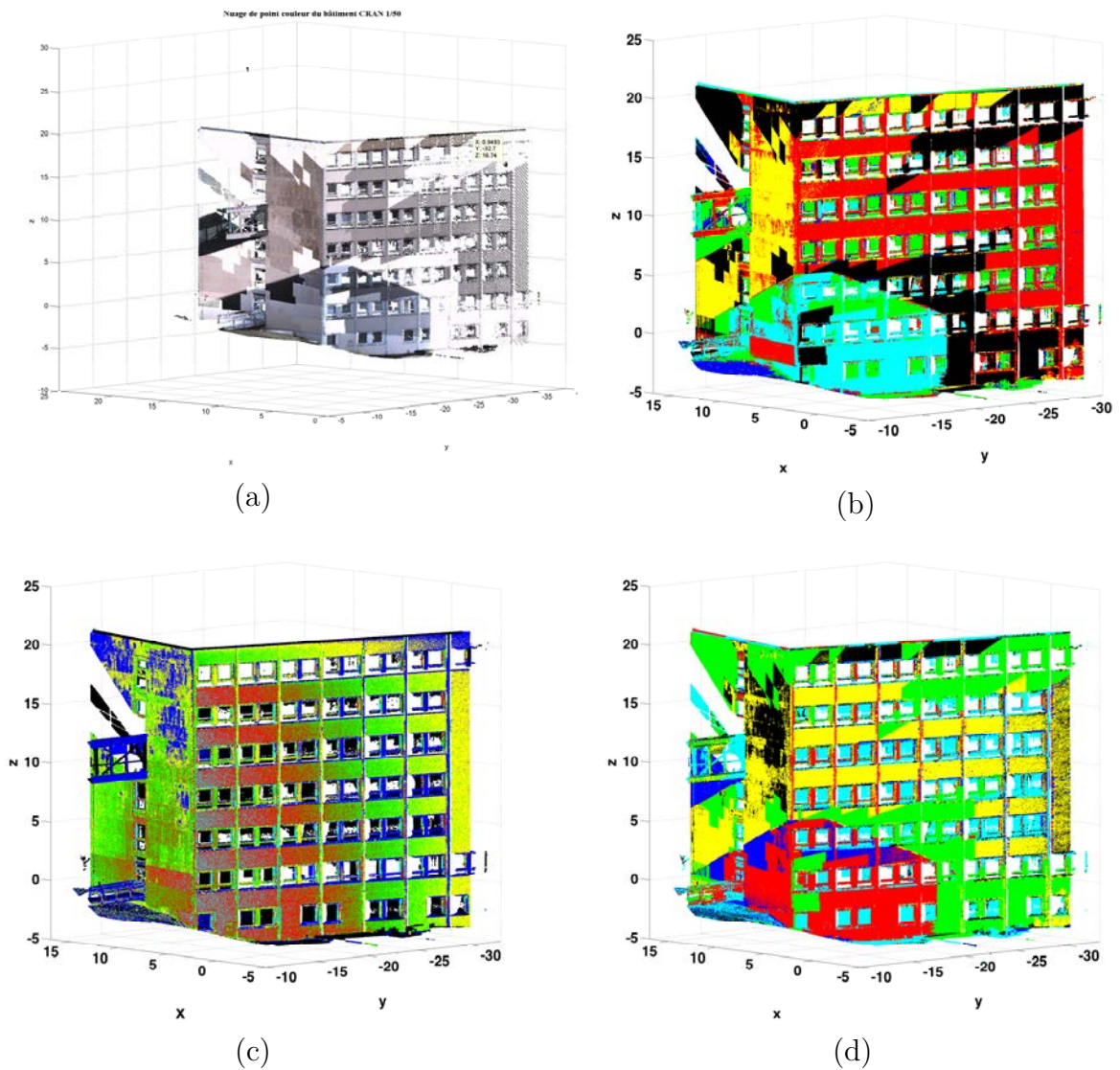


FIGURE 2.4 – Interpolation des résultats de classification sur le nuage de points dans le repère Cartésien : (a) nuage de points du bâtiment ; (b) classification utilisant le sous ensemble des caractéristiques  $[L, a, b]$  ; (c) classification n'utilisant que l'intensité LASER  $il$  ; (d) classification utilisant l'ensemble de caractéristiques  $[il, L, a, b]$

discriminantes aux variables colorimétriques permet une bonne séparabilité des classes, notamment dans le repère CIELab où on a une faible inertie intraclasse.

La prise en compte de l'intensité LASER ne semble pas être intéressante car :

- L'intensité LASER varie selon plusieurs critères : distance, angle d'incidence et caractéristiques de la surface (couleur, rugosité, matériau). La figure 2.4-c présente le résultat d'une classification du nuage de points prenant l'intensité LASER comme seule caractéristique discriminante. Contrairement à la classification par couleurs où les limites de classes sont strictes, celles-ci restent floues : la densité d'une classe décroît alors que celle de la classe adjacente s'amplifie.
- La distance entre deux points dans un espace de dimension quatre (cas où la classification prend en compte l'intensité LASER) est supérieure à leur distance dans une dimension inférieure.

Même si la qualité des couleurs affectées aux points reste à améliorer dans les SLT, celle-ci permet à terme un regroupement des points en des sous-ensembles homogènes.

### Algorithme de classification

La classification colorimétrique consiste à partitionner l'ensemble de points dans des sous-ensembles ayant des caractéristiques colorimétriques proches. Cette classification est réalisée en utilisant l'algorithme des K-means. Ici on considère le repère CIELab, défini suite à l'étude détaillée dans la section 2.2.1. Ainsi, chaque point est détaillé par trois composantes  $\mathbf{c}_i = [L_i, a_i, b_i]$ . L'algorithme de classification est défini dans l'algorithme 1. Il est à noter qu'on considère un nombre d'itérations maximal (`maxIteration`) qui est un paramètre défini par l'utilisateur. Celui-ci permet d'arrêter l'algorithme au cas où il n'arrive pas à converger vers une solution. Aussi, on définit  $\mathbb{P}'$  un sous-ensemble de points coplanaires exprimés dans le référentiel géométrique et  $\mathbb{C}'$  son homologue dans le référentiel colorimétrique.

$$\mathbb{C}' = \{\mathbf{c}_i | \mathbf{p}_i \in \mathbb{P}'\}.$$

De façon similaire,  $\mathbb{P}'$  est l'homologue dans le référentiel géométrique d'une classe  $\mathbb{C} \in \mathbb{C}$

$$\mathbb{P}' = \{\mathbf{p}_i | \mathbf{c}_i \in \mathbb{C}\}.$$

```

Entrées :
 $\mathbb{C} = \{\mathbf{c}_1, \dots, \mathbf{c}_N\}$  (nuage de points définis dans l'espace colorimétrique CIELab)
 $k$  (nombre de classes)
maxIteration (nombre d'itérations maximal)
Sorties :
 $\{\mathbb{C}_1, \dots, \mathbb{C}_k\}$  (Classes colorimétriques)

changer : bool
iteration  $\leftarrow 0$  : entier

Pour j de 1 à k faire
    |  $\mu_j \leftarrow \mathbf{c}_j$  (Initialisation des centres de classes)
Fin Pour

Pour i de 1 à  $|\mathbb{C}|$  faire
    |  $l_i \leftarrow \arg \min_j \text{dist}(\mathbf{c}_i, \mu_j)$ 
Fin Pour

Répéter
    changer  $\leftarrow$  'Faux'
    Pour j de 1 à k faire
        | Actualiser( $\mu_j$ )
    Fin Pour
    Pour i de 1 à  $|\mathbb{C}|$  faire
        |  $index \leftarrow \arg \min_j \text{dist}(\mathbf{c}_i, \mu_j)$ 
        Si ( $index \neq l_i$ ) Alors
            |  $l_i \leftarrow index$ 
            | changer  $\leftarrow$  'Vrai'
        Fin Si
    Fin Pour
    iteration  $\leftarrow$  iteration +1

jusqu'à ce que (changer = 'Faux' OU iteration > maxIteration )

Pour j de 1 à k faire
    |  $\mathbb{C}_j \leftarrow \{\forall \mathbf{c}_i \in \mathbb{C} | l_i = j\}$  (Création des ensembles de points)
Fin Pour

```

Algorithme 1 – Algorithme de classification colorimétrique par K-means

## 2.2.2 Intégration dans l'algorithme RANSAC

### Calcul du vecteur normal

Le vecteur normal  $\mathbf{n}_i$  de la surface en un point  $\mathbf{p}_i$  est déduit par l'analyse de covariance d'un ensemble de points voisins à ce point [MNG04]. En effet, les vecteurs propres ( $\vec{e}_1, \vec{e}_2, \vec{e}_3$ ) ainsi que les valeurs propres ( $\lambda_1, \lambda_2, \lambda_3$ ) définissent l'ellipsoïde de corrélation,

où les vecteurs propres représentent les directions principales de la surface en ce point, et les valeurs propres leurs amplitudes (voir figure 2.5). Le vecteur normal à la surface en ce point correspond au vecteur propre ayant la valeur propre minimale.

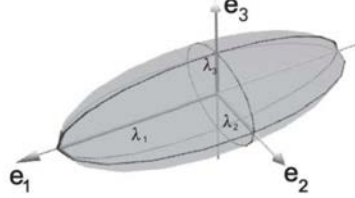


FIGURE 2.5 – Analyse de covariance d'un ensemble de points voisins

### Algorithme de segmentation proposé

Afin de prendre en compte les classes colorimétriques, des modifications ont été ajoutées à l'algorithme RANSAC :

- **Proposition :** Pour d'augmenter la probabilité de prendre un échantillon minimal non-contaminé par des outliers, une contrainte sur sa sélection a été introduite. Celle-ci exige que les trois points de échantillon minimal soient extraits de la même classe colorimétrique (équation 2.11). Durant la première itération l'échantillon minimal est tiré à partir de l'ensemble  $\mathbb{P}$ . Puis, dans les itérations suivantes, celui-ci est défini à partir de l'ensemble des points  $\mathbb{P}'$  :

$$\mathbb{E} = \{\mathbf{p}_a | \mathbf{p}_a \in \mathbb{P}'\}; \quad (2.11)$$

avec  $\mathbb{P}'$  ensemble de points dans  $\mathbb{P}$  tel que :

$$\mathbb{P}' = \{\mathbf{p}_i | \mathbf{c}_i \in \mathbb{C}_{opt}\};$$

où  $\mathbb{C}_{opt}$  représente la classe majoritairement représentée dans l'ensemble de points consistants avec le meilleur modèle (voir figure 2.7). Le vecteur normal est ensuite déduit en se référant aux travaux de [Bou10]. Les trois points constituant l'ensemble  $\mathbb{E}$  sont coplanaires. Pour qu'un point  $\mathbf{p}$  appartienne au même plan, il est nécessaire que le déterminant des trois vecteurs  $(\mathbf{p} - \mathbf{p}_1, \mathbf{p} - \mathbf{p}_2, \mathbf{p} - \mathbf{p}_3)$  soit nul (équation 2.12) :

$$\det \begin{pmatrix} \mathbf{p} - \mathbf{p}_1 & \mathbf{p} - \mathbf{p}_2 & \mathbf{p} - \mathbf{p}_3 \end{pmatrix} = \det \begin{pmatrix} x - x_1 & x_2 - x_1 & x_3 - x_1 \\ y - y_1 & y_2 - y_1 & y_3 - y_1 \\ z - z_1 & z_2 - z_1 & z_3 - z_1 \end{pmatrix} = 0 \quad (2.12)$$

d'où on peut déduire les paramètres du plan :

$$n_1 = \det \begin{pmatrix} y_2 - y_1 & y_3 - y_1 \\ z_2 - z_1 & z_3 - z_1 \end{pmatrix}$$

$$n_2 = \det \begin{pmatrix} x_2 - x_1 & x_3 - x_1 \\ z_2 - z_1 & z_3 - z_1 \end{pmatrix}$$

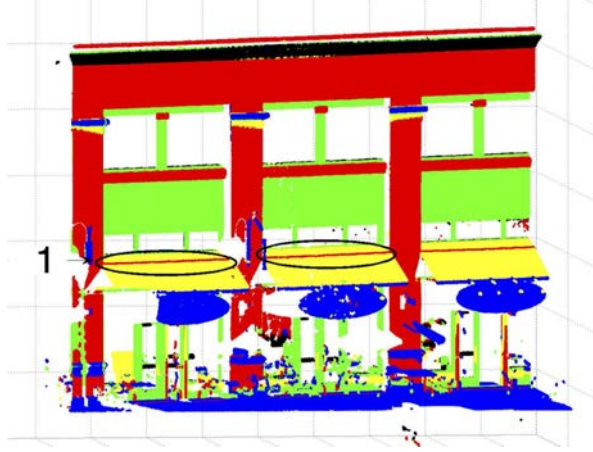


FIGURE 2.6 – Exemple d’une mauvaise segmentation : Régions 1 doivent appartenir au plan des stores (Jaune).

$$n_3 = \det \begin{pmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{pmatrix}$$

$$d = -a.x_1 - b.y_1 - c.z_1$$

avec  $[n_1, n_2, n_3]$  représentent le vecteur normal au plan et  $d$  sa distance par rapport à l’origine du repère.

- **Recherche de l’ensemble consistant** : la considération de la distance  $dist(\mathbf{p}_i, \mathbf{n})$  comme seul critère de consistance induit à une fausse segmentation. En effet, les plans issus d’une segmentation n’utilisant que la distance représentent des coupes du nuage de points. Des points appartenant à d’autres éléments architecturaux peuvent ainsi retenus comme points consistants (voir la figure 2.6). Cet inconvénient peut être évité en prenant en compte l’orientation du vecteur normal dans la segmentation du nuage de points. L’ensemble de points consistants avec le plan peut être défini comme l’intersection de deux ensembles (équation 2.13). Le premier  $\mathbb{P}_\beta$  représente l’ensemble de points ayant la même orientation que celle du plan proposé (équation 2.14). Le second  $\mathbb{P}_\delta$  représenté dans l’équation (2.15) définit l’ensemble de points dont la distance au plan est inférieure au seuil de consistance (équation 2.16).

$$\mathbb{P}_{in} = \mathbb{P}_\beta \cap \mathbb{P}_\delta \quad (2.13)$$

$$\mathbb{P}_\beta = \{\forall \mathbf{p}_i \in \mathbb{P} \mid \arccos(\langle \mathbf{n}_i, \mathbf{n} \rangle) < \beta\} \quad (2.14)$$

$$\mathbb{P}_\delta = \{\forall \mathbf{p}_i \in \mathbb{P} \mid d_i \leq \delta\} \quad (2.15)$$

où

$$d_i = \frac{|ax_i + by_i + cz_i + d|}{\sqrt{a^2 + b^2 + c^2}} \quad (2.16)$$

- **Classement du modèle** : Le modèle est classé en fonction du nombre de points consistants avec le modèle. Le modèle maximisant la fonction coût ( $\mathcal{J}(\mathbb{P}, \mathbf{n})$ ) est considéré comme le meilleur modèle tel que :

$$\mathbf{n}_{opt} = \arg \max_{\mathbf{n}} (\mathcal{J}(\mathbb{P}, \mathbf{n})) \quad (2.17)$$

où :

$$\mathcal{J}(\mathbb{P}, \mathbf{n}) = |\mathbb{P}_{in}| \quad (2.18)$$

- **Nombre d'itérations** : Le nombre d'itération  $\kappa'$  est défini prenant en compte les classes colorimétriques, en se référant la section 2.1.2 :

$$\kappa' = \frac{\log(\varepsilon_{th})}{\log(1 - q')} \quad (2.19)$$

où :

$$q' = \left( \frac{N_{Ic}}{N_c} \right)^3; \quad (2.20)$$

$N_{Ic}$  représente la cardinalité de l'ensemble de points appartenant à la classe majoritairement représentée  $\mathbb{C}_{opt}$  (voir figure 2.7) dans l'ensemble de points consistants :

$$N_{Ic} = |\mathbb{C}_{opt} \cap \mathbb{C}'|; \quad (2.21)$$

avec :

$$\mathbb{C}_{opt} = \arg \max_{\mathbb{C}} (\mathbb{C}_j \cap \mathbb{C}'); \quad j = 1, \dots, k; \quad (2.22)$$

où  $\mathbb{C}'$  représente l'ensemble de points consistants avec le meilleur modèle défini par leur composante colorimétrique.

$$\mathbb{C}' = \{\forall \mathbf{c}_i \in \mathbb{C} | \mathbf{p}_i \in \mathbb{P}_{in}\}$$

$$N_c = |\mathbb{C}_{opt}| \quad (2.23)$$

Cette phase de segmentation est suivie d'une étape de raffinement où le vecteur normal du plan est ré-estimé à partir de l'ensemble de points consistants. Le vecteur normal est défini par une analyse en composantes principales de l'ensemble  $\mathbb{P}_{in}$ . Ce processus de raffinement affecte aussi l'ensemble des points consistants, où de nouveaux points deviennent consistants et d'autres sont rejetés. Une présentation explicite de cette phase est donnée dans l'algorithme 2.

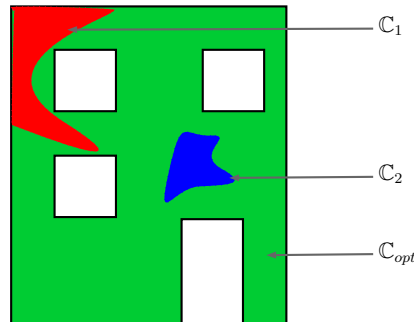


FIGURE 2.7 – Illustration d'un plan de façade composé de trois couleurs

**Entrées :** $\mathbb{P}$  (Ensemble de points dans le nuage) $\mathbb{P}_{in}$  (Ensemble de points consistants) $\delta$  (seuil de consistance) $\beta$  (seuil de consistance angulaire)**Sorties :** $\mathbb{P}_r$  (Ensemble de points consistants après raffinement) $\mathbf{n}_r$  (Vecteur normal après raffinement)**Algorithme :**changer : **bool** $\mathbb{P}_1 = \mathbb{P}_{in}$ changer  $\leftarrow$  'Vrai't  $\leftarrow$  2**Répéter** $\mathbf{n}_t \leftarrow \text{PCA}(\mathbb{P}_{t-1})$  $\mathbb{P}'_\beta \leftarrow \{\forall \mathbf{p}_i \in \mathbb{P} \mid \arccos(\langle \mathbf{n}_i, \mathbf{n}_t \rangle) < \beta\}$  $\mathbb{P}'_\delta \leftarrow \{\forall \mathbf{p}_i \in \mathbb{P} \mid \|\langle \mathbf{p}_i, \mathbf{n}_t \rangle \cdot \mathbf{n}_t\| \leq \delta\}$  $\mathbb{P}_t \leftarrow \mathbb{P}'_\beta \cap \mathbb{P}'_\delta$ **Si** ( $\mathbb{P}_t \setminus \mathbb{P}_{t-1} = \emptyset$ ) **Alors**| changer  $\leftarrow$  'Faux'**Fin Si**t  $\leftarrow$  t+1**jusqu'à ce que** (changer = 'Faux') $\mathbb{P}_r \leftarrow \mathbb{P}_t$  $\mathbf{n}_r \leftarrow \mathbf{n}_t$ 

Algorithme 2 – Algorithme de raffinement de la segmentation.

## 2.3 Évaluation des performances

Le premier paragraphe présente une évaluation du temps de calcul dans lequel une comparaison est réalisée par rapport aux performances de la méthode de segmentation par RANSAC. Le second paragraphe présente une évaluation qualitative des résultats en fonction des paramètres de consistances.

Ces évaluations ont été réalisées sur un nuage de points de bâtiment acquis à partir de six stations (figure 3.11). Le tableau 3.1 présente les caractéristiques de ce nuage de points ainsi que les paramètres d'acquisition utilisés.



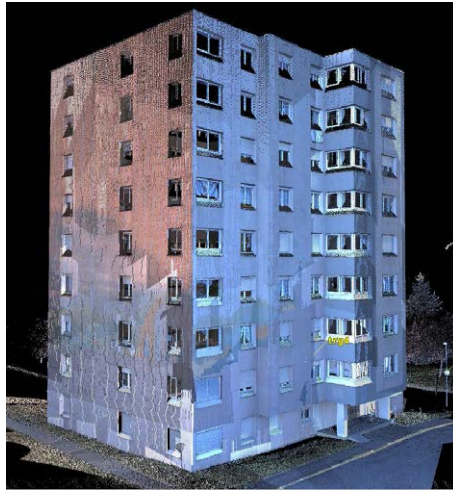


FIGURE 2.8 – Building point cloud

<b>Paramètres de numérisation</b>	
Pas d'échantillonnage angulaire	$5 \cdot 10^{-4}$ rad
Pas d'échantillonnage de la surface	0.05 m
Distance nominal ( $d_n$ )	100 m
Nombre de stations	6
Nombre de points total	36.325.675 points

TABLE 2.1 – Paramètres de numérisation du bâtiment.

### 2.3.1 Evaluation du temps de calcul

Pour évaluer l'apport de la classification colorimétrique, une comparaison des performances de la méthode utilisant uniquement la classification géométrique par RANSAC et celle utilisant la classification colorimétrique a été réalisée. On considère les valeurs de paramètres de classification géométrique : seuil de distance  $\delta = 2cm$ , seuil d'orientation des vecteurs normaux  $\beta = 40^\circ$  et  $k = 6$  nombre de classes colorimétriques.

La figure 2.10 présente les courbes de convergence des deux méthodes pour les cinq premiers plans extraits (voir figure 2.9), où la courbe bleue représente la convergence de la solution proposée. Pour les quatre premiers plans, on constate que la solution utilisant la classification colorimétrique converge plus rapidement. Aussi, le nombre d'itérations est faible par rapport à la méthode de classification utilisant seulement RANSAC. Cet écart décroît au fur et à mesure que des plans sont retirés du nuage de points. En effet, plus des plans sont retirés du nuage de points, moins les classes sont compactes et contiennent plus d'échantillons minimaux contaminés. La figure 2.10-e présente le cas où la solution proposée converge moins rapidement que la solution classique. Ainsi, l'apport de la colorimétrie décroît au fur et à mesure qu'on retire des plans de façade. Le tableau 2.2 présente les résultats de cette comparaison de manière quantitative en terme de nombre de points consistants, nombre d'itérations et l'erreur quadratique moyenne. Il est à noter que l'ordre d'extraction est différent pour les deux solutions. En effet, pour la méthode proposée, l'extraction des plans est conditionnée par les cardinalités des classes

colorimétriques présentes dans l'ensemble de points consistant avec le meilleur modèle.

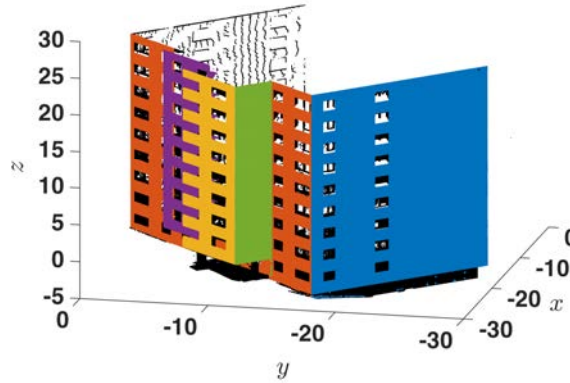


FIGURE 2.9 – Image des cinq premiers plans extraits.

méthode géométrique				méthode colorimétrique + géométrique			
Plan	$ \mathbb{P}_{in} $	$\kappa$	MSE ( $10^9$ )	Plan	$ \mathbb{P}_{in} $	$\kappa'$	MSE ( $10^9$ )
1	8.695.93	2603	2,64	1	8.695.54	956	2,71
2	2.769.70	35.65	5,33	4	2.747.23	16.36	8,18
3	2.314.52	43.01	9,76	2	2.313.68	4527	9,70
4	2.198.12	38.63	2,25	3	2.198.16	3016	2,17
5	1.970.44	39.61	3,70	5	1.970.46	2072	3,69

TABLE 2.2 – Comparaison des résultats de classification pour les deux méthodes en terme de nombre de points consistant ( $|\mathbb{P}_{in}|$ ), nombre d'itérations ( $\kappa$ ;  $\kappa'$ ) et erreur quadratique moyenne (MSE)

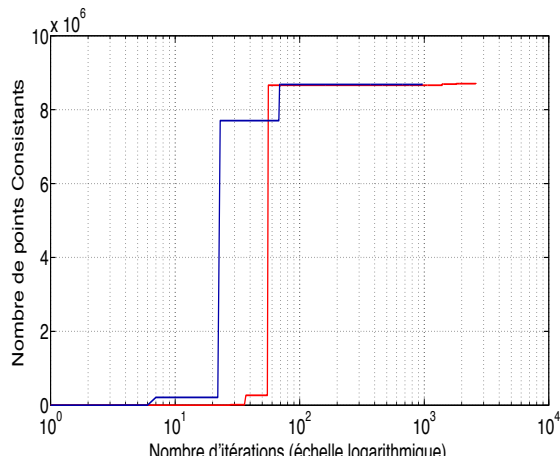
**Interprétation analytique des résultats** Le temps de calcul de RANSAC dépend de deux facteurs. Le premier est le nombre d'itérations suffisant pour avoir une solution considérée comme optimale. Le second est le temps nécessaire pour évaluer la qualité du modèle élaboré [CM08]. Ainsi, le temps nécessaire pour trouver une solution optimale peut être écrit comme :

$$t = \kappa (t_n + t_J \cdot N) \quad (2.24)$$

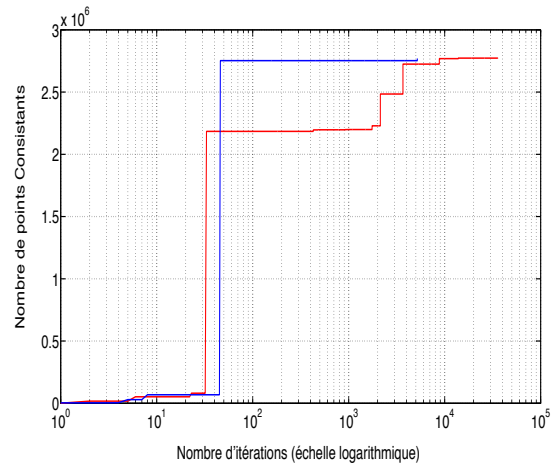
avec

- $\kappa$  : Nombre d'itérations (équation 2.7).
- $t_n$  : Temps nécessaire pour élaborer un modèle. C'est le temps généralement requis pour la sélection d'un EM ainsi que le calcul du vecteur normal.
- $t_J$  : Temps nécessaire pour évaluer la consistance d'un point au modèle concerné.
- $N$  : Nombre de points dans le nuage.

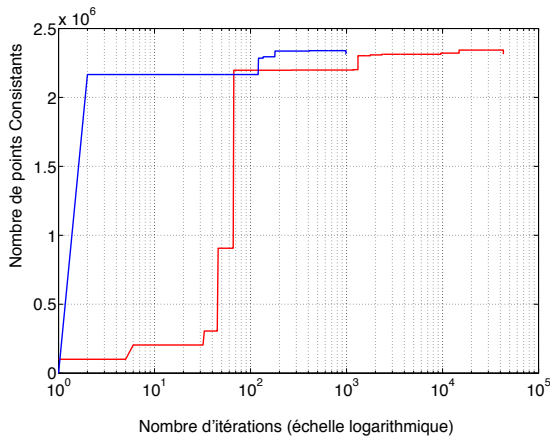
La réduction du nombre d'itérations par la prise en compte de la colorimétrie permet de diminuer le temps nécessaire pour avoir une solution optimale. Ce paragraphe présente de



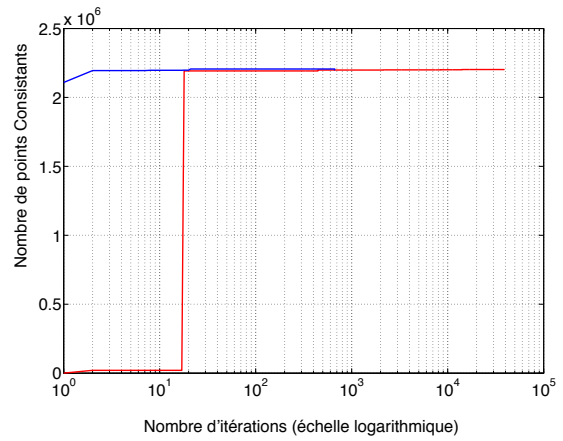
(a) Extraction plan 1.



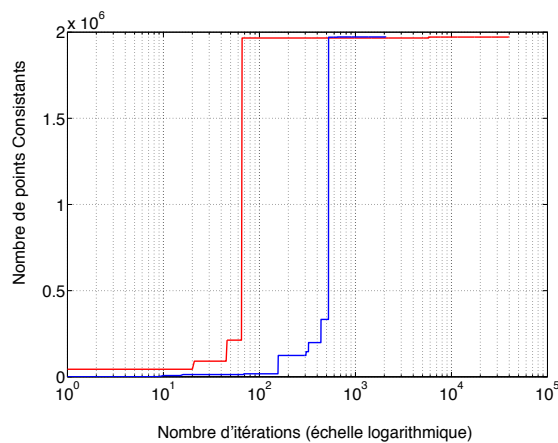
(b) Extraction plan 2.



(c) Extraction plan 3.



(d) Extraction plan 4.



(e) Extraction plan 5

FIGURE 2.10 – Courbes de convergence de l'algorithme RANSAC : nombre de points consistants par rapport au nombre d'itérations (la courbe bleue représente la convergence de la méthode de classification proposée).

manière analytique l'apport de la classification colorimétrique en amont de la classification géométrique.

On considère  $\mathbb{P}$  un ensemble de points composé de deux sous-ensembles, le premier  $\mathbb{P}_{in}$  représente un ensemble de points consistants avec un plan, le second  $\mathbb{P}_{out}$  représente l'ensemble de points aberrants ou appartenant à d'autres plans de façade (figure 2.11) :

$$\mathbb{P} = \mathbb{P}_{in} \cup \mathbb{P}_{out};$$

avec :

$$\mathbb{P}_{in} \cap \mathbb{P}_{out} = \emptyset;$$

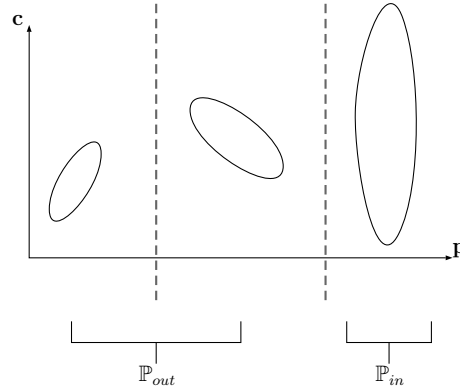


FIGURE 2.11 – Illustration 2D de l'ensemble de points partitionné en sous-ensembles.  $\mathbf{c}$  représente la composante colorimétrique et  $\mathbf{p}$  la composante géométrique.

Cette configuration est celle considérée par l'algorithme RANSAC, définissant ainsi le nombre d'itérations par :

$$\kappa = \frac{\log(\varepsilon_{th})}{\log(1 - q)} \quad (2.25)$$

où :

$$q = \left(\frac{N_I}{N}\right)^3; \quad (2.26)$$

avec,  $N_I$  cardinalité de l'ensemble  $\mathbb{P}_{in}$ . En intégrant la colorimétrie, les ensembles de points  $\mathbb{P}_{in}$  et  $\mathbb{P}_{out}$  sont composés de sous-ensembles de points appartenant à différentes classes colorimétriques (voir figure 2.12) :

$$\mathbb{P} = \left(\mathbb{P}_{in}^1 \cup \mathbb{P}_{in}^2 \dots \cup \mathbb{P}_{in}^j \dots \mathbb{P}_{in}^k\right) \cup \left(\mathbb{P}_{out}^1 \cup \mathbb{P}_{out}^2 \dots \cup \mathbb{P}_{out}^j \dots \mathbb{P}_{out}^k\right)$$

tel que :

$$\mathbb{P}_{in}^j = \{\forall \mathbf{p}_i \in \mathbb{P}_{in} | \mathbf{c}_i \in \mathbb{C}_j\};$$

En supposant que l'ensemble de points  $\mathbb{P}$  est majoritairement composé de points appartenant à une seule classe colorimétrique  $\mathbb{C}_{opt}$ , le nombre d'itération est défini par :

$$\kappa' = \frac{\log(\varepsilon_{th})}{\log(1 - q')}$$

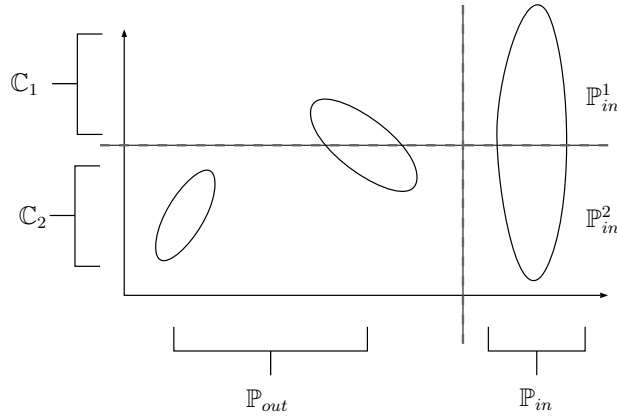


FIGURE 2.12 – Illustration 2D du partitionnement d'un ensemble de points selon deux modes, géométrique et colorimétrique.

où :

$$q' = \left( \frac{N_{Ic}}{N_c} \right)^3 ;$$

$N_{Ic}$  représente la cardinalité de l'ensemble  $\mathbb{P}_{in}^{opt}$  et  $N_c$  la cardinalité de la classe  $\mathcal{C}_{opt}$ .

Pour que la classification colorimétrique aide à réduire la complexité de RANSAC, le nombre d'itérations estimé en prenant en compte la colorimétrie doit être inférieur à celui RANSAC qui se base essentiellement sur les coordonnées géométriques :

$$\kappa' < \kappa \Rightarrow q' > q$$

On en déduit la condition nécessaire pour réduire la complexité de RANSAC par intégration de la colorimétrie

$$\frac{N_{Ic}}{N_c} > \frac{N_I}{N} \quad (2.27)$$

Cette condition est satisfaite durant l'extraction des plans de façade car ils ont -pour la majorité des bâtiments- une couleur unie, ce qui explique les courbes de convergence de la figure 2.10. Au fur est à mesure que des plans sont retirés du nuage de points, les classes deviennent moins denses, ainsi l'apport de la classification colorimétrique est réduit. Deux cas sont envisageables. Le premier où les ratios de l'équation (2.27) sont égaux, ce qui annule l'apport de la classification colorimétrique. Le second cas où la classification colorimétrique augmente considérablement le nombre d'itérations à cause de l'inversion de l'inégalité de l'équation (2.27).

### 2.3.2 Évaluation des paramètres de consistance

La condition de consistance d'un point avec un modèle (équation 2.13) fait appel à deux paramètres. Le premier  $\beta$  représente le seuil de colinéarité du vecteur normal d'un point avec le vecteur normal du plan. Le second  $\delta$  définit la distance à partir de laquelle un point est considéré comme appartenant au plan. Ces deux paramètres sont manuellement définis par l'utilisateur, et influent sur la qualité de segmentation du nuage de points.

La suite de cette section présente une étude de l'influence des seuils considérés pour la segmentation. Afin d'observer l'impact d'un seuil, celui-ci est modifié, en gardant le second seuil constant. L'impact de la variation du seuil est quantifié par l'erreur quadratique moyenne (MSE) et le nombre de points consistants avec le plan ( $|\mathbb{P}_{in}|$ ).

**Impact du seuil  $\beta$**  Ce seuil permet de rejeter les points n'ayant pas la même orientation que le plan estimé. L'étude de l'impact de ce seuil a été réalisée sur un intervalle angulaire compris entre  $10^\circ$  et  $75^\circ$ , avec un seuil de distance  $\delta = 2.9cm$ , considéré comme seuil optimal pour ce plan du bâtiment.

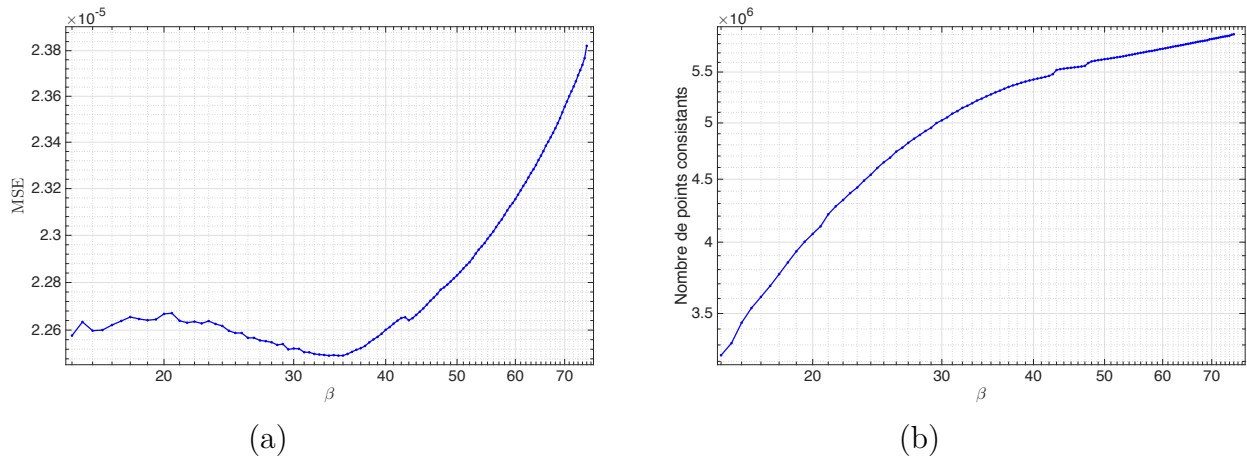


FIGURE 2.13 – Influence de la variation du seuil  $\beta$  sur l'erreur quadratique moyenne et le nombre de points consistants.

La figure 2.13 présente l'évolution de l'erreur quadratique moyenne ainsi que le nombre de points consistants en fonction de la variation du seuil. Sur l'intervalle gauche ( $[15^\circ; 35, 4^\circ]$ ), on constate que la (MSE) évolue d'une manière presque aléatoire avant de tendre un minimum à la valeur  $34, 5^\circ$ . Ce comportement est dû à la variance d'estimation du vecteur normal à chaque fois que le seuil est élargi, d'autres points sont ajoutés à l'ensemble consistant modifiant ainsi l'estimation du vecteur normal et par conséquent la MSE. Il est à noter que cet aspect dépend aussi du biais d'estimation du vecteur normal de chaque point, qui dépend de l'erreur de position des points ainsi que la taille du voisinage considéré pour l'estimation du vecteur normal. Sur l'intervalle droit ( $[34, 5^\circ; 75^\circ]$ ), des points appartenant à d'autres éléments sont ajoutés à l'ensemble des points consistants, ce qui augmente le biais d'estimation, et aboutit par conséquent à l'augmentation de la MSE.

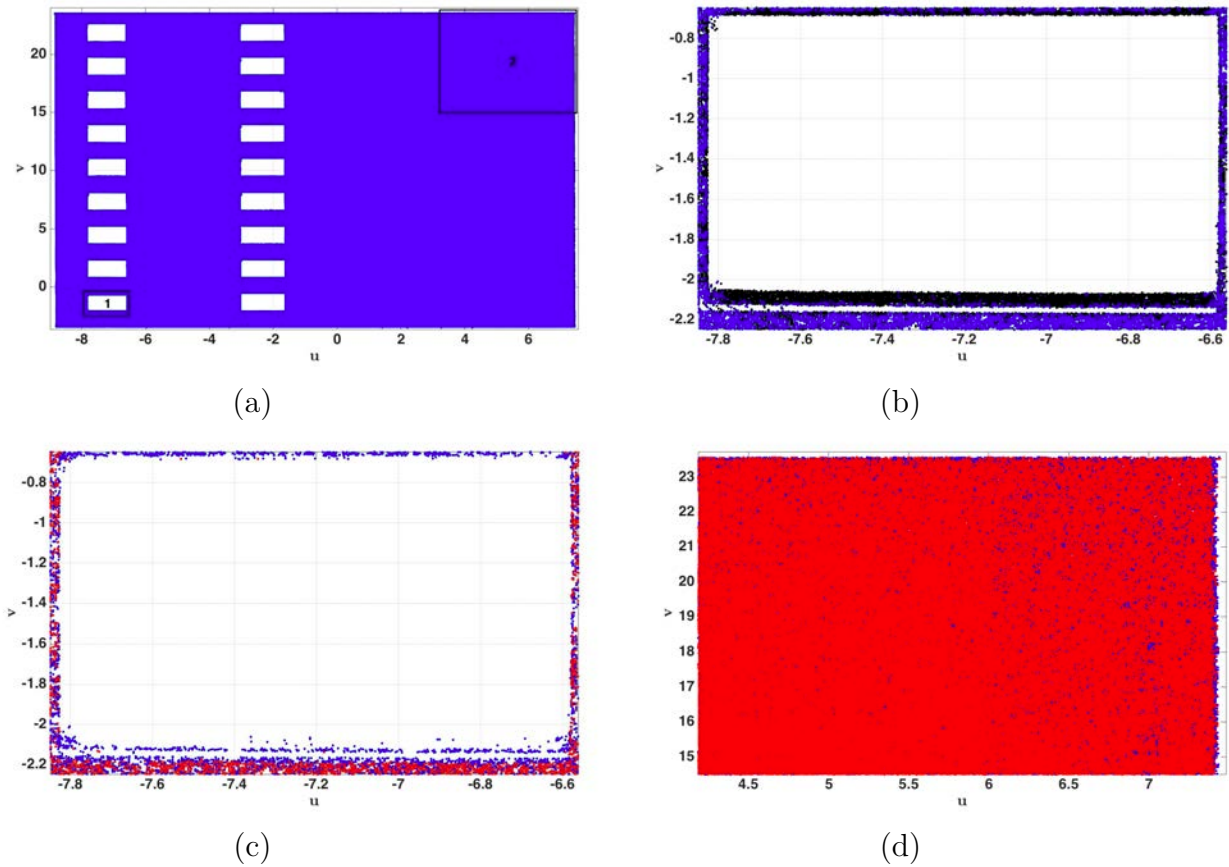


FIGURE 2.14 – Aperçu de l'influence du seuil  $\beta$  sur la segmentation du nuage de point.

La figure 2.14 donne un aperçu de l'impact du seuil  $\beta$  sur l'extraction du plan présenté dans la figure 2.14-a. La figure 2.14-b présente un zoom sur la région 1, où les points en bleu correspondent à l'ensemble de points consistants à un seuil  $\beta = 34,5^\circ$  et l'ensemble des points en noir sont consistants à un seuil  $\beta = 75^\circ$ . Sur ce dernier, on remarque que les points appartenant au linteau sont considérés comme consistants avec le plan de façade. La figure 2.14-c présente le cas contraire. Les points rouges représentent l'ensemble des points extraits avec un seuil  $\beta = 10^\circ$ . Sur cet ensemble, on constate que les points proches des bordures sont rejetés, car les points voisins au niveau des bordures appartiennent à des surfaces ayant des orientations différentes, ce qui biaise l'estimation du vecteur normal. La dernière figure 2.14-d présente le cas où des points à l'intérieur de la surface sont rejetés. Ceci est dû au biais d'estimation du vecteur normal induit par le bruit d'acquisition, car plus la région numérisée est distante du SLT, plus le bruit d'acquisition augmente.

**Impact du seuil  $\delta$**  La définition du seuil de distance trop large induit à une sous-segmentation du nuage de points, où des points appartenant à des plans parallèles sont considérés comme coplanaires. Dans le cas contraire, un seuil trop bas conduit à une sur-segmentation où un ensemble de points coplanaires sont segmentés en plusieurs plans (voir l'illustration présentée dans la figure 2.15).

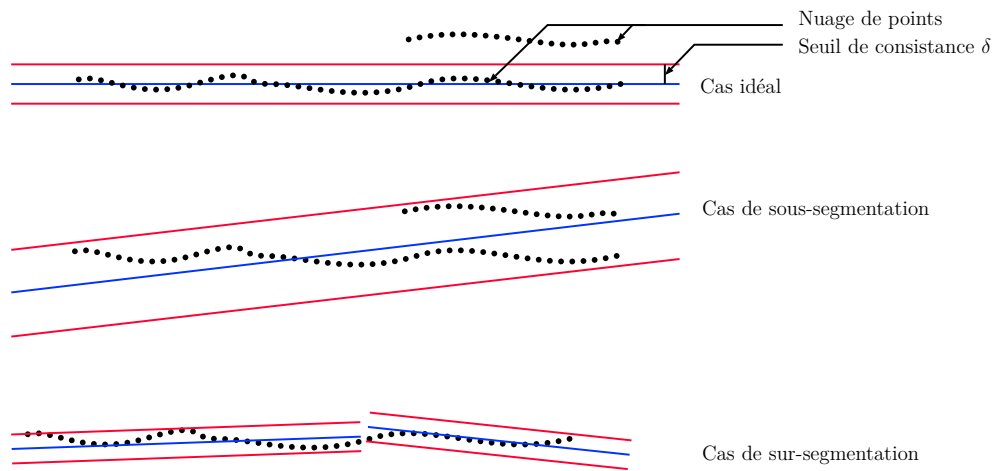


FIGURE 2.15 – Illustration 2D de l'impact du seuil  $\delta$  sur la segmentation du nuage de points.

Les figures 2.16-a et b présentent l'évolution du nombre de points consistants ainsi que de l'erreur quadratique moyenne pour un seuil  $\delta$  variant sur un intervalle entre 2 cm et 6,5 cm tout en gardant le seuil  $\beta$  constant à  $34,5^\circ$ . On constate que l'erreur quadratique augmente proportionnellement au nombre de points consistants. Cependant, on remarque un point d'inflexion de la courbe à la valeur de seuil 2,9cm. Cette valeur a été déduite par une dérivation de la courbe MSE (voir figure 2.17-a). La considération d'un seuil supérieur à cette valeur induit à une sous-segmentation où des points appartenant à des plans parallèles sont considérés comme consistants. La figure 2.17-b donne un aperçu sur le plan, où les points en rouge appartiennent à des plans parallèles au plan de la façade.

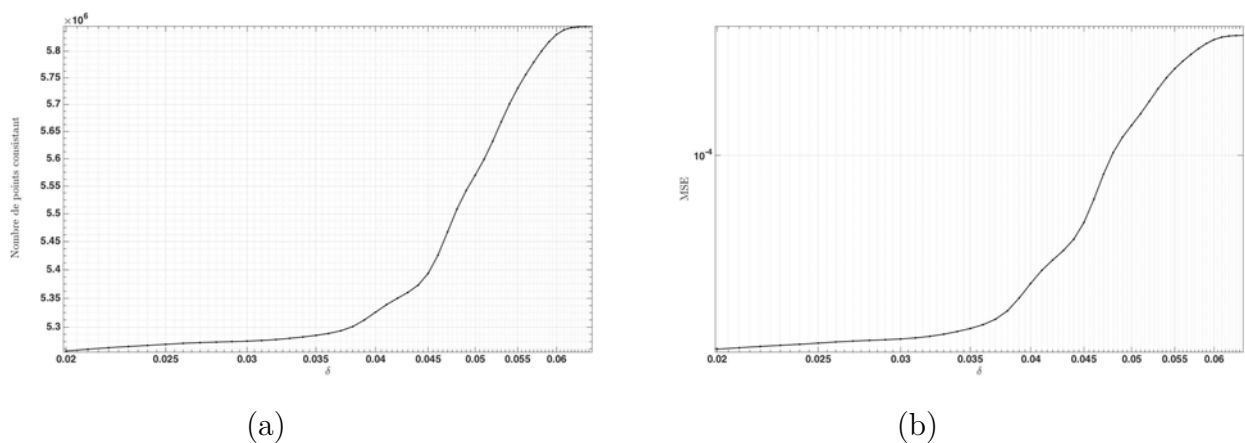


FIGURE 2.16 – Évolution du nombre de points consistants et de l'erreur quadratique moyenne en fonction du seuil de consistance  $\delta$ .



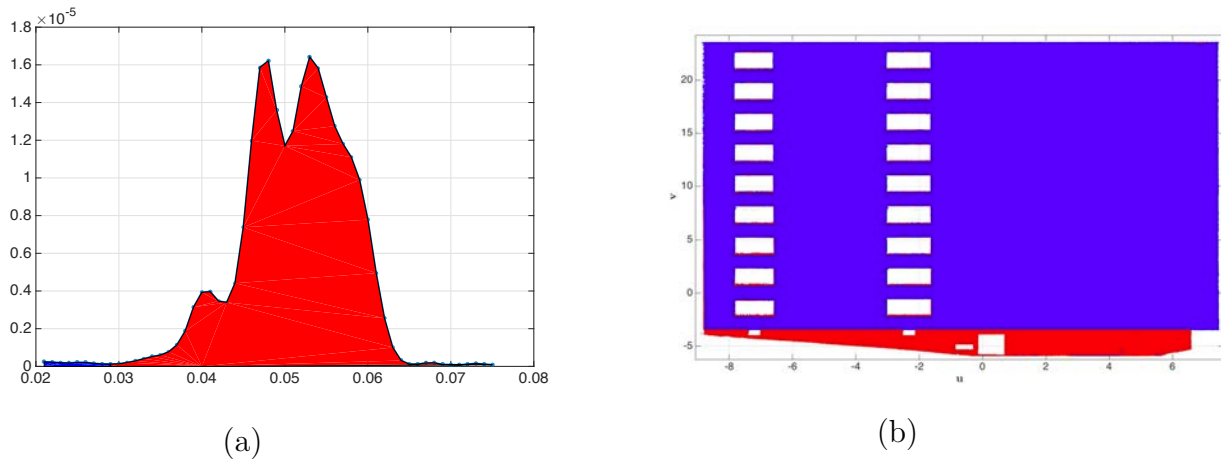


FIGURE 2.17 – Impact du seuil  $\delta$  : (a) Dérivée de l’erreur quadratique moyenne. (b) Points consistants utilisant deux seuils 2,9 cm et 5,5 représentés respectivement par les points bleus et rouges

## 2.4 Conclusion

Une méthode de segmentation du nuage de points a été proposée dans ce chapitre. Son intérêt réside dans la diminution du temps de calcul de la phase segmentation par RANSAC en prenant en compte la colorimétrie.

L’optimisation de la complexité algorithmique est réalisée sur deux niveaux. Le premier consiste à faire un tirage non-uniforme des points constituant l’échantillon minimal, qui est sélectionné à partir d’une seule classe colorimétrique. Le second niveau, consiste à réduire le nombre d’itérations. Pour cela, le calcul de la probabilité de tirer un échantillon minimal non-contaminé par des outliers est limitée aux points dans la classe colorimétrique.

Les limites de cette méthode se manifestent lors de l’extraction itérative des plans de façades. On constate que le nombre d’itérations nécessaires pour extraire les derniers plans est plus grand que le nombre d’itérations effectué par la méthode classique. Cependant, en comparant le nombre d’itérations total pour extraire tous les plans de façade, la méthode proposée reste nettement plus performante.

Afin de minimiser les erreurs de segmentation, une condition sur l’orientation des points consistants a été ajoutée, elle permet d’éviter que des points appartenant à d’autres éléments architecturaux ne soient considérés comme des points de façade.

# 3

## Modélisation de la densité du nuage de points et extraction des points de contours

### 3.1 Introduction

Over the last decade, environment and three dimensional object recording has become easier due to the development of acquisition technologies : photogrammetry, total station, terrestrial and airborne LIDAR. Terrestrial LASER scanner (TLS) are generally used in application such as building modeling, cultural heritage recording and deformation analysis ([RDvdHV07],[BH07], [Bre05], [PV09]) where highly resoled point clouds of large dimension objects are required.

The ability of a TLS to distinguish fine details is depending on the resolution of its components [LJ06]). Firstly, the range resolution, corresponds to the ability of the measurement system to distinguish object on the same line of sight. Second, the angular resolution, depends on both the laser beamwidth and the distance to the object since the beamwidth increases with the distance. It corresponds to the ability to distinguish two fine adjacent details. In practice, TLS manufacturers are providing the angular resolution for a given distance (generally 50 or 100 meters) which results in a minimum angular sampling step. In fact, the angular sampling step implicitly supposes that the object surface is orthogonal to the laser beam at a distance lower than the nominal distance for which the angular resolution is defined [Sch07b].

Let us now define the surface sampling step as the spatial sampling step at the surface of the object to be scanned. Having a constant surface sampling step imposes that the object is always orthogonal to the laser beam which is impossible. In general, the angular sampling step is kept constant for the whole scanning which results in a surface scanning sampling steps varying with the distance and the orientation of the object.

To handle this problem, an option is proposed on some TLS allowing the user to define highly resoled regions, that is regions where the angular sampling steps is low. However, a detailed analysis of the influence of the beamwidth on the scanner resolution [LJ06] reveals that the angular sampling steps does not reflect the actual resolution of

the TLS and that decreasing it does not necessarily result in an increased resolution. The authors, then, introduce the so-called Effective Instantaneous Field Of View (EIFOV) which better reflects the actual TLS resolution. Based on the EIFOV, [PTB11] proposes a method to determine the optimal angular sampling steps considering the distance and the detail size. However this method does not take into account the surface orientation with respect to the TLS as the presented results only consider planar surface orthogonal to the TLS.

The performances of algorithms for point cloud processing are highly depending of the point cloud density. This point was already mentioned by [SA02] for point cloud segmentation using a region growing approach ; their method consists in computing the normal vector for each point and then to merge the points whose normal vectors are similar and whose distance to seed surfaces is lower than a threshold. It is mentioned that using a unique global threshold may lead either to over segmentation in regions with low point cloud density or under segmentation in regions with a high point cloud density. This is actually nothing but the consequence of having a varying surface sampling step which results in a varying point density.

Another example of the influence of the point density can be found in feature and boundary point extraction methods. A number of approaches are trying to determine if a point is belonging to a boundary or surface features by examining the evolution of some estimated geometrical features calculated over a neighborhood surrounding the considered point. For example, [PKG03] uses the evolution of the variance evaluated on successive neighborhood size, while [BL06] considers the distance of the point to the center of the ellipsoid corresponding to a given neighborhood. [TTMM04] proposes a tensor voting approach which basically consists in evaluating the normal directions of points in the neighborhood of a given point. In [CWY<sup>+</sup>12], the approach of [TTMM04] is extended by considering the evolution of the normal directions over successive neighborhoods. In [PV09], [Bou10] the boundary point extraction of planar patches is performed by ordering the edges of a Triangular Irregular Network (TIN) -as provided by Delaunay triangulation- according to their lengths. TIN edge lengths are compared to a threshold. The Longest edges vertices correspond to the boundary points. Assuming a uniform point cloud density results in a global threshold. However, this may lead to wrong boundary point extraction as the surface sampling step is varying with the distance to the surface and the angular positions. In fact, all these approaches relies on the notion of neighborhood which appears to be dependent on the point cloud density. Having a large point cloud density will result in neighborhood covering a small spatial region while low point cloud density will yield large spatial region. As a consequence, it appears that determining feature and boundary points is strongly depending on the point density. In high density regions, it is possible to detect fine boundaries while in low density regions only coarse boundaries can be detected.

The solution developed in this paper consists in adapting the triangle edge length threshold to the local point density by deriving the surface sampling step and its application to TIN based boundary point extraction. It is organized as follows : section 3.2 presents TIN based methods for boundary point extraction, their limits and how they can be improved by considering a varying surface sampling step. Section 3.3, proposes an explicit method for surface sampling steps approximation. A special focus is given to the use of this result for density approximation. The proposed method is finally evaluated on

real data in section 3.4.

## 3.2 TIN based boundary point extraction

Building parametric model reconstruction from TLS point cloud is generally composed of two main phases. Firstly the point cloud is segmented into smaller co-planar point sets representing the building facades. Facade architectural features (windows, doors, extrusion...) are then modeled by fitting geometric primitives to the boundary points of the facade consistent point set. The modeling precision and the detail level of the building facade model depends on the boundary point algorithm accuracy. In this work, we present two algorithms for boundary point extraction based on Delaunay triangulation (defined in the next paragraph) : Delaunay triangulation edges length (paragraphe 3.2.2) and  $\alpha$ -shape (paragraphe 3.2.3). We also discuss how the point cloud density is influencing the results provided by these methods.

### 3.2.1 Delaunay triangulation

The Delaunay triangulation was introduced by Boris Delaunay [Del34]. The triangulation  $\mathcal{D}$  of a point set  $\mathbb{P}$  is constructed such as no point of  $\mathbb{P}$  lies inside circumcircle of any triangle (see figure 3.1). The triangulation is widely used in surface reconstruction for 3D point set and boundary point extraction for 2D point sets.

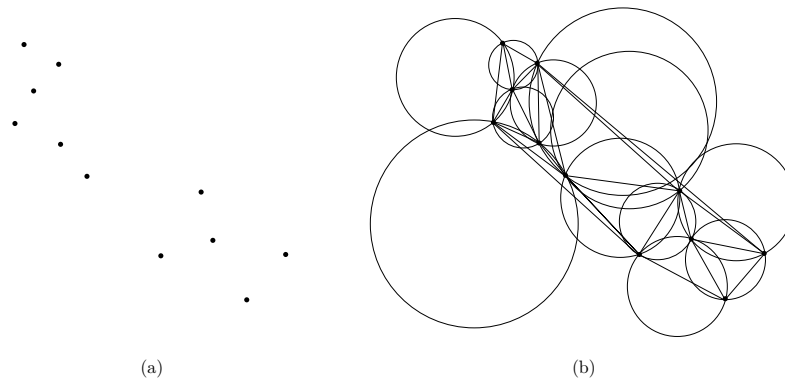


FIGURE 3.1 – Delaunay triangulation illustration : (a) point set ; (b) Delaunay triangulation

### 3.2.2 Boundary points extraction considering Delaunay triangulation edge lengths

The principle of this method is based on the triangle edges. The edges are ordered according to their lengths. Boundary points correspond to the vertices of edges having the higher lengths. As illustrated on figure 3.2, in the regular surface sampling case, the maximum length corresponds to the hypotenuse of the blue triangle. Each edge whose length is greater than this distance has vertices (red points) belonging to a boundary.

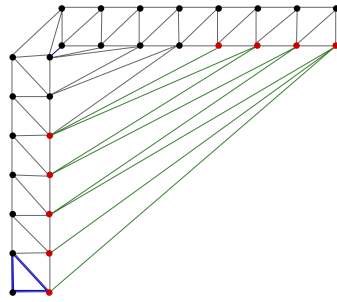


FIGURE 3.2 – Illustration of a Delaunay triangulation of a window corner point considering a uniform density. red point represent the boundary points extracted where green edge length is supposed to be larger than the user defined threshold

### 3.2.3 $\alpha$ -shapes

The  $\alpha$ -shape is a generalization of the convex hull of a point set. It can be defined as polytope bounding the point set with the minimal surface area [EKS83]. The  $\alpha$ -shape ( $\mathcal{P}_\alpha$ ) is composed by a set of straight lines segments called  $\alpha$ -exposed. According to [EM94] : a segment is said to be  $\alpha$ -exposed if the circumcircle it belongs to is empty and it has a radius  $\alpha$  lower than a user defined threshold  $\alpha_{th}$ . The  $\alpha$ -shape algorithm is composed of three main steps :

1. Delaunay triangulation of the point set  $\mathbb{P}$ .
2. The aim of the second step is to remove the edges which have a circumcircle radius larger than  $\alpha_{th}$ . It result from this step a simplicial complex  $\mathcal{SC}_\alpha$  composed of vertices, free and attached edges and triangles. Figure 3.3-c presents the simplicial complex of the point set, here we can distinguish two different sets as a small  $\alpha$  is considered.
3. In this third step, each triangle edge is classified into boundary and interior edge (dashed segment in figure 3.3-a). An edge is considered as interior edge if the largest circumcircle of the triangles it belongs to is not empty (see the illustration in figure 3.4-a). Otherwise, the edge is considered as a boundary point (figure 3.4-b). The resulting edges are presented in figure 3.3-b, where the boundary points are the edge vertices.

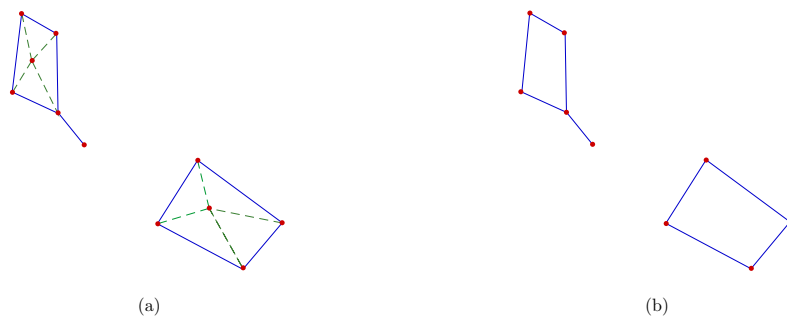


FIGURE 3.3 – Alpha shape process illustration : (a) simplicial complexe (interior edges in green); (b) Boundary edges.

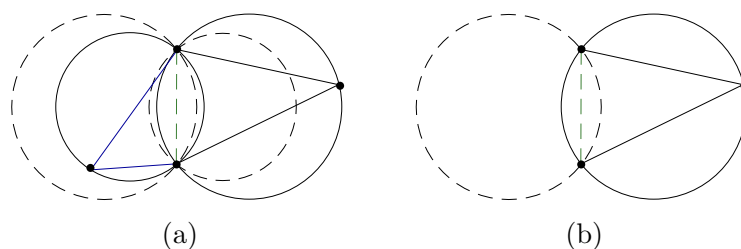


FIGURE 3.4 – Illustration of triangle edge classification : (a) An edge classified as interior ; (b) An edge classified as a boundary

Both of the presented methods consider a unique threshold, assuming a constant surface sampling step. In real case this assumption is wrong, as illustrated in figure 3.5. The surface sampling step vary with respect to the LASER spot incidence angle. These fact may lead to wrong boundary point extraction in most cases, i.e : a fine detail boundaries may not be extracted considering a large threshold in a region with a small surface sampling step. In order to improve the boundary point extraction reliability, both methods should consider a varying threshold according to the local surface sampling steps. The next section proposes an explicit method for surface sampling steps computation of a planar patch.

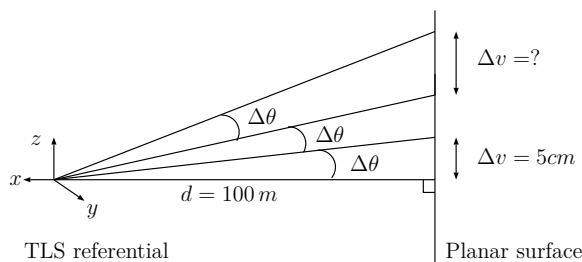


FIGURE 3.5 – Illustration of vertical surface sampling step ( $\Delta u$ ) evolution. Here  $d$  represents the orthogonal distance of the TLS to the planar patch, and  $\Delta\phi$  a constant angular sampling step

## 3.3 Local Surface Sampling Steps Estimation

### 3.3.1 Surface Sampling Function Approximation

Static TLS's are based on the theodolite principle [PB07a], where the point coordinates are measured in a spherical coordinate system representing the distance, the azimuth and the elevation of the surface point. We consider a point cloud representing a scanned planar surface as illustrated in figure 3.6, the point coordinates can be expressed in the spherical coordinates as  $(\rho, \theta, \phi)$  by considering the azimuth and elevation in a referential  $(\mathbf{n}, \mathbf{u}, \mathbf{v})$  attached to the plan,  $\mathbf{n}$  represents the normal vector and  $\mathbf{u}, \mathbf{v}$  a  $2D$  referential attached to the plane. The distance  $\rho$  can be expressed by the equation(3.1) which defines a planar surface in a spherical coordinates system.

$$\rho = -\frac{d}{a \cdot \cos \theta \cdot \cos \phi + b \cdot \sin \theta \cdot \cos \phi + c \sin \phi} \quad (3.1)$$

Here  $d$  is the orthogonal distance of the TLS to the planar patch and  $[a, b, c]$  represents the surface normal vector  $\mathbf{n}$  coordinates. In the referential attached to the plan, the normal vector is expressed as  $\mathbf{n} = [-1, 0, 0]$ , and thus equation (3.1) can be simplified to :

$$\rho = \frac{d}{\cos \theta \cdot \cos \phi} \quad (3.2)$$

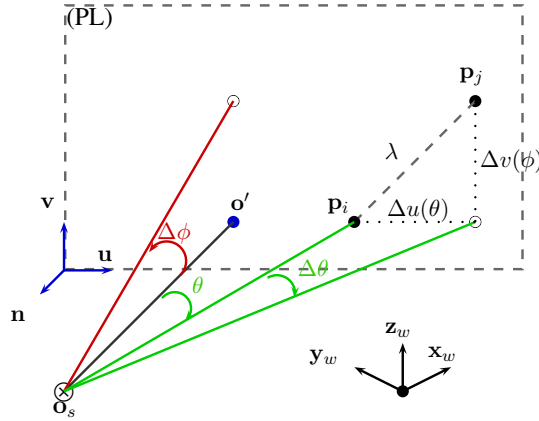


FIGURE 3.6 – Calculation of  $\lambda$  according  $\theta$  and  $\phi$ .

The scanning is made by considering a user defined angular sampling steps  $\Delta\theta$  and  $\Delta\phi$ . This results in the surface sampling steps  $(\Delta u, \Delta v)$  which depend on  $(\rho, \theta, \phi)$ . To calculate the surface sampling steps, we consider two points  $\mathbf{p}_i$  and  $\mathbf{p}_j$ , whose coordinates, defined in the plan referential, are respectively  $(n_i, u_i, v_i)$  and  $(n_j, u_j, v_j)$ . The relation between the spherical coordinates and the Cartesian coordinates of the point  $\mathbf{p}_i$  is given by :

$$\begin{pmatrix} n_i \\ u_i \\ v_i \end{pmatrix} = \rho_i \begin{pmatrix} \cos(\theta_i) \cdot \cos(\phi_i) \\ \sin(\theta_i) \cdot \cos(\phi_i) \\ \sin(\phi_i) \end{pmatrix} \quad (3.3)$$

Replacing  $\rho_i$  by its expression in equation (3.2) yields :

$$\begin{pmatrix} n_i \\ u_i \\ v_i \end{pmatrix} = d \begin{pmatrix} 1 \\ \tan \theta_i \\ \frac{\tan(\phi_i)}{\cos(\theta_i)} \end{pmatrix} \quad (3.4)$$

As  $\theta_j = \theta_i + \Delta\theta$  and  $\phi_j = \phi_i + \Delta\phi$ , the coordinates of the point  $\mathbf{p}_j$  are :

$$\begin{pmatrix} n_j \\ u_j \\ v_j \end{pmatrix} = d \begin{pmatrix} 1 \\ \tan(\theta_j) \\ \frac{\tan(\phi_j)}{\cos(\theta_j)} \end{pmatrix} = d \begin{pmatrix} 1 \\ \tan(\theta_i + \Delta\theta) \\ \frac{\tan(\phi_i + \Delta\phi)}{\cos(\theta_i + \Delta\theta)} \end{pmatrix} \quad (3.5)$$

The surface sampling step can be approximated by considering a first order Taylor expansion of each coordinate expression. Thus, the point  $\mathbf{p}_j$  coordinates can be expressed by the equation :

$$\begin{pmatrix} u_j \\ v_j \end{pmatrix} = d \begin{pmatrix} u_i \\ v_i \end{pmatrix} + d \begin{pmatrix} \frac{\partial}{\partial \theta_i} \tan(\theta_i) \cdot \Delta \theta \\ \frac{\partial}{\partial \phi_i} \tan \phi_i \frac{\Delta \phi}{\cos(\theta_i)} \end{pmatrix} \quad (3.6)$$

Therefore, the sampling function can be written as follows :

$$\begin{pmatrix} \Delta u(\theta_i) \\ \Delta v(\phi_i) \end{pmatrix} = d \begin{pmatrix} \frac{\Delta \theta}{\cos(\theta_i)^2} \\ \frac{\Delta \phi}{\cos(\phi_i)^2 \cos(\theta_i)} \end{pmatrix} \quad (3.7)$$

Thus, as illustrated in figure 3.6, the distance  $\lambda$  between  $\mathbf{p}_i$  and  $\mathbf{p}_j$  can be written as :

$$\lambda = \sqrt{\Delta u(\theta_i)^2 + \Delta v(\phi_i)^2} \quad (3.8)$$

### 3.3.2 Low resolution approximation of the surface sampling steps

According to the previous section, it is possible to perform a low resolution approximation of the surface sampling steps. Basically, the idea is to define angular patches over which the surface sampling functions will be considered as constant. Indeed, defining a piecewise constant surface sampling steps corresponds to consider a linear approximation of  $u(\theta)$  and  $v(\theta, \phi)$ . Let us consider the regular grid defined in the spherical coordinate system (see figure 3.7-a). Each vertex of this grid is defined by it's coordinates :

$$S_{k,l} = \begin{pmatrix} k\Theta \\ l\Phi \end{pmatrix}$$

The regular grid defined in  $(\theta, \phi)$  yields a curvilinear grid in the Cartesian coordinate systems as shown in figure 3.7-b. Here  $\Theta$  and  $\Phi$  stand for the grid steps (intervals) according -respectively- to  $\theta$  and  $\phi$ . whose vertices are given by :

$$C_{k,l} = \begin{pmatrix} d \tan(k\Theta) \\ d \frac{\tan(l\Phi)}{\cos(k\Theta)} \end{pmatrix} \quad (3.9)$$

Rather than considering these curvilinear grid, we proposed to approximate each curvilinear cell by a convex quadrilateral whose vertices are those of the corresponding curvilinear cell. This results in the polyhedral grid of figure 3.7-c. As in each cell the surface sampling function is considered as constant, we can evaluate the diagonal distance in each cell by :

$$\lambda_{k,l} = \sqrt{\Delta u(\bar{\theta}_k)^2 + \Delta v(\bar{\phi}_l)^2} \quad (3.10)$$

where  $\bar{\theta}_{k,l}$  and  $\bar{\phi}_{k,l}$  correspond to the geometrical center of each cell of the regular grid in  $(\theta, \phi)$  :

$$\bar{\theta}_k = \frac{2k+1}{2}\Theta; \text{ and } \bar{\phi}_l = \frac{2l+1}{2}\Phi; \quad (3.11)$$



The values  $\Delta u(\bar{\theta}_k)$  and  $\Delta v(\bar{\phi}_l)$  are calculated according to equation (3.10). We note that the value of  $\lambda_{k,l}$  corresponds to an "average value" (this is not the actual mean value but rather a median value).

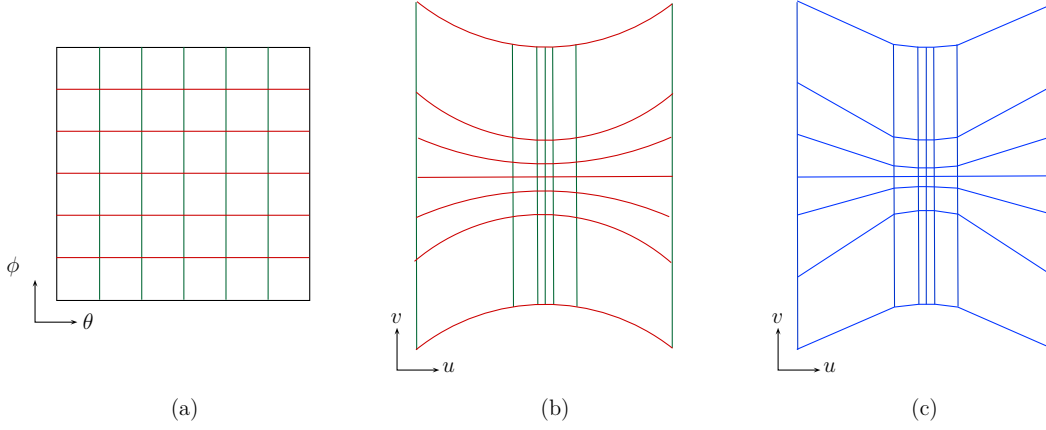


FIGURE 3.7 – Grid transformations : (a) rectangular regular grid defined in the spherical coordinate system. (b) Transformation of the rectangular grid in the Cartesian coordinate system, which leads to a curvilinear grid. (c) Simplification of the curvilinear grid to polyhedral grid.

### 3.3.3 Point cloud density approximation

To assess the quality of the surface sampling steps approximation, we propose to compare the corresponding point cloud density to point cloud density estimated on the actual data which will serve as ground truth. These densities will be referred as approximated density and estimated density respectively. The estimated density is estimated on a grid<sup>5</sup> which does not necessarily match the polyhedral grid on which the surface sampling step is calculated. To give an example, we can estimate the density on a rectangular grid whose vertices are those defined by the rectangular grid in  $\Theta$  and  $\Phi$ . Thus, it is necessary to match the polyhedral grid to the rectangular grid as illustrated in figure 3.8-c.

In the following, to simplify the notations, the dependence with respect to the indexes of the rectangular cells is omitted. For each rectangular cell, they are  $R$  intersecting polyhedral cells thus the rectangular cell surface satisfies  $A = \sum_{r=i} A_r$  where  $A_r$  represents the surface of the  $r^{th}$  cell intersecting the rectangular cell.

$$\begin{pmatrix} \Delta u(\tilde{\theta}_{k,l}) \\ \Delta v(\tilde{\phi}_{k,l}) \end{pmatrix} = d \begin{pmatrix} \sum_{r=1}^R \Delta u(\bar{\theta}_r) \cdot w_r \\ \sum_{r=1}^R \Delta v(\bar{\phi}_r) \cdot w_r \end{pmatrix} \quad r = 1, \dots, R; \quad (3.12)$$

In equation (3.12),  $w_r$  represents the weight affected to the  $r^{th}$  surface sampling step. This weight is obtained as the ratio of the  $r^{th}$  polyhedral surface cell to the rectangular

5. Here for simplicity we only consider rectangular grid

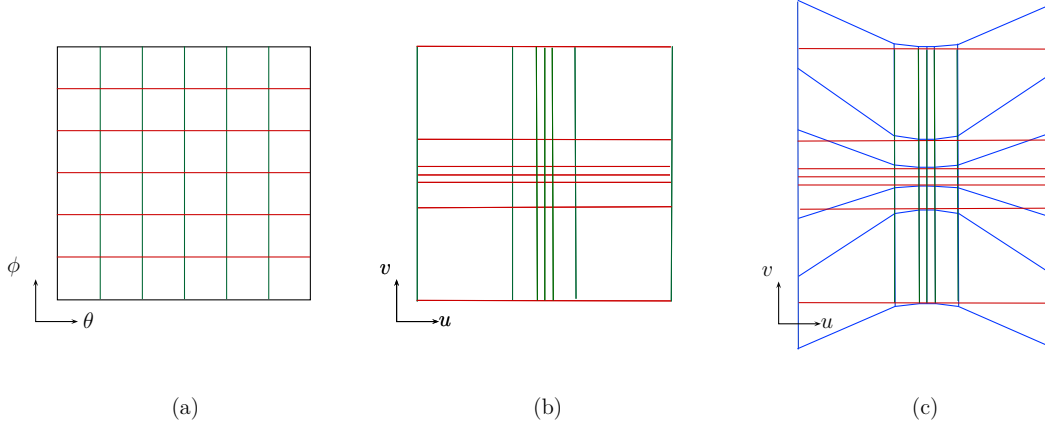


FIGURE 3.8 – Polyhedral and rectangular grid superposition : (a) Regular grid in the spherical coordinate system. (b) Rectangular grid. (c) Superposition of the rectangular and polyhedral grids

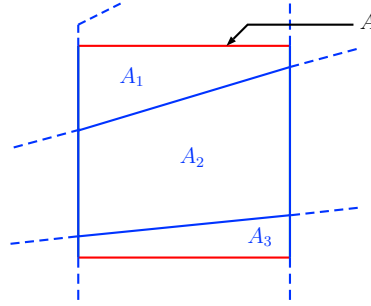


FIGURE 3.9 – Intersection of a rectangular cell  $A$  with polyhedral cells  $A_r$

surface cell :

$$w_r = \frac{A}{A_r} \quad (3.13)$$

Thus, we define  $\lambda_{k,l}$  for each cell by equation (3.14), which represents an average diagonal distance.

$$\lambda_{k,l} = \sqrt{\Delta u(\tilde{\theta})^2 + \Delta v(\tilde{\phi})^2} \quad (3.14)$$

To conclude this section, we address the problem of accounting for multiple point clouds acquired from multiple stations. We suppose that these point clouds have been already properly registered. Thus all the coordinates of the points belonging to the different clouds are expressed in a single referential.

The point cloud density is higher in overlapping areas, however the resolution improvement cannot be guaranteed. To give some insights into the understanding of these point, let us consider the case of two overlapping point cloud having the same resolution. Figure 3.10 illustrates the 3 possible cases in the overlapping regions. The case (a) represents the most favorable one where the resolution is doubled, while the case (b) shows a

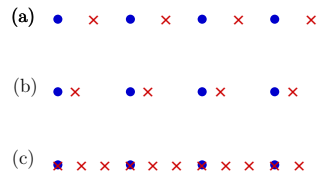


FIGURE 3.10 – Illustration of overlapping region : red points belong to the first station while the blue belong to a second station

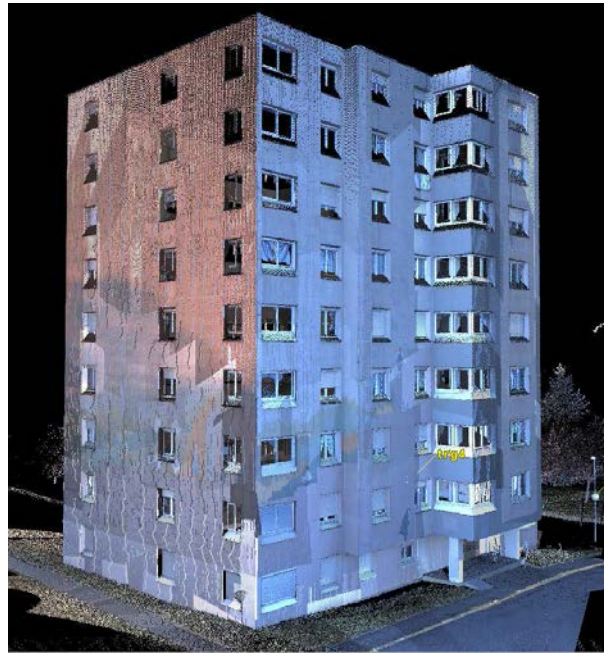


FIGURE 3.11 – Building point cloud

non uniform resolution for which the proposed surface sampling approximation cannot be extended straightforwardly. The case (c) corresponds to points of one station overlapping with some of the other stations. This is the least favorable situation since in that case the resolution remains that of the point cloud with highest resolution. To handle the case of multiple point clouds, we adopt a simple strategy consisting in using the minimum surface sampling which corresponds to the point cloud having the maximum density.

### 3.4 Experimental results

The accuracy of the proposed surface sampling approximation is evaluated on a building point cloud acquired from 6 stations with a TLS Leica C10 (see figure 3.11). The TLS specification and the recording parameters are given in table 3.1.

The accuracy of the surface sampling step approximation was evaluated according to two aspects. The first one consists in comparing the corresponding approximated point cloud density to the estimated point cloud density (section 3.4.2). The second one aims

<i>Range measurement :</i>	
Measurement principle	Time of flight
Maximum range	300 m
<i>Accuracy :</i>	
Position	$\pm 6\text{mm}$
Angle	$\pm 60\mu\text{rad}$
Range	$\pm 4\text{mm}$
Surface noise standard deviation	2mm
<i>Recording parameters :</i>	
Angular sampling step	$5 \cdot 10^{-4} \text{ rad m}$
surface sampling steps	0.05 m
Nominal distance ( $d_n$ )	100 m

TABLE 3.1 – TLS Specification and recording parameters

at evaluating the interest of taking into account the local surface sampling step in a TIN based edge points extraction methods (section 3.4.3). The evaluation is conducted on two different plans. The first one is extracted from a single station point cloud, while the second is extracted from overlapping point clouds.

### 3.4.1 Point cloud segmentation

The point cloud segmentation consists in decomposing the point cloud into plans. The plans having the largest number of points correspond to the building facades. The point cloud segmentation was performed using a RANSAC based algorithm proposed by [AZDZ10].

The singular value decomposition of the planar consistent point set directly gives the referential (n,u,v) attached to the plan and the coordinate of the points in this referential.

### 3.4.2 Density Comparison

The first way to asses the accuracy of the surface sampling step approximation consists in comparing the approximated density to the density directly estimated from the actual point cloud. In this evaluation, two cases of planar patches were considered; the first represents a planar patch acquired form a single point of view, while the second represents a planar patch acquired from two points of view.

For the first case, the comparison process is composed of three main steps. First an irregular rectangular grid (see section 3.3.2) is defined considering a regular spacing  $\delta\theta = \delta\phi = 0.02\text{rad}$ . The second step consists in computing the approximated ( $\tilde{\mathbf{D}}_{k,l}$ ) and real density ( $\mathbf{D}_{k,l}$ ) for each patch of the grid using respectively the equations (3.15) and (3.16). The density comparison is then made in the last step, where the density difference quantified with the ratio of the estimated density divided by the approximated density defined in equation (3.17).

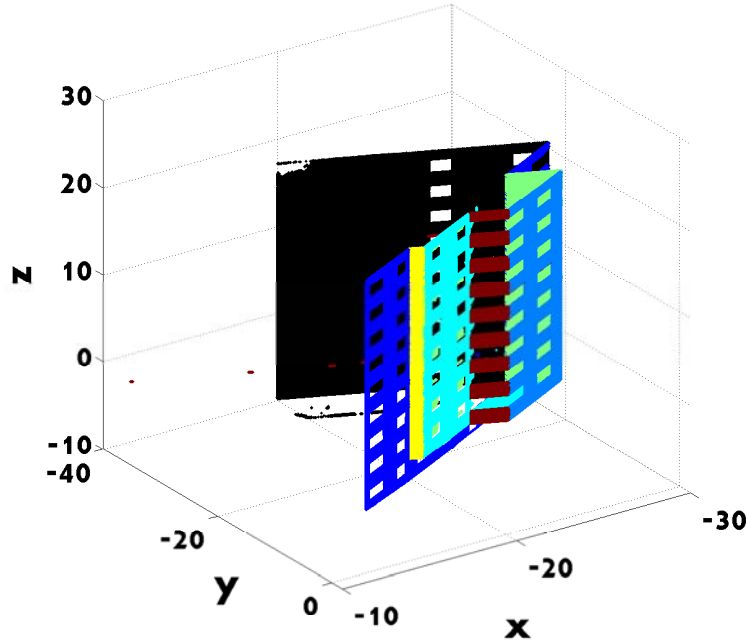


FIGURE 3.12 – Segmented point cloud

$$\tilde{\mathbf{D}}_{k,l} = \frac{1}{\Delta u(\tilde{\theta}_{k,l}) \cdot \Delta v(\tilde{\phi}_{k,l}) \cdot a_{k,l}} \quad (3.15)$$

$$\mathbf{D}_{k,l} = \frac{|\mathbb{S}_{k,l}|}{a_{k,l}} \cdot \frac{1}{a_{k,l}} \quad (3.16)$$

where  $a_{k,l}$  denotes the aerea of the patch and  $\mathbb{S}_{k,l}$  points set inside the patch.

$$\mathcal{R} = \frac{\mathbf{D}_{k,l}}{\tilde{\mathbf{D}}_{k,l}} \quad (3.17)$$

Figure 3.13 shows the results of each step according to the plan in figure 3.13-a representing the first case. Figure 3.13-b shows the real density, we can observe that the density decreases according to distance of the patch to the point (0,0), which corresponds to the orthogonal projection of the station referential origin on the plan. The same aspect can be observed in figure 3.13-c representing the approximated density. Figure 3.13-d shows the ratio of both densities, which is close to one in patches on façade region, and thus the densities are similar on those regions.

For the second case -which represents a plan acquired from two points of view-, the evaluation process is as follows. First, the approximated density is computed independently for each point of view using the same process of the first case (plan acquired from a single point of view). The resulting approximations for each point of view are then combined on a regular rectangular grid, which represents a partition of the plan into a regular square patches of  $25 \text{ cm}^2$ . The approximated density of each square is equal to the

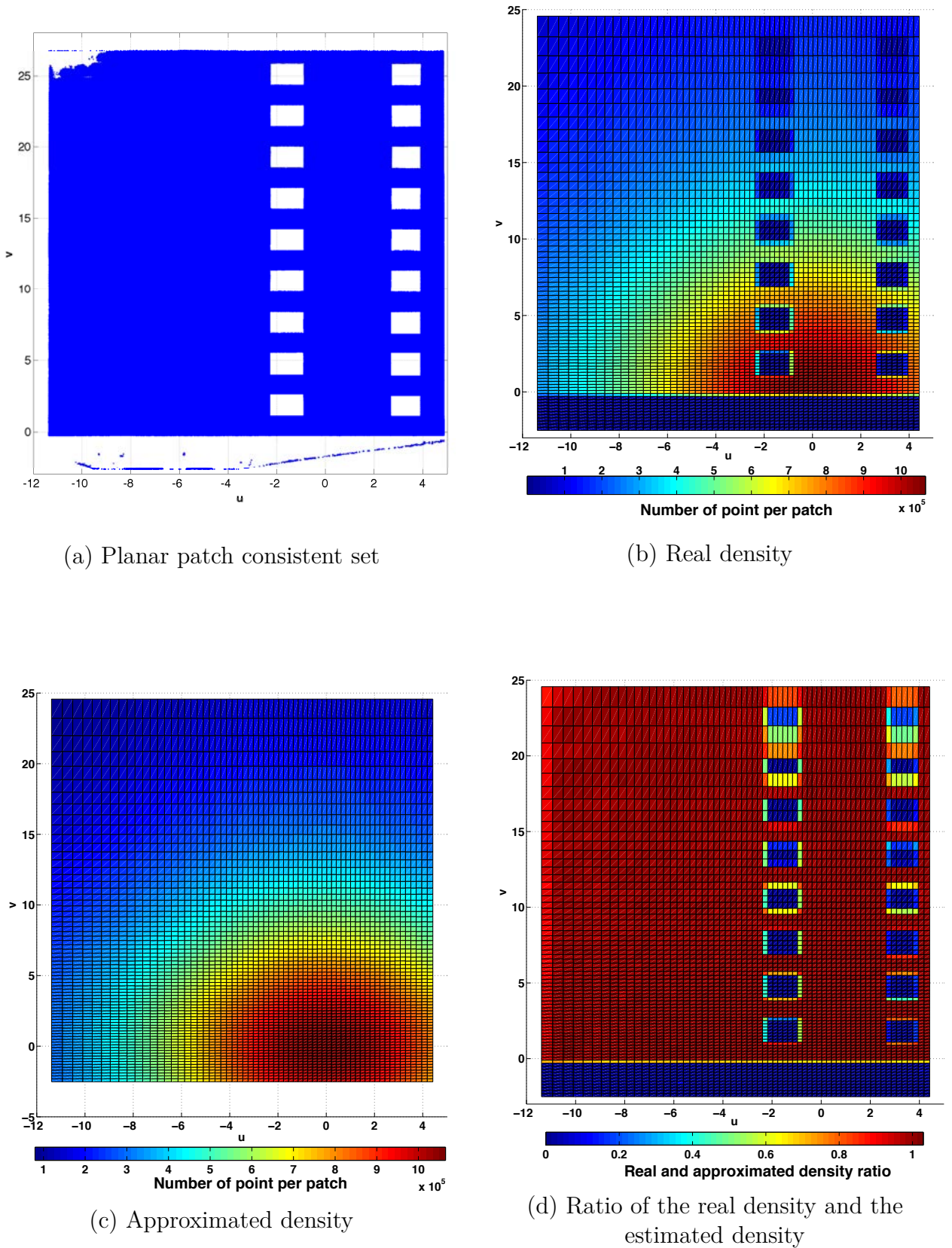


FIGURE 3.13 – Density comparison of a planar patch acquired from one point of view

sum of approximated densities of irregular patches it intersect with. Figure 3.14 shows the results of the density comparison. Figure 3.14-a presents the considered planar patch for the test, while the figure 3.14-b shows the real density of the regular square grid. Here again the density is higher in the patches close the points of view projections than in the farthest patches. The same observation can be made on Figure 3.14-c representing the approximated density. Figure 3.14-d presents the ratio of both densities; the ratio is close to one, except in patches on edges and inside windows borders. For both cases, the observed differences on patch on façade region are very small and can be attributed to the limitation of the first order Taylor approximation and the quantification of the surface sampling steps on the irregular grid.

### 3.4.3 Application to edge point extraction

In this work, we propose to consider a locally defined threshold for each point according to the rectangular cells  $RC_{k,l}$ . Considering only the threshold in equation (3.14) will certainly leads to a wrong boundary point extraction because of two major reasons. The approximated surface sampling steps are limited to the first order of Taylor expansion. The approximation does not take into account the uncertainty of the point position, due to the acquisition noise and surface irregularities. To take into account these factors, we consider a user defined parameter  $\gamma$ . Therefore, the threshold affected for point subset inside a rectangular cell is expressed as :

$$\lambda'_{k,l} = \gamma \lambda_{k,l} = \gamma \sqrt{\Delta u(\tilde{\theta}_{k,l})^2 + \Delta v(\tilde{\phi}_{k,l})^2}$$

The simplexe (a triangle or an edge) vertices may lies in different rectangular cells and/or belong to different stations. In this case, the threshold considered for  $\alpha$ -shape (circumcircle radius) corresponds to the maximal threshold affected for the triangle vertices. In the same way, the threshold considered for boundary point extraction according to triangulation edge lengths corresponds to the maximal threshold affected to its vertices.

To evaluate the edge point extraction considering local threshold reliability, the boundary points sets automatically extracted ( $\mathbb{B}_a$ ) are compared to a manually extracted boundary point set ( $\mathbb{B}_m$ ) considered as a ground truth. The extraction error ( $\mathcal{E}$ ) is quantified by the equation (3.18), where the nominator represents the cardinality of the difference between  $\mathbb{B}_a$  and  $\mathbb{B}_m$  and the denominator their intersection cardinality.

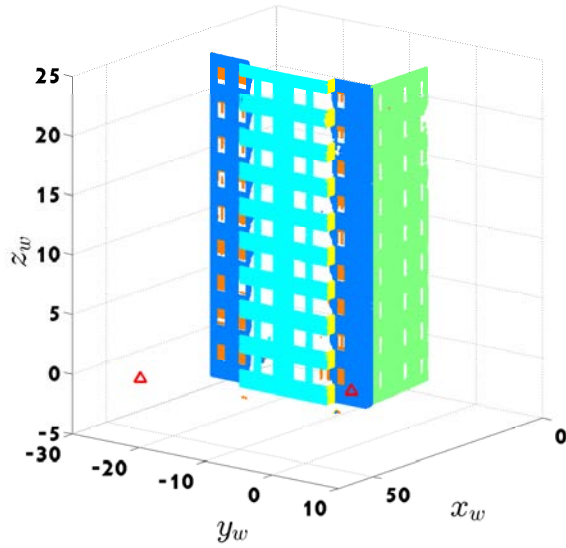
$$\mathcal{E} = \frac{|\mathbb{B}_m \oplus \mathbb{B}_a|}{|\mathbb{B}_m \cap \mathbb{B}_a|} \quad (3.18)$$

In order to make a reliable evaluation, the set  $\mathbb{B}_a$  was extracted using  $\gamma$  optimal value which minimize the error as follows :

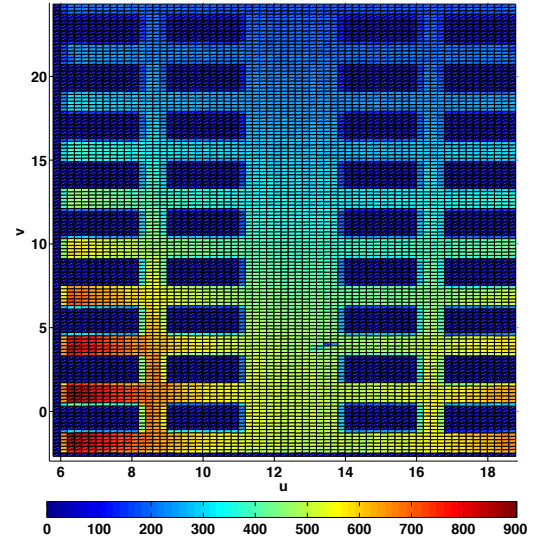
$$\gamma_{opt} = \arg \min_{\gamma} \frac{|\mathbb{B}_m \oplus \mathbb{B}_a(\gamma)|}{|\mathbb{B}_m \cap \mathbb{B}_a(\gamma)|} \quad (3.19)$$

Also, to quantify the contribution of using local threshold instead of global threshold, the extracted boundary set by each method using local threshold was compared to the set extracted using optimal global threshold. These last defined as follows :

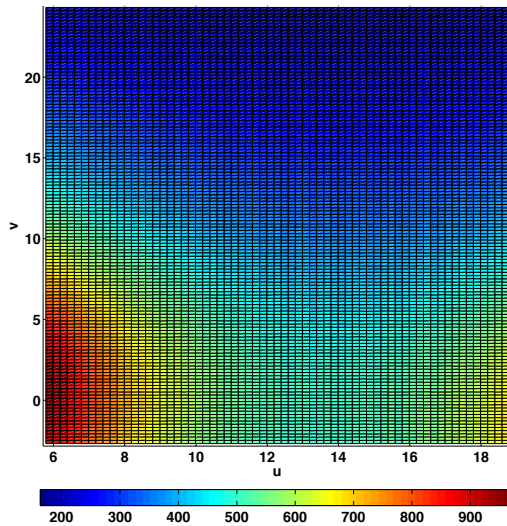




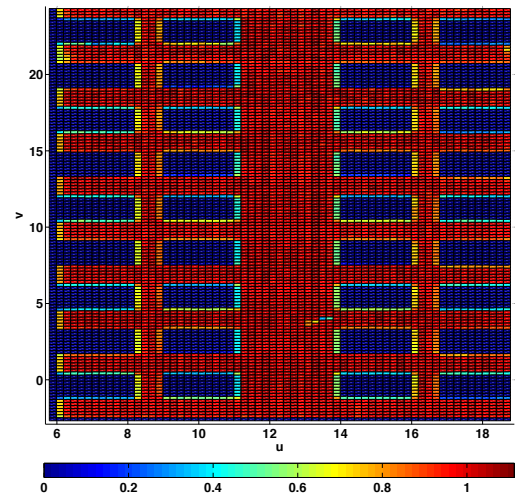
(a) Planar patch consistent set



(b) Real density



(c) Approximated density



(d) Ratio of the both densities

FIGURE 3.14 – Density comparison of a planar patch acquired from two points of view



Planar patches	$\alpha$ -shapes		Edges lengths	
	local threshold	unique threshold	local threshold	unique threshold
First plan	1,37	1,6	3,51	3,41
second plan	0,67	0,97	0,96	1,51

TABLE 3.2 – Difference results by comparing the automatically extracted boundary points sets to the manually extracted point set. The table shows the results for both algorithm  $\alpha$ -shape and Delaunay triangulation edges length

$$\gamma_{g_{opt}} = \arg \min_{\gamma_g} \frac{|\mathbb{B}_m \oplus \mathbb{B}_a(\gamma_g)|}{|\mathbb{B}_m \cap \mathbb{B}_a(\gamma_g)|} \quad (3.20)$$

where :

$$\lambda_{g_{opt}} = \gamma_{g_{opt}} \cdot \lambda_{g_{init}} \quad (3.21)$$

Here,  $\lambda_{g_{init}}$  stands for an initial value of the global threshold  $\gamma_{opt}$  and  $\gamma_{g_{opt}}$ , an optimal parameter value for local and global threshold.

Figure 3.15 shows the curves of error evolution according to  $\gamma$  for the planar patch acquired by one station (figure 3.15-a) and the planar patch acquired by two stations (figure 3.15-b). The red and black curves show -respectively- the error evolution considering local threshold for boundary point extraction Delaunay triangulation edge length, while green and blue curves shows the error evolution considering respectively local and global thresholds. For both cases plan acquired with from a single point of view and plan acquired from two points of view, the sets extracted using a local threshold show a smaller error than the extracted sets using a global threshold. The table 3.2 resumes the minimal error according to the manually extracted boundary point set.

Figures 3.16 and 3.17 show the differences between sets extracted using local thresholds and unique thresholds for each method. Blue squares represent boundary points extracted using local threshold considering ( $\gamma_{opt}$ ) estimated for each method, while red points show boundary points extracted using unique threshold optimal value ( $\gamma_{g_{opt}}$ ) for each method. The differences are exposed across two different regions for each case (see zoomed regions in figures 3.16 and 3.17). These regions were selected according to their respective positions to the TLS. As expected the difference amount depends on the region location. While in the regions close to the TLS show a big difference, the farthest regions show less differences. These results are mainly caused by the non-uniform aspect of the density. The unique threshold are usually defined to extract boundary points according to regions with low density, which cause a loss of boundary points in region with a high density. The consideration of a local threshold according to surface sampling step allows to define an optimal threshold for each region, which reduce boundary point loss.

### 3.4.4 Discussion

**Regular and irregular grid utilization for density comparison** The aim of using both types of densities stands for two reasons. Firstly, the generic application for facade patch density approximation for classification of facade feature to architectural elements

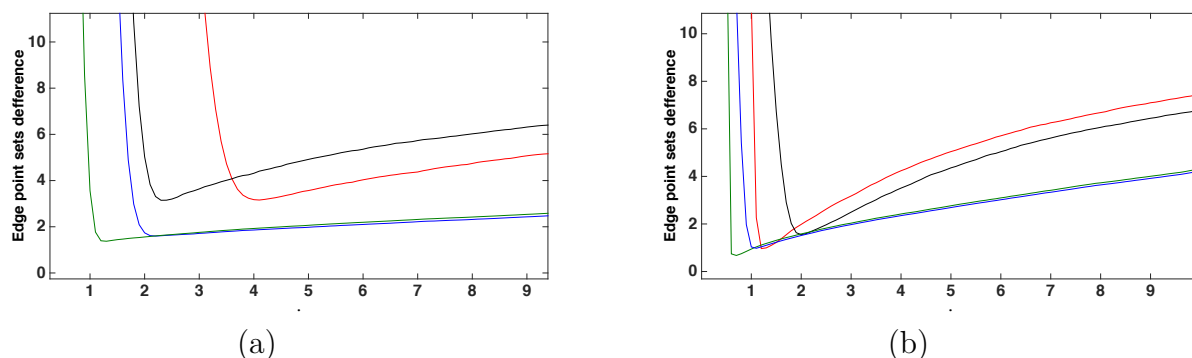


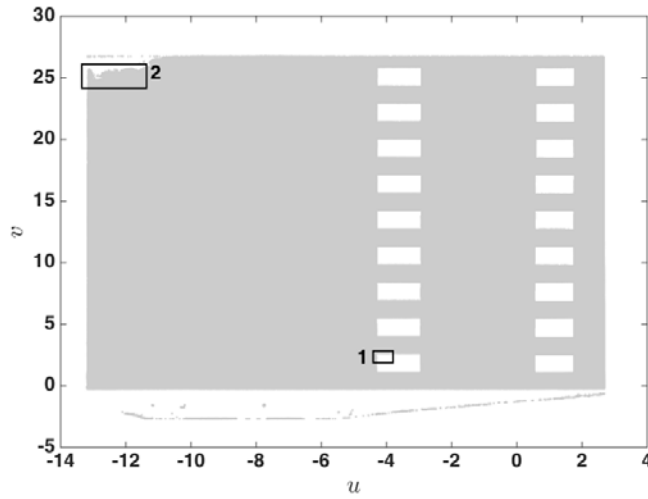
FIGURE 3.15 – Boundary point set difference evolution according to  $\gamma$  : (a) Test on the plan acquired from one point of view ; (b) Test on the plan acquired from two points of view.

(window,extrusion,doors,...) and/or zone of irregularity. The second reason is the results rendering for the second case -planar patch acquired from two points of view-, where the interpretation of two superposed irregular grid may be hard to interpret for the reader.

**Boundary point extraction considering local threshold** As it can be seen in figures 3.15, there are small differences between the sets of boundary points extracted by the two algorithms considering local thresholds. However this comparison is made according to optimal values parameter  $\gamma$ , which is hard to set manually. Also it is worth mentioning that the boundary point extraction via  $\alpha$ -shape algorithm considering either local or unique thresholds allows a more economic cost than the algorithm considering the triangles edge lengths.

## 3.5 Conclusion

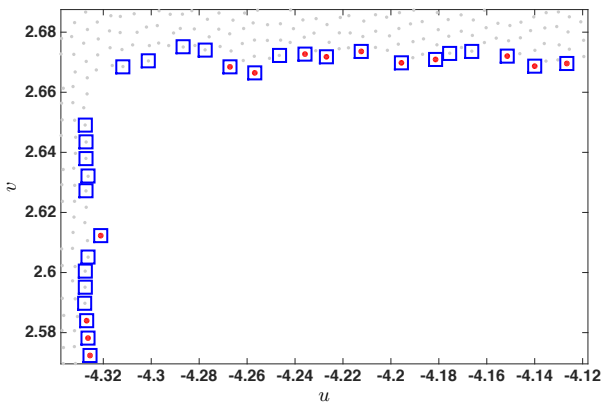
Point cloud modeling algorithm reliability mostly depends on the point cloud density, which evaluate according to the surface sampling steps. The main aim for this work is to improve the parameters setting of the algorithm. The idea consists in defining a local parameter for the algorithm according to the local point cloud density. Thus a model for surface sampling step is proposed, where the surface sampling steps are defined according to a regular grid patches in which we consider that the sampling steps are constants. The model evaluation by comparing the approximated density with the model to the actual point cloud density shows satisfying results. An application for boundary points extraction using local threshold is proposed. For the two considered algorithms,  $\alpha$ -shape and Delaunay triangulation edge length, an improvement induced was observed by comparing the extracted boundary sets using a locally defined threshold to a manually extracted set and to the sets extracted using a unique threshold.



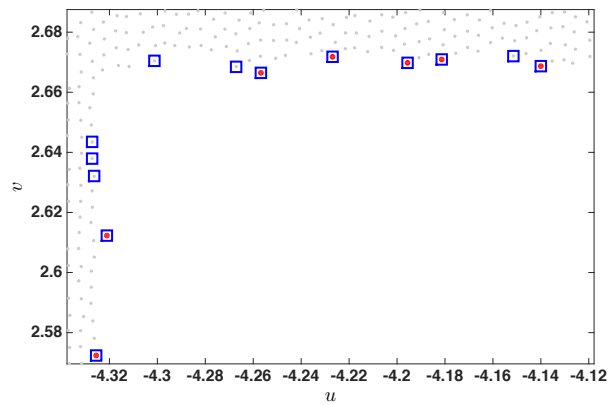
(a) Segmented plan acquired from a single point of view.

$\alpha$ -shape

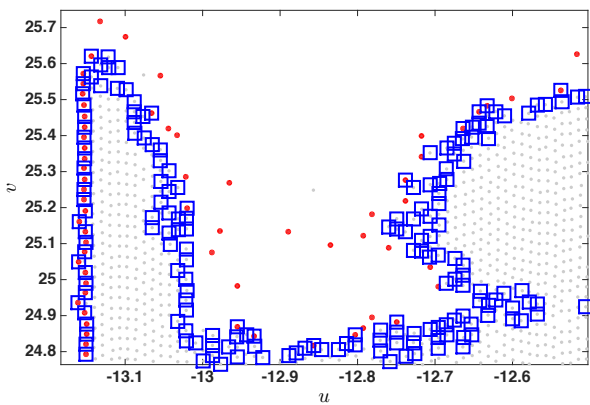
Triangulation edges length



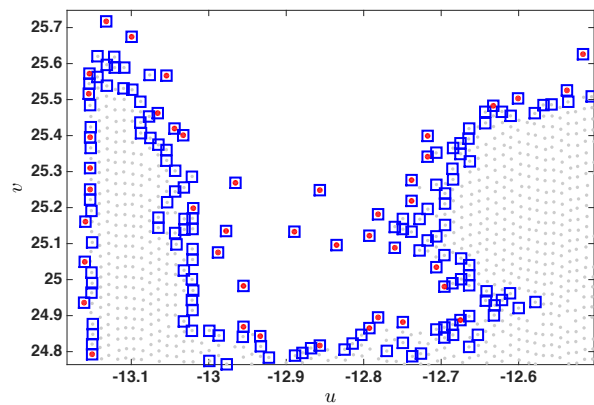
(b)



(c)

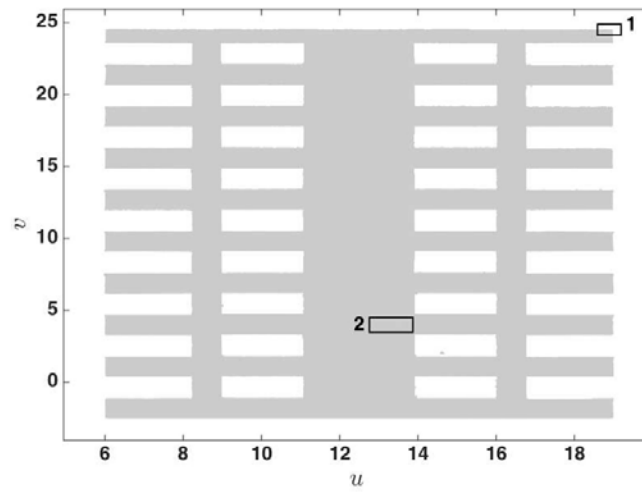


(d)



(e)

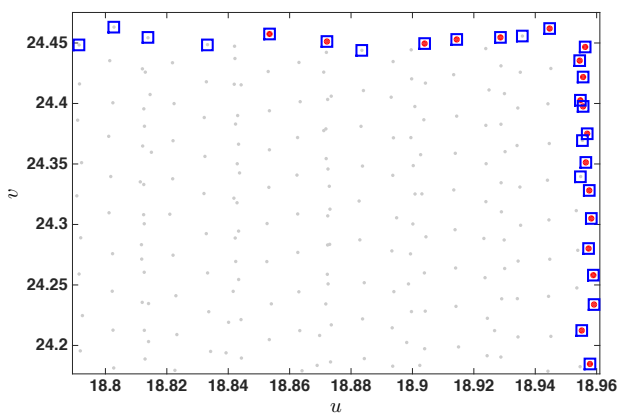
FIGURE 3.16 – Boundary point extraction for the planar patches : (a) planar patch acquired from one point of view ; The rectangles show the zoomed positions where the corresponding images are presented in figure (d) and (e)



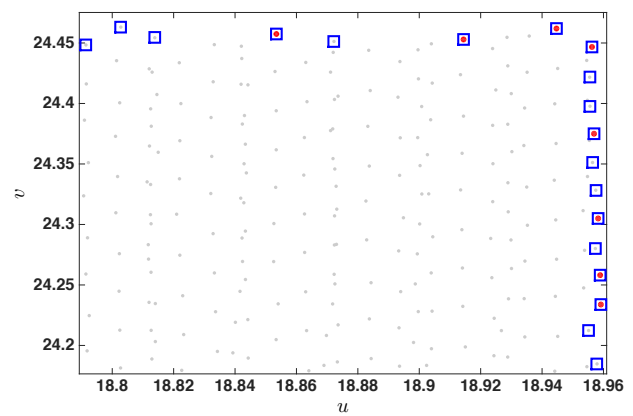
(a) Segmented plan acquired from two points of view.

 $\alpha$ -shape

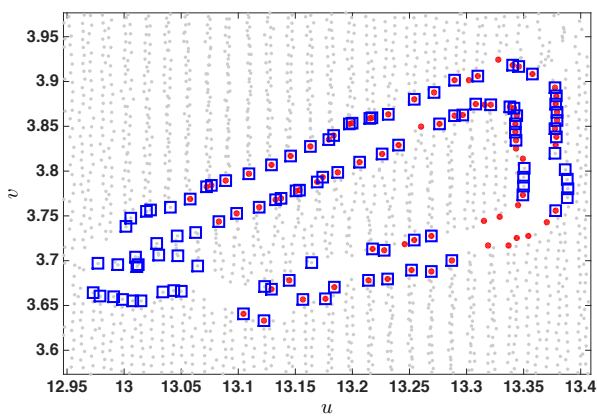
Triangulation edges length



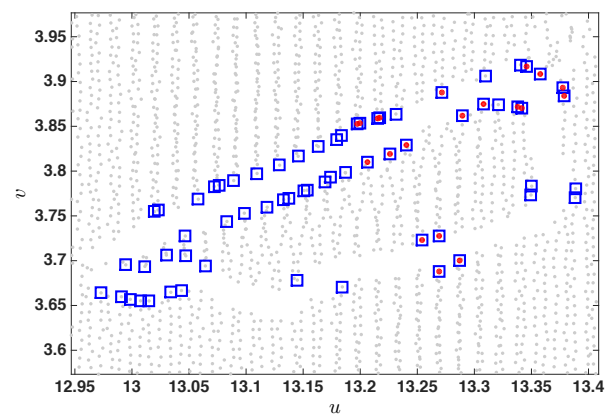
(b)



(c)



(d)



(e)

FIGURE 3.17 – Comparison of extracted boundary point set using local and global threshold for a planar patch acquired from two points of view



## 4

# Classification et modélisation des zones d'intérêts

Le dimensionnement des Modules d'Isolation Thermique (MIT) est essentiellement basé sur le modèle CAO. Les MIT sont rigides et difficilement modifiables, ainsi la phase de conception doit tenir compte des zones d'irrégularités de la façade ainsi que de l'erreur du positionnement des cotes des éléments architecturaux. Ces erreurs sont majoritairement compensées par les systèmes de fixation des MIT (équerres, rails). Cependant, chaque système a ses limites, et afin de prévoir la solution adéquate à la géométrie du bâtiment une quantification de l'erreur sur le modèle CAO s'avère indispensable.

Ce chapitre propose une méthode de reconstruction de modèle filaire à partir des points de contours extraits pour les différents plans. La méthode présentée dans la figure 4.1 se compose de trois phases principales. Dans un premier temps, les régions définies par les points de contours sont classées en Régions d'Irrégularité (RI), Régions d'Éléments Architecturaux (REA) et Région de Façade (RF). La seconde phase consiste à modéliser ces différentes régions par un maillage de Delaunay pour les RI, et des polyèdres pour les RF et REA. La dernière phase calcule une approximation de l'erreur sur le modèle.

La section 4.1 présente les différents types de régions ainsi que leur méthode de classification. La section 4.2 présente les méthodes de modélisation des régions. La section 4.3 propose une méthode d'approximation de l'erreur de modélisation des REA et des RF.

### 4.1 Classification des zones en zones d'irrégularités et éléments architecturaux

Les simplexes extraits par l'algorithme  $\alpha$ -shape (chapitre 3 section 3.2) représentent les contours des régions sur le plan de la façade. Les informations nécessaires pour la conception des MIT diffèrent en fonction du type de région définie par les contours. Pour chaque REA, le concepteur fait appel à quatre informations : la hauteur, la largeur, la profondeur (figure 4.2) et l'incertitude sur ces dimensions. Les régions représentant des RI peuvent être représentées par un maillage ou une interpolation surfacique afin d'avoir une représentation la plus fidèle que possible. Finalement, les contours extérieurs de la façade peuvent être représentés par l'intersection des différents plans des autres façades

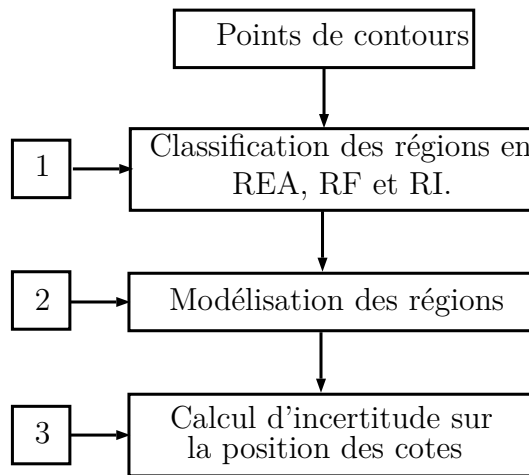


FIGURE 4.1 – Schéma bloc de la méthode de modélisation proposée.

ou plus simplement par une segmentation des points de contours extérieurs.

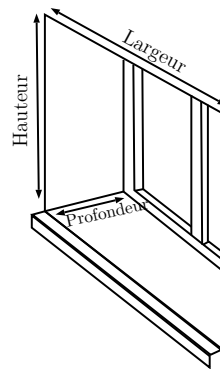


FIGURE 4.2 – Informations nécessaires pour une fenêtre lors de la conception des MIT.

Afin de définir la méthode de modélisation à considérer pour une région, leur classification est réalisée en amont de la phase modélisation. Dans cette optique, on fait appel aux propriétés géométriques d'une région, à savoir : l'aire, la densité et le périmètre. Ces dernières varient en fonction du seuil de consistance  $\delta$  considéré pour la segmentation du nuage de points, ainsi que du paramètre  $\lambda$  considéré pour l'extraction des points de contours.

Le premier paragraphe de cette section présente la méthode de sélection de la propriété géométrique permettant une distinction facile et moins coûteuse en temps de calcul des régions. Le second paragraphe présente la méthode de classification considérée.

#### 4.1.1 Évolution des propriétés géométriques des régions en fonctions du paramètre de segmentation et d'extraction des points de contours

Pour chaque type de région, on a fait varier un paramètre en gardant l'autre paramètre constant.

La variation du paramètre  $\lambda$  a un impact direct sur l'ensemble des points de contours et affecte donc le périmètre et l'aire de la région. Cependant, les changements apportés diffèrent en fonction du type de la région. Pour les RI, la considération d'une faible valeur du paramètre  $\lambda$  élargit le périmètre de la région, et a tendance à partitionner la région en régions plus petites. La considération d'un paramètre trop grand tend à réduire le périmètre de la région, voire à en négliger une grande partie (voir figure 4.3-c). Contrairement aux RI, les REA (voir figure 4.4) restent moins sensibles aux variations de ce paramètre. Il est à noter que le changement du paramètre  $\lambda$  n'affecte pas la densité du nuage des points à l'intérieur des régions.

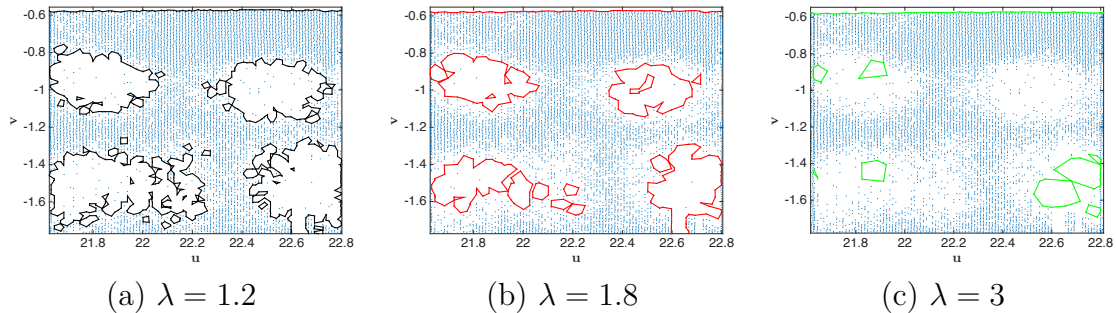


FIGURE 4.3 – Impact du changement du paramètre d'extraction des points des contours sur les zones d'irrégularité.

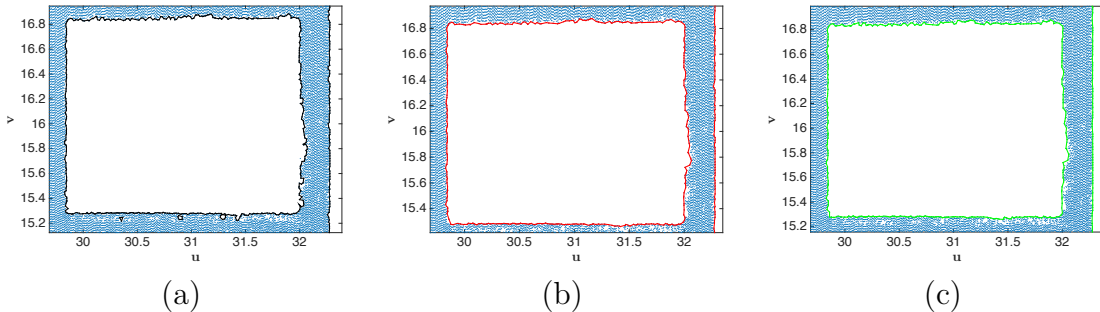


FIGURE 4.4 – Impact du changement du seuil des points de contours sur les contours d'éléments architecturaux

L'étude de l'évolution due à la variation du paramètre de segmentation  $\delta$  consiste à mesurer le nombre de points dans la région pour chaque valeur du seuil. Sur la figure 4.5-c, on remarque que pour une REA, la variation de la densité reste faible et son évolution reste limitée par rapport aux changements du seuil de consistance. Au contraire, une RI montre des variations de densité en relation avec la variation du seuil.

En comparant ces résultats, on constate que la considération de l'aire et/ou du périmètre de la région montrent des résultats tout aussi satisfaisant que la considération de la densité de points. Cependant, la mise en correspondance des régions reste très couteuse en temps de calcul car la chaîne de contours change en fonction du seuil. Aussi, l'utilisation du seuil d'extraction des points de bordure semble peu fiable, car cela risque de définir des



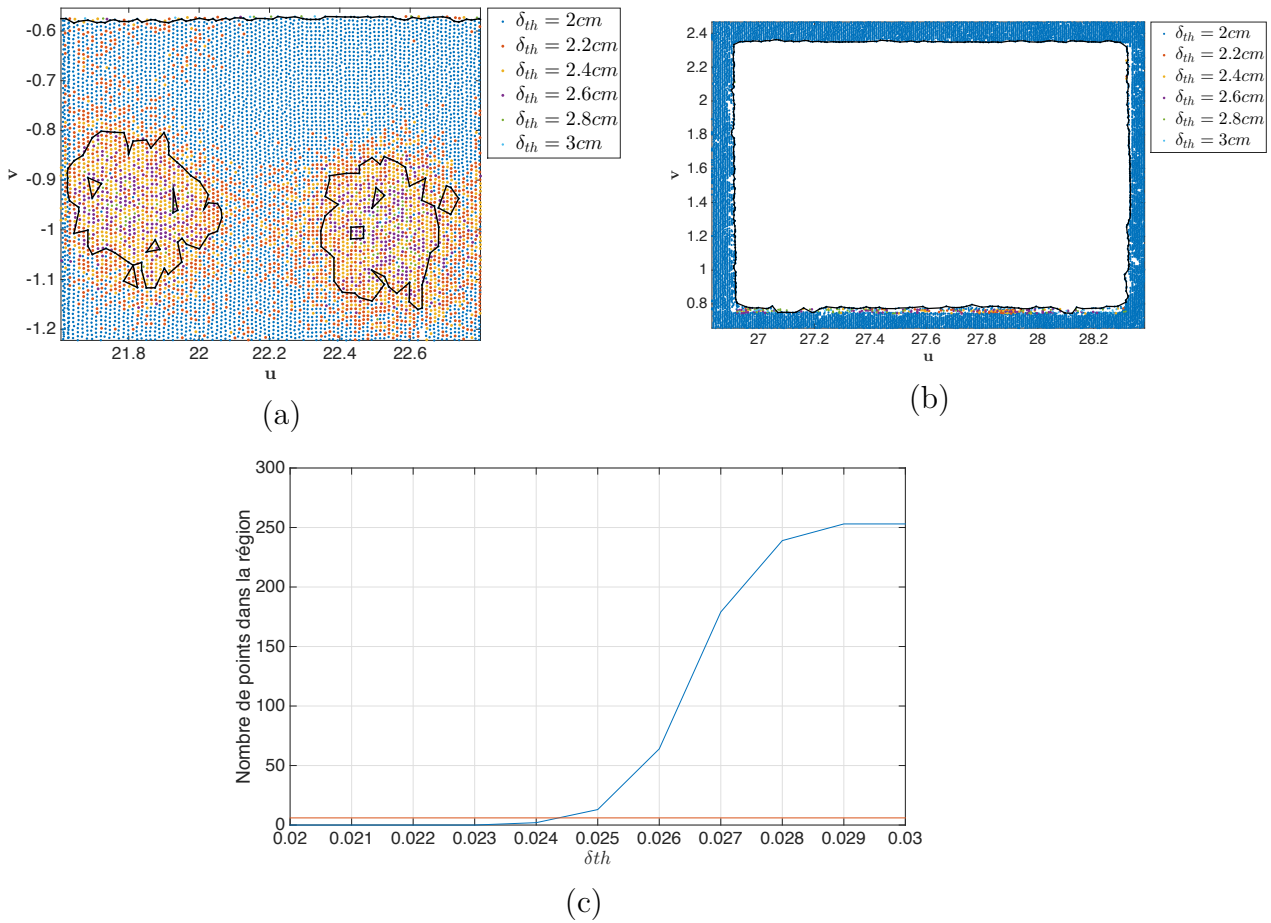


FIGURE 4.5 – Évolution de la densité des régions par rapport au seuil de consistance  $\delta$  : (a) Points dans une RI; (b) Points dans une REA; (c) courbes représentant l'évolution de la densité dans les deux régions (RI : courbe bleue et REA courbe rouge).

régions à faible densité comme des RI. Ainsi, il a été choisi d'effectuer une classification des régions en se basant sur l'aire et la variation de la densité de la région. Le paragraphe suivant présente de manière détaillée l'algorithme de classification proposé.

#### 4.1.2 Méthode de classification des régions

La méthode de classification se compose de trois étapes principales. La première définit l'enveloppe concave de chaque région. Dans la seconde étape, on estime la densité du nuage de points de chaque région pour une valeur du seuil de consistance supérieure à celle considérée dans la phase segmentation. La dernière étape classe les différentes régions en RF, RI et REA.

**Définition des enveloppes concaves des régions** L'ensemble des points de contours  $\mathbb{B}$  extrait par l'algorithme  $\alpha$ -shape définit les bordures des régions du plan de la façade. En effet, les points de contours représentent les sommets d'arêtes ( $\mathbf{e}_k$ ) des simplexes définis par  $\alpha$ -shape à partir de la triangulation de Delaunay (voir chapitre 3 section 2). L'objectif de cette étape est de définir à partir de l'ensemble  $\mathbb{B}$  et des arêtes  $\mathbf{e}_k$  des enveloppes concaves (voir figure 4.6) qu'on notera  $\mathbf{R}$ , représentant les différentes régions sur plan de la façade.

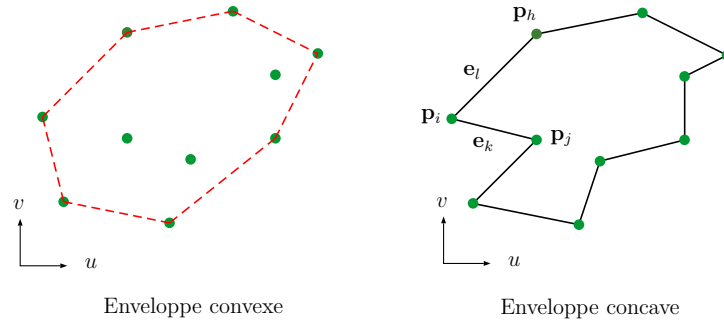


FIGURE 4.6 – Enveloppe concave et enveloppe convexe d'un ensemble de points de contours

La figure 4.7 présente les étapes de la méthode proposée pour définir les enveloppes concaves des différentes régions, qui se compose de trois phases principales.

1. Initialisation : l'algorithme commence par définir le sommet de départ comme le point le plus à gauche de l'ensemble  $\mathbb{B}$  (voir figure 4.7-a).
2. Labellisation : la boucle de labellisation consiste à donner le même label aux sommets non-labellisés. Pour cela, on définit les sommets connexes au dernier point labellisé comme sommets candidats. Le sommet suivant est celui dont l'arrête a l'angle minimal avec le vecteur  $(-1, 0)$  (voir figure 4.7-b et c).
3. Arrêt : l'algorithme s'arrête lorsqu'on ne trouve aucun segment non-labellisé connecté au dernier segment labellisé.

Ces trois étapes sont itérées jusqu'à ce que tous les segments de contours soient labellisés. S'il reste des points de contours non-labellisés, on incrémente le label et on réitère ces trois étapes.

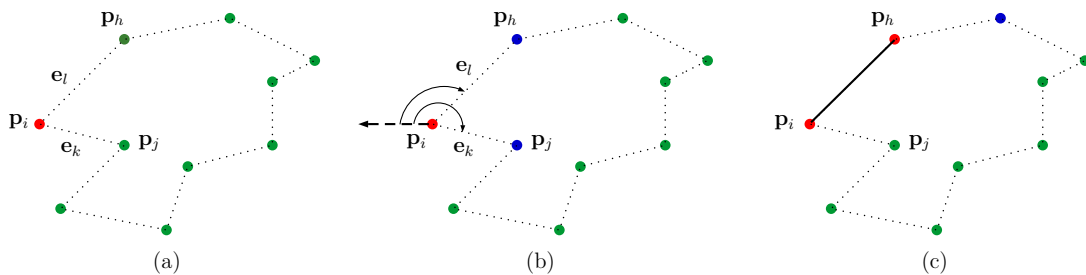


FIGURE 4.7 – Étapes de définition de l'enveloppe concave d'une région.

**Calcul de la densité** L'objectif de cette phase est d'estimer la densité à l'intérieur de la région. La méthode proposée, décrite dans l'organigramme de la figure 4.8 consiste à définir une valeur du seuil de consistance  $\delta_l = \delta + \psi$ . Où  $\psi$  est défini par l'utilisateur. On définit ensuite l'ensemble  $\mathbb{P}'$  des points consistants avec le plan  $\mathbf{n}$  en considérant le seuil  $\delta_l$ . l'ensemble  $\mathbb{P}'$  permet ensuite de calculer la densité d'une région  $\mathbf{R}_j$  par rapport au seuil  $\delta_l$ . On définit  $\mathbf{vd}$ , le vecteur enregistrant la densité des différents régions tel que :

$$\mathbf{vd} = \begin{bmatrix} a_1 \\ \vdots \\ a_q \end{bmatrix}; \quad (4.1)$$

où  $q$  est le nombre de régions et  $a_j$  la densité de la région donnée par :

$$a_j = \frac{pr_j}{aire(\mathbf{R}_j)} \quad (4.2)$$

avec  $pr_j$  cardinalité de l'ensemble de points à l'intérieur de la région, celui-ci est défini par l'algorithme Edge Intersect [Shi62].

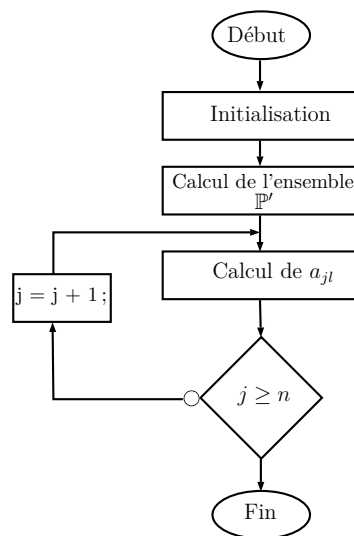


FIGURE 4.8 – Organigramme de la méthode de calcul de la densité.

**Classification des régions** Cette étape vise à classer chaque région en fonction de sa densité et de son aire. Ainsi, les régions sont classées en 3 différentes catégories (RF, RI et REA).

- Une RF est délimitée par les contours extérieurs du plan de la façade. Elle se caractérise par une forte densité et une très grande surface.
- Une RI est caractérisée par une forte densité ainsi qu'une surface généralement plus petite que celle d'une RF.
- Contrairement aux types précédents, une REA se caractérise par une faible densité, et par une surface plus petite que celle des régions de façade.

Les trois classes de régions peuvent donc être distinguées selon deux variables : l'aire et la variance de densité. On définit donc deux seuils :  $A_{th}$  le seuil de l'aire d'une région et  $D_{th}$  le seuil de la densité. La figure 4.9 présente une illustration de l'espace de paramètres définissant les différentes classes de régions. La figure 4.10 présente les différentes régions labellisées : en rouge les RI, les REA en vert et les RF en bleu.

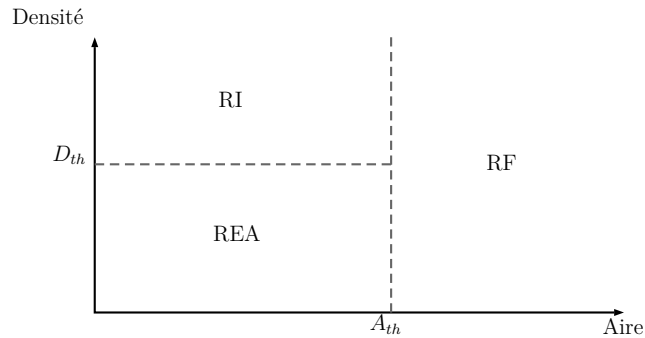


FIGURE 4.9 – Espace paramétrique représentant la classification des régions en fonction du seuil de l'aire et la variation de la densité par rapport au seuil de consistance  $\delta$

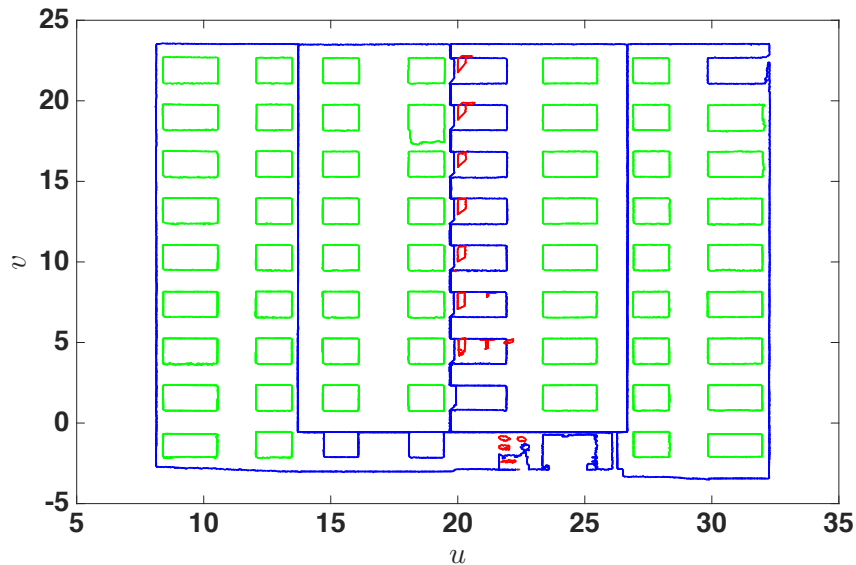


FIGURE 4.10 – Résultats de la classification des régions utilisant la variation de densité par rapport au seuil de consistance comme propriété discriminante.

## 4.2 Modélisation des éléments de façade

Pour chaque type de régions, une approche de modélisation spécifique est considérée. Les REA sont représentées par des polyèdres, les RF par des polygones et les RI par des maillages triangulaires. La suite de cette section présente la méthode de modélisation pour chaque type de régions.

### 4.2.1 Modélisation des éléments architecturaux

Les Éléments Architecturaux (EA) peuvent être généralement classés en deux catégories, les intrusions telles que les fenêtres et les portes, et les extrusions telles que les appuis de fenêtres et les balcons. Les intrusions et les extrusions se situent respectivement en arrière et en avant du plan de la façade selon la normale à ce plan. Ces EA sont généralement composés de primitives géométriques simples, majoritairement planes dans le cas de bâtiments dédiés aux logements sociaux. Ainsi, il a été choisi de représenter les EA par les arêtes de polyèdres, où chaque arête est constituée par deux sommets (figure 4.11). Chaque polyèdre est constitué d'un ensemble de polygones intérieurs ( $P_{int}$ ) au niveau de la profondeur de l'intrusion (i.e : cadre de fenêtre), et/ou polygone extérieurs  $P_{ext}$  au niveau de l'extrusion (rebord de fenêtre), des polygones latéraux ( $P_g$  et  $P_d$ ) ainsi que des polygones inférieurs et supérieurs ( $P_{inf}$  et  $P_{sup}$ ). La figure 4.12 présente une illustration des différents polygones cités.

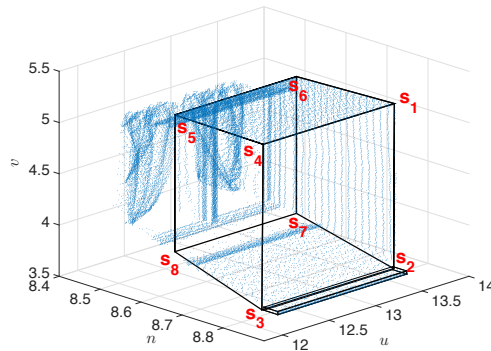


FIGURE 4.11 – Représentation de fenêtre par un polyèdre

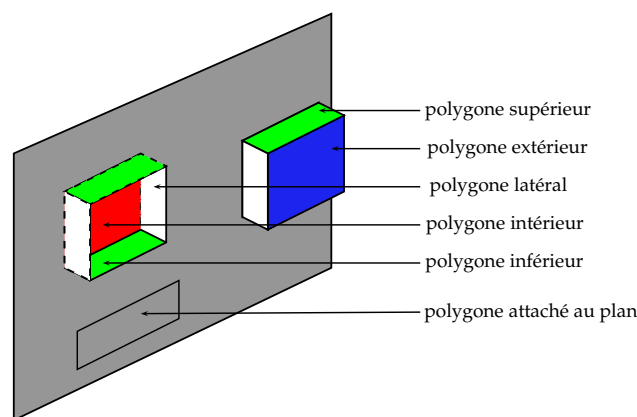


FIGURE 4.12 – Illustration des différents polygones constituant un polyèdre représentant une REA.

La méthode de modélisation définit les différents polygones constituant le polyèdre de l'EA. Elle se compose de trois étapes principales. La première étape consiste à définir

un polygone au niveau de la façade du bâtiment. La seconde étape définit le polygone au niveau de l'intrusion/extrusion à partir du sous ensemble de points à l'intérieur du polygone au niveau du plan de la façade. La dernière étape consiste à définir les polygones latéraux ainsi que les polygones inférieurs et supérieurs à partir des polygones issus des étapes 1 et 2.

Afin de faciliter l'explication des différentes étapes, on introduit les notation suivantes. On notera un polygone  $\mathbf{V}$  par :

$$\mathbf{V} = [\mathbf{s}_1 \dots \mathbf{s}_t \dots \mathbf{s}_k]$$

où  $\mathbf{s}_t$ , représentes les sommets du polygone défini dans le référentiel attaché au plan, et ordonnés dans le sens horaire.

$$\mathbf{s}_t = \begin{pmatrix} n_t \\ u_t \\ v_t \end{pmatrix} \quad (4.3)$$

Les paragraphes suivants décrivent les différentes étapes de la méthode de modélisation d'un EA.

**Définition du polygone attaché au plan de la façade** Pour définir le polygone attaché au plan de façade, la chaîne de contours représentant l'EA est segmentée en un ensemble de droites de contours par la transformée de Hough [Hou62]. Les sommets du polygone sont ensuite définis comme les intersections des droites de contours. La figure 4.13 présente les différentes phase de définition des polygones. La figure 4.13 présente les plans segmentés (bleu, jaune et orange) ainsi que leurs chaînes de contours (en noir). Le résultat de leur segmentation présenté dans la figure 4.13-b et les intersections dans la figure 4.13-c. La figure 4.13-d montre le résultat de la reconstruction des différents polygones attachés aux plans de façades.

**Définition des polygones intérieurs et extérieurs** Les polygones  $P_{int}$  et  $P_{ext}$  permettent de situer les extrémités du polyèdre représentant l'EA. Ils sont définis à partir de l'ensemble des points  $\mathbb{E}$ , délimités par les sommets du polygone attaché au plan de la façade. Une marge est introduite sur cet espace afin d'éviter une troncature des éléments qui peuvent être liés à ce polygone (figure 4.14). Pour le calcul des coordonnées  $n_t$  des sommets du polygone  $P_{int}$  ou  $P_{ext}$ , on suppose que le plan est vertical au niveau de l'intrusion/extrusion, et que ses arêtes sont verticales et horizontales.

Pour placer les polygones, la méthode proposée consiste à définir un ensemble de plans parallèles au plan de la façade à des intervalles prédéfinis. Chaque plan est situé à une position  $n_t$  par rapport à la normale au plan de façade. La figure 4.15 donne un exemple pour une fenêtre où les plans sont espacés de 2cm. Pour chacun des plans, on détermine l'ensemble des points consistants  $\mathbb{P}'$  (représenté par une couleur différente sur la figure 4.15), ainsi que l'ensemble  $\mathbb{B}$  de points de contours correspondant par l'algorithme  $\alpha$ -shape.

La position en  $n$  des polygones  $P_{int}$  et  $P_{ext}$  est déterminée à partir du rapport  $\mathcal{T}$  du nombre de points consistants sur le nombre de points de contours. On constate que ce

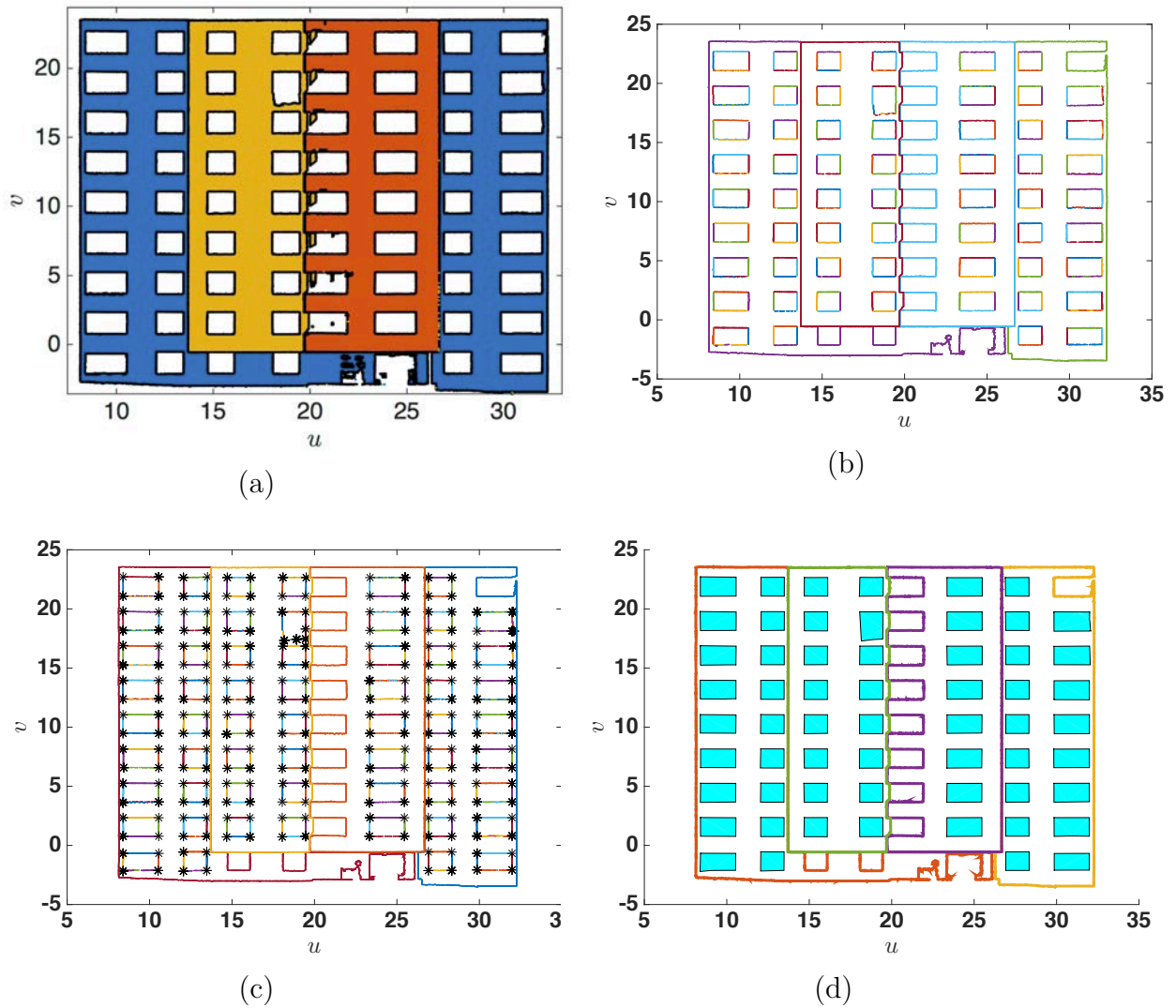


FIGURE 4.13 – Résultats des différentes étapes. (a) Plans segmentés. (b) Segmentation des chaînes de contours en droites. (c) Points d'intersections des droites segmentées pour chaque chaîne de contours. (d) Polygone défini à partir des points d'intersections.

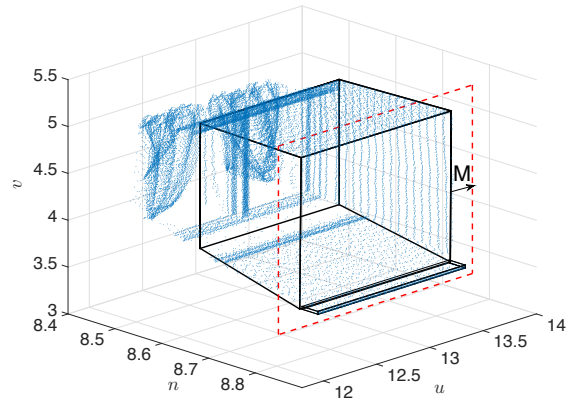


FIGURE 4.14 – Marge  $M$  considérée pour la définition des points à l'intérieur du polygone.

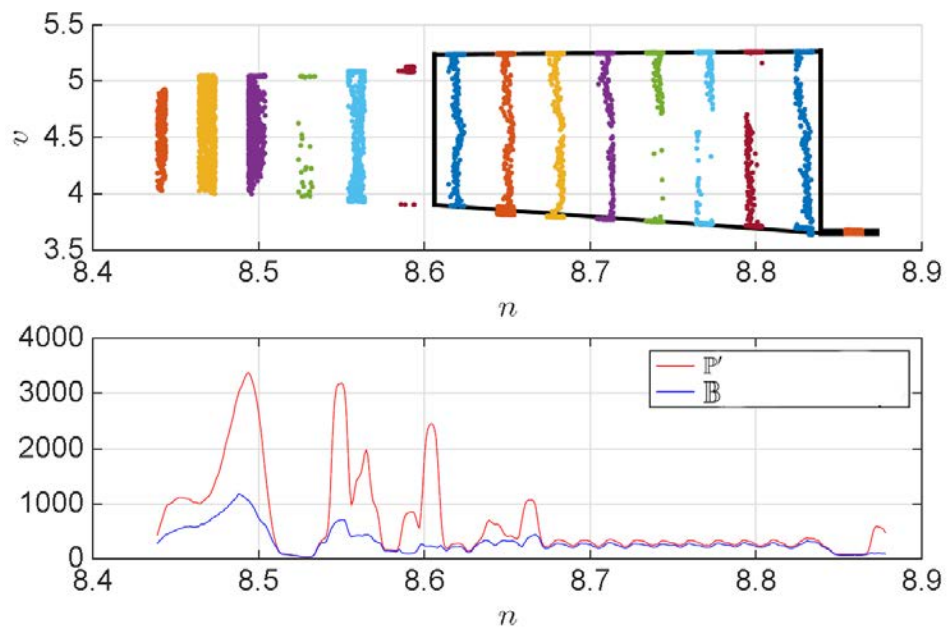


FIGURE 4.15 – Évolution de la cardinalité de l'ensemble  $\mathbb{P}'$ , points consistants avec les plans verticaux.



ratio  $\mathcal{T}$  (équation 4.4) est élevé lorsque le plan considéré est situé au niveau d'un appui de fenêtre ou linteau (extrusion) ou du cadre de la fenêtre (intrusion) (figure 4.16). On définit alors la position de  $P_{ext}$  comme la position en avant du plan de façade maximisant le ratio  $\mathcal{T}$ . De la même manière, on définit la position selon  $n$  de  $P_{int}$  comme la position en arrière du plan maximisant le même ratio (équation 4.5).

$$\mathcal{T} = \frac{|\mathbb{P}'|}{|\mathbb{B}|} \quad (4.4)$$

$$n_t = \arg \max_n \mathcal{T}(n); \quad (4.5)$$

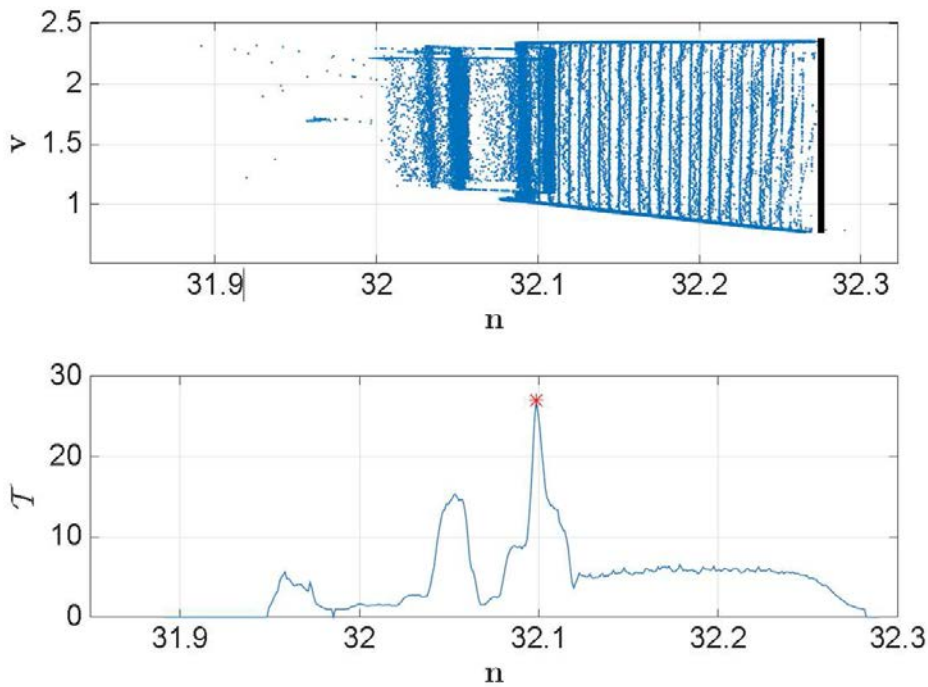


FIGURE 4.16 – Évolution du critère  $\mathcal{T}$

Les coordonnées  $u_t$  et  $v_t$  de chaque sommet sont ensuite déterminées à partir des extréma de l'ensemble de points consistants avec le plan à la position  $n_t$ .

**Définition des polygones  $P_g$ ,  $P_d$ ,  $P_{inf}$  et  $P_{sup}$**  Nous utilisons deux méthodes pour définir ces polygones. L'une est adaptée aux EA dont l'intrusion ou l'extrusion a une surface importante (figure 4.17-a). L'autre est réservée au EA ayant des intrusions/extrusions de petites dimensions (figure 4.17-b). Dans la première, les polygones latéraux, inférieur et supérieur sont définis par chaînage des sommets du polygone attaché au plan de la façade et ceux du polygone de l'intrusion ou l'extrusion. La seconde méthode est dédiée aux polygones pour les extrusions ou intrusions de faible surface. Dans ce cas les polygones latéraux, inférieur et supérieur sont définis par projection des sommets du polygone au niveau de l'intrusion ou l'extrusion sur le plan de la façade. Un exemple de modèle polyédrique est présenté dans la figure 4.18.

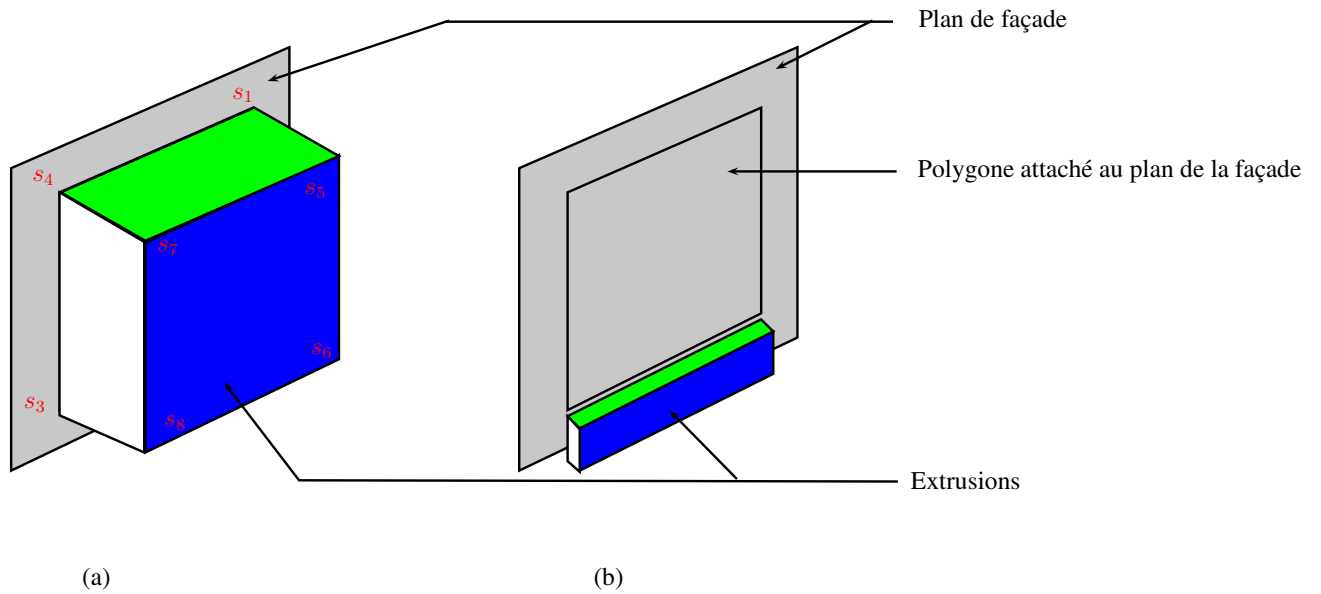


FIGURE 4.17 – Illustration des deux méthodes de chainage des sommets du polygone intérieur/extérieur et du polygone attaché au plan de la façade. (a) Chaîne par liaison des sommets les plus proches. (b) Chainage par projection horizontale des sommets sur le plan de la façade.

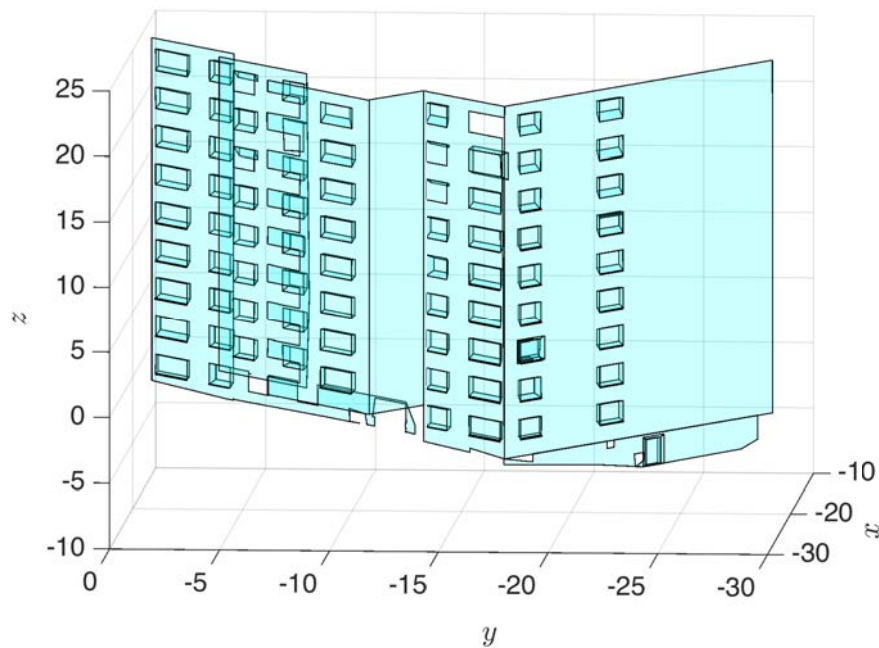


FIGURE 4.18 – Résultat de la reconstruction du modèle polygonal

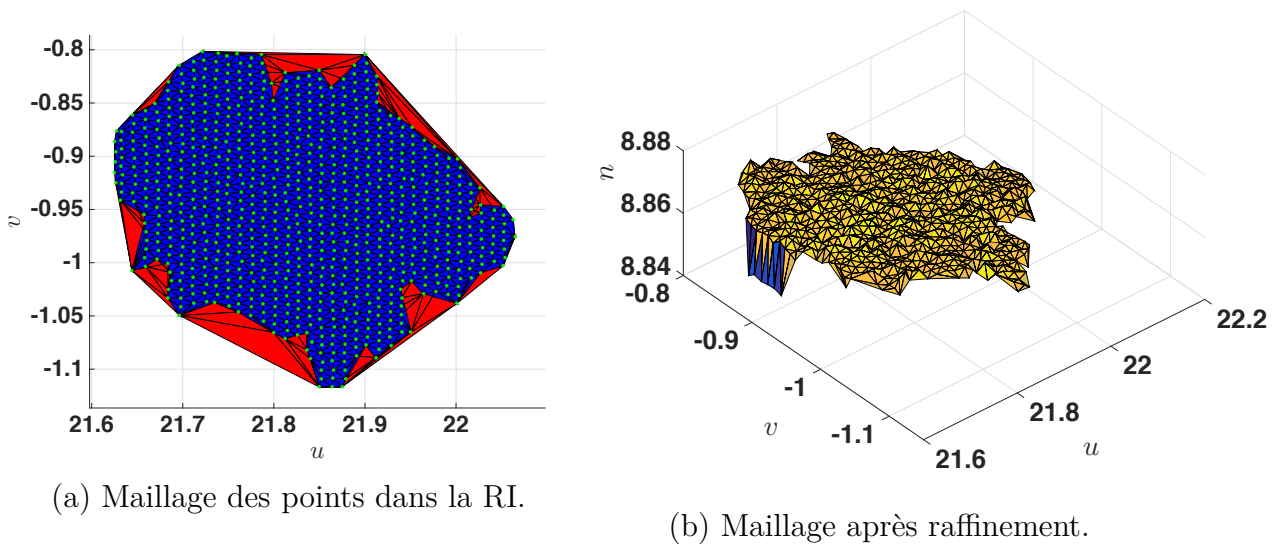


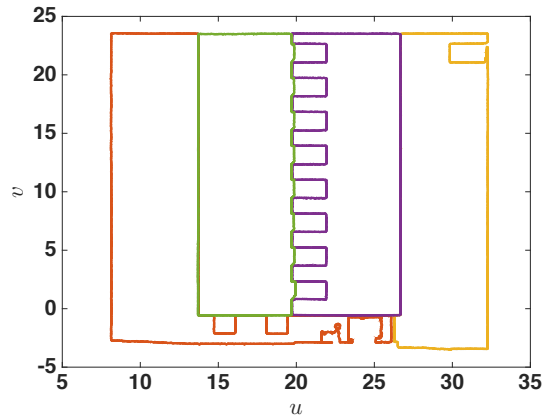
FIGURE 4.19 – Modélisation d'une RI.

## 4.2.2 Modélisation des régions d'irrégularités

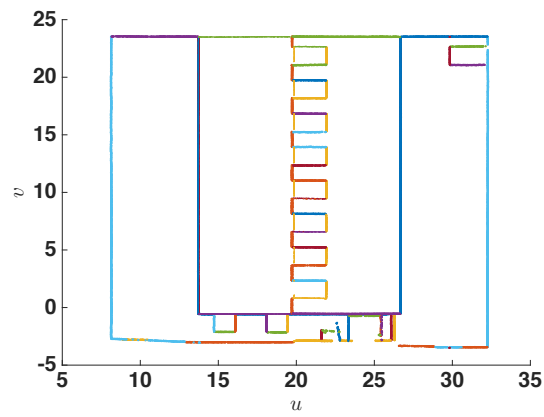
Afin de représenter les RI de façon fidèle, on utilise un maillage triangulaire des points à l'intérieur de la région. La méthode de reconstruction des RI se compose de 3 étapes. Dans un premier temps, on définit l'ensemble  $(\mathbb{P}')$  de points consistants avec le plan considérant un seuil  $\delta_l$ . On définit alors l'ensemble des points situés à l'intérieur de la RI  $(\mathbb{E})$  par l'algorithme de [Shi62]. La seconde étape consiste à définir un maillage triangulaire modélisant la surface de la RI. Le maillage contient l'ensemble des triangles définis dans l'enveloppe convexe de la RI. La dernière étape consiste à raffiner ce résultat en supprimant les triangles situés à l'extérieur de l'enveloppe concave. La figure 4.19-a représente cette dernière étape, où les triangles rouges sont à l'extérieur du polygone défini par l'enveloppe concave. La suppression se fait en comparant l'aire du triangle avec celle de l'intersection du triangle avec le polygone défini par l'enveloppe concave de la RI. Si l'aire de l'intersection est inférieure à celle du triangle, ce dernier est supprimé. La figure 4.19-b représente le maillage des points de la région après la phase raffinement.

## 4.2.3 Modélisation des régions de façades

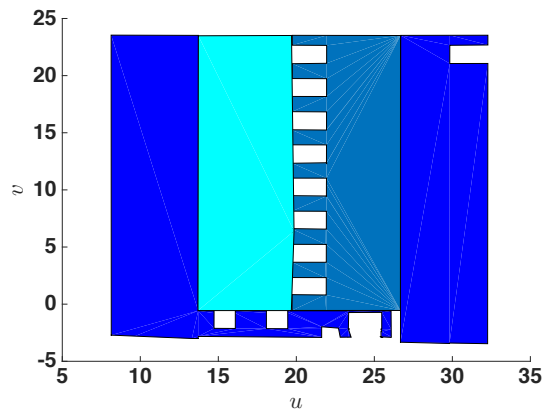
La reconstruction des RF se compose de trois étapes principales. Les deux premières visent la reconstruction polygonale des plans de façade et la troisième concerne l'intersection des plans des façades du bâtiment. Dans la première, les points de contours de chaque région sont segmentés en droites de contours (voir figure 4.20-a et b). La seconde étape consiste à définir des polygones à partir des intersections des droites de contours (voir figure 4.20-c).



(a) Points de contours des régions de façades.



(b) Segmentation des points de contours en droites.

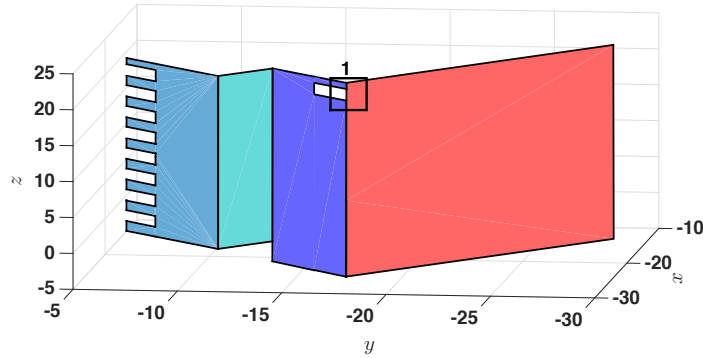


(c) Polygones définissant des régions des façades (plans bleu, jaune et orange) à partir des intersections des droites.

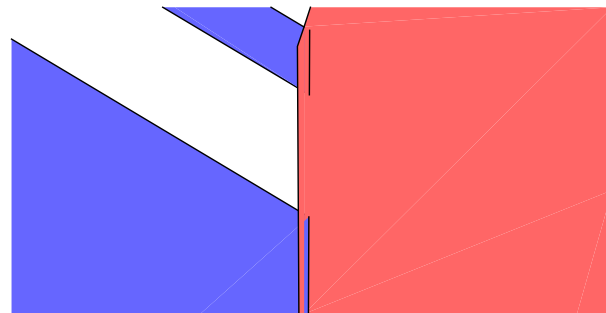
FIGURE 4.20 – Processus de définition des polygones des façades à partir des points de contours.

La reconstruction de chaque polygone de RF est réalisée dans le référentiel attaché au

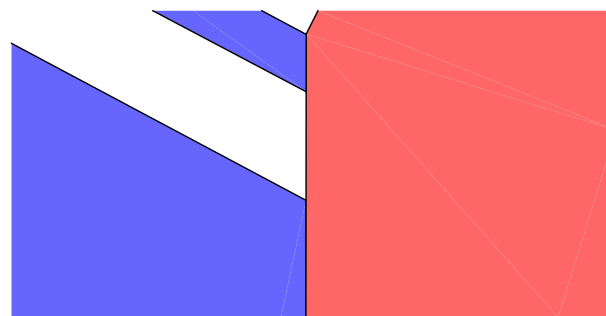
plan indépendamment des autres plans des façades. Ce procédé conduit à des erreurs de reconstruction au niveau des intersections des plans, une fois les polygones ramenés dans le référentiel absolu. La figure 4.21-b présente un exemple d'erreur au niveau de l'intersection. Pour corriger cette erreur, l'équation de la droite au niveau des intersections des plans est calculée à partir de leurs équations cartésiennes. Les sommets des polygones au niveau de l'intersection sont ensuite projetés sur la droite, la figure 4.21-c présente les polygones des régions de façade après projection sur la droite d'intersection.



(a) Polygones définissant des régions de façade.



(b) Intersection des polygones sans prise en compte de l'intersection des plans.



(c) Intersection des polygones avec prise en compte de l'intersection des plans de façades.

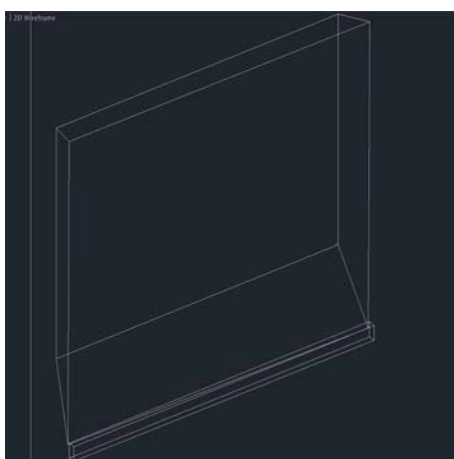
FIGURE 4.21 – Prise en compte des intersections entre plans.

A l'issue de ces étapes de modélisation, une représentation filaire des différentes façades

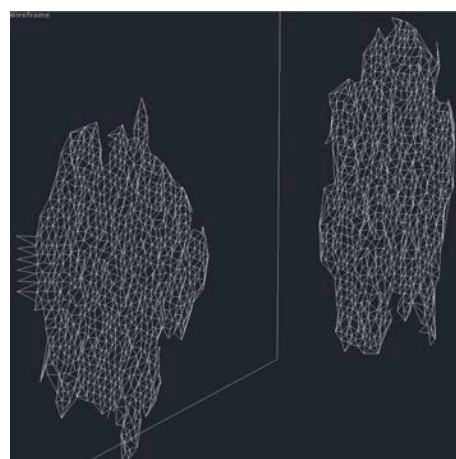
est exportée en fichier CAO en format DXF. Les figures suivantes présentent le résultat de l'exportation du modèle filaire du bâtiment. La figure 4.22 présente un modèle filaire du bâtiment et la figure 4.23 représente des zooms sur des régions du type REA (figure 4.23-a) et RI (figure 4.23-b).



FIGURE 4.22 – Modèle filaire exporté sous format DXF (visualisation sur logiciel Autocad)



(a) Zoom sur une REA



(b) Zoom sur une RI

FIGURE 4.23 – Exemple de reconstruction filaire de fenêtre (a) et d'une région d'irrégularité (b)

## 4.3 Erreur de modélisation

Bien que le modèle CAO reconstruit à partir de nuages de points reste plus fidèle aux bâtiments que le modèle conceptuel, celui-ci contient des erreurs de reconstruction dues à plusieurs facteurs. Les travaux de [Bou10] se sont penchés sur les causes d'erreur dans le modèle CAO, qui peuvent être classées en deux catégories.

- Les erreurs d'acquisition : ce sont les erreurs résultants du processus d'acquisition. On retrouve d'une part l'erreur de position 3D des points. Celle-ci peut être causée par le système de mesure de distance, de mesure d'angle et/ou du processus de recalage du nuage de points (voir chapitre 1 sections 1.1 et 1.2). D'autre part la résolution de numérisation de la surface influence sur le positionnement des cotes des éléments architecturaux. Plus le pas d'échantillonnage de la surface est petit, plus la précision de positionnement est grande [LBG12]. Bien que l'utilisateur définisse une haute résolution d'acquisition, celle-ci varie en fonction de la distance et de l'angle d'incidence (voir chapitre 3 section 1). Ainsi la précision des cotes des éléments architecturaux varie également en fonction de ces deux facteurs.
- Les erreurs résultants du paramétrage des algorithmes précédant la phase de modélisation, la segmentation et l'extraction des points de contours. Le seuil de consistance  $\delta$  représente le paramètre principal de la segmentation par RANSAC. La considération d'un seuil très large conduit à une sous-segmentation, où plusieurs éléments architecturaux sont vus comme coplanaires. À l'inverse, la considération d'un seuil de consistance trop faible induit une sur-segmentation où des points coplanaires sont segmentés en plusieurs plans. Le paramètre  $\lambda$  influence de la même manière l'extraction des points de contours. Une valeur élevée définit les points à l'intérieur de la surface comme points de contours, une valeur basse conduit à l'omission des éléments architecturaux et des régions d'irrégularités ayant une faible surface.

Les différents polygones sont reconstruits à partir d'un ensemble de sommets dont les coordonnées sont entachées des erreurs citées précédemment. Ces erreurs sont présentées dans la figure 4.24, où  $\xi_n$  le biais d'estimation du plan par rapport au plan réel, dépend en majeure partie du seuil de consistance choisi, de la rugosité de la surface, ainsi que du bruit d'acquisition. Les erreurs  $\xi_u$  et  $\xi_v$  dépendent quant à elles en majeure partie du pas d'échantillonnage de la surface ainsi que du seuil d'extraction des points de contours  $\alpha_{th}$ .

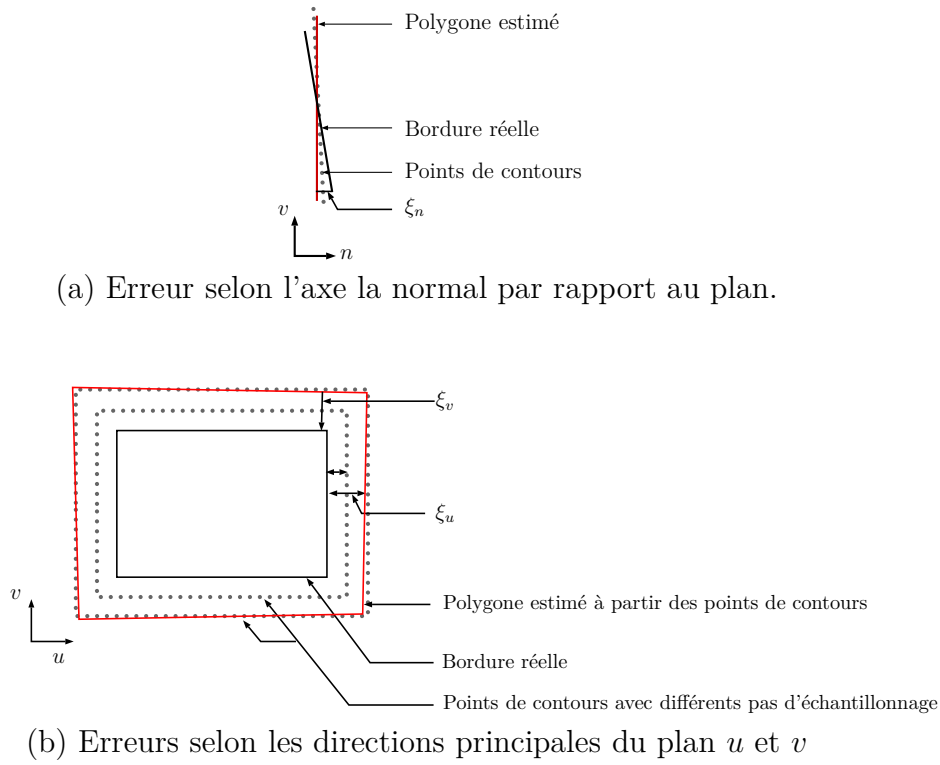


FIGURE 4.24 – Erreurs de modélisation

Ces erreurs représentent le biais d'estimation d'un sommet par rapport au sommet réel ( $\bar{\mathbf{s}}_t$ ). Ainsi, l'erreur d'estimation de la position d'un sommet peut être exprimée par l'équation (4.6).

$$|\bar{\mathbf{s}}_t - \mathbf{s}_t| \leq \sqrt{\xi_n^2 + \xi_u^2 + \xi_v^2} \quad (4.6)$$

### 4.3.1 Approximation de l'erreur selon les axes principaux du plan

Les erreurs sur les coordonnées par rapport à chaque axe  $\mathbf{u}$  et  $\mathbf{v}$  sont majorées par le pas d'échantillonnage. Comme illustré dans la figure 4.24, plus le pas d'échantillonnage est petit, plus les points de contours s'approchent des bordures réelles de l'élément architectural [Bou10]. Ainsi, la somme des erreurs sur la position du sommet est généralement inférieure au pas d'échantillonnage du SLT. On peut définir l'expression de l'erreur de position selon chaque axes par :

$$\begin{cases} \xi_u = |\bar{u}_t - u_t| \leq \Delta_u(\theta_t) \\ \xi_v = |\bar{v}_t - v_t| \leq \Delta_v(\phi_t) \end{cases} \quad (4.7)$$

où  $\theta_t$  et  $\phi_t$  représentent respectivement l'azimut et l'élévation par rapport à la station la plus proche du sommet.

Afin d'approximer cette erreur, la méthode consiste à superposer le modèle polygonal et la grille polyédrique estimée dans le chapitre 3 (voir figure 4.25). Pour chaque patch de la grille, on dispose d'une valeur moyenne du pas d'échantillonnage de la surface. Ainsi



l'erreur de chaque sommet du polygone sera calculée en fonction du patch dans lequel il se trouve. L'erreur de chaque coordonnée peut donc être approximée par :

$$\begin{cases} \xi_u = |\bar{u}_t - u_t| \leq \Delta_u(\bar{\theta}_r) \\ \xi_v = |\bar{v}_t - v_t| \leq \Delta_v(\bar{\phi}_q) \end{cases} \quad (4.8)$$

où  $\Delta_u(\bar{\theta}_r)$  et  $\Delta_v(\bar{\phi}_q)$  représentent respectivement le pas d'échantillonnage selon l'axe  $\mathbf{u}$  et l'axe  $\mathbf{v}$  de la surface mesurée aux centres des patches polyédraux  $r$  et  $q$ .

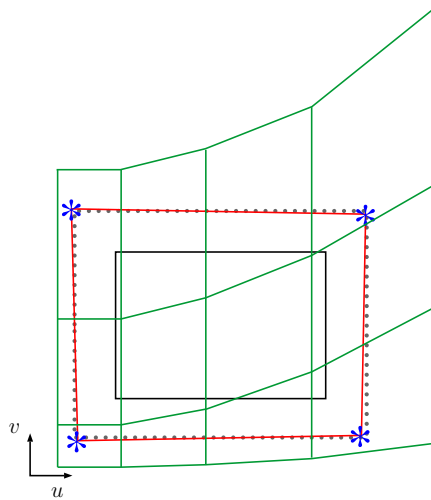


FIGURE 4.25 – Illustration de superposition du polygone et de la grille polyédrique.

### 4.3.2 Intégration de l'erreur sur le modèle CAO

La prise en compte de l'erreur sur le modèle permet de prévoir les systèmes optimaux de fixation et de compensation de l'erreur, ainsi que l'amplitude d'intervention sur le chantier de pose des MIT. L'intégration de l'erreur de modélisation est généralement plus facile sur les maquettes numériques (BIM-CAD au standard IFC) car les objets architecturaux sont représentés sous forme de structure permettant d'intégrer plusieurs informations sémantiques. Cependant, l'intégration de cette information sur un fichier CAO de bas niveau reste moins évidente car le fichier ne permet d'intégrer que les informations géométriques et la texture de surface.

Dans ces travaux, il a été choisi comme solution d'intégration de l'erreur d'exporter deux modèles sur le même fichier. De manière plus explicite, le fichier CAO représentant la façade de bâtiment sera composé de deux calques superposés. Le premier calque contiendra le modèle filaire estimé directement à partir des points de contours des façades, alors que le second modèle représente les modèles prenant en compte l'erreur d'estimation. Un exemple de cette exportation est présenté dans la figure 4.26, qui présente les deux modèles filaires d'une fenêtre, où les polyèdres en rouge représentent le modèle intégrant l'erreur de modélisation.

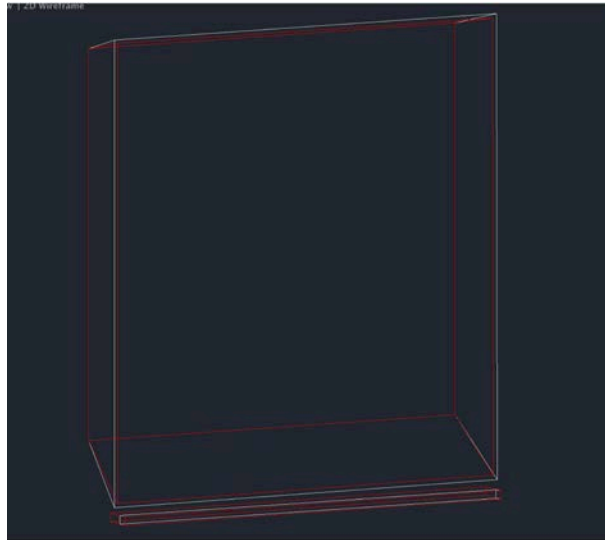
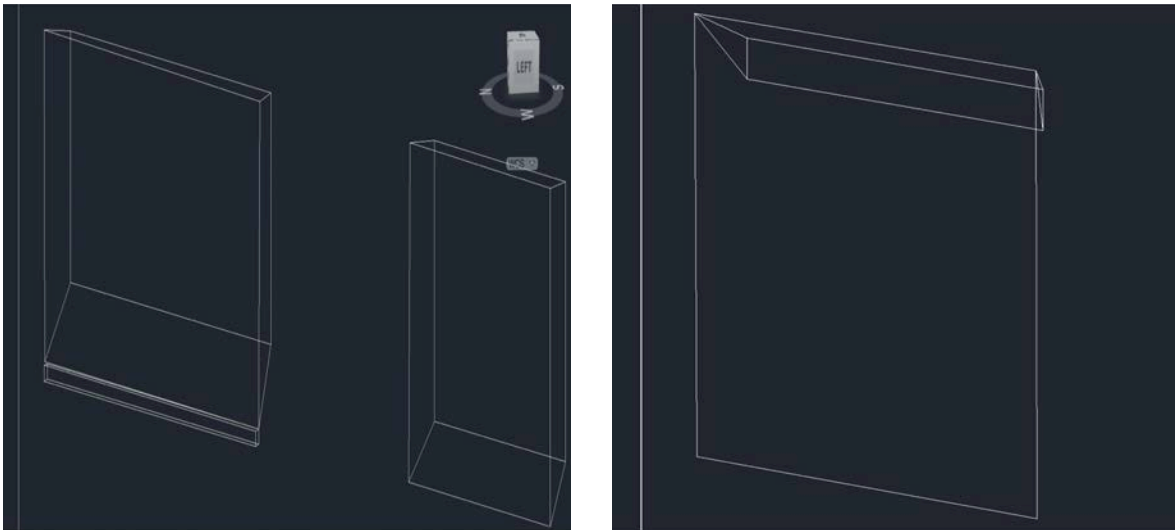


FIGURE 4.26 – Superposition des deux modèles avec et sans prise en compte de l’erreur de modélisation, respectivement rouge et blanc.

## Conclusion

Ce chapitre présente une méthode de reconstruction de modèles filaires de façade de bâtiments. Celle-ci se compose de trois étapes principales. Tout d’abord, les régions définies sont classées en trois types REA, RF et RI. Les REA sont modélisées par des polyèdres, les RF par des polygones et les RI par un maillage triangulaire des points dans la région. La dernière étape consiste à approximer et intégrer l’erreur de modélisation sur le modèle CAO, en la limitant aux régions représentant des éléments architecturaux.

Les limites de cette méthode sont généralement liées au processus d’acquisition. En effet, la fiabilité de détection des EA reste fortement liée au pas d’échantillonnage de la surface. Un exemple de ce cas est présenté dans la figure 4.27-a où l’appui de la fenêtre à droite n’est pas détecté. Aussi la reconstruction du polyèdre correspondant reste fortement liée à l’effet d’occlusion. Un exemple de ce cas est présenté dans la figure 4.27-b où on remarque que le polyèdre représentant la fenêtre est réduit à la partie non occultée.



(a) Défaut de détection de linteau sur la fenêtre à droite.

(b) Erreur de reconstruction de polyèdre due à l'occlusion de la fenêtre.

FIGURE 4.27 – Défauts de reconstruction liées à la phase d'acquisition.

# Conclusion et perspectives

Les travaux menés dans le cadre d'une collaboration entre le CRITTBois et le CRAN ont permis de développer une méthode automatique de modélisation 3D d'un bâtiment à partir de nuages de points obtenus par Scanner LASER Terrestre (SLT). L'objectif était de réaliser une reconstruction du modèle filaire de façades de bâtiments pour des applications de rénovation ou réhabilitation thermique par l'extérieur.

Le premier chapitre présente un état de l'art du domaine concerné. Celui-ci a été axé sur trois phases principales du processus de reconstruction, à savoir l'acquisition, le recalage des nuages de points et les méthodes de reconstruction. Plusieurs technologies sont utilisées par les SLT pour l'acquisition. Chaque combinaison entre un système de mesure de distance et de balayage angulaire est dédiée à un domaine en particulier (topographie, numérisation d'objets de petites tailles, numérisation des milieux industriels.etc). Bien qu'ils offrent plusieurs avantages par rapport à l'acquisition par stations totales et la photogrammétrie en terme de temps d'acquisition et de précision de mesure, les SLT sont limités par l'effet d'occlusion. Cet effet peut être réduit par l'acquisition à partir de plusieurs points de vue. Les nuages de points résultants sont ensuite ramenés dans un même référentiel par une procédure de recalage. Une phase de reconstruction peut être alors réalisée selon deux types. Le premier concerne les méthodes de reconstruction non-paramétriques visent à représenter le bâtiment par un maillage triangulaire. L'avantage majeur de ce type de représentation est de mettre en avant les détails fins de la surface numérisée. Cependant, le volume de données reste conséquent ce qui rend la manipulation et l'interprétation du modèle difficile pour l'utilisateur. Un autre inconvénient de ce type de représentation est l'impossibilité d'imbriquer des informations sémantiques sur le modèle, car aucune reconnaissance d'objets ou de formes n'est réalisée. Le deuxième type inclut les méthodes de reconstruction paramétriques. Ces méthodes permettent de faire une reconstruction des éléments architecturaux par des primitives géométriques simples, ce qui réduit le volume de données, et permet l'imbrication d'informations sémantiques sur le modèle CAO. Cependant, les modèles issus de ces méthodes de reconstruction restent trop grossiers et contiennent peu de détails sur la surface des objets.

La méthode de reconstruction proposée dans ce manuscrit représente un compromis entre les deux méthodes citées précédemment. Elle consiste à adapter la méthode de modélisation des régions dans le nuage de points en fonction du niveau de détails recherché pour chaque type de région. La méthode se compose de trois phases principales.

La première phase qui fait l'objet du chapitre 2, consiste à segmenter le nuage de points en un ensemble de plans représentant les plans de façade du bâtiment. La contribution majeure de la méthode proposée réside dans la diminution du temps de calcul par prise en

compte de la colorimétrie dans la segmentation par RANSAC. Pour cela, la réduction de la complexité algorithmique est réalisée sur deux niveaux. Le premier consiste à faire un tirage non-uniforme des points constituant l'échantillon minimal sélectionné à partir d'une seule classe colorimétrique. Le second niveau, consiste à réduire le nombre d'itérations. Pour cela, le calcul de la probabilité de tirer un échantillon minimal non-contaminé par des outliers est limité aux points de la classe colorimétrique considérée.

Le chapitre 3 présente la seconde phase, qui consiste à modéliser le pas d'échantillonnage de la surface et à extraire les points de contours des différents éléments de façade. La principale difficulté que notre méthode vise à lever tient dans le fait que les algorithmes d'extraction des points de contours existants utilisent un seuil global, alors que le pas d'échantillonnage de la surface varie en fonction de la distance et de l'angle d'exposition de la surface par rapport au SLT. La considération d'un seuil défini manuellement sans prendre en compte l'évolution du pas d'échantillonnage réduit considérablement la fiabilité de détection des points de contours. Afin de remédier à ce problème une approximation du pas d'échantillonnage de la surface a été proposée. Le but est de réduire les erreurs d'extraction des points de contours en considérant un seuil local, déduit de l'approximation du pas d'échantillonnage. L'intérêt d'utiliser un seuil local a été montré par des tests comparant les ensembles de points de contours extraits en considérant un seuil global et ceux extraits en utilisant un seuil local.

La dernière phase décrite dans le chapitre 4 consiste à modéliser les différents éléments de façade représentés par leurs points de contours. Cette phase est composée de trois étapes. La première consiste à classer les régions définies par les points de contours en trois types de régions REA, RF et RI. Ensuite, chaque région est modélisée de manière à fournir les informations nécessaires pour l'utilisateur. Ainsi, les REA sont représentées par des polyèdres, les RF par des polygones et les RI par un maillage triangulaire des points à l'intérieur de la région. La dernière étape consiste à intégrer l'erreur de modélisation des REA sur le modèle CAO, permettant ainsi à l'utilisateur de prévoir la marge d'erreur nécessaire lors de la conception des modules d'isolation thermique.

Ces travaux se sont concrétisés par une application développée sous Matlab, de niveau TRL 6, permettant de répondre aux exigences fixées par les industriels. Un projet de maturation a été financé par l'université de lorraine ainsi que BPI-France afin de produire une application indépendante développée en C++ et Qt atteignant un niveau TRL 8.

En terme de perspectives, la méthode peut être étendue pour traiter des architectures plus complexes en intégrant plusieurs primitives géométriques (cylindres, cône, sphères, etc.) dans l'algorithme de RANSAC. Cet enrichissement peut aussi être apporté dans la phase de modélisation en utilisant la transformation de Hough généralisée pour la détection de formes géométriques arbitraires.

Une texturation peut être apportée sur le modèle géométrique en réalisant un recalage entre des images photographiques et le modèle CAO du bâtiment. Aussi, le modèle géométrique représente une base fiable sur laquelle peuvent être intégrées des informations sémantiques sur les REA. Une phase d'attribution d'informations sémantiques pour chaque REA en fonction de sa topologie peut être considérée. Les informations sémantiques ne peuvent être imbriquées sur un fichier du type dxf, ainsi il faudra prévoir une exportation du modèle CAO sous un standard STEP ou IFC.

# Bibliographie

- [AZDZ10] T. AWWAD, Q. ZHUAND, Z. DU et Y. ZHANG : An improved segmentation approach for planar surfaces from unstructured 3d point clouds. *The Photogrammetric Record*, 25:523–529, 2010.
- [BBH07] Jan BÖHM, Susanne BECKER et Norbert HAALA : Model refinement by integrated processing of laser scanning and photogrammetry. *Proceedings of the 3D Virtual Reconstruction and Visualization of Complex Architectures (3D-Arch)*, 2007.
- [BBTB11] A. BENNIS, V. BOMBARDIER, P. THIRIET et D. BRIE : Recalage d’un nuage de points de scanner laser terrestre avec une image de bâtiment. In *XXIIIe Colloque GRETSI Traitement du Signal ets des Images, Bordeaux, France.*, August 2011.
- [BF12] Shahar BARNEA et Sagi FILIN : Extraction of objects from terrestrial laser scans by integrating geometry image and intensity data with demonstration on trees. *Remote Sensing*, 4(1):88–110, 2012.
- [BGTk07] H. BOULAASSAL, P. GRUSSENMEYER et F. TARSHA-KURDI : Automatic segmentation of building facades using terrestrial laser data. In *Laser07*, page 65, 2007.
- [BH07] Susanne BECKER et Norbert HAALA : Refinement of building fassades by integrated processing of lidar and image data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Science*, 36:7–12, 2007.
- [BL06] D. BELTON et D.D LICHTI : Classification and segmentation of terrestrial laserscanner point clouds using local variance information. *International Archives of Photogrammetry and Remote Sensing*, XXXVI, 5, 2006.
- [Bla04] François BLAIS : Review of 20 years of range sensor development. *Journal of Electronic Imaging*, 13(1), 2004.
- [BM92] Paul J BESL et Neil D MCKAY : Method for registration of 3-d shapes. In *Robotics-DL tentative*, pages 586–606. International Society for Optics and Photonics, 1992.
- [BMB03] David BARBER, Jon MILLS et Paul BRYAN : Towards a standard specification for terrestrial laser scanning of cultural heritage. *CIPA International Archives for Documentation of Cultural Heritage*, 19:619–624, 2003.

- [Bou06] Mounzer BOUBOU : *Contribution aux méthodes de classification on supervisée via des approches prétopologiques*. Thèse de doctorat, Univertié Claude Bernard - Lyon 1, 2006.
- [Bou10] H. BOULAASSAL : *Segmentation et modélisation géométrique de façades de bâtiments à partir de relevés laser terrestres*. Thèse de doctorat, INSA Strasbourg, 2010.
- [BPPU02] M BALZANI, A PELLEGRINELLI, N PERFETTI et F UCCELI : A terrestrial 3d laser scanner-accuracy tests. *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*, 34(5/C7):445–453, 2002.
- [Bre05] C. BRENNER : Building reconstruction from images and laser scanning. *International Journal of Applied Earth Observation and Geoinformation*, 6:187–198, 2005.
- [Bry] P. BRYAN : An addendum to the metric survey specifications for english heritage - the collection and archiving of point cloud data obtained by terrestrial laser scanning or other methods. Rapport technique, [www.heritage3d.org](http://www.heritage3d.org).
- [Can86] F. John CANNY : A Computational Approach to Edge Detection. *j-IEEE-PAMI*, 8(6):679,698, 1986.
- [CH99] C. CHEN et Y. HUNG : Ransac-based darces : A new aooroach to fast automatic registratioi partially overlapping range images. *IEEE, Transactions on pattern analysis and machine intellignece*, 21(11):1299–1234, Nov 1999.
- [CM08] O. CHUM et J. MATAS : Optimal randomized ransac. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 30:1–11, 2008.
- [CR04] J CLARK et S ROBSON : Accuracy of measurements made with a cyrax 2500 laser scanner against surfaces of known colour. *Survey Review*, 37(294):626–638, 2004.
- [CWY<sup>+</sup>12] J CAO, S WUSHOUR, X YAO, N LI, J LIANG et X LIANG : Sharp feature extraction in point clouds. *IET Image Processing*, 6(7):863–869, 2012.
- [Del34] Boris DELAUNAY : Sur la sphere vide. *Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk*, 7(793-800):1–2, 1934.
- [Dev06] Matthieu DEVEAU : *Utilisation conjointe de données image et laser pour la segmentation et la modélisation 3D*. Thèse de doctorat, 2006. Thèse de doctorat dirigée par Pierrot-Deseilligny, Marc et Stamon, Georges Informatique Paris 5 2006.
- [EKS83] H. EDELSBRUNNER, D. KIRKPATRICK et R. SEIDEL : On the shape of a set of points in the plane. *Information Theory, IEEE Transactions on*, 29(4):551–559, Jul 1983.
- [EM94] Herbert EDELSBRUNNER et Ernst P MÜCKE : Three-dimensional alpha shapes. *ACM Transactions on Graphics (TOG)*, 13(1):43–72, 1994.
- [Fal13] Marie FALCON : *Proposition d'un processus et d'outils pour industrialiser la rénovation énergétique des bâtiments*. Thèse de doctorat, EDSYS, Insitut national polytechnique de Toulouse, France., 2013.

- 
- [FB81] M.A. FISCHLER et R.C BOLLES : Random sample consensus : A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24:381–395, 1981.
- [FJZ05] Christian FRUEH, Siddharth JAIN et Avidesh ZAKHOR : Data processing algorithms for generating textured 3d building facade meshes from laser scans and camera images. *International Journal of Computer Vision*, 61(2): 159–184, 2005.
- [FK04] Eric K FORKUO et Bruce KING : Automatic fusion of photogrammetric imagery and laser scanner point clouds. *International Archives of Photogrammetry and Remote Sensing*, 35:921–926, 2004.
- [FM04] C FRÖHLICH et M METTENLEITER : Terrestrial laser scanning?new perspectives in 3d surveying. *International archives of photogrammetry, remote sensing and spatial information sciences*, 36(Part 8):W2, 2004.
- [GA05] Armin GRUEN et Devrim AKCA : Least squares 3d surface and curve matching. *{ISPRS} Journal of Photogrammetry and Remote Sensing*, 59(3): 151 – 174, 2005.
- [GL04] SJ GORDON et DD LICHTI : Terrestrial laser scanners with a narrow field of view : the effect on 3d resection solutions. *Survey Review*, 37(292):448–468, 2004.
- [Gou] Portail du gouvernement, 2013. dossier de presse - plan d?investissement pour le logement - les 20 mesures. Dernière accée : 21-03-2013.
- [HBK06] N. HAALA, S. BECKER et M. KADA : Cell decomposition for the generation of building models at multiple scales. *Photogrammetric Computer Vision*, 2006.
- [Hou62] Paul VC HOUGH : Method and means for recognizing complex patterns, dec 1962. US Patent 3,069,654.
- [HP07] Bernhard HÖFLE et Norbert PFEIFER : Correction of laser scanning intensity data : Data and model-driven approaches. *ISPRS Journal of Photogrammetry and Remote Sensing*, Volume 62(Issue 6):415–433, 2007.
- [HS88] C. HARRIS et M. STEPHENS : A combined corner and edge detector. *In Alvey Vision Conference*, pages 147,151, 1988.
- [Jac05] Geoff JACOBS : Registration and geo-referencing. *Professional Surveyor Magazine*, pages 30–37, 2005.
- [Jam95] *Parametric and featur-based CAD/CAM*. Jhon Wiley and sons, INC., 1995.
- [Jel92] A. JELALIAN : *Laser radar systems*. Artech Print, 1992.
- [Kow90] P. KOWALISKI : *Vision et mesure de la couleur*. Physique fondamentale et appliqu?e. Masson, 2 édition, 1990. actualis?e par F. Vi?not et R. S?ve.
- [LBG12] Tania LANDES, Hakim BOULAASSAL et Pierre GRUSSENMEYER : Quality assessment of geometric facade models reconstructed from tls data. *The Photogrammetric Record*, 27(138):137–154, 2012.



- [Lic07] Derek D LICHTI : Error modelling, calibration and analysis of an am-cw terrestrial laser scanner system. *ISPRS Journal of Photogrammetry and Remote Sensing*, 61(5):307–324, 2007.
- [LJ06] Derek D LICHTI et Sonam JAMTSHO : Angular resolution of terrestrial laser scanners. *Photogrammetric record*, 21(114):141–160, 2006.
- [LSTS00] D LICHTI, MP STEWART, M TSAKIRI et AJ SNOW : Calibration and testing of a terrestrial laser scanner. *International archives of Photogrammetry and Remote sensing*, 33(B5/2 ; PART 5):485–492, 2000.
- [MGPG04] Niloy J MITRA, Natasha GELFAND, Helmut POTTMANN et Leonidas GUIBAS : Registration of point cloud data from a geometric optimization perspective. *In Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 22–31. ACM, 2004.
- [MKL07] Klaus MECHELKE, Thomas P KERSTEN et Maren LINDSTAEDT : Comparative investigations into the accuracy behaviour of the new generation of terrestrial laser scanning systems. *Proc. in the Optical*, pages 319–327, 2007.
- [MNG04] Niloy J MITRA, An NGUYEN et Leonidas GUIBAS : Estimating surface normals in noisy point cloud data. *International Journal of Computational Geometry & Applications*, 14(04n05):261–276, 2004.
- [MTN<sup>+</sup>02] D. R. MYATT, P. H. S. TORR, S. J. NASUTO, J. M. BISHOP et R. CRADDOCK : Napsac : high noise, high dimensional robust estimation. *In In BMVC02*, pages 458–467, 2002.
- [PB07a] N. PFEIFER et C. BREISE : Geometrical aspects of airborne laser scanning and terrestrial laser scanning. *IAPRS*, 36:311,319, 2007.
- [PB07b] Norbert PFEIFER et Christian BRIESE : Laser scanning-principles and applications. *In 3rd International Exhibition and Scientific Congress on Geodesy, Mapping, Geology, Geophysics, Cadaster GEO-SIBERIA.*, 2007.
- [PKG03] Mark PAULY, Richard KEISER et Markus GROSS : Multi-scale feature extraction on point-sampled surfaces. *In Computer graphics forum*, volume 22, pages 281–289. Wiley Online Library, 2003.
- [PPPD05] Lionel PÉNARD, Nicolas PAPARODITIS et Marc PIERROT-DESEILLIGNY : 3d building facade reconstruction under mesh form from multiple wide angle views. *Proceedings of the ISPRS Working Group*, 4, 2005.
- [PTB11] Arianna PESCI, Giordano TEZA et Elena BONALI : Terrestrial laser scanner resolution : Numerical simulations and experiments on spatial sampling optimization. *Remote Sensing*, 3(1):167–184, 2011.
- [PV06] S. PU et G. VOSSELMAN : Automatic extraction of building features from terrestrial laser scanning. *International Archives of Photogrammetry and Remote Sensing*, 36:295–301, 2006.
- [PV09] S. PU et G. VOSSELMAN : Building facade reconstruction by fusing terrestrial laser points and images. *ISPRS Journal of Photogrammetry and Remote sensing*, 64:2525–4542, 2009.

- 
- [RDvdHV07] Tahir RABBANI, Sander DIJKMAN, Frank van den HEUVEL et George VOSSELMAN : An integrated approach for modelling and global registration of point clouds. *{ISPRS} Journal of Photogrammetry and Remote Sensing*, 61(6):355–370, 2007.
- [Res09] Yuriy RESHETYUK : *Self-calibration and direct georeferencing in terrestrial laser scanning*. Thèse de doctorat, Royal Institute of Technology (KTH), Department of Transport and Economics, Division of Geodesy, 2009.
- [SA02] I. STAMOS et P.K. ALLEN : Geometry and texture recovery of scenes of large scale. *Computer vision and image understanding*, 88:94–118, May 2002.
- [SB05] Sara SCHUHMACHER et Jan BÖHM : Georeferencing of terrestrial laser-scanner data for applications in architectural modeling. 2005.
- [Sch04] Hanns-F SCHUSTER : Segmentation of lidar data using the tensor voting framework. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 35(B3):1073–1078, 2004.
- [Sch07a] Emmanuel SCHMITT : *Contribution au Système d’Information d’un Produit «Bois»*. *Appariement automatique de pièces de bois selon des critères de couleur et de texture*. Thèse de doctorat, Université Henri Poincaré-Nancy I, 2007.
- [Sch07b] Thorsten SCHULZ : *Calibration of a Terrestrial Laser Scanner for Engineering Geodesy*. Thèse de doctorat, École polytechnique fédérale de Zurich, 2007.
- [SHA06] Majd AL SHAWA : Consolidation des nuages de points en lasergrammetrie terrestre. Mémoire de D.E.A., École Nationale Supérieure d’Architecture de Nancy Université Henri Poincaré, 2006.
- [Shi62] Moshe SHIMRAT : Algorithm 112 : position of point relative to polygon. *Communications of the ACM*, 5(8):434, 1962.
- [SMFF07] Joaquim SALVI, Carles MATABOSCH, David FOFI et Josep FOREST : A review of recent range image registration methods with accuracy evaluation. *Image and Vision Computing*, 25(5):578 – 596, 2007.
- [SRSD01] A SAPPA, Andres RESTREPO-SPECHT et Michel DEVY : Range image registration by using an edge-based representation. *In Proceedings of the Ninth International Symposium on Intelligent Robotic Systems (SIRS 01), Toulouse, France*, 2001.
- [SWK07] R. SCHNABEL, R. WAHL et R. KLEIN : Efficient ransac for point-cloud shape detection. *In Computer Graphics Forum*, volume 26, pages 214–226. Wiley Online Library, 2007.
- [TTMM04] Wai-Shun TONG, Chi-Keung TANG, Philippos MORDOHAI et Gérard MEDIONI : First order augmentation to tensor voting for boundary inference and multiscale analysis in 3d. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(5):594–611, 2004.
- [WL99] Aloysius WEHR et Uwe LOHR : Airborne laser scanning?an introduction and overview. *ISPRS Journal of Photogrammetry and Remote Sensing*, 54(2):68–82, 1999.

- [Woo08] WOODWISDOM : Tes energy façade - prefabricated timber based building system for improving the energy efficiency of the buildgin envelope. Rapport technique, 2008. Dernière accée : 09-06-2014.
- [XW+05] Rui XU, Donald WUNSCH *et al.* : Survey of clustering algorithms. *Neural Networks, IEEE Transactions on*, 16(3):645–678, 2005.