



Modélisation 3D de la végétation sur le territoire de Rennes Métropole

Coralie Leblan

► To cite this version:

Coralie Leblan. Modélisation 3D de la végétation sur le territoire de Rennes Métropole. Sciences de l'ingénieur [physics]. 2018. dumas-02093042

HAL Id: dumas-02093042

<https://dumas.ccsd.cnrs.fr/dumas-02093042>

Submitted on 8 Apr 2019

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.

CONSERVATOIRE NATIONAL DES ARTS ET METIERS
ECOLE SUPERIEURE DES GEOMETRES ET TOPOGRAPHES

MEMOIRE

présenté en vue d'obtenir

le DIPLOME D'INGENIEUR CNAM

SPECIALITE : Géomètre et Topographe

par

Coralie LEBLAN

Modélisation 3D de la végétation sur le territoire de Rennes Métropole

Soutenu le 04 Juillet 2018

JURY

PRESIDENT : M. Christophe PROUDHOM

**MEMBRES : Mme. Marie FOURNIER
Mr. Vincent HABCHI
Mme. Christelle GIBON**

**Examineur
Professeur référent
Maître de stage**

Remerciements

Le présent travail marque l'ultime chapitre ma formation d'ingénieur géomètre et topographe à l'ESGT.

Mes premiers remerciements sont avant tout dédiés à ma famille et plus particulièrement à ma mère qui m'aura toujours accompagné dans mes projets, même les plus fous, et sans le soutien de qui rien n'aurait été possible.

Je remercie ensuite Christelle GIBON pour m'avoir accordé sa confiance et proposé un sujet de TFE qui aura su me passionner pendant ces 6 mois. Je la remercie pour son naturel, sa bonne humeur, sa disponibilité et ses nombreux conseils avisés. Je n'oublie pas non plus ses nombreuses relectures de ce mémoire.

Parce qu'ils m'ont offert un accueil chaleureux et qu'ils auront tous contribué, au détour de simples conversations, de conseils, de taquineries ou d'encouragements, à la création d'une atmosphère plus que sympathique, je remercie l'ensemble des agents du service SIG de Rennes Métropole.

Toujours prêt à aider malgré le décalage horaire, je remercie mon ami Loïc MESSAL (bibliothécaire) pour ses nombreux débogages et son aide précieuse.

Je remercie également Jean NABUCET pour m'avoir orienté et conseillé à des moments clés de réalisation de ce TFE.

Enfin je remercie Vincent HABCHI, mon professeur référent, pour son apport pédagogique, ses remarques pertinentes et corrections apportées à ce mémoire.

Liste des abréviations

- TFE : Travail de Fin d'Etudes
- MNU : Maquette Numérique Urbaine
- LIDAR : Light Detection And Ranging
- LASER : Light Amplification by Simulation of Emission Radiation
- TLS : Terrestrial LASER Scanning
- ALS : Airbone LASER Scanning
- LOD : Level Of Details
- ITCD : Individual Tree Crown Delineation
- MHC : Modèle de Hauteur de Canopée
- MNT : Modèle Numérique de Terrain
- MNS : Modèle Numérique de Surface
- DFT : Digital Forestry Toolbox
- ASPRS : American Society for Photogrammetry and Remote Sensing
- SOR : Statistical Outlier Removal
- PCL : Point Cloud Library

Glossaire

- MAPTAM : La loi de Modernisation de l'Action Publique Territoriale et d’Affirmation des Métropoles est promulguée le 27 Janvier 2014 puis publiée au journal officiel le 28 Janvier 2014. Elle vise à clarifier les compétences de collectivités territoriales en créant des organes de concertation entre les collectivités et en réorganisant le régime juridique des métropoles.
- NOTRe : La loi relative à la Nouvelle Organisation Territoriale de la République est promulguée le 7 Août 2015 puis publiée le 8 Août 2015 au journal officiel. Elle vise à confier de nouvelles compétences aux régions.
- 3DM : Three Dimensional Modeling. Format de stockage de donnée natif du logiciel Rhinocéros
- FME : (Feature Manipulation Engine), logiciel développé par SAFE Software permettant la transformation, l'extraction et le chargement de données géographiques.
- Couronne (ou houppier) : Partie de l'arbre allant de la branche la plus basse à la cime.

Table des matières

Remerciements	2
Liste des abréviations	3
Glossaire	4
Table des matières	5
Introduction	8
I LE SERVICE SIG DE RENNES ET LE PROJET 3DEXPERIENCE CITY	10
I.1 STRUCTURE D'ACCUEIL.....	10
I.1.1 Rennes Métropole	10
I.1.1.1 Présentation	10
I.1.1.2 Compétences métropolitaines.....	10
I.1.1.3 Un label « Ville intelligente et mobilité »	11
I.1.2 Le service de l'Information Géographique de Rennes, Ville et Métropole	11
I.1.2.1 Un service mutualisé	11
I.1.2.2 Indicateurs d'activités.....	11
I.1.2.3 Le pôle Cartographie et Diffusion 2D/3D	12
I.2 LES MAQUETTES NUMERIQUES DE RENNES METROPOLE.....	12
I.2.1 Les usages de la maquette 3D : Exemple Rennais.....	12
I.2.1.1 La communication territoriale	13
I.2.1.2 La médiation urbaine.....	13
I.2.1.3 La médiation professionnelle	13
I.2.1.4 L'Aide à la décision	13
I.2.1.5 3DEXPERIENCE City Virtual Rennes : Vers une plateforme collaborative.....	14
I.2.2 La structure de la maquette 3D	14
I.2.2.1 Une maquette construite avec Rhinocéros 3D.....	14
I.2.2.2 Une maquette de LoD2.....	14
I.2.2.3 3D EXPERIENCECity Virtual Rennes : Le City Information Modeling	15
II DES DONNEES 3D ET DES DONNEES LiDAR.....	16
II.1 LA BASE DE DONNEES 3D DU SERVICE SIG	16
II.1.1 Les bâtiments dans la maquettes 3D	16
II.1.1.1 Le bâti sur le territoire de la ville de Rennes	16
II.1.1.2 Le bâti sur le territoire des autres communes de la métropole	16
II.1.2 La végétation dans la maquette 3D	16
II.2 UN NUAGE DE POINTS LiDAR	17
II.2.1 Le LiDAR, donnée privilégiée pour l'inventaire forestier.....	17
II.2.1.1 Définition	17
II.2.1.2 Composition d'un système ALS et positionnement des retours	17
II.2.1.3 Plusieurs types de LiDAR	18
II.2.1.4 Décomposition des retours	19
II.2.2 Le nuage de points LiDAR	19
II.2.2.1 Une acquisition effectuée par l'IGN	19
II.2.2.2 Un nuage de points pré-classifié par l'IGN.....	19
III UN ALGORITHME AUTOMATIQUE DE DETECTION ET DE SEGMENTATION DE LA VEGETATION	21
III.1 UNE APPROCHE TRADITIONNELLE ET ROBUSTE	21
III.2 UN ALGORITHME EN SIX PHASES	21
III.2.1 Prétraitements : Nettoyage du nuage LiDAR.....	23
III.2.2 Calcul d'un modèle numérique de hauteur de canopée (MHC)	23
III.2.3 Détection et segmentation de la végétation	24

III.2.4	Partition des arbres en zones dites « ouvertes » et « fermées »	25
III.2.5	Tentative d'amélioration de la segmentation	25
III.2.6	Un manque de gestion des systèmes de coordonnées	26
IV	RECONSTRUCTION 3D DE LA VEGETATION : ETAT DE L'ART	28
IV.1	TECHNIQUES DE RECONSTRUCTION D'ARBRES EN 3 DIMENSIONS	28
IV.1.1	Reconstruction 3D architecturale d'arbres individuels d'après des données terrestres ..	28
IV.1.2	De la segmentation des arbres au clustering du nuage de points LiDAR aérien	29
IV.1.3	Reconstruction 3D d'arbres individuels d'après des données LiDAR aériennes	30
IV.1.3.1	Diagramme de Voronoï et triangulation de Delaunay	30
IV.1.3.2	Estimateur convexe	32
IV.1.3.3	Estimateur α -convexe	33
IV.1.3.4	Reconstruction par formes « <i>alpha</i> »	35
V	MODELISATION 3D DE LA VEGETATION DE RENNES METROPOLE	37
V.1	PREPARATION DE LA CLUSTERISATION DU NUAGE DE POINTS LiDAR : ADAPTATION DE L'ALGORITHME DE DETECTION ET DE SEGMENTATION	37
V.1.1	Pré-traitements : Extraction de la végétation	37
V.1.1.1	Suppression des résidus de bâtiments	37
V.1.1.2	Nettoyage par test statistique : Statistical Outlier Removal	38
V.1.1.3	Nettoyage par analyse de densité	39
V.1.2	Détection et segmentation des arbres	40
V.1.2.1	Calcul d'un modèle numérique de canopée	40
V.1.2.2	Détection et segmentation	41
V.1.3	Préparation des données	42
V.1.3.1	Une distinction de zones nécessaires à la reconstruction	42
V.1.3.2	Une nouvelle approche de partition	42
V.2	RECONSTRUCTION <i>BILLBOARDS</i> DES ARBRES	44
V.2.1	Extraction des paramètres dendrométriques	44
V.2.1.1	Hauteur des arbres	45
V.2.1.2	Diamètres des couronnes	45
V.2.2	Positionnement des arbres	45
V.2.3	Résultats	46
V.3	RECONSTRUCTION 3D MAILLEE DES ARBRES	48
V.3.1	Un processus développé en collaboration avec <i>RhinoTerrain</i>	48
V.3.2	Une reconstruction des houppiers de type « Alpha Shape »	49
V.3.2.1	Préparation des données : création de clusters	49
V.3.2.2	De la 2.5D vers la 3D	49
V.3.2.3	Reconstruction de la base des houppiers	49
V.3.2.4	Une implémentation sous R, langage interopérable avec FME	50
V.3.2.5	Reconstruction 3D : la fonction « <i>ashape3d</i> »	51
V.3.2.6	Export des houppiers : Structure du format .OBJ	52
V.3.2.7	Résultats	52
V.3.3	Reconstruction des troncs par extrusion	53
V.3.3.1	Approximation du diamètre et de la hauteur des troncs	54
V.3.4	Résultat	54
VI	DISCUSSIONS	56
VI.1	EFFICACITE DU PROTOCOLE DE PRETRAITEMENTS	56
VI.2	LIMITES DE SEGMENTATION ET DE PARTITION DES ARBRES	57
VI.3	VERS UN CLUSTERING "ORIENTE NUAGE" : K-MOYENNES ET VOXELS	58
VI.4	RECONSTRUCTION 3D DES HOUPPIERS ET ECARTS AU NUAGE DE POINTS	58
VI.5	RECONSTRUCTION 3D DES HOUPPIERS : FACES NON SOUDEES ET NON ORIENTEES	59
VI.6	LA CORRELATION DENSE, UNE ALTERNATIVE ENVISAGEABLE ?	59

Conclusion.....	60
Bibliographie	61
Annexe 1 Exécuter un script R dans FME	64
Annexe 2 Implémentation de la fonction « ashape3d » de R	66
Annexe 3 Développement d’une interface graphique	67
Liste des figures.....	68
Liste des tableaux	70

Introduction

Nouvelle métropole depuis le 1^{er} Janvier 2015 (loi MAPTAM), Rennes détient également le label « Ville intelligente et mobilité » en vertu du pacte Etat-Métropole au sein duquel elle est engagée depuis le 27 Janvier 2017 *#PacteEtatMetropole* [15]. Poursuivant l'objectif de développer une « smart city à la rennais », pour reprendre le terme employé par Hélène Bailleul¹ dans le cadre de la présentation du projet « SMARTRENNES² », la métropole ne cesse de développer des initiatives pour penser la ville de demain : plus fonctionnelle, économe et durable. Citons, par exemple, l'évènement *SMART CITY Rennes, Ville Intelligente & Connectée* ayant eu lieu au parc des expositions les 25 et 26 Avril de cette année, ou encore le projet *3DEXPERIENCE City Virtual Rennes* visant à développer, en partenariat avec *Dassault System*, une plateforme collaborative de partage de données géographiques sur la base de la maquette 3D existante.

Afin de relever le défi d'un développement urbain à la fois durable et attractif, comme définit par Rudolf Giffinger³, les villes misent aujourd'hui sur le numérique. Si la maquette 3D constituait déjà un formidable support tant en matière de communication, que de médiation (urbaine comme professionnelle) que d'aide à la décision, aujourd'hui, la mode est à la transformation du service public lui-même. Ainsi, dans un objectif de fédération de l'ensemble des acteurs urbains autour d'un projet commun, la maquette 3D évolue et devient plateforme collaborative. C'est le *City Information Modeling* (CIM).

L'intégration des bâtiments dans la maquette 3D est aujourd'hui un processus largement maîtrisé. Il n'en va pas de même pour la végétation constituant pourtant la deuxième donnée volumétrique la plus importante à l'échelle de la ville. Dans un double objectif de représentation et d'intégration de cette dernière dans les simulations urbaines, Rennes Métropole s'intéresse aujourd'hui au développement d'un processus de reconstruction en 3D des arbres. Cette problématique a été amorcée en 2017 dans le cadre d'un précédent TFE réalisé par C. Doceul [4] et ayant mené à la mise en place d'un processus de détection et de segmentation de la végétation d'après un nuage de points LiDAR aérien.

Le présent travail de recherche s'inscrit dans la suite logique de ces travaux antérieurs et poursuit le processus jusqu'à permettre la création d'objets intégrables dans la maquette 3D de la métropole.

¹ Hélène Bailleul est maîtresse de conférences en Aménagement de l'espace et urbanisme à l'UMR Espaces et Sociétés (ESO) à l'université Rennes 2

² SMARTENNE – Acteur et conduite du projet de smart city sur le territoire de Rennes Métropole (Janvier 2018) est une réponse à un appel à projet du PUCA (Plan Urbanisme, Construction, Architecture) intitulé « La gouvernance de/par l'innovation ». Le projet s'intéresse à la mise en place des principes de la ville intelligente à l'échelle de la métropole.

³ Rudolf Giffinger, est expert en recherche analytique sur le développement urbain et régional à l'université technologique de Vienne

Cette étude présentera dans un premier temps l'organisme d'accueil : le service SIG de Rennes ville et métropole ainsi que sa maquette 3D. Dans un second temps, nous nous intéresserons aux différents types de données intervenant dans le processus puis détaillerons les travaux de C. Doceul [4] dans une troisième partie. Dans un quatrième temps, nous effectuerons un état de l'art des techniques de géométrie computationnelle propres à la reconstruction de formes à partir de sous-ensembles discrets avant de développer, dans un cinquième temps, notre processus de reconstruction 3D des arbres d'après des formes dites « *alpha* ». Finalement, nous apporterons un regard critique sur nos résultats dans une sixième et dernière partie.

I Le service SIG de Rennes et le projet 3DEXPERIENCE City

I.1 Structure d'accueil

I.1.1 Rennes Métropole

I.1.1.1 Présentation

Située dans l'ouest de la France, Rennes est la capitale de la région Bretagne et le Chef-Lieu du département d'Ille-et-Vilaine.

Récemment devenue métropole de droit commun depuis le 1^{er} Janvier 2015 en vertu de la loi MAPTAM du 27 Janvier 2014, elle regroupe 43 communes et compte 438 868 habitants⁴. Son président actuel, Emmanuel Couet, est également maire de la commune de Saint-Jacques-de-la-Lande.

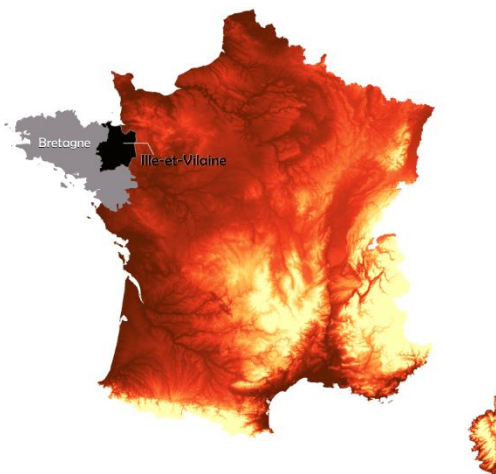


Figure 1 : Carte localisant le département de l'Ille-et-Vilaine

I.1.1.2 Compétences métropolitaines

L'acquisition du statut de métropole s'est accompagnée de l'extension du champ de compétences initial de la ville au développement économique, à l'enseignement supérieur et de recherche, au logement, à la mobilité durable et à l'énergie.



Figure 2: Logo de Rennes Métropole

La métropole exerce de plein droit, en lieu et place des communes membres, les compétences énumérées à l'article L.5212-2 du code général des collectivités territoriales soit en matière de :

- Développement et d'aménagement économique, social et culturel
- D'aménagement de l'espace métropolitain
- Politique de la ville
- Gestion des services d'intérêts collectif
- Protection et de mise en valeur de l'environnement et de politique du cadre de vie

En vertu des lois MAPTAM et NOTRe, elle s'est également vu attribuer certaines compétences départementales, dont la gestion des routes classées dans le domaine public

⁴ INSEE, 2015

routier départemental, ainsi que de leurs dépendances et accessoires, du fonds d'aide aux jeunes, de la prévention spécialisée et du fond de solidarité pour le logement.

I.1.1.3 Un label « Ville intelligente et mobilité »

Le pacte État-Métropole vise à définir une stratégie nationale de développement des métropoles fondé sur l'innovation *#EtatMetropole* [15].

Les « pactes métropolitains d'innovation » renforcent le contenu innovant des politiques conduites par les métropoles. Pour Rennes ce pacte, signé le 27 Janvier 2017, porte la mention “Ville intelligente et mobilité⁵” et doit permettre à la métropole rennaise de devenir un laboratoire d'expérimentation des nouvelles mobilités.

Ensemble les pactes “Etat-Métropole” et “Métropolitain d'innovation”, positionnent Rennes parmi les métropoles françaises motrices en matière d'innovation urbaine et en font une destination à “haut-potentiel” pour les investissements internationaux.

I.1.2 Le service de l'Information Géographique de Rennes, Ville et Métropole

I.1.2.1 Un service mutualisé

Mutualisé, le service SIG répond aux sollicitations de deux donneurs d'ordres principaux : Rennes Métropole et la Ville de Rennes pour lesquels il assure diverses missions d'acquisition, d'exploitation, de gestion ou encore de diffusion de données géographiques. Le service comprend quatre unités distinctes :

Unité	Fonctions
Administrative	Ressources humaines, Marchés, Conventions, Finances
Données topographiques	Acquisition et gestion de données topographiques. Travaux spécifiques requérant expertise et précision
Études et Développement	Architecture des bases de données SIG Mise en place et maintenance d'outils logiciels de gestion et d'exploitation de données
Cartographie et Diffusion 2D/3D	Gestion, diffusion et exploitation des bases de données

Tableau 1: Unités du service SIG de Rennes Métropole

I.1.2.2 Indicateurs d'activités

Selon des données fournies par le service, l'unité “Données Topographiques” assure une moyenne annuelle de près de 130 dossiers de relevés de terrain. Elle est également responsable du contrôle et de la gestion des quelques 11 000 stations de références

⁵ Le plan national propose trois thématiques possibles : “Transition énergétique et environnement”, “Villes intelligentes et mobilités”, “Rayonnement international”

géodésiques et 1 200 repères de nivellement répartis sur le territoire de la métropole. De son côté, l'unité "Etude et Développement" joue un rôle important auprès des agents du service public en assurant l'installation et la maintenance de logiciels de SIG sur près de 350 postes de travail. Elle est également à l'origine du développement d'une trentaine d'applications à la fois grand public (*GéoRennesMétropole*) et métier (*GéoJardins* pour la gestion des espaces verts, *Urbavue* pour la consultation des PLU et droit des sols, ...), applications pour lesquelles elle constate une utilisation croissante. Enfin, l'unité « Cartographie et Diffusion 2D/3D », chargée de la mise à jour régulière des données 2D et 3D, de leur exploitation (création de la maquette 3D entre autres) ainsi que de leur gestion et diffusion, répond aux sollicitations croissantes des services de la ville, de ses partenaires et du grand public, notamment en alimentant les données placées en *Open Data*.

I.1.2.3 Le pôle Cartographie et Diffusion 2D/3D

Le pôle "Cartographie et Diffusion 2D/3D" est placé sous la direction du responsable Frédéric CHAUVIN et comprend 10 agents répartis selon l'organigramme suivant :

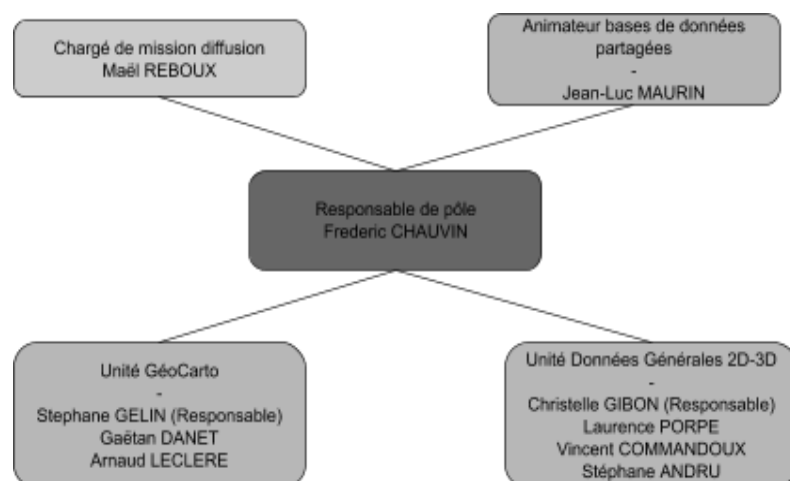


Figure 3: Organigramme de l'unité Cartographie et Diffusion 2D/3D

Le pôle « Données générales 2D/3D » est autonome et chargé de l'acquisition et de la gestion de la base de données générale (base référentielle à l'échelle 1:2 000) qui sert à la production et à la gestion de la maquette 3D. Cette maquette participe à la mise en place de dispositifs d'aide à la décision, de concertation et de communication à destination des élus et du grand public.

I.2 Les maquettes numériques de Rennes Métropole

I.2.1 Les usages de la maquette 3D : Exemple Rennais

Une Maquette Numérique Urbaine (MNU), ou maquette 3D, est communément assimilée à une représentation en 3 dimensions d'un territoire ou d'une ville à partir de restitutions de prises de vues aériennes et de modélisations à partir de données disponibles Valente [21].

I.2.1.1 La communication territoriale

En 1999, pour fêter le passage à l'an 2000, les élus rennais proposent aux habitants une reproduction numérique de la ville en 3D. C'est la première maquette 3D de Rennes. Face à l'engouement suscité par le grand public, deux autres maquettes historiques proposent ensuite un voyage dans le temps jusque dans les années 1740 et 1860 (*figure 4*). Une fois encore, le succès est au rendez-vous et la maquette 3D se révèle un véritable outil ascendant permettant l'appréhension du territoire dans son ensemble et dans son histoire.



Figure 4: Maquette 3D et information territoriale

Aujourd'hui, les maquettes en ligne « Rennes 2030 » et « De Condate⁶ à Rennes » présentent respectivement les grands projets urbains prévus sur le territoire d'ici à 2030 et des représentations historiques de la ville du XVII^e siècle à aujourd'hui.

I.2.1.2 La médiation urbaine

Visible, évolutive et interactive, la maquette 3D permet de vulgariser l'information. Or, une meilleure compréhension entraîne une participation plus active du grand public. La maquette 3D s'impose comme support de concertation efficace, déclencheur de parole et de débat. Par une approche plus ludique, le service SIG a également su sensibiliser la jeune population rennaise aux règles d'urbanisme générale (Chasse au trésor 3D *Festimômes* et *RennesCraft*).

I.2.1.3 La médiation professionnelle

Du point de vue du développement économique et de l'attractivité du territoire, la maquette 3D permet de promouvoir les projets de développement urbain à un niveau professionnel. Elle est notamment utilisée dans le cadre de congrès, salons ou encore délégations comme outil de médiation professionnelle.

I.2.1.4 L'Aide à la décision

Enfin, la maquette 3D est un outil puissant d'aide à la décision permettant la modélisation et l'intégration de plusieurs variantes d'un projet dans son milieu d'accueil. La figure ci-dessous, à titre d'exemple, illustre la modélisation du « bruit » (sonore) pour des bâtiments identifiés comme pouvant être impactés par un projet d'aménagement.

⁶ « Condate » est l'ancien nom de la ville de Rennes



Figure 5 : Modélisation du bruit 10 secteurs à enjeux "bruit" sur Rennes

I.2.1.5 3DEXPERIENCE City Virtual Rennes : Vers une plateforme collaborative

Nous l'avons vu, la maquette 3D constitue un support puissant d'analyse à l'échelle de la ville. Cependant, ces dernières sont réalisées de façon déconnectée et ne tiennent pas compte de l'influence d'un facteur urbain sur un autre (à l'image de l'influence des modes de déplacements sur le taux de pollution de l'air par exemple). Pour réaliser des analyses transversales, il est nécessaire de fédérer tous les acteurs urbains autour d'un socle de données commun. Dans ce but, la maquette 3D de Rennes tend depuis Juin 2015 à se transformer en une plateforme collaborative dans le cadre du projet *3D EXPERIENCE City Virtual Rennes*, développé en partenariat avec la société *Dassault System*.

I.2.2 La structure de la maquette 3D

I.2.2.1 Une maquette construite avec Rhinocéros 3D

Rhinoceros est un logiciel de CAO développé par *Robert McNeel&Associates* permettant de créer, éditer, analyser, documenter, rendre, animer et traduire des courbes et des surface NURBS (*Non-Uniform Rational B-Spline*, ou représentation mathématique pouvant décrire précisément toute forme en deux ou trois dimensions) ainsi que des solides, des nuages de points et des maillages. Le format 3DM (*Three Dimensional Modeling*) en est le format natif.



Rhinoceros®

Figure 6: Logo du logiciel Rhinoceros

Ce logiciel a servi à la réalisation de la maquette 3D de Rennes Métropole, qui utilise notamment les extensions *RhinoTerrain* et *RhinoCity*, développées par la société *RhinoTerrain*. Ces extensions permettent respectivement de créer et d'afficher la maquette 3D à partir d'importants volumes de données d'une part et, d'autre part, de créer, de texturer et d'exporter ses composantes au format CityGML.

I.2.2.2 Une maquette de LoD2

Le standard CityGML est "un format en Open Data ayant pour enjeu principal d'assurer l'interopérabilité entre les différents logiciels et acteurs modélisant et exploitant une maquette numérique à l'échelle urbaine." *Valente* [20].

Standard officiel international d'échange développé par l'OGC, le CityGML répond à la norme ISO TC211 et s'inspire du langage GML, lui-même dérivé du langage XML créé pour encoder, manipuler et échanger de la donnée géographique. Le CityGML présente deux caractéristiques principales : La modélisation thématique d'une part, au travers d'une large gamme de composants urbains (bâtiments, réseaux de transports, hydrographie, végétation, relief, occupation du sol, ...) et la gestion multi-échelle d'autre part, de la région au bâtiment en s'appuyant sur le concept des LoD dont il spécifie cinq niveaux (*figure 7*)

- **LoD 0 (Le sol)** : MNT drapé par une orthoimage ou une carte ;
- **LoD1 (La ville)** : Modélisation des bâtiments par des blocs à toits plats souvent issus d'une extrusion ;
- **LoD2 (Le quartier)** : Modélisation des toits des bâtiments et apparition du mobilier urbain et de la végétation ;
- **LoD 3 (Le modèle architectural extérieur)** : Détail de la structure des façades et des toits (balcons, fenêtres) et texturation haute résolution de la végétation et du mobilier urbain ;
- **LoD 4 (Le modèle architectural intérieur)** :
Représentation de l'architecture intérieure des bâtiments.
Rencontre entre le MNU et le MNB



Figure 7: LoD en CityGML

Aujourd'hui, la maquette 3D de Rennes Métropole respecte un LoD 2.

I.2.2.3 3D EXPERIENCE City Virtual Rennes : Le City Information Modeling

La conception, la simulation et l'implémentation d'analyses dans un environnement urbain nécessitent de disposer d'une importante quantité de données pouvant aussi bien concerner les caractéristiques géométriques des bâtiments, que leur norme de constructions ou leurs connexions aux réseaux d'eau et d'énergie. Par exemple, la génération d'une carte de potentiel solaire requiert de connaître la géométrie et leur orientation des toitures pour déterminer individuellement leur potentiel photovoltaïque et génération d'énergie thermique Wolisz *et al* [24].

Si les plateformes BIM constituent des solutions adaptées à la réalisation de telles simulations dans le cas de bâtiments isolés, elles ne sont pas conçues pour les généraliser à l'échelle de la ville. C'est pourquoi le *City Information Modeling*, fusion du BIM et de données géographiques au sein d'un système de coordonnées unique, a été créé. Ce partage de données entre les divers acteurs urbains et leur intégration au sein d'une plate-forme collaborative constitue l'objectif du projet *3DEXPERIENCE City Virtual Rennes*.

II Des données 3D et des données LiDAR

Nous disposons pour notre étude d'un nuage de points LiDAR aérien acquis par l'IGN durant l'été 2017 ainsi que de couches de données fournies par le service SIG de Rennes Métropole.

II.1 La base de données 3D du service SIG

II.1.1 Les bâtiments dans la maquettes 3D

Pour la reconstruction 3D du bâti, le service SIG distingue les bâtiments de la ville de Rennes de ceux des autres communes de la métropole.

II.1.1.1 Le bâti sur le territoire de la ville de Rennes

Pour la ville de Rennes, avant 2005, le bâti était défini par des lignes de toiture (faîtage et arêtes) et des emprises au sol. Depuis, une mise à jour est faite considérant principalement les plans joints aux permis de construire et, parfois, des orthophotographies ou levés topographiques. L'objectif est de renseigner quatre types de primitives géométriques à l'aide du logiciel *TopStation* :

Désignation	Primitive à renseigner
Emprise au sol	Polygone
Lignes de découpe de toiture	Point
Ligne d'arêtes	Point
Ligne de Faîtage	Polygone

Tableau 2: Primitives géométriques des bâtiments sur le territoire de la ville de Rennes

Sur la base de ces primitives, le module *RhinoCity* génère des objets 3D au format CityGML puis opère une texturation automatique à partir de photographies aériennes renouvelées tous les trois ans par l'IGN.

II.1.1.2 Le bâti sur le territoire des autres communes de la métropole

Sur les autres communes de la métropole, depuis 2011 le service SIG fait appel à un prestataire en charge de la mise à jour (tous les 3 ans également) des primitives propres aux lignes de toitures de la totalité du bâti, par photogrammétrie et ajustements manuel. Une fois les toitures définies, l'emprise au sol correspondante est obtenue par projection. Considérant ces primitives, les bâtiments sont ensuite reconstruits et texturés par le module *RhinoTerrain*.

II.1.2 La végétation dans la maquette 3D

La modélisation de la végétation dans la maquette 3D repose actuellement sur un processus interne combinant une chaîne de traitement FME avec le logiciel *Rhinocéros*. Considérant une couche de points au format .shp renseignant la position des arbres (centre de leur tronc) ainsi que leurs hauteurs et envergures (diamètre de couronne), le module

reconstruit les arbres selon des *billboards* texturés aléatoirement à partir d'une bibliothèque d'images recensant une vingtaine de modèles.

Chaque arbre est alors représenté par deux faces perpendiculaires qui, vue du dessus forment une croix ; d'où la dénomination de modélisation « en croisillon » (*figure 8*)



Figure 8: Représentation de type "billboard" texturé des arbres

Les travaux de détection et de segmentation de la végétation réalisés par C. Doceul conduisent aujourd'hui à la génération de deux couches shapefiles ; l'une propre aux cimes détectées (points) et l'autre aux couronnes segmentées correspondantes (polygones). Nous verrons par la suite que de ces couches sont facilement extraits les paramètres nécessaires à une modélisation « en croisillon » des arbres.

Bien que beaucoup moins fidèle à la réalité qu'une reconstruction géométrique maillée (ou en volume), la modélisation en « croisillon » n'en reste pas moins rapide à produire, et peut s'avérer utile dans des cas spécifiques de représentations simplifiées.

II.2 Un nuage de points LiDAR

II.2.1 Le LiDAR, donnée privilégiée pour l'inventaire forestier

II.2.1.1 Définition

La technologie LiDAR est une technique de mesure de distance basée sur l'analyse des propriétés d'un faisceau lumineux généré par un système LASER et renvoyé vers son émetteur. Dans le cas de systèmes d'acquisition aérien (ALS) la portée, distance séparant le capteur et la surface ayant provoqué le retour, est déduite de l'analyse du temps de parcours aller-retour de l'onde lumineuse mesurée par une horloge à haute fréquence.

II.2.1.2 Composition d'un système ALS et positionnement des retours

On distingue les systèmes LIDAR terrestres (TLS) et aéroportés (ALS). Principalement embarqués à bord d'avions ou d'hélicoptères, les systèmes ALS sont particulièrement utiles pour démasquer le relief et fournir des informations détaillées sur le sol. Les principaux produits dérivés des survols sont les MNT (Modèles Numériques de Terrain) et les MNS (Modèles Numériques de Surface).

Un système LIDAR aéroporté est constitué de deux systèmes principaux : un système POS, comprenant un système de positionnement GPS couplé à une centrale inertielle, permet d'obtenir la position ainsi que l'orientation du capteur. Un système "LSS", d'autre part,

comprenant la technologie LIDAR ainsi qu'un miroir à balayage (scanner), permet de générer et d'orienter le faisceau lumineux. Finalement, connaissant les coordonnées de l'origine du faisceau, sa portée ainsi que son orientation, on en déduit les coordonnées du point ayant généré un retour.

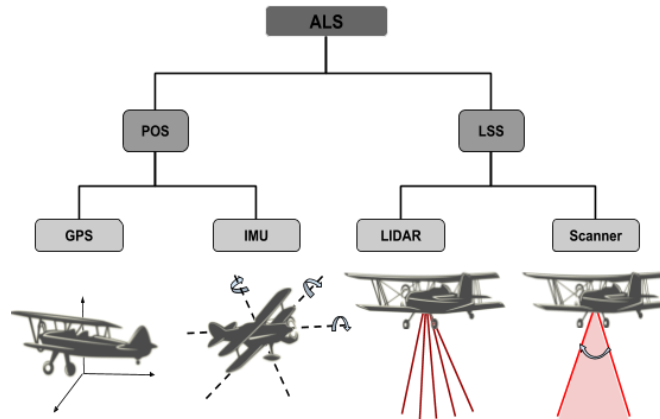


Figure 9: Composition d'un système ALS

II.2.1.3 Plusieurs types de LIDAR

Aussi bien pour les systèmes ALS ou TLS, la portée est généralement mesurée selon le principe du “Time of Flight” (ou « *Temps de vol* »), consistant à calculer le temps mis par l'onde lumineuse depuis son émission jusqu'à sa réception, selon la formule suivante :

$$d = \frac{v \times t}{2}$$

c : célérité ($m.s^{-1}$)
 t : temps de parcours de l'onde aller-retour ($m.s^{-1}$)
 d : portée (m)

On parle de LIDAR à retours discrets car le laser est pulsé. L'intensité du laser est amplifiée pour permettre une durée d'impulsion plus longue et une puissance plus élevée et ainsi augmenter la capacité de pénétration du faisceau, notamment au travers des végétaux.

On distingue trois autres principes de mesure de distances LASER :

- Le « Phase Shift » (ou « *décalage de phase* ») : Repose sur l'analyse de la différence de phases entre le faisceau émis et le faisceau reçu. On parle dans ce cas de LiDAR à onde continue, plus limités en puissance de faisceau et donc utilisés dans le cadre d'acquisitions terrestres (*figure 10*).
- La triangulation active : Repose sur la résolution de triangles en considérant deux angles et une distance. Pouvant atteindre une précision jusqu'à 0.05mm pour des distances inférieures à 5cm, elle est sensible à la luminosité et est donc surtout utilisée dans le cadre de levés d'intérieur et sur de petits objets (*figure 11*).

- Le bras de levier : Repose sur un principe similaire à celui de la triangulation active mais présente l'intérêt de pouvoir se déplacer autour de la cible.

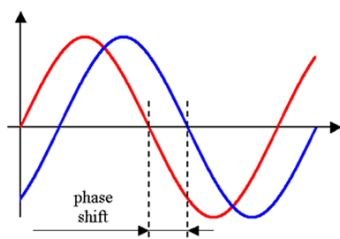


Figure 10 : Phase Shift.



Figure 11: par triangulation active sur des feuilles

II.2.1.4 Décomposition des retours

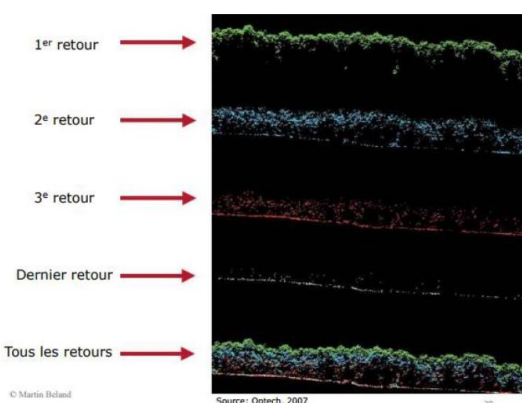


Figure 12 : Décomposition de retours LiDAR

La technologie LiDAR est aujourd'hui considérée en foresterie comme une solution d'acquisition rapide de volumes de données important et sur de vastes espaces permettant l'extraction de nombreux paramètres dendrométriques.

De plus, la décomposition des retours offre une perspective intéressante car un LiDAR suffisamment puissant traverse les feuilles et met à jour les structures recouvertes par la canopée (figure 12).

II.2.2 Le nuage de points LiDAR

II.2.2.1 Une acquisition effectuée par l'IGN

Dans le cadre d'un partenariat avec Rennes Métropole visant à acquérir des prises de vues obliques nécessaires à la mise à jour de la maquette 3D, l'IGN a procédé en 2017 un relevé LiDAR aérien simultanément à sa campagne d'acquisition photogrammétrique sur l'ensemble du territoire de la métropole. Nous disposons d'un nuage de points LiDAR dont les caractéristiques sont traduites dans le tableau suivant.

Mandaté	IGN
Système LiDAR	ALS70-HP
Densité de points	10 pts / m ² (tâche urbaine) 6 pts / m ² (périphérie)
Période de vol	Été 2017

Tableau 3: Caractéristiques techniques du nuage de points LiDAR

II.2.2.2 Un nuage de points pré-classifié par l'IGN

Le standard d'échange de la donnée LiDAR aérienne est le format. LAS défini par l'*American Society for Photogrammetry and Remote Sensing* (ASPRS). De type binaire, ce format permet d'attribuer un label numérique à chaque point du nuage, indiquant la nature

de l'objet ayant généré le retour de l'impulsion. On parle alors de segmentation. Les différentes classes d'objets ainsi que leur label numérique sont présentés dans la (figure14).

La segmentation constitue une étape clé dans la préparation des données puisqu'elle identifie des sous-ensembles et les transforme en entités indépendantes. Dans notre exemple, il est intéressant d'identifier les sous-ensembles « sol nu » et « végétation » dans une optique d'en déduire le Modèle Numérique de Terrain et le Modèle numérique de Hauteur de Canopée correspondant. (figure 13)

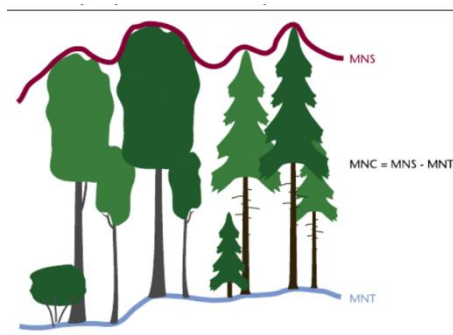


Figure 13: Calcul d'un MHC

Un MHC est une matrice pour laquelle les valeurs des pixels correspondent à la hauteur de la canopée forestière. Ces hauteurs sont obtenues par calcul de la différence entre un MNT, représentant le relief au sol et un MNS, représentant le relief des éléments en sursol⁷.

Dans notre cas, l'IGN a réalisé une pré-segmentation automatique de notre nuage. Une deuxième segmentation plus fine de la végétation est en cours de réalisation par un prestataire privé. Elle doit être livrée à la fin du mois de juin.

ASPRS Standard LIDAR Point Classes	
Classification Value (bits 0:4)	Meaning
0	Created, never classified
1	Unclassified ¹
2	Ground
3	Low Vegetation
4	Medium Vegetation
5	High Vegetation
6	Building
7	Low Point (noise)
8	Model Key-point (mass point)
9	Water
10	Reserved for ASPRS Definition
11	Reserved for ASPRS Definition
12	Overlap Points ²
13-31	Reserved for ASPRS Definition

Figure 14: Classification propre au format.LAS définie par l'ASPRS

⁷ Les points constitutifs des bâtiments et autres mobiliers urbain étant nettoyés lors de la phase de pré-traitement, le calcul du MNS ne considère plus que la végétation comme élément constituant le sursol. Ainsi, la différence des deux matrices conduit à la génération d'un MHC

III Un algorithme automatique de détection et de segmentation de la végétation

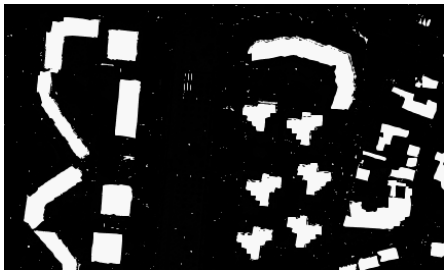
Nous présentons ici l'algorithme automatique de détection et de segmentation de la végétation développé en 2017 par C. DOCEUL exploitant une méthode traditionnelle d'ITCD basée sur l'analyse d'un modèle numérique de canopée.

III.1 Une approche traditionnelle et robuste

Dans les années 1980, les recherches relatives à la détection et la segmentation de la végétation reposaient alors uniquement sur des images satellitaires ou autres données aériennes (imageries) peu adaptées, au vu de leurs résolutions, à des analyses menées à l'échelle de l'arbre. La démocratisation de l'acquisition de données de télédétection à haute résolution au début du XXI^e siècle s'est accompagnée d'une forte croissance de l'utilisation des données actives et notamment de la donnée LiDAR, dont l'utilisation a augmenté de 80% entre 1990 et 2015, jusqu'à représenter aujourd'hui 97% de la donnée utilisée dans le domaine *Zhen Zhen et al* [21]. Depuis les vingt dernières années, de nombreux algorithmes semi- et totalement automatiques ont été développés. Ils peuvent être classés selon quatre grandes catégories d'approches : *Orientée raster* (pour la grande majorité, 66%), *orientés nuage de points* (20.6%), combinant les deux (9.6%) ou encore visant à reconstruire la forme des arbres (3.7%). Parmi ces quatre catégories c'est sans grande surprise que nous constatons une utilisation privilégiée de la première catégorie. L'utilisation de données passives ayant prédominé jusque dans les années 2005, ces méthodes reposent sur un socle solide de recherches et de développement. Elles consistent à calculer des modèles numériques de canopée (CHM) à partir desquels les cimes et couronnes sont ensuite détectées puis segmentées à l'aide de maxima locaux et d'algorithmes de « bassins versants ».

L'algorithme de *C. Doceul* [4] respecte ce protocole et comprend six phases succinctement présentées dans le tableau 4, avant d'être vue plus en détails :

III.2 Un algorithme en six phases

Illustration	Phase / Explications
	Phase 1 : Prétraitements Nettoyage du nuage de points LIDAR avant le calcul du MHC. Suppression des points de bâti autres mobiliers urbains.

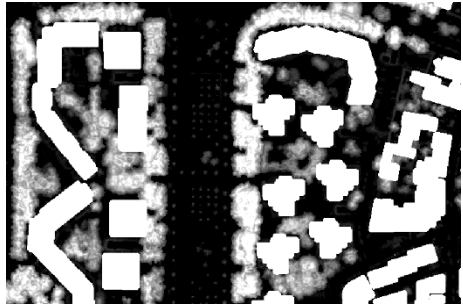

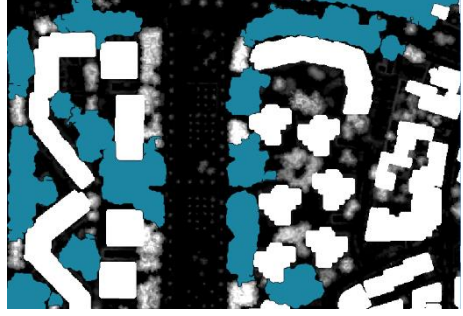
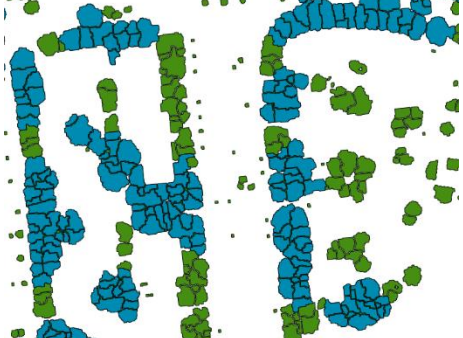

	<p>Phase 2 : Calcul du MHC</p> <p>Transformation des données discrètes en raster. Calcul d'un MHC puis amélioration par application filtres et pondération.</p>
	<p>Phase 3 : Détection et segmentation de la végétation en zones ouvertes</p> <p>Détection des maxima locaux d'abord pour identifier les cimes. Segmentation des houppiers ensuite par application d'un algorithme de bassin versant.</p> <p>Résultats sous forme de fichiers shapefiles</p>
	<p>Phase 4 : Partition "zones ouvertes" / "zones fermées"</p> <p>Distinction d'arbres isolés et de regroupements selon un label "zone ouverte" ou "zone fermée".</p>
	<p>Phase 5 : Détection et segmentation de la végétation en zone fermée</p> <p>Nouveau calcul de détection et de segmentation avec ajustement de paramètres de calculs en zone fermée</p>
	<p>Phase 6 : Préparation à la modélisation</p> <p>Normalisation du nuage LIDAR et export au format shapefile. Labélisation de chaque point par l'identifiant unique du houppier auquel il appartient.</p>

Tableau 4: Présentation des six phases de l'algorithme de détection et de segmentation développé par C.DOCEUL

III.2.1 Prétraitements : Nettoyage du nuage LiDAR

La phase de prétraitement a pour objectif principal d'isoler les points appartenant au sous-ensemble constituant la végétation du reste du nuage LiDAR.

D'un point de vue purement théorique, cette étape ne consiste qu'en la sélection des points labellisés selon l'un des trois codes numériques (3, 4 ou 5) associés aux classes de végétation basse, moyenne et haute. Toutefois, la pré-classification réalisée par l'IGN présente une confusion entre les classes « bâtis » et « végétation ». Un nettoyage complémentaire est nécessaire.

L'algorithme développé par C. Doceul [4] prévoit un nettoyage du nuage de points LiDAR basé sur une technique dite "d'emporte-pièce". Considérant les couches de données relatives aux emprises des bâtiments et autres mobiliers urbain sur le territoire de la métropole, la méthode consiste à supprimer les points contenus à l'intérieur de ces surfaces (méthode illustrée par la figure 15).

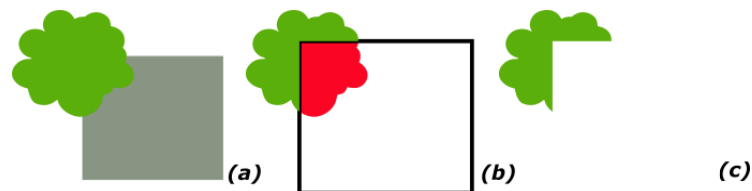


Figure 15: Nettoyage du nuage LiDAR. (a) houpier d'un arbre surplombant le toit d'un bâtiment, (b) application d'un buffer positif puis d'un clipper sur l'emprise du bâtiment (c) résultat

Si à première vue, cette méthode semble efficace, elle présente néanmoins un inconvénient majeur illustré par la figure 15. Un arbre situé à proximité d'un bâtiment peut, en fonction de la taille de buffer choisie, se voir amputé d'une partie de sa couronne. Cette troncature aura des conséquences néfastes sur la segmentation de l'arbre et donc par extension sur sa reconstruction 3D.

Dans un objectif d'extraction fine de la végétation, il est nécessaire de réfléchir à un nouveau protocole de prétraitements du nuage de points LIDAR.

III.2.2 Calcul d'un modèle numérique de hauteur de canopée (MHC)

Le MHC et le MNT comptent parmi les principaux produits dérivés des levés LiDAR aériens. Cette transformation de la donnée discrète (nuage de points) en une donnée *raster* constitue une phase dite « de *rasterisation* », réalisée au cours de la deuxième phase de l'algorithme avec le logiciel de traitement de données LiDAR ; FUSION/LDV.

Développé par l'USDA Forest Service (USFS), agence du département de l'Agriculture des États-Unis en charge de la gestion des forêts nationales du pays, FUSION/LDV est un logiciel gratuit permettant de combiner à la fois la manipulation de données LiDAR et leur analyse cartographique grâce à une interopérabilité de formats d'échanges avec des logiciels de SIG (ArcGIS de ESRI, QGIS, ...). *ESRI* [8]



Figure 16 : Logo du logiciel FUSION/LDV

Par définition, une image numérique est une matrice de pixels et, la distance séparant deux pixels correspond au pas (ou résolution) de celle-ci. On choisit ici un pas de 0.5m pour le MNT comme pour le MHC ce qui veut dire qu'une grille de pixels de dimensions $0.5 \times 0.5m$ vient se superposer aux sous-ensembles de points correspondants (le sol-nu pour le MNT et la végétation pour le MNS) et que chacun prend la valeur altimétrique moyenne des points qu'il contient.

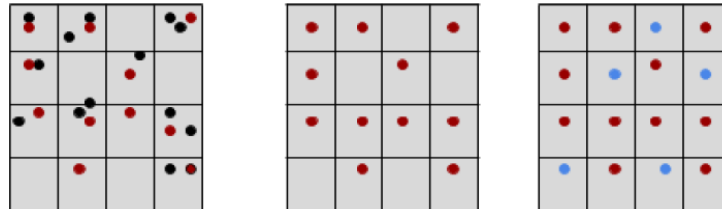


Figure 17: Rasterisation de donnée discrète

On aperçoit sur la figure 17 que certains pixels vides de points LiDAR peuvent rester sans valeur. C'est une situation courante à laquelle il est possible de remédier soit par une interpolation du nuage de points, soit par un filtrage numérique du MHC. Dans notre cas, l'algorithme prévoit un filtrage numérique médian du MHC.

III.2.3 Détection et segmentation de la végétation

Rappelons que la détection et la segmentation de la végétation selon une approche traditionnelle *orientée raster* comprend deux étapes fondées sur l'analyse du MHC : détecter les cimes par recherche de maxima locaux puis segmenter les couronnes associées par un algorithme de bassin versant.

Dans son algorithme, C. Doceul [4] utilise successivement les fonctions "TreePeaks" et "TreeWatershed", développées par Matthew PARKAN et disponibles dans la "Digital Forestry Toolbox" (DFT) du logiciel Matlab.

Dans un premier temps, la fonction « TreePeaks » parcourt le MHC et recherche des maxima locaux considérant une fenêtre circulaire de rayon fixe⁸ de 3.5m et un seuil de hauteur minimal de 3.0m. Il en résulte une couche de points des cimes au format SHP.

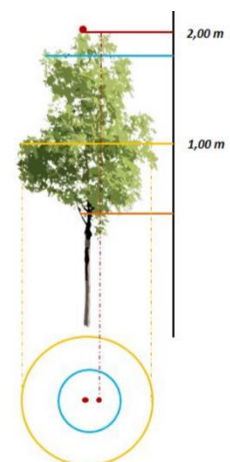


Figure 18: Segmentation des houppiers et seuil minimal de détection

Dans un second temps, à partir de chacun des sommets détectés, l'algorithme « TreeWatershed » segmente les couronnes correspondantes selon un principe de bassin

⁸ La fonction « TreePeaks » propose également un rayon de recherche variable mais les travaux de C.DOCEUL présentent cette options comme moins efficace.

versant et un seuil minimal de hauteur de 3.0m. Il en résulte une nouvelle couche *shapefile* représentant les emprises de couronnes.

Nous remarquons sur la figure 18 qu'un seuil unique de hauteur minimum pour la recherche des cimes et des couronnes peut réduire segmentation de ces dernières à un point, voire même être inexistante. Une adaptation du script est donc nécessaire pour distinguer deux seuils différents de recherche.

III.2.4 Partition des arbres en zones dites « ouvertes » et « fermées »

Par zones « ouvertes » on entend « arbres individuels » ; c'est-à-dire des arbres isolés. Par zone « fermée », on entend à l'inverse « arbres regroupés » ; soit des arbres à proximité immédiate les uns des autres et pour lesquels les houppiers s'entremêlent pour former des amalgames.

L'algorithme distingue ces zones d'après un enchaînement de buffers de tailles variables appliqués sur les couronnes (*figure 19*)



Figure 19: Partition des arbres selon l'application de buffers géométriques sur les couronnes. (a) 4 arbres différents, (b) buffer positif sur les couronnes, (c) union des polygones s'intersectant, (d) buffer négatifs (d) buffer positif pour retrouver les emprises originelles (e) 3 arbres différents dont 2 individuels (zone ouverte) et deux regroupés (zone fermée).

Les résultats présentent deux sortes d'incohérences : le regroupement d'arbres individuelles d'une part, et la considération de regroupements comme des arbres isolés d'autre part. Cette dernière se traduit visuellement par des formes géométriques complexes en zone ouvertes, très éloignées de la forme circulaire attendue pour un houppier isolé. La méthode de partition des arbres ne tient pas compte de la vraisemblance géométrique de la segmentation. Elle mérite d'être revue pour gagner en cohérence.

III.2.5 Tentative d'amélioration de la segmentation

Ayant remarqué des phénomènes de sur ou de sous-segmentation dans les zones de végétation denses, C. Doceul [4] avait envisagé d'accentuer les sommets du MHC par une pondération avec un raster de pente. Les pentes, approximativement nulles à la cime des arbres (maximum local), constituent théoriquement un moyen efficace d'identifier ces points stratégiques.

L'algorithme prévoit donc un lissage du MHC initial (MHC médian) par l'application d'un filtre gaussien (MHC dit « gaussien ») à partir duquel un raster de pentes est calculé puis utilisé pour pondérer le MHC initial (MHC dit « pondéré ») (*figure 20*). Le coefficient de pondération, noté X, est adapté en fonction du type de zone « ouverte » ou « fermée ».

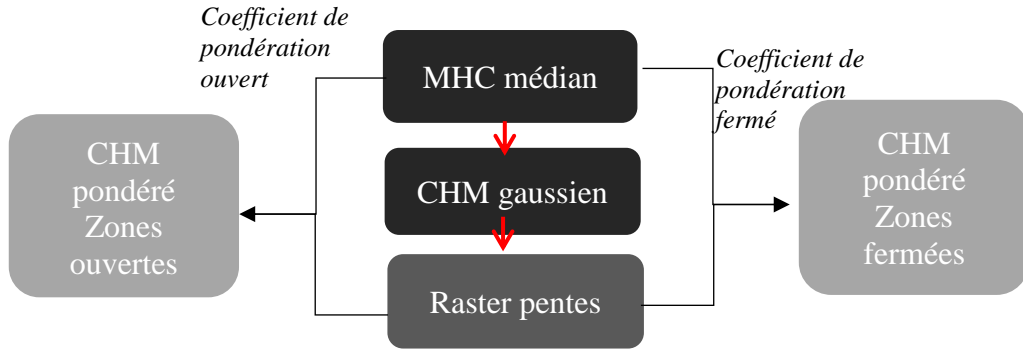


Figure 20: Pondération du MHC par un raster de pentes

$$MHC_{final} = MHC_{médian} \times \left(1 - \frac{Pente}{X}\right) ; X \in \{100; 400\}$$

Cependant, C.Doceul [4] concluait d’une inefficacité de la méthode en remarquant d’une part que les transformations successives appliquées au MHC risquaient de modifier le positionnement originel des cimes et, d’autre part, qu’en dépit de cette tentative seulement 35% des arbres détectés en zones « fermées » constituaient une bonne détection.

Tenant compte de ces observations et jugeant la méthode chronophage, nous décidons de ne pas en tenir compte dans la suite.

III.2.6 Un manque de gestion des systèmes de coordonnées

Actuellement, l’algorithme n’est pas en mesure de considérer différents systèmes de coordonnées. En entrée comme en sortie, les variables doivent être et sont projetées dans le système WGS84-UTM30 au lieu de quoi l’algorithme ne peut fonctionner.

Commençons par rappeler que tout point sur la surface terrestre peut être exprimé par des coordonnées de types planes (il s’agit alors de coordonnées dites “projetées”), géocentriques cartésiennes (X, Y, Z) ou encore géographiques (λ, φ, h) (figure 21).

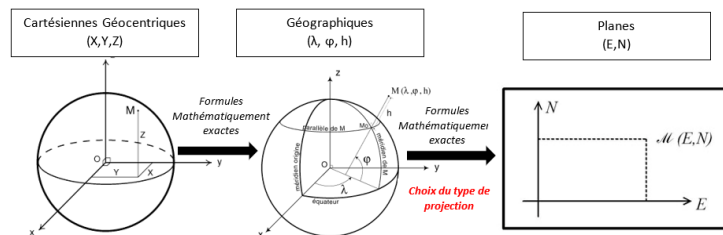


Figure 21 : Les différents types de coordonnées

Ces types de coordonnées sont dépendants du choix d’un système de coordonnées, lequel comprend trois composantes fondamentales : un système de référence (défini par des techniques géodésiques), un ellipsoïde et son méridien origine (approximation mathématique de la surface terrestre permettant le passage des coordonnées cartésiennes aux coordonnées géographiques) et un système de projection (permettant le passage des coordonnées géographiques aux coordonnées planes).

Il existe différentes méthodes permettant le passage d'un système de coordonnées à un autre et pouvant s'appliquer à chacun des types de coordonnées avec pour chacune des résultats jugés plus ou moins fiables. En général, on préfère appliquer une similitude 3D aux coordonnées cartésiennes (*figure 22*).

Il serait intéressant d'adapter l'algorithme pour qu'il soit en mesure de gérer différents systèmes de coordonnées en entrée tout en produisant des résultats projetés dans le système RGF93-CC48 (EPSG: 3948), utilisé par le service SIG de Rennes Métropole

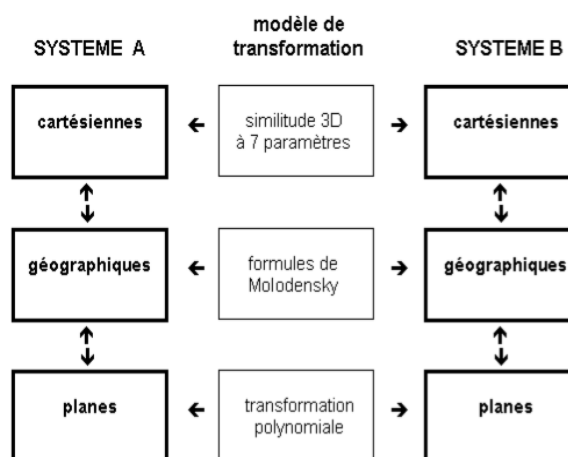


Figure 22: Tansformation de systèmes de coordonnées

IV Reconstruction 3D de la végétation : Etat de l'Art

IV.1 Techniques de reconstruction d'arbres en 3 dimensions

S'il existe aujourd'hui de nombreux travaux portant sur la détection et la segmentation de zones végétalisées basés sur des données de télédétection (images photogrammétriques ou satellitaires et levers LiDAR), leur reconstruction 3D, en revanche, ne fait pas l'objet d'une littérature abondante. Néanmoins, nous pouvons y distinguer deux types d'approches, chacune susceptible d'utiliser des données satellitaires à haute résolution, photogrammétriques ou encore lasergrammétrique : la reconstruction architecturale d'une part *Landes et al* [12] et celle de la canopée d'autre part *Zhen Zhen et al* [25].

Visant une reconstruction très fine de la forme des arbres, la première approche s'appuie sur des nuages terrestres de plusieurs millions de points pour chaque arbre. Plus approximative, et limitée à la caractérisation géométrique de la forme des arbres (volume, surface, limites ...), la reconstruction de la canopée s'appuie sur des acquisitions aériennes moins denses mais couvrant des zones très vastes.

Une première section s'intéressera d'abord brièvement aux techniques de reconstructions de l'architecture des arbres puis, dans une seconde, nous aborderons des méthodes d'estimation de formes plus générales et notamment l'approche par formes dites « alpha ».

IV.1.1 Reconstruction 3D architecturale d'arbres individuels d'après des données terrestres

Considérant des nuages de points très denses (plusieurs millions de points) acquis par système LiDAR terrestre, *Bournez et al* [2] écartent l'hypothèse de la reconstruction surfacique (ou «*meshing* ») qui produit un volume de données trop important et ne facilite pas l'interprétation des caractéristiques géométriques des arbres. Ils citent en revanche deux autres approches majeures : l'ajustement de primitives géométriques à un nuage de points et la squelettisation.

Il existe aujourd'hui de nombreuses méthodes de reconstruction d'arbres individuels basées sur l'ajustement de primitives géométriques sur des nuages terrestres denses. De manière générale, la primitive cylindrique est privilégiée *Rahman et al.* [18], *Pfeifer, et al* [17] car considérée comme la seule parvenant à reconstruire le tronc et les branches d'un arbre selon un modèle réaliste rendant compte de leurs différentes connectivités et produisant des volumes de bois précis (*figure 23*).

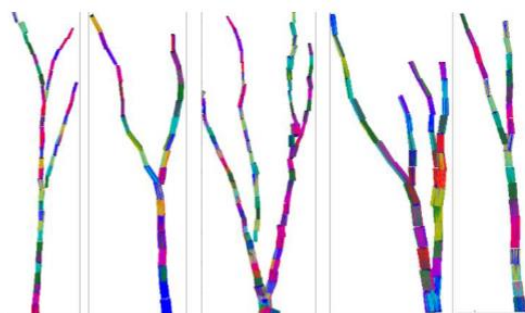


Figure 23:Reconstruction de tronc et de branches par approximation cylindrique d'un nuage de points ; Crédit : Rahman et al. 2015

La seconde approche majeure, la squelettisation, consiste à construire des modèles filaires d'épaisseur nulle (1D) traduisant une version simplifiée de la géométrie et de la topologie d'objets 3D. S'inspirant fortement des travaux de *Au et al.* [1], visant à définir une forme contractée à partir d'une surface volumique, *Cao et al.* [3] ont développé un algorithme permettant de calculer un squelette à partir d'un nuage de points, sans passer par une reconstruction surfacique intermédiaire.

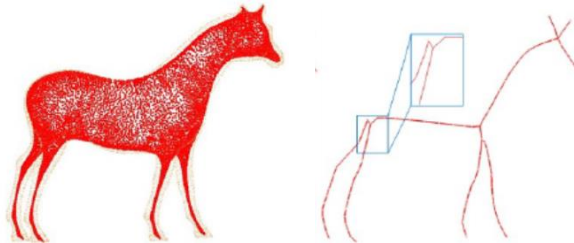


Figure 24: Illustration du principe de la squelettisation selon Coa et al. 2010. A gauche un nuage de point, à droite le squelette et ses connexités

Implémenté *Landes et al* [12] pour reconstruire les arbres dans le cadre d'un projet d'étude visant à comprendre et simuler les flux et les interactions des composantes du paysage urbain, l'algorithme de squelettisation développé par *Cao et al.* [3] a permis de reconstruire jusqu'aux pousses des arbres de manière très précise.

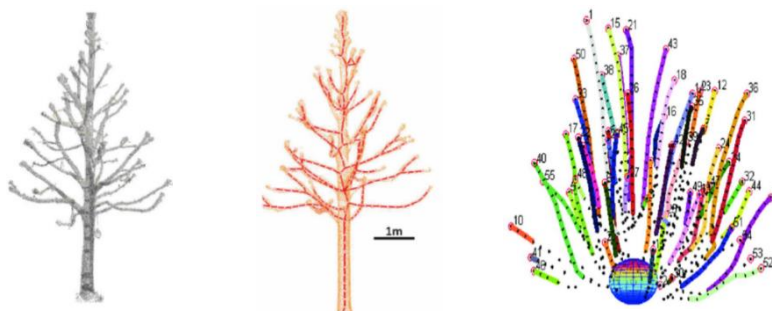


Figure 25 : Résultats obtenus par l'implémentation de l'algorithme de Cao et al à un arbre individuel. De gauche à droite ; le nuage de points LiDAR, le squelette, le squelette de pousses (extrémité des branches)
Crédit : Tania et al.

Il existe donc des algorithmes très précis en matière de reconstruction volumique de la structure d'arbres individuels. Toutefois ces approches nécessitent des nuages de points denses et supposent donc des acquisitions terrestres.

IV.1.2 De la segmentation des arbres au clustering du nuage de points LiDAR aérien

La reconstruction 3D des arbres depuis un nuage de points LiDAR aérien nécessite le « *clustering* » préalable de la donnée discrète en sous-ensembles indépendants (*clusters*). Le clustering consiste en un découpage de la donnée LiDAR en sous-groupes réunissant l'ensemble des points appartenant à un même arbre. Or, l'identification des différents arbres au sein d'un nuage de points constitue une problématique dite de « segmentation » de la végétation, problématique abordée en 2017 dans les travaux *C. Doceul* [4] selon une approche dite « basée raster » *Zhen Zhen et al* [25].

Cette approche de segmentation constitue la première des quatre approches définies par *Zhen Zhen et al* [25]. Une deuxième approche, dite « mixte », intègre la donnée LiDAR en la découpant suivant les polygones issus de la première. Par cette opération (*Reitberger et al*, [19] *C. Doceul*[4]), nous réalisons le « clustering » de notre nuage de points et obtenons les sous-ensembles (*clusters*) propres à chaque arbre. La troisième approche est dite « orientée nuage ». Dans cette hypothèse, les clusters sont directement identifiés au sein de la donnée LiDAR grâce à des algorithmes de K-moyenne ou de voxelisation ⁹. Disposant déjà dans notre cas de sous-ensembles générés selon une approche mixte, nous ne développerons pas plus ces techniques. Enfin, la quatrième approche est dite « orientée reconstruction de forme ». Elle s'appuie sur des clusters réalisés au préalable par l'une des techniques précédentes et regroupe différentes techniques géométriques de reconstruction 3D de formes à partir de données discrètes. Parmi les deux plus connues : les techniques de « *convex hull* » et d'« *alpha shape* » ont déjà très largement fait leurs preuves dans le domaine de la reconstruction 3D d'arbres individuels.

IV.1.3 Reconstruction 3D d'arbres individuels d'après des données LiDAR aériennes

Les techniques géométriques de reconstruction de formes à partir d'échantillons discrets de points d'enveloppe convexe ("*convex hull*") et de formes alpha ("*alpha shape*") sont étroitement liées à triangulation dite « *de Delaunay* » et à son graphe dual : le diagramme de Voronoï. Intéressons-nous donc dans un premier temps à la notion de subdivision de l'espace selon des régions de Voronoï ou des triangles de Delaunay.

IV.1.3.1 Diagramme de Voronoï et triangulation de Delaunay

Soit $X = \{X_1, \dots, X_n\}$ un ensemble fini de points dans R^n . Le diagramme de Voronoï constitue une partition de l'espace en n régions V_i dites « *de Voronoï* » où, pour chaque $i \in \{1, \dots, n\}$, la région V_i contient tous les points plus proches de X_i que de n'importe quel autre $X_{j \neq i}$. On dit que X_i constitue le « germe » de la région V_i . Ainsi, considérant deux points distincts deux à deux X_i et X_j , la région V_i peut être définie de la façon suivante :

$$V_i = \{x \in R^n : \|x - X_i\| \leq \|x - X_j\| \text{ pour tout } X_{j \neq i} \in X\}$$

La frontière séparant les régions de Voronoï V_i de deux germes distincts se situe sur la médiatrice les séparant. En effet, les points se trouvant sur cette médiatrice étant équidistants des deux germes on ne peut affirmer qu'ils se situent dans l'une ou l'autre des régions. Pour une meilleure compréhension, la *figure (26)* illustre la création de région dites « *de Voronoï* » pour un nombre croissant de « germes » considérés.

⁹ Le voxel constitue une analogie 3D du pixel, autrement dit un cube. Cette notion est développée en VI.4

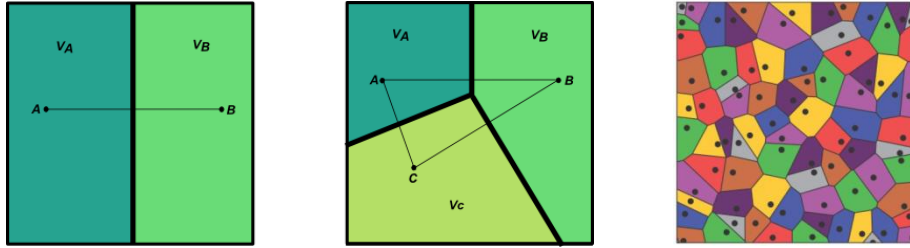


Figure 26: Construction d'un diagramme de Voronoï pour un nombre croissant de germes

Parmi les propriétés des polygones « *de Voronoï* », citons :

- Les régions de Voronoï forment des polygones clos et convexes ;
- Pour un germe X_i situé sur une frontière de l'enveloppe convexe de l'échantillon, la région V_i est infinie ;
- Pour chaque sommet du diagramme de Voronoï, le cercle passant par les points de l'ensemble X voisins ne contient aucun autre point de X (*figure 27*)

Nous rappelons également que des « *voisins de Voronoï* » sont des « germes » X_i et $X_{j \neq i}$ pour lesquels les régions V_i et $V_{j \neq i}$ partagent une frontière commune.

Graphe dual du diagramme de Voronoï, la triangulation de Delaunay est une deuxième technique de partition de l'espace qui relie par des segments de droite les germes X_i et X_j « voisins de Voronoï ». (*figure 28*).

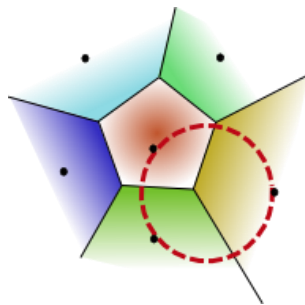


Figure 27: Cercle reliant des voisins de Voronoï

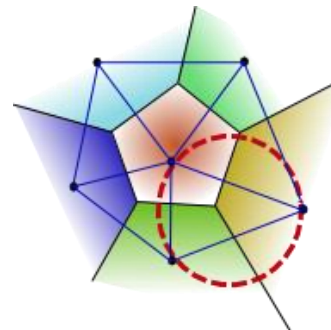


Figure 28: Triangulation de Delaunay

On appelle triangle de Delaunay, un triangle dont le cercle circonscrit ne contient aucun autre point que ses sommets. Nous comprenons alors la dualité des deux graphes ; chaque germe du diagramme de Voronoï est un sommet de la triangulation de Delaunay. Nous remarquons que cette dualité peut être étendue dans un espace à trois dimensions où les régions de Voronoï deviennent des polyèdres et les triangles des tétraèdres de Delaunay. Dans cette hypothèse, la sphère circonscrite à un tétraèdre de Delaunay ne contient aucun autre point que ses sommets.

En 2016, la ville de Montréal (Canada) a expérimenté un processus automatisé de génération d'un modèle numérique de canopée par corrélation dense d'images photogrammétriques aériennes [13]. La reconstruction 3D de la canopée (*figure 29*) reposait

alors sur une approche de type « *meshing* » consistant à construire une surface triangulaire (TIN) d'après une triangulation de Delaunay (figure 30).

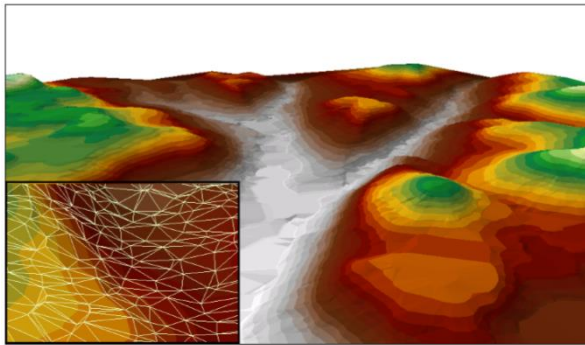


Figure 30 : Structure TIN, Crédit Stéphane R. U LAVAL



Figure 29: Reconstruction de la canopée d'après une surface TIN

Nous avons déjà noté *Bournez et al. [2]* que le *meshing* ne permet pas une interprétation efficace des caractéristiques géométriques des arbres. Néanmoins, *Zhen Zhen et al. [25]* présentent les techniques de l'enveloppe convexe (de l'anglais *convex hull*) et de la forme alpha (de l'anglais *alpha shape*), qui s'appuient toute deux sur la triangulation de Delaunay, comme les plus reconnues en termes de reconstruction 3D des arbres.

IV.1.3.2 Estimateur convexe

L'étude de la structure d'un nuage de points est un problème récurrent et fait actuellement l'objet de nombreux travaux de recherche. En géométrie computationnelle, la reconstruction de surfaces (ou de volumes) à partir de données discrètes suppose la définition d'un estimateur de la forme attendue tenant compte de contraintes géométriques. À titre d'exemple, la couronne d'un arbre peut être typiquement considérée comme présentant une forme convexe, faisant de l'enveloppe convexe son estimateur naturel.

Soit E un sous ensemble de R^n . E est dit convexe si pour tout couple de point $(x, y) \in E^2$, le segment $[x, y]$ est inclus dans E . On définit alors l'enveloppe convexe comme le plus petit (au sens de l'inclusion) ensemble convexe contenant un ensemble donné.

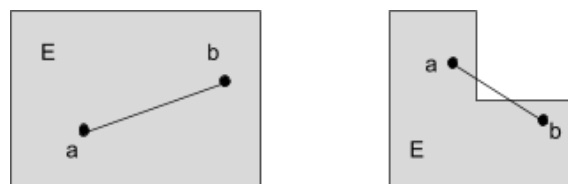


Figure 31: Illustration d'un ensemble convexe (a) et non convexe (b)

La figure 32 illustre les résultats obtenus par *Rahman et al. [18]* après combinaison d'une reconstruction de tronc par des cylindres et du houppier d'après un estimateur de type convexe. Bien que déjà très intéressants, on observe une forte simplification du volume des houppiers. En effet, de par sa forme irrégulière, les points formant la frontière de la couronne d'un arbre n'appartiennent pas tous à son enveloppe convexe (figure 33). La reconstruction selon un estimateur convexe entraîne alors une forte généralisation de leur forme.

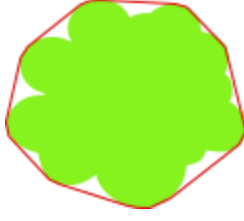


Figure 33: Forme irrégulière des couronnes

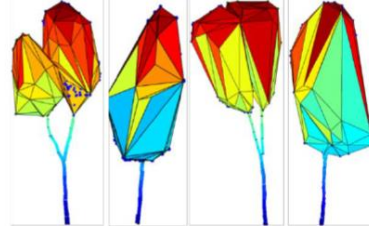


Figure 32: Reconstruction d'arbres individuels par combinaison d'enveloppe convexes et de modèles cylindriques. Crédit Rahman et al.

Dans un objectif de considérer des formes plus complexes, deux nouveaux estimateurs ont été développés visant à généraliser l'approche par enveloppe convexe et à permettre de nouvelles perspectives de reconstruction. Il s'agit des estimateurs " α -convex hull" Pateiro-Lopez et al. [16] et " α -shape" (Edelsbrunner et al, [6] Edelsbrunner and Mücke [7]). Dans chacune de ces deux hypothèses, la reconstruction de forme est étroitement liée à la triangulation de Delaunay et, le niveau de détail est directement dépendant du critère α correspondant au rayon d'une boule ouverte.

IV.1.3.3 Estimateur α -convexe

Dans un premier temps, présentons brièvement la technique de généralisation de l'enveloppe convexe d'un ensemble discret de points ; la reconstruction d'une enveloppe α -convexe.

Soit $B(P, r)$ une boule ouverte centrée en P et de rayon r , définie dans un espace métrique E comme l'ensemble des points pour lesquels la distance P est strictement inférieure à r :

$$B(P, r) := \{M \in E \mid d(M, P) < r\}$$

Soit $X = \{X_1, \dots, X_n\} \in E^n$ un échantillon de points de E , $C(X)$ l'enveloppe convexe de X et $VD(X)$ le diagramme « de Voronoï » associée. Soit C_X son complémentaire et ∂X sa frontière.

Pateiro-Lopez et al. [16] définissent alors l'enveloppe α -convexe de X notée $C_\alpha(X)$ d'après la relation suivante :

$$C_\alpha(X) = \bigcap_{\{B(x, \alpha) : B(x, \alpha) \cap X = \emptyset\}} C_{B(x, \alpha)}$$

Ils définissent ensuite le complémentaire de l'enveloppe α -convexe de X , noté $C_{X(\alpha)}$, comme l'union de l'ensemble des boules ouvertes de rayon α ne contenant aucun point de X .

$$C_{X(\alpha)} = \bigcup_{\{B(x, \alpha) : B(x, \alpha) \cap X = \emptyset\}} B(x, \alpha)$$

Définir le complément $C_{X(\alpha)}$ permet donc de définir les espaces où il n'existe aucun point de X .

Enfin, considérant le « Lemme » suivant de *Edelsbrunner et al. [6]* :

Lemme : Soit $B(x, \alpha)$ une boule ouverte ne contenant aucun point d'un ensemble (X). Soit $B(x, \alpha)$ se trouve entièrement à l'extérieur de l'enveloppe convexe de X soit il existe une boule ouverte disjointe de X contenant $B(x, \alpha)$ et centrée sur un segment « de Voronoï »

Il résulte que définir le complément de l'enveloppe α -convexe implique de déplacer une boule ouverte de rayon α et centrée sur la frontière commune à deux régions « de Voronoï ». Deux « germes » voisins « de Voronoï » situés sur la frontière de $B(x, \alpha)$ sont reliés par un arc de cercle (figure 34).

Sur la figure 35 ci-dessous pour les trois germes A, B et C, nous posons l'hypothèse d'une boule ouverte $B(x, r)$ centrée sur la frontière commune aux régions de Voronoï correspondantes. Appartenant chacun à la frontière de la boule ouverte, les germes sont reliés par un arc de cercle.

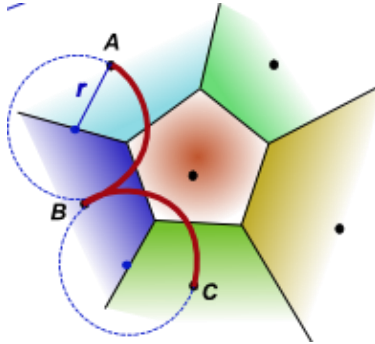


Figure 34: Voisins de Voronoï joint par un arc de cercle

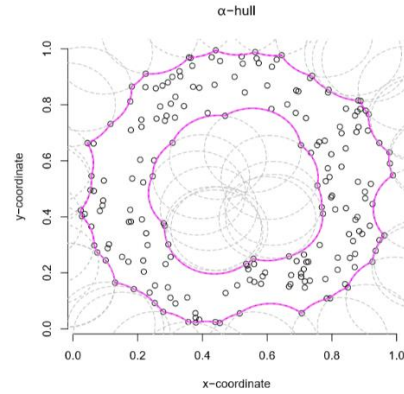


Figure 35: Enveloppe α -convexe et son complément

La figure 36 ci-dessous illustre les nouvelles perspectives de reconstruction de formes offertes par l'enveloppe α -convexe par comparaison avec l'enveloppe convexe simple d'un ensemble. Néanmoins, définies par des arcs de cercles (en 2D) ou des capuchons sphériques (en 3D), les formes géométriques générées par une enveloppe α -convexe peuvent rapidement devenir lourdes à traiter. Ainsi, dans un souci de simplification nous considérons la reconstruction par « formes alpha » visant à remplacer les arcs de cercles par des polygones et les capuchons sphériques par des surfaces polyèdres.

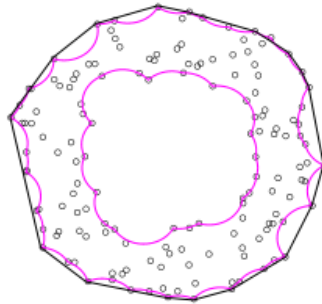


Figure 36: Comparaison des critères convexe stricte (noir) et alpha (rose) dans un espace 2D

IV.1.3.4 Reconstruction par formes « *alpha* »

D'abord développées dans un espace euclidien 2D [Edelsbrunner, Kirkpatrick, and Seidel (1983)] avant d'être étendue à un espace 3D, Edelsbrunner and Mucke (1994), les formes « *alpha* » constituent une deuxième forme de généralisation de l'enveloppe convexe d'un ensemble E .

Soit $X = \{X_1, \dots, X_n\} \in E^n$ un échantillon de points E , $C(E)$ l'enveloppe convexe stricte de E , $VD(E)$ le diagramme de Voronoï et $VT(E)$ la triangulation de Delaunay associée.

L'algorithme de Edelsbrunner and Mucke (1994) repose sur deux notions principales : les α -extremes, appartenant à la forme « *alpha* » que l'on cherche à reconstruire, et les α -voisins qui sont des paires de points α -extremes reliés par un segment de droite.

Soit $i \in \{1, \dots, n\}$, $C(E)$ et $VD(E)$ et W_i l'ensemble des voisins de X_i au sens de $VD(E)$ (c'est-à-dire, l'ensemble des germes dont les régions associées partagent une frontière avec la région associée à X_i). X_i est un α -extreme si l'une des deux conditions suivantes est vérifiée :

- $X_i \in C(E)$.
- $\max\{\|X_i - Y\| \geq \alpha, Y \in W_i\}$

Dans l'exemple de la figure 37, les points X_1 , X_2 et X_3 appartiennent à l'enveloppe convexe (en rouge) et sont donc des α -extremes pour tout $\alpha > 0$. Le point X_4 quant à lui est un α -extreme si d_{max} (en bleu) est supérieur à α

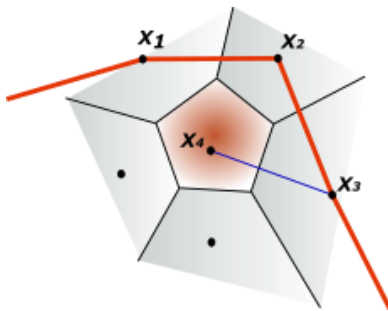


Figure 37: α -extreme

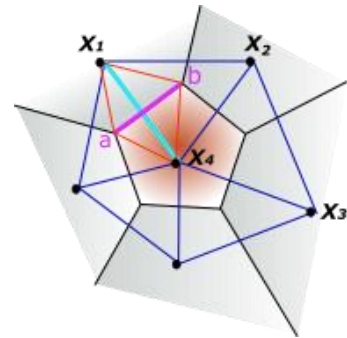


Figure 38: α -voisins

Une fois les α -extreme identifiés, l'algorithme recherche parmi eux les points constituant les α -voisins à relier par des segments de droite pour former la forme « *alpha* »

Considérant un segment dans $VT(E)$ reliant deux α -extremes X_i et X_j et son bord dual dans $VD(E)$, les points X_i et X_j sont des α -voisins si parmi les quatre distances en rouge sur la figure 38 par rapport aux extrémités du segment $VD(E)$ ($[a, b]$ sur la figure), les distances d_{min} et d_{max} respectent la condition suivante :

$$d_{min} < \alpha < d_{max}$$

Finalement, la forme alpha définit le graphe en ligne droite reliant les α -voisins. Ainsi pour un rayon infiniment grand, la forme « alpha » sera identique à l'enveloppe convexe. À l'inverse, plus le rayon considéré sera petit, plus la forme alpha rétrécira jusqu'à se résumer au nuage de points initial.

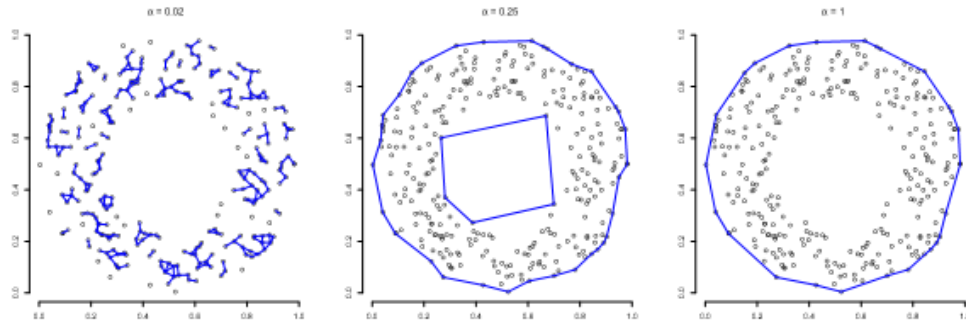


Figure 39: Influence du rayon sur la forme alpha. Valeur croissante du rayon de la gauche vers la droite. Crédit : Lafarge et al

Pour une étude approfondie des algorithmes d'enveloppes convexes, α -convexes et des formes « alpha », nous nous référons notamment *Pateiro-Lopez et al. [16]*, *Lafarge et al. [11]*

Implémentée par *Zhu et al. [26]*, la reconstruction de type forme « alpha » appliquée à un érable conduit au résultat suivant :

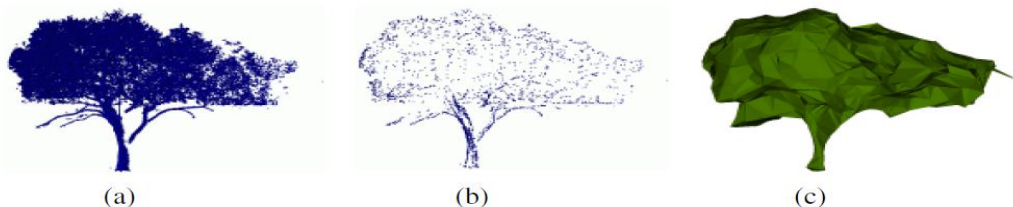


Figure 40 : Reconstruction de type alpha shape d'un érable

Résultats obtenus par C.Zhu et al. 2008 par expérimentation de la reconstruction géométrique de type "alpha shape". Extraction de la forme de la canopée d'un érable : (a) le nuage de points source, (b) les points appartenant à l'enveloppe convexe (limite extérieure du nuage), (c) limite maillée finale

On l'aura remarqué, ces algorithmes de reconstruction 3D (aussi bien l'approximation par primitives géométriques, que la squelettisation, l'enveloppe convexe ou encore les formes alpha) supposent de renseigner des nuages de points constituant un arbre individuel. On parle alors de "clusters", c'est-à-dire de regroupement de données discrètes d'un ensemble en sous-échantillons propres à chaque arbre. Profitant des travaux antérieurs de détection et de segmentation de la végétation, nous clustérons notre nuage de points LiDAR d'après les emprises de couronnes définies géométriquement par *C. Doceul [4]* sur la base d'une analyse de CHM.

V Modélisation 3D de la végétation de Rennes Métropole

Nous avons présenté nos données (LiDAR et 3D), les travaux de détection et de segmentation réalisés par C. DOCEUL ainsi qu'un état de l'art relatif aux techniques existantes en matière de reconstructions d'objets 3D à partir de données discrètes, intéressons-nous donc maintenant au cœur de ce travail de recherche : La reconstruction 3D de la végétation de Rennes Métropole.

Nous l'avons vu en II.1.2, le service SIG dispose d'ores et déjà d'un module de reconstruction de type *billboards* de la végétation mais peu adapté à la réalisation de simulations urbaines pertinentes.

Parmi les différentes méthodes présentées en IV, nous considérons l'utilisation des formes alpha comme la solution la plus adaptée à notre problématique. En effet, permettant d'estimer au plus près la forme suggérée par un nuage de points selon un niveau de détail paramétrable, elles présentent un intérêt particulier pour la reconstruction d'objets destinés à être intégrés dans une maquette 3D.

V.1 Préparation de la clusterisation du nuage de points LiDAR : Adaptation de l'algorithme de détection et de segmentation

V.1.1 Pré-traitements : Extraction de la végétation

L'objectif principal de cette première étape est d'extraire du nuage LiDAR le sous-ensemble constituant la végétation. Théoriquement, sur la base d'un nuage finement classifié (selon les spécifications de l'ASPRS), cela se résumerait à sélectionner les points labellisés selon les codes numériques 3, 4 et 5 distinguant respectivement la végétation basse, moyenne et haute. Or, comme nous l'avons déjà remarqué une confusion existe entre les classes « bâti » et « végétation ». En effet, nous observons dans ce sous-ensemble la présence de points résiduels pouvant appartenir à des toitures, balcons ou encore arêtes de façades.

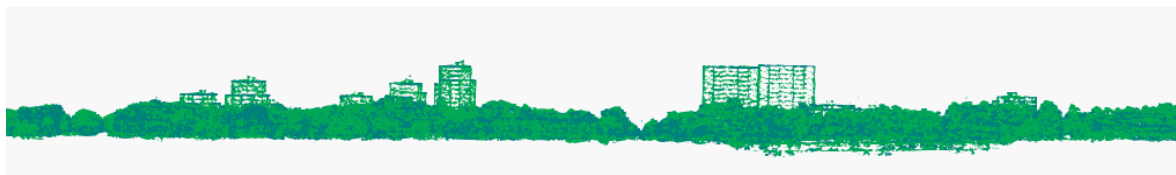


Figure 41: Ligne d'horizon présentant la classe "Végétation" du nuage de points LiDAR

Cette confusion pouvant impacter la détection des cimes (basée sur la recherche de maxima locaux) et, par extension, la segmentation des couronnes, un nettoyage complémentaire du sous-ensemble « végétation » est nécessaire.

V.1.1.1 Suppression des résidus de bâtiments

Dans l'algorithme de C. Doceul, les points résiduels de bâtis sont détectés à l'intérieur de surfaces définies par les emprises des bâtiments. D'un point de vue 3D, cette technique tend à supprimer l'ensemble des points se trouvant à l'intérieur d'un emporte-

pièce de hauteur infinie, impactant alors la végétation pouvant surplomber certains bâtiments. Pour limiter l'impact du nettoyage sur la végétation, il est nécessaire de considérer un seuil de hauteur maximale de détection des points de bâtis définie par la hauteur effective de ces constructions. La modélisation des bâtiments en LoD 2 rend compte de la forme de leur toit (I.2.2.2). La couche de donnée correspondante est exploitable dans la plupart des logiciels de manipulation de donnée 3D dont CloudCompare. Considérant, d'une part, le sous-ensemble de points « végétation » et, d'autre part, la couche 3D du bâti, il est possible de sélectionner les points se trouvant dans un intervalle de distance fixé par rapport aux faces constituant les bâtiments (plugin « Cloud/Mesh»).

Estimant très largement une profondeur de balcon de 1.5m, nous sélectionnons puis supprimons les points se trouvant dans un intervalle de 0 à 1.5m par rapport à chacune des faces de ces solides (figures 42 et 43).

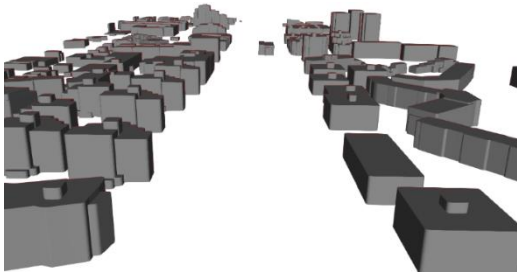


Figure 42 : Couche objet de bâtis de Rennes Métropole



Figure 43 : Points résiduels identifiés par le plugin Cloud/Mesh de CloudCompare

Bien qu'efficace cette méthode n'est pas parfaite. Des résidus persistent notamment en présence de lucarnes de toits, structures non modélisées en LoD 2. Nous observons également la présence de bruit dans le nuage qu'il est donc nécessaire de filtrer.



Figure 44: Lucarne de toit

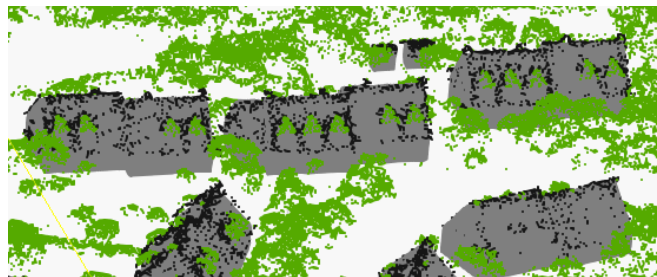


Figure 45: Résidus liés aux lucarnes de toits dans le nuage de points LiDAR

V.1.1.2 Nettoyage par test statistique : Statistical Outlier Removal

Le bruit est traité par un test statistique de type SOR (*Statistical Outlier Removal*), disponible dans CloudCompare, fortement inspiré de l'algorithme implémenté dans la *Point Cloud Library* (PCL).

Le principe consiste à éliminer les points isolés. Pour chaque point étudié, une distance moyenne est calculée par rapport à ses voisins. En considérant alors que l'ensemble des distances moyennes du nuage suivent une distribution gaussienne (μ, σ), on identifie comme

points aberrants tous les points éloignés de leurs voisins d'une distance d respectant la condition suivante :

$$d \geq \mu + a \times \sigma$$

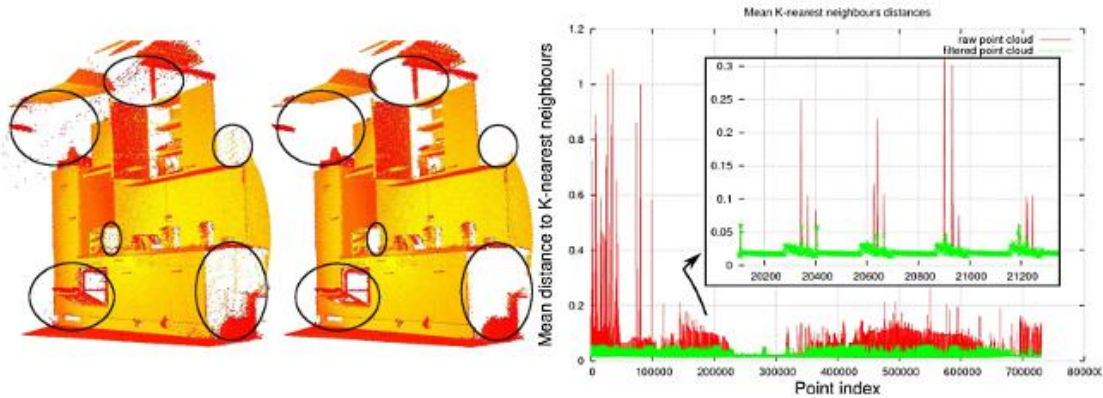


Figure 46 : Statistical Outlier Removal. Crédit PCL

Nous remarquons que les plus petits arbres présentant un intérêt en termes de reconstruction 3D comportent une vingtaine de points. Pour un nombre trop important de voisins étudiés, le risque est de fausser les distances moyennes en recherchant des points trop éloignés (arbre voisin par exemple). Nous calculons donc les distances moyennes pour chaque point en fonction d'échantillons de 20 voisins et pour une déviation standard de un.

V.1.1.3 Nettoyage par analyse de densité

Les petits agglomérats peuvent regrouper des résidus de bâti (lucarnes de toits), ou des arbres définis par trop peu de points pour espérer être correctement segmentés et reconstruits en 3D. Pour les nettoyer, nous utilisons un nouvel outil disponible dans CloudCompare : « Density », permettant l'analyse de la densité d'un nuage de points selon trois aspects :

- Le nombre de voisins : analyse du nombre de pixel voisins dans un rayon R donné autour d'un pixel source
- Une densité de surface : Nombre de pixels voisins divisés par la surface

$$D = \frac{N}{4\pi \times R^2}$$

- Une densité en volume : Nombre de pixels voisins divisé par le volume de la sphère

$$D = \frac{N}{\frac{4}{3} \times \pi \times R^3}$$

On comprend naturellement que les points situés à la frontière des couronnes voient le nombre de leurs voisins diminuer de manière considérable par rapport à ceux se trouvant en leur centre. De ce fait, une distinction basée sur l'analyse du nombre de voisins n'est pas adaptée car entraîne la suppression des frontières des houppiers (figure 47).

L'analyse par densité de surface considère le nombre de points contenus dans un cylindre de hauteur infinie et propose donc une solution efficace de détection des petits agrégats isolés.

En effet, on ne compare plus le nombre de voisins propre à chaque point mais, la densité de points présents dans une surface centrée sur ce point.

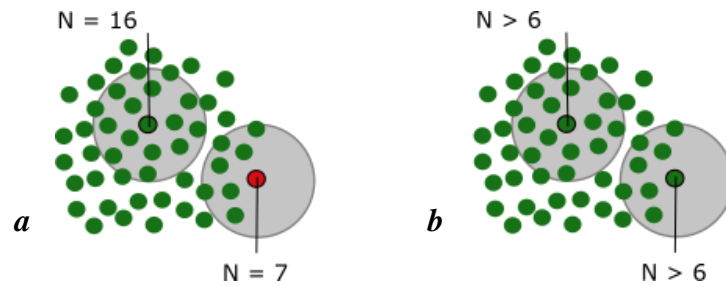


Figure 47: Comparaison de l'analyse de densité par nombre de voisins ou par surface

Dans la situation (a), un point sur la frontière de la couronne n'a que 7 voisins par rapport à un point au centre du houppier en présentant 16. Il sera supprimé par une analyse de type « nombre de voisins ». Dans le cas (b), chacun des deux points étudiés présente plus de 6 voisins dans un cylindre de hauteur infini centré sur ces points. Le point à la frontière ne sera donc pas supprimé d'après une analyse de densité.

Considérant un rayon R de 1.5m et un seuil de densité de 2.8, nous supprimons alors du nuage de points tous les agrégats présentant moins d'une vingtaine de points dans un cylindre infini de 3 m de diamètre.

Finalement la combinaison des test SOR et de densité permettent d'affiner l'extraction du sous-ensemble constituant la végétation en y supprimant le bruit ainsi que les points résiduels de bâti tout en minimisant l'impact du nettoyage sur les couronnes. Le sous-ensemble « végétation » est extrait en tant que nouveau nuage de points indépendant appelé « nuage végétation ».

V.1.2 Détection et segmentation des arbres

Considérant à présent deux nuages de points en entrées : le nuage LiDAR brut et le « nuage végétation », nous adaptons les étapes de calcul du CHM et de détection et segmentation de la végétation.

V.1.2.1 Calcul d'un modèle numérique de canopée

Initialement prévue pour ne considérer qu'un seul nuage de points en entrée, nous adaptons la méthode de calcul du CHM développée par *C. Doceul* pour que celle-ci considère nos deux nuages de points : le LiDAR brut comme source de calcul du MNT et, le « nuage végétation » comme source de calcul du MHC.

Sur la base du nuage LiDAR brut, la fonction « GroundSurfaceCreate » de FUSION génère d’abord une grille MNT avec une résolution de 0.5m. S’appuyant ensuite sur ce MNT et considérant le nuage de points « végétation », la fonction « CanopyModel » génère dans un second temps un CHM de résolution identique. Enfin, un filtre médian est appliqué pour affecter une valeur interpolée aux pixels vides de la matrice. Nous pouvons observer l’impact positif de notre phase de prétraitement des données sur le calcul du CHM où nous ne sommes distinguons que peu de résidus (figure 48).

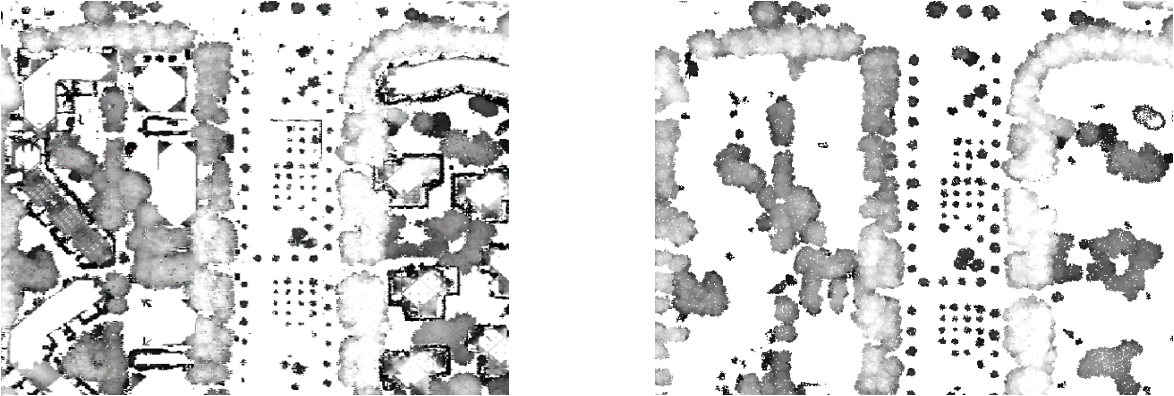


Figure 48: Calcul de CHM sur le secteur de l'avenue Henri Freville. A gauche le résultat obtenu par C. Doceul, à droite le nôtre

Il est intéressant de soulever que si la fonction « CanopyModel » propose par défaut de réaliser elle-même une interpolation pour renseigner des valeurs aux pixels vides, il est préférable de la désactiver dans notre cas car elle dégrade la qualité du CHM en sortie en assignant des valeurs à des pixels sensés rester nuls (espaces entre les arbres).

V.1.2.2 Détection et segmentation

La détection et la segmentation des arbres est réalisée d'après la méthode développée par C. DOCEUL. Néanmoins, suite à la remarque soulevée en III.2.3, nous adaptons l’algorithme et distinguons deux seuils différents de hauteurs minimales de recherche pour les cimes et les houppiers. Cette distinction propose d’ailleurs une perspective intéressante puisqu’elle offre la possibilité de mener des analyses en tenant compte d'un critère de hauteur paramétrable.

Dans un objectif de segmenter la végétation dans sa globalité, nous appliquons des seuils très larges de recherche : 1,5 pour les cimes et 1 m pour la canopée. De cette phase résultent alors deux couches de données au format .SHP. La première ("Cime_all.shp") traduisant l’emplacement des cimes détectées, la seconde (Couronnes_all_clean") représentant les emprises des couronnes segmentées (voir tableaux 5 et 6). Il est important de noter que pour une cime détectée, une unique couronne est segmentée.

"Cimes_all.shp" Système de projection : RGF93-CC48		Type : points
Attributs	Description	
X, Y, Z	Coordonnées de chaque cime détectée	

Tableau 5 : Description de la couche "Cimes_all.shp"

"Couronnes_all_clean.shp"	Système de projection : RGF93-CC48
	Type : polygone

Attributs	Description
ID	Identifiant numérique unique attribué à chacune des couronnes segmentées

Tableau 6: Description de la couche "Couronne_all_clean.shp"

Une analyse visuelle de la couche "Couronnes_all_clean.shp" nous amène à définir quatre catégories d'entités segmentées : Les houppiers individuels (caractérisés par une couronne sensiblement circulaire et d'un seul tenant), les houppiers fragmentés (caractérisés par plusieurs entités proches et supposant une forme circulaire), les "donuts" (houppiers pour lesquels la surface est trouée) et enfin, les amalgames (caractérisés par des formes géométriques complexes résultant du regroupement de plusieurs entités). Ces différentes catégories, illustrées par la figure (figure 49), traduisent une limite propre à la détection et la segmentation de la végétation d'après une approche orientée raster.

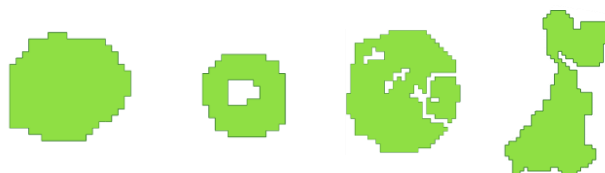


Figure 49: Quatre catégories de houppiers segmentés. De la gauche vers la droite : houppier individuel ; donuts; houppier fragmenté (et donuts); amalgame

V.1.3 Préparation des données

V.1.3.1 Une distinction de zones nécessaires à la reconstruction

Rappelons d'abord que l'objectif de cette section consiste découper le nuage "végétation" en autant de sous-ensembles indépendants que de couronnes segmentées. Rappelons également que le standard CityGML, format dans lequel seront exprimés les arbres en 3D à la fin du traitement, distingue les arbres isolés (classe "*SolitaryVegetationObject*") des zones de végétation denses (classe "*PlantCover*"). Dans ce contexte, nous devons distinguer des zones « ouvertes » et « fermées » afin de produire des couches de données adaptées.

Nous l'avons vu en III.2.4, l'algorithme de détection et de segmentation prévoit une phase de partition de la végétation en secteurs plus ou moins dense mais ne tient pas compte de la vraisemblance géométrique des couronnes. Nous proposons dans la suite d'adapter la méthode existante en observant comme critère principal non plus un enchainement de buffers mais la cohérence géométrique des couronnes segmentées. Bien que ce postulat ne soit pas applicable sur des arbres taillés (cas d'ailleurs très minoritaire sur la métropole de Rennes), nous considérons à présent qu'un houppier "correctement" segmenté se doit de respecter une forme sensiblement circulaire.

V.1.3.2 Une nouvelle approche de partition

La partition de la végétation fait l'objet du script « Partition.fmw » développé sous FME. L'idée générale consiste à définir le plus petit cercle englobant propre à chacune des couronnes segmentées puis à comparer les aires de chacun des couples cercle / houppier. En

effet, on comprend naturellement que plus une couronne aura une forme circulaire, plus sa surface aura tendance correspondre à celle de son plus petit cercle englobant (figure 50).



Figure 50 : Suppression des objets incohérents d'après leur forme. De gauche à droite : un houppier bien segmenté, amalgame d'arbre, résidu divers

Nous l'avons remarqué en V.1.2.2, parmi les quatre catégories de couronnes segmentées les *donuts* ne constituent pas des surfaces pleines sur lesquelles baser cette analyse.

Notre méthode reposant d'abord et avant tout sur l'analyse d'un rapport de surface, nous commençons par nous assurer de ne considérer que des surfaces pleines en traitant les "*donuts*" à l'aide du transformer "*DonutHoleExtractor*".

Ensuite, après avoir calculé l'aire propre à chacune des surfaces et supprimé les petits éléments (surface inférieure à 10m²), nous générons les plus petits cercles englobants leur étant associés (transformer "*MinimumSpanningCircleReplacer*"). Nous appliquons alors un filtre considérant appartenir à une zone ouverte les entités respectant la condition suivante :

$$R = \frac{S_{\text{Couronne}}}{S_{\text{Cercle}}} \geq 60\%$$

En réalité, la forme d'une couronne étant plus de forme ellipsoïdale que circulaire, nous ne pouvons pas appliquer une condition trop stricte sur le rapport des aires. Sur la base d'un contrôle visuel effectué sur notre échantillon, un seuil de 60% apparaît être un bon compromis pour distinguer les formes sensiblement circulaires des formes trop complexes. Pour chaque couronne, la valeur de ce rapport constitue un attribut "Probabilité".

Les couronnes ne respectant pas ce critère sont considérées comme de formes trop complexes traduisant des regroupements d'arbres. Les houppiers à proximité les uns des autres de moins de 0.50m sont fusionnés au sein d'agrégats par la succession d'un buffer positif de 0.5m, de la fusion des frontières communes puis d'un buffer négatif de 0.5m. Nous obtenons ainsi les emprises globales des secteurs dits « fermés ». Notons qu'après une nouvelle application d'un transformer "*DonutHoleExtractor*" nous nous assurons que ces surfaces soient fermées.



Figure 51: Regroupement des zones fermées

Finalement, suite à cette phase de partition, nous disposons de deux couches shapefile, « Zones_ouvertes.shp » et « Zones_fermees.shp », représentant les emprises pleines et fermées des couronnes réparties dans chacune des zones.

Zones_ouvertes.shp	
ID	Identifiant unique de couronne
Proba	Rapport des aires des couples cercle/couronne

Tableau 7: Description de la couche "Zones_ouvertes.shp"

Zones_fermees.shp	
ID	Identifiant unique de couronne

Tableau 8: Description de la couche "Zones_fermees"

V.2 Reconstruction *billboards* des arbres

Nous l'avons vu en II.2.3, Rennes Métropole dispose d'ores et déjà d'un module interne de reconstruction des arbres de type *billboards* d'après une couche de points au format.SHP renseignant, la position, la hauteur et le diamètre de la couronne des arbres.

S'il existe aujourd'hui une grande variété d'algorithmes de détection et de segmentation de la végétation c'est qu'ils permettent facilement d'extraire les paramètres dendrométriques des arbres et notamment ceux relatifs à leur hauteur et diamètre. Ainsi, considérant les résultats obtenus suite à la phase de détection et de segmentation de la végétation, nous proposons dans cette nouvelle partie de générer les couches de données nécessaires au module de reconstruction interne du service.

En ne considérant qu'un seul diamètre de couronne, ce module vise à reconstruire les arbres selon une forme circulaire. Cependant, nous l'avons vu dans la partie précédente, si les couronnes individuelles tendent à respecter un tel critère (présentant d'ailleurs des formes plus ellipsoïdales que circulaires), ce n'est pas le cas de celles segmentées en zone de végétation dense, pouvant présenter des formes très complexes. Fort de ce constat, nous distinguerons deux scripts FME d'extraction de paramètres dendrométriques en fonction des zones ouvertes et fermées, respectivement intitulés "*Modelisation_ouverte.shp*" et "*Modelisation_fermee.shp*".

V.2.1 Extraction des paramètres dendrométriques

Avant de nous intéresser à l'extraction des paramètres, remarquons que le traitement des zones fermées nécessite une intervention préalable visant à récupérer l'ensemble des couronnes fusionnées en zones fermées. Cette étape consiste simplement à extraire de la couche de données "Couronnes_clean_all.shp" les couronnes intersectant les surfaces de la couche "Zones_fermees.shp" à l'aide d'une analyse spatiale (figure 52).

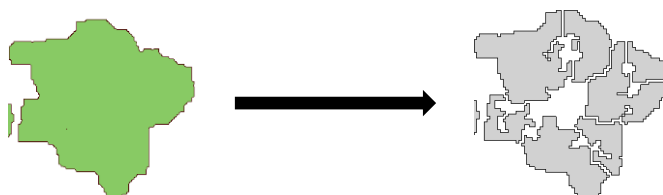


Figure 52: Récupération des couronnes en zones fermées

V.2.1.1 Hauteur des arbres

Dans un premier temps, pour les deux types de zones, les couples cimes / couronnes (ainsi que leurs attributs respectifs) sont réunis à l'aide d'une analyse spatiale (transformer « *SpatialFilter* »). Ainsi donc, chaque cime incluse dans l'emprise d'un houppier vient ajouter ses coordonnées X, Y et Z aux attributs de la couronne. La coordonnée Z renseigne alors le paramètre de hauteur de l'arbre détectée depuis le CHM.

V.2.1.2 Diamètres des couronnes

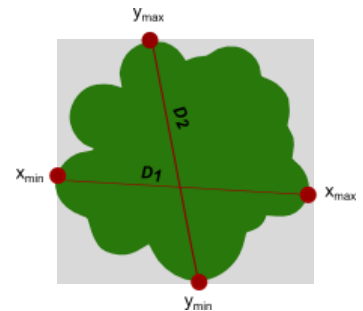
Dans le cas de zones ouvertes, les houppiers sont généralement considérés plus elliptiques que circulaires et peuvent donc être définis par deux diamètres. Bien que le module n'en considère qu'un seul, le service trouve intéressant d'avoir connaissance de ces deux valeurs parmi lesquelles il sera ensuite capable de distinguer la maximale.

À l'aide du transformateur "*BoundsExtractor*", nous calculons la plus petite boîte englobante des couronnes, isolons les extrema de chaque houppier et en déduisons alors deux valeurs de diamètres (D_1 et D_2), lesquels viennent ensuite s'ajouter à la liste des attributs.

$$D_1 = |x_{max} - x_{min}|$$

$$D_2 = |y_{max} - y_{min}|$$

Figure 53 : Extraction des diamètres de couronnes en zone ouverte



Dans le cas des zones fermées (regroupements d'arbres), la définition d'un diamètre est plus complexe. Cependant, considérant que la modélisation « en croisillons » n'est destinée qu'à un usage de représentation et non de simulations ce n'est pas tant la forme exacte des houppiers qui importe mais leur emprise globale.

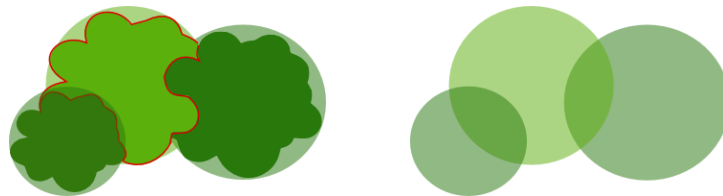


Figure 54 : Considération de l'emprise des houppiers en zone fermées

Nous approximations la forme de chaque houppier par son plus petit cercle englobant (transformer "*MinimumSpanningCircleReplacer*") et en déduisons un diamètre unique de couronne. Nous remarquons que cette méthode est également applicable aux zones ouvertes.

V.2.2 Positionnement des arbres

A ce stade du processus, nous disposons de couches shapefiles représentant l'emprise des couronnes et renseignant des attributs de hauteur (coordonnée Z de la cime détectée) et de diamètre de couronne. Mais, le module considérant des géométries de type "point", nous devons remplacer l'emprise des couronnes par des points indiquant la position de l'arbre (et

donc le centre du tronc). Deux hypothèses peuvent alors être considérées : La position de l'arbre est définie par celle de la cime détectée ou bien par le centroïde de la couronne segmentée.

Pour tenter de définir laquelle des deux options apparaît être la meilleure, nous considérons la position définie par le service DJB¹⁰ comme référence et observons les écarts pouvant exister avec les cimes détectées d'une part et, les centroïdes de couronnes d'autre part.

Moyenne des écarts avec la couche de référence (DJB) (m)		
	Cimes	Centroïdes
Zones ouvertes	1.7	1.8
Zones fermées	4.4	3.9

Pour les zones ouvertes, la considération de l'une ou l'autre des deux options n'a pas d'impact majeur. Cette remarque permet d'ailleurs d'apprécier la cohérence de la détection des cimes en zones ouvertes, les écarts pouvant être dus à une courbure du tronc. En zones fermées par contre, les écarts sont plus marqués et l'on observe une position améliorée de 50 cm par considération des centroïdes. Forts de ce constat, nous décidons d'associer le centre des troncs à la position du centroïde de leur houppier. Cette solution permet également de prévenir la présence éventuelle de troncs courbés pouvant générer des écarts entre les cimes et le centre du tronc.

De plus, un faible écart entre la cime et le tronc aura tendance à traduire une forte cohérence entre la position de la cime détectée et la segmentation de la couronne lui étant associée. A l'inverse, un écart important pourra mettre en évidence des troncs très courbés ou des erreurs (de détection et de segmentation). Nous décidons alors de renseigner la valeur (F) suivante comme attribut de « Fiabilité »

$$F = 100 - (||\text{position cime} - \text{position centroïde}||)$$

V.2.3 Résultats

Il résulte de ces scripts deux couches de points shapefile, « Modelisation_ouverte » et « Modelisation_fermée », exploitables par le module de reconstruction en interne.

Modelisation_ouverte.shp	
ID	Identifiant unique de couronne
Proba	Rapport des aires des couples cercle/couronne
Hauteur	Coordonnée Z de la cime détectée
Fiabilité	Distance entre la cime et le centroïde de son houppier

Tableau 9: Présentation de la couche "Modelisation_ouverte.shp"

Modélisation_fermée.shp	
ID	Identifiant unique de couronne
Hauteur	Coordonnée Z de la cime détectée
Fiabilité	Distance entre la cime et le centroïde de son houppier

Tableau 10: Présentation de la couche "Modélisation_fermée.shp"



Figure 55: Résultat de la modélisation "en croisillon"

V.3 Reconstruction 3D maillée des arbres

Nous venons de voir une méthode d'extraction de paramètres dendrométriques utiles à une modélisation « en croisillon » des arbres. Très légère et simple à produire, une représentation de ce type n'en reste pas moins minimaliste et peu représentative de la réalité. Aujourd'hui, le service SIG souhaite développer un nouveau processus de reconstruction 3D de sa végétation pour produire des objets maillés sur la base de données LiDAR aériennes. Suite à notre État de l'Art (IV), relatif aux techniques actuelles de reconstruction 3D des arbres d'après un ensemble de données discrètes à haute résolution, l'utilisation de formes alpha est apparue comme une solution répondant à cette attente. En effet, d'un niveau de détail paramétrable, les formes alpha présentent un attrait tout particulier pour la production d'objets intégrables à une maquette 3D.

Dans cette troisième partie, nous proposons un nouveau processus de modélisation 3D des arbres s'inspirant d'un algorithme de type « α -shape ». Similairement à l'approche considérée par *Rahman et al [18]* nous définissons un arbre comme un objet composé de deux entités distinctes : un houppier et un tronc lui étant associé. Cette approche facilite la gestion ultérieure de leurs texturations respectives.

V.3.1 Un processus développé en collaboration avec *RhinoTerrain*

En tant que prestataire du service SIG, la société *RhinoTerrain* intervient régulièrement pour réaliser des missions de retranscription des objets créés par le service SIG dans un format CityGML, standard d'échange propre au partage de la donnée géographique et développé par l'OGC. Dans le cadre de notre projet, elle est mandatée pour assurer la transcription des objets "arbres" générés par notre modèle.

En termes de végétation, le CityGML distingue les arbres isolés (classe "*SolitaryVegetationObject*") des zones de végétation dense (classe "*PlantCover*"). C'est d'ailleurs principalement pour cette raison que l'ensemble des traitements antérieurs distinguaient de manière similaire les zones dites "ouvertes" et "fermées".

Après entretien avec la société *RhinoTerrain*, il est convenu que les objets (houppiers et troncs) devront être codés dans un format .OBJ et répartis dans deux dossiers distincts que nous intitulerons "Ouvert" et "Ferme". Pour faciliter l'identification et le regroupement des couples tronc / houppier, la nomenclature suivante devra être respectée :

Élément	Nomenclature
Houppier	Arbre3D_ID ¹¹ .obj
Tronc	Tronc3D_ID.obj

Tableau 11: Nomenclature des objets arbres

Le processus de modélisation 3D des arbres fait l'objet d'un script FME intitulé "Assemblage.fmw" et intégré à la suite de l'algorithme de détection et de segmentation de la

¹¹ L'Identifiant fait référence à l'ID unique attribué aux couronnes lors de la phase de segmentation

végétation que nous avons adapté. Il comprend deux étapes principales, l'une de reconstruction des houppiers et l'autre des troncs.

V.3.2 Une reconstruction des houppiers de type « Alpha Shape »

V.3.2.1 Préparation des données : création de clusters

Rappelons à ce stade que l'intérêt majeur de l'adaptation de l'algorithme de détection et de segmentation de la végétation était de nous permettre de réaliser le "*clustering*" du nuage de points que nous appellerons "végétation normalisé". Ce nuage est issu du MHC est enregistré dans un format. xyz

Considérant les couches de données shapefile "Zones_ouvertes" et "Zones_fermées" générées en V.1.3.2, le découpage du nuage est simplement réalisé à l'aide d'une technique de découpe (transformer "*Clipper*") (figure 56). La partition des houppiers en zones dites "ouvertes" et "fermées" affecte chaque sous-ensemble de points à l'une des deux classes de végétation du CityGML.

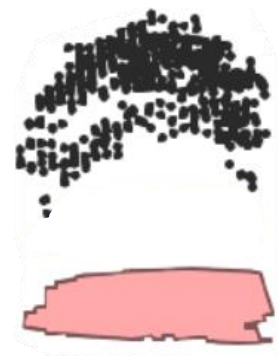


Figure 56: clustering du nuage de points

V.3.2.2 De la 2.5D vers la 3D

Nous l'avons vu en IV.1.3.3, une reconstruction par « formes *alpha* » est une généralisation de l'enveloppe convexe d'un ensemble de points. Dans le cas d'un levé LiDAR aérien, selon la hauteur de vol et le nombre de retours, l'affaiblissement du faisceau LASER entraîne une forte diminution, voir une absence totale, de points sur la partie basse des houppiers. Le nombre points chute encore plus dans des secteurs de végétation dense où chaque arbre constitue un masque pour son voisin. On ne peut donc reconstruire les houppiers qu'en 2.5D (figure 57).

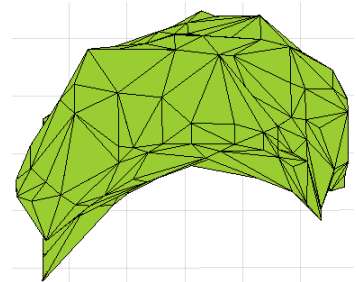


Figure 57: Houppier en 2.5D

Afin de modéliser des houppiers en véritable 3D, nous proposons une méthode visant à extrapoler des points sur leur base.

V.3.2.3 Reconstruction de la base des houppiers

La méthode d'extrapolation de points proposée s'appuie à la fois sur les sous-ensembles issus du clustering du nuage « végétation normalisé » et sur les couches shapefile « Zones_ouvertes.shp » et « Zones_fermées.shp ». Le tableau suivant illustre les étapes principales :

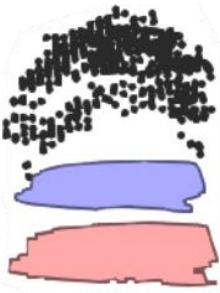
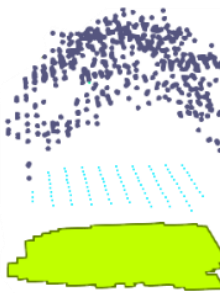

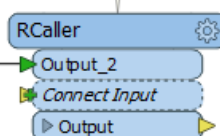
Illustration	Explication	Transformer
	<ul style="list-style-type: none"> Extraction de la valeur de la coordonnée "Z_{min}" du point le plus bas constituant le houppier Duplication de la couronne correspondante Affectation d'une hauteur correspondant à la valeur Z_{min} extraite à la couronne dupliquée 	<i>BoundsExtractor</i> <i>Copier</i> <i>3DForcer</i>
	<ul style="list-style-type: none"> Transformation du duplicata surélevé en nuage de points 	<i>PointCloudCombiner</i>
	<ul style="list-style-type: none"> Regroupement des points constituant chaque houppier d'arbre (base + couronne) Écriture des nouveaux sous-ensembles selon la nomenclature suivante : ID¹².xyz 	<i>PointCloudCombiner</i> <i>FeatureWriter</i>
	<ul style="list-style-type: none"> Calcul des formes alpha avec R¹³. Export des volumes dans un format objet selon la nomenclature établie avec RhinoTerrain : Arbre3D_ID.obj 	<i>RCaller</i>

Tableau 12: Méthode de reconstruction des 3D des houppiers

V.3.2.4 Une implémentation sous R, langage interopérable avec FME

De nombreux algorithmes "Alpha Shape" existent (Python, R, Matlab, C++ avec CGAL et la PCL ...). Dans un premier temps, par souci de cohérence, nous envisageons une implémentation sous Python ou MATLAB, langages de programmation et logiciel utilisés jusqu'alors. Bien que dans les



Figure 58: Logo R

¹² ID correspond à l'identifiant unique de la couronne segmentée. L'écriture de fichiers de points intermédiaires est indispensable préalablement à l'exécution du script R

¹³ Script détaillé dans la suite

deux cas les algorithmes¹⁴ fonctionnent, nous rencontrons néanmoins des problèmes d'export.

En langage R le package "alphashape3d" et sa fonction "ashape3d" sont une solution intéressante car ont fait l'objet d'une étude menée par *Lafarge et al. [11]* portant sur le fonctionnement de l'algorithme et l'export du résultat.

De plus, interopérable avec FME par le biais du transformeur « RCaller », R s'impose comme une solution n'entravant pas l'automatisation de l'algorithme. Une aide relative au paramétrage nécessaire à l'exécution de scripts R avec FME est disponible en *annexe 1*.

V.3.2.5 Reconstruction 3D : la fonction « ashape3d »

La fonction « ashape3d », permet la reconstruction 3D d'un objet selon des formes *alpha*. Suivant un protocole détaillé par *Lafarge et al. [11]*, considérant les sous-ensembles de points issus de la succession des phases de clustering et d'extrapolation de points sur la base des houppiers, nous développons un script R reconstruisant un maillage 3D d'après les sous-ensembles de points obtenus. Intégré au sein du transformeur « RCaller », ce script fait partie de la chaîne de traitement « Assemblage.fmw » et est lancé automatiquement à la création des sous-ensembles de points propre à chaque arbres (voir tableau 12).

Afin de rendre cette chaîne de traitement automatique, les chemins d'accès aux différentes variables sont renseignés de façon dynamique. L'utilisateur choisit alpha selon qu'il souhaite une reconstruction d'arbres isolés ou regroupés. Rappelons qu'en secteur de végétation dense, l'absence de points sur la partie basse des houppiers peut entraîner, avec un critère alpha trop fin, la création de volumes « troués ». Néanmoins comme le rappelle *Lafarge et al. [11]*, pour un alpha infiniment grand, la forme reconstruite équivaut à l'enveloppe convexe du nuage de points considéré (figure 59). Ainsi, optant pour un niveau de détail moins élevé en zone de végétation dense, on limite les risques de générer des volumes troués.



Figure 59: Reconstruction alpha-shape pour des critères alpha croissants de gauche à droite

Développé sous forme d'une boucle, le programme de reconstruction parcourt les fichiers de points au format .xyz et enregistre les objets 3D construits au format .OBJ en respectant la nomenclature Arbre_3D.OBJ définie communément avec la société *RhinoTerrain*.

¹⁴ Sous python, se référer à la bibliothèque graphique interactive « plotly.py » développée par l'entreprise Plotly [\[REF\]](#) Sous MATLAB, existence de plusieurs algorithmes de type « alpha shape » dont « alphaShape » [\[https://fr.mathworks.com/help/matlab/ref/alphashape.html\]](https://fr.mathworks.com/help/matlab/ref/alphashape.html) et « alphaHull » [\[REF\]](#)

Il est intéressant de remarquer que la fonction « ashape3d » ne peut fonctionner qu’avec des nuages de points exprimés en coordonnées relatives. On translate donc les coordonnées vers l’origine, puis on les rétablit après traitement.

L’algorithme est disponible en *annexe 2*

V.3.2.6 Export des houpriers : Structure du format .OBJ

Lafarge et al.[11] détaille une méthode d’export des objets de type « ashape3d » dans des fichiers au format "Object File Format" (fichier .OFF). Bien que pris en charge par la plupart des logiciels de manipulation 3D, dont Rhinocéros et CloudCompare, nous avons convenu avec la société *RhinoTerrain* d’un export au format .OBJ. Ces formats sont très similaires (figure 60).

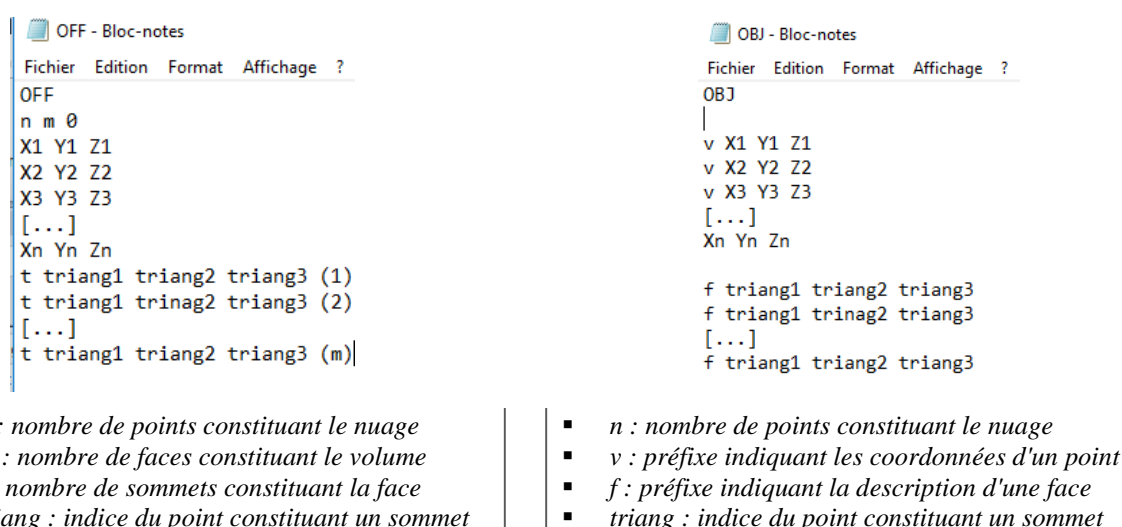


Figure 60: Comparaison des structures des formats .OFF et .OBJ

Nous adaptons alors la méthode proposée par *Lafarge et al. [11]* et l’ajoutons à notre script R. Cette portion de l’algorithme est également disponible en *annexe 2*.

Les fichiers de points d’entrée respectant la nomenclature « ID.xyz », le programme repère le numéro d’identifiant propre à chaque objet qu’il renseigne ensuite dans le nom du fichier .OBJ selon la nomenclature commune définie avec la société *RhinoTerrain* : « Arbre_3D_ID.obj »

V.3.2.7 Résultats

Finalement, la chaîne de traitement « Assemblage.fmw » est entièrement automatisée et permet la reconstruction fine de houpriers 3D selon des formes alpha avec des niveaux de détails paramétrable en zones ouvertes et fermées. Les volumes sont stockés dans un fichier .OBJ attendu par la société *RhinoTerrain*.

La figure suivante présente les résultats obtenus ainsi qu’un comparatif en fonction de la considération ou non d’une extrapolation de points sur la base des houpriers.

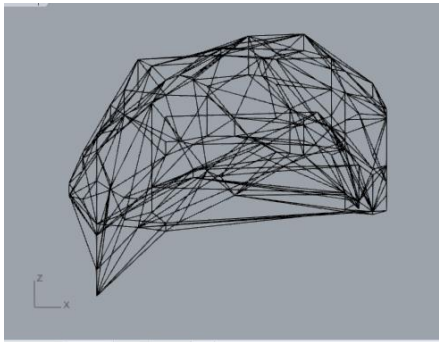
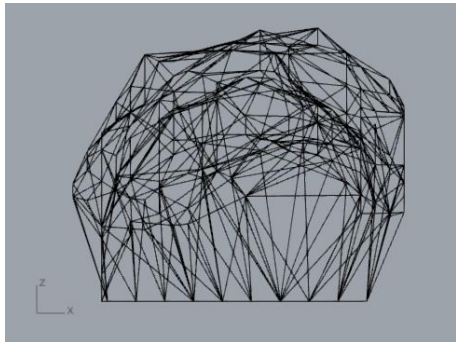
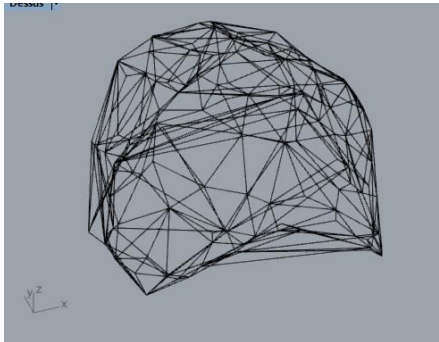
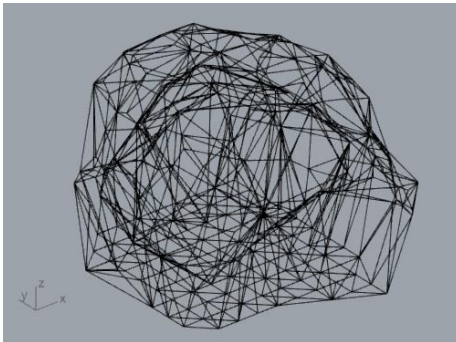
	Sans reconstruction de la base	Avec reconstruction de la base
Vue de droite		
Vue en perspective		

Tableau 13: De la 2.5D à la 3D, reconstruction des houppiers

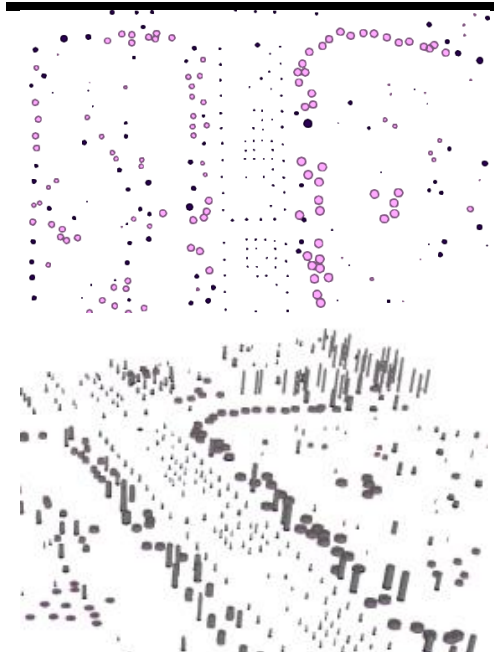
Parallèlement à la reconstruction des houppiers, la chaîne de traitement "Assemblage.fmw" génère également les troncs, identifiés par l'attribut "ID" de la couronne segmentée.

V.3.3 Reconstruction des troncs par extrusion

Une approche plus simpliste consiste à modéliser les troncs par des cylindres dont la hauteur et le diamètre sont dépendants de la hauteur des arbres. Rappelons que nous avons défini le centre des troncs comme correspondant aux centroïdes des couronnes auxquelles ils sont associés, centroïdes constituant les couches "Modelisation_ouvertes" et "Modelisation_fermées" générées en V.2.2.

Les paramètres dendrométriques sont déduits des extrema (X_{\min} , X_{\max} , Y_{\min} , Y_{\max} , Z_{\min} , Z_{\max}) appartenant aux différents sous-ensembles de points propres aux houppiers extraits au cours de la première phase du traitement (Transformer "BoundsExtractor", V.3.2.4).

Nous ajoutons alors une chaîne de traitements au script "Assemblage.fmw", dédiée à la modélisation des troncs utilisant ces deux couches. Les phases principales sont listées dans le tableau suivant :

Illustration	Explication	Transformer
	<ul style="list-style-type: none"> Remplacement de chaque centre de tronc (issus des couches « Modelisation_ouverte » et « Modelisation_fermee » par un cercle dont le diamètre est calculé en fonction de la hauteur de l'arbre 	<i>GeographicBufferer</i>
	<ul style="list-style-type: none"> Extrusion de chaque cercle selon une hauteur définie en fonction de la coordonnée Z_{min} du point le plus bas 	<i>Extruder</i>
	<ul style="list-style-type: none"> Export des objets au format .obj selon la nomenclature suivante : Tronc_ID.obj 	

V.3.3.1 Approximation du diamètre et de la hauteur des troncs

Afin de produire un rendu le plus réaliste possible, nous générons des troncs possédant des paramètres de diamètre et de hauteur définis en fonction de ceux du houppier auquel ils se rattachent.

Le diamètre des troncs est défini selon la relation suivante :

$$\phi_{tronc} = |Z_{max} - Z_{min}|$$

La hauteur du tronc d'après la relation suivante :

$$Hauteur_{tronc} = Z_{min} + 1.0$$

(Augmenter la hauteur du tronc de 1.0m permet de s'assurer de son raccord avec le houppier correspondant).

Ces relations sont empiriques et ne produisent donc pas des résultats très réalistes. Elles permettent toutefois dans un premier temps de fournir un rendu plus réaliste et dynamique.

V.3.4 Résultat

Finalement, après assemblage puis texturation simpliste des couples d'objets troncs / houppiers, nous obtenons des objets "Arbres" parmi lesquels nous distinguons les éléments isolés (figure 62), associant un tronc à un houppier unique, des regroupements d'arbres

(figure 61), associant plusieurs troncs à un houppier unique. La société *RhinoTerrain* est aujourd'hui en charge de la retranscription de ces objets dans un format CityGML, en vue de leur intégration dans la maquette 3D.



Figure 62: Objet "arbre isolé"



Figure 61: Objet "regroupement d'arbres"

Afin d'améliorer l'interactivité du processus une interface a été développée sous Tkinter (Python), intégrant l'ensemble des traitements de détection, de segmentation et de modélisation 3D de la végétation. Elle présente un intérêt majeur car elle permet à l'utilisateur de renseigner un système de projection quelconque en entrée, lequel est transformé en variable puis géré par la librairie GDAL/OGR pour produire des résultats systématiquement exprimés en RFG93-CC48. Plus de détail concernant cette interface est disponible en *annexe 3*

Une société privée s'est montrée très intéressée par le projet et serait susceptible de développer un partenariat avec le service pour se charger de sa mise en production. En attendant, la figure ci-dessous propose un aperçut du résultat obtenu sur une portion de l'avenue Henri Fréville. On distingue les arbres individuels (vert clair) des regroupements d'arbres (vert foncé) et on peut également noter un rendu intéressant des arbres d'alignement.

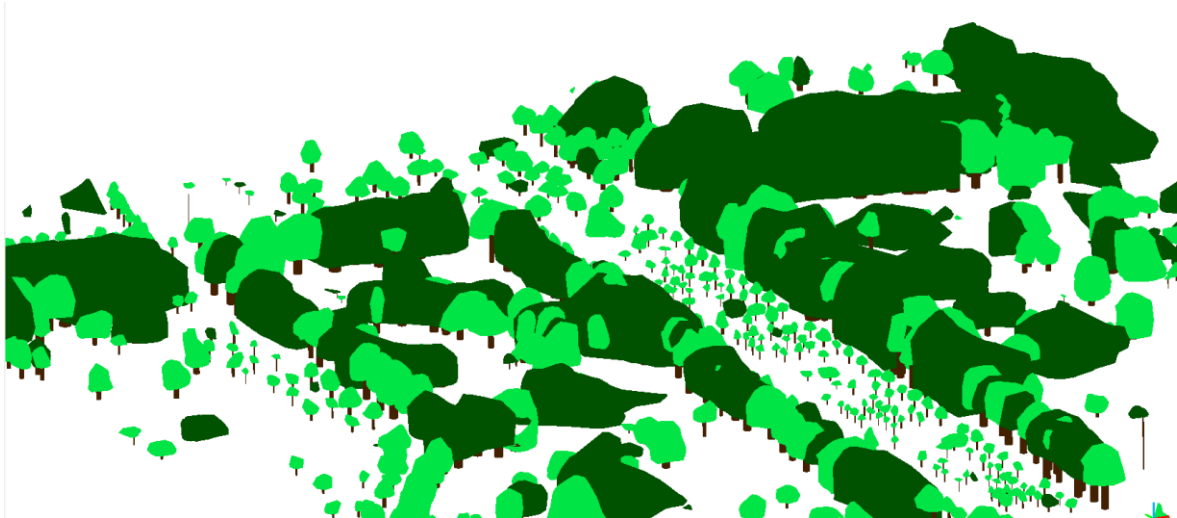


Figure 63: Résultat final de la modélisation 3D de la végétation sur le territoire de Rennes Métropole

VI Discussions

VI.1 Efficacité du protocole de prétraitements

Nous l'avons vu en II.2.2.2, un prestataire privé est actuellement chargé de la classification fine de la végétation dans notre nuage LiDAR. Ce dernier nous ayant fourni un échantillon test, il est intéressant d'effectuer une comparaison entre la classification fine qu'il propose et le nuage « nettoyé » que nous obtenons suite à notre processus de prétraitements.

Dans un premier temps, nous comparons la classification fine réalisée par le prestataire à la pré-classification automatique réalisée par l'IGN. Nous observons alors une nette amélioration concernant la confusion qui existait entre les classes « bâti » et « végétation » (spécifications de l'ASPRS), comme l'illustre la figure suivante.



En rouge la classe végétation de l'IGN, en vert celle de GEOFIT. Zoom croissant sur un quartier résidentiel de la gauche vers la droite. Les toitures disparaissent dans la classe végétation de l'IGN
Figure 64: Comparaison des classifications LiDAR réalisées par l'IGN et GEOFIT

Dans un second temps nous comparons la classification du prestataire avec le nuage "végétation" obtenu d'après notre protocole de prétraitements. Si la classification fine démontre une segmentation efficace de la classe « bâti » (totalement absente de la classe « végétation »), elle laisse par contre apparaître des confusions entraînant la classification d'une partie ou de la totalité de certains houppiers à proximité de bâtiment en tant qu'élément de « bâti ». La figure suivante présente un comparatif entre l'échantillon fourni par le prestataire et notre nuage « végétation » sur des secteurs posant problème.

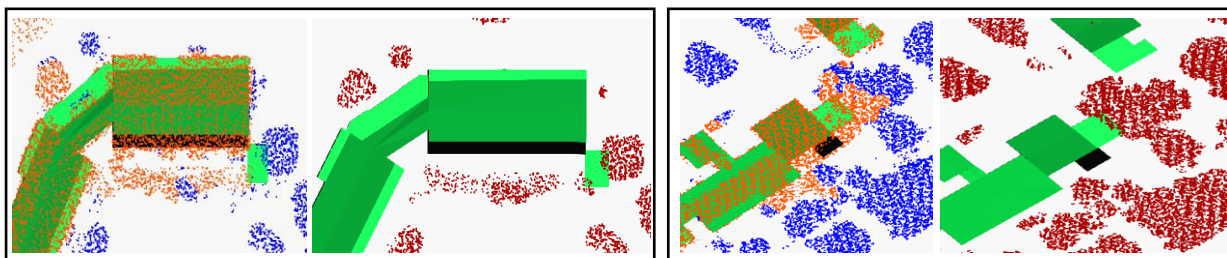


Figure 65: Comparatif entre la classification du prestataire (à gauche) et notre nuage « végétation » (à droite) sur deux secteurs. En rouge le nuage « végétation » ; en orange : la classe « bâti » du prestataire ; en bleu : la classe « végétation » du prestataire

Nous concluons que dans l'état actuel, nous proposons une extraction plus fine de la végétation. Une demande d'ajustement sera faite par la suite au prestataire. En effet, il serait intéressant de pouvoir disposer d'un nuage finement classifié et de s'affranchir de la phase de prétraitements qui, bien qu'elle soit efficace, n'est pas automatisée.

VI.2 Limites de segmentation et de partition des arbres

Bien que traditionnelle, la segmentation des arbres selon une approche dite « raster » *Zhen Zhen et al. [25]* présente des limites. Parmi les plus récurrentes *Weinacker et al. [23]* : une mauvaise délimitation des couronnes, une sous-estimation de la hauteur des arbres ou encore la modification du positionnement originel des points.

Concernant la délimitation des couronnes dans un premier temps, les incohérences géométriques ont très vite été remarquées et ont d'ailleurs entraîné la distinction des zones dites "ouvertes" et "fermées". Nous avons adapté le processus de « partition » en concentrant notre attention sur la vraisemblance géométrique des houppiers plutôt que sur la distance les séparant de leurs voisins (C. Doceul). C'est donc sans surprise que nous constatons une nette amélioration des éléments classés en zones ouvertes due à une sélection plus stricte d'un point de vue géométrique. Néanmoins, quelques faiblesses sont à remarquer.

Tous d'abord la gestion des houppiers que nous qualifions « éclatés ». En effet, si plusieurs approches de réunion des fragments ont été abordées (buffers, analyses spatiales, ...) aucune n'a permis d'atteindre ce but sans impacter les autres houppiers (*figure 67*).

Ensuite, bien qu'elle permette une partition plus cohérente des arbres, la méthode proposée ne propose pas encore une solution optimale. En effet, nous constatons la présence d'arbres isolés en zones fermées (*figure 66*). Cette confusion est liée à un seuil de partition (V.3.1.1) trop stricte mais qui, s'il est abaissé, ne permet plus de distinguer efficacement les arbres isolés des regroupements. Cette faiblesse est plus dérangeante du point de vue de la transcription des objets au format CityGML (confusion des classes « solitaryObject » et « PlantCover »).

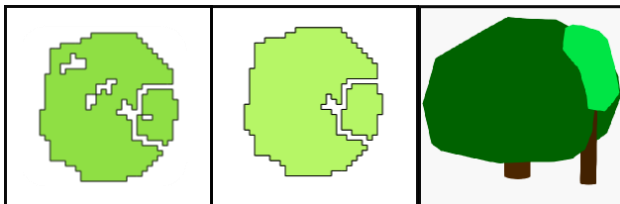


Figure 67 : Des houppiers "éclatés"

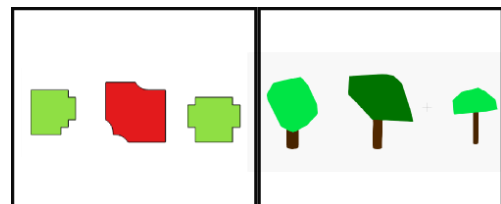


Figure 66: Confusion de zone ouverte et fermée

La principale différence entre les arbres isolés et les regroupements, d'un point de vue structurel, est le nombre de troncs associé à un houppier. Ainsi, une solution pourrait consister à analyser le nombre de tronc associés aux houppiers en zones fermées pour identifier les éléments isolés.

Concernant la sous-estimation de la hauteur des arbres ainsi que la modification de la position originelle des cimes, il serait effectivement intéressant de réaliser des analyses comparatives complémentaires. Il serait notamment intéressant d'étudier l'impact lié à la normalisation du nuage car nous considérons le nuage « végétation normalisé » pour la reconstruction des arbres.

VI.3 Vers un clustering "orienté nuage" : K-moyennes et voxels

Pour parer aux faiblesses liées à la rasterisation, on observe depuis 2006 un intérêt croissant pour les approches de segmentation dites « orienté nuage » Zhen Zhen et al. [1] et visant donc à segmenter les arbres directement au sein du nuage LiDAR. Deux méthodes sont particulièrement utilisées : Le clustering par k-moyenne ou la segmentation individuelle par "voxels".

La technique de *clustering* par k-moyenne consiste à regrouper des points dans différents *clusters* selon un critère de distance (figure 69). Associant les maxima locaux d'un nuage de points (et donc les cimes) à des nœuds, Morsdorf et al. 2004 [14] parviennent à déduire les paramètres de position, hauteur et diamètres des couronnes des arbres.

Dans un second temps l'utilisation des grilles voxels (figure 68), déjà très présente en imagerie médicale où elle a été introduite en 1973 par Godfrey Hounsfield, fait également de plus en plus parler d'elle dans le domaine de la segmentation de données LiDAR. Analogie 3D du pixel, le voxel vise à diviser un espace 3D en cellules 3D, typiquement des cubes. Ainsi, en matière d'ITCD, des travaux plus récents s'intéressent à cette nouvelle technique (Douillard et al [5], Gebhardt et al [9]).

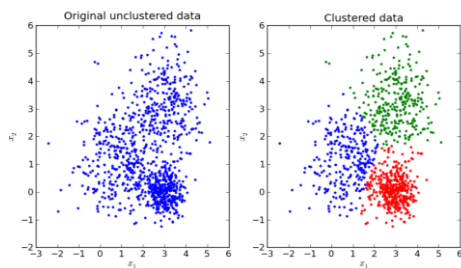


Figure 69: Clustering par K-moyenne

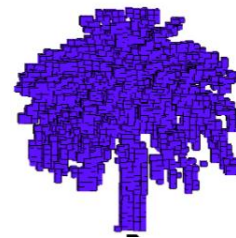


Figure 68: Clustering par voxels
(Crédit : B. Douillard)

Limitant les interventions pouvant affecter la précision de la donnée initiale, il pourrait être intéressant de développer une méthode de segmentation du nuage LiDAR selon une approche "orientée nuage" préalablement à la phase de reconstruction 3D.

VI.4 Reconstruction 3D des houppiers et écarts au nuage de points

Une fois les houppiers reconstruits en volume, il est naturel d'analyser les écarts existants entre ces objets et le nuage de points (végétation normalisée) dont ils sont issus.

Après avoir importé chacune de ces deux couches de données sous CloudCompare, nous analysons les écarts existants entre le nuage de point et les houppiers (à l'exclusion des troncs) à l'aide du plugin « Cloud/Mesh » (V.1.1.1).

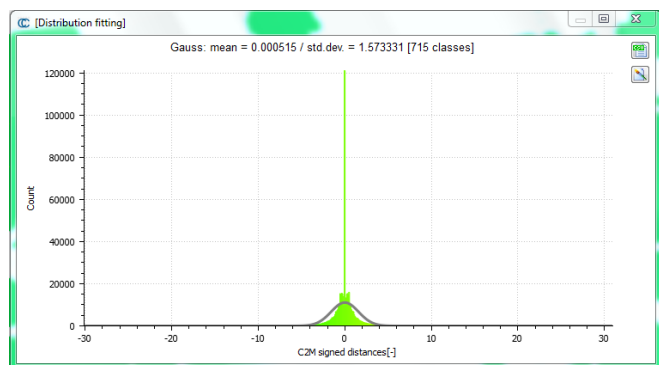


Figure 70: Répartition des écarts entre le nuage de points et la modélisation 3D

La figure ci-dessus traduit une répartition gaussienne des écarts selon une moyenne $\mu = 0.000m$ et un écart-type $\sigma = 1.573m$. La valeur de la moyenne est à la fois surprenante et rassurante. En effet, elle traduit un drapage quasi-parfait de la reconstruction par formes alpha. La valeur de l'écart type est liée à la considération de points résiduels exclus de modélisation selon des critères de surface (exclusion des surfaces inférieures à $10m^2$) ou de géométrie (exclusion des formes non circulaires)

VI.5 Reconstruction 3D des houppiers : Faces non soudées et non orientées

Une analyse des objets « houppier », bien que déjà très prometteuse, met en évidence deux défauts majeurs de reconstruction.

D'abord, l'absence de normales associées aux faces (figure 71(a)). Néanmoins de manière générale les logiciels de visualisation et de manipulation d'objets 3D disposent d'outils propres à la définition des normales. C'est le cas Rhinocéros et son outil « Unifier les normales ». Après entretien avec la société *RhinoTerrain* cette défaillance ne représente pas un obstacle à la transcription des arbres au format CityGML. Par ailleurs le package « alphashape3d », propose une fonction complémentaire permettant le calcul des normales que nous analysons actuellement.

Seconde défaillance, les faces ne sont pas soudées les unes aux autres (figure 71(b)). Le risque est alors d'entraîner des déformations au cours des manipulations. Néanmoins, une fois encore, cette défaillance ne semble pas poser un obstacle majeur pour la société *RhinoTerrain*.

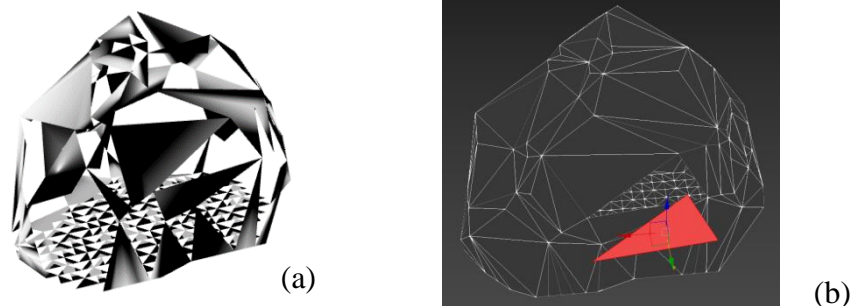


Figure 71: Défaillances de reconstruction des houppiers

VI.6 La corrélation dense, une alternative envisageable ?

La corrélation dense est une technique visant à produire des nuages de points 3D à partir de prises de vues stéréoscopiques numériques très redondantes.

Source d'acquisition de données à très haute résolution, la photogrammétrie pourrait peut-être fournir une alternative envisageable à la production de nuages de points LiDAR, plus onéreuse.

Le service SIG de Rennes Métropole réalisant, en partenariat avec l'IGN, des prises de vues obliques à très haute résolution sur le territoire, il serait intéressant de réaliser une étude de faisabilité de l'utilisation de nuages de points construits par corrélation dense sur la base de ces clichés.

Conclusion

L'objectif de ce travail était de mettre en place un processus de modélisation 3D de la végétation sur le territoire de Rennes Métropole à partir d'un nuage de points LiDAR aérien.

Aspirant à un futur plus fonctionnel, économe et durable, les villes réfléchissent à leur développement en s'appuyant sur la maquette 3D. Jumeau numérique du territoire qu'elle représente la maquette, parmi ces nombreux usages, est un support d'exploration et d'analyse de ses opportunités de développement et se doit donc de lui être le plus fidèle possible. La modélisation 3D du bâti est une mécanique largement maîtrisée mais, ce n'est pas le cas pour la végétation, pourtant deuxième donnée volumique la plus importante à l'échelle de la ville.

Les formes « alpha », par leur souplesse de reconstruction et leur niveau de détail paramétrable, constituent une solution intéressante pour la reconstruction 3D de la végétation. Très fidèles à la réalité, les objets modélisés présentent de bonnes dispositions pour une intégration dans la maquette 3D de la métropole. Néanmoins, cette étape reste encore une marche supplémentaire à franchir.

Aujourd'hui, le service SIG de Rennes Métropole dispose d'un outil semi-automatique de détection, de segmentation et de modélisation de la végétation présente sur son territoire, fruit de deux travaux successifs de recherche menés par des étudiants de l'ESGT.

Appliqué jusqu'à présent sur des secteurs d'études de taille restreinte, une mise en production est maintenant nécessaire afin de pouvoir appliquer le processus à l'ensemble de la Métropole. Sur ce sujet, un prestataire privé s'est montré fortement intéressé par les résultats obtenus et un partenariat pourrait être envisagé avec le service SIG pour franchir cette nouvelle étape vers une intégration dans la maquette 3D.

D'un point de vu plus personnel, ce travail marque l'aboutissement de ma formation d'ingénieur Géomètre et Topographe au sein de l'ESGT.

Cette expérience se sera révélée gratifiante tant en termes professionnel qu'humain. En effet, elle m'aura d'abord permis de prendre conscience du chemin parcouru, des connaissances acquises et de mieux appréhender l'avenir. J'aurais également eu la chance de faire de belles rencontres dans un service accueillant et chaleureux.

Bibliographie

- [1] Au et al. "Skeleton extraction by mesh contraction," 2008, ACM Trans. on Graph, volume 27, no. 3, pp. 44:1–44:10.
- [2] Bournez E, Landes T, Saudreau M et al. *From TLS Point Clouds to 3D models of trees : A comparison of existing algorithms for 3D tree reconstruction*. Remote Sensing and Spatial Information Science, 2017, Volume XLII-2/W
- [3] Cao J, Olson M, Zhang H et al. *Point cloud skeletons via Laplacian-based contraction*. In: Proceedings of the IEEE International Conference on Shape Modeling and Applications, Aix en Provence, France, 2010, pp.187-197.
- [4] Doceul C. *Caractérisation de la végétation de Rennes Métropole par relevé LiDAR en vue de sa modélisation*, 2017
- [5] Douillard B, Underwood J, Kuntz N et al. *On the segmentation of 3D LiDAR point clouds*. Conference Paper, In : Proceedings – IEEE International Conference of Robotics and Automation, June 2011
- [6] Edelsbrunner H, Kirkpatrick DG, Seidel R. *On the shape of a Set of Points in the Plane*. *IEEE Transaction on Information Theory*. 1983, 29(4), pp.551-559
- [7] Edelsbrunner and Mücke. *Three-Dimensional Alpha Shapes*. ACM Transactions on Graphics. 1994, 13(1), pp.43-72
- [8] ESRI. *USDA Forest Service FUSION Offers Powerful Lidar Tools*, ESRI NEWS, Spring 2013
- [9] Gebhardt S, Payzer E, Salemann L. *Polygons, Point-Clouds, and Voxels, a Comparison of High-Fidelity Terrain Representations*
- [10] Gupta S, Weinacker H, Koch B. *Comparative analysis of clustering-based approaches for 3-D single tree detection using airborne fullwave LiDAR data*. Remote Sens. 2010, vol.2, pp.968–989
- [11] Lafarge T, Pateiro-Lopez B, Possolo A et al. *Implementation of a Polyhedral Approximation to a 3D Set of Points Using the α -Shape*, Journal of Statistical Software, January 2014, volume 56, issue 4.
- [12] Landes T, Najjar G, Saudreau M et al. *Modélisation 3D d'arbre pour comprendre le climat urbain, un projet multidisciplinaire ambitieux*. Revue XYZ, 2014, 141(4^{ème} trimestre), pp.61-68
- [13] *Les modèles numériques de canopée (MNC) de Montréal*. GÉOMATIQUE (revue de l'Ordre des Arpenteurs-Géomètres du Québec), Printemps 2016, Volume 43 – Numéro 1, pp.16-19

- [14] Morsdorf F, Meiera E, Kötza B et al. LiDAR-based geometric reconstruction of boreal type forest stands at single tree level for forest and wildland fire management. *Remote Sens. Environ.* 2004, 92, pp.353-362
- [15] #PacteEtatMétropole : l'innovation au service des territoires, [<https://www.gouvernement.fr/pacteetatmetropoles-l-innovation-au-service-des-territoires>], 27 Janvier 2017
- [16] Pateiro-Lopez B, Possolo A. *alphahull: Generalization of the Convex Hull of a Sample of Points in the Plane*. R package version 0.2-1, 2010, [<http://CRAN.R-project.org/package=alphahull>].
- [17] Pfeifer N, Gorte B, Winterhalder D. *Automatic Reconstruction of Single Trees from Terrestrial Laser Scanner Data*. In Proceedings of 20th ISPRS Congress, 2004, pp 118-119.
- [18] Rahman M-Z-A, Majid Z, Rasib A-W et al. *Individual tree measurement in tropical environment using terrestrial laser scanning*. Jurnal Teknologi, 2015, 73(5), pp. 127–133.
- [19] Reitberger J, Stilla U, Krzystek P. *3D segmentation of single trees exploiting full waveform LiDAR data*, ISPRS Journal of Photogrammetry and Remote Sensing , 2009, vol.64, pp 561-574
- [20] Valente. *Niveau de précision d'une maquette urbaine*. [<https://bimbt.com/conception/niveaux-de-precision-dune-maquette-numerique-urbaine>]. (publié le 6 Janvier 2018)
- [21] Valente. *Les intérêts d'une maquette urbaine*. [<https://bimbt.com/conception/les-interets-dune-maquette-numerique-urbaine>]. (Publié le 24 Décembre 2017)
- [22] Van Leeuwen et al. *Canopy surface reconstruction from a LiDAR point cloud using Hough transform*. Remote Sensing and Spatial Information Sciences. Lett. 2010, 1, pp.125–132
- [23] Weinacker H, Koch B, Heyder U et al. *Development of filtering, segmentation and modelling modules for LiDAR and Multispectral data as a fundament of an automatic forest inventory system*. Remote Sens., Volume XXXVI, pp.50-55
- [24] Wolisz H, Boese L, Harb H et al. *District Information Modeling as a foundation for simulation and evaluation of smart city approaches*. Conference: Building Simulation and Optimization Conference, At London, UK, 2014
- [25] Zhen Zhen, Quackenbush L-J, Zhang L. *Trends in Automatic Individual Tree Crown Detection and Delineation-Evolution of LiDAR Data*. Remote Sensing and Spatial Information Science, 2016, Volume 8, pp.333
- [26] Zhu C, Zhang X, Baogang H et al. *Reconstruction of Tree Crown Shape from Scanned Data*, Conference Paper, Juin 2008

Table des annexes

Annexe 1 Exécuter un script R dans FME	64
Annexe 2 Implémentation de la fonction « ashape3d » de R.....	66
Annexe 3 Développement d'une interface graphique	67

Annexe 1

Exécuter un script R dans FME

L'exécution d'un script R dans FME est permise par le transformer "RCaller" pour lequel une documentation est disponible en cliquant sur le lien suivant :

http://docs.safe.com/fme/2018.0/html/FME_Desktop_Documentation/FME_Transformers/Transformers/rcaller.htm

Les versions antérieures à la version 2018 de FME sont limitées à un temps de calcul inférieur à 30s. Dans notre cas, la version FME 2018 est indispensable.

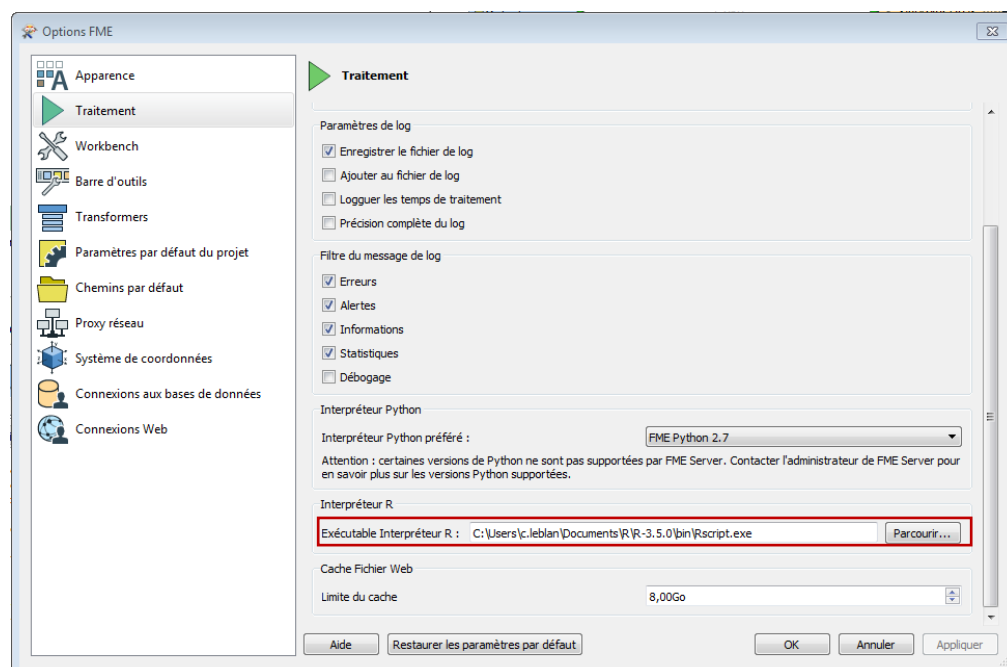
Paramétrage de FME

Tout d'abord, il faut renseigner à FME le chemin d'accès à l'exécutable '*Rscript.exe*'. Ce chemin peut être obtenu de la manière suivante :

- Dans la barre de recherche de windows taper "R"
- Clique droit sur l'icône de R
- Sélectionner "Ouvrir l'emplacement du fichier"
- Copier le chemin d'accès au dossier "bin" dans lequel se situe l'exécutable

Ensuite, dans le workbench de FME, dans l'onglet "outil" de la barre de menu, sélectionner "Option FME" puis "Traitement". Faire défiler la fenêtre et renseigner le champ "Exécutable Interpréteur R" par le chemin d'accès enregistré en amont.

Cliquer sur parcourir et sélectionner l'unique exécutable R proposé et appelé "*Rscript.exe*"



Paramétrage de R

Pour fonctionner sous FME, R requiert aussi d'être paramétré. D'après la documentation du transformateur "RCaller", il suffit pour cela de lancer R en ligne de commande et en tant qu'administrateur et d'exécuter le code suivant :

```
R> install.packages("sqldf")
```

Puis d'installer ensuite de la même façon les packages nécessaires au script, soit dans notre cas :

```
R> install.packages("alphashape3d")
```

```
R> install.packages("rgdal")
```

```
R> install.packages("maptools")
```

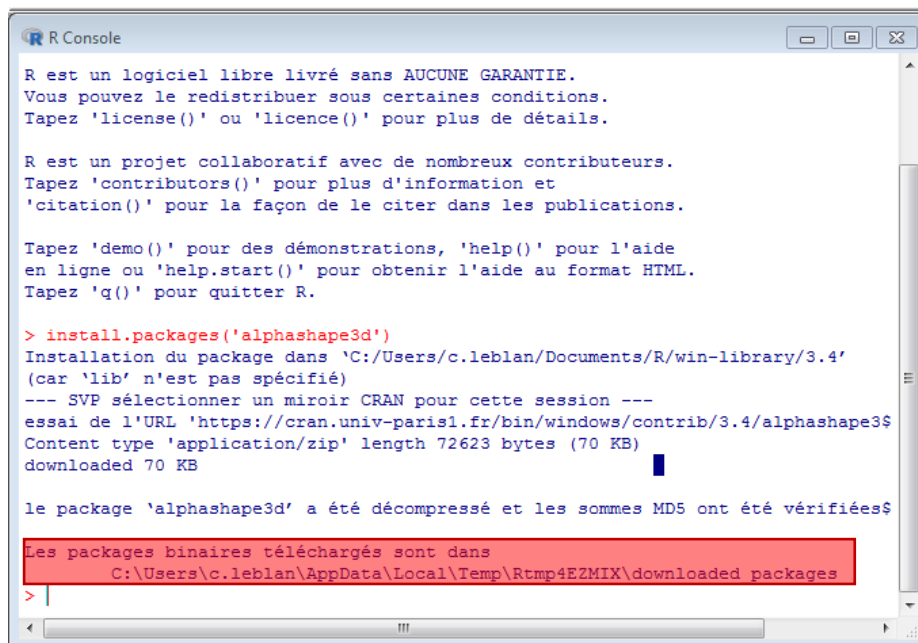
Normalement, à ce stade R et FME sont tous deux paramétrés de manière à être interoperables et le script R peut être exécuté à partir d'un script FME.

Problème de chargement de Packages lors du lancement du script sous FME

Malgré les paramétrages exécutés en amont, R peut rencontrer des difficultés lors du chargement de ses librairies. Il s'agit souvent de problèmes liés au renseignement du chemin d'accès au dossier au PATH.

Si FME indique un message d'erreur concernant un problème de chargement de librairie :

- *Charger le package en question dans R*
- *Récupérer le chemin d'accès au dossier temporaire dans lequel il a été enregistré*
- *Extraire le dossier et le coller dans le dossier library (C:\Program Files\R\R-3.4.4\library)*



```
R Console

R est un logiciel libre livré sans AUCUNE GARANTIE.
Vous pouvez le redistribuer sous certaines conditions.
Tapez 'license()' ou 'licence()' pour plus de détails.

R est un projet collaboratif avec de nombreux contributeurs.
Tapez 'contributors()' pour plus d'information et
'citation()' pour la façon de le citer dans les publications.

Tapez 'demo()' pour des démonstrations, 'help()' pour l'aide
en ligne ou 'help.start()' pour obtenir l'aide au format HTML.
Tapez 'q()' pour quitter R.

> install.packages('alphashape3d')
Installation du package dans 'C:/Users/c.leblan/Documents/R/win-library/3.4'
(car 'lib' n'est pas spécifié)
--- SVP sélectionner un miroir CRAN pour cette session ---
essai de l'URL 'https://cran.univ-paris1.fr/bin/windows/contrib/3.4/alphashape3d'
Content type 'application/zip' length 72623 bytes (70 KB)
downloaded 70 KB

le package 'alphashape3d' a été décompressé et les sommes MD5 ont été vérifiées$

Les packages binaires téléchargés sont dans
C:\Users\c.leblan\AppData\Local\Temp\Rtmp4EZMIX\downloaded packages

>
```

Annexe 2

Implémentation de la fonction « ashape3d » de R

```

setwd("C:/Users/coco1/Desktop/R")

#Créer une boucle qui parcourt l'ensemble des fichiers .xyz
fichiers <-list.files(pattern="*.xyz")

for (i in fichiers){
  data<-read.csv(i,header=FALSE,sep=' ',dec=' ')

  #Mettre les données sous forme matricielle
  datamat<-as.matrix(data)

  #Les données passe en réels
  storage.mode(datamat)<-"numeric"

  meanX = mean(datamat[,1])
  meanY = mean(datamat[,2])
  meanZ = mean(datamat[,3])

  datamat[,1] <- datamat[,1] - meanX
  datamat[,2] <- datamat[,2] - meanY
  datamat[,3] <- datamat[,3] - meanZ

  ##### CALCUL DE L'ALPHA SHAPE
  alpha <-5
  Arbre <- ashape3d(datamat, alpha = alpha, pert=TRUE)
  #Arbre$x : Coordonnées des sommets

  sommets = Arbre$x
  sommets[,1]<-sommets[,1]+meanX
  sommets[,2]<-sommets[,2]+meanY
  sommets[,3]<-sommets[,3]+meanZ

  ##### RECUPERATION DES VOLUMES

  ##### EXPORT DES OBJETS MESH CREEES
  #Création du chemin d'accès au fichier.obj
  nameFichier <-unlist(strsplit(i, ".csv")) #Découpage de la chaine de caractère / \
  Arbre.obj <-paste0("Arbre3D", "_",nameFichier,".obj")

  #Ecrire la première ligne du fichier pour renseigner l'extension (Facultatif)
  write("#OBJ",file=Arbre.obj,append=FALSE,sep=" ")

  #Ecrire les coordonnées des sommets | v x y z
  write(t(cbind("v",sommets)),file=Arbre.obj, ncolumns=4,append=TRUE,sep=" ")

  #Ecrire les faces | f sommet1 sommet2 sommet3
  Triangles<-Arbre$triang
  tr<-Triangles[Triangles[, 9]==2 | Triangles[, 9]==3, c("tr1","tr2","tr3")]
  write(t(cbind("f",tr)),file=Arbre.obj, ncolumns = 4, append =TRUE, sep=" ")

  #Ecrire le nombre de faces
  write(cbind("#",dim(tr)[1]),file=Arbre.obj,ncolumns=2,append=TRUE,sep=" ")}

```

Dans FME, le chemin d'accès (renseigné par la fonction setwd) et la valeur du critère « alpha » sont dynamique.

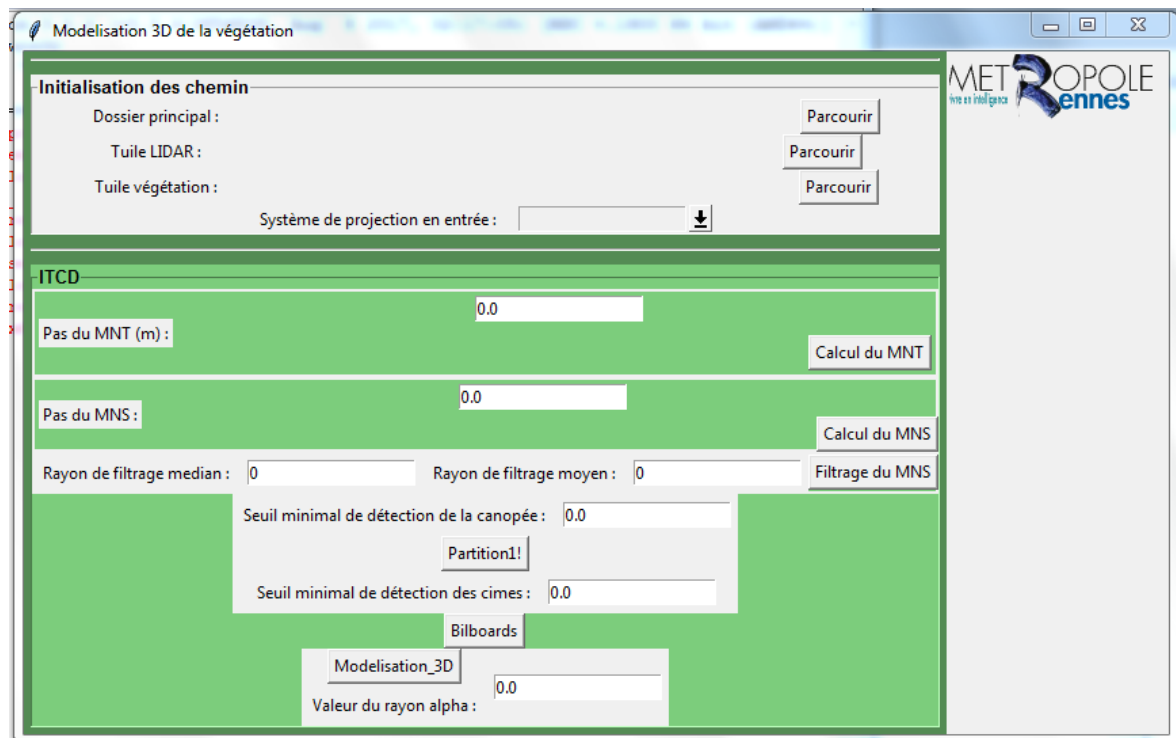
Annexe 3

Développement d'une interface graphique

TKinter : Le module commun de création de GUI sous Python

Une interface graphique (ou GUI) est un dispositif permettant une simplification du dialogue entre l'utilisateur et la machine. Développées à la fin des années 1970 pour simplifier des opérations jusqu'alors réalisées par le biais de lignes de commandes considérées lourdes et obscures par les utilisateurs, elles sont composées de fenêtres, icônes, menus et systèmes de pointage permettant de rendre leur fonctionnement plus intuitif.

L'algorithme de détection et de segmentation de la végétation ayant été développé dans un langage de programmation Python, c'est naturellement que le package TKinter, librairie standard et la plus couramment utilisée, s'est imposé. De type "cross-platform", TKinter s'adresse avant tout aux développeurs familiers avec la programmation python mais moins à l'aise avec le développement d'interfaces de type GUI. À noter cependant qu'il existe de nombreuses autres librairies offrant des perspectives de développement plus complexes (PyQT, WxPython, ...).



Liste des figures

Figure 1 : Carte localisant le département de l'Ille-et-Vilaine.....	10
Figure 2: Logo de Rennes Métropole.....	10
Figure 3: Organigramme de l'unité Cartographie et Diffusion 2D/3D.....	12
Figure 4: Maquette 3D et information territoriale.....	13
Figure 5 : Modélisation du bruit 10 secteurs à enjeux "bruit" sur Rennes	14
Figure 6: Logo du logiciel Rhinoceros.....	14
Figure 7: LoD en CityGML	15
Figure 8: Représentation de type "billboard" texturé des arbres.....	17
Figure 9: Composition d'un système ALS.....	18
Figure 10 : Phase Shift.	19
Figure 11: par triangulation active sur des feuilles	19
Figure 12 : Décomposition de retours LiDAR	19
Figure 13: Calcul d'un MHC	20
Figure 14: Classification propre au format.LAS définie par l'ASPRS	20
Figure 15: Nettoyage du nuage LiDAR.	23
Figure 16 : Logo du logiciel FUSION/LDV	23
Figure 17: Rasterisation de donnée discrète.....	24
Figure 18: Segmentation des houppiers et seuil minimal de détection	24
Figure 19: Partition des arbres selon l'application de buffers géométriques sur les couronnes.	25
Figure 20: Pondération du MHC par un raster de pentes	26
Figure 21 : Les différents types de coordonnées.....	26
Figure 22: Transformation de systèmes de coordonnées.....	27
Figure 23:Reconstruction de tronc et de branches par approximation cylindrique d'un nuage de points	28
Figure 24: Illustration du principe de la squelettisation selon Coa et al. 2010.....	29
Figure 25 : Résultats obtenus par l'implémentation de l'algorithme de Cao et al	29
Figure 26: Construction d'un diagramme de Voronoï pour un nombre croissant de germes	31
Figure 27: Cercle reliant des voisins de Voronoï.....	31
Figure 28: Triangulation de Delaunay	31
Figure 29: Reconstruction de la canopée d'après une surface TIN	32
Figure 30 : Structure TIN	32
Figure 31: Illustration d'un ensemble convexe.....	32
Figure 32: Reconstruction d'arbres individuels par combinaison d'enveloppe convexes et de modèles cylindriques.....	33
Figure 33: Forme irrégulière des couronnes.....	33
Figure 34: Voisins de Voronoï joint par un arc de cercle	34
Figure 35: Enveloppe α -convexe et son complément	34
Figure 36: Comparaison des critères convexes et alpha dans un espace 2D.....	34
Figure 37: α -extreme	35
Figure 38: α -voisins	35
Figure 39: Influence du rayon sur la forme alpha.	36
Figure 40 : Reconstruction de type alpha shape d'un érable	36
Figure 41: Ligne d'horizon présentant la classe "Végétation" du nuage de points LiDAR.....	37
Figure 42 : Couche objet de bâtis de Rennes Métropole.....	38
Figure 43 : Points résiduels identifiés par le plugin Cloud/Mesh de CloudCompare	38
Figure 44: Lucarne de toit	38
Figure 45: Résidus liés aux lucarnes de toits dans le nuage de points LiDAR	38
Figure 46 : Statistical Outlier Removal. Crédit PCL.....	39
Figure 47: Comparaison de l'analyse de densité par nombre de voisins ou par surface	40
Figure 48: Calcul de CHM sur le secteur de l'avenue Henri Freville.....	41
Figure 49: Quatre catégories de houppiers segmentés	42
Figure 50 : Suppression des objets incohérents d'après leur forme.....	43

Figure 51: Regroupement des zones fermées.....	43
Figure 52: Récupération des couronnes en zones fermées.....	44
Figure 53 : Extraction des diamètres de couronnes en zone ouverte	45
Figure 54 : Considération de l'emprise des houppiers en zone fermées.....	45
Figure 55: Résultat de la modélisation "en croisillon"	47
Figure 56: clustering du nuage de points	49
Figure 57: Houppier en 2.5D.....	49
Figure 58: Logo R	50
Figure 59: Reconstruction alpha-shape pour des critères alpha croissants de gauche à droite	51
Figure 60: Comparaison des structures des formats .OFF et .OBJ	52
Figure 61: Objet "regroupement d'arbres"	55
Figure 62: Objet "arbre isolé"	55
Figure 63: Résultat final de la modélisation 3D de la végétation sur le territoire de Rennes Métropole	55
Figure 64: Comparaison des classifications LiDAR réalisées par l'IGN et GEOFIT	56
Figure 65: Comparatif de classifications.....	56
Figure 66: Confusion de zone ouverte et fermée	57
Figure 67 : Des houppiers "éclatés"	57
Figure 68: Clustering par voxels (Crédit : B. Douillard)	58
Figure 69: Clustering par K-moyenne.....	58
Figure 70: Répartition des écarts entre le nuage de points et la modélisation 3D.....	58
Figure 71: Défaillances de reconstruction des houppiers.....	59

Liste des tableaux

Tableau 1: Unités du service SIG de Rennes Métropole.....	11
Tableau 2: Primitives géométriques des bâtiments sur le territoire de la ville de Rennes	16
Tableau 3: Caractéristiques techniques du nuage de points LiDAR	19
Tableau 4: Présentation des six phases de l'algorithme de détection et de segmentation développé par C.DOCEUL	22
Tableau 5 : Description de la couche "Cimes_all.shp"	41
Tableau 6: Description de la couche "Couronne_all_clean.shp"	42
Tableau 7: Description de la couche "Zones_ouvertes.shp"	44
Tableau 8: Description de la couche "Zones_fermees"	44
Tableau 9: Présentation de la couche "Modelisation_ouverte.shp"	46
Tableau 10: Présentation de la couche "Modélisation_fermee.shp"	47
Tableau 11: Nomenclature des objets arbres.....	48
Tableau 12: Méthode de reconstruction des 3D des houppiers.....	50
Tableau 13: De la 2.5D à la 3D, reconstruction des houppiers	53

Développement d'un processus de modélisation 3D de la végétation à partir de donnée discrète et dans un objectif d'intégration dans une maquette numérique urbaine.

Mémoire d'Ingénieur C.N.A.M., Paris 1992

RESUME

Un processus de détection et de segmentation de la végétation a permis le clustering d'un nuage de points LiDAR aérien en sous-ensembles constituant des houppiers d'arbres.

Généralisant l'enveloppe convexe d'un ensemble discret de points, l'utilisation de « formes alpha » a permis une modélisation 3D (au sens volumique) de ces houppiers selon un niveau de détail paramétrable.

La transcription des objets générés dans un format CityGML rendrait possible leur intégration dans une maquette numérique urbaine.

Mots clés : LiDAR, segmentation de la végétation, formes alpha, modélisation 3D, CityGML, maquette numérique urbaine.

SUMMARY

A process of vegetation's detection and segmentation allowed the clustering of an aerial LiDAR points cloud into subsets constituting crowns of trees.

Generalizing the convex hull of a discreet set, the use of « alpha forms » allowed a 3D modelling (in the volume sense) of these crowns according to a parameterized level of detail.

The transcription of objects generated in a CityGML format would make it possible to integrate them into an urban numerical model.

Key words : LiDAR, vegetation's segmentation , alpha forms, 3D modelisation, CityGML, Urban Numerical Model.