

# De la carte de profondeur au maillage surfacique : reconstruction de scènes 3D complexes.

Arnaud BLETTERER<sup>1,2</sup>, Frédéric PAYAN<sup>1</sup>, Marc ANTONINI<sup>1</sup>, Anis MEFTAH<sup>2</sup>

<sup>1</sup>Université Côte d’Azur, CNRS, I3S, 2000 route des Lucioles, 06903 Sophia Antipolis, France,

<sup>2</sup>Cintoo3D, 1047 Route des dolines, 06560 Valbonne, France

{blettere, fpayan, am}@i3s.unice.fr ; meftah@cintoo3d.com

**Résumé** – Nous présentons une méthode originale de reconstruction de surfaces à partir de nuages de points. Notre méthode est spécifiquement dédiée aux nuages de points massifs représentant des scènes 3D complexes acquises par des scanners laser. Afin d’exploiter la régularité d’échantillonnage intrinsèque à de telles acquisitions, nous utilisons les cartes de profondeur comme domaine de paramétrisation. Cela fournit une connectivité implicite aux nuages de points, une information rarement considérée par les autres méthodes de reconstruction qui travaillent généralement dans l’espace 3D. Cela nous permet de reconstruire une surface 3D de bonne qualité directement à partir des données natives des scanners 3D.

**Abstract** – We present an original surface reconstruction method from point clouds. Our method is dedicated to massive point clouds representing complex 3D scenes acquired by range scanners. In order to exploit the sampling regularity inherent to such acquisitions, we choose to use depthmaps as parameterisation domain. It gives an implicit connectivity to point clouds, an information rarely considered by many prior methods, that generally work in 3D space. It allows us to reconstruct a surface of high quality, directly from native 3D scanners data.

## 1 Introduction

Les scanners 3D ont connu un véritable engouement durant les deux dernières décennies. Il est relativement facile maintenant pour un utilisateur lambda de numériser aussi bien la surface d’objets 3D que de vastes scènes telles que des bâtiments, des sites historiques, etc. Néanmoins, ces appareils fournissent des nuages de points, alors que la plupart des personnes souhaite manipuler la surface sous-jacente à ces nuages de points : par exemple en visualisation (gestion des occultations), dans l’animation (collision de surfaces) ou encore en métrologie (mesures sur la surface). De nombreuses méthodes de reconstruction ont donc été développées pour transformer ces nuages de points en maillages polygonaux [2]. Ces méthodes consistent généralement à connecter l’ensemble des points acquis, tout en atténuant les artefacts (bruit, trous, etc.).

La résolution de certains scanners 3D est devenue telle que l’on peut maintenant acquérir plusieurs dizaines de millions de points en une seule prise de vue. L’avantage est de modéliser des surfaces extrêmement détaillées. L’inconvénient est que la reconstruction directe de données aussi massives devient difficile, voire impossible avec la plupart des méthodes de l’état de l’art. C’est pourquoi il est fréquent de simplifier le nuage de points [12] avant de le mailler, ou de reconstruire la surface au fil de l’eau [1, 7]. Malheureusement, même ces approches peuvent s’avérer inefficaces pour des nuages de points repré-

sentant des scènes complexes, telles que celui représenté en figure 1 : nombreux bords, zones non connexes, fortes variations de densité, objets de tailles très différentes, etc.

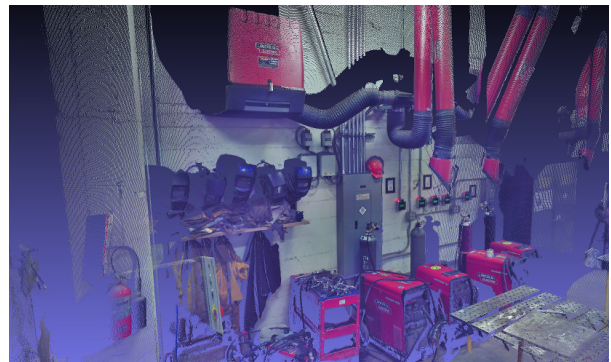


FIGURE 1 – Nuage de points acquis dans une scène complexe dont on souhaite obtenir la surface sous-jacente.

Dans cet article, nous présentons une méthode originale pour générer des maillages triangulaires à partir de tels nuages de points. Nous proposons pour cela de travailler sur les cartes de profondeur fournies par de nombreux scanners 3D, au lieu de travailler directement dans l’espace 3D. De cette manière, la connectivité des données et leur distance au scanner, liées à la méthode d’acquisition, sont prises en compte tout au long de notre algorithme. Nous considérons comme seules données d’entrée une carte de profondeur ainsi qu’une fonction de pa-

ramétrisation, permettant le passage de la carte de profondeur 2D à l'espace 3D et inversement.

## 2 Motivations

Une carte de profondeur est une image en deux dimensions dont l'intensité des pixels mesure la distance entre la position du scanner et la position des échantillons acquis sur la surface. Cela peut donc être vu comme une paramétrisation discrète 2D d'une scène 3D depuis un point de vue spécifique (position et orientation). Cette paramétrisation fournit une information de connectivité au nuage de points, les pixels voisins dans l'image représentant des points voisins sur la surface. Cette connectivité implicite facilite les requêtes de voisinage, complexes à gérer sur le nuage de points [11], et d'une manière générale tous les traitements nécessitant une navigation au sein du nuage de points.

C'est pourquoi l'ensemble des étapes de notre algorithme de reconstruction se déroule dans le domaine 2D fourni par les cartes de profondeur. Cela permet en plus de reconstruire un maillage directement à partir des données issues de l'acquisition, simplifiant ainsi le processus classique de reconstruction.

## 3 Description de notre méthode

Pour mailler une surface, de nombreuses méthodes [3] commencent par partitionner la géométrie à l'aide d'un diagramme de Voronoï. L'intérêt est que la représentation duale d'un tel diagramme permet d'obtenir directement une triangulation.

Inspirés par ces méthodes, nous proposons de partitionner la géométrie de la surface à reconstruire en utilisant plutôt la carte de profondeur associée au nuage de points acquis. Plus précisément, la carte de profondeur sert de domaine de paramétrisation pour générer un diagramme de Voronoï centroïdal de la surface acquise. A partir de cette partition 2D, nous obtenons un premier maillage 2D, grâce à sa représentation duale. Celui-ci n'a plus qu'à être plongé dans l'espace 3D pour obtenir le maillage final représentant la surface acquise.

Pour reconstruire des scènes complexes telles que celle de la figure 1, une telle approche ne sera pas satisfaisante. En effet, si la plupart du temps il est correct de considérer que des pixels voisins dans une carte de profondeur correspondent à des points voisins sur la surface, ce n'est pas le cas lors de fortes variations de profondeur dans la carte. Nous proposons donc de commencer par segmenter la carte en différentes régions, en fonction de leur éloignement par rapport au scanner. Les contours de ces régions, qui représentent les zones de fortes variations de profondeur, vont ensuite contraindre le partitionnement de la carte, évitant ainsi de connecter lors de la triangulation des régions voisines dans la carte, mais éloignées dans l'espace 3D. La figure 2 illustre les principales étapes de notre algorithme (détection de contours, partitionnement et triangulation), détaillées ci-après.

**Détection des contours** Afin de séparer des zones voisines dans la carte de profondeur mais représentant des morceaux de surfaces distincts et non-connectés en 3D, nous proposons de

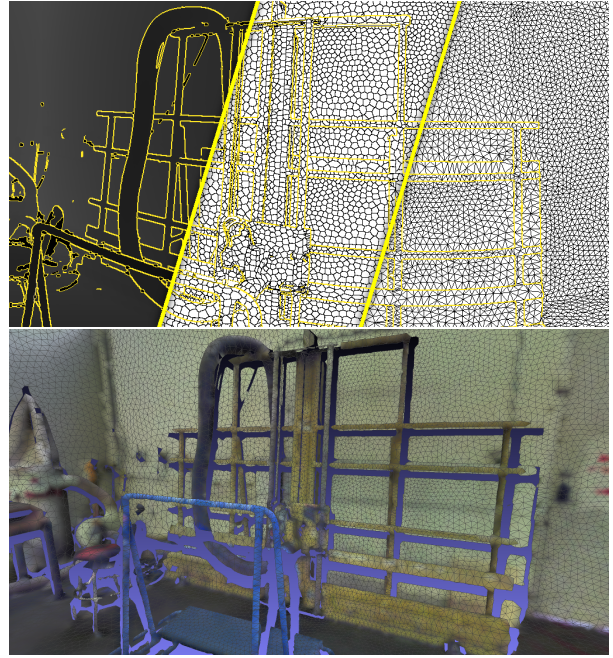


FIGURE 2 – Étapes de l'algorithme : détection des contours, partitionnement contraint, triangulation duale et maillage surfacique final. Donnée initiale : 10 millions de points.

déterminer les contours présents dans la carte de profondeur. Nous avons utilisé la méthode des gradients morphologiques [10]. Afin de distinguer des contours distincts de chaque côté des arêtes, nous identifions les gradients internes  $G_i(f)$  et externes  $G_e(f)$  (figure 3) :

$$G_i(f) = f - f \ominus b \quad (1)$$

$$G_e(f) = f \oplus b - f, \quad (2)$$

où  $f$  représente la carte de profondeur et  $b$  est un élément structurant. Dans notre contexte, comme nous souhaitons extraire des contours fins étant 8-connexes, nous avons choisi  $b$  comme étant un carré de taille  $3 \times 3$ .

Non seulement d'être simple en terme de complexité algorithmique par rapport à une approche 3D, cette méthode s'abstrait des problèmes liés aux différences de densité qui sont présents dans les données issues d'acquisition 3D. En effet, au sein d'une même acquisition, les surfaces proches du scanner et celles plus éloignées n'ont pas été échantillonnées à la même densité.

**Partitionnement basé Voronoï** La première étape pour partitionner un espace à l'aide d'un diagramme de Voronoï est de positionner le nombre souhaité de sites dans cet espace. Afin de répartir efficacement ces sites dans la carte de profondeur (pour obtenir une bonne partition de départ), nous utilisons une technique d'échantillonnage en disques de Poisson appelée "lancer de fléchettes" [5] (*dart throwing* en anglais).

Cette méthode sélectionne les échantillons de manière aléatoire, en imposant une contrainte d'espacement minimum entre les échantillons, grâce au positionnement d'un disque de rayon fixé, centré autour de chaque nouvel échantillon ajouté.



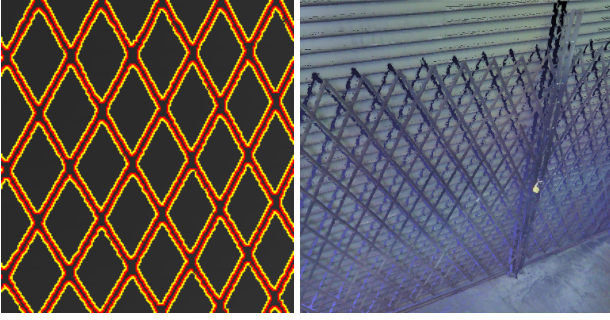


FIGURE 3 – Différents contours détectés grâce à notre méthode (en jaune les gradients internes et en rouge les gradients externe), à gauche, à partir du nuage de points présenté à droite.

Afin d'échantillonner la surface de manière uniforme, nous utilisons des disques géodésiques, approximés à partir du domaine de paramétrisation. De plus, afin de préserver les zones proches des contours détectés auparavant, on s'assure qu'un nombre minimum d'échantillons est positionné sur ces contours. La figure 4 montre un résultat obtenu avec notre approche. On y voit les échantillons retenus comme sites pour la génération d'un premier diagramme de Voronoï.

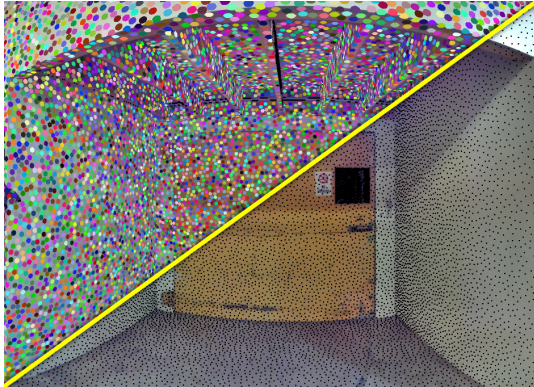


FIGURE 4 – Disques de Poisson distribués sur la carte de profondeur (à gauche). Échantillons obtenus grâce à ces disques (à droite), servant d'initialisation pour le diagramme de Voronoï.

Un diagramme de Voronoï  $V(X)$  partitionne un espace  $\mathbb{R}^p$  en un ensemble de cellules convexes  $\{V(x_i)\}$ , associées à un ensemble de  $n$  échantillons  $X = (x_1 \dots x_n) \in \mathbb{R}^p$  tel que

$$V(x_i) = \{x \in \mathbb{R}^p / d(x, x_i) \leq d(x, x_j), \forall j\},$$

où  $d(., .)$  est une métrique imposée. La métrique euclidienne est souvent utilisée mais il est possible d'utiliser d'autres métriques telles que la distance de Manhattan, la distance géodésique, etc. Dans notre contexte, bien qu'effectué dans le domaine de paramétrisation que nous fournit la carte de profondeur, le diagramme généré est basé sur les géodésiques calculées dans l'espace 3D (dans ce cas,  $p = 3$ ), afin d'obtenir *in fine* un partitionnement le plus uniforme possible de la surface sous-jacente.

De plus, afin de ne pas connecter les régions détectées lors de notre segmentation, notre diagramme est construit en imposant

des contraintes au niveau des contours identifiés, de manière similaire à [13].

Une fois le premier diagramme construit, nous appliquons l'algorithme de Lloyd [9] sur celui-ci pour converger vers un diagramme de Voronoï centroïdal [6], c'est-à-dire un diagramme dont les sites sont les centroïdes des cellules. L'intérêt est que la triangulation duale d'un tel diagramme génère un maillage surfacique présentant la meilleure qualité possible de triangles (triangles équilatéraux). Plus précisément, cet algorithme minimise itérativement sur l'ensemble des sites de Voronoï  $\{x_i\}$

$$\sum_{i=1}^n \int_{y \in V(x_i)} \rho(y) \|y - x_i\|^2 dy,$$

où  $\rho$  est une densité de probabilité définie sur le domaine  $\mathbb{R}^3$ . A chaque itération, chaque site  $x_i$  est déplacé sur le centroïde de sa cellule, puis le diagramme est redéfini, et ainsi de suite jusqu'à convergence.

De manière similaire à ce qui été fait lors des calculs des rayons de disques de Poisson, cette étape va être effectuée sur la carte de profondeur, mais en utilisant les géodésiques estimées sur le nuage de points associé. De plus, afin de bien reconstruire les zones proches des contours (voir figure 5), lorsque une cellule est adjacente à un contour, son site est déplacé par projection orthogonale du centroïde sur ce contour.

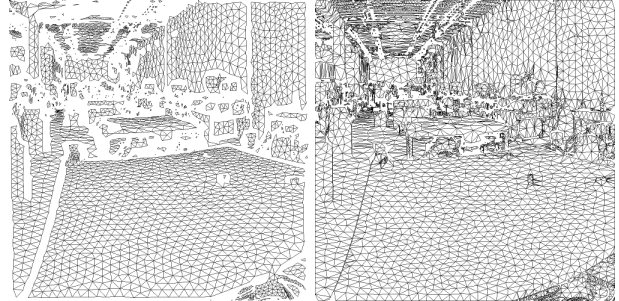


FIGURE 5 – Reconstruction d'une vue sans projection des sites sur les contours lors de la relaxation (à gauche), et avec projection (à droite).

**Triangulation** Après convergence de l'algorithme de Lloyd, on génère la triangulation 2D du diagramme obtenu en connectant tous les sites des cellules voisines. Puis, le maillage surfacique final est obtenu par simple plongement dans l'espace 3D du maillage 2D grâce à la fonction de paramétrisation associée à la carte de profondeur (fournie par les scanners 3D).

## 4 Résultats

L'ensemble des acquisitions présentées dans cet article provient du *ReCap community data center*<sup>1</sup>. Les figures 2 et 6 montrent des résultats de surfaces reconstruites avec notre méthode, contenant respectivement 200k et 700k triangles environ chacune, à partir d'acquisitions LiDAR de respectivement 6M et 10M de points, environ. Nous pouvons facilement voir

1. <https://autodesk.app.box.com/v/recapdata>

que notre méthode préserve les différentes structures contenues dans la scène, même les plus ardues à identifier, produisant ainsi un maillage de haute qualité. Notons qu’il n’est pas possible d’obtenir de tels résultats avec des algorithmes classiques de l’état de l’art, tels que [8], qui sont dans l’incapacité de reconstruire des maillages visuellement satisfaisants à partir de telles données en ne travaillant que dans l’espace 3D.

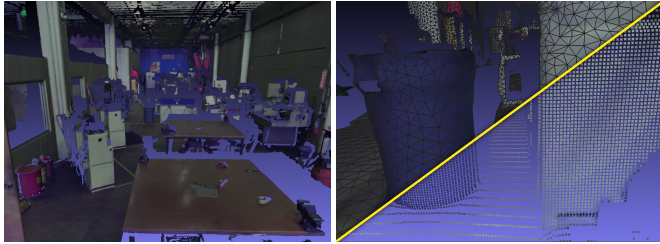


FIGURE 6 – A gauche, une surface reconstruite avec notre méthode. A droite, une vue rapprochée de la triangulation et nuage de points original (environ 10 millions).

La figure 7 montre l’impact de la taille du rayon des disques de Poisson sur la surface reconstruite à partir d’une acquisition LiDAR d’environ 10 millions de points. L’augmentation du rayon permet de réduire la finesse du maillage obtenu, donnant ainsi la possibilité de reconstruire, selon le besoin, un maillage plus ou moins détaillé. Pour s’assurer que ces maillages restent fidèles géométriquement à la surface, nous avons calculé la racine carrée de l’erreur quadratique moyenne (RMSE, en anglais) par rapport au nuage de points d’origine avec l’outil Metro [4]. Nous avons observé que cette erreur (normalisée par la diagonale de la boîte englobante du maillage original) est faible, de l’ordre de  $10^{-4}$ , même lorsque le maillage reconstruit contient peu de faces.

Enfin, pour avoir un ordre de grandeur des temps de calcul, la reconstruction d’une surface composée de 1,5 million de faces à partir de 10 millions de points prend environ 5mn (la majeure partie du temps étant dédiée à l’obtention du diagramme de Voronoï centroïdal).



FIGURE 7 – Surface reconstruite à différentes résolutions avec notre méthode à partir du nuage de points de gauche.

## 5 Conclusion

Cet article présente une nouvelle approche permettant de reconstruire efficacement des surfaces à partir de nuages de

points représentant des scènes 3D complexes. L’originalité vient du fait que l’on traite la surface sous-jacente directement à partir de la carte de profondeur générée par la plupart des scanners 3D actuels. Une telle carte peut être utilisée comme domaine de paramétrisation du nuage de points initial, fournissant ainsi une topologie à une donnée *a priori* sans connectivité. Grâce à une détection de contours combinée à un diagramme de Voronoï 2D appliqués sur la carte de profondeur, notre méthode est capable de reconstruire des maillages géométriquement fidèles à l’acquisition initiale, tout en préservant les différentes structures contenues dans la scène acquise. Nos travaux futurs se focaliseront sur l’extension de notre méthode à plusieurs vues acquises pour une même scène.

## Références

- [1] Rémi Allègre, Raphaëlle Chaine, and Samir Akkouche. A streaming algorithm for surface reconstruction. In *Symposium on Geometry Processing*, pages 79–88, 2007.
- [2] Matthew Berger, Andrea Tagliasacchi, Lee Seversky, Pierre Alliez, Joshua Levine, Andrei Sharf, and Claudio Silva. State of the art in surface reconstruction from point clouds. In *EUROGRAPHICS star reports*, volume 1, pages 161–185, 2014.
- [3] Siu-Wing Cheng, Tamal K Dey, and Jonathan Shewchuk. *Delaunay mesh generation*. CRC Press, 2012.
- [4] Paolo Cignoni, Claudio Rocchini, and Roberto Scopigno. Metro : Measuring error on simplified surfaces. In *Computer Graphics Forum*, volume 17, pages 167–174. Wiley Online Library, 1998.
- [5] Robert L. Cook. Stochastic sampling in computer graphics. *ACM Trans. Graph.*, 5(1) :51–72, January 1986.
- [6] Qiang Du, Vance Faber, and Max Gunzburger. Centroidal voronoi tessellations : Applications and algorithms. *SIAM Rev.*, 41(4) :637–676, December 1999.
- [7] Martin Isenburg, Yuanxin Liu, Jonathan Shewchuk, and Jack Snoeyink. Streaming computation of delaunay triangulations. *ACM Trans. Graph.*, 25(3) :1049–1056, July 2006.
- [8] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Trans. Graph.*, 32(3) :29 :1–29 :13, July 2013.
- [9] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2) :129–137, 1982.
- [10] Jean-Francois Rivest, Pierre Soille, and Serge Beucher. Morphological gradients. *J. Electronic Imaging*, 2(4) :326–336, 1993.
- [11] Jagan Sankaranarayanan, Hanan Samet, and Amitabh Varshney. A fast k-neighborhood algorithm for large point-clouds. In *Proceedings of the 3rd Eurographics / IEEE VGTC Conference on Point-Based Graphics*, SPBG’06, pages 75–84, Aire-la-Ville, Switzerland, Switzerland, 2006. Eurographics Association.
- [12] Bao-Quan Shi, Jin Liang, and Qing Liu. Adaptive simplification of point cloud using k-means clustering. *Computer-Aided Design*, 43(8) :910–922, 2011.
- [13] Jane Tournois, Pierre Alliez, and Olivier Devillers. 2d centroidal voronoi tessellations with constraints. *Numerical Mathematics : Theory, Methods and Applications*, 3(2) :212–222, 2010.