



UNIVERSITÉ
DE GENÈVE



Conservatoire
et Jardin botaniques
Genève

CERTIFICAT COMPLÉMENTAIRE EN GÉOMATIQUE 2020



ESTIMATION DE LA SURFACE DE CANOPÉE ET POSITIONNEMENT DES ARBRES ISOLÉS DU CANTON DE GENÈVE À L'AIDE DE DONNÉES LIDAR

RAPPORT DE STAGE

Antonioli Guillaume
Octobre 2020 - Janvier 2021

Superviseur : Dr P. Martin
Enseignant chargé du suivi académique : Dr G. Giuliani
Membre du comité académique : Prof. A. Lehmann

Table des matières

Liste des Figures.....	4
Liste des Tableaux.....	5
Liste des Abréviations.....	6
I. Résumé.....	7
II. Introduction	8
1. Présentation de la structure d'accueil	8
2. Contexte général	8
3. Objectifs	9
4. Travaux précédents.....	9
III. Présentation des données.....	11
1. LiDAR	11
a. Jeu de données	12
b. Biais détectés	13
2. Autres données utilisées	15
IV. Méthodologie.....	16
1. Modèle conceptuel global.....	16
2. Prétraitement du jeu de donnée LAS.....	17
a. Création d'un buffer autour des tuiles LAS (LAStools).....	17
b. Détection des murs de soutènement faussement classifiés (ArcGIS, Python).....	18
c. Correction et allègement des données LAS (FME)	19
3. Production des Modèles Numériques de Canopée (Python).....	21
a. Rastérisation et création du MNHC	21
b. Découpe du MNHC selon une grille définie	21
c. Vectorisation et création du MNC	22
d. Correction du MNHC.....	22
e. Création du MNC corrigé	22
f. Assemblage du MNHC et MNC sur l'ensemble du Canton	22
4. Détection de l'emplacement des troncs	23
a. Détection des troncs selon différents paramètres (Python/Octave)	23
b. Création de la couche de position des troncs (Python/ArcGIS).....	25

5.	Ajout d'attributs au Modèle Numérique de Canopée	25
V.	Résultats.....	27
1.	Prétraitements	27
2.	Modèle Numérique de Hauteur de Canopée.....	29
3.	Modèle Numérique de canopée	31
4.	Emplacement des arbres isolés.....	34
VI.	Discussion.....	40
1.	Corrections de la classification.....	40
2.	Production des couches de canopée	40
3.	Détection des troncs	41
4.	Déroulement du stage.....	43
VII.	Conclusion	44
VIII.	Remerciements	45
IX.	Références.....	46
X.	Annexes.....	48

Liste des Figures

Figure 1 - Relevé topographique par laser scanning aéroporté	11
Figure 2 - Détail du fonctionnement d'une impulsion LiDAR et des différents retours	11
Figure 3 - Couverture des données LiDAR 2019 et zone d'étude	12
Figure 4 - Exemple de biais détectés	14
Figure 5 - Modèle conceptuel général.....	16
Figure 6 - Interface de l'outil LASclip.exe	17
Figure 7 – Workbench FME du modèle LAS_Pretraitement.....	20
Figure 8 - Aperçu des paramètres du script MNC_calculator.py	21
Figure 9 - Illustration de la méthode de détection des troncs	24
Figure 10 – Exemple au parc des Bastion des différentes étapes de prétraitements des données LAS.....	27
Figure 11 - Retours LiDAR de végétation supprimés au-dessus du mur de la treille.	28
Figure 12 - Exemple à Carouge de la préservation des hauteurs des canopée avec cette méthode	29
Figure 13 - Illustration de la correction des hauteurs de MNHC.....	30
Figure 14 – Exemple au parc du monument Brunswick de l'optimisation du MNC.....	31
Figure 15 - Aperçu à l'échelle du canton de Genève des deux MNC produits.....	32
Figure 16 - Zoom sur MNC en centre-ville de Genève (Jonction).....	33
Figure 17 - Zone d'étude du parc des Bastions.....	35
Figure 18 - Zone sélectionnée pour l'évaluation des paramètres de détection.	36
Figure 19 - Histogramme de comparaison avec ICA du nombre de troncs détectés dans la zone d'étude selon 4 jeux de paramètres différents.	36
Figure 20 - Représentation du nuage de point LiDAR de végétation superposé à l'orthophoto du parc des Bastions	37
Figure 21 – Visualisation des différences de détection des troncs entre les paramètres 'Derniers retours' et 'Tous les retours'	37
Figure 22 - Comparaison de variation du paramètre minLength sur le nombre d'arbres détectés avec la méthode 'Tous les retours' sur l'ensemble du canton.	38
Figure 23 – Troncs détectés et arbres ICA au quartier de Plainpalais.....	39
Figure 24 – Troncs détectés et arbres ICA au Jardin Botanique.....	39

Liste des Tableaux

Tableau 1 - Principaux attributs des données LAS.....	12
Tableau 2 - Caractéristiques de l'acquisition LiDAR 2019	12
Tableau 3 - Biais détectés dans les données LiDAR brutes.....	14
Tableau 4 - Données externes utilisées pour la réalisation du projet.....	15
Tableau 5 - Paramètres du script Stems_detection.py	24
Tableau 6 - Comparaison du jeu de donnée LiDAR avant et après traitement.....	27
Tableau 7 - Comparaison des deux MNC.....	32
Tableau 8 - Comparaison des troncs détectés suivants différentes valeurs de paramètres	35
Tableau 9 - Comparaison de la part de polygones de canopée possédants au moins un tronc suivant les différentes méthodes de détection.	38
Tableau 10 - Jeu de paramètres de la couche de tronc finale.....	39

Liste des Abréviations

CJBG.....	Conservatoires et Jardin Botaniques de la Ville de Genève
DIT.....	Direction de l'Information du Territoire
DT.....	Département du Territoire
ICA.....	Inventaire Cantonal des Arbres
LiDAR.....	Light Detection And Ranging
MNC.....	Modèle Numérique de Canopée
MNHC.....	Modèle Numérique de Hauteur de Canopée
MNT.....	Modèle Numérique de Terrain
MNTB.....	Modèle Numérique de Terrain et du Bâti
MNS.....	Modèle numérique de Surface
OCAN.....	Office Cantonal de l'Agriculture et de la Nature
SEVE.....	Service des Espaces Verts
SIPV.....	Système d'Information du Patrimoine Vert
SITG.....	Système d'Information du Territoire à Genève
VDG.....	Ville de Genève

I. Résumé

Ce rapport présente le travail produit dans le cadre d'un stage de quatre mois à 80 % au Conservatoire et Jardin Botaniques de la Ville de Genève au sein du Certificat Complémentaire en Géomatique de l'université de Genève. À l'aide de données LiDAR acquises par le canton de Genève en mars 2019, l'objectif de ce travail est de produire des indicateurs représentatifs et qualitatifs du patrimoine arboré du canton. Grâce à une nouvelle approche des données LiDAR, un Modèle Numérique de Canopée a été généré pour tout le territoire ainsi qu'une couche déterminant l'emplacement des arbres isolés du canton. En addition de ces couches produites, l'ensemble des méthodes et scripts sont joints aux résultats et détaillés afin de permettre la transmission et l'amélioration de cette méthodologie. Une bonne confiance est attribuée au Modèle Numérique de Canopée. Calibrée afin de ne pas « oublier » d'arbre, la détection des troncs est fiable pour une majorité des arbres isolés et de taille modeste, néanmoins cette confiance diminue fortement dans le cas de massifs d'arbres ou d'arbres volumineux.

II. Introduction

Ce document présente le travail effectué durant le stage en géomatique au sein des Conservatoire et Jardin Botaniques de la Ville de Genève (CJBG) entre octobre 2020 et janvier 2021, dans le cadre du Certificat Complémentaire en Géomatique à l'Université de Genève. Le travail de ce stage a été en grande partie effectué en collaboration avec Gilles Gay des Combes lui aussi en stage au Service des Espaces Verts (SEVE) et aux CJBG.

1. PRESENTATION DE LA STRUCTURE D'ACCUEIL

Les Conservatoire et Jardin Botaniques la Ville de Genève sont une institution genevoise multifacette. C'est à la fois un musée naturel à ciel ouvert, une bibliothèque recensant plus de 120 000 documents et l'un des herbiers parmi les plus importants au monde. C'est aussi un pôle scientifique développant des « recherches centrées sur l'étude des organismes végétaux et fongiques ainsi que sur la biodiversité qu'ils représentent¹ ». Ce stage s'est déroulé dans l'unité Flores, Conservation et Systèmes d'Information, dans le cadre du projet du « Système d'Information du Patrimoine Vert » (SIPV) dont l'objectif principal est de centraliser toutes les données sur la biodiversité végétale de la région genevoise dans un même référentiel.

2. CONTEXTE GENERAL

Dans un contexte de réchauffement climatique à l'échelle globale, les villes sont en première ligne face à l'augmentation de la fréquence des vagues de chaleur et de leur intensité. Un consensus de plus en plus large se fait autour de l'importance du patrimoine arboré dans le milieu urbain afin d'améliorer la résilience des villes aux vagues de chaleur. Consciente de ces enjeux, la ville de Genève s'est engagée depuis 2001 en faveur d'un développement durable². En juin 2020 une nouvelle politique de gestion des arbres³ est annoncée dans le but « d'offrir plus d'espaces ombragés » et de « réduire les îlots de chaleur ». Dans ce cadre, il est important de pouvoir évaluer et quantifier l'ampleur du patrimoine arboré genevois afin de fournir des outils permettant d'observer et de mettre en œuvre cette politique. Depuis 2001, Genève mène des campagnes régulières d'acquisition de données LiDAR afin de compléter son panel d'outils de cartographie. Souvent perçues comme lourdes et difficilement exploitables, ces données ont un potentiel qui reste encore largement inexploité, notamment dans les domaines de l'aménagement, de la foresterie et de la nature et du paysage.

Dans ce contexte, il est de plus en plus essentiel d'exploiter tout le potentiel de cette technologie de pointe afin de la mettre au service de l'environnement.

¹ https://www.ville-ge.ch/cjb/activites_scientifiques.php

² <https://www.geneve.ch/fr/actualites/dossiers-information/changement-climatique-geneve>

³ <https://www.geneve.ch/fr/actualites/ville-geneve-augmente-plantation-arbres-afin-repondre-urgence-climatique>

3. OBJECTIFS

Ce stage s'inscrit dans une démarche d'élaboration de méthodes permettant d'évaluer et de quantifier le patrimoine arboré du canton de Genève à l'aide de données LiDAR récoltées en mars 2019. Ce stage a vocation à explorer ces données LiDAR et les méthodes existantes afin de répondre à deux objectifs principaux :

- Réaliser un Modèle Numérique de Canopée (MNC) pour l'ensemble du canton donnant une information de surface recouverte par les arbres.
- Élaborer une méthode de détection de l'emplacement individuel des troncs afin de pouvoir quantifier le nombre d'arbres et si possible à l'avenir, mettre à jour l'Inventaire Cantonal des Arbres (ICA).

Il est important de préciser que l'attente principale de ces résultats concerne les arbres isolés. On entend par arbres isolés l'ensemble des arbres n'étant pas dans des zones forestières telles que définies par le cadastre forestier. Les parcelles de forêt sont évidemment traitées lors de l'élaboration des résultats, mais ne sont pas examinées en tant que produit final.

Les résultats attendus par ce stage sont donc autant les couches de données produites que les méthodes/scripts mis au point pour y parvenir. Un effort de transparence et d'explication de l'utilisation des outils produits dans un objectif de transmission est donc attendu.

4. TRAVAUX PRECEDENTS

La photographie aérienne couplée aux relevés de terrains représente la méthode traditionnelle afin de réaliser un inventaire arboré dans une zone donnée. Depuis le début des années 2000, l'utilisation de la technologie LiDAR n'a cessé d'augmenter dans ce domaine. Une grande quantité d'études existe sur les applications du LiDAR sur le domaine forestier afin de modéliser, entre autres, son extension, sa structure et son volume (Lim et al., 2003 ; Hopkinson et al., 2004). La constante augmentation de quantité et de précision des données au fil des ans permet maintenant d'appliquer le potentiel du LiDAR au service des villes et de leur patrimoine arboré.

À l'échelle du Canton de Genève, sur la base de l'acquisition LiDAR de 2017, un Modèle Numérique de Canopée ainsi qu'une détection des arbres isolés ont été réalisés lors d'un stage de 6 mois à l'OCAN (Tiffay, 2018). La méthode utilisée afin de produire ces données est une « approche raster ». C'est-à-dire que les données LiDAR ont d'abord été rastérisées avant d'en extraire les informations recherchées. Bien que très efficace pour travailler sur un grand jeu de données, cette étape de rastérisation implique une perte en information des données brutes. Le MNC produit par M-C. Tiffay repose sur la formule suivante :

$$MNC = MNS^i - MNTB^{ii}$$

ⁱMNS = Modèle Numérique de Surface

ⁱⁱMNTB = Modèle Numérique de Terrain avec Bâtiments

Quelques soucis techniques sont à relever avec cette méthode, tels que le découpage de la canopée par la bordure des bâtiments ou des valeurs aberrantes issues par les méthodes d'interpolation afin de combler les pixels sans valeurs lors de la création du raster MNTB.

La détection des arbres a, quant à elle, été réalisée à l'aide du script `treeDetectLmax.py`⁴ de M. Parkan. La méthode consiste à détecter les cimes d'arbres en utilisant le raster du Modèle Numérique de Hauteur de Canopée (MNHC).

Dans ce stage une nouvelle approche est utilisée. Afin d'explorer au maximum le potentiel des données LiDAR, la majorité des traitements se fait depuis le nuage de points afin d'en extraire directement les modèles numériques de canopée et la localisation des arbres isolés.

⁴<https://sites.google.com/site/lidareforet/ressources/6-tutoriels/python/detection-des-cimes-maxima-local?tmpl=%2Fsystem%2Fapp%2Ftemplates%2Fprint%2F&showPrintDialog=1>

III. Présentation des données

1. LiDAR

Le LiDAR (Light Detection And Ranging) est une technique de télédétection optique de haute précision qui utilise des impulsions laser afin d'échantillonner la surface terrestre sous forme de nuage de points avec des coordonnées x, y, z. Principalement aéroportée en vue d'une application cartographique, la technologie LiDAR tend à se démocratiser et à s'imposer comme un outil de choix dans de nombreux domaines tels que, par exemple, l'océanographie, la biodiversité ou encore l'archéologie (Guo et al., 2017 ; Churnside, 2013 ; Inomata, 2018). Depuis 2020, elle est même intégrée dans les tout derniers smartphones⁵.

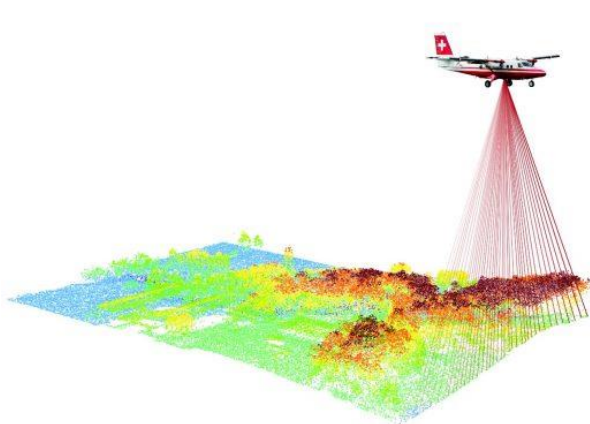


Figure 1 - Relevé topographique par laser scanning aéroporté. Source : www.swisstopo.ch

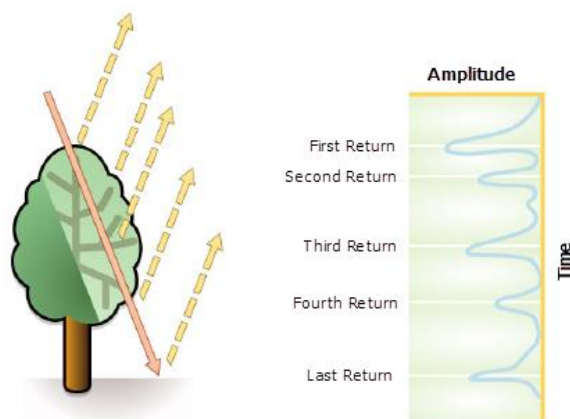


Figure 2 - Détail du fonctionnement d'une impulsion LiDAR et des différents retours. Source : <https://desktop.arcgis.com>

D'un point de vue technique, le LiDAR émet plusieurs centaines de milliers de pulsations lumineuses par seconde qui sont renvoyées au capteur par les différents objets rencontrés (Figure 1). La position est calculée en fonction du temps que mettent les pulsations à faire l'aller-retour. Les pulsations laser émises peuvent revenir au capteur sous forme d'un ou plusieurs retours. En effet, une pulsation laser peut traverser plusieurs surfaces de réflexion qui renverront chacune un retour différent. D'une manière générale, le premier retour correspond à l'objet le plus haut dans le paysage tel que la cime d'un arbre ou le toit d'un bâtiment, les retours intermédiaires sont associés à la structure de la végétation, et le dernier retour au sol (Figure 2). Évidemment, le dernier retour ne correspond pas toujours au sol. Un toit bétonné, un tronc ou même une grosse branche empêchera l'impulsion de traverser, le dernier retour correspondra donc au dernier objet reflété. Les nuages de points ainsi détectés sont stockés dans un format binaire appelé LAS. C'est un format de fichier standard qui permet de conserver les informations

⁵ https://www.sciencesetavenir.fr/high-tech/reseaux-et-telecoms/a-quoi-correspond-vraiment-le-scanner-lidar-qui-equipe-l-iphone-12-pro-d-apple_148385

spécifiques aux données LiDAR et de faciliter leur échange. Les principaux attributs du nuage de point LiDAR sont présentés dans le Tableau 1 ci-dessous.

Attributs	Description
Coordinates (x,y,z)	Coordonnées x, y et z du point.
Intensity	La force de retour de l'impulsion laser qui a généré le point lidar.
Return Number	Une impulsion laser émise peut comporter jusqu'à cinq retours, selon les entités sur lesquelles la réflexion s'effectue et selon les fonctionnalités du scanner laser utilisé pour collecter les données. Le premier retour est signalé comme retour numéro un, le deuxième comme retour numéro deux et ainsi de suite.
Number of returns	Le nombre de retours correspond au nombre total de retours pour une impulsion donnée. Par exemple, un point de données laser peut être renvoyé deux fois (numéro de retour) au sein d'un nombre total de cinq retours.
Class Code	Chaque point lidar qui est post-traité peut avoir une classification qui définit le type d'objet ayant reflété l'impulsion laser. Les différentes classes sont définies à l'aide de codes numériques entiers dans les fichiers LAS.
GPS Time	Horodatage GPS auquel le point laser a été émis de l'avion. L'heure est exprimée en secondes GPS de la semaine.

Tableau 1 - Principaux attributs des données LAS. Source : <https://desktop.arcgis.com/fr/arcmap/10.3/manage-data/las-dataset/what-is-lidar-data-.htm>

a. Jeu de données

Les données LiDAR 2019 du canton de Genève ont été collectées par l'entreprise bernoise *FLOTTRON Ingenieure* sous mandat du Département du Territoire (DT) de l'État de Genève. L'acquisition a été réalisée durant la nuit du 5 au 6 mars 2019. Les données sont divisées en 1567 tuiles de 500 mètres sur 500 mètres recouvrant le canton de Genève ainsi qu'une

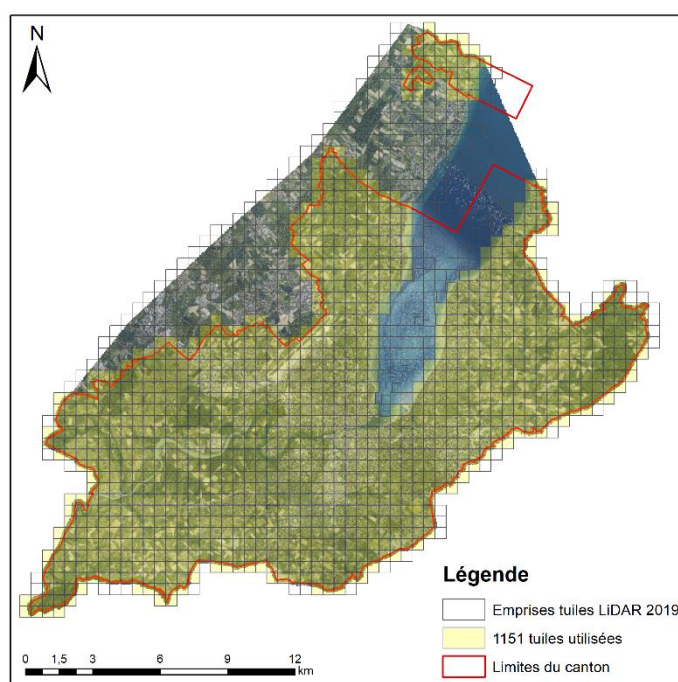


Figure 3 - Couverture des données LiDAR 2019 et zone d'étude.

LIDAR 2019 - DT / FLOTTRON Ingenieure	
Date de collecte	5 - 6 mars 2019 (nuit)
Densité	20 - 40 pts /m2
Précision :	
- Planimétrique	< 20 cm
- Altimétrique	< 10 cm
Tuiles	1567 tuiles de 500 m x 500 m
Format	Las 1.2
Classes	<ul style="list-style-type: none"> 1 Non-classifié 2 Sol 3 Végétation basse 5 Végétation haute 6 Bâtiments 7 Points bas ou isolés 9 Eau 13 Ponts/Passerelles 15 Sol (points additionnels) 16 Bruit 19 Hors-périmètre

Tableau 2 - Caractéristiques de l'acquisition LiDAR 2019.

zone tampon de 100 mètres à l'extérieur des frontières, une partie du canton de Vaud comprise entre l'enclave de Céligny et le reste du territoire genevois ainsi qu'une partie du territoire français. La méthodologie élaborée lors de ce stage n'a utilisé qu'une sélection de 1151 tuiles recouvrant l'ensemble du canton de Genève (Figure 3). L'essentiel des informations et caractéristiques du jeu de donnée LiDAR acquis par l'entreprise *FLOTTRON Ingenieure* sont présentées sur le Tableau 2 ci-dessus.

b. Biais détectés

D'une manière générale la qualité de l'acquisition LiDAR 2019 est excellente, autant au niveau de la densité de points que de leur précision. Néanmoins, l'exploration des données brutes a révélé l'existence de biais qui ont une influence plus ou moins marquée sur le produit final de ce travail.

Tout d'abord, l'application première des données pour le DT est la création des MNT et MNS du canton de Genève à partir des points de sol, de bâtiments et d'eau (classes 2, 6 et 9). Un feuillage dense ayant tendance à réduire la densité des points du sol qu'il recouvre, le vol d'acquisition a été planifié en fin d'hiver afin de réduire ce biais et d'augmenter la qualité des produits dérivés. À cette époque, le feuillage de la plupart des arbres feuillus (angiospermes) ne s'est pas encore développé ou n'en est qu'au commencement. Cela représente un biais important qui aura un impact sur l'issue de ce travail : la surface de canopée calculée à partir de ces données ne sera pas une estimation de l'étendue maximale couverte par le feuillage des arbres à Genève. Cependant, grâce à la densité et la précision des données, la structure du branchage est presque entièrement détectée et permet de reconstituer une surface de canopée proche de l'étendue maximale. Seuls les arbres taillés en « tête de chat » (principalement des platanes et tilleuls en milieu urbain) verront l'estimation de leur surface de feuillage considérablement réduite en fonction de l'avancée de la taille des branches. Dans une optique de comparaison temporelle de la canopée, cela ne présente pas un biais majeur tant que les prochaines acquisitions se font à la même période. D'un autre point de vue, cette acquisition hivernale est un avantage certain pour la détection de l'emplacement des troncs d'arbres. L'absence de feuillage permettant une meilleure détection de la structure des arbres par les données LiDAR.

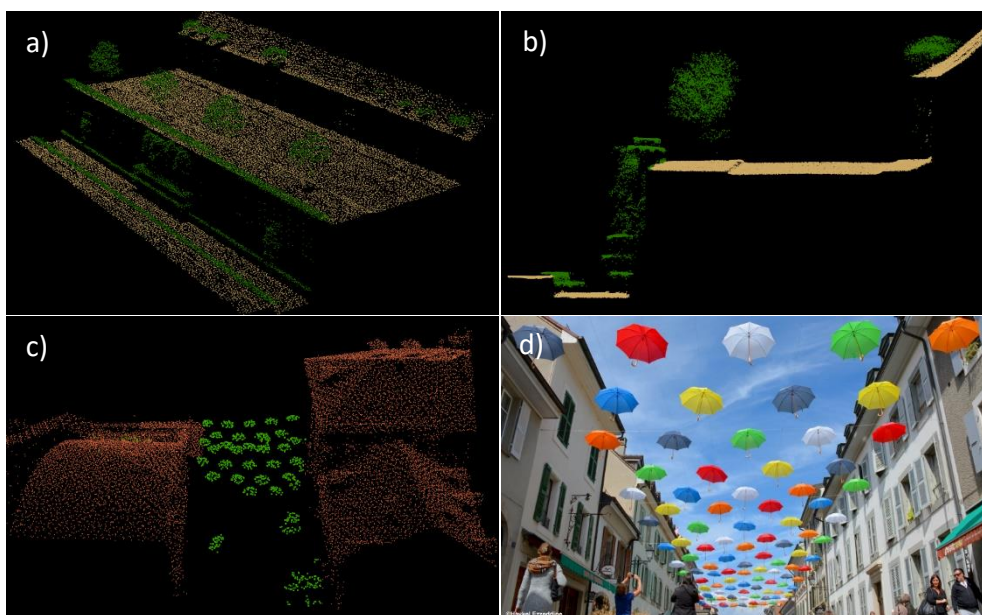


Figure 4 - Exemple de biais détectés. a) Vue 3D du nuage de point LiDAR du mur des réformateurs classifié en végétation (vert) ; b) Coupe transversale du mur des réformateurs ; c) Vue 3D des parapluies de la rue St-Joseph classifiés en végétation ; d) Photo des parapluies de la rue St-Joseph à Carouge

Ensuite, plusieurs erreurs de classifications subsistent au sein de la classe 5 (Végétation haute). Les principaux objets faussement classifiés comme végétation haute sont les murs, et plus particulièrement les murs de soutènement, c'est-à-dire des murs servant à soutenir un terrain en pente et ayant donc une différence d'altitude du terrain de part et d'autre du mur. Cette erreur étant systématique sur l'ensemble des données, un traitement correctif a été appliqué dans la phase de prétraitement des données afin de réduire au maximum le biais résultant (cf. 3. Méthodologie). D'autres erreurs de classification moins fréquentes ou non systématiques n'ont pas été corrigées. Les objets concernés par une mauvaise classification sont listés dans le Tableau 3.

Catégories de biais		Biais	Correction appliquée
Contexte	Date de collecte	Tous les arbres ne sont pas en feuilles	✗
Qualité des données	Problèmes de classification	Murs de soutènement	✓
		Murs de séparation	✗
		Terasses	✗
		Fosses de chantier, cours intérieures étroites	✗
		Fontaines, hautes cheminées	✗
		Objets suspendus (lignes électriques, lampadaires suspendus, etc)	✗

Tableau 3 - Biais détectés dans les données LiDAR brutes.

Toutes ces erreurs de classification n'étaient pas attendues et non présentes dans les données LiDAR 2017. Le choix d'appliquer une correction à ces erreurs a représenté un important travail au détriment des objectifs initiaux du travail.

2. AUTRES DONNEES UTILISEES

En plus des données LiDAR, six couches ont été récupérées depuis les services en Open Data du SITG. Quelques modifications mineures ont été apportées aux couches avant leur utilisation dans les différentes étapes du traitement. Les couches utilisées sont visibles dans le Tableau 4 ci-dessous.

Organisme	Nom de la couche	Type	Description	Modifications	Utilisation	Mode de distribution	Métadonnées
DIT	CAD_BATIMENT_HORSOL_TOIT	Polygone	Emprises des toitures des bâtiments hors-sol	—	MNC_calculator.py	Open Data	https://ge.ch/sitg/fiche/0635
DIT	CAD_COMMUNES	Polygone	Périmètres des communes genevoises	Fusion des communes et buffer de 5m	MNC_calculator.py	Open Data	https://ge.ch/sitg/fiche/5661
DIT	CAD_OBJETDIVERS_POLY	Polygone	Objets divers de la couverture du sol	—	Murs_hauteur_2019.py	Open Data	https://ge.ch/sitg/fiche/1333
O CAN	FFP_CADASTRE_FORET	Polygone	Cadastre forestier	Buffer 10 et 15m	MNC_calculator.py	Open Data	https://ge.ch/sitg/fiche/2022
DT	MNA_TERRAIN_2019	Raster	Modèle Numérique de Terrain 2019	Rééchantillonnage en résolution 2m	MNC_calculator.py	Open Data	https://ge.ch/sitg/fiche/8821
VDG	SIPV_ICA_ARBRE_ISOLE	Point	Inventaire cantonal des arbres isolés	—	Comparaison/Ajout d'attributs	Open Data	https://ge.ch/sitg/fiche/4571

Tableau 4 - Données externes utilisées pour la réalisation du projet.

IV. Méthodologie

1. MODELE CONCEPTUEL GLOBAL



Figure 5 - Modèle conceptuel général.

La production de MNHC, MNC et de détection des arbres isolés sur le canton de Genève est le fruit de plusieurs processus utilisant les logiciels ArcGIS, FME, LAStools ainsi que les langages informatiques Python et Octave. La méthodologie peut être divisée selon 4 grandes étapes présentées sur la Figure 5. Comme indiqué précédemment ce stage s'est déroulé conjointement avec le stage de Gilles Gay des Combes. Afin d'améliorer la compréhension globale du travail, la méthodologie présentée ci-dessous détaille l'ensemble des étapes du projet de ces deux stages. N'ayant pas participé à la réalisation de la dernière étape (4. Ajout d'attributs au MNC), ce rapport ne s'attardera pas sur la présentation des résultats de cette partie. Pour plus d'informations concernant cette partie, se référer au rapport de Gilles Gay des Combes.

L'ensemble des outils et scripts utilisés lors de ce stage se trouvent en libre accès sur la VDGbox : <https://vdgbox.ville-geneve.ch/index.php/s/mqenx66CwosTfqN>. Des informations complémentaires relatives à l'environnement nécessaire ainsi qu'à leur utilisation sont fournies dans l'en-tête de chaque script.

2. PRETRAITEMENT DU JEU DE DONNEE LAS

a. Création d'un buffer autour des tuiles LAS (LAStools)

L'outil *LASclip.exe* (Figure 6) fait partie du logiciel LAStools développé par la firme rapidlasso GmbH. Il nécessite en entrée toutes les tuiles LAS ainsi que la couche vectorielle des polygones d'emprise des tuiles LAS auxquels on a ajouté un buffer de 10 m. L'outil sert à ajouter autour de chaque tuile LAS une zone tampon avec les points LiDAR de ses tuiles voisines ce qui augmente la superficie des tuiles LAS afin qu'elles se chevauchent les unes sur les autres. Cette zone tampon sera plus tard supprimée lors des prochains traitements. Cette étape permet d'éviter les effets de bord lors de l'assemblage des tuiles.

N.B. : La plupart des outils LAStools sont utilisables en libre accès et ne nécessitent pas de licences. Cependant, l'utilisation en libre accès sur des données volumineuses a comme conséquence une légère modification des coordonnées x, y et z des points LiDAR des tuiles en sortie. De plus, le traitement supprime les attributs Intensity, GPS Time, User Data et Point Source ID. L'achat d'une licence permet d'éviter ces problèmes. Dans notre cas, les changements des coordonnées étant de maximum 2 mm, et les attributs perdus étant non essentiels pour la suite, nous avons jugé acceptable d'utiliser l'outil sans acheter la licence.

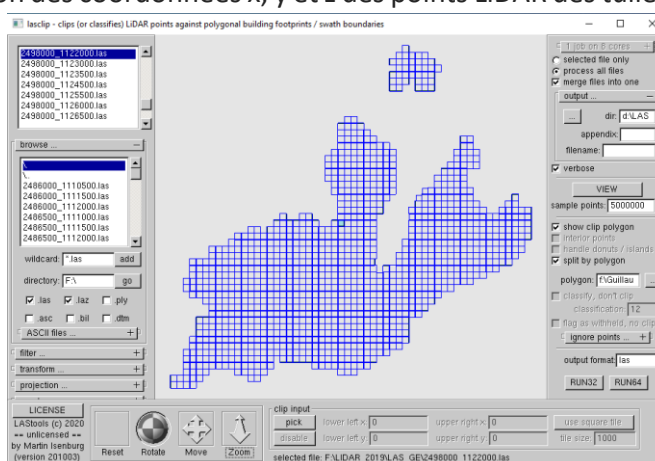


Figure 6 - Interface de l'outil LASclip.exe.

b. Détection des murs de soutènement faussement classifiés (ArcGIS, Python)

Les principaux murs concernés par une fausse classification dans les données LiDAR sont les murs de soutènement. Ces derniers ont pour but principal de stabiliser un terrain en pente et sont caractérisés par une différence d'altitude de part et d'autre du mur. La couche vectorielle du cadastre des objets divers (*CAD_OBJETDIVERS_POLY.shp*) ne fait pas la distinction entre les types de murs, il a donc fallu isoler la catégorie concernée. Le choix a été fait de sélectionner les murs répertoriés dans le cadastre des objets divers et ayant une pente de plus de 50°. La création de la couche *Hmurs_2019* s'est donc déroulée en deux étapes : 1) la localisation des murs de soutènement qui a été faite manuellement sur ArcGIS Pro et 2) l'estimation de leur hauteur à l'aide d'un script Python automatisé (*Hauteurs_murs_2019.py*). La couche résultante sera par la suite utilisée comme input dans le modèle FME *LAS_Pretraitement.fmw*.

1) Localisation des murs de soutènement (ArcGIS Pro)

Dans la couche du cadastre des objets divers, la catégorie « murs » a été sélectionnée. En parallèle, les pentes ont été calculées à partir d'une couche raster MNT présente sur la SDE (*MNA_TERRAIN_2019*) et recouvrant tout le canton de Genève. Les pixels (50 cm) d'une valeur de plus de 50° ont été isolés et vectorisés. Cette couche a ensuite été utilisée pour clipper la sélection des murs. Un buffer de 50 cm a finalement été ajouté. L'output de cette étape est un shapefile contenant des polygones d'emprise des murs situés sur une pente de plus de 50° et entourés d'un buffer de 50 cm.

2) Estimation de la hauteur des murs (*Murs_hauteur_2019.py*)

À partir des points de sol (classe 2) et d'eau (classe 9) des tuiles LAS, deux MNT de résolution de 5 m ont été créés. Le 1er (MNTmax) avec comme ordre de choisir la valeur d'altitude la plus élevée. Le 2ème (MNTmin) avec comme ordre de choisir la valeur d'altitude la plus basse. Cela permet que dans le cas d'un pixel localisé sur un mur de soutènement, le MNTmax ait comme valeur l'altitude du sommet du mur, et le MNTmin celle de sa base. Les surfaces planes auront au contraire la même valeur dans les deux MNT. La différence entre les deux MNT permet de déterminer la hauteur des murs de soutènement. La hauteur totale des murs est volontairement surestimée de 2 m afin d'être certain que la couche « englobe » la totalité du mur réel. Les valeurs de hauteur du raster sont ensuite extraites en fonction de l'emprise vectorielle des murs créée à l'étape 1. Cette étape est réalisée en boucle sur les tuiles LAS contenant un mur qui sont ensuite assemblées pour obtenir un raster couvrant toute l'étendue du canton de Genève.

On obtient en sortie une couche raster de résolution de 0,4 m de l'emprise des murs situés sur une pente de plus de 50° et avec comme valeur de pixel la hauteur surestimée de 2 m. Les pixels ne correspondant pas à l'emprise d'un mur sont des NoData. Le script *Murs_hauteur_2019.py* tourne sur la version 2.7 de Python et fait appel aux bibliothèques *arcpy*, *os* et *time*.

c. Correction et allègement des données LAS (FME)

Le modèle *LAS_Pretraitement.fmw* (Figure 7) créé sur FME Workbench 2020.1 sert à prétraiter les données LAS afin de les optimiser pour les prochains traitements. Le modèle est lancé de manière automatisée avec l'option Batch Deploy⁶ et applique le prétraitement successivement sur chaque tuile LAS. Il nécessite deux fichiers en entrée : la tuile LAS à traiter, à laquelle on a préalablement ajouté un buffer de 10 m avec l'outil *LASclip.exe*, ainsi que la couche raster des hauteurs de murs *Hmurs_2019.tif* créée précédemment avec ArcGIS et *Hauteurs_murs_2019.py*. Le modèle est divisé en deux parties : 1) le calcul de la hauteur et la sélection des points de végétation (classe 5) et 2) la correction des points de murs faussement classifiés comme végétation.

1) Calcul de la hauteur et sélection des points de végétation. (Figure 7)

D'abord un MNT est calculé à l'aide des points de sol (classe 2) et d'eau (classe 9). En parallèle, les points de végétation (classe 5) sont isolés et on ajoute à chacun un nouvel attribut *z_ground* auquel on attribue la valeur du MNT correspondant à l'altitude du sol sous le point de végétation. Ensuite, on utilise la valeur de *z_ground* pour transformer la valeur d'altitude de l'attribut *z* en valeur de hauteur avec l'expression « $z = z - z_ground$ ». Un dernier filtre est appliqué pour supprimer les points ayant une hauteur *z* négative ou aberrante de plus de 60 m.

2) Correction des murs. (Figure 7)

On ajoute aux points de végétation un nouvel attribut *h_mur* auquel on assigne la valeur de la couche *Hmurs_2019.tif* correspondant à la hauteur du mur dans le cas où le point se situe au-dessus de l'emprise d'un mur. Si le point se situe sur une autre partie du raster, la valeur NoData lui est attribuée. Ensuite l'ensemble des points dont la valeur de hauteur *z* est inférieure à leur valeur *h_mur* sont éliminés, car ils sont considérés comme étant un mur mal classifié et non pas de la végétation.

On obtient en sortie des tuiles LAS allégées, avec uniquement des points de végétation ayant dans l'attribut *z* une valeur de hauteur et nettoyées des points de murs faussement classifiés comme végétation. Le modèle ne modifie pas les tuiles LAS en entrée, mais en crée des nouvelles.

⁶https://docs.safe.com/fme/html/FME_Desktop_Documentation/FME_Workbench/Workbench/Running_a_Batch_Translation.htm

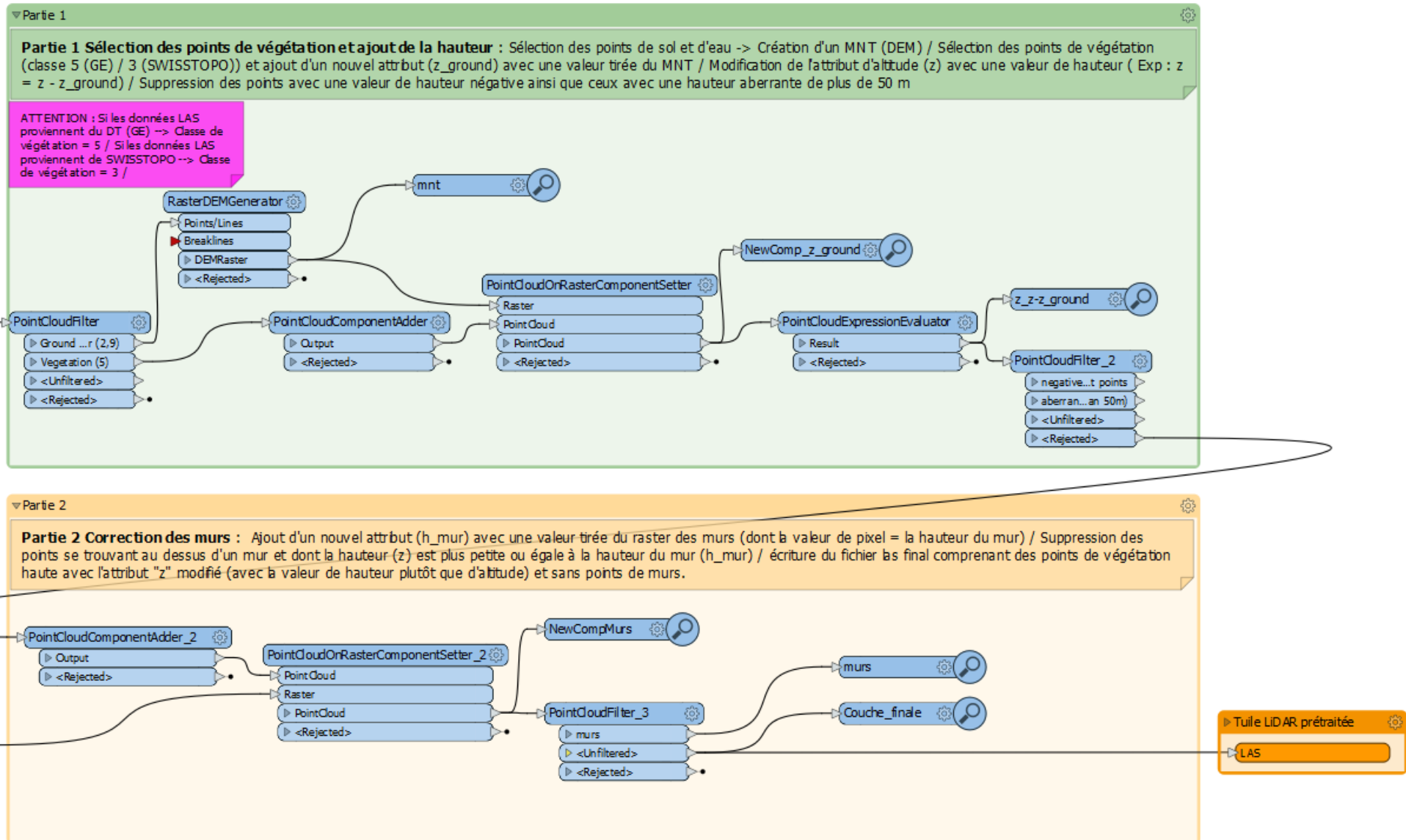
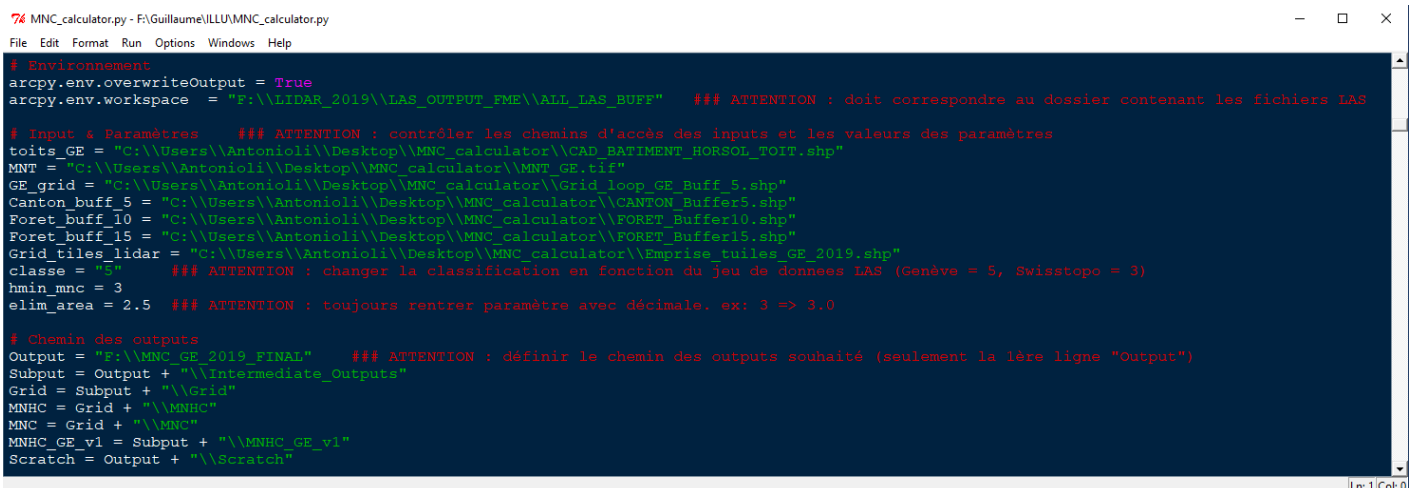


Figure 7 – Workbench FME du modèle LAS_Pretraitement.

3. PRODUCTION DES MODELES NUMERIQUES DE CANOPEE (PYTHON)

La production du MNC est le processus central de ce stage. Sur la base des tuiles LiDAR prétraitées (Prétraitement du jeu de donnée LAS), l'ensemble des étapes de production du MNC (ainsi que du MNHC) sont contenues dans le script Python *MNC_calculator.py*. Ce script de 700 lignes fait principalement appel à des fonctions ArcGIS, il requiert les librairies os, glob, arcpy, time et datetime ainsi qu'un environnement Python 2.7. Il est organisé en 6 parties détaillées ci-dessous. Les chemins d'accès des inputs et outputs ainsi que les paramètres sont à définir dans la première partie du script (Figure 8).



```
74 MNC_calculator.py - F:\Guillaume\ILLU\MNC_calculator.py
File Edit Format Run Options Windows Help

# Environnement
arcpy.env.overwriteOutput = True
arcpy.env.workspace = "F:\LIDAR_2019\LAS_OUTPUT_FME\ALL_LAS_BUFF"  ### ATTENTION : doit correspondre au dossier contenant les fichiers LAS

# Input & Paramètres  ### ATTENTION : contrôler les chemins d'accès des inputs et les valeurs des paramètres
toits_GE = "C:\Users\Antonioli\Desktop\MNC_calculator\CAD_BATIMENT_HORSOL_TOIT.shp"
MNT = "C:\Users\Antonioli\Desktop\MNC_calculator\MNF_GE.tif"
GE_grid = "C:\Users\Antonioli\Desktop\MNC_calculator\Grid_loop_GE_Buff_5.shp"
Canton_buff_5 = "C:\Users\Antonioli\Desktop\MNC_calculator\CANTON_Buffer5.shp"
Foret_buff_10 = "C:\Users\Antonioli\Desktop\MNC_calculator\FORET_Buffer10.shp"
Foret_buff_15 = "C:\Users\Antonioli\Desktop\MNC_calculator\FORET_Buffer15.shp"
Grid_tiles_lidar = "C:\Users\Antonioli\Desktop\MNC_calculator\Emprise_tuiles_GE_2019.shp"
classe = "5"  ### ATTENTION : changer la classification en fonction du jeu de donnees LAS (Genève = 5, Swisstopo = 3)
hmin_mnc = 3
elim_area = 2.5  ### ATTENTION : toujours rentrer paramètre avec décimale. ex: 3 => 3.0

# Chemin des outputs
Output = "F:\MNC_GE_2019_FINAL"  ### ATTENTION : définir le chemin des outputs souhaité (seulement la 1ère ligne "Output")
Subput = Output + "\\Intermediate_Outputs"
Grid = Subput + "\\Grid"
MNHC = Grid + "\\MNHC"
MNC = Grid + "\\MNC"
MNHC_GE_v1 = Subput + "\\MNHC_GE_v1"
Scratch = Output + "\\Scratch"
```

Figure 8 - Aperçu des paramètres du script *MNC_calculator.py*.

a. Rastérisation et création du MNHC

La première étape consiste en la rastérisation des données LiDAR. Pour cela une boucle est générée et appliquée sur chaque tuile LAS. Les données LAS ne contenant que les points de végétation avec des valeurs de hauteur recalculées, leur rastérisation revient à la création de la première version du Modèle Numérique de Hauteur de Canopée (MNHC).

b. Découpe du MNHC selon une grille définie

Cette étape correspond à la préparation des données pour la suite du traitement. En effet, afin que le script puisse fonctionner sans être trop gourmand en ressources, il est nécessaire de subdiviser le MNHC du canton de Genève en plusieurs sous-parties. Ainsi lors de cette étape, le MNHC est découpé selon une grille vectorielle (*Grid_GE*) qui doit être fournie en entrée. Dans notre cas un découpage selon une grille de 9 km² était optimal. Afin d'empêcher des effets de bords lors de l'assemblage final, un buffer de 10 m est appliqué à cette grille.

Une boucle est ensuite créée sur l'ensemble des zones produites par le découpage du MNHC par cette grille. Les étapes c., d. et e. sont effectuées à l'intérieur de cette boucle.

c. Vectorisation et création du MNC

Lors de cette étape, le raster MNHC est vectorisé. Pour cela, il est tout d'abord transformé en raster « booléen ». L'ensemble des valeurs NoData et des valeurs inférieures à la hauteur minimale paramétrée en entrée dans le script (3 m dans notre cas) sont converties en valeur 0. Au reste est attribuée la valeur 1. Une fois vectorisé, plusieurs nettoyages successifs sont nécessaires afin d'éliminer les polygones d'une surface inférieure à la valeur paramétrée en entrée (2,5 m² dans notre cas). Enfin, la couche ne comprenant que les polygones de végétation est extraite. C'est la première version du Modèle Numérique de Canopée.

d. Correction du MNHC

Les étapes d. et e. de ce script, permettent de corriger les valeurs de MNHC et ensuite du MNC pour la végétation se trouvant sur les bâtiments. En effet, la hauteur de végétation présente sur les toits est exagérée par la hauteur des bâtiments. Afin de savoir si une correction est nécessaire (sur une zone donnée), la couche du MNC est comparée à la couche des toits du cadastre (CAD_BATIMENT_HORSOL_TOIT). Si au moins un centre de polygone du MNC intersecte un objet de la couche des toits, alors les corrections sont appliquées.

La hauteur des bâtiments où de la végétation est présente sur le toit est calculée en faisant la différence d'altitude entre le toit et le MNT à l'aide de l'outil *ZonalStatisticsAsTable*. Les hauteurs de bâtiments calculées dont les valeurs sont négatives sont remplacées par 0. La sélection de polygones de toits est ensuite rastérisée sur la base de leur valeur de hauteur puis le MNHC est corrigé par la soustraction des hauteurs de bâtiments. Les pixels se voyant attribuer une valeur négative sont assignés NoData.

e. Création du MNC corrigé

Le MNHC étant corrigé pour la végétation présente sur les bâtiments, le nouveau MNC est produit en reprenant la procédure de l'étape c. Enfin, une dernière étape de simplification des polygones est réalisée afin d'alléger et d'optimiser la couche finale. Une valeur de tolérance de simplification de 2,5 m a été sélectionnée comme bon compromis. Pour les zones ne nécessitant pas de correction, les premières versions de MNHC et MNC sont conservées. La simplification finale du MNC est aussi appliquée.

f. Assemblage du MNHC et MNC sur l'ensemble du Canton

Cette dernière partie correspond à l'assemblage de l'ensemble des zones pour réaliser les produits finaux.

Les différentes parties de MNHC sont assemblées dans un nouveau raster puis découpées par les limites du canton avec un buffer de 5 m afin de produire le *MNHC_GE.shp*.

Les différentes parties de MNC sont assemblées dans un nouveau vecteur puis dissoutes. Le MNC est découpé par les limites du canton avec un buffer de 5 m afin de représenter l'ensemble du couvert végétal du canton : *MNC_GE.shp*.

Le cadastre des forêts (buffer de 10 m) est utilisé afin de supprimer l'ensemble des zones forestières du MNC_GE. Une étape de nettoyage des reliquats de parcelle de forêt est appliquée : *MNC_GE_noforest.shp*.

4. DETECTION DE L'EMPLACEMENT DES TRONCS

La création de la couche de points de l'emplacement des troncs d'arbres se fait en deux étapes séparées. Elle fait appel à des scripts Octave de l'outil *Digital-Forestry-Toolbox*⁷ de Matthew Parkan afin de réaliser les détections des troncs et à des outils ArcGIS afin d'assembler la couche et de la corriger pour l'ensemble du canton. Il n'était pas possible d'exécuter un script faisant appel à ces deux outils, car ils demandent des environnements spécifiques distincts.

a. Détection des troncs selon différents paramètres (Python/Octave)

La détection des troncs d'arbres se fait à l'aide du script Python *Stems_detection.py* couplé au script octave *Stems_arg.m*. Ces scripts permettent de choisir les paramètres de détection et d'appeler des fonctions octaves dérivées de l'outil de M. Parkan : *LASread.m*, *treeStems.m*, *rasterize.m* et *Shapewrite.m*. Octave doit donc être installé sur la machine ainsi que l'outil Digital-Forestry-Toolbox. Pour exécuter, le script Python il faut un environnement Python et des bibliothèques spécifiques : *numpy*, *scipy*, *oct2py*. Afin de simplifier et d'accélérer le processus, le programme *Anaconda*⁸ a été utilisé pour installer ces bibliothèques dans un environnement Python 3.5. L'utilisation est détaillée étape par étape dans l'en-tête du script *Stems_detection.py*.

⁷ <https://mparkan.github.io/Digital-Forestry-Toolbox/>

⁸ <https://conda.io/projects/conda/en/latest/user-guide/index.html>

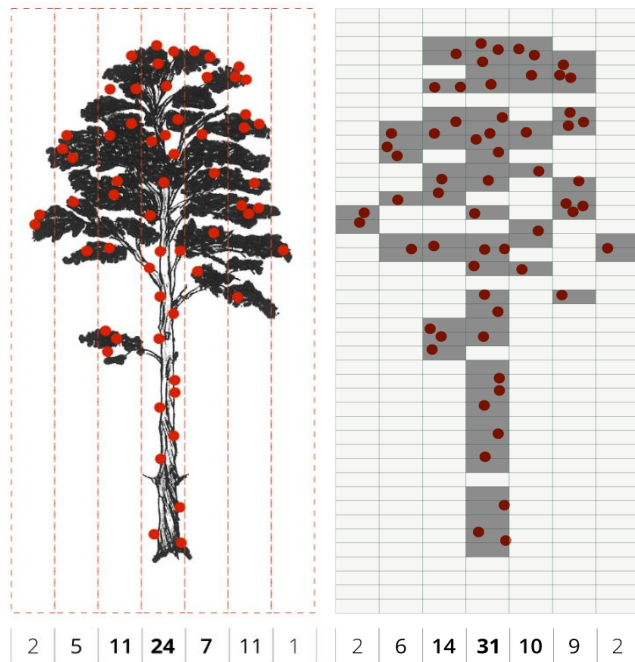


Figure 9 - Illustration de la méthode de détection des troncs. a) Densité de retour LiDAR plus élevée le long du tronc. b) Détection du maximum local de densité sur la somme de cellules possédant un retour LiDAR.

Source : <https://mparkan.github.io/Digital-Forestry-Toolbox/tutorial-3.html>

La méthode de détection des troncs d'arbres est basée sur une observation simple : la densité (verticale) de points LiDAR est de plus en plus élevée à mesure que l'on s'approche du tronc. Afin d'homogénéiser les données et de simplifier la détection des pics de densité, les données LAS sont rastérisées en plusieurs couches d'une résolution et d'une épaisseur paramétrable. La Figure 9 illustre la méthode de détection des maximums locaux de densité. Le maximum de densité n'est pas calculé directement sur le nombre de retours LiDAR, mais sur le total de couches raster ayant au moins un retour (Figure 9).

Le script *Stems_detection.py* est composé de 2 boucles successives sur l'ensemble des tuiles LiDAR permettant de choisir un jeu de paramètre par boucle. Chaque boucle appelle la fonction octave *Stems_arg.m* afin de traiter les tuiles avec les paramètres définis dans la boucle. Cette étape permet de détecter les troncs selon deux jeux de paramètres indépendants. Un fichier shapefile est généré par tuile LiDAR et

Paramètre	Description	Explications supplémentaires
returnMethod	Type de retour LiDAR considéré pour la détection des arbres	'all_return' permet de détecter les petits arbres 'last_return' est plus précis dans les massifs arborés
cellSize	Taille de la cellule de raster (m)	
bandWidth	Épaisseur de chaque couche de raster horizontal (m)	Diminuer pour augmenter la détection des petits arbres.
verticalStep	Distance verticale entre chaque couche raster (m)	Valeur initiale de l'outil Digital-Forestry-Toolbox
searchRadius	Le rayon de recherche utilisé pour détecter un maximum local de densité	Rayon de recherche proportionnel à la taille de cellule. Ex: cellSize 2m & searchRadius 3 => Rayon de recherche de 6m
minLength	Le compte minimum de cellules verticales pour qu'un maximum local soit considéré comme un tronc	Hauteur minimum proportionnelle à l'épaisseur du raster. Ex: bandWidth 0.5m & minLength 3 => hauteur minimum de détection de 1.5m

Tableau 5 - Paramètres du script *Stems_detection.py*

par méthode de détection. Les paramètres sont détaillés dans le Tableau 5.

Les paramètres de la première boucle (*Stems1*) sont sélectionnés dans l'optique de détecter les petits et très petits arbres. Les paramètres de la seconde boucle (*Stems2*) peuvent être choisis afin d'être plus réalistes dans le cas de massifs arborés.

b. Création de la couche de position des troncs (Python/ArcGIS)

La couche de position des troncs est produite à l'aide du script *Stems_correction.py*. Le script est composé de trois parties. Les 1^{ère} et 2^{ème} parties sont similaires. Elles sont constituées d'une boucle sur l'ensemble des couches de troncs pour chacun des deux jeux de données en entrée (*Stems1* & *Stems2*). Chaque couche est découpée par l'empreinte de la tuile LiDAR correspondante puis nettoyée des points d'arbres de moins de 3 m de hauteur (conformément à notre MNC). Enfin la couche est allégée des points d'arbres détectés en dehors des polygones de la couche MNC. Une couche de troncs de plus de 3 m pour l'ensemble du canton pour chaque jeu de donnée est générée et stockée en tant que produit intermédiaire (*Stems1_GE.shp* & *Stems2_GE.shp*).

La 3^{ème} partie correspond à la création de la couche de troncs finale. Cette dernière est une réalisation hybride entre les deux couches de troncs produites précédemment (*Stems_GE_mix.shp*). Les deux couches de troncs sont combinées en fonction d'une valeur de hauteur seuil paramétrable (dans notre cas 14 m). Les troncs de la couche *Stems1_GE* sont injectés seulement si leur attribut de hauteur est plus petit que la valeur seuil tandis que les troncs de la couche *Stems2_GE* ne sont gardés que s'ils sont supérieurs à la valeur de hauteur. L'introduction d'une valeur seuil est un choix subjectif visant à combiner une méthode de détection plus précise (rayon de recherche réduit) pour les petits arbres et donc préserver leur détection tout en évitant de trop surestimer le nombre de troncs dans les grands massifs arborés.

5. AJOUT D'ATTRIBUTS AU MODELE NUMERIQUE DE CANOPEE

Cette partie a été réalisée par Gilles Gay des Combes. La description de la méthode est néanmoins décrite à titre informatif. Pour plus d'informations, se référer au rapport de Gilles.

Le script Python *MNC_Attributes.py* sert à ajouter des attributs dans la table de la couche du MNC créée par *MNC_Calculator.py*. En addition au MNC, il nécessite la couche raster du MNHC, la couche ICA ainsi que la couche shapefile des troncs en entrée. Les attributs ajoutés sont :

- 1) **Les hauteurs maximale, minimale et moyenne** pour chaque polygone de canopée. Les valeurs de hauteur sont extraites du MNHC.
- 2) **Le nombre de troncs** intersectant chaque polygone de canopée.
- 3) **Le nombre de points ICA** intersectant chaque polygone de canopée.
- 4) **L'Indice de Miller**, indicateur de circonvolution :

$$I_{Miller} = \frac{4\pi A}{P^2}$$

$A = \text{Aire} / P = \text{Périmètre}$

Plus I_{Miller} s'approche de 1, plus le polygone est de forme circulaire.

5) **Le Shape Index**, indicateur du degré de réticulation :

$$I_{Shape} = \frac{P}{2\sqrt{\pi A}}$$

$A = \text{Aire} / P = \text{Périmètre}$

Un I_{Shape} de 1 décrit un disque parfait. Plus l'objet est réticulé plus I_{Shape} est grand.

De plus, l'attribut « GRIDCODE » qui est un vestige d'un traitement de [MNC_Calculator.py](#) et supprimé, car il n'apporte aucune information utile à la couche du MNC. Le script tourne sur la version 2.7 de Python et fait appel aux librairies arcpy, os et time.

V. Résultats

1. PRETRAITEMENTS

La Figure 10 illustre les étapes de correction de la tuile LAS de la zone du parc des Bastions. Les points mal classifiés sont bien détectés et supprimés (Figure 10). L'ensemble des points de classes autres que végétation haute sont retirés et la hauteur des points de végétation est calculée (Figure 10). Le jeu de données en ressort considérablement allégé et optimisé pour la suite des traitements (Tableau 6).

	Poids total (Go)	Moyenne de points par tuile	Tuile avec le plus de points	Tuile avec le moins de points
Jeu de donnée LiDAR brut	323	10 Millions	23 168 697	287 490
Après prétraitement	97	3 Millions	16 760 002	39

Tableau 6 - Comparaison du jeu de donnée LiDAR avant et après traitement.

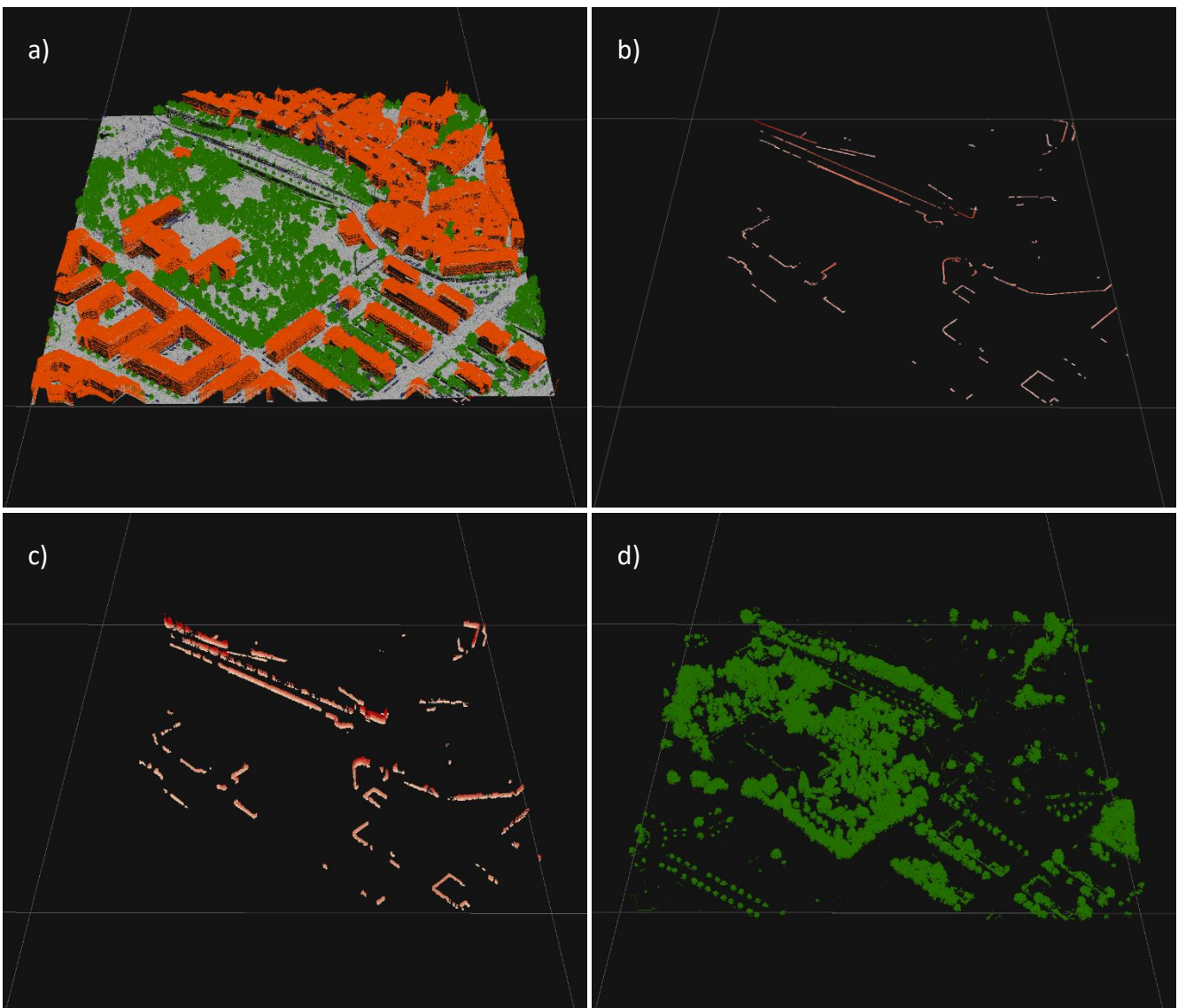


Figure 10 – Exemple au parc des Bastion des différentes étapes de prétraitements des données LAS. a) LAS brut ; b) Couche raster des murs à corriger ; c) Ensembles des points de végétation mal classifiés qui sont éliminés ; d) Nuage de point en sortie, allégé et corrigé.

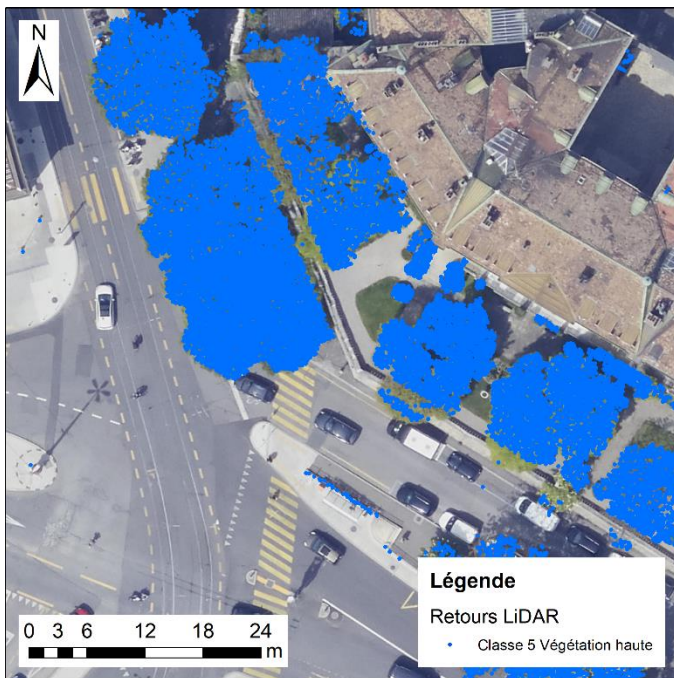


Figure 11 - Retours LiDAR de végétation supprimés au-dessus du mur de la treille.

Néanmoins la correction n'est pas parfaite. Quelques résidus subsistent autour de murs plus larges que la zone corrigée. En outre, certains endroits de réelle canopée au-dessus de murs de soutènement se retrouvent supprimés, on observe alors une canopée « découpée » par les murs sous-jacents. Cependant, cela n'est le cas que pour des murs de grande hauteur, par exemple le mur de la Treille sur la Figure 11.

Hormis ces exceptions localisées, l'étape d'élimination des points LiDAR de murs de soutènement mal classifiés est pleinement satisfaisante et améliore la confiance dans les produits finaux.

2. MODELE NUMERIQUE DE HAUTEUR DE CANOPEE

Le MNHC reflète la rasterisation des fichiers LiDAR prétraités et ne comprenant que la végétation. Il n'y a pas de limite de hauteur minimum appliquée au MNHC, ainsi il correspond à l'ensemble des retours de végétation des données LiDAR. Aucune méthode de remplissage du vide n'est donc nécessaire, ce qui évite les erreurs d'interpolation. Les bâtiments n'étant pas considérés dans la réalisation du MNHC, il n'y a donc pas de découpage des arbres par ces derniers comme illustré sur la Figure 12.

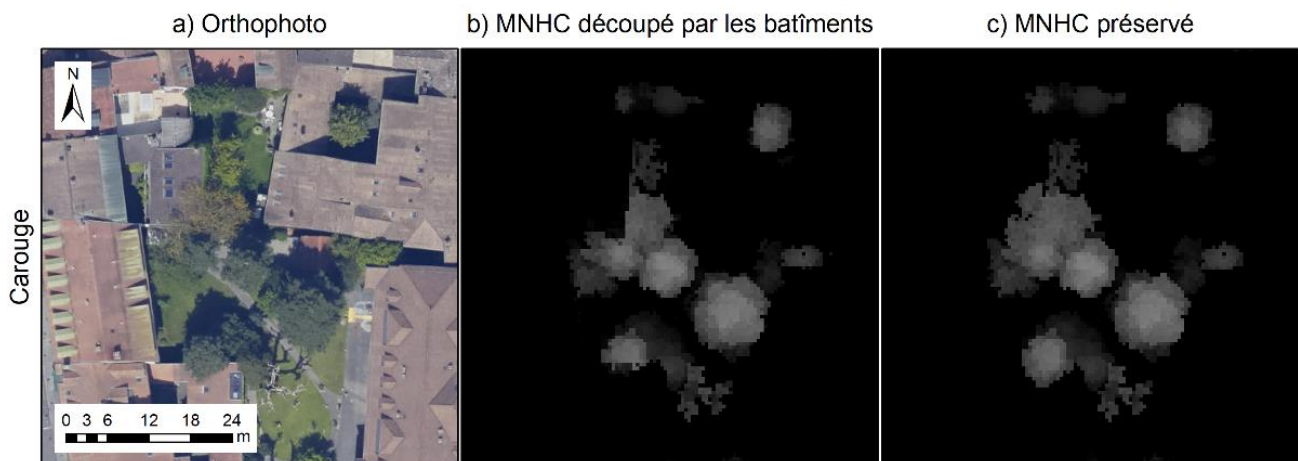


Figure 12 - Exemple à Carouge de la préservation des hauteurs des canopée avec cette méthode.
a) Orthophoto ; b) MNHC avec méthode traditionnelle (MNS-MNTB) ; c) MNHC avec méthode actuelle

Cependant avec cette méthode les valeurs de hauteurs de végétation se trouvant sur des toits se retrouvent exagérées. Une correction est donc nécessaire. La Figure 13 présente l'application de cette correction. Les valeurs de hauteur des pixels de végétation se trouvant sur les toits sont en grande majorité bien corrigées. Néanmoins, il est toujours possible d'observer des pixels non corrigés (en bleu foncé), car débordant du cadastre du toit (Figure 13b). Les plus grandes valeurs de hauteur du MNHC sont donc majoritairement occupées par ces pixels qui ont échappé à la correction. Dans un souci d'allègement de la couche raster, le MNHC comprend des valeurs NoData pour l'ensemble des pixels attribués 0.

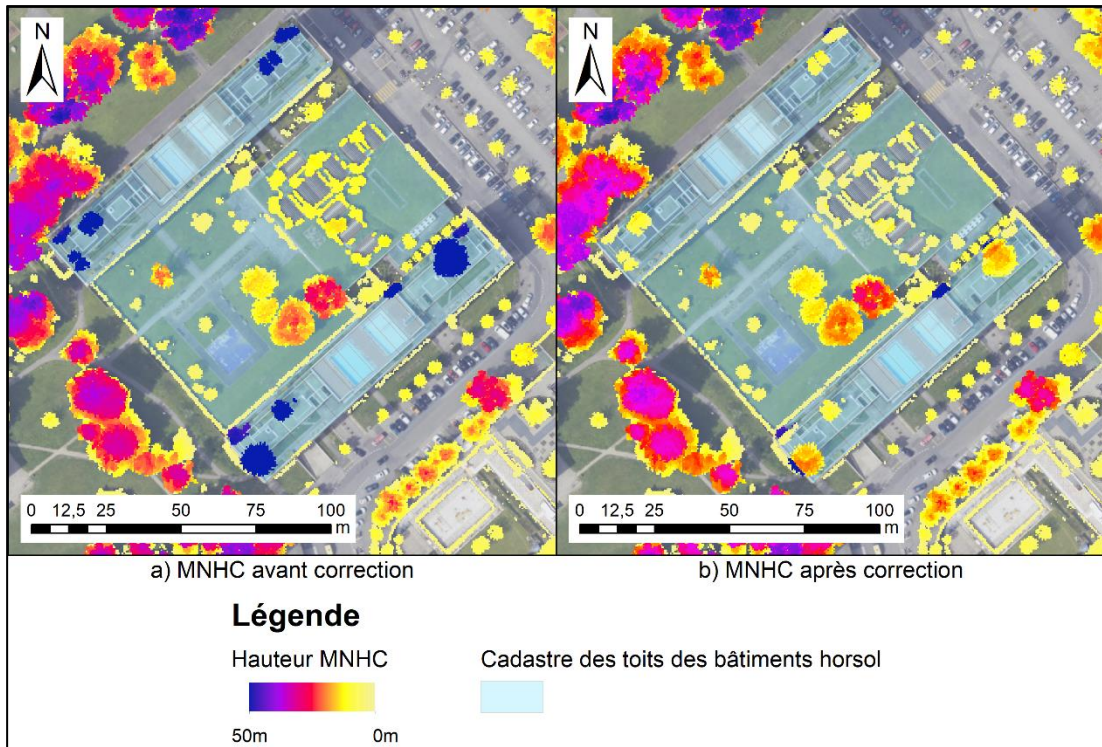


Figure 13 - Illustration de la correction des hauteurs de MNHC.
 a) MNHC brut ; b) MNHC avec valeurs de hauteur corrigées par les bâtiments

3. MODELE NUMERIQUE DE CANOPEE

Dérivé directement de la vectorisation du MNHC, le MNC produit ici correspond à la végétation de plus de 3 m de hauteur et d'au moins 2,5 m² de surface. La hauteur minimale de 3 m est le seuil généralement accepté dans la création des MNC sur le canton de Genève. Celle-ci permet d'éviter qu'une majorité de haies soient comptabilisées comme de la canopée. Concernant le minimum de surface de 2,5 m², celui-ci a un double usage. Il correspond à la surface minimale pour qu'un polygone de canopée et/ou un polygone sans végétation soient préservés. Ce paramètre permet donc de combler le mitage dans la canopée tout en supprimant les petits polygones de canopée isolés. Le nettoyage des petits polygones ainsi qu'une dernière simplification des polygones permet d'épurer la couche ainsi que de casser les angles des polygones afin d'être plus représentatifs d'un feuillage. Ces étapes permettent d'optimiser grandement l'affichage de la couche de MNC comme illustré sur la Figure 14.

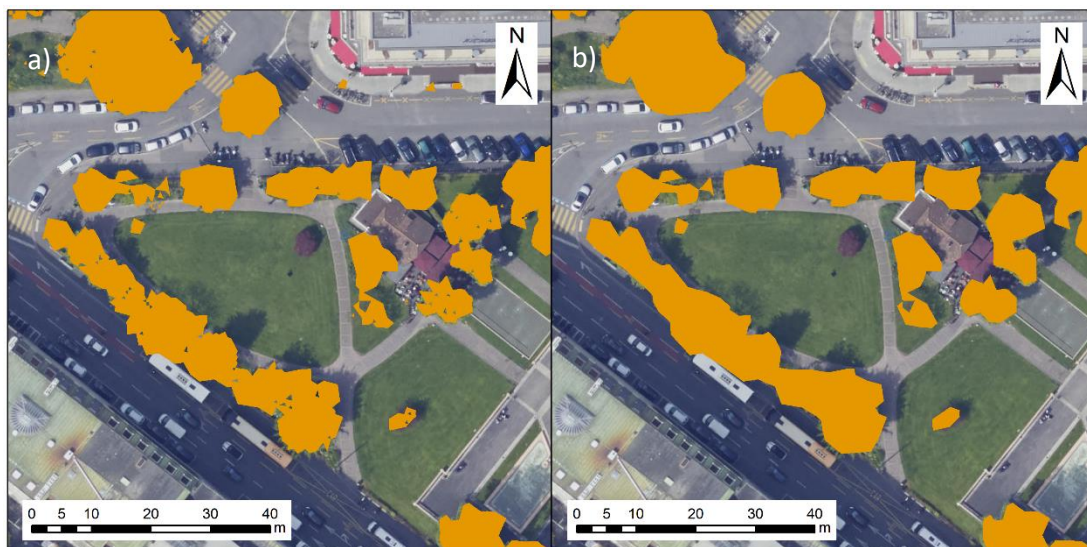


Figure 14 – Exemple au parc du monument Brunswick de l'optimisation du MNC.
a) MNC brut ; b) MNC éliminé des petits polygones et simplifié

Deux modèles sont produits, le MNC global du canton et un MNC sans les parcelles de forêt. Les deux modèles sont présentés sur la Figure 15. Les MNC produits dépassent volontairement de 5 m le long des frontières du canton afin de préserver la structure des arbres en bordure. Les caractéristiques de ces MNC sont détaillées dans le Tableau 7, dans ce cas la surface de canopée correspond à la canopée effectivement présente sur le canton, c'est-à-dire sans le buffer de 5m. Après suppression du MNC présent sur le cadastre de forêt, on constate une diminution logique du nombre de polygones (Tableau 7). Cependant, cette diminution devrait être encore plus marquée, car le découpage par le cadastre de forêt crée de nombreux polygones résiduels. En effet, le cadastre de forêt correspond à une emprise au sol et ne correspond pas forcément à la bordure du canton ni à la canopée effective. Malgré un buffer de 10 m au cadastre forêt et un nettoyage supplémentaire des polygones entièrement compris dans une zone tampon de 15 m, il

persiste des reliquats de MNC de parcelle de forêt. Plusieurs reliquats sont visibles sur la Figure 16 par exemple de part et d'autre des arbres bordant le cours de l'Arve.

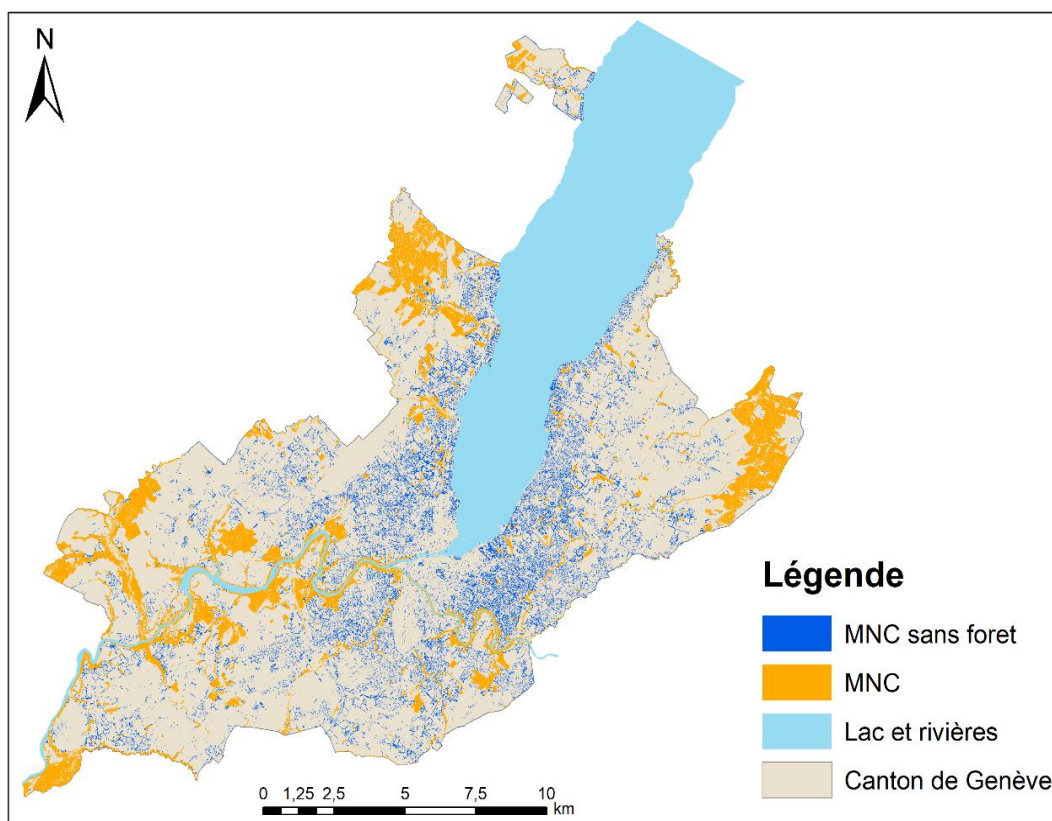


Figure 15 - Aperçu à l'échelle du canton de Genève des deux MNC produits.

	Surface (ha)	Pourcentage de recouvrement du canton	Nombre de polygones	Aire moyenne par polygone (m ²)
MNC	5 441,10	22,1%	196 458	278,8
MNC sans forêt	2 240,60	9,1%	184 106	122,3
Canton hors Lac	24 575,60	-	-	-

Tableau 7 - Comparaison des deux MNC.

Enfin, la qualité générale du MNC dépend directement de la qualité de la classification du LiDAR. Ainsi de nombreux polygones de MNC ne correspondent pas à de la vraie végétation. Ils correspondent aux biais détectés dans la partie présentation des données LiDAR (Tableau 3).

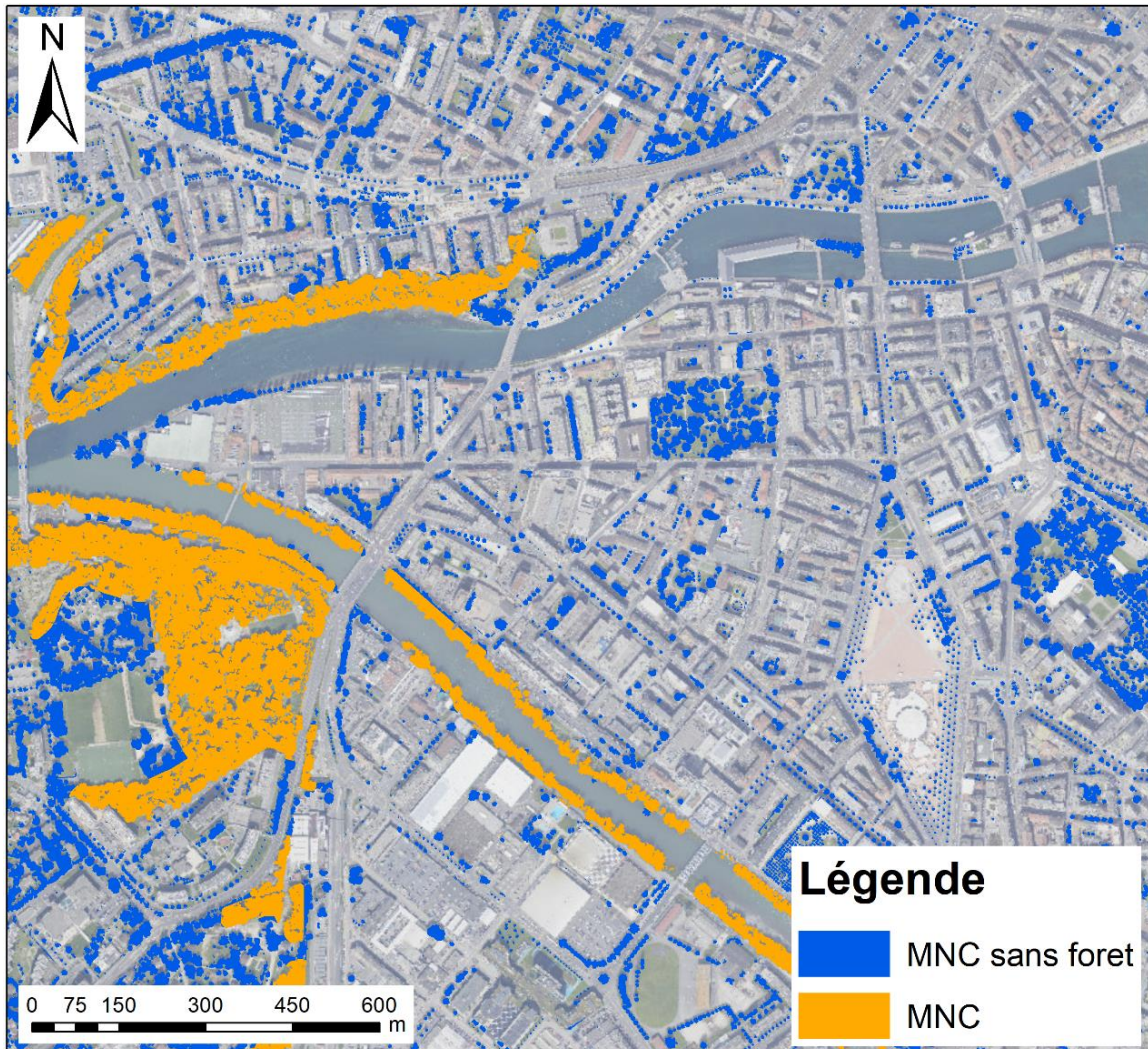


Figure 16 - Zoom sur MNC en centre-ville de Genève (Jonction).

4. EMBLACEMENT DES ARBRES ISOLÉS

Afin de déterminer les valeurs des paramètres pour la détection des troncs d'arbres, plusieurs étapes ont été nécessaires. Tout d'abord une zone d'étude bien renseignée par ICA a été sélectionnée. L'idée étant de tester plusieurs détections dans cette zone et de comparer les résultats avec l'inventaire afin de déterminer la meilleure valeur de paramètre. Une partie du parc des Bastions a donc été choisie comme zone d'étude (Figure 17). Cette zone présente un patrimoine arboré plutôt hétérogène avec des alignements d'arbres isolés sur les trottoirs et dans le parc ainsi que des massifs d'arbres de différentes hauteurs. La couche ICA de cette zone est très bien renseignée, elle contient 207 arbres. La précision s'avère même un peu trop élevée pour la comparaison avec notre méthode de détection. Ainsi, un nettoyage de certains points ICA a été appliqué afin d'enlever une majorité de thuyas, hamamélis et viornes présents sous la canopée d'arbres plus imposants. De plus, 5 points d'arbres supplémentaires ont été rajoutés à la couche grâce aux orthophotos et à l'outil googlemap streetview. La couche ICA mise à jour compte ainsi 184 arbres.

Les paramètres `cellSize`, `bandWidth` et `verticalStep` ont donc été calibrés dans cette zone. Le Tableau 8 présente pour différentes valeurs de ces paramètres le nombre d'arbres détectés dans la zone. Afin de simplifier la comparaison et de faire ressortir l'implication de chaque paramètre, plusieurs détections ont été produites en ne changeant qu'un seul paramètre, l'ensemble des autres paramètres ont été fixés (cellules vertes Tableau 8). Par exemple, les différentes valeurs de `bandWidth` ont été évaluées avec des paramètres similaires, ainsi la variation du nombre de troncs correspond uniquement à l'implication du paramètre `bandWidth`.

Pour le paramètre `cellSize`, on observe une très forte variation du nombre de troncs. Plus la taille de cellule est grande, plus le nombre de troncs augmente. Cela s'explique par le fait que plus la taille de cellule est petite, moins la différence de densité de retours LiDAR entre les cellules est marquée, ainsi la probabilité de détecter un maximum local diminue.

La valeur de `bandWidth` est inversement corrélée au nombre de troncs détectés. En diminuant l'épaisseur des bandes raster, cela implique une augmentation du nombre de couches raster du nuage de points. Ainsi, la probabilité d'avoir au moins 2 cellules superposées possédant des retours LiDAR augmente et avec elle la probabilité de détecter un tronc. Néanmoins, la variation du nombre de troncs est moins marquée qu'avec le paramètre `cellSize`.

Enfin le `verticalStep` est positivement corrélé avec le nombre de troncs détectés. Les implications de ce paramètre sont les moins évidentes, ainsi la valeur d'origine de l'outil a été conservée.

Les valeurs finales sélectionnées sont donc `cellSize` = 1 m, `bandWidth` = 0,5 m et `verticalStep` = 0,25 m. Avec ces paramètres, l'outil détecte 202 troncs d'arbres dans cette zone. Il est important de noter que dans cette zone d'étude, il existe des murs classifiés en végétation haute dans les données LiDAR. Sur chaque détection réalisée, il y a en

moyenne une dizaine de faux troncs qui sont localisés sur ces murs (Façade sud-est de la bibliothèque de Genève Figure 17). En faisant abstraction de ces troncs, ces paramètres permettent une détection bien représentative du nombre d'arbres ICA de cette zone.

		cellSize			ICA corrigée
Valeur	0.5 m	1 m*	1.5 m		
Nombre de troncs	96	202	265		
		bandWidth			
Valeur	0.25 m	0.5 m*	1 m		
Nombre de troncs	211	202	199	184	
		verticalStep			
Valeur	0.1 m	0.25 m*	0.5 m		
Nombre de troncs	199	202	213		
		searchRadius			
		4 m*			
		minLength			
		2*			
		returnMethod			
		Tous les retours*			

*Valeur de référence utilisée pour la détection

Tableau 8 - Comparaison des troncs détectés suivants différentes valeurs de paramètres.

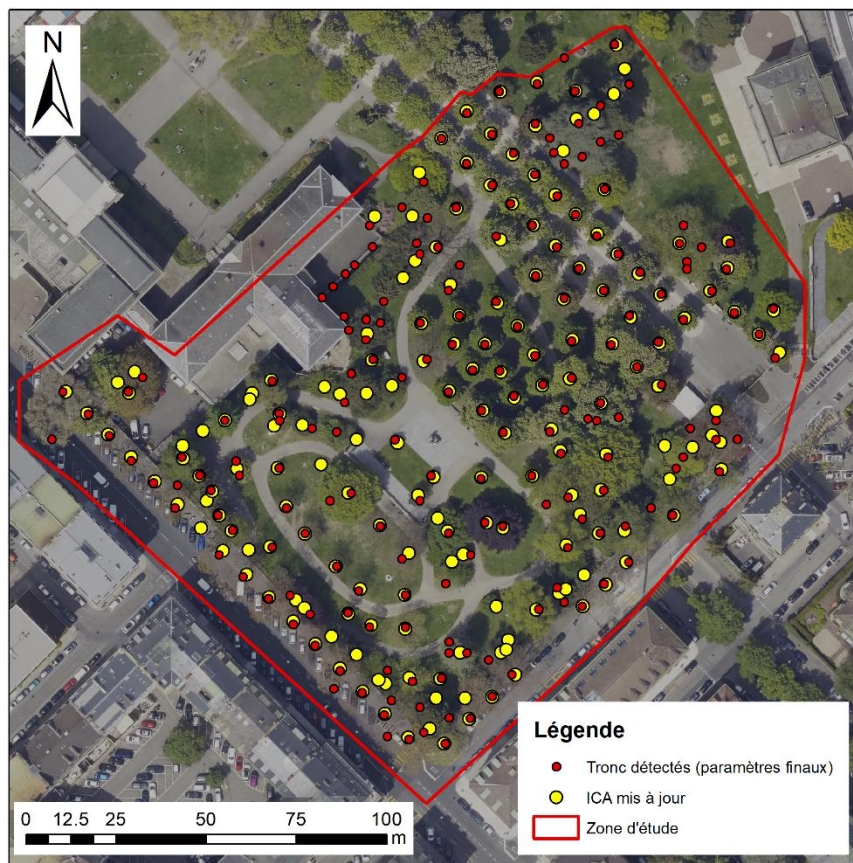


Figure 17 - Zone d'étude du parc des Bastions.

Une fois ces 3 premiers paramètres fixés, une zone d'étude de 60 tuiles LAS (12 km²) en centre-ville (Figure 18) a été utilisée afin de comparer les derniers jeux de paramètres de détection. Cette zone est bien représentative du milieu urbain genevois. De plus, l'inventaire ICA est ici bien renseigné et mis à jour. Il est important de noter que l'inventaire ICA n'est pas utilisé ici comme référence absolue pour le nombre d'arbres de la zone. Mais celui-ci est un bon indicateur utile à la calibration de la méthode.

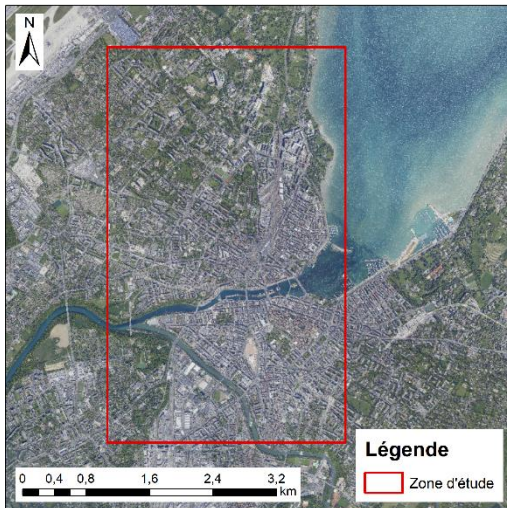


Figure 18 - Zone sélectionnée pour l'évaluation des paramètres de détection.

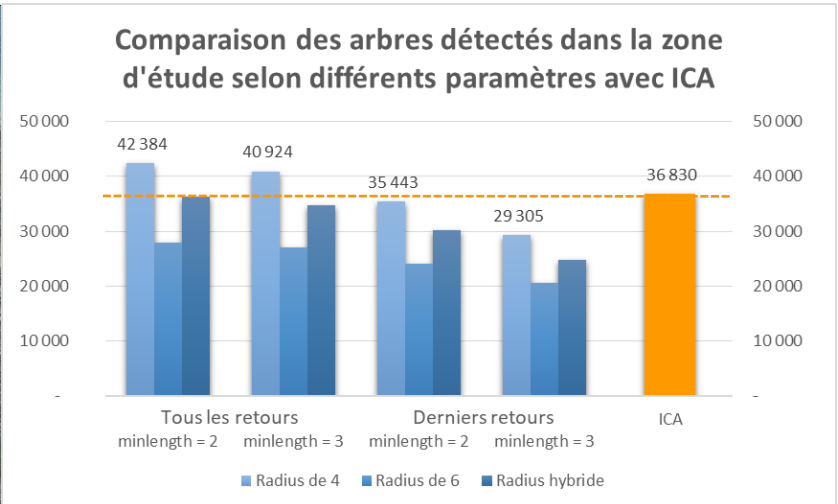


Figure 19 - Histogramme de comparaison avec ICA du nombre de troncs détectés dans la zone d'étude selon 4 jeux de paramètres différents.

L'outil de détection des troncs a été lancé quatre fois sur cette zone afin de déterminer le meilleur jeu de paramètres, en s'aidant de l'inventaire ICA. La Figure 19 présente les différents résultats. On observe tout de suite que les deux essais de détection à l'aide de la méthode 'Derniers retours' peinent fortement à atteindre le nombre d'arbres inventoriés dans ICA. Cela est le cas, peu importe le rayon de recherche et la hauteur minimum de détection utilisés. La méthode qui n'utilise que les derniers retours LiDAR est directement limitée par le nombre effectif de ces derniers retours appartenant à la végétation. En effet, peu importe l'affinage du set de paramètres utilisé pour la détection de troncs, si dans les retours LiDAR sont inexistant aucun tronc ne pourra être détecté. La Figure 20 illustre le manque de points 'Derniers retours' pour les petits arbres (cercles noirs), alors que dans le cas d'arbres mieux formés la structure interne de l'arbre apparaît plus clairement (points jaunes/oranges). Ainsi, en ne choisissant que les derniers retours une grande majorité de petits arbres seront « oubliés » alors qu'ils seront bien détectés avec tous les retours LiDAR (Figure 21). Passé un certain gabarit, les troncs sont tout aussi bien détectés avec les deux méthodes.

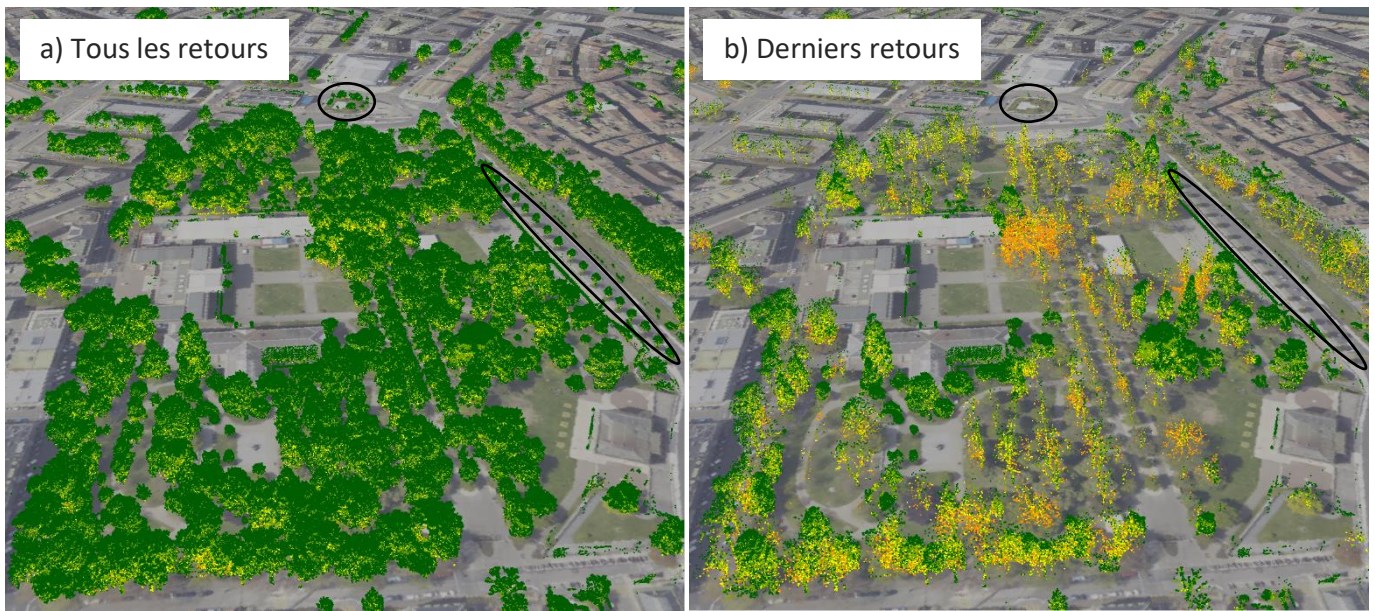


Figure 20 - Représentation du nuage de point LiDAR de végétation superposé à l'orthophoto du parc des Bastions. La couleur des points correspond au numéro du retour (de vert foncé pour le 1^{er} retour à rouge pour le 5^{ème} retour).
 a) Ensemble des retours de végétation ; b) Uniquement les derniers retours.

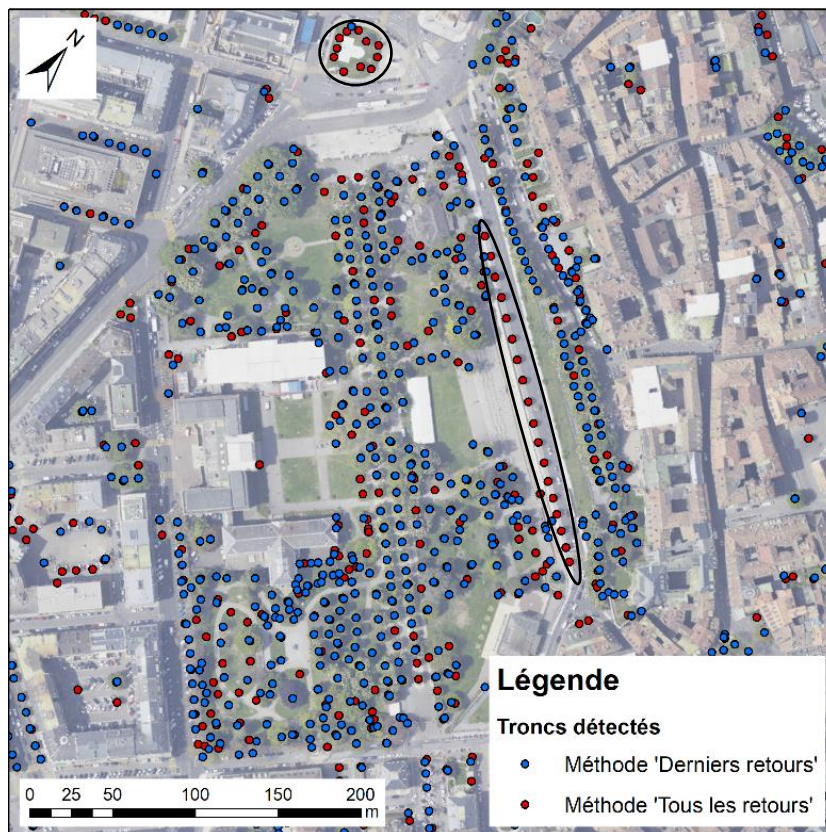


Figure 21 – Visualisation des différences de détection des troncs entre les paramètres 'Derniers retours' et 'Tous les retours'. Les autres paramètres de détections sont fixes.

Bien que la méthode 'Derniers retours' soit très efficace afin de faire ressortir la structure des grands arbres et ainsi détecter avec précision l'emplacement de leur tronc, celle-ci s'avère être trop limitée pour la détection de troncs d'arbres de taille réduite. Elle n'est donc pas utilisée pour la production de la couche finale d'emplacement des troncs.

Afin de déterminer le choix du paramètre minLength, l'outil de détection des troncs a été exécuté deux fois sur l'ensemble du canton, la première fois avec une valeur de 2 et la seconde avec une valeur de 3. La quantité de troncs obtenus pour ces deux méthodes est visible sur la Figure 22. Diminuer la valeur minlength revient à valider plus facilement un maximum local de densité comme potentiel tronc. Logiquement, plus la valeur de minLength est petite, plus le nombre de troncs détectés augmente. Le Tableau 9 présente le pourcentage de polygones du MNC (sans forêts) possédant exactement un ou au moins un tronc. On observe qu'une valeur de minlength de 2 par rapport à une valeur de 3 permet systématiquement d'augmenter le nombre de polygones de MNC possédant au moins un tronc.

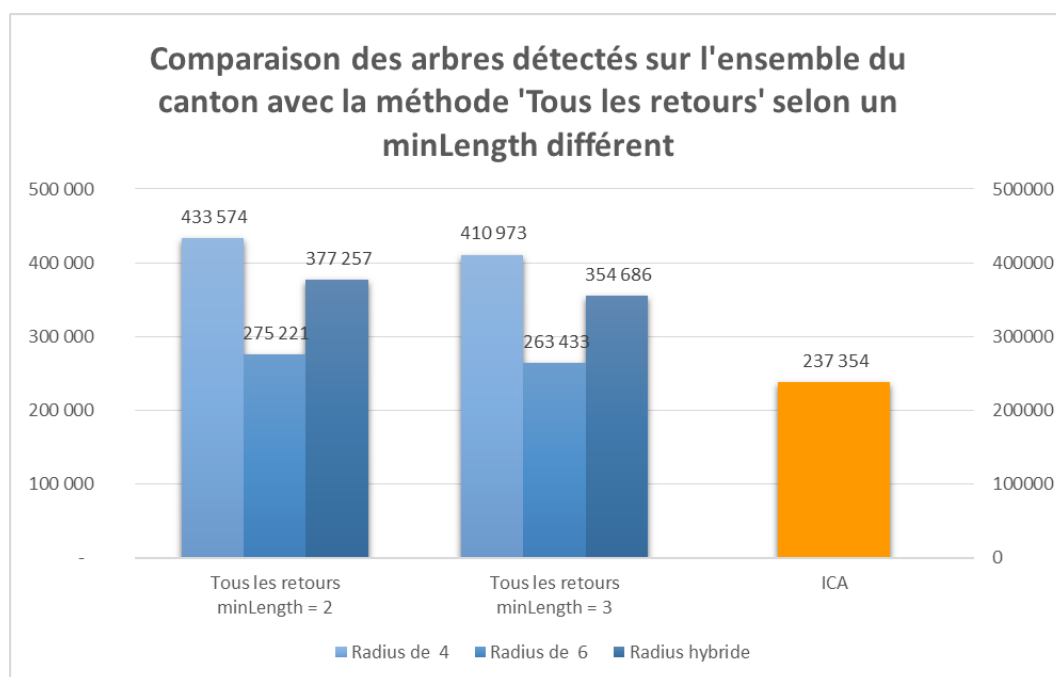


Figure 22 - Comparaison de variation du paramètre minLength sur le nombre d'arbres détectés avec la méthode 'Tous les retours' sur l'ensemble du canton.

		Tous les retours					
		minlength = 2			minlength = 3		
		Radius = 4	Radius = 6	Radius hybride	Radius = 4	Radius = 6	Radius hybride
MNC sans forêts 184 106 Polygones	Polygones avec au moins 1 tronc	87,6%	75,0%	87,5%	79,6%	70,0%	79,5%
	Polygones avec exactement 1 tronc	61,5%	57,4%	62,2%	55,0%	53,2%	55,8%

Tableau 9 - Comparaison de la part de polygones de canopée possédants au moins un tronc suivant les différentes méthodes de détection.

Dans le cas d'une valeur minlength de 2, alors que la couche de tronc hybride compte 56 000 troncs de moins que celle avec un radius de 4 m (Figure 22), la proportion de polygones de canopée comportant au moins un tronc n'est que très faiblement réduite (moins de 300 polygones). Concrètement, cela signifie que la création de la couche hybride permet bien de limiter le nombre de troncs dans les massifs dépassant 14 m de hauteur sans toutefois supprimer des arbres isolés.

Le jeu de paramètres sélectionné pour la réalisation de la couche de troncs finale est présenté dans le Tableau 10, ci-contre. La couche comporte 377 257 troncs avec une hauteur supérieure à 3 m et qui se trouvent à l'intérieur de polygones de canopée. Sur l'ensemble des troncs détectés hors forêts, plus de 114 000 sont spécifiques à un polygone unique de canopée. La table attributaire de cette couche comporte une valeur de hauteur pour chaque tronc. Cependant, cette valeur représente la hauteur de canopée à la verticale de l'emplacement du tronc détecté ainsi elle n'est pas toujours représentative de la hauteur maximum de l'arbre.

Paramètres de détection finaux	
returnMethod	'all_return'
cellSize	1 m
bandWidth	0.5 m
verticalStep	0.25 m
searchRadius	4 et 6 (hybride)
minLength	2

Tableau 10 - Jeu de paramètres de la couche de tronc finale.

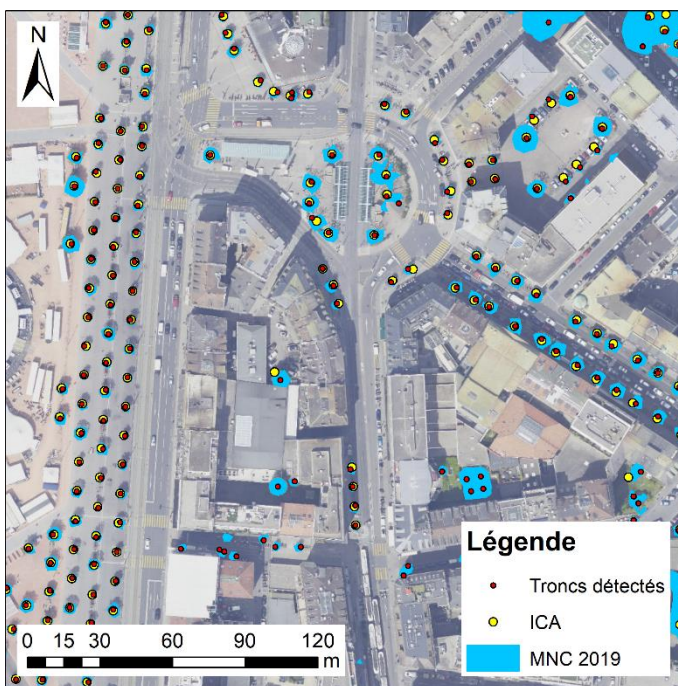


Figure 23 – Troncs détectés et arbres ICA au quartier de Plainpalais.

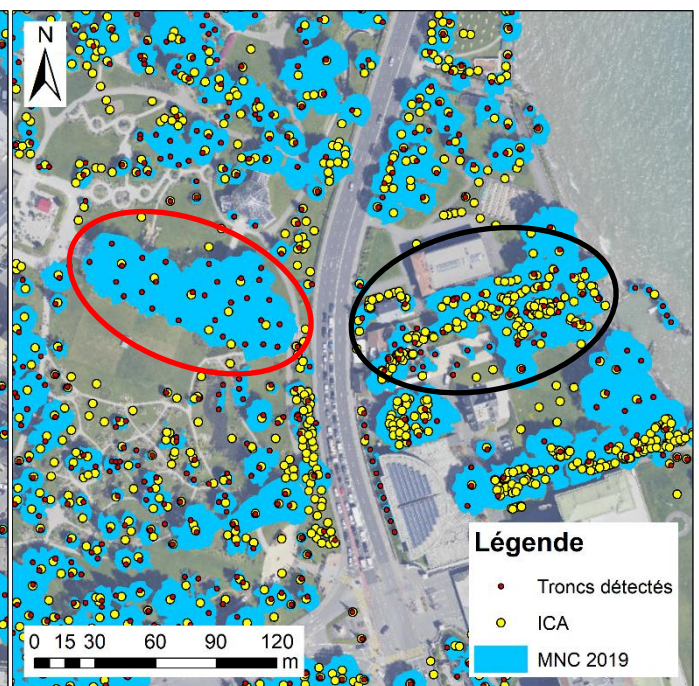


Figure 24 – Troncs détectés et arbres ICA au Jardin Botanique.

Une très bonne confiance est attribuée à la détection des petits arbres isolés, même s'il existe toujours quelques oublis dus à la fenêtre de recherche (arbres trop proches) ou simplement à une absence de différence de densité entre les cellules. Le résultat d'emplacement des arbres dans le quartier de Plainpalais montre, en effet, une très bonne détection des arbres isolés (Figure 23). Pour les arbres de très grand gabarit comme les 5 platanes alignés du Jardin botanique, plus de 20 troncs sont détectés (Figure 24 cercle rouge). Dans le cas de massifs arborés denses (Figure 24 cercle noir), le nombre de troncs est dans ce cas sous-estimé par rapport au relevé ICA.

VI. Discussion

1. CORRECTIONS DE LA CLASSIFICATION

Beaucoup de temps a été passé afin d'apporter une correction à la mauvaise classification des murs de soutènement. Cette erreur est systématique sur l'ensemble du canton et elle n'est pas présente dans les données LiDAR 2017. Ainsi il était important de corriger ce biais afin de pouvoir comparer l'évolution de la canopée sur une base similaire. La correction appliquée a permis d'éliminer efficacement les retours mal classifiés.

Néanmoins, en comparaison avec l'excellente qualité et résolution des données LiDAR 2019, il est regrettable qu'une erreur de classification aussi étendue n'ait pas été détectée et corrigée, d'autant plus que les données 2017 n'ont pas ce type de problème. Un retour a été adressé au DT afin de les informer sur les erreurs répertoriées et améliorer leur correction pour les acquisitions LiDAR des prochaines campagnes.

2. PRODUCTION DES COUCHES DE CANOPEE

Plusieurs choix ont été pris pour la production des couches finales. Comme expliqué précédemment, la couche de canopée produite correspond à l'ensemble de la végétation de plus de 3 m de hauteur et d'au moins 2,5 m² de superficie. Ces choix ont été justifiés par la volonté d'une part de ne pas intégrer l'ensemble des haies dans la canopée et d'autre part pour réduire les erreurs de classification ponctuelles comme pour certains lampadaires. Enfin, la production du MNC final implique une dernière simplification. Une valeur de simplification de 2,5 a été appliquée comme bon compromis afin d'arrondir la bordure des polygones sans trop altérer la surface de canopée. La différence relative de surface de canopée avant et après correction et simplification n'excède pas 1 %.

En ce qui concerne la correction appliquée à la végétation sur les toits, une grande quantité de temps y a été consacrée alors que l'impact final peut sembler relatif. La principale amélioration est visible sur le raster du MNHC où les valeurs de hauteurs sont effectivement corrigées. Pour la correction de surface de MNC, la différence est faible. L'autre option qui se présentait était de ne simplement pas tenir compte des arbres présents sur le cadastre des toits. Or beaucoup de cours intérieures sont considérées comme des toits et la suppression des polygones entraîne la création de 'trous' dans le MNC. C'est d'ailleurs le cas de la couche de MNC 2017 qui présente des zones relativement grandes sans canopée sur le cadastre des toits. Dans l'optique de produire une donnée représentant au mieux la réalité, la correction des valeurs a été préférée.

Enfin, concernant le script produit, une limite structurelle est à signaler. En effet, ce script est long et dans le cas d'un volume de donnée important (le canton), il a tendance à « freeze » au bout d'un certain temps. La cause n'a pas très bien été déterminée, mais c'est probablement lié au cache de mémoire vive qui se remplit. Néanmoins, ce n'est pas un problème de fonctionnement du script en soi. Dans le cas de grosse quantité de données, il est conseillé de faire tourner le script en plusieurs parties. Par exemple d'abord les parties a et b, puis dans un second temps le reste du script. En outre, dans les

paramètres il est possible de rentrer une grille de découpage personnalisée afin de pouvoir découper le canton en parties de tailles spécifiques en fonction du volume de données à traiter.

3. DETECTION DES TRONCS

Basé sur Digital-Forestry-Toolbox et donc appliqué originellement aux forêts, les scripts de cet outil ont ici été adaptés pour un milieu urbain et périurbain. Une grande quantité d'essais a été nécessaire afin de comprendre en profondeur les paramètres de détection des troncs. Étant tous plus ou moins interdépendants, il était donc difficile de trouver la meilleure combinaison. Afin d'échanger sur l'outil et de présenter le projet, une séance zoom a ainsi été organisée avec M. Parkan. Il a été d'une aide précieuse pour la compréhension du script ainsi que d'excellents conseils pour les choix de modifications de la méthode originelle. Ainsi les paramètres `cellSize`, `bandWidth` et `verticalStep` ont été choisis et fixés en premier afin de pouvoir déterminer les valeurs optimales des paramètres `returnMethod`, `minLength` et `searchRadius` sans trop de variables.

- La taille de cellule (`cellSize`) a été définie sur 1 m en partant du principe qu'une majorité d'arbres possède un tronc d'un diamètre inférieur à 1 m ; ainsi les retours correspondants à un même tronc ont plus de chance d'être comptabilisés dans la même cellule.
- Le paramètre `bandWidth` qui correspond à la hauteur de la cellule de raster a été fixé à 50 cm, car c'est le meilleur compromis pour la détection d'arbres de petite taille.
- Enfin le `verticalStep` a été conservé à une valeur de 0,25 m comme paramétré initialement dans l'outil.

Initialement cette méthode utilise seulement les derniers retours LiDAR, car ils permettent de mieux faire ressortir la structure intérieure des arbres. C'est très efficace dans le cas de forêts/massifs compacts et en général pour un ensemble d'arbres ayant une morphologie homogène. En revanche, cette méthode peine énormément à détecter les arbres avec des troncs fins et les petits arbres de manière générale, car ils ne sont pas/peu représentés par les derniers retours LiDAR. Ainsi l'utilisation de la méthode 'last_return' ne permettra pas de détecter tous les petits arbres, car même en modifiant la sensibilité de détection, si le nombre de points LiDAR en entrée n'est pas suffisant la détection est impossible.

Utiliser l'ensemble des retours LiDAR permet donc d'augmenter fortement le nombre de points sur les petits arbres et ainsi favoriser leur détection. Néanmoins, prendre en compte tous les retours augmente la densité de retours sur la totalité du feuillage. Ainsi, la détection de maximums locaux de densité (et donc de troncs) n'est plus aussi évidente qu'avec les derniers retours uniquement. Cela n'empêche pas de détecter des troncs, mais altère la précision de l'emplacement du tronc. Cette technique n'est donc pas optimisée pour les arbres déjà bien développés et les forêts.

Les scripts de détection des troncs ont donc été paramétrés afin de choisir le type de retour LiDAR à utiliser selon l'environnement d'étude. Dans notre cas, l'environnement urbain et périurbain est composé d'arbres de toute taille et morphologie. Initialement, l'idée était de coupler une détection avec tous les retours pour les petits arbres et une détection 'derniers retours' pour les arbres plus grands afin d'utiliser les avantages des deux méthodes. Cependant après réflexion, cette approche a été mise de côté au profit d'une détection uniquement avec tous les retours. L'idée étant de ne pas coupler deux méthodes de détection basée sur des données différentes qui apporteraient chacune des biais différents. L'objectif étant de ne pas « oublier des arbres » quitte à sur détecter. Les résultats de couche de troncs présentés ici sont donc orientés pour la détection d'arbres isolés.

Enfin, la confédération à travers Swisstopo a réalisé une campagne d'acquisition LiDAR le 22 février 2019 sur le canton de Genève. L'intervalle de temps entre cette acquisition et celle du canton étant de moins de deux semaines. Il serait intéressant d'essayer de fusionner ces deux jeux de données et de voir le potentiel d'utilisation de la méthode 'Dernier retours' dans ce cas.

La principale contrainte pour la détection réside dans le rayon de recherche (searchRadius). En effet, celui-ci doit être défini dans les paramètres. Ainsi pour une taille de cellule de 1 m et un rayon de recherche de 6 l'outil va comparer tous les autres maximums locaux de densité dans un radius de 6 m. Implicitement, cela revient à définir qu'il ne peut exister qu'un seul tronc tous les 6 m. Le choix du rayon de recherche en entrée empêche la détection de plusieurs troncs à l'intérieur de cette aire. Cela veut dire que cette méthode s'adapte difficilement dans des milieux hétérogènes. En effet, définir un petit rayon de recherche va être efficace pour localiser l'ensemble des troncs d'arbres rapprochés (type alignements urbain) en revanche le nombre de troncs détectés sous la canopée de grands arbres sera artificiellement surestimé. Afin de pallier en partie à ce biais, la couche de tronc finale est composée des troncs détectés avec un rayon de recherche de 4 m et de ceux détectés avec un rayon de 6 m. Une valeur seuil de 14 m de hauteur a été choisie où au-dessous de celle-ci les arbres retenus sont ceux détectés avec un rayon de 4 m et au-dessus ceux avec un rayon de recherche de 6 m. Évidemment, l'ajout d'une variable subjective (valeur seuil) n'est jamais idéal et apporte d'autres biais, mais cette technique permet un bon compromis pour assurer la détection de la grande majorité des arbres tout en limitant la surestimation du nombre de troncs. Une méthode de détection avec un rayon de recherche adaptable en fonction de la hauteur de l'arbre peut être intéressante, mais ne résoudrait aussi que partiellement le problème. En effet, la morphologie de l'arbre est tout aussi, voire plus importante. Par exemple, un peuplier possédant une hauteur de 35 m empêcherait la détection d'autres troncs dans un grand rayon autour de lui alors que sa couronne serait très réduite. Idéalement, afin de détecter un tronc unique par arbre il faudrait un rayon de recherche qui puisse s'adapter en fonction de la couronne de ce dernier. Une méthode de clustering de la canopée afin d'obtenir une couronne approximative est une piste qui mérite d'être explorée (Liu et al., 2013).

Une autre limite observée de cette méthode est la difficulté à déterminer l'emplacement des troncs pour les arbres très denses (Cèdres, Cyprès...). Le LiDAR n'arrivant pas à traverser la structure de l'arbre, il n'y a aucun retour reflétant le tronc de l'arbre. Ainsi la densité de points la plus élevée se trouve en bordure de canopée et un, voire plusieurs troncs peuvent être détectés sur le pourtour de l'arbre. Pour le moment, aucune technique de correction n'a été trouvée pour ce genre de biais.

Finalement, d'un point de vue structurel, afin de pouvoir réaliser la détection des troncs, il faut préalablement installer tout un environnement de travail compatible à l'exécution du script. L'appel de fonctions Octave via Python n'est pas optimal, car beaucoup repose sur la librairie oct2py. Les logiciels étant constamment en train d'évoluer, de potentielles mises à jour pourraient compromettre la pérennité du script sur le long terme. Migrer l'ensemble des fonctions de détection des troncs dans un environnement Python est nécessaire pour la bonne transmission et réutilisation de cet outil.

L'exploration et l'analyse des données LiDAR est un sujet très vaste et encore largement sous exploité. Ainsi, durant ces quatre mois de stage une grande partie du travail a été consacrée à cette exploration et la mise en place d'une méthode. Malheureusement, le temps a manqué pour les étapes de nettoyage de la couche des troncs et la mise en relation avec la couche ICA dans le but de mettre à jour les arbres historiques.

4. DEROULEMENT DU STAGE

Le bon déroulement de ce stage a été en grande partie conditionné par l'évolution de la situation sanitaire globale. Néanmoins, tout a été mis œuvre afin d'être accueilli dans les meilleures conditions possibles en présentiel. Je pense que ma motivation générale et l'expérience acquise auraient été largement diminuées dans le cas d'un télétravail. La grande plus-value du stage réside dans les échanges avec les personnes spécialisées dans le domaine toujours très disponibles. En outre, le stage s'est déroulé en parallèle avec celui de Gilles et cela s'est avéré très positif. Le travail en équipe a maintenu une bonne dynamique de travail et amélioré les sujets de réflexion.

Durant ce stage, plusieurs réunions et présentations (via zoom) ont été organisées avec différents acteurs de Suisse romande intéressés par le déroulement de ce projet. Ces interactions très enrichissantes m'ont aidé à prendre confiance en moi sur la maîtrise du sujet et sur les divers choix à faire. D'une manière générale, la qualité du jeu de donnée, l'attente suscitée par les résultats et la grande liberté de manœuvre offerte par ce stage ont été très valorisantes professionnellement et personnellement.

Ce très vaste projet m'a amené à me familiariser avec une large panoplie d'outils SIG et de langages informatique ce qui représente un atout professionnel. Cependant, comprendre et maîtriser ces nouveaux logiciels m'a demandé un certain temps d'adaptation. Idéalement, j'aurais souhaité avoir un peu plus de temps pour approfondir et perfectionner les résultats de ce travail.

VII. Conclusion

Les résultats produits lors de ce stage, bien que perfectibles, peuvent être utilisés comme indicateurs du patrimoine arboré Genevois en 2019. Une bonne confiance peut être accordée au Modèle Numérique de Canopée. L'emplacement des arbres isolés se révèle être très fiable dans le cas d'arbres isolés et de morphologie relativement modeste. Un travail supplémentaire est nécessaire afin d'augmenter la fiabilité des troncs détectés sous la canopée de grands arbres et de massifs.

La méthodologie présentée dans ce rapport est le fruit de nombreux essais, pistes de recherche et de choix qui ont évolué tout au long de ces quatre mois de stage. Comme toute méthode, elle est encore améliorable tant sur le fond que sur la forme, mais elle permet de donner de nouvelles pistes d'extraction de l'information des données LiDAR.

L'approche et le traitement des données LiDAR directement depuis le nuage de point a demandé un certain temps d'adaptation et d'exploration de différents outils afin de la maîtriser au mieux. Il est vrai que l'approche raster est plus accessible et rapide pour produire des modèles de canopée voir de détecter des cimes des arbres. Cependant, la force des données LiDAR réside dans leur caractère tridimensionnel. C'est grâce au travail directement sur le nuage de point que des biais ont pu être corrigés. Le côté encombrant des données LAS peut aussi facilement être contourné en éliminant les informations non nécessaires dès le début du traitement. Finalement, c'est véritablement pour la détection de l'emplacement des troncs d'arbres que l'approche tridimensionnelle offre le plus grand potentiel.

VIII. Remerciements

Mes remerciements s'adressent à toutes les personnes que j'ai côtoyées durant ce stage. Je tiens bien évidemment à remercier Pascal Martin et Nicolas Wyler des CJBG, Sylvain Greutert du SEVE et Gregory Giuliani de l'Université de Genève pour leur accompagnement, leur disponibilité et les discussions intéressantes durant ces quatre mois. Merci à Gilles Gay des Combes pour la collaboration et le soutien mutuel (et pour les quarantaines !). Merci à mes collègues de bureau Anouk Mentha et Gwenolé Blanchet pour leur aide et leur bonne humeur. Merci à Mayeul Gayet du DT pour les échanges concernant l'acquisition des données LiDAR. Merci à Olivier Travaglini pour l'aide apportée au traitement du nuage de points LiDAR. Merci à Matthew Parkan pour sa disponibilité et l'aide apportée afin d'améliorer la méthodologie de détection des troncs. Merci à Frédéric Sandoz des CJBG pour la visite de la banque de semences. Enfin, je remercie l'ensemble des collègues des CJBG pour les petits moments partagés autour d'un café et pour le chocolat !

IX. Références

Churnside, J. H. (2013). Review of profiling oceanographic lidar. *Optical Engineering*, 53(5), 051405.

Guo, X., Coops, N. C., Tompalski, P., Nielsen, S. E., Bater, C. W., & Stadt, J. J. (2017). Regional mapping of vegetation structure for biodiversity monitoring using airborne lidar data. *Ecological informatics*, 38, 50-61.

Hopkinson, C., Chasmer, L., Young-Pow, C., & Treitz, P. (2004). Assessing forest metrics with a ground-based scanning lidar. *Canadian Journal of Forest Research*, 34(3), 573-583.

Inomata, T., Triadan, D., Pinzón, F., Burham, M., Ranchos, J. L., Aoyama, K., & Haraguchi, T. (2018). Archaeological application of airborne LiDAR to examine social changes in the Ceibal region of the Maya lowlands. *PLoS one*, 13(2), e0191619.

Lim, K., Treitz, P., Wulder, M., St-Onge, B., & Flood, M. (2003). LiDAR remote sensing of forest structure. *Progress in physical geography*, 27(1), 88-106.

Liu, J., Shen, J., Zhao, R., & Xu, S. (2013). Extraction of individual tree crowns from airborne LiDAR data in human settlements. *Mathematical and Computer Modelling*, 58(3-4), 524-535.

Parkan, M. (2017). Airborne LiDAR data: relevance of visual interpretation for forestry. *Schweizerische Zeitschrift für Forstwesen*, 168(3), 127-133.

Parkan, M., & Tuia, D. (2015, July). Individual tree segmentation in deciduous forests using geodesic voting. In *2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* (pp. 637-640). IEEE.

Tiffay, M.-C. (2018). Identification des arbres via des données Lidar. *Rapport de stage, Certificat complémentaire en Géomatique, OCAN Etat de Genève*.

Sitographie

Site des CJBG :

https://www.ville-ge.ch/cjb/activites_scientifiques.php

Site de la Ville de Genève :

<https://www.geneve.ch/fr/actualites/dossiers-information/changement-climatique-geneve>

<https://www.geneve.ch/fr/actualites/ville-geneve-augmente-plantation-arbres-afin-repondre-urgence-climatique>

Détection des cimes d'arbres :

<https://sites.google.com/site/lidaretforet/ressources/6-tutoriels/python/detection-des->

[cimes-maxima-](#)

[local?tmpl=%2Fsystem%2Fapp%2Ftemplates%2Fprint%2F&showPrintDialog=1](#)

Sciences et avenir :

https://www.sciencesetavenir.fr/high-tech/reseaux-et-telecoms/a-quoi-correspond-vraiment-le-scanner-lidar-qui-equipe-l-iphone-12-pro-d-apple_148385

Running a Batch Translation in FME :

https://docs.safe.com/fme/html/FME_Desktop_Documentation/FME_Workbench/Workbench/Running_a_Batch_Translation.htm

Digital-Forestry-Toolbox :

<https://mparkan.github.io/Digital-Forestry-Toolbox/>

Guide d'utilisateur Anaconda :

<https://conda.io/projects/conda/en/latest/user-guide/index.html>

X. Annexes

L'ensemble des scripts et processus utilisés lors de ce travail sont disponibles sur la VDGbox à l'adresse suivante :

<https://vdgbox.ville-geneve.ch/index.php/s/mqenx66CwosTfqN>