



**HAL**  
open science

# Reconnaissance automatique des contours de pierres sur nuages de points 3D acquis par laser scanner terrestre

Pauline Boucher

► **To cite this version:**

Pauline Boucher. Reconnaissance automatique des contours de pierres sur nuages de points 3D acquis par laser scanner terrestre. Sciences de l'ingénieur [physics]. 2015. dumas-01334205

**HAL Id: dumas-01334205**

**<https://dumas.ccsd.cnrs.fr/dumas-01334205>**

Submitted on 20 Jun 2016

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

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

**CONSERVATOIRE NATIONAL DES ARTS ET MÉTIERS**  
**ÉCOLE SUPÉRIEURE DES GÉOMÈTRES ET TOPOGRAPHES**

---

**MÉMOIRE**

**présenté en vue d'obtenir**

**le DIPLÔME D'INGÉNIEUR CNAM**

**Spécialité : Géomètre et Topographe**

**par**

**Pauline BOUCHER**

---

**Reconnaissance automatique des contours de pierres sur nuages de points 3D  
acquis par laser scanner terrestre**

**Soutenu le 7 juillet 2015**

---

**JURY**

**PRESIDENT : M. Jean-Michel FOLLIN**

**Président du Jury**

**MEMBRES : M. Gwenaël SAGNE  
Mme Élisabeth SIMONETTO  
Mme Nathalie THOMMERET  
M. Vincent HABCHI  
M. Michel BARNOUD  
M. Olivier POILPRÉ**

**Maître de stage  
Maître de stage  
Professeur référent  
Examineur  
Examineur  
Examineur**

## Remerciements

Je tiens tout d'abord à remercier Monsieur Gwenaël SAGNE, géomètre-expert à GEOMAT, pour m'avoir accompagnée tout au long de ce TFE et pour ses conseils avisés qui ont permis d'aboutir à la réalisation de ce mémoire.

Je souhaite également remercier tout particulièrement Madame Élisabeth SIMONETTO, ma co-maître de stage, pour sa disponibilité et son aide précieuse durant ces cinq mois de travail.

De même, je remercie vivement Madame Nathalie THOMMERET, ma professeur référente, pour sa bienveillance et ses remarques constructives.

Je tiens également à remercier Monsieur et Madame ATTENCIA du cabinet GEOMAT pour leur grande gentillesse et leur soutien durant la réalisation de mon TFE.

Un grand merci à Jordi ATTENCIA pour nos nombreux échanges et conseils qui nous ont permis d'avancer dans nos travaux respectifs.

Merci à l'ensemble des personnes travaillant au sein du cabinet GEOMAT pour l'accueil chaleureux qu'ils m'ont réservé pendant ces cinq mois.

Enfin, je remercie tout particulièrement mes parents, ma sœur et mes proches pour leur soutien permanent tout au long de mes études.

# Table des matières

Remerciements .....	2
Table des matières .....	3
Introduction .....	5
<b>I OBJECTIFS ET CONTEXTE SCIENTIFIQUE .....</b>	<b>8</b>
I.1 PRESENTATION DES OBJECTIFS.....	8
I.2 LA LITTERATURE SCIENTIFIQUE ET LA DETECTION DE CONTOURS.....	9
I.2.1 Définitions des données images et nuages de points .....	9
I.2.1.1 L'image numérique .....	9
I.2.1.2 Le nuage de points .....	9
I.2.1.3 Comparaison des données .....	9
I.2.2 La détection de contours sur image .....	10
I.2.2.1 Détection de contours par filtrage linéaire et classification .....	10
I.2.2.2 Détection de contours par la transformée de Hough .....	10
I.2.2.3 Segmentation par croissance de régions (Region Growing).....	11
I.2.3 Les recherches scientifiques sur la données 3D .....	11
I.2.3.1 Les paramètres influant sur les mesures d'acquisition 3D .....	11
I.2.3.2 L'algorithme de croissance de régions sur nuage de points .....	15
I.2.3.3 L'algorithme alpha-shape sur nuage de points .....	15
I.3 LES METHODES ENVISAGEES .....	15
I.3.1 Détection des contours à partir d'orthoimages extraites du nuage de points texturé .....	16
I.3.2 Détection des contours directement sur le nuage de points .....	16
I.3.3 Justification du choix de la deuxième méthode.....	16
I.4 PRESENTATION DES NUAGES DE POINTS .....	17
<b>II CLASSIFICATION ET SEGMENTATION.....</b>	<b>20</b>
II.1 CLASSIFICATION DU NUAGE DE POINTS SELON L'INTENSITE LASER.....	20
II.1.1 Définition de la classification .....	20
II.1.2 Etude des histogrammes selon l'intensité laser.....	20
II.1.3 Seuillage simple.....	21
II.1.3.1 Le principe du seuillage simple.....	21
II.1.3.2 Détermination d'un seuil S : la méthode adaptive thresholding .....	21
II.1.4 Résultats de la classification selon l'intensité laser.....	22
II.1.4.1 Qualité de la détection par classification .....	24
II.2 SEGMENTATION PAR CROISSANCE DE REGIONS .....	25
II.2.1 Présentation de l'algorithme .....	25
II.2.2 Critère d'homogénéité et seuil de validation .....	28
II.2.3 Résultats de la segmentation par croissance de régions.....	28
II.2.4 Qualité de la détection par croissance de régions .....	29
II.3 COMPARAISON DES DEUX METHODES .....	30
<b>III EXTRACTION DES POINTS DE CONTOURS.....</b>	<b>31</b>
III.1 L'ALGORITHME ALPHA-SHAPE .....	31
III.1.1 Présentation de l'algorithme .....	31
III.1.2 Résultats obtenus par la méthode alpha-shape .....	34
III.1.2.1 Test sur un extrait du nuage de points n°1 .....	34
III.1.2.1.1. Application au nuage résultant de la classification du nuage de points n°1.....	34

III.1.2.1.2.	Application au nuage n°1 classifié et filtré.....	36
III.1.2.1.3.	Application au nuage résultant de la segmentation par croissance de régions du nuage de points n°1.....	36
III.1.2.1.4.	Qualité de la détection des contours par notre algorithme alpha-shape en langage python.....	37
III.1.2.2	Test sur un extrait du nuage de points n°2.....	38
III.1.2.2.1.	Application au nuage résultant de la classification du nuage de points n°2.....	38
III.1.2.2.2.	Application au nuage résultant de la segmentation du nuage de points n°2.....	39
III.1.2.2.3.	Qualité de la détection des contours par notre algorithme alpha-shape en langage python.....	40
III.2	DETERMINATION DES CONTOURS SOUS LE LOGICIEL 3DRESHAPER.....	40
III.2.1	Détection des contours à partir d'un maillage : alpha-shape.....	40
III.2.1.1	Traitement du nuage de points n°1.....	41
III.2.1.1.1.	Traitement du nuage de points n°1.....	41
III.2.1.1.2.	Création d'un maillage 3D.....	42
III.2.1.1.3.	Création des polygones 3D de contours.....	44
III.2.1.1.4.	Calcul d'un plan moyen au nuage de points.....	46
III.2.1.1.5.	Obtention des polygones 2D et qualité des données.....	47
III.2.1.1.6.	Export des polygones 2D au format DXF.....	47
III.2.1.2	Traitement du nuage de points n°2.....	48
III.2.1.3	Qualité de la détection de contours par la méthode alpha-shape de 3DReshaper.....	49
III.2.2	Détection des contours par extraction des lignes de rupture de pente.....	50
III.2.2.1	Réalisation d'un maillage en deux étapes.....	50
III.2.2.2	Extraction des lignes de rupture de pente.....	51
<b>IV</b>	<b>DISCUSSIONS DES RESULTATS.....</b>	<b>52</b>
IV.1	LES PROCEDES DE RECONNAISSANCE AUTOMATIQUE DE CONTOURS DE PIERRES SUR NUAGES DE POINTS.....	52
IV.2	ANALYSES ET CHOIX DES METHODES.....	52
IV.2.1	Etape 2 : Détection des points de pierres par classification ou segmentation par croissance de régions.....	52
IV.2.2	Etape 3 : Elimination des points isolés par filtrage/nettoyage du nuage.....	53
IV.2.3	Etape 4 : Reconnaissance des contours par l'algorithme en langage python ou par le logiciel 3DReshaper.....	53
Conclusion.....		54
Bibliographie.....		55
Liste des figures.....		56
Liste des tableaux.....		59
Résumé.....		60
Poster Scientifique.....		64
Abstract.....		66

## Introduction

L'acquisition de données 3D s'est rapidement développée et répandue au cours de ces dix dernières années au sein de diverses structures telles que les sociétés de géomètres-experts, de travaux publics, les collectivités territoriales et même plus largement auprès du grand public.

En effet, aujourd'hui nombre d'infrastructures professionnelles sont équipées en matériel d'acquisition de données tridimensionnelles afin de modéliser différents espaces et volumes souhaités, améliorant ainsi d'une part le temps d'acquisition et d'autre part, la qualité géométrique et graphique du résultat. Par ailleurs, l'émergence de nouvelles applications telles que la navigation mobile proposée par certains GPS ainsi que les visites virtuelles de sites en trois dimensions étendent l'utilisation de la donnée 3D au grand public.

Les méthodes de levés tridimensionnels répondent à de nombreuses et diverses demandes telles que les modélisations et calculs volumétriques, levés de façades, levés d'éléments complexes (auscultation de carrières, de berges). Plusieurs méthodes d'acquisition 3D existent de nos jours, par exemple la photogrammétrie, la radargrammétrie et la lasergrammétrie.

La photogrammétrie est une technique mise au point dès la moitié du XIX<sup>ème</sup> siècle par l'officier topographe français Aimé Laussedat, permettant d'effectuer des mesures 3D à partir de deux images formant un couple stéréoscopique et rendant possible la vision en relief. Ainsi, un modèle tridimensionnel est obtenu grâce à un ensemble d'images acquis sous différents points de vue et couvrant une même zone géographique. La photogrammétrie permet d'aboutir à la création d'un nuage de points par identification de points ou lignes similaires entre deux images.

La radargrammétrie repose sur le même principe que la photogrammétrie avec l'exploitation d'images radar permettant de déterminer les caractéristiques géométriques du terrain étudié et ainsi en déduire les formes et les détails. Dans ce domaine, c'est cependant l'interférométrie d'images radar qui est préférée.

La lasergrammétrie consiste en la détermination de position 3D à partir d'un capteur laser qui balaye l'espace en émettant des impulsions dans le visible ou le proche infra-rouge. Les échos reçus par le capteur sont ensuite analysés. La mesure du temps aller-retour de la propagation de l'onde permet de déterminer la distance de la surface réfléchissante (objet mesuré) par rapport au capteur.

Il existe actuellement une demande grandissante de la part du Centre des Monuments Nationaux concernant la réalisation de levés architecturaux.

En effet, cet établissement public administratif placé sous la tutelle du Ministre de la Culture et de la Communication a pour principales missions de gérer, conserver et restaurer les monuments nationaux qui sont propriétés de l'État Français. Ainsi, il est nécessaire pour ce dernier d'avoir une connaissance exhaustive de ces monuments et plus particulièrement des pierres composant l'édifice afin de pouvoir estimer le coût d'éventuels travaux d'entretien et de rénovation. C'est pourquoi, il peut être demandé dans les appels d'offre de rendre compte de l'état actuel des monuments avec leurs déformations et pathologies existantes en effectuant un levé pierre à pierre des édifices et une restitution graphique vectorielle sous AutoCAD.

Pour répondre à cette demande, la méthode initialement employée par la structure d'accueil GEOMAT consiste à détourner manuellement chaque pierre de l'édifice sur des nuages de points colorisés obtenus par levés lasergrammétriques (Figures 1 et 2). Il s'agit d'un travail fastidieux et chronophage

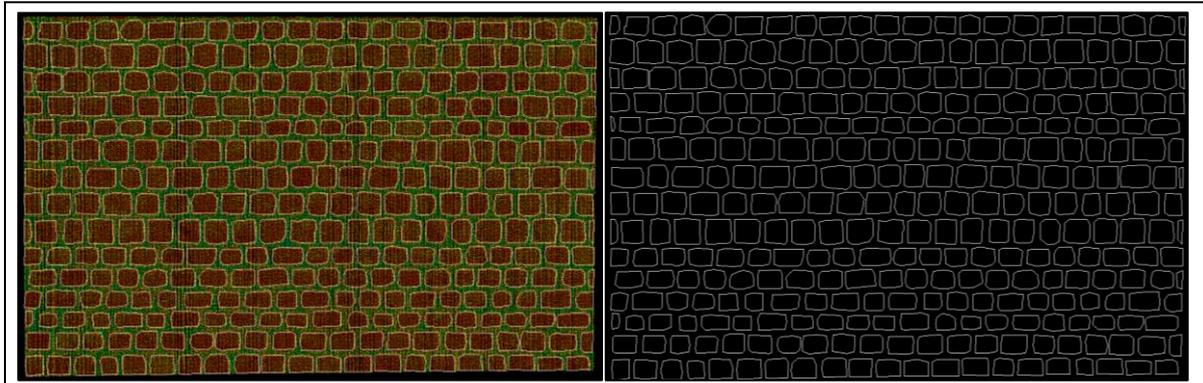


Figure 1 : Exemple d'un résultat obtenu actuellement par GEOMAT en détourant manuellement chaque pierre sur le nuage de points [CloudWorx pour AutoCAD]

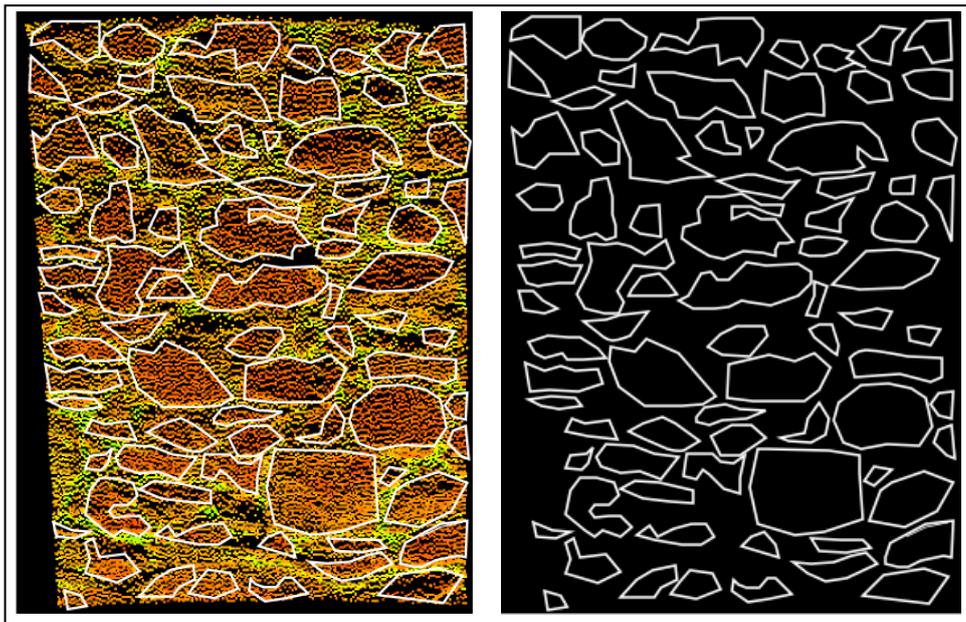


Figure 2 : Exemple d'un deuxième résultat obtenu actuellement par GEOMAT en détourant manuellement chaque pierre sur le nuage de points [CloudWorx pour AutoCAD]

La mise en place d'un processus de reconnaissance automatique de contours de pierres sur nuages de points s'est alors rapidement imposée au sein de la société. C'est dans ce cadre d'étude que s'inscrit mon Travail de Fin d'Études. Ainsi, pendant ces cinq mois de stage, il m'est donné de rechercher et d'analyser les différentes méthodes permettant d'aboutir à ce résultat.

La réalisation de ce mémoire a été suivie et supervisée tout au long de ces cinq mois par deux organismes à la fois distincts et complémentaires : le cabinet de géomètres-experts GEOMAT et le laboratoire de recherche Géomatique et Foncier GeF.

GEOMAT est une Société d'Exercice Libéral par Actions Simplifiée (SELAS) composée de neuf géomètres-experts associés dont le siège social se situe à Fougères, en Ile-et-Vilaine. Constituée de quinze agences réparties sur les régions Bretagne et Basse-Normandie, cette entreprise possède des compétences pluridisciplinaires lui permettant de répondre à de nombreux marchés en matières de foncier, de topographie, d'aménagement urbain et paysager, et enfin, d'aménagement rural.

Le laboratoire Géomatique et Foncier (GeF), installé dans les locaux de l'ESGT au Mans, se compose de deux équipes de recherche : le Laboratoire de Géodésie et de Géomatique (L2G) et l'Équipe de Recherche en Aménagement, Droit immobilier et Foncier (ERADIF).

- Le Laboratoire de Géodésie et de Géomatique (L2G) effectue des recherches orientées sur les techniques géodésiques de positionnement, de mesures de déformations, sur les développements en gravimétrie mobile, la restitution, la modélisation, la gestion de l'information tridimensionnelle et des informations géographiques dynamiques.
- Les recherches de l'Équipe de Recherche en Aménagement, Droit immobilier et Foncier (ERADIF) reposent sur l'évaluation du droit et de ses transformations, notamment en matière de politiques publiques liées à l'aménagement et l'urbanisme, la gestion du foncier et l'application du droit immobilier.

Nous commencerons ce mémoire par la présentation du contexte et des objectifs visés en nous attachant à préciser la méthodologie qui sera adoptée dans la suite de ce travail.

Dans une deuxième partie, nous nous pencherons plus spécifiquement sur les différents algorithmes de classification et de segmentation testés avant d'aborder, dans une troisième partie, les méthodes permettant l'extraction des points de contours des pierres présents au sein du nuage de points traité.

Enfin, dans la quatrième et dernière partie de ce travail, nous nous attacherons à dresser une analyse comparative des résultats obtenus selon les différentes méthodes étudiées.

# I Objectifs et contexte scientifique

## I.1 Présentation des objectifs

Ce travail de recherche a pour principal objectif la détection de contours de pierres sur nuages de points afin de faciliter par la suite l'analyse pathologique du bâti par les architectes des bâtiments de France et les bureaux d'études en charge de cette étude. Le produit final doit permettre à ces derniers d'établir un estimatif de surface et de volume des pierres du bâtiment à étudier et ainsi chiffrer le montant des travaux à effectuer. Pour ce faire, la précision de notre produit final doit être égale à 3 cm en absolu.

Ainsi, après acquisition lasergrammétrique de l'édifice, il s'agit de détecter la présence de pierres sur le nuage de points afin d'en extraire les contours. Autrement dit, pour chaque pierre représentée sur le nuage de points, nous souhaitons obtenir les coordonnées des points formant son contour. De manière plus précise, la détection des pierres et des joints sur le nuage s'effectuera par les étapes de segmentation et de classification du nuage de points (Figure 3).

En effet, la segmentation consiste à partitionner le nuage de points en régions homogènes et constitue l'étape fondamentale du processus de détection. La qualité de détection des points de contours dépend directement de la qualité de la segmentation. En l'absence d'une segmentation optimale, l'extraction des contours ne sera pas satisfaisante et le résultat sera inexploitable pour l'analyse pathologique qui succédera à notre travail.

La classification permet l'attribution de points à une classe thématique. Ici, deux classes nous intéressent : la première correspondant aux pierres et la seconde aux joints.

Le rendu final devant être plan pour une meilleure visualisation sur papier, il faudra alors projeter les polygones 3d sur un plan afin d'obtenir des polygones 2d tout en minimisant la dégradation des données. En effet, nous nous attacherons à conserver une précision des coordonnées des points de contours pour rester dans les tolérances imposées par le cahier des charges, soit 3 cm en positionnement planimétrique.

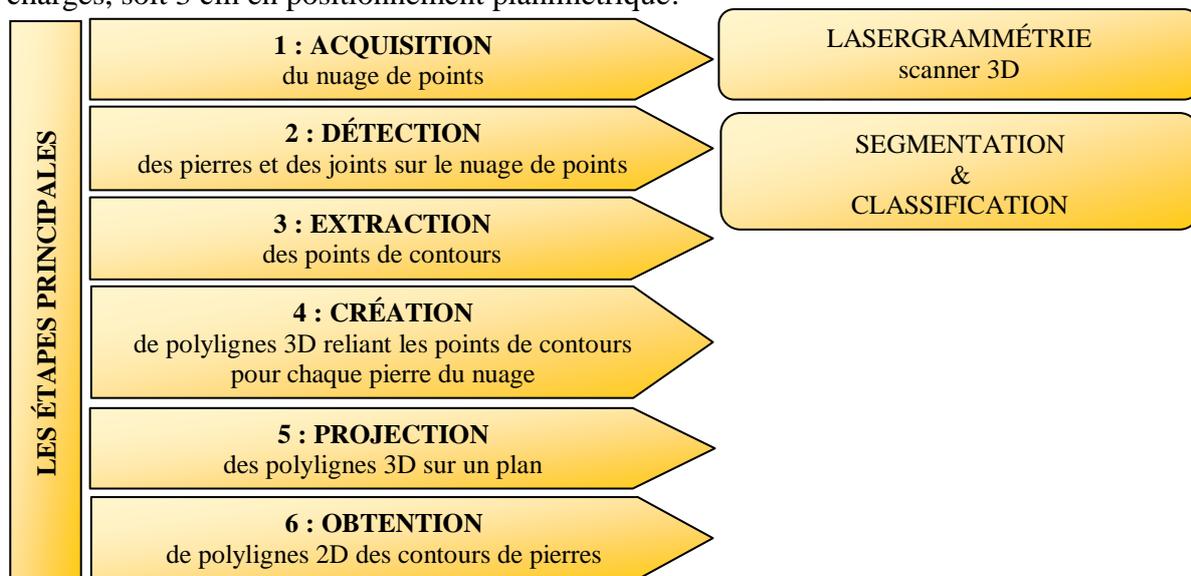


Figure 3 : Chaîne de traitement général permettant l'obtention des contours de pierres sur nuage de points

Par ailleurs, afin de proposer une solution répondant au mieux aux besoins de l'entreprise d'accueil GEOMAT, il s'avère nécessaire de mettre en place un processus de détection simple et facilement exploitable par les employés de la structure.

C'est pourquoi, la recherche de méthodes actuellement existantes sur le thème de la détection de contours est indispensable afin de pouvoir établir par la suite les différents procédés répondant aux objectifs de notre étude et de privilégier le plus adapté à l'entreprise.

## I.2 La littérature scientifique et la détection de contours

Si la détection de contours est un sujet qui a fait l'objet de nombreuses études en matière de traitement numérique des images, celui-ci est encore en pleine évolution dans le domaine de la 3D. Au premier abord, on pourrait penser que le processus de détection sur image pourrait facilement s'adapter sur de la donnée tridimensionnelle. Toutefois, le principal frein à cette hypothèse réside dans la définition même de la donnée : l'image numérique est formée par des pixels et constitue une entité continue et régulière tandis que le nuage de points, défini par une multitude de points, est une entité discontinue et irrégulière.

### I.2.1 Définitions des données images et nuages de points

#### I.2.1.1 L'image numérique

L'image numérique généralement utilisée en traitement numérique des images se présente sous forme matricielle dont chaque élément représente un pixel (Figure 4). Mathématiquement, elle est définie par une fonction bidimensionnelle  $f(x,y)$  où les variables  $x$  et  $y$  sont discrètes. Chaque pixel est entouré de voisins dont les caractéristiques radiométriques sont plus ou moins liées entre elles.



Figure 4 : Photographie du château médiéval de Fougères

#### I.2.1.2 Le nuage de points



Le nuage de points, comme son nom l'indique, est formé par une multitude de points référencés dans un système tridimensionnel (Figure 5).

Figure 5 : Nuage de points du château médiéval de Fougères [Cyclone]

#### I.2.1.3 Comparaison des données

	<b>Image</b>	<b>Nuage de points</b>
<b>Unité</b>	Pixel	Point
<b>Structure de la donnée</b>	Régulière (matricielle) et bidimensionnelle	Irrégulière et tridimensionnelle
<b>Représentation de la donnée</b>		

Tableau 1 : Comparaisons des données image/nuage de points

## I.2.2 La détection de contours sur image

Il existe actuellement de nombreuses méthodes de détection de contours sur image. Il s'agira dans cette partie de retracer les plus connues.

### I.2.2.1 Détection de contours par filtrage linéaire et classification

Les contours d'une image correspondent aux hautes fréquences spatiales de l'image. Ainsi, pour extraire les contours présents sur l'image, il suffit d'appliquer un filtre passe-haut linéaire de type Gradient, Prewitt, Robert ou encore Sobel, de calculer le gradient dans toutes les directions en chaque pixel pour sélectionner le gradient maximum en valeur absolue et enfin, de déterminer un seuil de décision qui permet de mettre en évidence les points de contour de l'image.

Ainsi, à l'aide d'un script python, nous avons testé cette méthode sur une photographie d'un mur, dont voici les résultats obtenus (Figures 6, 7, 8, 9 et 10) :



Figure 6 : Image original

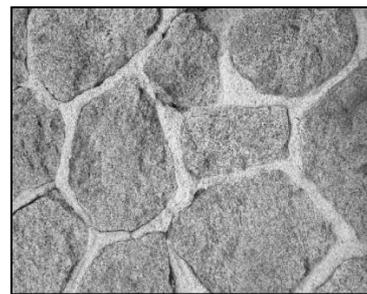


Figure 7 : Image convertie en niveau de gris [Python]

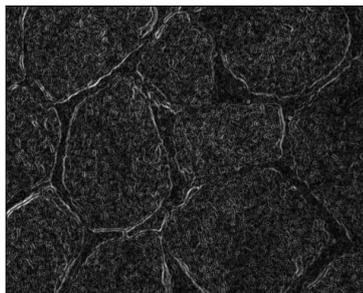


Figure 8 : Image de la norme des gradients maximum en x et y avec un filtre Sobel [Python]

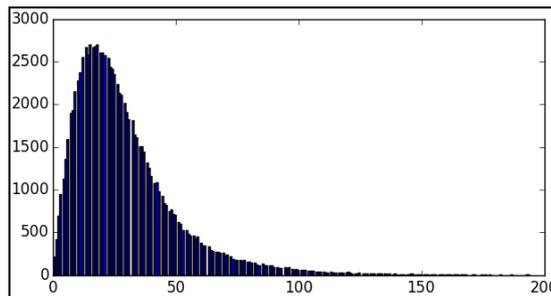


Figure 9 : Histogramme de l'image de la norme des gradients maximum en x et y avec un filtre Sobel [Python]

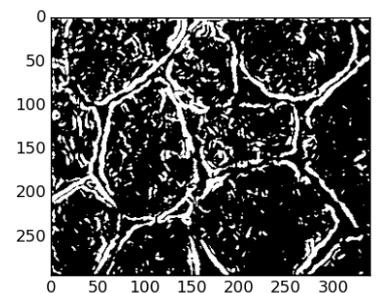


Figure 10 : Image seuillée pour un seuil de 20 [Python]

Il existe d'autres filtres permettant de mettre en évidence les points de contours.

### I.2.2.2 Détection de lignes de contour par la transformée de Hough

Ce principe repose sur la détection de courbes paramétriques sur des images binaires. Afin de passer de l'espace image 2D à l'espace transformé ND (N : nombre de paramètres décrivant la courbe recherchée), on détermine le nombre de pixels décrivant la forme recherchée dans l'espace des paramètres permettant d'obtenir une nappe d'accumulation où le pic nous informe de la forte probabilité de la présence de la courbe correspondante.

De même que précédemment, nous avons testé cette méthode en élaborant un script python et avons obtenu les résultats suivants (Figures 11 et 12) :



Figure 11 : Image originale

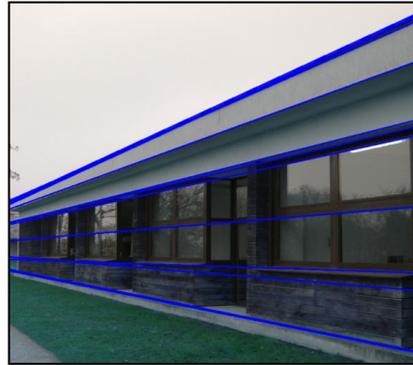


Figure 12 : Extraction des lignes de contours par la méthode de la transformée de Hough [Python]

### I.2.2.3 Segmentation par croissance de régions (Region Growing)

Cet algorithme consiste à partir de germes (un germe correspond à un pixel de l'image) qui grossissent par agglomération des pixels voisins dès qu'ils vérifient les prédicats posés (par exemple, fixation d'une valeur seuil maximale du contraste, de l'entropie ou de l'écart-type des intensités des pixels) pour former les régions homogènes de l'image. La croissance s'interrompt dès qu'il n'est plus possible d'ajouter de pixels à une région connexe. Ainsi, on obtient l'image partitionnée en plusieurs régions (Figure 13).

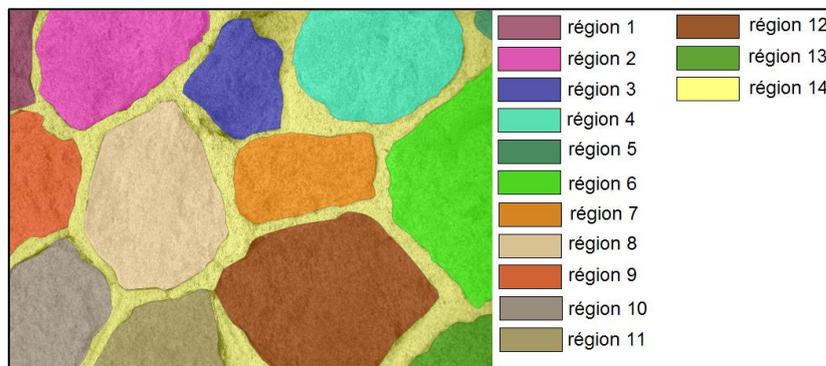


Figure 13 : Exemple de résultat de segmentation par croissance de régions [Inkscape]

## I.2.3 Les recherches scientifiques sur la donnée 3D

Des recherches scientifiques ont été menées sur ce sujet à l'échelle planétaire par différents laboratoires et organismes. En effet, avec l'émergence de la donnée 3D, les scientifiques se sont intéressés d'une part aux limites des appareils d'acquisition de données 3D et d'autre part, aux différentes possibilités de segmentation sur nuage de points permettant d'extraire les points de contours au sein du nuage.

### I.2.3.1 Les paramètres influant sur les mesures d'acquisition 3D

Comme le démontre plusieurs études scientifiques, les erreurs de positionnement géométriques dépendent directement de la portée d'acquisition. En effet, plus la portée est petite, meilleure est la précision géométrique. S'agissant de l'intensité laser, celle-ci est peu influencée par la distance d'acquisition mais diminue lorsque l'angle d'incidence du laser s'éloigne de la normale au mur<sup>1</sup>.

<sup>1</sup> [Voegtler et Wakalul, 2009]

Voici les résultats d'une étude portée sur les paramètres influençant les mesures d'acquisition 3D<sup>2</sup> (Tableaux 2, 3 et 4) :

➤ Tests réalisés avec un scanner laser 3D Trimble GX pour des portées de 14 m :

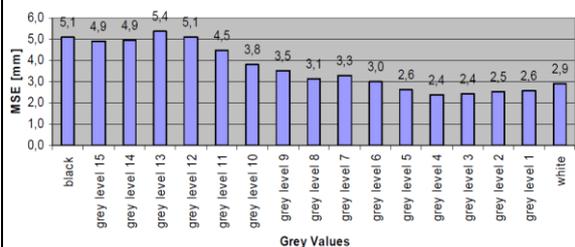
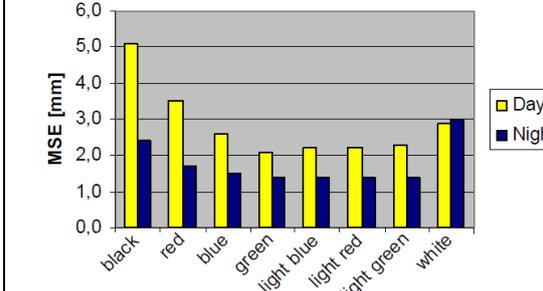
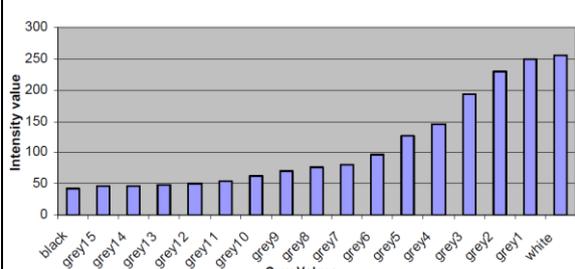
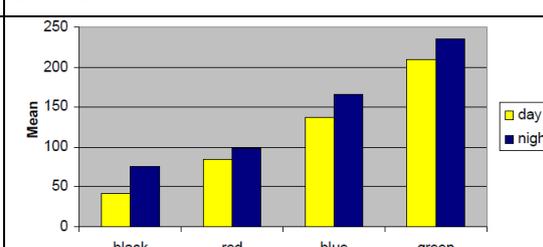
	Nuances de gris	Couleur
<b>Précision géométrique</b>	 <p>Figure 14 : Influence des nuances de gris sur la précision géométrique [Voegtle et al., 2008]</p> <p>Il existe une dépendance entre la clarté de l'objet et la précision géométrique. Plus l'objet est clair, meilleure est la précision géométrique.</p>	 <p>Figure 15 : Influence de la couleur sur la précision géométrique [Voegtle et al., 2008]</p> <p>L'erreur moyenne quadratique décroît du noir vers le rouge, le bleu, le vert, le bleu clair, le rouge clair, le vert clair et enfin le blanc.</p> <p>Cette erreur est plus élevée de jour que de nuit (~ facteur 2) sauf pour le blanc.</p> <p>Les couleurs sensiblement différentes de celle du rayon laser produisent un écart plus grand.</p>
<b>Intensité laser</b>	 <p>Figure 16 : Influence des nuances de gris sur l'intensité laser [Voegtle et al., 2008]</p> <p>L'intensité laser croît du noir vers le blanc de façon non linéaire.</p>	 <p>Figure 17 : Influence de la couleur sur l'intensité laser [Voegtle et al., 2008]</p> <p>De même que pour les nuances de gris, les valeurs d'intensité moyenne augmentent du noir vers le rouge, le bleu puis le vert, et sont plus élevées de nuit que de jour.</p>

Tableau 2 : Influences de la couleur et des niveaux de gris sur l'intensité laser

<sup>2</sup> [Voegtle et al., 2008]

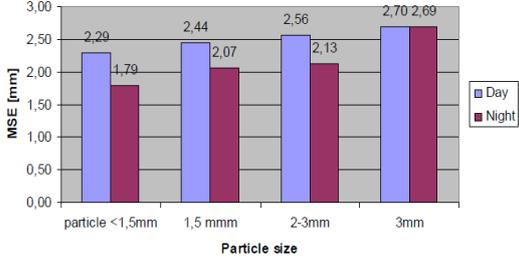
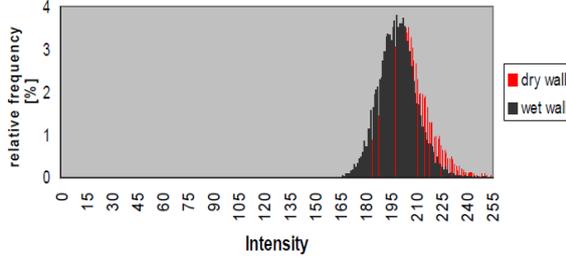
	Plâtre	Humidité sur bois et mur
<b>Précision géométrique</b>	 <p>Figure 18 : Erreurs quadratiques moyennes en fonction de la taille des particules de plâtre [Voegtle et al., 2008]</p> <p>L'erreur quadratique moyenne a une valeur plus faible de nuit que de jour et diminue avec la taille des particules de plâtre.</p> <p>Il existe une dépendance entre cette erreur et la taille des particules de plâtre due à la variation de rugosité des surfaces.</p>	<p>L'erreur quadratique moyenne ne diffère pas sensiblement en fonction de l'humidité ce qui atteste de l'indépendance des mesures et des conditions d'humidité.</p> <p>Les résultats sont similaires sur un mur.</p>
<b>Intensité laser</b>	<p>Dû à la même couleur des objets en plâtre testés (blanc), l'intensité laser ne varie quasiment pas.</p>	 <p>Figure 19 : Histogramme des intensités laser sur mur sec et humide [Voegtle et al., 2008]</p> <p>L'humidité du bois entraîne une légère diminution de la valeur d'intensité laser.</p> <p>Les résultats sont similaires sur un mur.</p>

Tableau 3 : Influences des matériaux en plâtre, bois et de l'humidité sur l'intensité laser

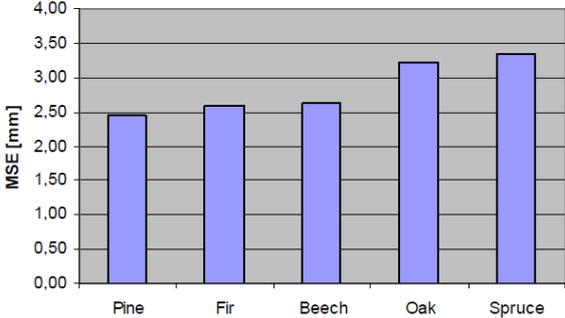
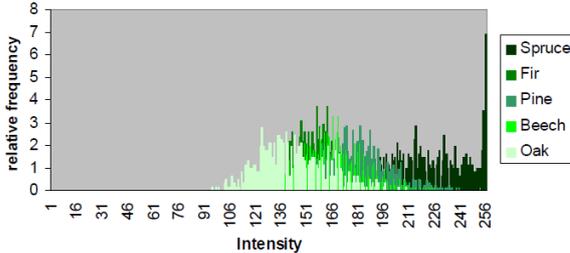
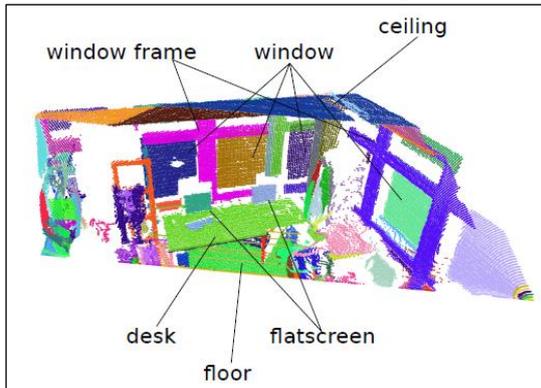
	<b>Espèces de bois : pin, sapin, hêtre, chêne, épicéa</b>	<b>Métal</b>																																	
<b>Précision géométrique</b>	 <p>Figure 20 : Influence des espèces de bois sur la précision géométrique [Voegtle et al., 2008]</p> <p>Les erreurs quadratiques moyennes sont relativement proches selon l'espèce de bois étudié. Ainsi, la précision géométrique ne dépend pas de la nature de bois de l'objet.</p>	<table border="1"> <thead> <tr> <th>Incidence Angle [ gon ]</th> <th>MSE [mm]</th> <th>Number of points</th> </tr> </thead> <tbody> <tr><td>100</td><td>± 7.0</td><td>63 602</td></tr> <tr><td>90</td><td>± 4.3</td><td>72 086</td></tr> <tr><td>80</td><td>± 3.1</td><td>66 779</td></tr> <tr><td>70</td><td>± 2.4</td><td>65 247</td></tr> <tr><td>60</td><td>± 1.9</td><td>56 756</td></tr> <tr><td>50</td><td>± 1.6</td><td>51 783</td></tr> <tr><td>40</td><td>± 1.4</td><td>44 214</td></tr> <tr><td>30</td><td>± 1.2</td><td>33 791</td></tr> <tr><td>20</td><td>± 1.0</td><td>23 424</td></tr> <tr><td>10</td><td>± 0.9</td><td>11 009</td></tr> </tbody> </table> <p>Figure 21 : Influence de l'angle d'incidence sur la précision géométrique sur du métal [Voegtle et al., 2008]</p> <p>L'erreur quadratique moyenne dépend directement de l'angle d'incidence du rayon laser sur l'objet en métal.</p> <p>Plus l'angle d'incidence diminue, plus l'erreur quadratique moyenne et le nombre de points diminuent.</p>	Incidence Angle [ gon ]	MSE [mm]	Number of points	100	± 7.0	63 602	90	± 4.3	72 086	80	± 3.1	66 779	70	± 2.4	65 247	60	± 1.9	56 756	50	± 1.6	51 783	40	± 1.4	44 214	30	± 1.2	33 791	20	± 1.0	23 424	10	± 0.9	11 009
Incidence Angle [ gon ]	MSE [mm]	Number of points																																	
100	± 7.0	63 602																																	
90	± 4.3	72 086																																	
80	± 3.1	66 779																																	
70	± 2.4	65 247																																	
60	± 1.9	56 756																																	
50	± 1.6	51 783																																	
40	± 1.4	44 214																																	
30	± 1.2	33 791																																	
20	± 1.0	23 424																																	
10	± 0.9	11 009																																	
<b>Intensité laser</b>	 <p>Figure 22 : Influence des espèces de bois sur l'intensité laser [Voegtle et al., 2008]</p> <p>L'intensité laser croît selon l'espèce de bois : chêne, hêtre, pin, sapin et épicéa.</p>	<table border="1"> <thead> <tr> <th>Incidence Angle [ gon ]</th> <th>Mean Intensities [grey values]</th> <th>Number of points</th> </tr> </thead> <tbody> <tr><td>100</td><td>253.8</td><td>63 602</td></tr> <tr><td>90</td><td>231.4</td><td>72 086</td></tr> <tr><td>80</td><td>149.7</td><td>66 779</td></tr> <tr><td>70</td><td>99.3</td><td>65 247</td></tr> <tr><td>60</td><td>68.2</td><td>56 756</td></tr> <tr><td>50</td><td>49.8</td><td>51 783</td></tr> <tr><td>40</td><td>36.8</td><td>44 214</td></tr> <tr><td>30</td><td>28.8</td><td>33 791</td></tr> <tr><td>20</td><td>16.6</td><td>23 424</td></tr> <tr><td>10</td><td>9.1</td><td>11 009</td></tr> </tbody> </table> <p>Figure 23 : Valeurs des intensités moyennes sur un objet en métal pour différents angles d'incidence [Voegtle et al., 2008]</p> <p>Lorsque l'angle d'incidence est orthogonal, la réflectivité est totale et l'intensité laser est dans le domaine de saturation ce qui produit une déformation de la pulsion laser.</p> <p>L'intensité laser diminue lorsque l'angle d'incidence diminue.</p>	Incidence Angle [ gon ]	Mean Intensities [grey values]	Number of points	100	253.8	63 602	90	231.4	72 086	80	149.7	66 779	70	99.3	65 247	60	68.2	56 756	50	49.8	51 783	40	36.8	44 214	30	28.8	33 791	20	16.6	23 424	10	9.1	11 009
Incidence Angle [ gon ]	Mean Intensities [grey values]	Number of points																																	
100	253.8	63 602																																	
90	231.4	72 086																																	
80	149.7	66 779																																	
70	99.3	65 247																																	
60	68.2	56 756																																	
50	49.8	51 783																																	
40	36.8	44 214																																	
30	28.8	33 791																																	
20	16.6	23 424																																	
10	9.1	11 009																																	

Tableau 4 : Influences de différentes espèces de bois et du métal sur l'intensité laser

### I.2.3.2 L'algorithme de croissance de régions sur nuage de points

De la même manière qu'en traitement numérique des images, l'algorithme de croissance



de région<sup>3</sup> consiste à partitionner le nuage de points en régions homogènes en partant de points germes qui grossissent par agglomération de points voisins dès qu'ils vérifient un critère d'homogénéité (Figure 24). Le processus s'interrompt dès qu'il n'est plus possible d'ajouter de points à une région.

Nous reverrons plus en détails ce principe dans la suite de ce mémoire<sup>4</sup>.

Figure 24 : Segmentation par croissance de régions d'un nuage de points [Eich et al., 2010]

### I.2.3.3 L'algorithme alpha-shape sur nuage de points

Cette méthode<sup>5</sup> permet d'identifier les points de contours de formes incluses dans des ensembles de points répartis de manière irrégulière dans l'espace (Figure 25). L'algorithme se base sur la triangulation de Delaunay à partir de laquelle chaque triangle, dont le rayon du cercle circonscrit est inférieur au paramètre alpha, est conservé. Les points de contours recherchés correspondent alors aux limites des facettes appartenant à un seul triangle. Ce principe sera abordé plus en détails par la suite<sup>6</sup>.

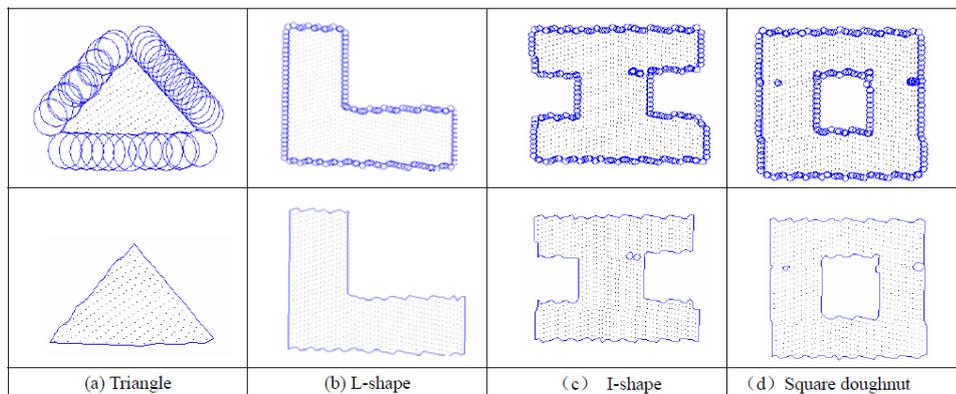


Figure 25 : Extraction des points de contour par l'algorithme alpha-shape [Shen, 2008]

Après s'être penché sur les différentes techniques de détection de contours de pierres sur images et sur nuages de points, il apparaît maintenant judicieux d'envisager plusieurs procédés permettant d'aboutir au résultat souhaité et d'identifier pour chacun d'entre eux les points difficiles afin d'en déduire le plus approprié à notre contexte d'étude.

## I.3 Les méthodes envisagées

Deux étapes étaient initialement envisagées, la première se basant sur une détection de contours à partir d'orthoimages issues du nuage de points colorisé, et la seconde, sur une détection de contours directement sur le nuage de points. Pour chacune des méthodes, nous nous sommes attachés d'une part, à identifier les points difficiles, et d'autre part, à rechercher les différents moyens d'y remédier afin de garantir par la suite un processus viable.

<sup>3</sup> [Aitelkadi et al., 2013]

<sup>4</sup> cf. II.2. Segmentation par croissance de régions

<sup>5</sup> [Eich et al., 2010]

<sup>6</sup> cf. III.1. L'algorithme alpha-shape

### I.3.1 Détection des contours à partir des orthoimages extraites du nuage de points colorisé

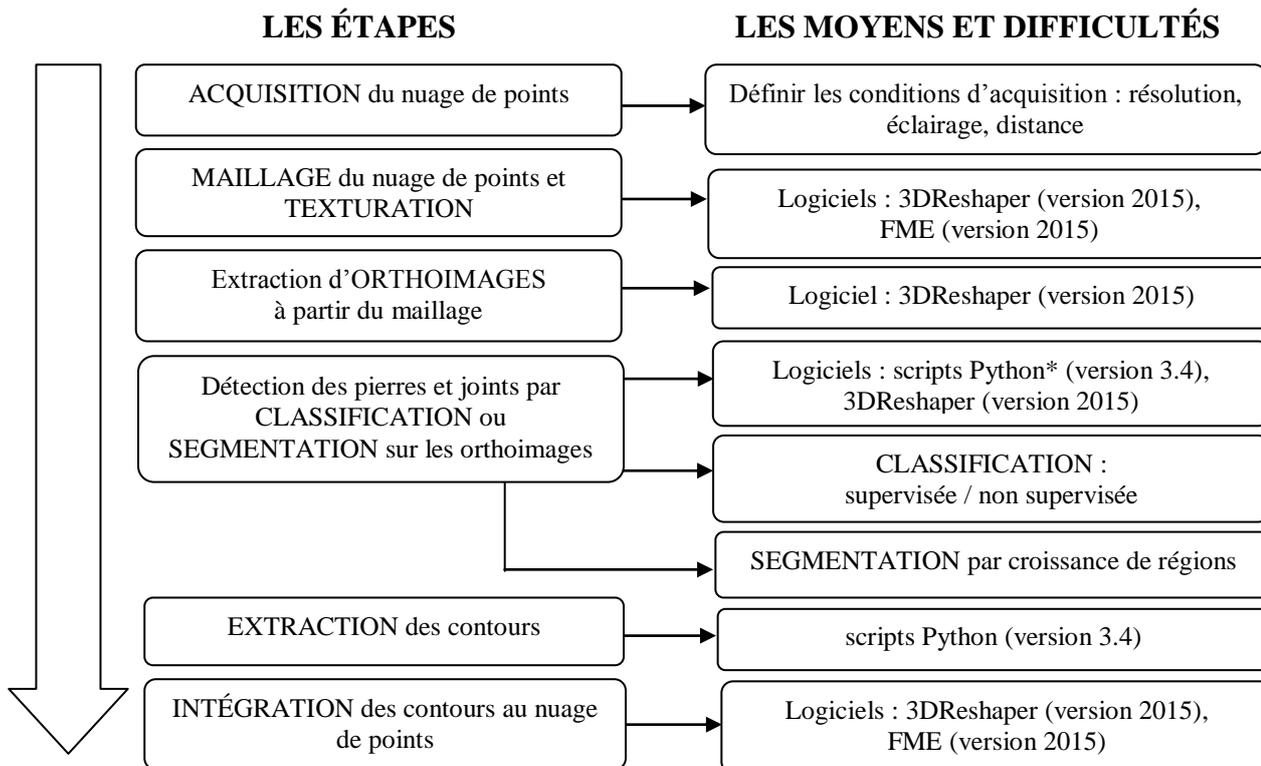


Figure 26 : Organigramme présentant la méthode de détection de contours de pierres à partir d'orthoimages extraites du nuage de points colorisé

### I.3.2 Détection des contours directement sur le nuage de points

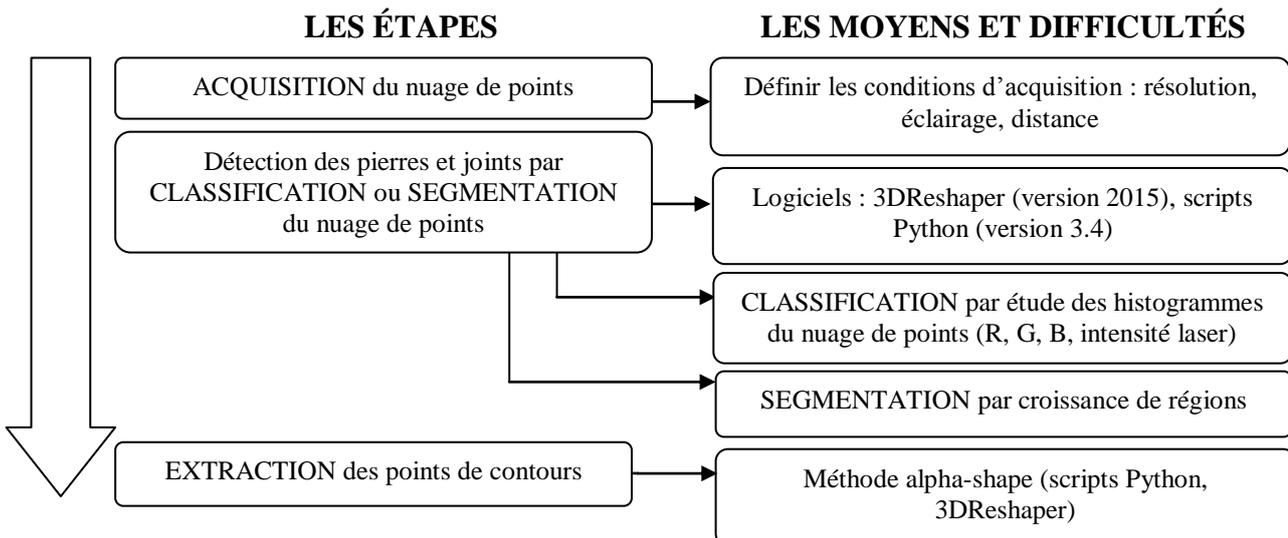


Figure 27 : Organigramme présentant la méthode de détection de contours de pierres directement sur nuage de points

\* Bibliothèques : Numpy, Pylab, Math, Scipy, Matplotlib.

### I.3.3 Justification du choix de la deuxième méthode

Après avoir établi deux processus permettant d'aboutir au résultat souhaité, nous avons décidé d'orienter la suite de notre étude sur la deuxième approche pour plusieurs raisons.

En effet, la première méthode est d'avantage chronophage que la seconde car elle nécessite plusieurs opérations avant de débiter véritablement la détection de contours. De plus,

lorsque les contours sont identifiés sur l'orthoimage, une dernière étape reste nécessaire pour intégrer ces contours sur le nuage de points initial. La deuxième méthode, quant à elle, propose un traitement basé uniquement sur le nuage de points assurant ainsi non seulement un temps de traitement plus efficace mais aussi une continuité de la donnée.

Par ailleurs, les bâtiments concernés par cette étude sont des édifices dont l'intérieur est relativement sombre (églises, châteaux). Ainsi, le traitement sur orthoimages proposé par la première méthode est limité ici car il s'appuie sur la couleur de l'image. Or, si l'image est sombre, la distinction des pierres et des joints de l'édifice est impossible. L'avantage que présente la deuxième méthode, et nous le verrons plus nettement dans la section suivante, c'est qu'il est possible de traiter le nuage de points selon la couleur mais aussi selon l'intensité laser. Comme nous avons pu le voir précédemment, nous savons qu'il est possible de détecter différents types de matériaux (et donc de différencier les joints des pierres) par l'intensité laser.

#### I.4 Présentation des nuages de points

Dans cette partie, il s'agit de présenter quelques exemples de nuages de points acquis par laser scanner terrestre C10 (*Leica*) et de présenter de manière succincte leurs caractéristiques respectives.

➤ **Présentation des deux modes de représentation du nuage de points n°1 :**

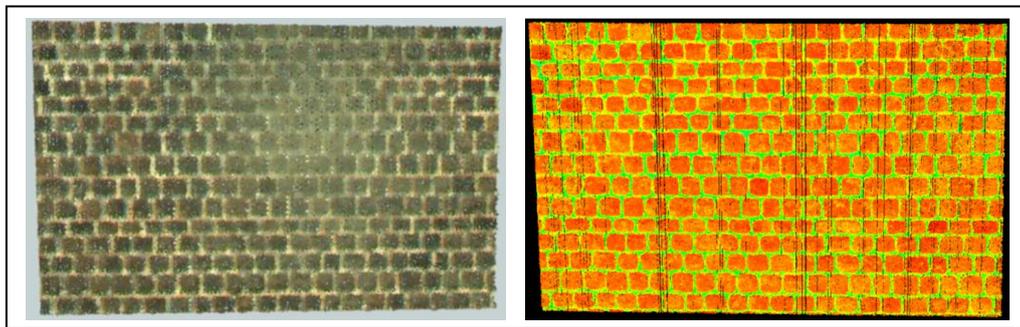


Figure 28 : Nuage de points n°1 en couleurs RGB (à gauche) et en intensité laser (à droite) [Cyclone, *ESGT*]

Sur ce nuage de points n°1 extrait d'un levé sur une cathédrale, la distinction entre les pierres et les joints est visible sur les deux modes de représentation mais apparaît toutefois de manière plus marquée sur le nuage de points en intensité laser. Cela s'explique par la présence de pluie lors de l'acquisition des photographies par le scanner (gouttes d'eau sur l'objectif) qui engendre une mauvaise colorisation des points du nuage.

➤ **Présentation des deux modes de représentation du nuage de points n°2 :**

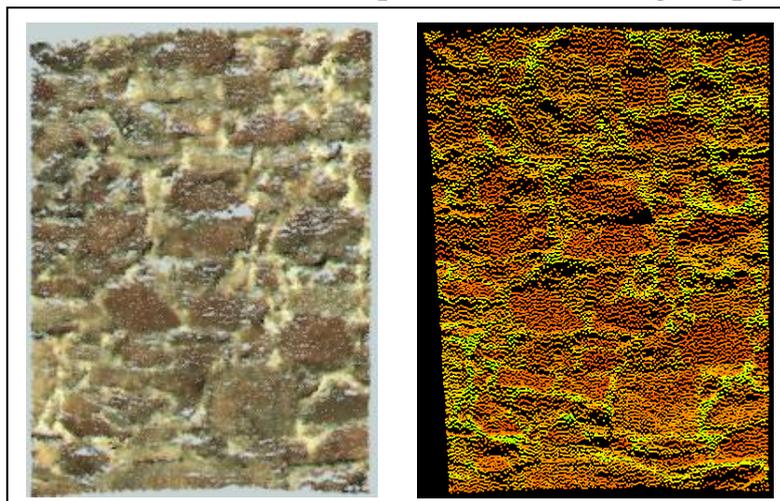


Figure 29 : Nuage de points n°2 en couleurs RGB (à gauche) et en intensité laser (à droite) [Cyclone, *GEOMAT*]

Ce nuage de points est un extrait du haut d'une tour du château médiéval de Fougères ce qui explique la discontinuité de la donnée. En effet, il existe par endroit un manque d'information dû au relief du mur et à l'inclinaison du faisceau laser du scanner 3D lors de l'acquisition. Outre ce phénomène qui engendre de petites zones d'occlusion, nous sommes toutefois en mesure de distinguer les pierres des joints sur les deux modes de représentation. L'irrégularité des formes des pierres et des joints (parfois même inexistantes) est caractéristique des édifices de l'époque médiéval.

➤ **Présentation des deux modes de représentation du nuage de points n°3 :**

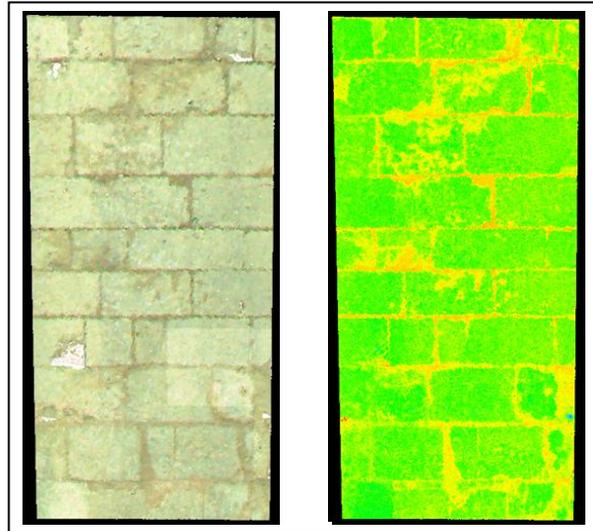


Figure 30 : Nuage de points n°3 en couleurs RGB (à gauche) et en intensité laser (à droite) [Cyclone, *ESGT*]

Il s'agit ici d'un extrait d'un nuage de points d'une cathédrale. La différenciation des pierres et des joints est visible sur les deux modes de représentation. Toutefois, nous pouvons d'ores-et-déjà nous rendre compte que la détection des pierres sera altérée par la présence du dépôt situé sur ce mur. Ainsi, une intervention humaine sera alors nécessaire pour corriger les erreurs engendrées par ce dépôt. De plus, la finesse des joints peut constituer une difficulté pour la détection des contours (une densité de points élevée est nécessaire). La recherche de lignes de contours par la transformée de Hough serait peut-être une solution.

➤ **Présentation des deux modes de représentation du nuage de points n°4 :**

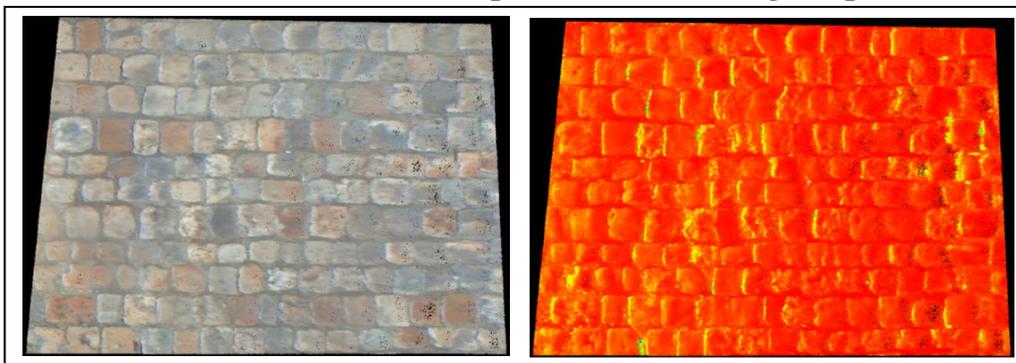


Figure 31 : Nuage de points n°4 en couleurs RGB (à gauche) et en intensité laser (à droite) [Cyclone, *ESGT*]

Sur ce nuage de points extrait d'un levé du pavage d'une rue du Vieux Mans, la distinction des pierres et des joints est uniquement visible sur le nuage de points en couleurs. Si l'œil humain ne distingue pas cette différence sur le mode intensité laser, nous pouvons alors supposer qu'un traitement algorithmique basé sur l'intensité laser ne donnera pas de meilleurs résultats.

➤ **Présentation des deux modes de représentation du nuage de points n°5 :**

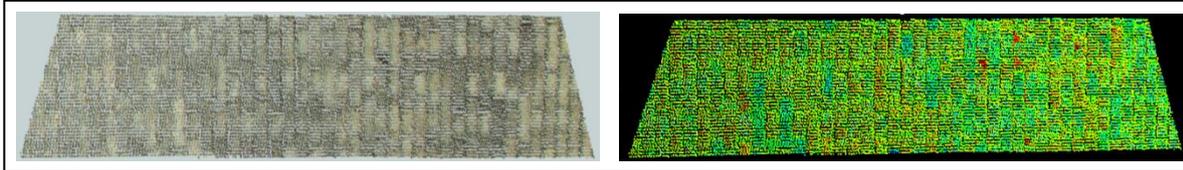


Figure 32 : Nuage de points n°5 en couleurs RGB (à gauche) et en intensité laser (à droite) [Cyclone, *GEOMAT*]

Ce nuage de points est extrait d'un lycée de Chartres. La détection de contours semble difficile pour les deux modes de représentation. L'intervention humaine est la seule solution pour tracer les contours. Aucune aide algorithmique ne peut être proposée ici. Il serait toutefois intéressant d'observer la scène avec un pas d'échantillonnage beaucoup plus petit.

➤ **Présentation des deux modes de représentation du nuage de points n°6 :**

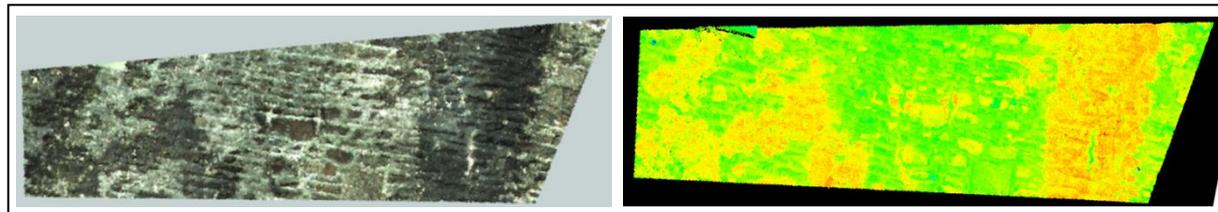


Figure 33 : Nuage de points n°6 en couleurs RGB (à gauche) et en intensité laser (à droite) [Cyclone, *GEOMAT*]

Il s'agit ici d'un deuxième extrait du levé du château médiéval de Fougères. De même que pour le nuage de points précédent, une aide algorithmique semble impossible ici car la différenciation des pierres et des joints est altérée par la présence de végétation.

Suite à toutes ces observations, nous avons décidé de poursuivre cette étude en testant notre méthode de détection de contours de pierres sur les nuages de points n°1 et n°2.

En effet, nous privilégions une détection basée sur l'intensité laser puisqu'au vu des édifices concernés par cette détection, c'est-à-dire où la luminosité est faible (églises, cathédrales, châteaux), l'intensité laser semble être la meilleure solution pour distinguer les différents matériaux composant l'édifice.

Ainsi, nous avons choisi d'établir la méthodologie prévue sur le nuage de points n°1 car il s'agit de celui où la distinction entre les pierres et les joints est la plus nette des six nuages de points présentés précédemment. Cela nous permettra d'estimer la qualité de notre méthode de détection sur un nuage de points que l'on peut considérer comme idéal.

Par ailleurs, nous avons fait le choix de tester notre méthode sur un nuage de moins bonne qualité, le nuage de points n°2, où la donnée est irrégulière et moins dense que le nuage n°1. Cela nous permettra d'identifier les limites de notre produit et d'établir les précautions à prendre lors de l'acquisition terrain pour optimiser la détection ultérieure.

## II Classification et Segmentation

Cette deuxième partie a pour objectif de présenter les méthodes de classification et de segmentation employées qui constituent les deux étapes fondamentales du processus de détection de contours de pierres sur nuages de points. En effet, la précision de la détection dépend principalement de la qualité de ces dernières comme nous le verrons plus amplement dans la suite de ce mémoire.

### II.1 Classification des nuages de points selon l'intensité laser

#### II.1.1 Définition de la classification

La classification consiste à labelliser chaque point du nuage par un numéro de classe thématique, ici deux classes nous intéressent : la classe « pierres » et la classe « joints ». Le label d'un point est indépendant de celui de ses voisins.

#### II.1.2 Étude des histogrammes selon l'intensité laser

Afin de classifier nos nuages de points, nous procédons à un seuillage simple, appelé également binarisation. Pour ce faire, nous affichons l'histogramme du nuage de points à partir des valeurs d'intensité laser de chaque point du nuage, via un script Python.

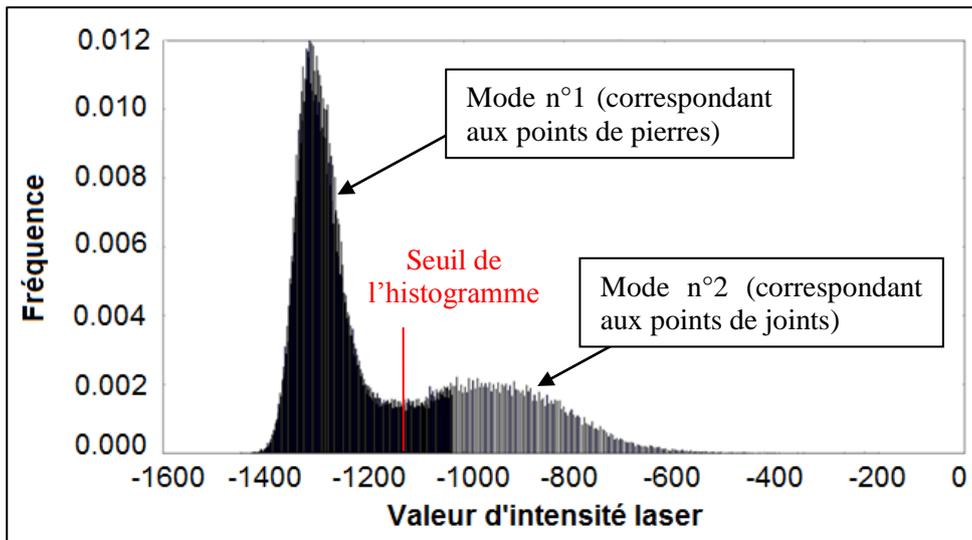


Figure 34 : Histogramme du nuage de points n°1 selon l'intensité laser [Python]

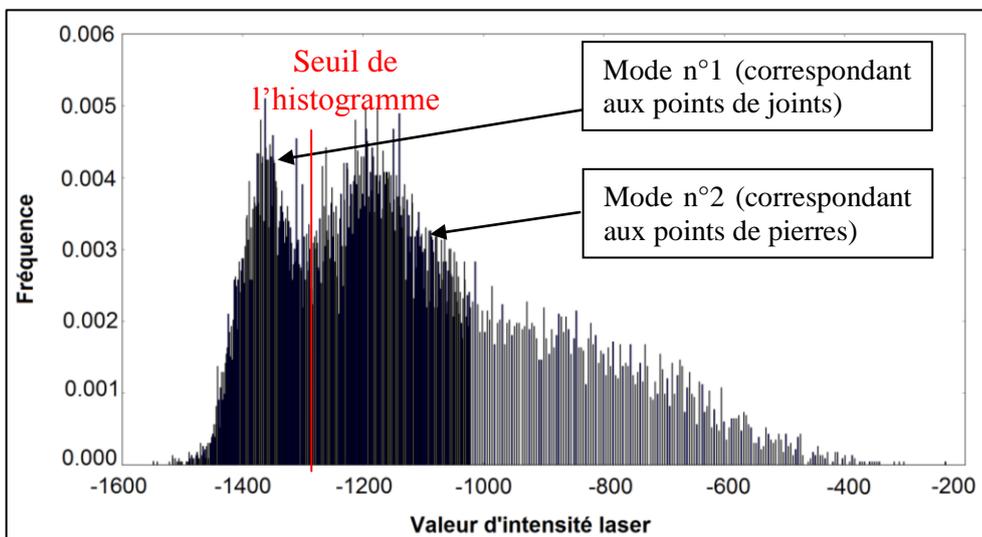


Figure 35 : Histogramme du nuage de points n°2 selon l'intensité laser [Python]

Les histogrammes sont constitués de deux modes : l'un correspond aux points de pierres et l'autre aux points de joints. Afin d'identifier à quelle classe correspond chaque mode, nous allons procéder au seuillage des nuages initiaux ce qui nous permettra de visualiser les points classifiés et ainsi de reconnaître les points de pierres des points de joints.

Pour seuiller nos nuage de points, il nous faut alors déterminer un seuil S permettant de classier les points selon deux classes.

### II.1.3 Seuillage simple

#### II.1.3.1 Le principe du seuillage simple

Si la valeur d'intensité laser du point traité est inférieure au seuil S, alors celui-ci appartient à la classe n°1, sinon il est attribué à la classe n°2.

$$I(p(x,y,z)) < S \rightarrow p(x,y,z) \in \text{classe1}$$

$$I(p(x,y,z)) > S \rightarrow p(x,y,z) \in \text{classe2}$$

Nous allons maintenant nous intéresser à automatiser la sélection du seuil S.

#### II.1.3.2 Détermination d'un seuil S : la méthode seuillage adaptatif

Pour pouvoir procéder à la classification de notre nuage, la détermination d'un seuil S est indispensable. Pour ce faire, nous avons testé la méthode par seuillage adaptatif<sup>7</sup> dont l'algorithme est présenté ci-dessous.

- *étape 1 : choix d'un seuil initial fixé par l'utilisateur*

$$S = S_{\text{init}} \quad \text{avec} \quad S : \text{seuil}$$

- *étape 2 : spécifier les classes*

$$\text{if } I(p) < S : \quad \text{avec} \quad I(p): \text{ Valeur d'intensité laser du point } p$$

$$p \in \text{classe1} \quad p : \text{ Point étudié}$$

$$\text{if } I(p) > S : \\ p \in \text{classe2}$$

- *étape 3 : calcul de la moyenne des valeurs d'intensité*

$$\mu_0 = \frac{\sum_{I=0}^S f(I) \cdot I}{\sum_{I=0}^S f(I)} \quad \mu_1 = \frac{\sum_{I=S+1}^{i-1} f(I) \cdot I}{\sum_{I=S+1}^{i-1} f(I)}$$

avec  $f(I)$  : Fréquence de l'intensité I dans le nuage de points

- *étape 4 : calcul du nouveau seuil*

$$S = \frac{\mu_0 + \mu_1}{2}$$

- *étape 5 : répétition des étapes 2 à 4 jusqu'à ce que la valeur S ne change quasiment plus*

À l'aide de l'algorithme de seuillage adaptatif, nous obtenons un seuil de -1115.82 pour le nuage de points n°1 et un seuil de -1286.905 pour le nuage de points n°2 ce qui semble correct au vu des histogrammes (Figures 34 et 35).

---

<sup>7</sup> [Dev, 2013]

#### II.1.4 Résultat de la classification selon l'intensité laser

Après application des seuillages respectifs à nos deux nuages de points, nous obtenons les résultats suivants (Figures 36, 37, 38 et 39) :

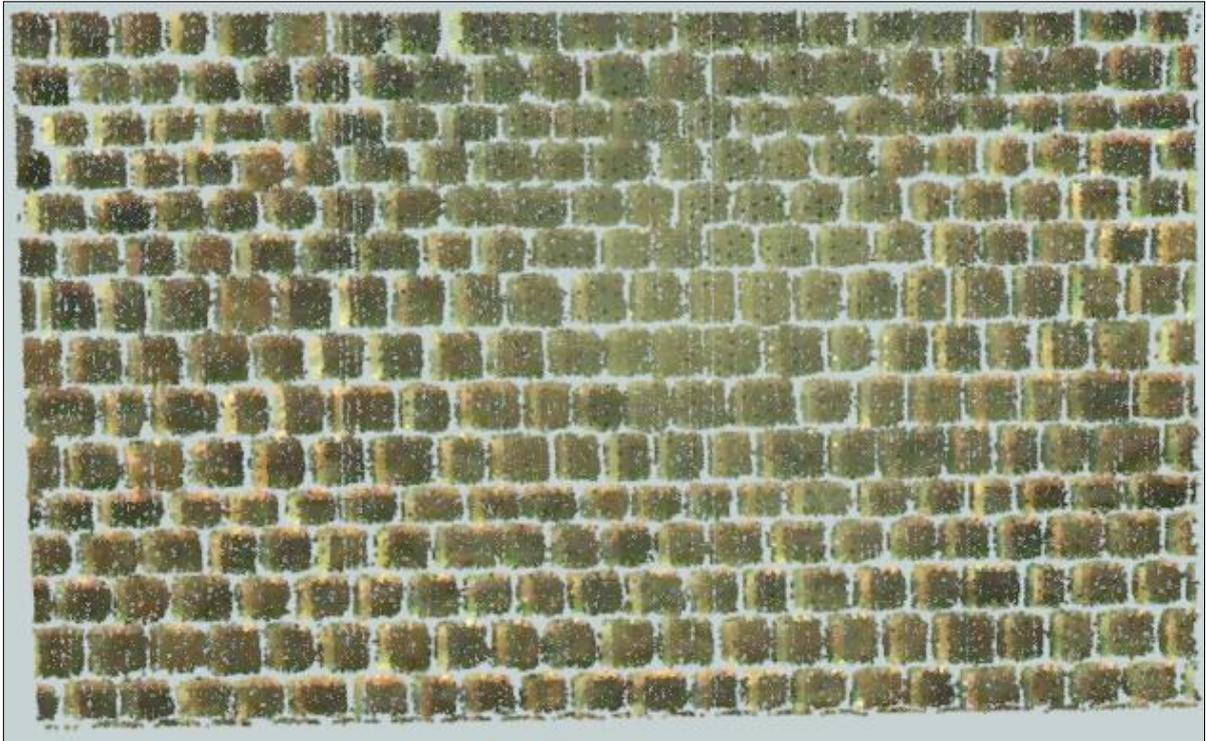


Figure 36 : Nuage de points des pierres résultant de la classification du nuage de points n°1 pour un seuil de -1115.82 [Cyclone]

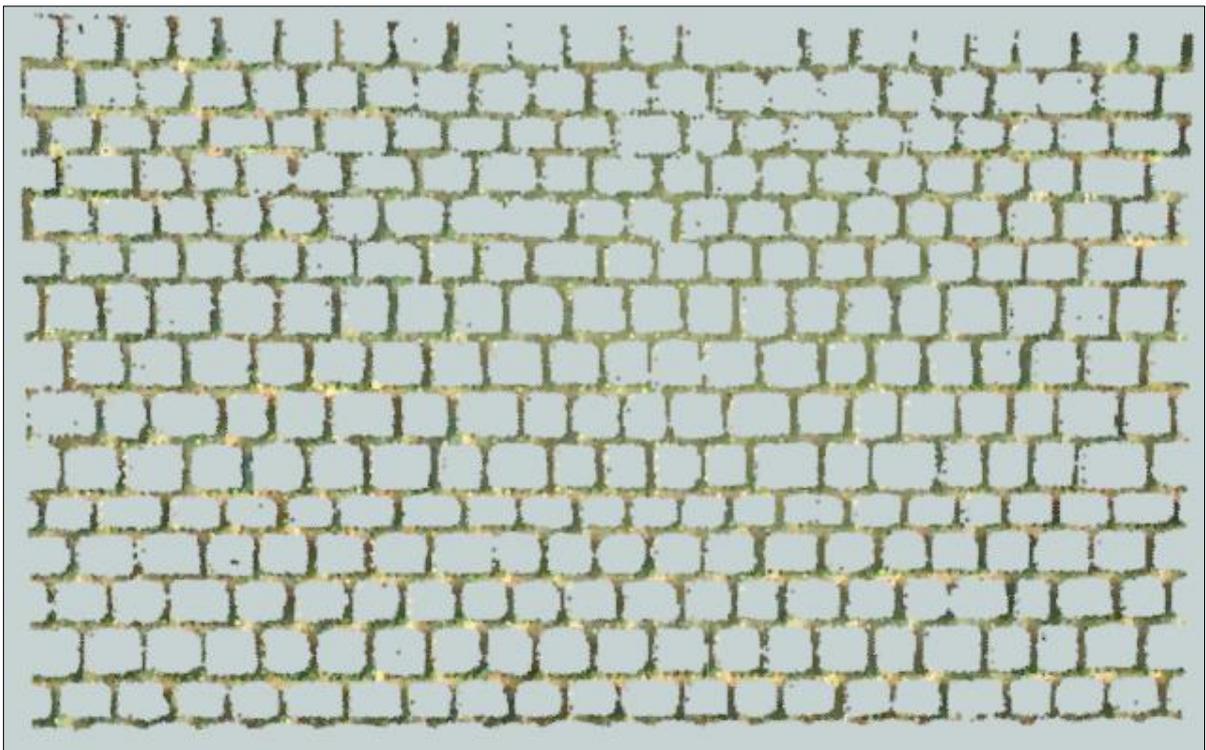


Figure 37 : Nuage de points des joints résultant de la classification du nuage de points n°1 pour un seuil de -1115.82 [Cyclone]



Figure 38 : Nuage de points des pierres résultant de la classification du nuage de points n°2 pour un seuil de -1286.905 [Cyclone]



Figure 39 : Nuage de points des joints résultant de la classification du nuage de points n°2 pour un seuil de -1286.905[Cyclone]

Les points correspondant aux pierres du nuage n°1 sont les points dont les valeurs d'intensité laser sont inférieures au seuil défini par l'algorithme par seuillage adaptatif, à savoir -1115.82 (Figure 36). Ainsi, le premier mode de l'histogramme correspond à la classe des pierres et le second à la classe des joints.

Le seuil obtenu pour le nuage de points n°2 n'est pas pertinent. Une manipulation a été ajoutée : elle consiste à retirer la queue de l'histogramme. Face à cette limite, une autre méthode telle que celle de Otsu pourrait être utilisée. À l'inverse du nuage n°1, les points de pierres du nuage n°2 sont les points dont les valeurs d'intensité laser sont supérieures au seuil de -1286.905 fixé par l'algorithme de seuillage adaptatif (Figure 38). Le premier mode de l'histogramme correspond à la classe des joints et le second à la classe des pierres.

S'agissant des classifications effectuées sur les nuages de points n°1 et n°2, celles-ci sont satisfaisantes puisque très peu de points ont fait l'objet d'une mauvaise classification. En effet, nous pouvons observer quelques points isolés sur les quatre figures (Figure 36 à 39), qui ont été attribués à la mauvaise classe.

Notons toutefois que la classification du nuage de points n°1 est de meilleure qualité que la classification du nuage n°2 pour la détection ultérieure des contours. En effet, nous visualisons facilement les contours des pierres sur les figures 36 et 37 tandis que sur les figures 38 et 39, cela est moins nette. Ces résultats sont cohérents avec les données de départ : pour un nuage de points dense où la distinction entre les points de pierres et les points de joints est nette, la classification sera de meilleure qualité que pour un nuage où les données sont discontinues et moins denses.

Ainsi, nous pouvons doré-et-déjà prédire que la détection de contours sera meilleure sur le premier nuage que sur le second. Après plusieurs tests où nous avons fait varier manuellement la valeur du seuil pour chacun des deux nuages, nous avons décidé de conserver les valeurs fixées par l'algorithme de seuillage adaptatif car les résultats correspondent au plus près de la réalité.

### II.1.4.1 Qualité de la détection par classification

Dans ce paragraphe, nous considérons le nombre de points de pierres total du nuage égal au nombre de points de pierres sélectionné manuellement (méthode initialement employée par la structure d'accueil).

- **Classification du nuage de points n°1**

➤ *Taux de bonne détection des pierres P1 :*

$$P1 = \frac{\text{nombre de points de pierres correctement détectés}}{\text{nombre de points de pierres total réel}} \times 100$$

$$P1 = \frac{161\,131}{172\,542} \times 100$$

$P1 = 93,39 \%$
-----------------

➤ *Taux de fausses alarmes P2 :*

$$P2 = \frac{\text{nombre de points de pierres incorrects détectés}}{\text{nombre de points de pierres détectés}} \times 100$$

$$P2 = \frac{5\,612}{166\,743} \times 100$$

$P2 = 3,36 \%$
----------------

➤ *Taux de détection des joints P3 :*

$$P3 = \frac{\text{nombre de points de joints correctement détectés}}{\text{nombre de points de joints total réel}} \times 100$$

$$P3 = \frac{34\,649}{40\,261} \times 100$$

$P3 = 86,06\%$
----------------

➤ *Taux de fausses alarmes P4 :*

$$P4 = \frac{\text{nombre de points de joints incorrects détectés}}{\text{nombre de points de joints détectés}} \times 100$$

$$P4 = \frac{11\,411}{46\,060} \times 100$$

$P4 = 24,77\%$
----------------

Ces statistiques nous permettent d'affirmer que la classification basée sur l'intensité laser du premier nuage de points est très satisfaisante.

- **Classification du nuage de points n°2**

➤ *Taux de bonne détection des pierres P1 :*

$$P1 = \frac{8\,631}{13\,530} \times 100$$

$P1 = 63,79\%$
----------------

➤ *Taux de fausses alarmes P2 :*

$$P2 = \frac{1\,746}{10\,377} \times 100$$

$P2 = 16,83 \%$
-----------------

➤ *Taux de bonne détection des joints P3 :*

$$P3 = \frac{8\,053}{9\,799} \times 100$$

$P3 = 82,18\%$
----------------

➤ *Taux de fausses alarmes P4 :*

$$P4 = \frac{4\,899}{12\,952} \times 100$$

$P4 = 37,82\%$
----------------

Ces taux sont de moins bonne qualité que ceux résultant de la classification du premier nuage de points. Cela s'explique par les différences des données initiales : la densité de points et la distinction des pierres et des joints sont de meilleure qualité sur le nuage n°1 que sur le nuage n°2.

## II.2 Segmentation par croissance de régions

### II.2.1 Présentation de l'algorithme

L'algorithme de croissance de régions<sup>8</sup> consiste à ajouter à chaque point germe, choisi de manière aléatoire dans le nuage, les points situés dans un rayon R fixé autour de celui-ci (en général égal à deux fois la distance moyenne entre les points du nuage) afin de former une surface-germe. Cette surface-germe doit ensuite être validée par un test vérifiant que son critère d'homogénéité est bien en cohérence avec celui de l'élément recherché dans le nuage de points. Une fois le test validé, la surface-germe s'accroît en se voyant attribuer de nouveaux points selon le principe suivant : pour chaque point constituant la surface-germe, on ajoute un à un les points situés dans un rayon R autour de celui-ci et qui vérifie le test d'homogénéité, agrandissant ainsi la surface-germe initiale. Le processus se poursuit de manière itérative jusqu'à ce qu'aucun point du nuage ne puisse être ajouté à la surface-germe.

---

<sup>8</sup> [Aitelkadi et al., 2013]

**OBJECTIF:** Segmenter le nuage de points de manière à isoler chaque pierre

**LES ÉTAPES :**

**Étape 1 : Choix des germes de manière aléatoire**

- *Contrainte d'intensité* mise en place pour que les germes choisis soient des points de pierres (et non des joints) : nécessite une étude préalable de l'histogramme du nuage de point selon l'intensité laser.
- *Nombre de germes* défini par l'utilisateur : celui-ci doit être supérieur au nombre de pierres afin de palier au fait que plusieurs germes sont susceptibles d'être situés sur une même pierre.

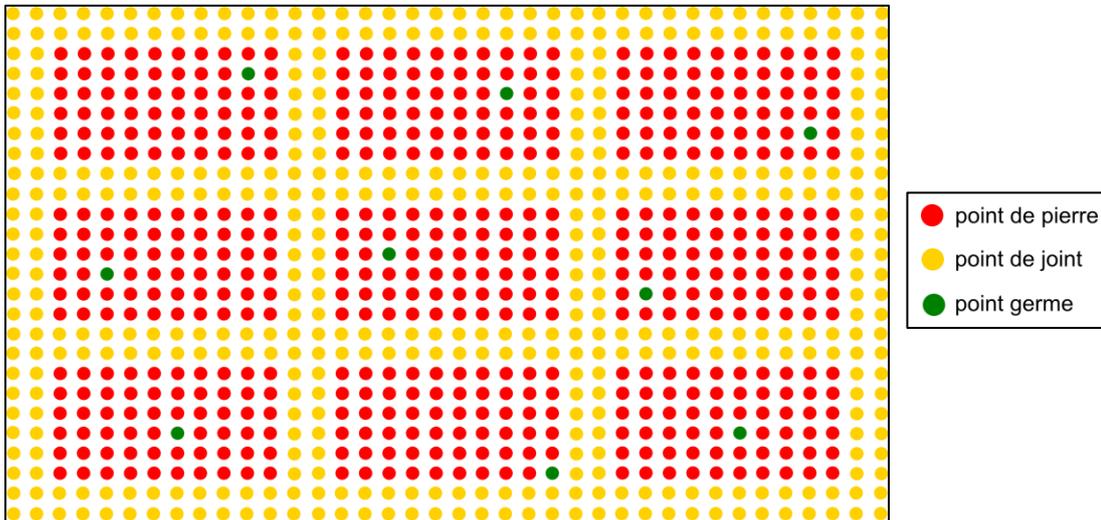


Figure 40 : Schéma de la sélection des germes sur un nuage de points représentant un mur

**Étape 2 : Évolution des germes en surfaces-germes initiales**

- *Rayon des surfaces-germes initiales  $R$*  défini par l'utilisateur : doit être égale à deux fois la distance moyenne entre les points du nuage.
- *Formation des surfaces-germes initiales* : tous les points autour de chaque germe, situés à une distance inférieure au rayon  $R$  sont sélectionnés pour former les surfaces-germes initiales respectives.



Figure 41 : Schéma de l'évolution des germes en surfaces-germes initiales

### Étape 3 : Validation des surfaces-germes initiales

- Critère d'homogénéité  $C1$  basé sur l'intensité laser doit être validé pour continuer le processus de détection :

$$C1 = \text{Variance}(\text{intensité laser des points de la surface-germe}) < \text{seuil}$$

Si celui-ci n'est pas validé, alors l'algorithme retourne à l'étape 1 pour choisir un nouveau germe jusqu'à ce que la surface-germe soit validée.

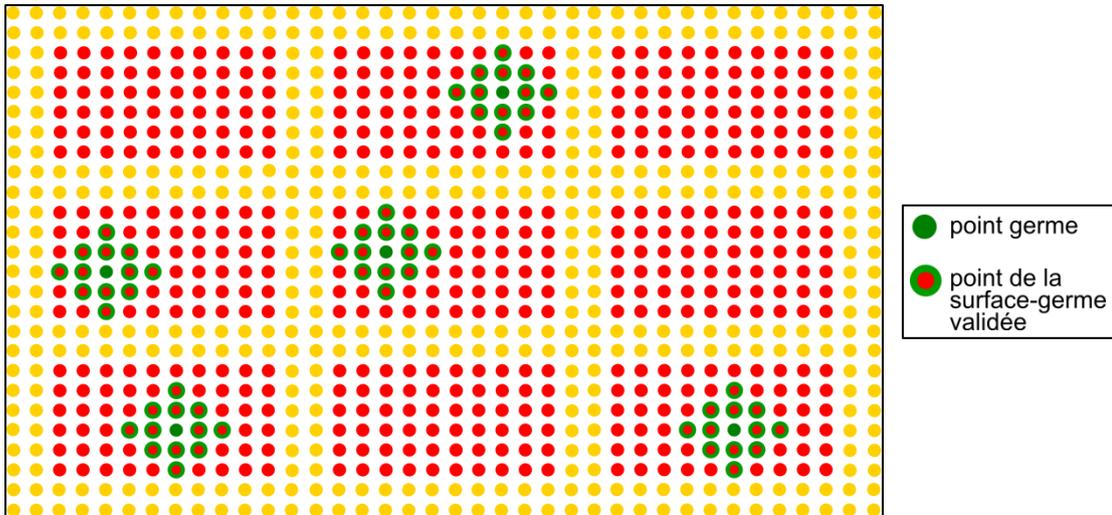


Figure 42 : Schéma des surfaces-germes validées

### Étape 4 : Croissance de régions à partir des surfaces-germes validées

- Traitement itératif :

itérations jusqu'à ce qu'il n'y est plus de points à ajouter

- Création d'un cercle de rayon  $R$  pour chaque point de la surface-germe validée tel que chaque point constitue le centre de ce cercle
- Calcul du nouveau critère d'homogénéité  $Cn < \text{seuil}$   
 $Cn = \text{Variance}(\text{intensité laser des points de la surface courante}) < \text{seuil}$
- Ajout des points un à un si le critère d'homogénéité  $Cn$  est validé.

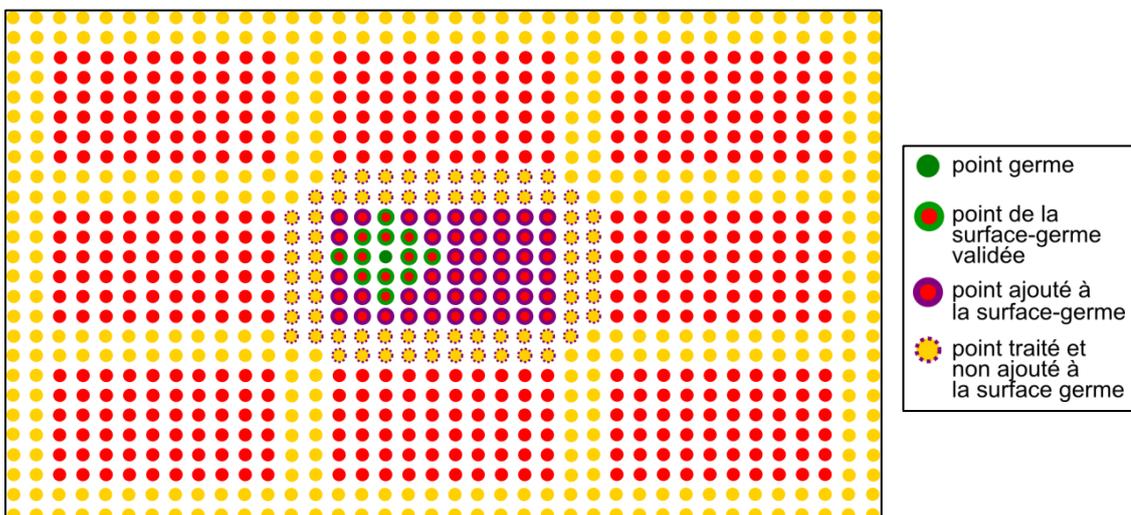
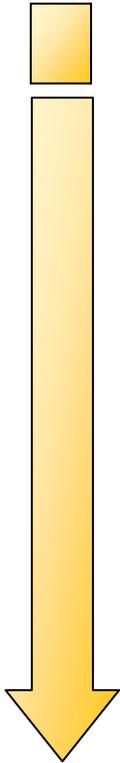


Figure 43 : Schéma de la croissance d'une surface-germe validée



Ainsi, pour des surfaces-germes validées dans chaque pierre présente au sein du nuage de points, le résultat obtenu par l'algorithme de croissance de régions est le suivant :

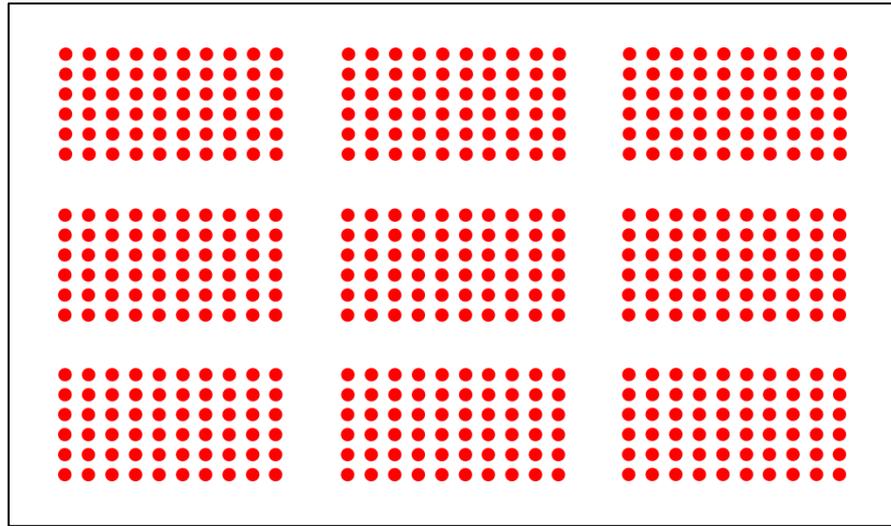


Figure 44 : Schéma du nuage de points résultant de la segmentation par croissance de régions

## II.2.2 Critère d'homogénéité et seuil de validation

La validation du critère d'homogénéité est une étape importante et délicate puisque c'est à partir de celle-ci qu'un point va être ajouté ou non à la liste des points de pierres.

Ainsi, il est important de savoir à quoi correspond le critère d'homogénéité pour pouvoir fixer un seuil de validation adéquate. Pour ce faire, nous avons défini le critère d'homogénéité comme étant égal à la variance des intensités laser des points de la surface courante :  $Var(X) = \frac{1}{n} \sum_{i=1}^n (x_i - m)^2$  et nous avons fixé la valeur du seuil de proche en proche jusqu'à obtenir un résultat correct. Cette méthode empirique nous a permis d'aboutir à un résultat qui reste exploitable pour la suite de notre étude.

## II.2.3 Résultats de la segmentation par croissance de régions

Pour des raisons de rapidité de traitement algorithmique, nous avons testé cette méthode sur un extrait du nuage de points n°1 (Figure 45). En effet, sur le nuage n°1 entier, le traitement s'arrêtait brutalement et ne nous permettait pas d'obtenir de résultats. En revanche, nous avons pu tester la segmentation sur le nuage de points n°2 (Figure 46) car celui-ci possède moins de points que le nuage n°1.

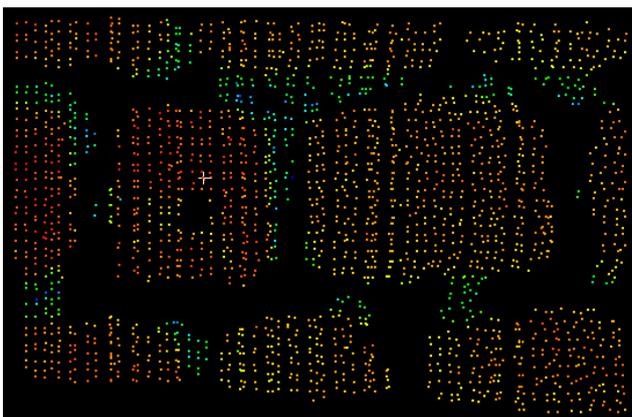
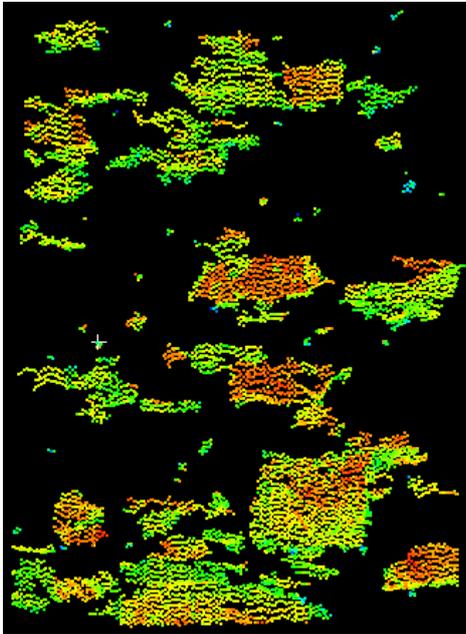


Figure 45 : Nuage résultant de la segmentation par croissance de régions du nuage de points n°1 [Cyclone]



À première vue, nous pouvons dores-et-déjà nous rendre compte que la segmentation par croissance de régions est de moins bonne qualité que la classification testée précédemment. Les résultats de cette segmentation pourraient être améliorés par l'étude plus approfondie de la validation du critère d'homogénéité.

Figure 46 : Nuage résultant de la segmentation par croissance de régions du nuage de points n°2 [Cyclone]

## II.2.4 Qualité de la détection par croissance de régions

De la même manière que pour la classification, nous considérons le nombre de points de pierres total du nuage égal au nombre de points de pierres sélectionné manuellement (méthode initialement employée par la structure d'accueil).

- **Segmentation du nuage de points n°1**

➤ *Taux de bonne détection des pierres P1 :*

$$P1 = \frac{\text{nombre de points de pierres correctement détectés}}{\text{nombre de points de pierres total réel}} \times 100$$

$$P1 = \frac{2\,210}{2\,235} \times 100$$

$P1 = 98,88\%$
----------------

➤ *Taux de fausses alarmes P2 :*

$$P2 = \frac{\text{nombre de points de pierres incorrects détectés}}{\text{nombre de points de pierres détectés}} \times 100$$

$$P2 = \frac{208}{2\,418} \times 100$$

$P2 = 8,60 \%$
----------------

Ces statistiques révèlent bien ce que nous observons sur la figure 45 : la majorité des points de pierres ont été détecté (taux de bonne détection = 98,88%), mais trop de points de joint ont été sélectionnés (taux de fausses alarmes = 8,60%). Ces résultats confortent notre impression de départ : une étude plus approfondie de la validation du critère d'homogénéité est nécessaire pour améliorer la segmentation.

- **Segmentation du nuage de points n°2**

➤ *Taux de bonne détection des pierres P1 :*

$$P1 = \frac{5\,616}{11\,589} \times 100$$

$P1 = 48,46\%$
----------------

➤ *Taux de fausses alarmes P2 :*

$$P2 = \frac{555}{6\,171} \times 100$$

$P2 = 8,99\%$
---------------

La qualité de la détection des points de pierres par la méthode de segmentation par croissance de régions est à améliorer. En effet, les taux sont relativement bas et ne nous permettent pas d'assurer pour la suite de notre étude une reconnaissance de contours satisfaisante.

### II.3 Comparaison des deux méthodes

	<b>Classification basée sur l'intensité laser</b>	<b>Segmentation par croissance de régions</b>
<b>Avantages</b>	<ul style="list-style-type: none"> <li>- rapidité du traitement</li> <li>- simple à mettre en œuvre</li> </ul>	<ul style="list-style-type: none"> <li>- temps de traitement assez long</li> <li>- complexe à mettre en œuvre</li> </ul>
<b>Inconvénients</b>	<ul style="list-style-type: none"> <li>- avoir un nuage de points où l'on distingue nettement les pierres et les joints en intensité laser</li> <li>- savoir analyser l'histogramme du nuage de points pour pouvoir fixer un seuil initial à l'algorithme de seuillage adaptatif</li> </ul>	<ul style="list-style-type: none"> <li>- avoir un nuage de points où l'on distingue nettement les pierres et les joints en intensité laser</li> <li>- avoir une densité de points au niveau des joints supérieure au rayon R (égal à deux fois la distance moyenne entre chaque point)</li> <li>- prévoir un nombre de germes supérieur au nombre de pierres à détecter pour palier au cas où plusieurs germes se situent sur une même pierre (peut également être fait de proche en proche).</li> </ul>

Tableau 5 : Comparaison avantages/inconvénients des deux méthodes

Ainsi, la classification est la plus satisfaisante des deux méthodes testées au regard de la qualité de détection des points de pierres. Les résultats de la segmentation par croissance de régions ne sont actuellement pas assez satisfaisants pour assurer une reconnaissance des contours des pierres optimale. Toutefois, cette segmentation pourrait produire de meilleurs résultats si un approfondissement de la méthode de validation du critère d'homogénéité était réalisé.

Nous allons maintenant nous pencher sur les procédés permettant d'aboutir à l'objectif de notre étude : l'extraction des points de contours de pierres sur nuages de points.

### III Extraction des points de contours

#### III.1 L'algorithm alpha-shape

##### III.1.1 Présentation de l'algorithm

L'algorithm alpha-shape permet la reconnaissance de formes g n r es par des ensembles de points. Gr ce   cette m thode bas e sur la triangulation de Delaunay, nous sommes alors en mesure d'extraire les points de contours de ces ensembles de points.

La fonction permettant de cr er la triangulation de Delaunay en langage python ne prend en arguments que des points d finis par deux coordonn es. Ainsi, non seulement une projection sur un plan est n cessaire mais  tant donn  que nos points actuels constituent des triplets de coordonn es, il est  galement n cessaire de red finir notre syst me de coordonn es afin que chaque point soit d fini par un doublet.   la derni re  tape du traitement alpha-shape, nous attribuerons les coordonn es 3D r sultant de la projection   chaque point.

####  tape 1 : Projection du nuage de points sur un plan et pr cision des donn es

- Calcul de la normale au plan sur lequel on va projeter le nuage de points : M thode des Moindres Carr s Lin aires.

- o  quation du plan :  $Z = aX + bY + c \Leftrightarrow aX + bY - Z + c = 0$

- o normale au plan :  $\Leftrightarrow \vec{n}(a, b, -1)$

- o on pose le syst me lin aire suivant :  $AX=B$ , avec :

$$A \begin{pmatrix} X_1 \text{ nuage} & Y_1 \text{ nuage} & 1 \\ \dots & \dots & \dots \\ X_n \text{ nuage} & Y_n \text{ nuage} & 1 \end{pmatrix} : \text{matrice } n * 3 \text{ des coefficients devant les 3 inconnues et pour les } n \text{ observations du probl me,}$$

$$X \begin{pmatrix} a \\ b \\ -1 \end{pmatrix} : \text{vecteur contenant les 2 inconnues et la valeur -1 qui correspondent au vecteur de la normale } n \text{ au plan,}$$

$$B \begin{pmatrix} Z_1 \text{ nuage} \\ \dots \\ Z_n \text{ nuage} \end{pmatrix} : \text{vecteur des } n \text{ observations}$$

- o On rend le vecteur  $\vec{n}$  unitaire :  $\vec{n} = \left( \frac{a}{\|\vec{x}\|} \quad \frac{b}{\|\vec{x}\|} \quad \frac{-1}{\|\vec{x}\|} \right)$

- Projection des points sur le plan   l'aide du vecteur normal unitaire  $\vec{n}$  et d'un point  $p(0,0,0)$  appartenant au plan :

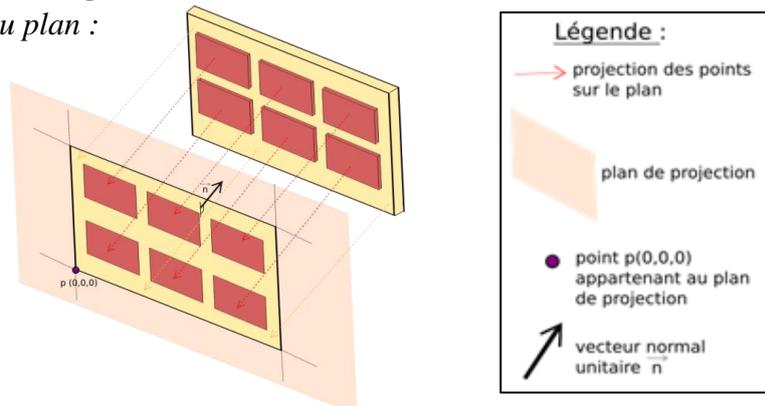


Figure 47 : Sch ma de la projection du nuage de points sur un plan

- *Précision des données* : La projection s'effectue sur un plan parallèle au plan épousant le mieux la forme de notre nuage de points. Ainsi, la précision des données semble suffisamment peu altérée pour ne pas gêner notre étude étant donné la précision demandée (3cm).

## Étape 2 : Calcul d'un nouveau système de coordonnées à deux variables

- *Définition d'un premier vecteur* : formé par les deux premiers points du nuage nommés p1 et p2.
- *Création d'un deuxième vecteur orthogonal au premier* :

- o  $\vec{v1} \cdot \vec{v2} = 0$  :

$$\begin{cases} \vec{v1} = \overline{p1p2} \\ \vec{v2} = \overline{p1p3} \end{cases} \Rightarrow \begin{cases} \vec{v1} = \begin{pmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{pmatrix} \\ \vec{v2} = \begin{pmatrix} x_3 - x_1 \\ y_3 - y_1 \\ z_3 - z_1 \end{pmatrix} \end{cases}$$

$$\Leftrightarrow (x_2 - x_1)(x_3 - x_1) + (y_2 - y_1)(y_3 - y_1) + (z_2 - z_1)(z_3 - z_1) = 0$$

- o *équation du plan* :

$$z_3 = ax_3 + by_3 + c$$

$$\Leftrightarrow ax_3 + by_3 + c - z_3 = 0, \quad \text{avec } a, b, c \text{ déterminés précédemment par la méthode des moindres carrés linéaires}$$

- o *Résolution du système d'équation* :

$$\begin{cases} (x_2 - x_1)(x_3 - x_1) + (y_2 - y_1)(y_3 - y_1) + (z_2 - z_1)(z_3 - z_1) = 0 \\ ax_3 + by_3 + c - z_3 = 0 \end{cases}$$

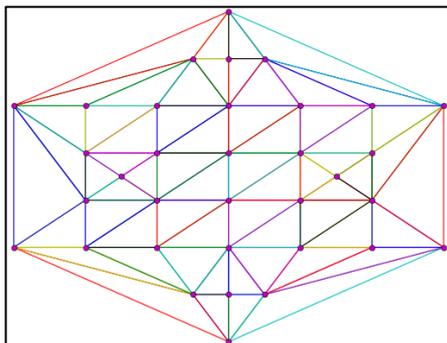
$$\Leftrightarrow x_3 = -\frac{(y_2 - y_1)(y_3 - y_1) - (x_2 - x_1)x_1 + (z_2 - z_1)(by_3 + c - z_1)}{x_2 - x_1 + az_2 - az_1}$$

Il existe une infinité de vecteurs vérifiant ce système d'équation, c'est pourquoi nous fixons une des coordonnées :  $y_3 = 0$ .

À l'aide d'un traitement algorithmique, nous résolvons le système d'équations et rendons unitaires les deux vecteurs v1 et v2. Nous avons ainsi créé un système orthonormé de deux coordonnées nécessaire à l'utilisation de la fonction générant la triangulation de Delaunay sous python.

## Étape 3 : Calcul de la triangulation de Delaunay sur le nuage de points

- Grâce à la fonction python "*triang.delaunay(x,y)*" nous établissons la triangulation de Delaunay.



Critère de la triangulation de Delaunay :

Aucun des cercles circonscrits aux triangles ne doit contenir d'autres points du nuage.

Les triangles doivent ainsi être les plus équilatéraux possibles

Figure 48 : Triangulation de Delaunay sur un nuage de points [Python]

#### Étape 4 : Calcul des rayons des cercles circonscrits de l'ensemble des triangles issus de la triangulation de Delaunay

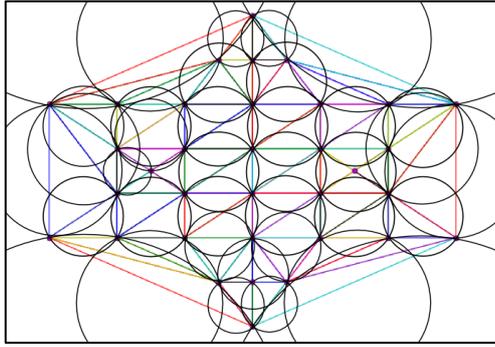


Figure 49 : Cercles circonscrits des triangles issus de la triangulation de Delaunay [Python]

#### Étape 5 : Fixation du paramètre alpha

- *Le paramètre alpha* correspond au rayon d'un cercle qui circule autour des ensembles de points. (Tous les points intersectés avec le cercle de rayon alpha sont alors reliés entre eux et constituent les points de contours de la forme détectée.)

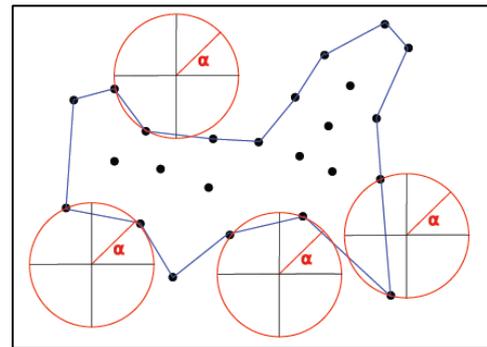


Figure 50 : Schéma du principe de l'alpha-shape [Eich et al., 2010]

#### Étape 6 : Conservation des triangles dont le rayon du cercle circonscrit est inférieur à alpha

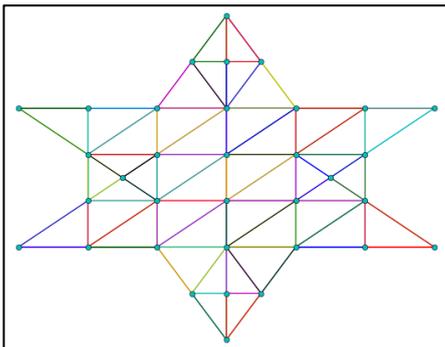


Figure 51 : Triangles conservés [Python]

#### Étape 7 : Identification des contours

- *Les arêtes des triangles correspondant au contour* sont celles qui appartiennent à un seul triangle. Une fois ces arêtes identifiées, nous pouvons alors afficher les contours.

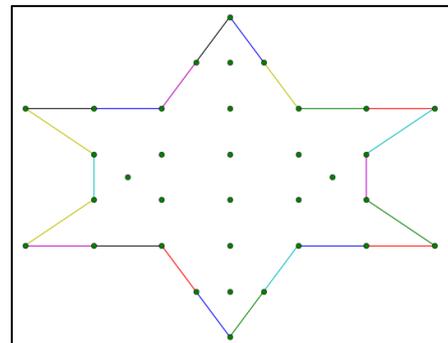


Figure 52 : Contours obtenus par la méthode alpha-sape [Python]

### III.1.2 Résultats obtenus par la méthode alpha-shape

#### III.1.2.1 Test sur un extrait du nuage de points n°1

Pour des raisons de rapidité de traitement algorithmique, nous avons testé cette méthode sur un extrait du nuage de points n°1.



Figure 53 : Extrait du nuage de points initial sur lequel la méthode alpha-shape va être testée [Cyclone]

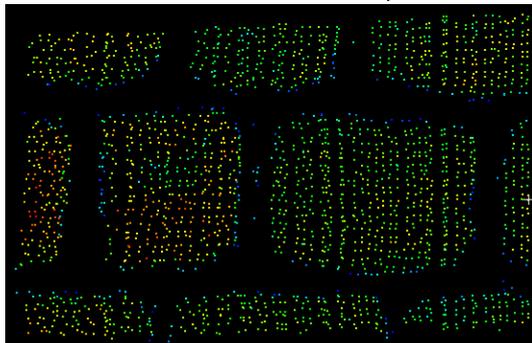
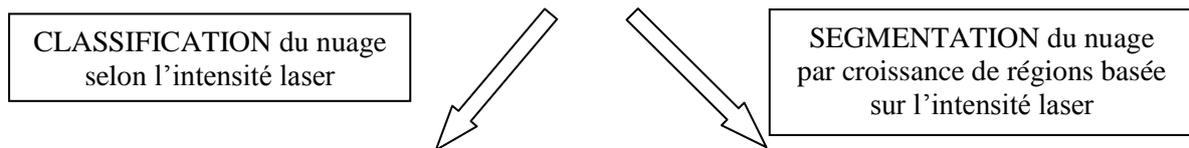


Figure 54 : Nuage de points n°1 résultant de la classification [Cyclone]

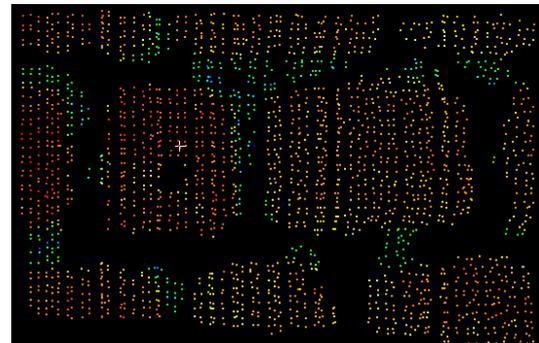


Figure 55 : Nuage de points n°1 résultant de la segmentation par croissance de régions [Cyclone]

#### III.1.2.1.1. Application au nuage résultant de la classification du nuage de points n°1

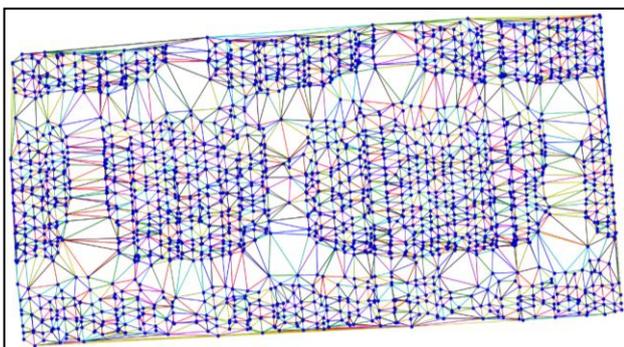


Figure 56 : Triangulation de Delaunay sur l'extrait du nuage de points n°1 classifié et projeté [Python]

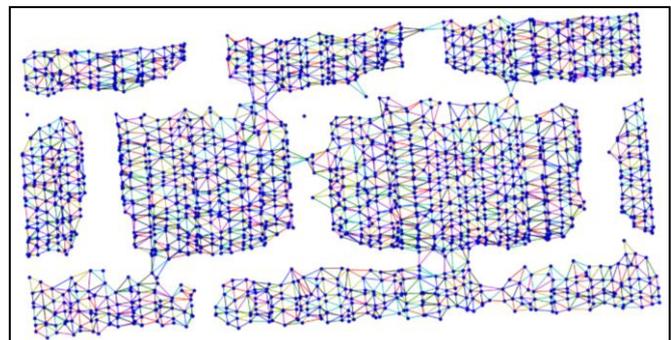


Figure 57 : Conservation des triangles dont le cercle circonscrit est inférieur à alpha 5mm [Python]

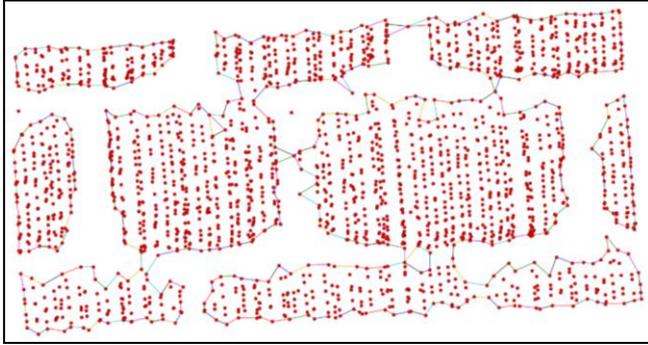


Figure 58 : Identification des contours pour un alpha de 5mm [Python]

La détection des contours est faussée par la mauvaise classification de certains points. En effet, la présence de points isolés dans le nuage considérés comme des points de pierres alors que ce sont des points de joints, altère la détection des contours.

C'est pourquoi, nous décidons de filtrer le nuage de points sous le logiciel 3DReshaper afin de supprimer les points isolés.

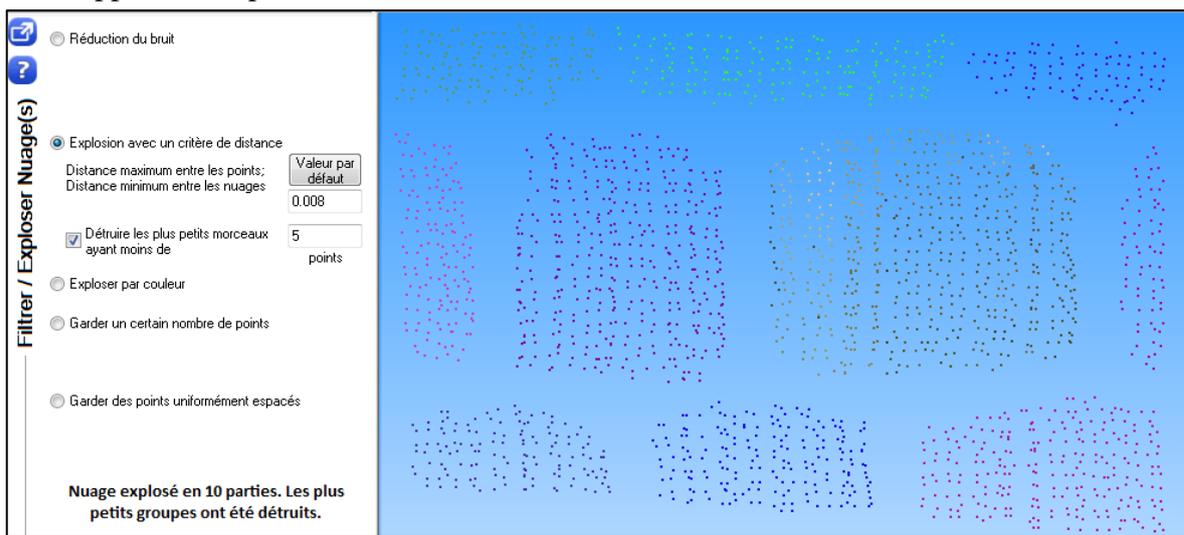


Figure 59 : Filtrage de l'extrait du nuage de points n°1 classifié [3DReshaper]

Ce filtrage a permis de supprimer 17 points isolés. Par ailleurs, lors de cette opération, le nuage de points a été explosé en 10 parties, chacune correspondante à une pierre comme en atteste la figure 59.

Pour la suite de notre traitement algorithmique sous python, nous fusionnons le nuage de points sous 3Dreshaper avant de l'exporter au format pts.

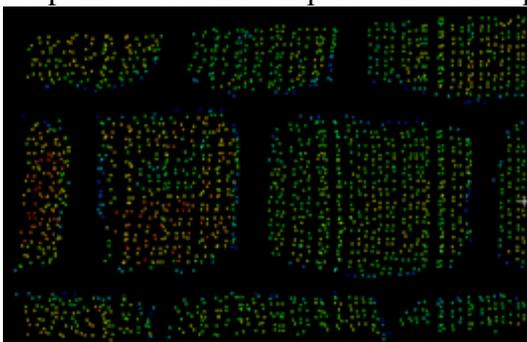


Figure 60 : Extrait du nuage de points n°1 classifié filtré en intensité laser [Cyclone]

Ainsi, nous allons maintenant tester la méthode de détection de contours alpha-shape sur le nuage de points classifié et filtré.

### III.1.2.1.2. Application au nuage de points n°1 classifié et filtré

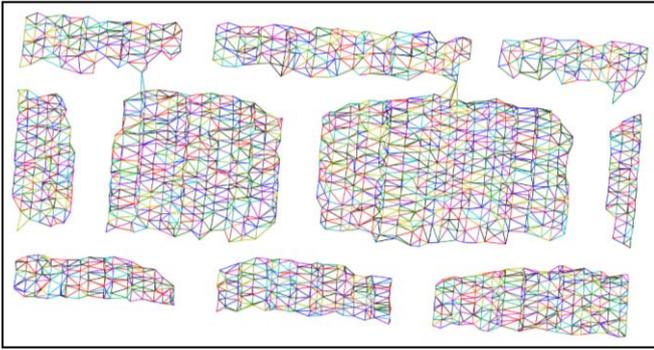
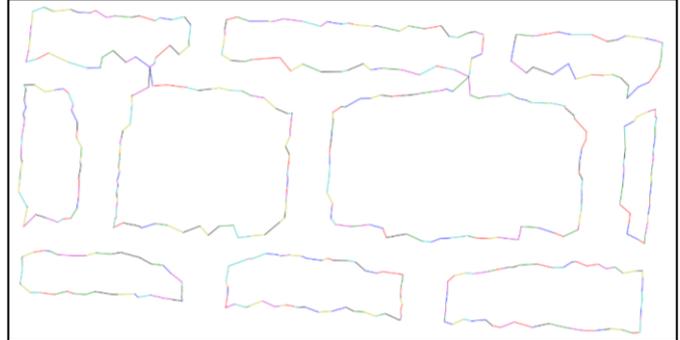


Figure 62 : Identification des contours pour un alpha de 5mm sur l'extrait du nuage de points n°1 classifié et filtré [Python]

Figure 61 : Conservation des triangles dont le cercle circonscrit est inférieur à alpha 5mm sur l'extrait du nuage de points n°1 classifié et filtré [Python]



Nous pouvons qualifier cette détection de contours obtenue après filtrage du nuage de points classifié, de satisfaisante au vu de la précision demandée (3cm en absolu). En effet, ce résultat répond à nos attentes même si un traitement manuel resterait nécessaire pour rectifier les erreurs locales et notamment les jonctions de certaines pierres entre elles.

### III.1.2.1.3. Application au nuage résultant de la segmentation par croissance de régions du nuage de points n°1

Pour appliquer notre méthode alpha-shape et obtenir un résultat satisfaisant, nous sommes contraints de nettoyer à la main notre nuage de points en supprimant les points de joints. En effet, un filtrage automatique sous 3Dresaper ne nous donnera pas un résultat satisfaisant puisqu'il ne s'agit pas ici de supprimer des points isolés mais des portions de points de joints situées entre des points de pierres. Notons que sur un nuage de points de plus grande taille, ce nettoyage représente un travail long et fastidieux.

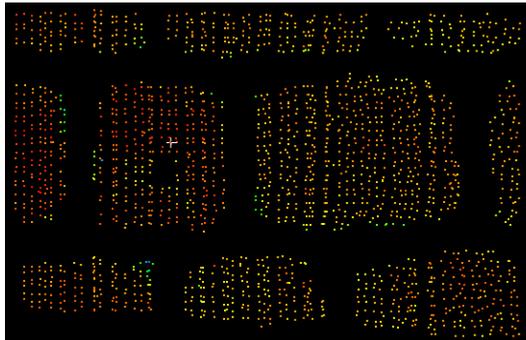


Figure 63 : Extrait du nuage de points n°1 segmenté et nettoyé [Cyclone]

Figure 64 : Triangulation de Delaunay sur l'extrait du nuage de points n°1 segmenté, nettoyé et projeté [Python]

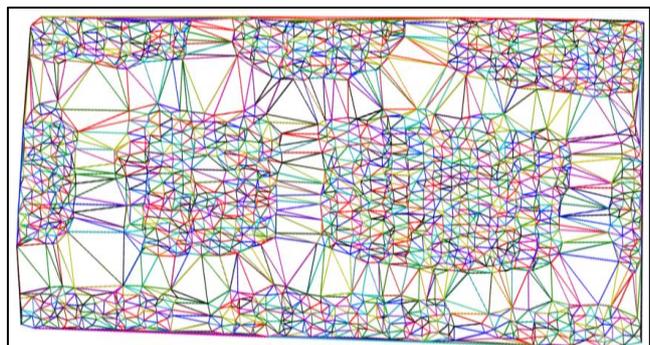
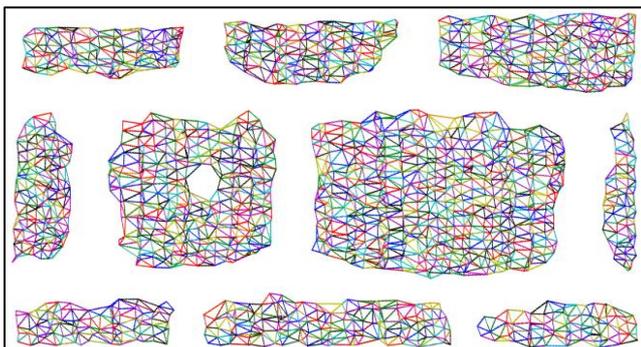


Figure 65 : Conservation des triangles dont le cercle circonscrit est inférieur à alpha de 6mm [Python]

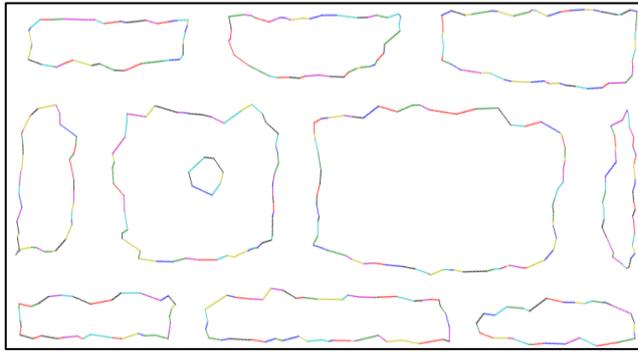


Figure 66 : Identification des contours pour un alpha-shape de 6mm sur l'extrait du nuage de points n°1 segmenté et nettoyé [Python]

Le résultat obtenu par la méthode alpha-shape sur ce nuage de points segmenté et nettoyé est très satisfaisant. En effet, les contours des pierres sont bien détectés et il n'existe qu'un seul contour à supprimer correspondant au trou formé par la non-sélection de certains points de pierres lors de la segmentation.

#### III.1.2.1.4. Qualité de la détection de contours par notre algorithme alpha-shape en langage python

Nuage de points n°1 classifié et filtré	Nuage de points n°1 segmenté
<p><i>Taux de bonne détection des contours P5 :</i></p> $P5 = \frac{\text{nb de contours correctement détectés}}{\text{nb de contours total}}$ $P5 = \frac{8}{10}$ <div style="border: 1px solid black; padding: 2px; width: fit-content;">P5 = 80 %</div>	<p><i>Taux de bonne détection des contours P5 :</i></p> $P5 = \frac{\text{nb de contours correctement détectés}}{\text{nb de contours total}}$ $P5 = \frac{10}{10}$ <div style="border: 1px solid black; padding: 2px; width: fit-content;">P5 = 100%</div>
<p><i>Taux de fausses alarmes P6 :</i></p> <p>Tous les contours détectés correspondent à un contour réel. Seul deux pierres ont été jointes par un point commun à deux reprises qui nécessitera une correction manuelle.</p>	<p><i>Taux de fausses alarmes P6 :</i></p> $P6 = \frac{\text{nb de contours incorrect détectés}}{\text{nb de contours détecté}}$ $P6 = \frac{1}{10}$ <div style="border: 1px solid black; padding: 2px; width: fit-content;">P6 = 10%</div>

Tableau 6 : Qualité de détection des contours par l'algorithme en langage python du nuage n°1

Dans les deux cas (nuage classifié et filtré/nuage segmenté et nettoyé), nous pouvons considérer que la détection des contours est correcte. En effet, nous distinguons la présence des dix pierres sur les deux résultats. Un traitement manuel reste toutefois nécessaire pour supprimer les quelques anomalies présentent sur les deux résultats.

S'agissant des contours détectés sur le nuage de points classifié et filtré, la faible largeur du joint a entraîné la jonction de deux pierres à deux reprises. En effet, un point a été malencontreusement considéré comme étant un point commun à deux contours de pierres ce qu'il conviendra de rectifier manuellement par l'utilisateur.

Concernant la détection des contours du nuage de points segmenté et nettoyé, un contour a été dessiné à l'intérieur d'un autre. Ce phénomène est dû à la non-sélection de certains points au centre d'une pierre formant un trou qui a été considéré par notre algorithme comme étant un contour. Ceci pourrait être évité par la mise en place d'un contrôle de position des

contours par notre algorithme : si un contour est contenu dans un autre alors nous demandons à l'opérateur s'il est à supprimer ou non.

Ces résultats sont très satisfaisants avec néanmoins une qualité de détection légèrement supérieure sur le nuage de points segmenté et nettoyé puisqu'une seule rectification est à prévoir (au lieu de deux sur le nuage classifié et filtré). Cela s'explique par le fait que nous avons nettoyé manuellement le nuage de points segmenté alors que le nuage classifié a été filtré par le logiciel 3Dreshaper. Ainsi, la suppression des points de joints sera plus efficace lorsqu'elle sera effectuée manuellement par l'opérateur plutôt que de manière automatique sous 3Dreshaper. Notons toutefois que sans filtrage ou nettoyage préalable du nuage de points, la détection des contours est de meilleure qualité sur le nuage de points classifié.

### III.1.2.2 Test sur le nuage de points n°2

#### III.1.2.2.1. Application au nuage résultant de la classification du nuage de points n°2

Après plusieurs tests, nous conservons le résultat produit pour le paramètre alpha fixé à 1cm.

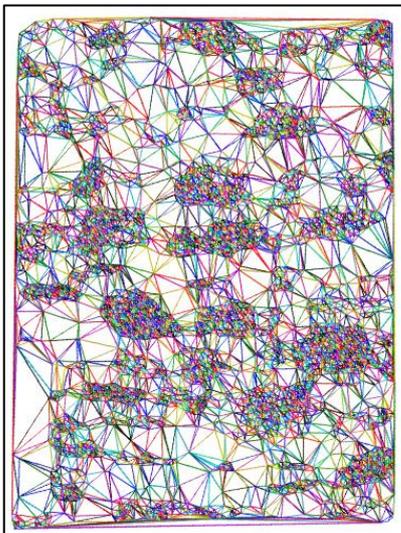


Figure 67 : Triangulation de Delaunay sur le nuage de points n°2 classifié [Python]

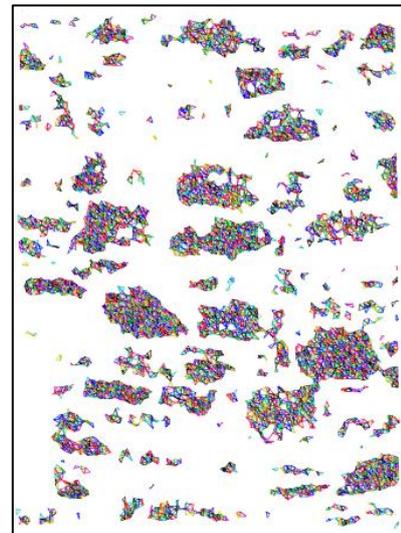


Figure 68 : Conservation des triangles dont le cercle circonscrit est inférieur à alpha de 1cm [Python]

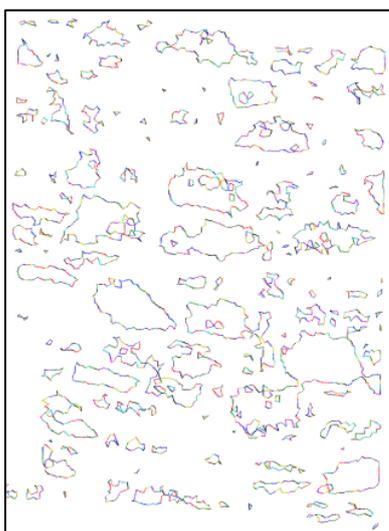


Figure 69 : Identification des contours pour un alpha de 1cm sur le nuage de points n°2 classifié [Python]

Le résultat obtenu est satisfaisant au vu de la discontinuité de la donnée de départ. Par ailleurs, le bruit provoqué par les points isolés lors de la classification entraîne une détection de petits contours qu'il convient de rectifier par un filtrage préalable de notre nuage classifié.

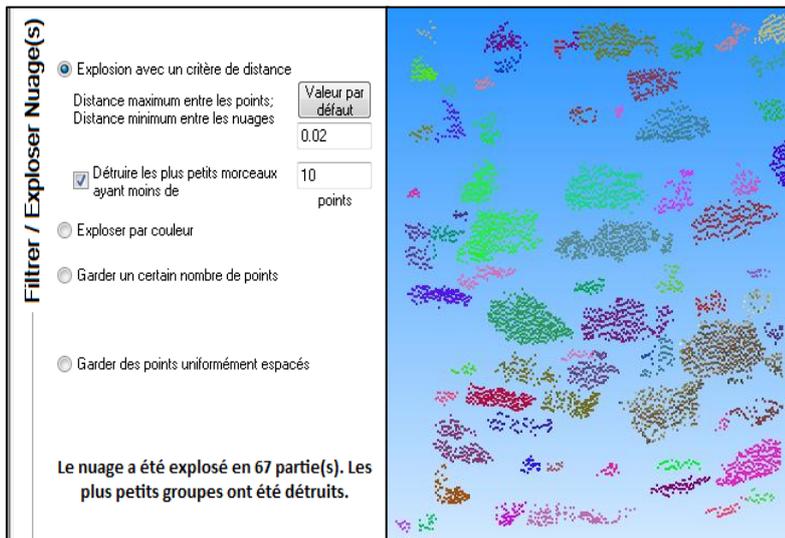


Figure 70 : Filtrage du nuage de points n°2 classifié [3Dreshaper]

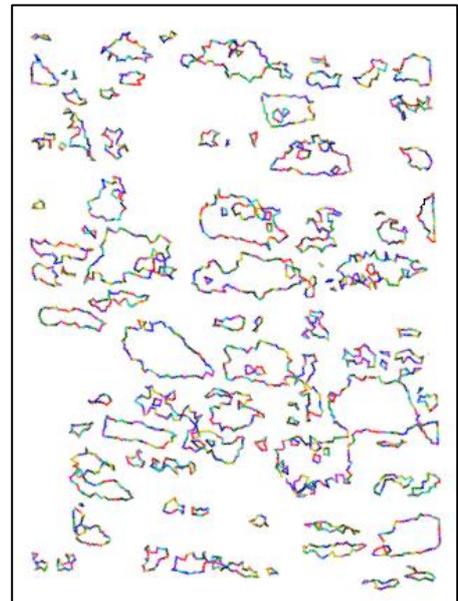


Figure 71 : Identification des contours pour un alpha de 1cm sur le nuage de points n°2 classifié et filtré [Python]

Après filtrage du nuage de points classifié, les contours de petites tailles ont été supprimés et nous obtenons une détection des contours de pierres satisfaisante. En effet, même s'il existe des contours imbriqués les uns dans les autres qui nécessiteraient un traitement algorithmique pour supprimer ce phénomène, le résultat est satisfaisant : nous sommes parvenus à détecter une majorité des pierres même si ces dernières ne sont pas toujours entièrement détectées provoquant alors plusieurs contours au lieu d'un seul, cela reste acceptable au vu de la discontinuité de la donnée initiale. Un traitement manuel sera nécessaire pour rétablir ces défauts.

### III.1.2.2.2. Application au nuage résultant de la segmentation du nuage de points n°2

Le résultat obtenu n'est pas satisfaisant. En effet, un seul contour peut être considéré comme correctement détecté (situé en bas à droite de la Figure 72) ce qui est très insuffisant.

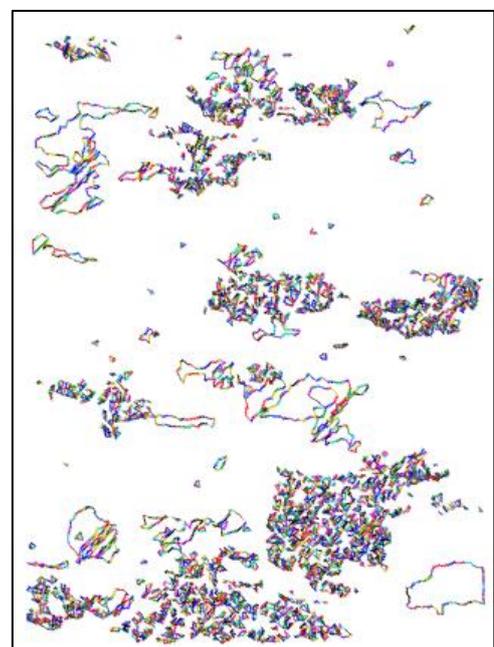


Figure 72 : Résultats de la détection des contours des pierres pour un alpha de 1cm sur le nuage de points n°2 segmenté [Python]

### III.1.2.2.3. Qualité de la détection de contours par notre algorithme alpha-shape en langage python

Nuage de points n°2 classifié et filtré	Nuage de points n°2 segmenté
<p>Taux de bonne détection des contours P5 :</p> $P5 = \frac{\text{nb de contours correctement détectés}}{\text{nb de contours total}}$ $P5 = \frac{21}{79}$ <div style="border: 1px solid black; padding: 2px; display: inline-block;">P5 = 26,58 %</div>	<p>Taux de bonne détection des contours P5 :</p> $P5 = \frac{\text{nb de contours correctement détectés}}{\text{nb de contours total}}$ $P5 = \frac{1}{79}$ <div style="border: 1px solid black; padding: 2px; display: inline-block;">P5 = 1,26%</div>
<p>Taux de fausses alarmes P6 :</p> <p>Il est difficile ici de quantifier le taux de fausses alarmes étant donné le nombre de petits contours détectés. Ainsi, le taux de fausses alarmes est assez élevé.</p>	

Tableau 7 : Qualité de détection des contours par l'algorithme en langage python du nuage n°2

Après avoir développé un algorithme permettant la détection de contours de pierres sur nuage de points 3D, il paraît maintenant intéressant d'étudier les méthodes de détection de contours proposées par le logiciel 3DReshaper.

## III.2 Détermination des contours avec le logiciel 3DReshaper

Le logiciel 3DReshaper ne permet pas de traiter les nuages de points selon l'intensité laser. À la lecture du nuage de points, le logiciel ne considère que les données de positionnement (x,y,z) et les valeurs R, G et B qui colorisent les points du nuage. Ainsi, nous traiterons dans un premier temps une méthode de reconnaissance des contours sur des nuages ayant fait l'objet d'une segmentation préalable selon l'intensité laser et dans un second temps, une détection des contours basée sur les caractéristiques géométriques du nuage.

### III.2.1 Détection des contours à partir d'un maillage : alpha-shape

Ce procédé est constitué de deux étapes : la réalisation d'un maillage 3D sur un nuage de points segmenté et la détection des contours de ce maillage. Ainsi, il s'agit de la méthode alpha-shape. Le maillage utilisé par le logiciel 3DReshaper tient compte de la densité de points pour détecter la présence de trous et compare les dimensions des rayons des cercles circonscrits des triangles voisins pour créer un maillage cohérent. De cette manière, les triangles éliminés sont ceux dont le rayon des cercles circonscrits est supérieur à un maximum prédéfini par le logiciel.

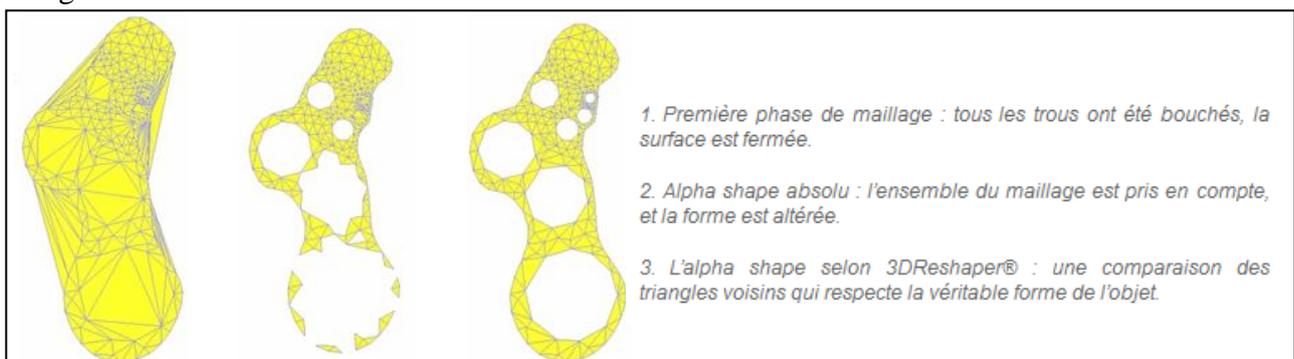


Figure 73 : Illustration de la méthode alpha-shape utilisée par le logiciel 3DReshaper [@ Technodigit, 2015]

Pour tester cette méthode, nous utilisons les nuages de points résultant de la classification et représentant les points de pierre. En effet, nous ne traiterons pas ce procédé sur les nuages segmentés car, comme nous avons pu l'observer précédemment, la segmentation ne produit pas actuellement de résultats assez satisfaisants pour permettre une détection automatique optimale des contours.

### III.2.1.1 Traitement du nuage de points n°1

Nous allons procéder aux étapes suivantes :

- nettoyage du nuage de points
- création d'un maillage 3D
- création des polygones de contours 3D
- calcul du plan moyen au nuage de points
- projection des polygones 3D sur le plan pour obtenir des polygones 2D

#### III.2.1.1.1. Nettoyage du nuage de points n°1

Afin d'obtenir un résultat convenable, il est nécessaire de nettoyer préalablement notre nuage de points. En effet, bien que la classification réalisée soit satisfaisante, celle-ci comporte des erreurs. Nous pouvons observer par endroit des jonctions entre les pierres qui ne devraient pas exister et qui résultent d'une mauvaise classification des points de joints en points de pierres. Ces jonctions seront source d'erreurs pour notre détection de contours puisque si les pierres ne sont pas bien séparées les unes des autres, le logiciel considérera l'ensemble comme une seule entité et formera un contour englobant plusieurs pierres.

Maillage du nuage de points où il existe des jonctions entre les pierres

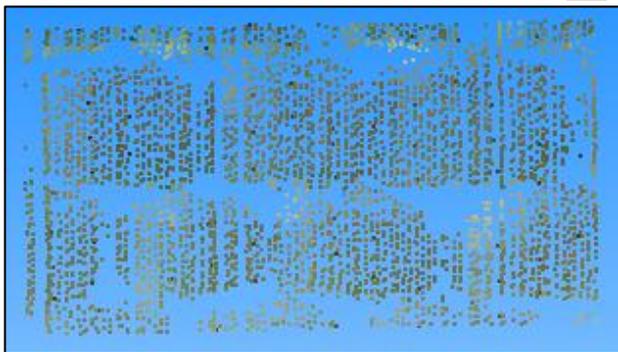


Figure 74 : Extrait du nuage classifié où il existe des jonctions entre les pierres [3DReshaper]



Figure 75 : Maillage du nuage de points extrait [3DReshaper]

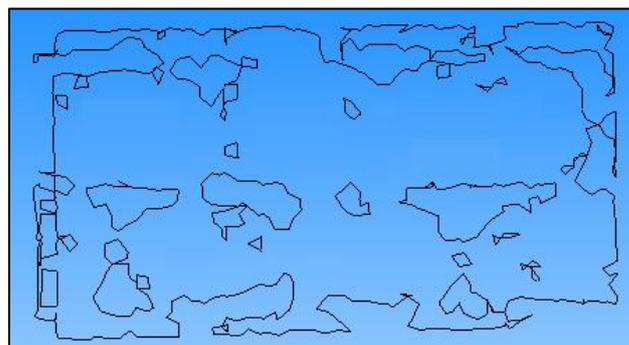


Figure 76 : Mauvaise détection de contours [3DReshaper]

Détection des contours erronée

Pour remédier à ce problème, nous nettoyons le nuage de points en utilisant l'outil « Nettoyer/Séparer Nuage(s) » du logiciel où l'on sélectionne à la main les points que l'on souhaite supprimer ce qui constitue un travail assez long et vient limiter l'automatisation de notre processus.

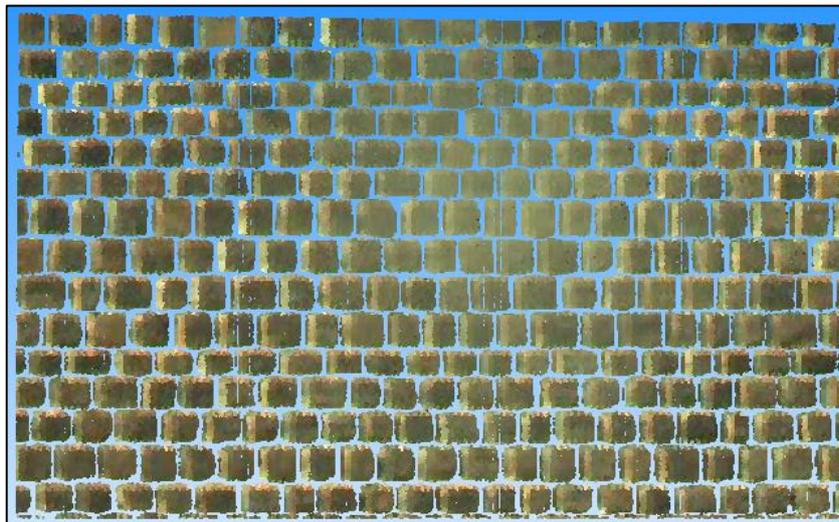


Figure 77 : Nuage de points nettoyé [3DReshaper]

### III.2.1.1.2. Création d'un maillage 3D

Nous poursuivons par la création d'un maillage de notre nuage à l'aide de la fonctionnalité « Maillage 3D ». Nous décidons de garder tous les points de notre nuage et d'opter pour une gestion des trous basée sur la taille des triangles. La « Taille triangle » correspond à la taille de la plus grande arête du triangle du maillage. Comme la distance moyenne entre les points d'une pierre est d'environ 1cm, nous avons donc entré cette valeur.

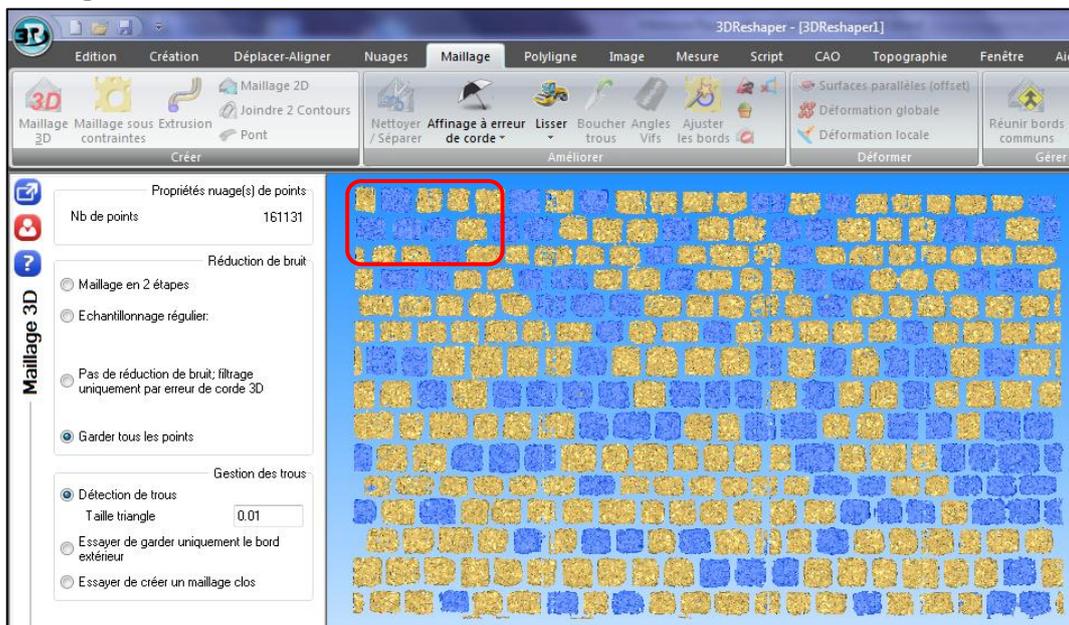
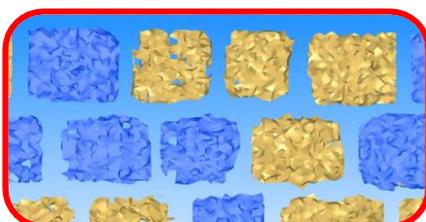


Figure 78 : Réalisation du maillage 3D sur le nuage de points classifié et nettoyé [3DReshaper]



Le maillage ainsi créé comporte des trous à l'intérieur des pierres qu'il convient de combler avec l'outil « Boucher les trous/Lisser les bords » proposé dans l'onglet « Maillage ».

Figure 79 : Présence de trous au sein du maillage des pierres [3DReshaper]

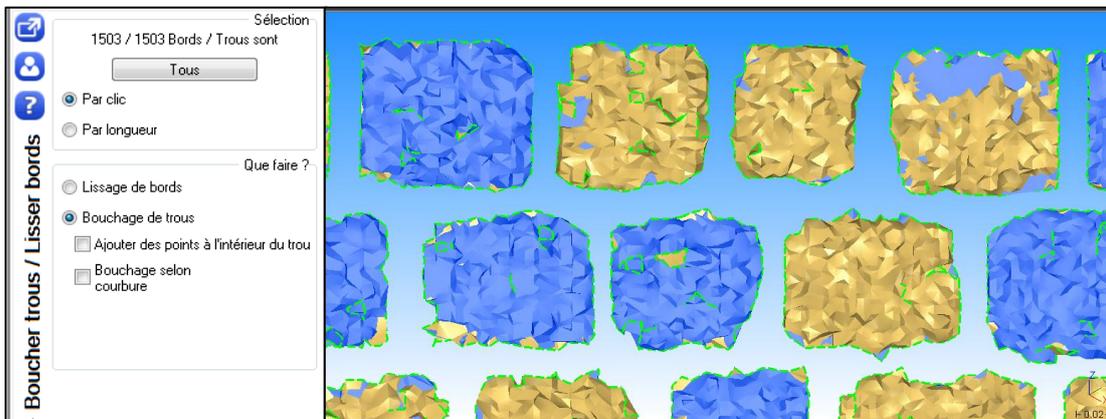


Figure 80 : Méthode pour boucher les trous [3DReshaper]

Suite au traitement des trous au sein des pierres, nous décidons d'exploder le maillage afin de visualiser les entités détectées.



Figure 81 : Explosion des pierres après le bouchage des trous [3DReshaper]

Ici, nous pouvons nous assurer que la plupart des pierres ont bien été identifiées excepté quelques endroits où deux pierres forment une même entité.

Pour y remédier, nous utilisons l'outil « Nettoyer/Séparer maillage » situé dans l'onglet « Maillage ».

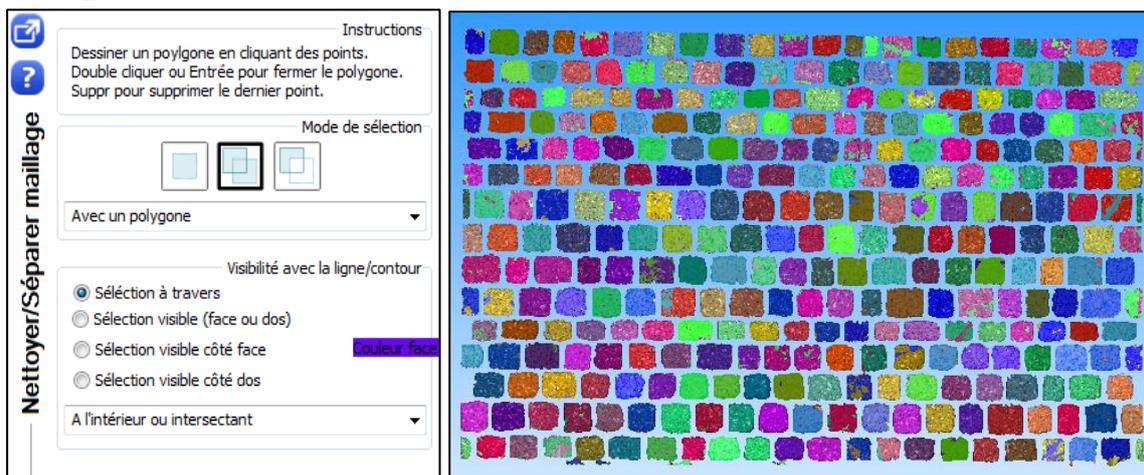


Figure 82 : Pierres du nuage de points individualisées [3DReshaper]

Pour extraire le contour des pierres, nous regroupons tous ces maillages en un maillage composé (onglet « Maillage » → boîte « Gérer » → « Crée un maillage composé »).

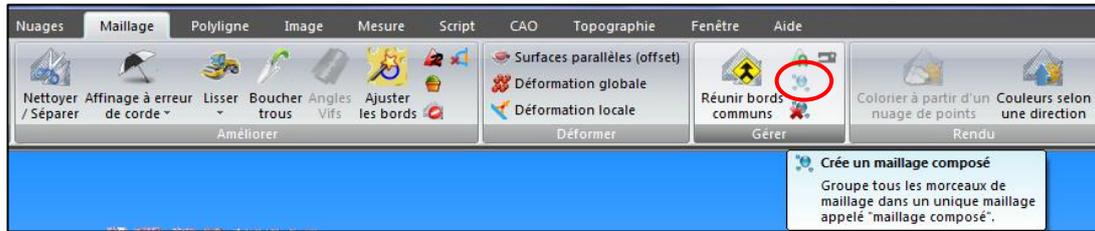


Figure 83 : Option de création d'un maillage composé [3DReshaper]

### III.2.1.1.3. Création des polygones 3D de contours

Nous procédons maintenant à la détection des contours des pierres grâce à la fonctionnalité « Contour/Trou » dans l'onglet « Polygône ».

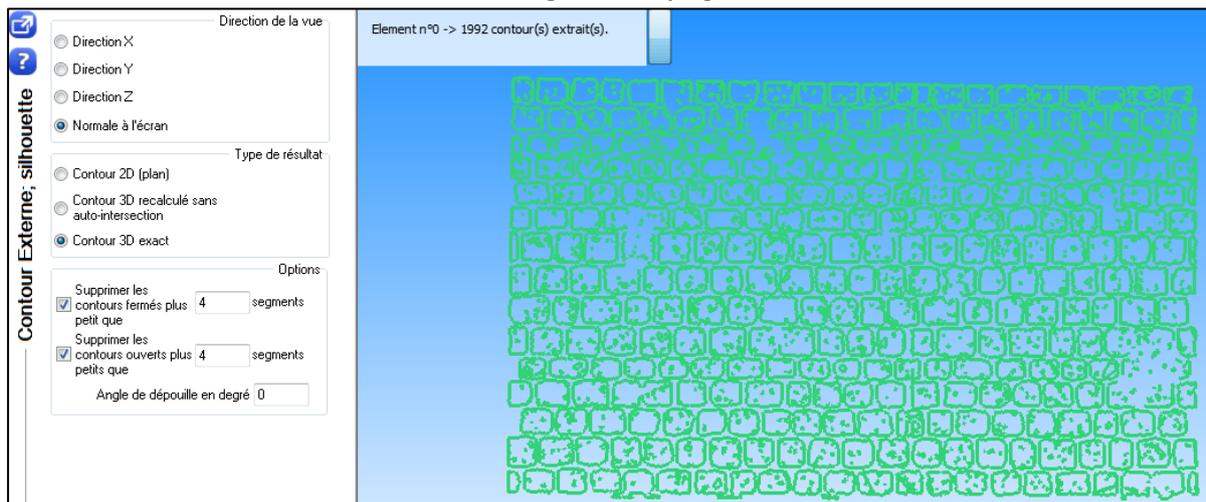


Figure 84 : Détection des contours des pierres [3DReshaper]

Les polygones créés forment une seule et même entité, nous allons donc les différencier les uns des autres grâce à l'outil « Exploder polygones » de l'onglet « Polygône ». De plus, nous supprimons les polygones de petite taille présents au sein des pierres qui ne nous intéressent pas.

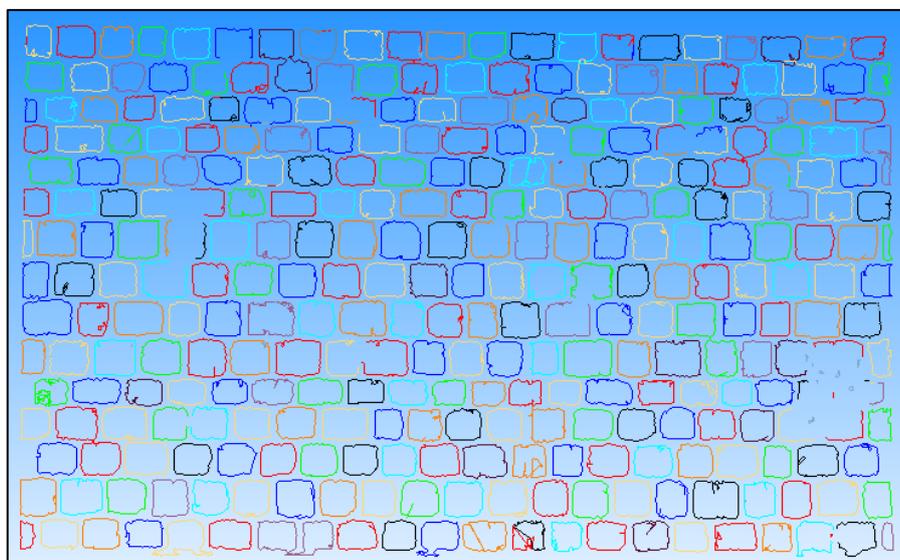


Figure 85 : Polygones des pierres individualisées [3DReshaper]

Nous pouvons alors recenser quatre anomalies qu'il convient de résoudre comme suit :

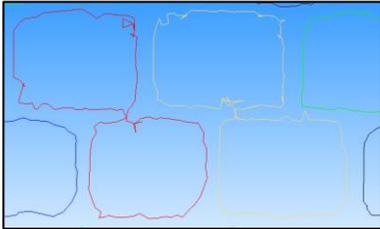
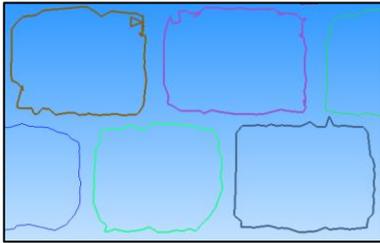
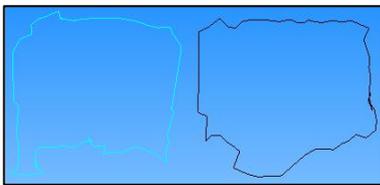
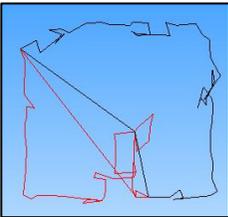
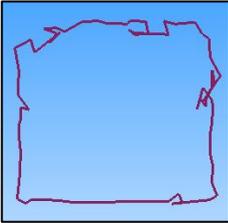
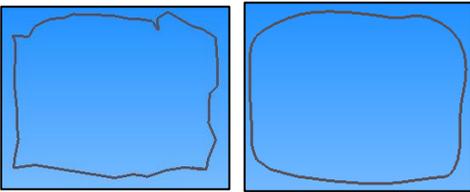
Anomalies	Exemples	Outils utilisés pour y remédier	Résultats
Polylignes englobant plusieurs pierres	 <p>Figure 86 : Deux polygones (rouge et jaune) englobent chacune deux pierres [3DReshaper]</p>	<p>Onglet Polyligne → Couper Polygones</p> <p>Onglet Polyligne → Chainer/Grouper → Fermer les polygones</p>	 <p>Figure 87 : Création de quatre polygones distinctes [3DReshaper]</p>
Polygones non fermés	 <p>Figure 88 : Polygones non fermés [3DReshaper]</p>	<p>Onglet Polyligne → Chainer/Grouper → Fermer les polygones</p>	 <p>Figure 89 : Polygones fermés [3DReshaper]</p>
Deux polygones pour une seule pierre	 <p>Figure 90 : Deux polygones pour le contour d'une seule pierre [3DReshaper]</p>	<p>Onglet Polyligne → Couper Polygones</p> <p>Onglet Polyligne → Chainer/Grouper → Fermer les polygones</p>	 <p>Figure 91 : Regroupement des deux polygones en une seule polygone [3DReshaper]</p>
Présence d'artefacts	 <p>Figure 92 : Présence d'artefacts [3DReshaper]</p>	<p>Onglet Polyligne → Couper Polygones</p> <p>Onglet Polyligne → Chainer/Grouper → Fermer les polygones</p> <p>Onglet Polyligne → Lisser contour</p>	 <p>Figure 93 : Lissage des contours [3DReshaper]</p>

Tableau 8 : Correction des polygones de contours sous 3DReshaper

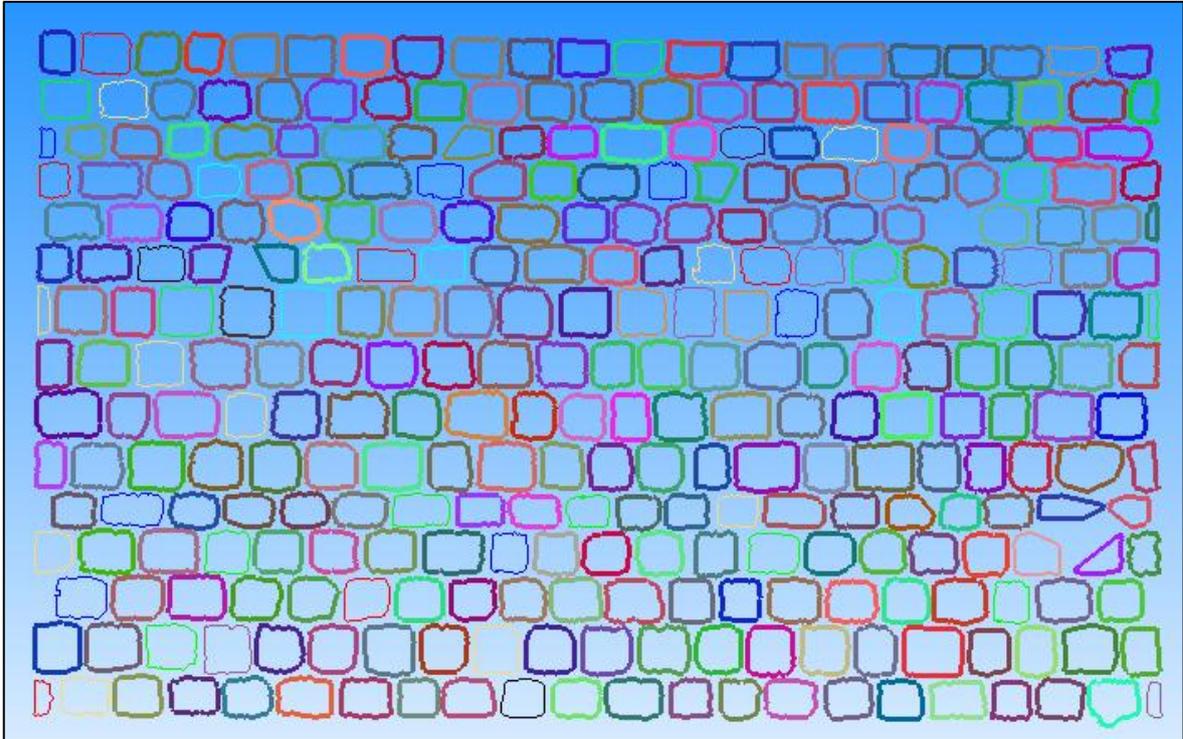


Figure 94 : Contours après corrections manuelles [3DReshaper]

La correction de ces anomalies est un travail long et répétitif.

Rappelons que l'objectif de ce travail est d'obtenir des polygones 2D. Ainsi, nous allons projeter ces polygones 3D sur un plan de manière à obtenir des polygones 2D.

Après lissage, la précision des 3cm n'est sans doute pas respectée.

#### III.2.1.1.4. Calcul d'un plan moyen au nuage de points

Pour déterminer le plan moyen de notre nuage de points, nous calculons sa normale en faisant appel à la fonctionnalité « Mesure → Meilleure Forme → Plan ».

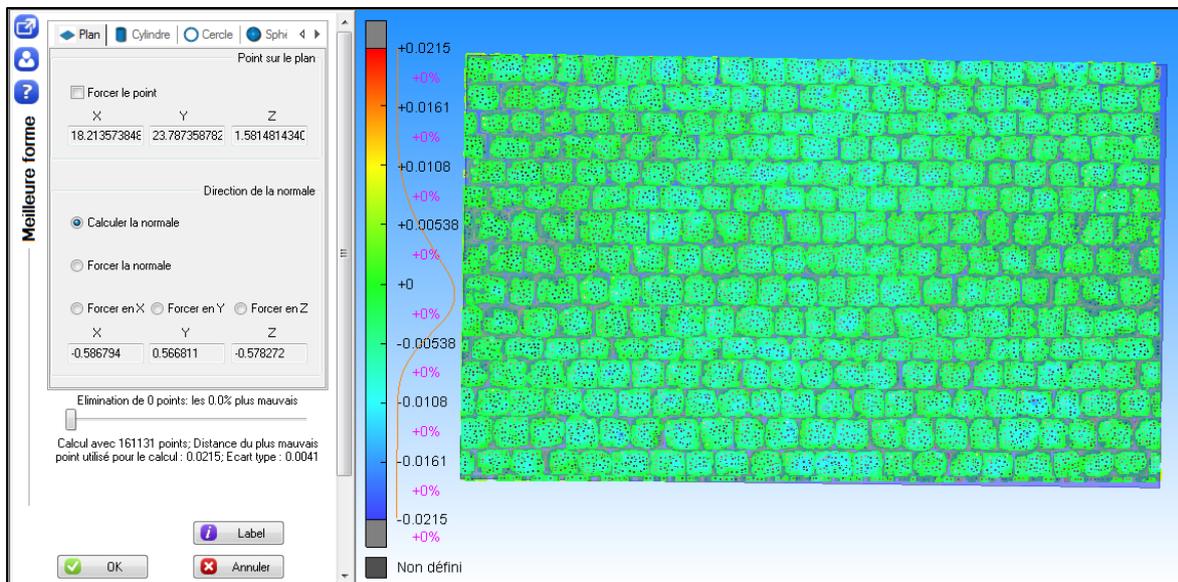


Figure 95 : Calcul de la normale du meilleur plan moyen au nuage de points [3DReshaper]

### III.2.1.1.5.Obtention des polygones 2D et qualité des données

Nous projetons les polygones 3D sur ce plan afin d'obtenir nos polygones en 2D.

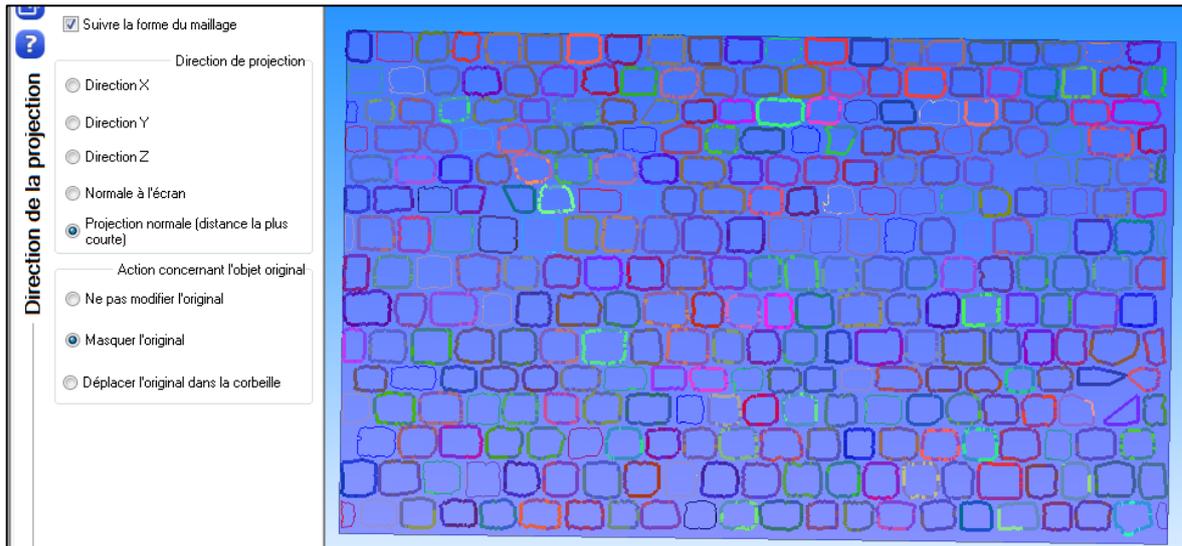


Figure 96 : Projection des polygones 3D sur le plan moyen du nuage de points [3DReshaper]

La précision des coordonnées des polygones 2D dépend directement de la capacité de notre nuage de point à s'apparenter à un plan. Or, la distance du plus mauvais point par rapport à ce plan et l'écart-type résultant du calcul du plan moyen sont respectivement de 2,15cm et de 4,1mm. Ce dernier correspond à l'écart des points initiaux par rapport à leur projection sur le plan. Ainsi, cela signifie que notre nuage de points s'apparente bien à un plan et donc que la qualité des coordonnées est n'est quasiment pas dégradée.

Nous exportons ensuite ces polygones 2D au format DXF.

### III.2.1.1.6.Export des polygones 2D au format DXF

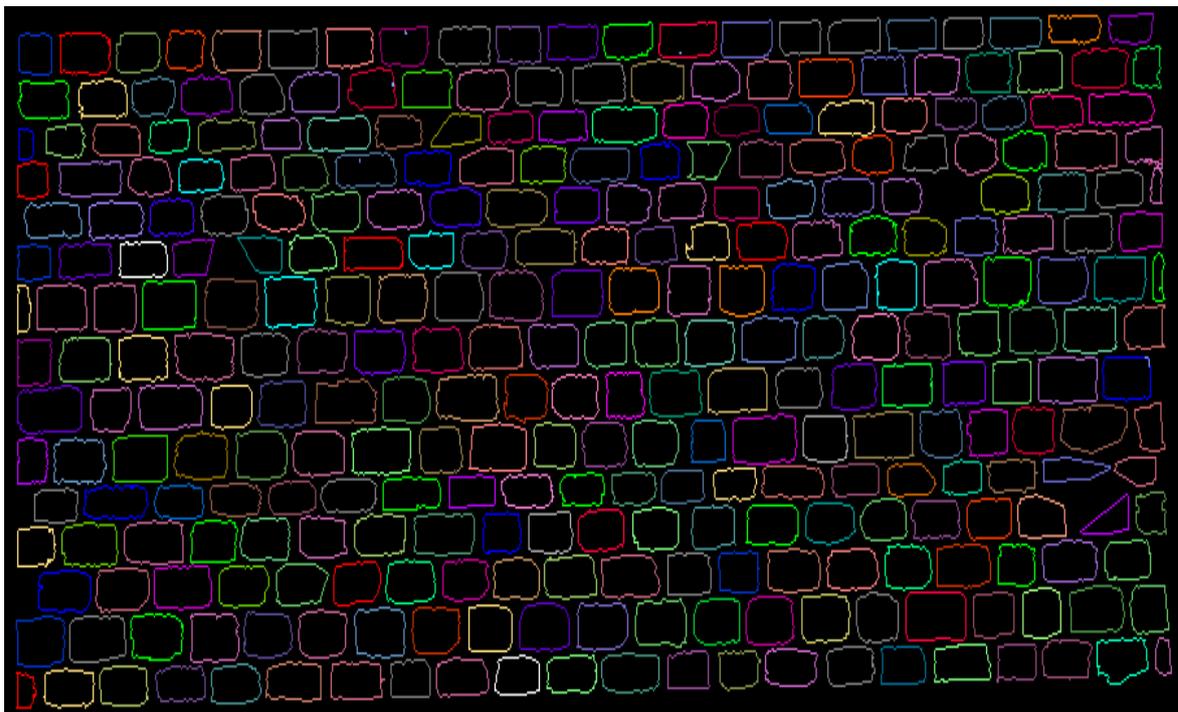


Figure 97 : Polygones 2D exportées au format DXF [AutoCAD]

### III.2.1.2 Traitement du nuage de points n°2

De la même manière, nous obtenons les résultats suivants pour le nuage n°2 :

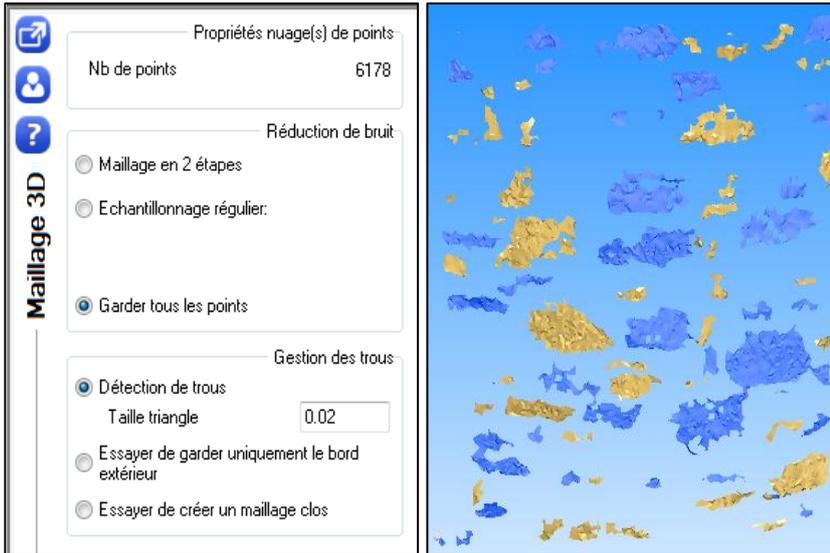


Figure 98 : Réalisation du maillage 3D sur le nuage de points classifié et filtré [3DReshaper]

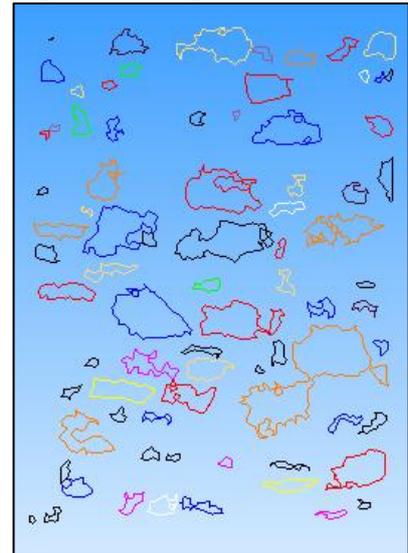
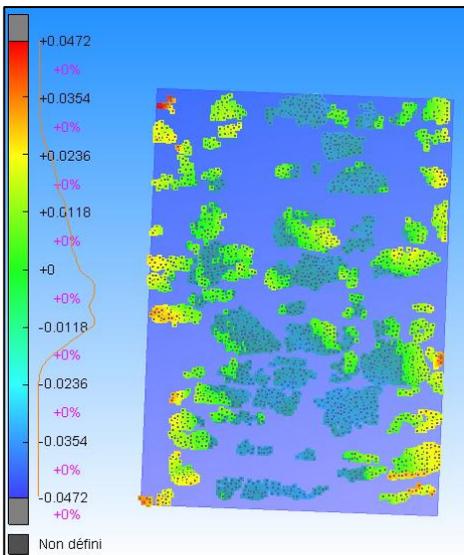


Figure 99 : Contours détectés après bouchage des trous du maillage 3D [3DReshaper]



⇨ distance du plus mauvais point utilisé pour le calcul : 4,72 cm et écart-type : 1,14 cm.

Figure 100 : Calcul de la normale du meilleur plan moyen au nuage de points [3DReshaper]



Figure 101 : Projection des polygones 3D sur le plan moyen du nuage [3DReshaper]

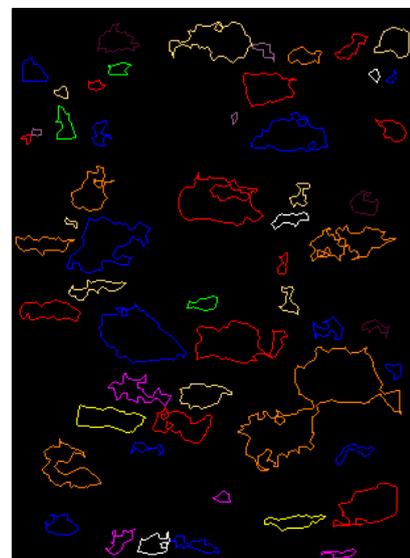


Figure 102 : Polygones 2D exportées au format DXF [AutoCAD]

Les contours de pierres détectés sous le logiciel 3DReshaper sur le nuage de points classifié et filtré sont similaires à ceux obtenus par notre algorithme python. Ce résultat est cohérent avec les données initiales : la détection des contours n'est pas optimale et nécessite un ajustement manuel.

### III.2.1.3 Qualité de la détection de contours par la méthode alpha-shape de 3DReshaper

Dans cette partie, nous considérons le nombre de contours correctement détectés par une analyse purement visuelle des résultats en se basant sur les contours dessinés manuellement sous CloudWorx pour AutoCAD (cf. Figure 1 et Figure 2).

- **Détection des contours de pierres sur le nuage de points n°1**

➤ *Taux de bonne détection des contours P1 :*

$$P1 = \frac{\text{nombre de contours correctement détectés}}{\text{nombre de contours total}} \times 100$$

$$P1 = \frac{306}{320} \times 100$$

$P1 = 95,62\%$
----------------

➤ *Taux de fausses alarmes P2 :*

$$P2 = \frac{\text{nombre de contours incorrects}}{\text{nombre de contours détectés}} \times 100$$

$$P2 = \frac{13}{315} \times 100$$

$P2 = 4,13\%$
---------------

La méthode de détection de contours proposée par 3DReshaper est de très bonne qualité sur ce premier nuage de points puisque 95,62% des contours de pierres ont été correctement détectés et seulement 4,13% ont été mal ou pas détectés. Par ailleurs, sur 320 contours de pierres présentent seul 9 ont été mal détectés et 5 n'ont pas été détectés du tout. Ces chiffres sont très satisfaisants.

- **Détection des contours de pierres sur le nuage de points n°2**

➤ *Taux de bonne détection des contours P1 :*

$$P1 = \frac{10}{79} \times 100$$

$P1 = 12,65\%$
----------------

➤ *Taux de fausses alarmes P2 :*

$$P2 = \frac{47}{57} \times 100$$

$P2 = 82,46\%$
----------------

S'agissant de la qualité de détection du second nuage, celle-ci est beaucoup moins satisfaisante que pour le premier nuage. Cette qualité découle directement de la qualité de classification du nuage réalisée dans la partie II.1. En effet, cette détection de contours se base sur cette classification qui, comme nous avons pu le voir précédemment, est de moins bonne qualité que celle du premier nuage de points.

Cette première méthode proposée par 3DReshaper nous permet non seulement d'obtenir les polygones des contours des pierres en 3D et 2D mais présente également l'avantage de pouvoir les exporter au format DXF et de les exploiter sous AutoCAD comme il est demandé dans le cahier des charges.

De plus, la dégradation des données lors de la projection des polygones 3D sur un plan est négligeable au vue des faibles valeurs de l'écart-type et de la variance des écarts entre le nuage de points et le plan ce qui nous assure une bonne qualité des coordonnées.

En outre, la qualité de détection des contours dépend directement de la qualité de classification du nuage de points à l'entrée du processus. Ainsi, pour un nuage correctement classifié, la détection est très satisfaisante avec un taux de réussite de l'ordre de 95,62%.

Ainsi, nous pouvons considérer que cette méthode répond à la problématique de ce TFE tout en respectant le cahier des charges.

### **III.2.2 Détection des contours par extraction des lignes de rupture de pente**

Cette deuxième méthode consiste en la création d'un maillage du nuage de points initial à partir duquel seront extraites les lignes de rupture de pente correspondantes aux contours des pierres. L'application de ce procédé nécessite donc un nuage de points où les joints doivent être en léger retrait par rapport aux pierres, comme c'est le cas sur notre nuage.

#### **III.2.2.1 Réalisation d'un maillage en deux étapes**

Cette méthode nécessite la réalisation d'un maillage en deux étapes afin d'assurer une précision des contours (Figure103) :

- maillage selon un échantillonnage régulier de 2cm.
- maillage affiné par interpolation de nouveaux points et calcul de l'« erreur cordale avec les meilleurs points uniquement » (pour ne sélectionner que les points les moins bruités).

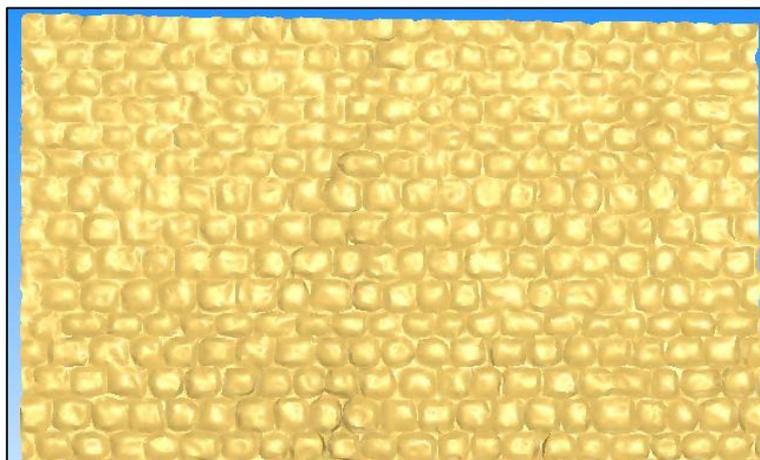


Figure 103 : Résultat du maillage en deux étapes du nuage de points initial [3DReshaper]

### III.2.2.2 Extraction des lignes de rupture de pente

À partir de ce maillage, nous allons extraire les lignes de rupture de pente : les lignes convexes du maillage sont en rouge et les lignes concaves sont représentées en bleu (Figure 104).

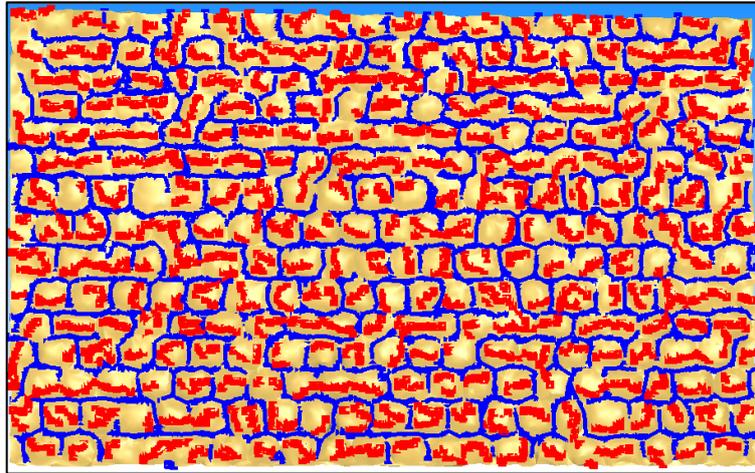


Figure 104 : Extraction des lignes de rupture de pente [3DReshaper]

Ici, nous conserverons uniquement les lignes concaves correspondantes aux contours des pierres (Figure 105).

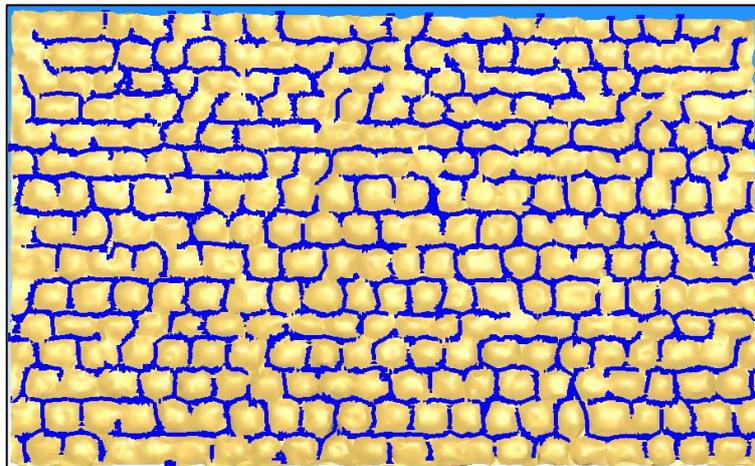


Figure 105 : Lignes concaves correspondantes aux contours des pierres [3DReshaper]

Cette deuxième méthode ne répond pas à nos attentes puisqu'elle ne nous permet pas de distinguer les contours respectifs de chaque pierre. En effet, les polygones tracés ici correspondent au creux séparant les pierres et non aux contours réels recherchés. Ainsi, nous ne retiendrons pas cette méthode dans la suite de notre analyse.

Cependant dans certains cas, nous pourrions imaginer utiliser ce résultat pour guider la localisation de points germes dans notre algorithme de segmentation par croissance de régions.

## IV Discussions des résultats

Après avoir testé plusieurs méthodes de détection de contours, il apparaît maintenant nécessaire de récapituler les différents processus qui s'offrent à nous et d'établir une analyse critique afin de sélectionner le plus approprié d'entre eux.

### IV.1 Les procédés de reconnaissance automatique de contours de pierres sur nuages de points testés dans notre travail

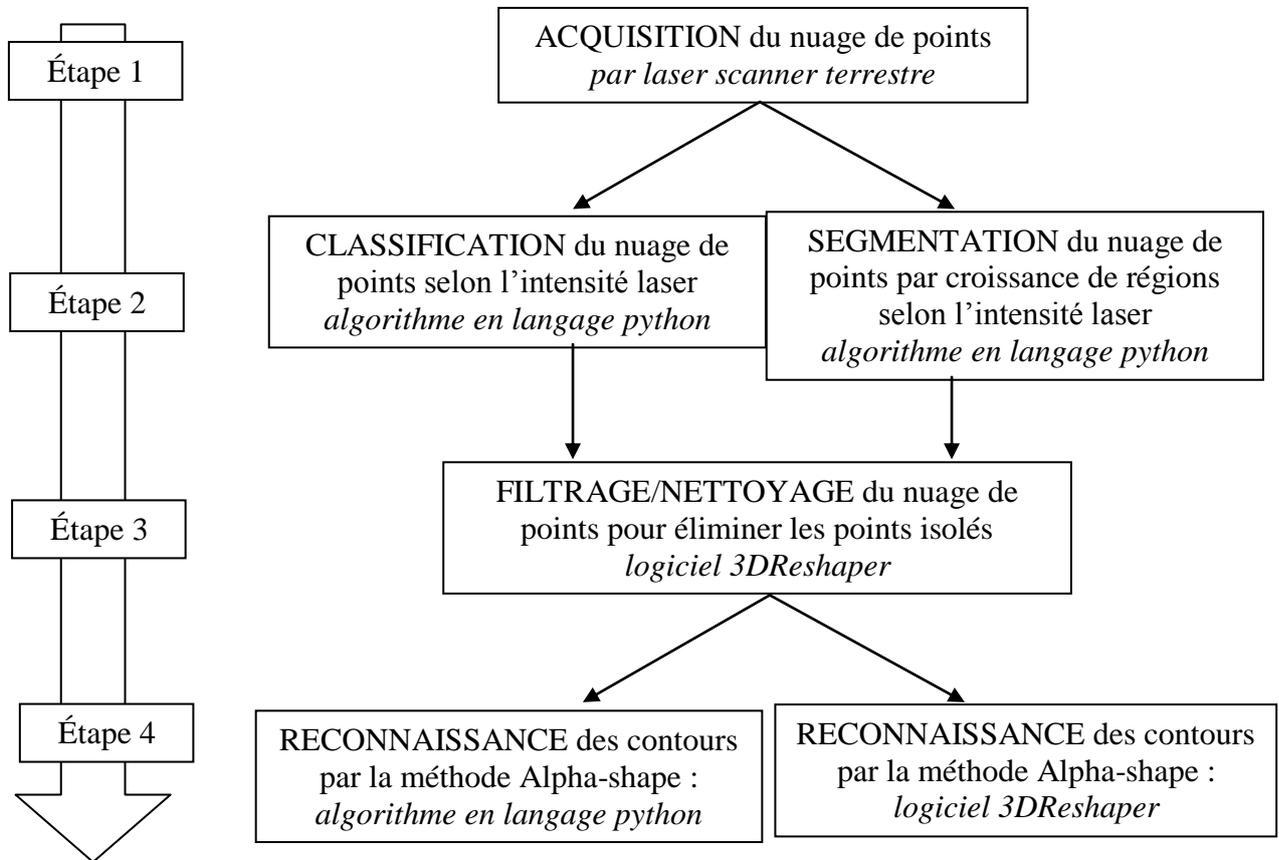


Figure 106 : Organigramme des méthodes testées

### IV.2 Analyses et choix des méthodes

#### IV.2.1 Étape 2 : Détection des points de pierres par classification ou segmentation par croissance de régions

Comme nous avons pu le voir précédemment, cette étape de détection des points de pierres est la plus importante de notre processus puisque c'est à partir de ce résultat que la reconnaissance des contours va être effectuée.

Actuellement, la méthode de classification nous donne de meilleurs résultats que celle de segmentation par croissance de régions. En effet, une étude plus approfondie sur la fixation du seuil de validation du critère d'homogénéité serait nécessaire pour une meilleure sélection des points de pierres. Les taux de réussite et de fausses alarmes de la segmentation, respectivement de 98,88% et de 8,60% pour le nuage de ponts n°1, illustrent parfaitement le phénomène rencontré : trop de points du nuage ont été sélectionnés. Autrement dit, les points correspondant aux pierres ont pratiquement tous été détectés mais trop de points de joints ont été conservés

Ainsi, nous retenons la méthode de classification avec un taux de bonne détection des points de pierres de 93,39% et un taux de fausses alarmes de seulement 3,36% pour assurer un meilleur résultat de notre processus de détection de contours.

Cependant, des travaux complémentaires seront nécessaires pour rendre robuste le choix du seuil de binarisation à un panel plus large de cas possibles.

#### IV.2.2 Étape 3 : Élimination des points isolés par filtrage/nettoyage du nuage

Pour assurer une détection correcte des contours de pierres, il est indispensable d'éliminer les points isolés correspondant aux points malencontreusement conservés lors de la classification du nuage. L'outil de filtrage proposé par 3DReshaper permet de les supprimer automatiquement selon un critère de distance. Toutefois, il peut arriver que certains points de joints, notamment situés en bordure des pierres, ne soient pas éliminés lors du filtrage ce qui nécessite alors un nettoyage manuel obligatoirement réalisé par l'opérateur. Cette opération de nettoyage peut être plus ou moins longue selon la qualité de la classification effectuée précédemment.

#### IV.2.3 Étape 4 : Reconnaissance des contours par l'algorithme alpha-shape en langage python ou par le logiciel 3DReshaper

La méthode mise en avant par notre algorithme et par le logiciel 3DReshaper pour la reconnaissance des contours est celle de l'alpha-shape.

Si l'algorithme développé en langage python pour cette étude produit de bons résultats, celui-ci n'est actuellement pas en mesure de traiter une grande quantité de données contrairement au logiciel 3DReshaper. De plus, des recherches seraient encore nécessaires sur le moyen d'exporter directement les polygones 2D dans un format compatible avec le logiciel AutoCAD.

De ce fait, la méthode proposée par 3DReshaper répond aujourd'hui le mieux au cahier des charges (traitement d'une grande quantité de données et exports des polygones au format DXF, exploitables sous le logiciel AutoCAD).

Ainsi, la solution retenue est la suivante :

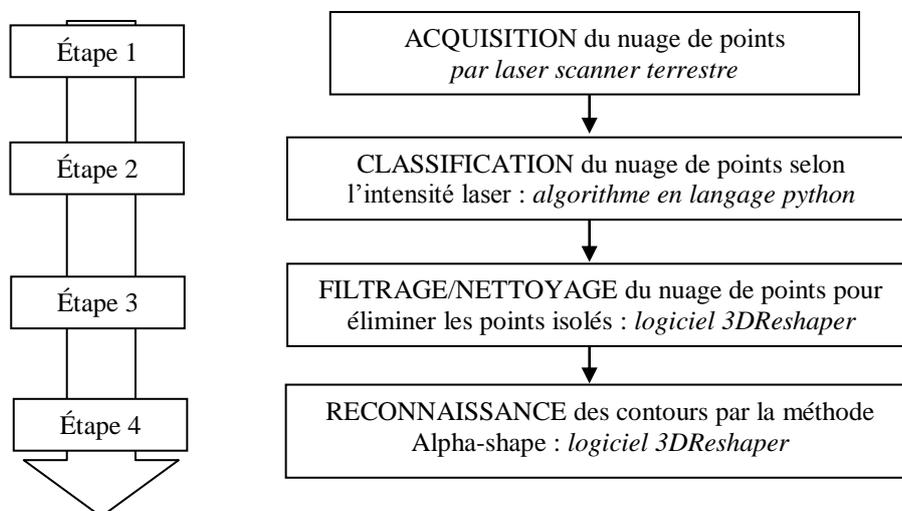


Figure 107 : Organigramme de la procédure retenue

## Conclusion

L'objectif de cette étude est la mise en place d'une procédure de reconnaissance automatique des contours de pierres sur nuages de points 3D acquis par laser scanner terrestre. Pour estimer le coût d'éventuels travaux d'entretien et de réparation sur des monuments historiques, les organismes en charge de leur conservation souhaitent connaître l'état actuel de ces édifices avec leurs déformations et pathologies. Pour ce faire, il est demandé la réalisation de levés pierre à pierre de ces constructions et une restitution graphique vectorielle sous AutoCAD.

Les intérieurs des édifices concernés par cette étude sont peu lumineux et une reconnaissance des pierres basée sur la couleur viendrait limiter la viabilité de notre produit. De ce fait, après avoir effectué des recherches sur l'intensité laser et les paramètres pouvant l'influencer, il nous a paru intéressant d'orienter notre travail sur la détection des contours basée sur l'intensité laser

Afin d'automatiser la procédure de reconnaissance des contours de pierres, plusieurs méthodes ont été développées et testées. En premier lieu, nous nous sommes attachés à développer un algorithme en langage python pouvant répondre à cette problématique. Cette solution produit actuellement des résultats satisfaisants mais nécessite toutefois d'être améliorée sur plusieurs points. En effet, les traitements algorithmiques de détection de contours, basés sur la méthode alpha-shape, ne permettent de traiter à l'heure actuelle qu'une petite quantité de données. De plus, une étude plus approfondie de la validation du critère d'homogénéité lors de la segmentation par croissance de régions du nuage de points serait la bienvenue pour améliorer cette procédure de détection.

Malgré cette conclusion, notons que les développements réalisés permettront d'adapter plus facilement les méthodes à des cas variés.

En deuxième lieu, nous avons étudié deux méthodes proposées par le logiciel 3DReshaper, l'une basée sur la méthode alpha-shape et l'autre sur l'extraction des lignes de rupture de pente. Seule la première méthode répond à notre problématique.

Suite à ces différents tests, nous avons alors dressé une analyse comparative des différentes méthodes afin de sélectionner celle répondant actuellement le mieux à notre objectif.

De cette analyse résulte l'adoption du procédé suivant :

- classification du nuage de points selon l'intensité laser par l'algorithme développé en langage python
- filtrage et nettoyage du nuage classifié sous le logiciel 3DReshaper
- reconnaissance des contours par la méthode alpha-shape du logiciel 3DReshaper

Ainsi, ce produit répond au cahier des charges avec une fiabilité pouvant atteindre les 95%. Toutefois, ce résultat nécessite certains traitements manuels et varie d'un nuage à l'autre puisqu'il dépend de plusieurs facteurs, à savoir la présence d'une intensité laser distincte entre les joints et les pierres et une densité de points élevé pour permettre une meilleure détection des contours. De ce fait, des précautions sont à prendre lors de l'acquisition 3D du nuage de points par laser scanner terrestre notamment pour assurer une densité de points élevée. Les traitements manuels sur polygones pourront s'inspirer des propositions de [Pierre Alban, 2014].

# Bibliographie

## *Articles et mémoires*

[Aitelkadi et al., 2013] Aitelkadi, K., Tahiri, D., Simonetto, E., Sebari, I., Polidori, L. Segmentation of heritage building by means of geometric and radiometric components from terrestrial laser scanning. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Vol.II-5/W1, XXIV International CIPA Symposium, Strasbourg, France, 2-6 September 2013.

[Alban, 2014] Alban, P., Développement d'un outil de segmentation et de classification semi-automatique de pierres, *Mémoire de TFE*, ESGT, 2014.

[Carette, 2008] Carette, V., Amélioration de la représentation géométrique 2D et 3D des agrégations de poissons en support à l'étude de leur évolution spatio-temporelle. *Mémoire pour l'obtention du grade de maître en sciences*, Université de Laval, Québec, 2008.

[Dev, 2013] Dev Bahadur, P., 3D Point Cloud Segmentation, 2013, sur academia.edu.

[Eich et al., 2010] Eich, M., Dabrowxka, M., Kirchner, F., Semantic labeling : classification of 3D entities based on spatial feature descriptors. In: *ICRA 2010, Anchorage, USA*, 2010.

[Rabbani et al., 2006] Rabbani, T., Van den Heuvel, F. A., Vosselman, Segmentation of point clouds using smoothness constraint. In: *ISPRS Commission V Symposium 'Image Engineering and Vision Metrology'*. IAPRS Volume XXXVI, Part 5, Dresden 25-27 September 2006.

[Shen, 2008] Shen Wei, Building boundary extraction based on lidar point clouds data. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Vol.XXXVII, Part B3b, Beijing, 2008.

[Voegtle et al., 2008] Voegtle, T., Schwab, I., Landes, T. Influences of different materials on the measurements of terrestrial laser scanner (TLS). In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Vol.XXXVII, Part B5, Beijing 2008.

[Voegtle et Wakaluk, 2009] Voegtle, T., Wakaluk, S. Effects on the measurements of the terrestrial laser scanner HDS 6000 (Leica) caused by different object materials. In: *Bretar F, Pierrot-Deseilligny M, Vosselman G (Eds) Laser scanning 2009, IAPRS*. Vol.XXXVIII, Part3/W8 – Paris, France, September 1-2, 2009.

## *Documents de formation*

[Simonetto, 2013] Simonetto, E. Introduction au traitement numérique des images, cours de l'ESGT, février 2013.

[Simonetto, 2013] Simonetto, E. Traitement numérique des images, cours de l'ESGT, octobre 2013.

[Simonetto, 2013] Simonetto, E. Traitement numérique des images, Segmentation et Classification des images de télédétection, cours de l'ESGT, octobre 2013.

[Cali, et al., 2013] Cali, J., Chandelier, L., Heno, R., Polidori, L., Simonetto, E. Photogrammétrie et lasergrammétrie, cours de l'ESGT, février 2014.

## *Sites internet*

[@Technodigit, 2015] (2015) URL : <http://www.technodigit.com/>

## Liste des figures

Figure 1 : Exemple d'un résultat obtenu actuellement par GEOMAT en détournant manuellement chaque pierre sur le nuage de points [CloudWorx pour AutoCAD].....	6
Figure 2 : Exemple d'un deuxième résultat obtenu actuellement par GEOMAT en détournant manuellement chaque pierre sur le nuage de points [CloudWorx pour AutoCAD].....	6
Figure 3 : Chaîne de traitement général permettant l'obtention des contours de pierres sur nuage de points.....	8
Figure 4 : Photographie du château médiéval de Fougères.....	9
Figure 5 : Nuage de points du château médiéval de Fougères [Cyclone].....	9
Figure 6 : Image original.....	10
Figure 7 : Image convertie en niveau de gris [Python].....	10
Figure 8 : Image de la norme des gradients maximum en x et y avec un filtre Sobel [Python].....	10
Figure 9 : Histogramme de l'image de la norme des gradients maximum en x et y avec un filtre Sobel [Python]..	10
Figure 10 : Image seuillée pour un seuil de 20 [Python].....	10
Figure 11 : Image originale.....	11
Figure 12 : Extraction des lignes de contours par la méthode de la transformée de Hough [Python].....	11
Figure 13 : Exemple de résultat de segmentation par croissance de régions [Inkscape].....	11
Figure 14 : Influence des nuances de gris sur la précision géométrique [Voegtler et al., 2008].....	12
Figure 15 : Influence de la couleur sur la précision géométrique [Voegtler et al., 2008].....	12
Figure 16 : Influence des nuances de gris sur l'intensité laser [Voegtler et al., 2008].....	12
Figure 17 : Influence de la couleur sur l'intensité laser [Voegtler et al., 2008].....	12
Figure 18 : Erreurs quadratiques moyennes en fonction de la taille des particules de plâtre [Voegtler et al., 2008]	13
Figure 19 : Histogramme des intensités laser sur mur sec et humide [Voegtler et al., 2008].....	13
Figure 20 : Influence des espèces de bois sur la précision géométrique [Voegtler et al., 2008].....	14
Figure 21 : Influence de l'angle d'incidence sur la précision géométrique sur du métal.....	14
Figure 22 : Influence des espèces de bois sur l'intensité laser [Voegtler et al., 2008].....	14
Figure 23 : Valeurs des intensités moyennes sur un objet en métal pour différents angles d'incidence [Voegtler et al., 2008].....	14
Figure 24 : Segmentation par croissance de régions d'un nuage de points [Eich et al., 2010].....	15
Figure 25 : Extraction des points de contour par l'algorithme alpha-shape [Shen, 2008].....	15
Figure 26 : Organigramme présentant la méthode de détection de contours de pierres à partir d'orthoimages extraites du nuage de points colorisé.....	16
Figure 27 : Organigramme présentant la méthode de détection de contours de pierres directement sur nuage de points.....	16
Figure 28 : Nuage de points n°1 en couleurs RGB (à gauche) et en intensité laser (à droite) [Cyclone, ESGT]....	17
Figure 29 : Nuage de points n°2 en couleurs RGB (à gauche) et en intensité laser (à droite) [Cyclone, GEOMAT].....	17
Figure 30 : Nuage de points n°3 en couleurs RGB (à gauche) et en intensité laser (à droite) [Cyclone, ESGT]....	18
Figure 31 : Nuage de points n°4 en couleurs RGB (à gauche) et en intensité laser (à droite) [Cyclone, ESGT]....	18
Figure 32 : Nuage de points n°5 en couleurs RGB (à gauche) et en intensité laser (à droite) [Cyclone, GEOMAT].....	19
Figure 33 : Nuage de points n°6 en couleurs RGB (à gauche) et en intensité laser (à droite) [Cyclone, GEOMAT].....	19

Figure 34 : Histogramme du nuage de points n°1 selon l'intensité laser [Python].....	20
Figure 35 : Histogramme du nuage de points n°2 selon l'intensité laser [Python].....	20
Figure 36 : Nuage de points des pierres résultant de la classification du nuage de points n°1 pour un seuil de -1115.82 [Cyclone].....	22
Figure 37 : Nuage de points des joints résultant de la classification du nuage de points n°1 pour un seuil de -1115.82 [Cyclone].....	22
Figure 38 : Nuage de points des pierres résultant de la classification du nuage de points n°2 pour un seuil de -1286.905 [Cyclone].....	23
Figure 39 : Nuage de points des joints résultant de la classification du nuage de points n°2 pour un seuil de -1286.905[Cyclone].....	26
Figure 40 : Schéma de la sélection des germes sur un nuage de points représentant un mur.....	26
Figure 41 : Schéma de l'évolution des germes en surfaces-germes initiales.....	26
Figure 42 : Schéma des surfaces-germes validées.....	27
Figure 43 : Schéma de la croissance d'une surface-germe validée.....	27
Figure 44 : Schéma du nuage de points résultant de la segmentation par croissance de régions.....	28
Figure 45 : Nuage résultant de la segmentation par croissance de régions du nuage de points n°1 [Cyclone].....	28
Figure 46 : Nuage résultant de la segmentation par croissance de régions du nuage de points n°2 [Cyclone].....	29
Figure 47 : Schéma de la projection du nuage de points sur un plan.....	31
Figure 48 : Triangulation de Delaunay sur un semi de points [Python].....	32
Figure 49 : Cercles circonscrits des triangles issus de la triangulation de Delaunay [Python].....	33
Figure 50 : Schéma du principe de l'alpha-shape [Eich et al., 2010].....	33
Figure 51 : Triangles conservés [Python].....	33
Figure 52 : Contours obtenus par la méthode alpha-sape [Python].....	33
Figure 53 : Extrait du nuage de points initial sur lequel la méthode alpha-shape va être testée [Cyclone].....	34
Figure 54 : Nuage de points n°1 résultant de la classification [Cyclone].....	34
Figure 55 : Nuage de points n°1 résultant de la segmentation par croissance de régions [Cyclone].....	34
Figure 56 : Triangulation de Delaunay sur l'extrait du nuage de points n°1 classifié et projeté [Python].....	34
Figure 57 : Conservation des triangles dont le cercle circonscrit est inférieur à alpha 5mm [Python].....	34
Figure 58 : Identification des contours pour un alpha de 5mm [Python].....	35
Figure 59 : Filtrage de l'extrait du nuage de points n°1 classifié [3Dreshaper].....	35
Figure 60 : Extrait du nuage de points n°1 classifié filtré en intensité laser [Cyclone].....	35
Figure 61 : Conservation des triangles dont le cercle circonscrit est inférieur à alpha 5mm sur l'extrait du nuage de points n°1 classifié et filtré [Python].....	36
Figure 62 : Identification des contours pour un alpha de 5mm sur l'extrait du nuage de points n°1 classifié et filtré [Python].....	36
Figure 63 : Extrait du nuage de points n°1 segmenté et nettoyé [Cyclone].....	36
Figure 64 : Triangulation de Delaunay sur l'extrait du nuage de points n°1 segmenté, nettoyé et projeté [Python].....	36
Figure 65 : Conservation des triangles dont le cercle circonscrit est inférieur à alpha de 6mm [Python].....	36
Figure 66 : Identification des contours pour un alpha-shape de 6mm sur l'extrait du nuage de points n°1 segmenté et nettoyé [Python].....	37
Figure 67 : Triangulation de Delaunay sur le nuage de points n°2 classifié [Python].....	38
Figure 68 : Conservation des triangles dont le cercle circonscrit est inférieur à alpha de 1cm [Python].....	38

Figure 69 : Identification des contours pour un alpha de 1cm sur le nuage de points n°2 classifié [Python].....	38
Figure 70 : Filtrage du nuage de points n°2 classifié [3DReshaper].....	39
Figure 71 : Identification des contours pour un alpha de 1cm sur le nuage de points n°2 classifié et filtré [Python].....	39
Figure 72 : Résultats de la détection des contours des pierres pour un alpha de 1cm sur le nuage de points n°2 segmenté [Python].....	39
Figure 73 : Illustration de la méthode alpha-shape utilisée par le logiciel 3DReshaper [@ Technodigit, 2015]....	40
Figure 74 : Extrait du nuage classifié où il existe des jonctions entre les pierres [3DReshaper].....	41
Figure 75 : Maillage du nuage de points extrait [3DReshaper].....	41
Figure 76 : Mauvaise détection de contours [3DReshaper].....	41
Figure 77 : Nuage de points nettoyé [3DReshaper].....	42
Figure 78 : Réalisation du maillage 3D sur le nuage de points classifié et nettoyé [3DReshaper].....	42
Figure 79 : Présence de trous au sein du maillage des pierres [3DReshaper].....	42
Figure 80 : Méthode pour boucher les trous [3DReshaper].....	43
Figure 81 : Explosion des pierres après le bouchage des trous [3DReshaper].....	43
Figure 82 : Pierres du nuage de points individualisées [3DReshaper].....	43
Figure 83 : Option de création d'un maillage composé [3DReshaper].....	44
Figure 84 : Détection des contours des pierres [3DReshaper].....	44
Figure 85 : Polygones des pierres individualisées [3DReshaper].....	44
Figure 86 : Deux polygones (rouge et jaune) englobent chacune deux pierres [3DReshaper].....	45
Figure 87 : Création de quatre polygones distonctes [3DReshaper].....	45
Figure 88 : Polygones non fermées [3DReshaper].....	45
Figure 89 : Polygones fermées [3DReshaper].....	45
Figure 90 : Deux polygones pour le contour d'une seule pierre [3DReshaper].....	45
Figure 91 : Regroupement des deux polygones en une seule polygone [3DReshaper].....	45
Figure 92 : Présence d'artefacts [3DReshaper].....	45
Figure 93 : Lissage des contours [3DReshaper].....	45
Figure 94 : Contours après corrections manuelles [3DReshaper].....	46
Figure 95 : Calcul de la normale du meilleur plan moyen au nuage de points [3DReshape].....	46
Figure 96 : Projection des polygones 3D sur le plan moyen du nuage de points [3DReshaper].....	47
Figure 97 : Polygones 2D exportées au format DXF [AutoCAD].....	47
Figure 98 : Réalisation du maillage 3D sur le nuage de points classifié et filtré [3DReshaper].....	48
Figure 99 : Contours détectés après bouchage des trous du maillage 3D [3DReshaper].....	48
Figure 100 : Calcul de la normale du meilleur plan moyen au nuage de points [3DReshaper].....	48
Figure 101 : Projection des polygones3D sur le plan moyen du nuage [3DReshaper].....	48
Figure 102 : Polygones 2D exportées au format DXF [AutoCAD].....	48
Figure 103 : Résultat du maillage en deux étapes du nuage de points initial [3DReshaper].....	50
Figure 104 : Extraction des lignes de rupture de pente [3DReshaper].....	51
Figure 105 : Lignes concaves correspondantes aux contours des pierres [3DReshaper].....	51
Figure 106 : Organigramme des méthodes testées.....	52
Figure 107 : Organigramme de la procédure retenue.....	53

## Liste des tableaux

Tableau 2 : Comparaisons des données image/nuage de points.....	9
Tableau 2 : Influences de la couleur et des niveaux de gris sur l'intensité laser.....	12
Tableau 3 : Influences des matériaux en plâtre, bois et de l'humidité sur l'intensité laser.....	13
Tableau 4 : Influences de différentes espèces de bois et du métal sur l'intensité laser.....	14
Tableau 5 : Comparaison avantages/inconvénients des deux méthodes.....	30
Tableau 6 : Qualité de détection des contours par l'algorithme en langage python du nuage n°1.....	37
Tableau 7 : Qualité de détection des contours par l'algorithme en langage python du nuage n°2.....	40
Tableau 8 : Correction des polygones de contours sous 3DReshaper.....	45

# Résumé

## Introduction

Il existe actuellement une demande grandissante du Centre des Monuments Nationaux concernant la réalisation de levés architecturaux. En effet, cet établissement public administratif a pour principales missions de gérer, conserver et restaurer les monuments nationaux. Ainsi, ce dernier souhaite avoir une connaissance exhaustive de ces édifices et notamment des pierres les composant afin de pouvoir estimer le coût d'éventuels travaux d'entretien et de rénovation. C'est pourquoi, il est demandé dans certains appels d'offre de rendre compte de leur état actuel en effectuant un levé pierre à pierre et une restitution graphique vectorielle sous AutoCAD.

Pour répondre à cette demande, la méthode employée par la structure d'accueil GEOMAT consiste à détourer manuellement chaque pierre de l'édifice sur des nuages de points colorisés obtenus par levés lasergrammétriques ce qui constitue un travail fastidieux et chronophage.

La mise en place d'un processus de reconnaissance automatique de contours de pierres sur nuages de points s'est alors rapidement imposée au sein de la société.

## I La démarche suivie

### I.1. La détermination des objectifs de recherches

L'objectif de ce travail est la détermination des coordonnées des points formant le contour des pierres présentes sur le nuage de points étudié. Pour ce faire, deux étapes sont nécessaires :

- Étape 1 : Distinction des points de pierre et des points de joints
- Étape 2 : Reconnaissance des contours des pierres

Une phase de recherches bibliographiques a été réalisée pour identifier les méthodes scientifiques correspondantes à ces étapes.

Par ailleurs, le cahier des charges impose que le produit final soit plan. De ce fait, il faudra concevoir un procédé permettant de calculer le plan épousant le mieux la forme du nuage de points pour pouvoir projeter les polygones des contours de pierres sur celui-ci tout en conservant une précision des données de 3 cm en absolu.

Enfin, les parties intérieures des édifices concernés par cette étude constituent des lieux sombres où il est difficile sur des nuages de points 3D colorisés ou des photographies de distinguer les contours des pierres. Ainsi, nous préconisons une méthode permettant de remédier à ce problème.

### I.2. Stratégie de traitement

#### I.2.1. Une étude basée sur l'intensité laser

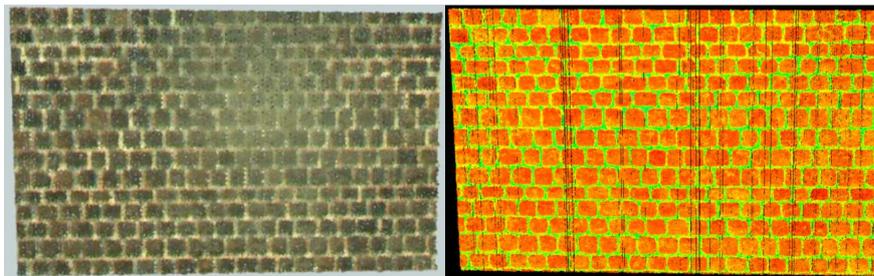


Figure 1 : Nuage de points n°1 en couleurs RGB (à gauche) et en intensité laser (à droite) [Cyclone, ESGT]

Afin de proposer un processus permettant de traiter un maximum de nuages de points, nous nous sommes alors intéressés aux caractéristiques de l'intensité laser. Après la lecture d'articles scientifiques élaborés par différents laboratoires de recherches, il s'avère que l'intensité laser présente l'avantage de pouvoir traiter certains cas où la luminosité est faible. En effet, l'intensité laser évolue selon la nature des matériaux, leur couleur et l'inclinaison du faisceau lumineux lors de l'acquisition du nuage de points 3D ce qui signifie que si nous ne

parvenons pas à distinguer les pierres des joints sur le nuage de points colorisé, l’affichage en intensité laser peut le permettre. Ainsi, si nous sommes en mesure de différencier visuellement les pierres des joints sur le nuage de points en intensité laser, nous pouvons alors supposer qu’un algorithme sera lui aussi en mesure de séparer correctement les points de pierres des points de joints.

## I.2.2. Développement d’un algorithme en langage Python

Il n’existe pas à l’heure actuelle de logiciel permettant le traitement automatique de nuages de points selon l’intensité laser qui ne requiert pas de compétences en programmation. En effet, les fonctionnalités de traitement proposées par exemple par le logiciel 3DReshaper se basent uniquement sur les caractéristiques colorimétriques et géométriques des nuages de points. Certains logiciels tels que FME peuvent traiter les nuages de points selon l’intensité laser mais nécessitent toutefois l’acquisition préalable de compétences en programmation.

Afin de proposer une solution simple et facilement utilisable par les employés de la structure GEOMAT, nous nous sommes attachés à développer un algorithme en langage Python détectant les contours des pierres sur nuages de points selon l’intensité laser de manière automatique.

## I.2.3. Les méthodes scientifiques répondant aux deux étapes

Suite à une phase de recherches bibliographiques, nous avons alors décidé de développer et de tester les méthodes suivantes :

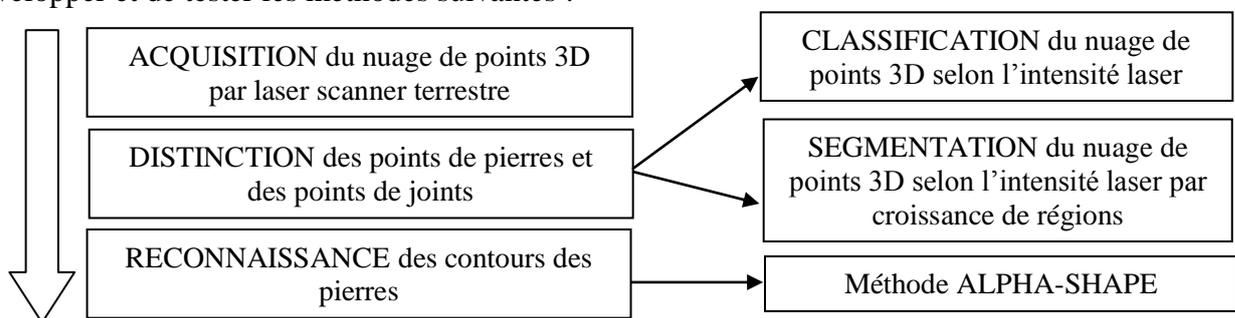


Figure 2 : Organigramme de la procédure de reconnaissance automatique de contours de pierres

### I.2.3.1. Les méthodes de classification et de segmentation

La classification consiste à labelliser chaque point du nuage par une classe thématique.

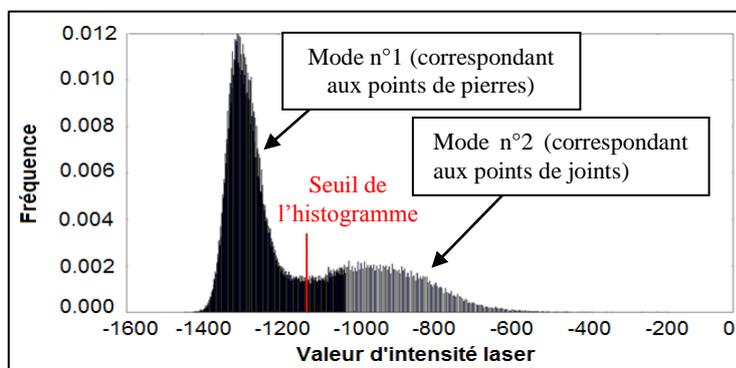


Figure 3 : Histogramme du nuage de points n°1 selon l’intensité laser [Python]

Ici, deux classes nous intéressent : la classe « pierres » et la classe « joints ». Le label d’un point est indépendant de celui de ses voisins.

Pour classifier nos nuages de points, nous avons procédé à la réalisation d’un seuillage simple, appelé aussi binarisation, qui permet de séparer les points de pierres de ceux des joints en créant deux nuages distincts. L’étude de l’histogramme du nuage selon l’intensité laser nous permet de

mieux comprendre ce fonctionnement. Sur les nuages étudiés dans ce TFE, nous avons pu remarquer que les histogrammes sont constitués de deux modes : l’un correspond aux points de pierres et l’autre aux points de joints. Pour déterminer la valeur du seuil, nous avons développé un algorithme en langage Python qui se base sur le principe du seuillage adaptatif. Si nous

sommes en mesure de différencier visuellement les pierres des joints sur le nuage de points colorisé en intensité laser, l'algorithme développé en langage Python fonctionnera correctement.

S'agissant de la segmentation par croissance de régions, cette méthode consiste à partitionner le nuage de points en régions homogènes à partir de points germes qui grossissent par agglomération de points voisins dès qu'ils vérifient un critère d'homogénéité. Le processus s'interrompt dès qu'il n'est plus possible d'ajouter de points à une région. Une méthode a été élaborée en langage Python.

### I.2.3.2. La méthode alpha-shape

La méthode alpha-shape permet la reconnaissance de formes générées par des ensembles de points. L'algorithme se base sur la triangulation de Delaunay à partir de laquelle les triangles ayant leur rayon de cercle circonscrit inférieur au paramètre alpha sont conservés. Ainsi, les contours correspondent aux arêtes appartenant à un seul triangle.

## II Les résultats obtenus

### II.1. La distinction des points de pierres et des points de joints par classification et segmentation

La classification produit par notre algorithme est très satisfaisante puisque très peu de points ont été attribués à la mauvaise classe :

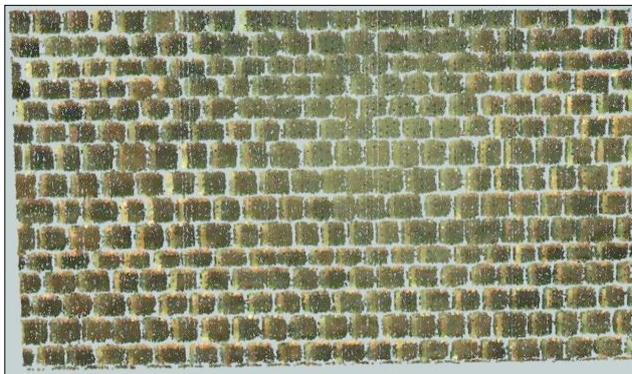


Figure 4 : Nuage de points des pierres résultant de la classification du nuage de points n°1 pour un seuil de -1115.82 [Cyclone]

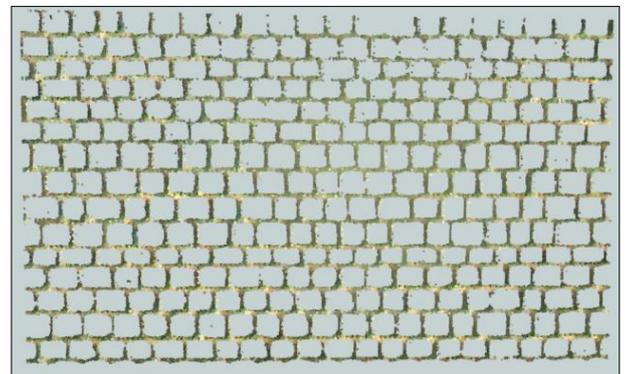


Figure 5 : Nuage de points des joints résultant de la classification du nuage de points n°1 pour un seuil de -1115.82 [Cyclone]

- Taux de bonne détection des points de pierres : 93.39%
- Taux de fausses alarmes des points de pierres : 3.36%
- Taux de bonne détection des points de joints : 86.06%
- Taux de fausses alarmes des points de joints : 24.77%

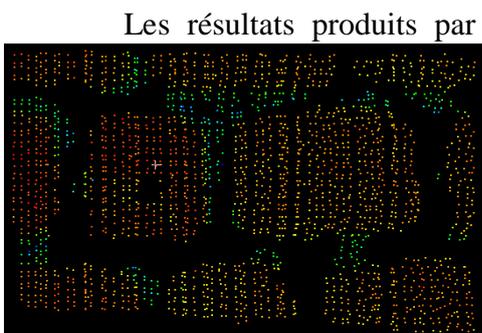


Figure 6 : Nuage résultant de la segmentation par croissance de régions d'un extrait du nuage de points n°1 [Cyclone]

Les résultats produits par la segmentation sont de moins bonne qualité que ceux résultant de la classification. Ceci provient de l'étape de validation du critère d'homogénéité par notre algorithme. En effet, nous avons défini le critère d'homogénéité comme étant égal à la variance des intensités laser des points de la surface courante :  $Var(X) = \frac{1}{n} \sum_{i=1}^n (x_i - m)^2$  et avons fixé la valeur du seuil de proche en proche jusqu'à obtenir un résultat correct. Il s'agit d'une méthode empirique qui pourrait être améliorée par la réalisation d'une étude plus approfondie de la validation de ce critère d'homogénéité.

- Taux de bonne détection des points de pierre : 98.88%
- Taux de fausses alarmes : 8.60%

La classification produit actuellement de meilleurs résultats que la segmentation. Toutefois, nous pouvons supposer que la segmentation par croissance de régions constituerait une méthode plus efficace pour notre processus de détection de contours si la validation du critère d'homogénéité était améliorée. En effet, la détection de points de pierres par classification fait l'objet d'un seuillage global tandis que la segmentation sélectionne les points de pierres par rapport à une variation locale. De plus, la segmentation par croissance de régions permet de labelliser chaque pierre, c'est-à-dire que les pierres sont individualisées.

## II.2. La reconnaissance des pierres et l'isolement des contours par la méthode alpha-shape

L'algorithme développé inclut la projection des polygones de contours sur un plan et produit de très bons résultats. Pour assurer une reconnaissance optimale des formes, il est préférable d'appliquer cette méthode sur un nuage de points classifié et nettoyé (suppression des points isolés). De plus, la précision des données reste dans les tolérances imposées par le cahier des charges (inférieure à 3cm) et peu de corrections des contours sont à apportées.

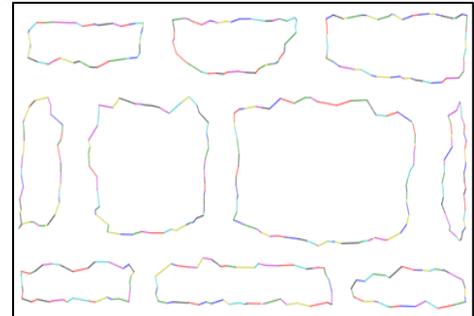


Figure 7 : Identification des contours pour un alpha fixé à 5mm sur l'extrait du nuage de points n°1 classifié et nettoyé [Python]

Le logiciel 3DReshaper permet également d'appliquer la méthode alpha-shape sur un nuage de points préalablement classifié ou segmenté. Ainsi, cette méthode produit de très bons résultats après toutefois un traitement manuel des polygones. En effet, étant donné le sens d'une polygone, la normale à celle-ci n'est pas toujours orientée vers le centre de la pierre ce qui provoque des artefacts en forme de boucles (cf. Figure 8). Ce phénomène entraîne des corrections manuelles assez longues. Par ailleurs, cette méthode conserve la précision des données et présente deux avantages : l'export des polygones des contours au format .dxf qui sont alors exploitables sous AutoCAD, et le traitement de plus grandes quantités de données par rapport à notre algorithme développé en langage Python.

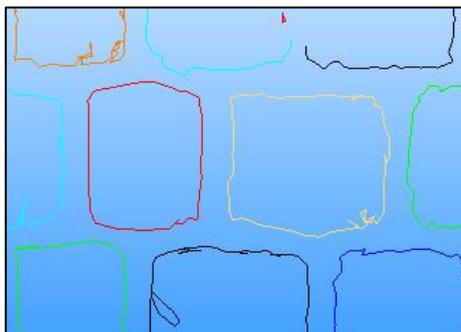


Figure 8 : Polygones des contours des pierres produites sous le logiciel 3DReshaper

## Conclusion

Plusieurs stratégies sont possibles pour répondre à cette problématique. Dans notre cas, le processus de classification du nuage de points selon l'intensité laser par notre algorithme → nettoyage manuel des points isolés éventuels → méthode alpha-shape du logiciel 3DReshaper répond le mieux au cahier des charges avec une qualité de détection des contours pouvant aller jusqu'à 95%. Toutefois, ce chiffre est à relativiser puisqu'il dépend de plusieurs facteurs tels que la présence d'une intensité laser distincte entre les pierres et les joints, et une densité de points élevée. Dans le cas contraire, la segmentation par croissance de régions optimisée ainsi que notre méthode alpha-shape apporteraient des résultats prometteurs.

## Poster Scientifique

## Contexte et objectifs

**Contexte :** Des levés pierre à pierre et des restitutions graphiques de monuments nationaux font l'objet d'une demande par le Centre des Monuments Nationaux afin qu'il puisse estimer les coûts de travaux d'entretien et de rénovation. Pour répondre à cette demande, la méthode actuellement employée par la structure GEOMAT consiste à détourer manuellement chaque pierre sur des nuages de points colorisés obtenus par levés lasergrammétriques.

**Objectifs :** Réduire le temps de traitement manuel par la mise en place d'un processus de reconnaissance automatique de contours de pierres sur nuages de points.

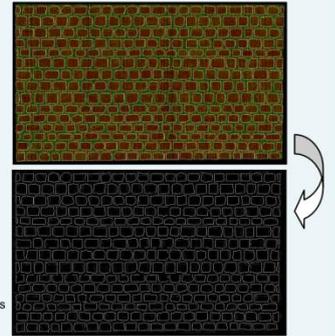


Figure 1 : Exemple de nuage de points dont les pierres sont détournées manuellement par GEOMAT [CloudWorx]

## Les méthodes étudiées

### Classification selon l'intensité laser

**Principe :** Labelliser chaque point du nuage par un numéro de classe thématique : classe n°1 : "pierres" et classe n°2 : "joints".

Réalisation d'un seuillage simple pour séparer les points de pierres des points de joints selon le principe du seuillage adaptatif.

⇒ Méthode développée en langage Python

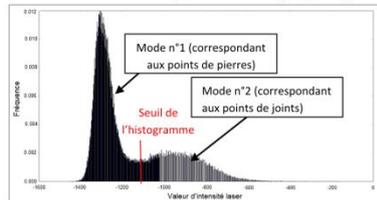


Figure 2 : Histogramme du nuage de points selon l'intensité laser [Python]

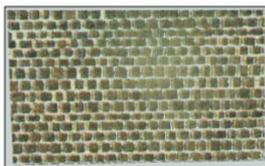


Figure 3 : Nuage de points des pierres résultant de la classification du nuage de points initial [Cyclone]

- taux de réussite : 93.39%
- taux de fausses alarmes : 3.36%

Très peu de points ont fait l'objet d'une mauvaise classification.

### Segmentation par croissance de régions selon l'intensité laser

**Principe :** Partitionner le nuage de points en régions homogènes à partir de points germes qui grossissent par agglomération de points voisins dès qu'ils vérifient un critère d'homogénéité.

⇒ Méthode développée en langage Python



Figure 4 : Nuage de points des pierres résultant de la segmentation du nuage de points initial [Cyclone]

- taux de réussite : 98.88%
- taux de fausses alarmes : 8.60%

Trop de points de joints ont été sélectionnés pour permettre une détection correcte des contours.

Une étude plus approfondie de la validation du critère d'homogénéité serait nécessaire pour obtenir de meilleurs résultats.

### Détection des contours par la méthode Alpha-shape après nettoyage du nuage de points classifié ou segmenté

Figure 5 : Contours du nuage classifié détectés par l'algorithme développé en langage Python

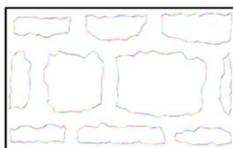


Figure 6 : Contours du nuage classifié détectés par le logiciel 3DReshaper



Si nous suivons un sens donné d'une polyligne créée sous 3DReshaper, nous remarquons que la normale en chaque point de cette polyligne n'est pas toujours orientée vers le centre de la pierre ce qui provoque des artefacts en forme de boucles qui sont longs à corriger. Ce phénomène n'est pas présent dans la méthode développée en langage Python. Cependant notre algorithme ne permet pas de traiter une grande quantité de données ce qui vient limiter le produit.

## La méthode retenue

### Classification selon l'intensité laser

L'algorithme de classification produit actuellement de meilleurs résultats que celui de segmentation.

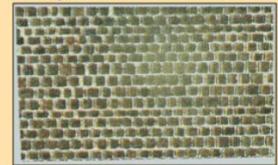


Figure 7 : Nuage de points des pierres résultant de la classification du nuage de points initial [Cyclone]

### Suppression des points isolés

Afin de garantir une détection optimale des contours, la suppression des points isolés est nécessaire.

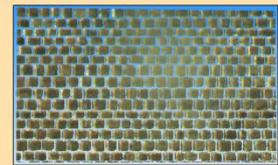


Figure 8 : Nuage de points des pierres résultant de la classification nettoyé [3DReshaper]

### Reconnaissance des formes



Figure 9 : Contours détectés par la méthode alpha-shape du logiciel 3DReshaper

### Correction des artefacts et export des polygones au format dxf

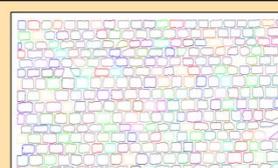


Figure 10 : Polygones corrigés et exportés au format dxf [AutoCAD]

**Conclusions et perspectives :** Plusieurs stratégies sont possibles pour répondre à cette problématique. Dans notre cas, la méthode retenue est celle qui répond actuellement le mieux à la demande avec une qualité de détection des contours pouvant aller jusqu'à 95%. Toutefois, ce chiffre est à relativiser puisqu'il dépend de plusieurs facteurs tels que la présence d'une intensité laser distincte entre les points de pierres et les points de joints, l'absence de végétation sur le nuage de points et une densité de points élevée.

Par ailleurs, la segmentation par croissance de régions améliorée ainsi que la détection alpha-shape développées en langage Python permettront d'adapter plus facilement les méthodes à des cas variés.

# **Reconnaissance automatique des contours de pierres sur nuages de points 3D acquis par laser scanner terrestre**

**Mémoire d'Ingénieur ESGT-Cnam, Le Mans 2015**

---

## **SYNTHÈSE**

Pour répondre à la demande actuelle en matière de levés architecturaux et plus spécifiquement concernant les levés pierre à pierre de monuments nationaux, la société GEOMAT s'est intéressée au développement d'une procédure de reconnaissance automatique des contours de pierres sur nuages de points 3D acquis par laser scanner terrestre.

Face à cette problématique, plusieurs méthodes ont été testées. Les méthodes de classification et de segmentation par croissance de régions sur nuages de points 3D selon l'intensité laser ont été développées en langage Python afin d'isoler les points de pierres des points de joints. Ensuite, la méthode alpha-shape, permettant la détection des contours de pierres sur les nuages classifiés ou segmentés, a été testée de deux manières différentes : à travers le développement d'un algorithme en langage Python d'une part, et sous le logiciel 3DReshaper d'autre part.

Ces différents tests ont permis de mettre en avant le processus de classification et de l'alpha-shape du logiciel 3DReshaper qui répond actuellement le mieux au cahier des charges. Toutefois, l'amélioration des développements de segmentation par croissance de régions et de l'alpha-shape en langage Python permettront par la suite de traiter plus facilement des cas variés.

**Mots clés : lasergrammétrie, nuage de points 3D, reconnaissance de formes, contours, classification, segmentation, alpha-shape, scripts Python, 3DReshaper.**

---

## **ABSTRACT**

In order to answer the current request regarding architectural surveys and more specifically stone determining surveys of national monuments, the company GEOMAT is interested in developing an automatic stone outlines recognition procedure on 3D point cloud acquired by terrestrial laser scanner.

Facing this problem, several methods have been tested. Classification and region growing segmentation on 3D point cloud using laser intensity have been developed in Python language in order to isolate stone points from bond points. After that, the alpha-shape procedure, allowing stone outlines detection on classified or segmented points clouds, has been tested in two different ways: with a Python algorithm development and with the software 3DReshaper.

These different tests demonstrated that the classification and the alpha-shape algorithms of 3DReshaper are the best answer to the surveys' needs. However, the improvement of the region growing segmentation and the alpha-shape algorithms in Python will allow easier processing of varied cases.

**Key words : laser scanning, 3D point cloud, pattern recognition, outlines, classification, segmentation, alpha-shape, Python scripts, 3DReshaper.**