

Génération automatique de marqueurs pour la ligne de partage des eaux 3D

S. Delest G. Pageot R. Boné H. Cardot

Laboratoire Informatique

Université François-Rabelais de Tours
64, avenue Jean Portalis, 37200 TOURS

{sebastien.delest, romuald.bone, hubert.cardot}@univ-tours.fr

Concours Jeune Chercheur : Oui

Résumé

Cet article présente une nouvelle méthode de segmentation de maillage triangulaire basée sur la ligne de partage des eaux, initialisée par des marqueurs issus de la squelettisation de l'objet 3D. Dans cette méthode, le modèle est d'abord transformé en une représentation de voxels ; un algorithme de squelettisation est ensuite utilisé pour extraire le squelette constitué de voxels. Chaque branche du squelette est labellisée et les voxels de surface prennent le label des voxels du squelette qui leur sont associés. Les voxels de surface non ambigus peuvent ensuite servir de marqueurs pour la ligne de partage des eaux 3D. Cette méthode, qui associe la décomposition en partie et la décomposition en patches surfaciques, est particulièrement bien adaptée aux problématiques de segmentation d'objet qui comportent des parties significatives.

Mots clefs

Ligne de partage des eaux, squelettisation, marqueurs.

1 Introduction

Les maillages polygonaux sont couramment utilisés pour représenter des surfaces 3D ; en particulier les maillages triangulaires, qui offrent une structure simple et qui sont présents dans de nombreuses applications. Cet article concerne la segmentation de maillages triangulaires, cependant la méthode est tout aussi bien adaptée aux autres types de maillages.

La segmentation de maillages a de nombreuses applications dans les domaines de la visualisation et de la modélisation. La forme des modèles est importante et peut amener à différentes approches de segmentation selon qu'il s'agisse de formes naturelles ou de parties mécaniques. Les méthodes de segmentation de maillages sont classées principalement en deux groupes : la décomposition en patches surfacique qui tient compte des propriétés de planéité, de taille et de convexité et la décomposition en parties qui cible davantage le partitionnement en parties significatives de l'objet.

La décomposition en parties intervient dans de nombreux domaines comme l'appariement de formes, l'indexation et la reconstruction de formes par la reconnaissance des objets 3D, le morphing, la compression et la simplification de forme, la détection de collision, le mappage de texture, etc. Katz et al. [1] ont proposé une décomposition d'objets à partir du squelette pour permettre la déformation et l'animation du modèle. Wu et Levine [2] font intervenir les propriétés des charges électriques qui s'accumulent dans les zones de fortes convexités et disparaissent dans les zones de fortes concavités pour décomposer un modèle en parties. Koschan [3] utilise des opérateurs morphologiques comme outils de marquage et la ligne de partage des eaux pour segmenter les objets en parties. Lavoué et al. [4] ont fait intervenir des procédés de classification et de croissance de régions pour identifier les parties les plus significatives des objets 3D. Bruner et al. [5] offrent une décomposition du maillage par squelettisation et par association des branches du squelette aux faces du modèle.

Nous proposons ici une segmentation de maillages par décomposition en parties basée sur la squelettisation puis la ligne de partage des eaux (LPE). Le maillage 3D est d'abord voxelisé ; chaque face du modèle est convertie en groupe de voxels et la surface fermée qui en résulte est ensuite remplie. Nous avons alors utilisé l'algorithme de squelettisation proposé par Kálmán Palágyi [6] pour obtenir le squelette du modèle. Chacune de ses branches obtient un label différent et les voxels du squelette sont associés à des voxels de surfaces de l'objet, eux même associés aux faces du maillage d'origine. Les faces qui sont connectées à des voxels non ambigus reçoivent leur label et vont servir de marqueurs pour la LPE.

2 La Ligne de Partage des Eaux

La méthode de Ligne de Partage des Eaux proposée par Digabel et Lantuéjoul [7] est un outil morphologique qui a été longtemps considéré comme l'étape finale d'un processus de segmentation. De nombreux pré-traitements (filtres, opérateurs morphologiques) ont eu alors pour but de réduire le nombre de régions non significatives tout en conservant les contours réels. Néanmoins, les travaux de

Beucher [8] ont démontré les limites de ces méthodes et ont mis en avant une approche de niveau supérieur, en considérant alors la LPE comme un opérateur morphologique de base. De cette approche a résulté l'apparition des algorithmes de LPE hiérarchique et de LPE par marqueurs. Dans ce qui suit, nous proposons une LPE par marqueurs générés à partir du squelette du modèle 3D.

2.1 La LPE 3D

En deux dimensions, l'algorithme de la LPE consiste à simuler la montée des eaux sur le gradient (ou bien un autre paramètre associé aux pixels) de l'image d'entrée depuis ses minima locaux ou ses marqueurs (fig. 1). Cela permet de générer des lignes de partage aux endroits où les bassins se rejoignent, définissant ainsi un SKIZ géodésique de l'image. Un SKIZ géodésique est un ensemble de lignes continues divisant une image en un ensemble de régions d'influence équivalentes. Ainsi, on obtient au final un découpage de l'image en régions dans lesquelles l'intensité lumineuse des pixels est relativement homogène.

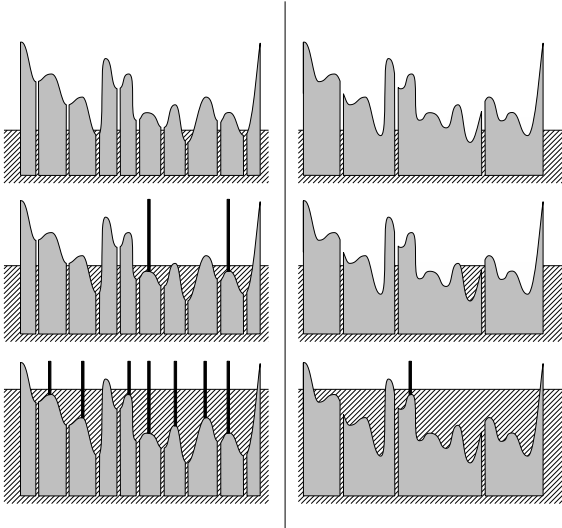


Figure 1 – Trois niveaux d'inondation pour la LPE avec minima à gauche et la LPE par marqueurs à droite. Les minima et les marqueurs sont les sources d'inondation.

En trois dimensions, l'intensité ou le gradient des pixels est remplacé par la courbure des vertex, de même que la connexité fixe des pixels devient alors variable (fig. 2).

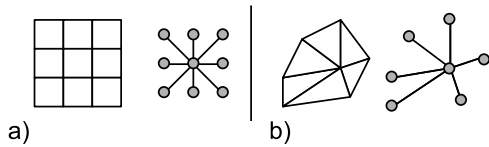


Figure 2 – a) Structure de pixels connectés, b) structure de vertex connectés

Pour construire la LPE, nous avons utilisé l'algorithme de calcul rapide de la LPE 2D par Files d'Attentes Hiérarchique (FAH) sans biais proposé par Serge Beucher [9] et adapté en trois dimensions dans [10]. Cette méthode consiste à créer autant de piles qu'il y a de niveaux de courbure dans le modèle traité. Les vertex observés dans le voisinage des vertex traités tout au long de l'immersion seront ainsi placés dans la pile correspondant à leur niveau. Les vertex labellisés comme minima ou marqueurs sont les premiers empilés. Un label différent est attribué aux vertex ou groupes de vertex isolés. Leurs voisins sont ensuite extraits et rangés dans la file correspondant à leur niveau de courbure (fig. 3). Ils reçoivent le label de leur vertex parent et lorsqu'un conflit apparaît, le vertex est marqué comme ligne de partage des eaux.

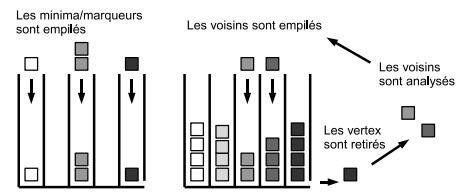


Figure 3 – Ligne de partage des eaux basée sur les Files d'Attentes Hiérarchiques.

2.2 Critère de courbure

Plusieurs approches ont été explorées pour obtenir une estimation de la courbure; Meyer et al. [11] ont proposé une étude intéressante sur les opérateurs discrets pour estimer la courbure et Mangan et Whitaker ont mis en avant l'efficacité de la norme de la matrice de covariance dans [12]. Dans notre cas, cette dernière méthode s'avère la plus adaptée pour caractériser la courbure des vertex qui correspond au critère de hauteur pour la ligne de partage des eaux. Le calcul de la courbure par la matrice de covariance repose sur un concept statistique qui consiste à évaluer les variances et covariances des coordonnées des différents vertices appartenant au voisinage. Pour un vertex donné, la courbure C est définie par la norme de la matrice de covariance :

$$C = \|M\| \text{ avec } M = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix}$$

$$\sigma_{uu}^2 = \frac{1}{N} \sum_{i=0}^N (u_i + \bar{u})^2$$

$$\sigma_{uv}^2 = \frac{1}{N} \sum_{i=0}^N (u_i + \bar{u})(v_i + \bar{v})$$

où σ_{uu} représente l'écart type des coordonnées en u du voisinage du vertex, and σ_{uv} représente la racine carrée de la covariance entre les composantes en u et les composantes en v . N correspond au nombre de triangles associés au vertex et $[x_t \ y_t \ z_t]^T$ est le vecteur de la normale du triangle t .

2.3 La sur-segmentation

L'utilisation de la ligne de partage des eaux seule ne permet pas réellement une bonne segmentation car beaucoup trop de régions sont détectées. Il existe deux principales méthodes pour limiter cette sur-segmentation : la segmentation hiérarchique et l'utilisation de marqueurs.

La segmentation hiérarchique L'approche hiérarchique peut consister à générer un arbre de régions à partir du résultat de la LPE. Les régions et les lignes de partages des eaux sont d'abord indexées, puis le processus de segmentation hiérarchique fait fusionner les régions dont les frontières communes sont les plus faibles. Il en résulte un arbre dans lequel il est possible d'explorer les différents niveaux de fusion des régions. La figure 4 propose deux segmentations avec des niveaux de fusions différents. Le modèle de gauche contient 208 régions et celui de droite 57 régions.



Figure 4 – Segmentation hiérarchique à partir de la ligne de partage des eaux. Modèle Cow avec 208 régions à gauche et 57 régions à droite.

Les marqueurs Ils vont définir les sources depuis lesquelles l'algorithme de la LPE va simuler la montée des eaux. Afin d'éviter la création de bassins au niveau des minima locaux, il est nécessaire d'effectuer une modification de l'homotopie de la structure d'entrée qui consiste à mettre les zones marquées au niveau le plus bas de la structure (fig. 1).

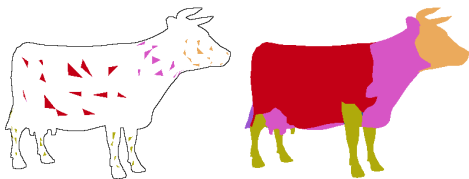


Figure 5 – marqueurs et résultat de la LPE.

Cette technique fournit des caractéristiques très intéressantes en terme de qualité de segmentation, de robustesse et de temps de calcul, aussi bien sur des objets industriels que naturels. Les faces ont été marquées manuellement dans l'exemple de la figure 5. Nous allons voir dans ce qui suit comment définir ces marqueurs de façon automatique.

3 Génération des marqueurs

Le marquage des régions qui vont servir de source d'inondation pour la LPE est réalisé à partir du squelette du modèle. La génération du squelette fait intervenir plusieurs procédés tels que la transformation du maillage en contour fermé de voxels, le remplissage du volume (voxelisation) et enfin la squelettisation du modèle.

3.1 La voxelisation

Brunner et Brunnet [5] ont proposé une méthode efficace pour stocker les voxels et réaliser la voxelisation sur un maillage fermé. La structure qui contient les voxels ne correspond pas à une image 3D mais à un plan ou tableau en deux dimensions qui, pour chaque case, intègre des couples de voxels. Ces voxels peuvent être associés aux entrées et sorties du rayon qui traverserait l'objet suivant une direction perpendiculaire au plan (fig. 6).

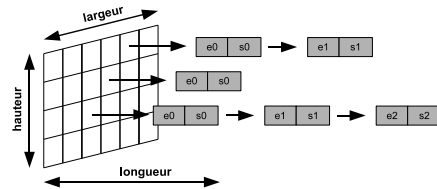


Figure 6 – Structure de stockage des voxels. Seuls les couples de voxels (entrée/sortie) sont référencés.

La voxelisation revient ici à déterminer l'intersection du rayon avec les faces du maillage. Le rayon entre dans l'objet puis en ressort, il peut y entrer à nouveau et en ressortir. Karabassi et al. [13] ont proposé un algorithme très rapide de voxelisation basé sur l'utilisation de 6 z-buffers (un z-buffer est lié à une direction de visualisation de l'image 3D); cependant, cette méthode ne prend pas en compte les parties internes ou cachées de l'objet.

3.2 La squelettisation

Pour extraire le squelette de voxels, nous avons utilisé l'algorithme de squelettisation de Palágyi dans [6] qui présente des avantages de rapidité et d'efficacité en terme d'érosion des différentes couches du volume. Cet algorithme supprime successivement les voxels dans l'image 3D selon certaines contraintes géométriques. Pour réaliser une érosion symétrique, six érosions sont successivement appliquées sur les voxels de surface dans les directions Haut, Bas, Nord, Sud, Est et Ouest (fig. 7a). A chaque érosion, seuls les voxels directement visibles à partir de la direction donnée sont testés; si ceux-ci peuvent être enlevés sans que leur suppression ne modifie la topologie de l'objet, alors ils sont rangés dans une liste sans être encore retirés et sont considérés comme *points simples*. Cette liste est ensuite consultée et ces voxels peuvent être supprimés si leur simplicité n'a pas été affectée par la suppression des autres points simples. Un voxel p de l'objet est appelé point

simple si sa suppression ne modifie pas la topologie de l'objet, c'est-à-dire si le nombre de composantes connexes et le nombre de trous de l'objet et de son complémentaire, dans le voisinage $N_{26}(p)$, restent inchangés après suppression de p . Le point p est simple s'il réunit les conditions suivantes :

1. Le point p ne doit pas être un point isolé ni un point extrémité.
2. Les voisins pleins dans le groupe $N_{26}(p) \setminus \{p\}$ sont 26-connectés à ce même groupe. Le nombre de composantes connexes doit rester le même.
3. Le point p est 6-adjacent à un point blanc. C'est un point de bord.
4. Les voisins vides dans le groupe $N_6(p) \setminus \{p\}$ sont 6-connectés au groupe de voxels vides $N_{18}(p) \setminus \{p\}$.

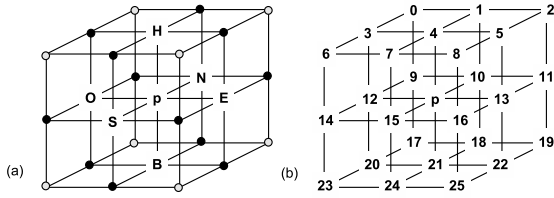


Figure 7 – (a) Le groupe $N_6(p)$ contient le point central p et les six points marqués H, B, N, S, E et O. Le groupe $N_{18}(p)$ contient le groupe $N_6(p)$ et les 12 points marqués "•". Le groupe $N_{26}(p)$ contient le groupe $N_{18}(p)$ et les 8 points marqués "○". (b) Indices assignés aux points dans le groupe $N_{26}(p) \setminus \{p\}$.

L'algorithme de squelettisation fait intervenir une fonction qui réalise la suppression successive des points simples suivant l'ordre Haut, Bas, Nord, Sud, Est et Ouest jusqu'à l'obtention du squelette. A chaque itération, seuls les voxels de surface par rapport à la direction donnée sont traités, ce qui valide la condition 3 ; ces voxels ne doivent pas être isolés ou extrémités pour valider la condition 1. Les conditions 2 et 4 peuvent alors être testées suivants les algorithmes **Cond2Satisfaite** et **Cond4Satisfaite**. Ces algorithmes sont extraits de [6] et corrigés dans notre article car la phase de labellisation n'était pas complète.

La fonction **Cond2Satisfaite** utilise deux structures de données auxiliaires : la première correspond au tableau d'entiers L , où $L[i]$ stocke les labels assignés aux éléments représentés par $Np[i]$ ($i = 0, \dots, 25$). La seconde est la clé du processus de labellisation : $S26$ est un tableau d'indice, où $S26[i] = \{j | j \in N_{26}[i] \text{ et } 0 \leq j < i\}$ ($i = 0, \dots, 25$). On aura ainsi $S26[0] = \emptyset$, $S26[1] = \{0\}$, \dots , $S26[25] = \{13, 15, 16, 21, 22, 24\}$ (voir fig. 7b). Tous les groupes $S26[0], \dots, S26[25]$ peuvent être stockés dans un tableau prédéfini. Les voisins pleins du voxel p sont 26-connectés si le même label est attribué à chacun des voisins pleins de p .

Fonction Cond2Satisfaite(Np)

début

```

label ← 0
lst ← nouvelle liste vide
pour i ← 0 à 25 faire L[i] ← 0
si Np[0] = 1 alors
  label ← 1
  L[0] ← label
pour i ← 1 à 25 faire
  si Np[i] = 1 alors
    label ← label + 1
    L[i] ← label
    pour chaque j ∈ S26[i] faire
      si L[j] > 0 alors
        pour k ← 0 à i - 1 faire
          si L[k] = L[j] alors Insert(lst,k)
    tant que lst ≠ ∅ faire
      l ← Retire(lst)
      L[l] ← label
pour i ← 0 à 25 faire
  si Np[i] = 1 et L[i] ≠ label alors
    retourner [FAUX]
retourner [VRAI]

```

fin

Fonction Cond4Satisfaite(Np)

début

```

label ← 0
lst ← nouvelle liste vide
pour i ← 0 à 25 faire L[i] ← 0
si Np[4] = 0 alors
  label ← 1
  L[4] ← label
pour i ← 1 à 17 faire
  indice ← N18[i]
  si Np[indice] = 0 alors
    label ← label + 1
    L[indice] ← label
    pour chaque j ∈ S18[i] faire
      si L[j] > 0 alors
        pour k ← 0 à indice - 1 faire
          si L[k] = L[j] alors Insert(lst,k)
    tant que lst ≠ ∅ faire
      l ← Retire(lst)
      L[l] ← label
pour i ← 0 à 5 faire
  indice ← N6[i]
  si Np[indice] = 0 et L[indice] ≠ label alors
    retourner [FAUX]
retourner [VRAI]

```

fin

La fonction **Cond4Satisfaite** utilise les mêmes principes mais cette fois-ci, ce sont les connexions entre points blancs qui sont analysées et ces points, dans le groupe $N_6(p)$, doivent être *6-connectés* au groupe de points blancs $N_{18}(p)$. Les voisins *6-adjacents* ont les indices 4, 10, 12, 13, 15 et 21 dans la figure 7b. La clé de labellisation S18 intègre les groupes suivants $S18[0] = \emptyset$, $S18[1] = \emptyset$, ... $S18[2] = \{1,3\}$, $S18[3] = \{4\}$, ... $S18[17] = \{15,21\}$

3.3 Création du marquage

En considérant le squelette du modèle comme un graphe, il apparaît deux principaux types d'éléments : les arcs et les sommets. Les arcs contiennent tous les voxels étant connectés à un ou deux autres voxels et les sommets correspondent aux voxels de jonction qui sont connectés à au moins trois autres voxels.

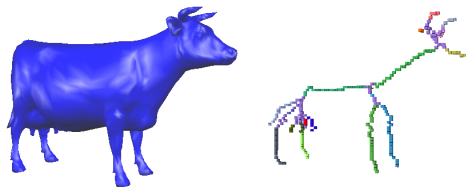


Figure 8 – Le modèle "Cow" à gauche et son squelette à droite. Les branches du squelette apparaissent d'une couleur différente.

Le marquage peut commencer par l'attribution d'un label différent à chaque arc comme sur la figure 8. Lors de la création du contour fermé de voxels, les voxels de surface deviennent liés aux faces du maillage qui leur correspondent. Ces connexions sont transmises aux nouveaux voxels de surface lors de chaque érosion (Figure 9).

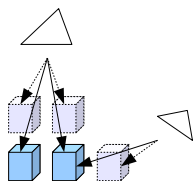


Figure 9 – Transmission des connexions de faces aux nouveaux voxels de surface les plus proches

Le squelette du modèle contiendra alors des voxels liés aux faces du maillage et les labels des voxels du squelette pourront être directement transmis aux faces (Figure 10). Les faces qui sont liées à des voxels ambigus ne seront pas labellisées à cette étape mais le seront après la segmentation par ligne de partage des eaux.

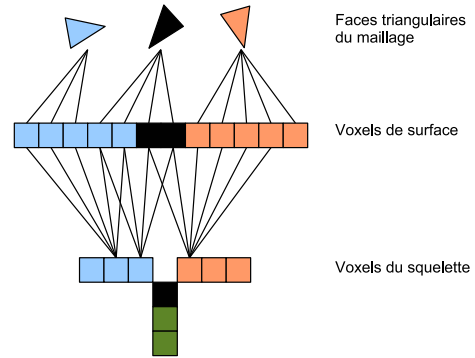


Figure 10 – Marquage des faces triangulaires du maillage. Les voxels de surface ambigus sont labellisés en noir ; leurs faces associées ne vont pas servir de marqueurs pour la LPE. Seules les faces liées à des voxels de couleur non noire dans l'exemple pourront servir de marqueurs.

4 Résultats



Figure 11 – Marqueurs générés par squelettisation à gauche et résultat de la segmentation par LPE avec marqueurs à droite.

Les expérimentations ont été réalisées avec un ordinateur équipé d'un processeur cadencé à 2.8Ghz et de 512Mo de mémoire. La figure 11 montre les résultats de la génération automatique des marqueurs et de la segmentation par ligne de partage des eaux. La résolution utilisée correspond à 100 voxels pour la dimension longueur, hauteur ou largeur maximale. Les marqueurs font ressortir les principales régions et la méthode de ligne de partage des eaux établie une frontière entre ces régions par rapport à la courbure de la surface du modèle.

La segmentation apparaît comme très efficace avec la combinaison de la génération automatique des marqueurs et la

ligne de partage des eaux. Les processus de squelettisation et de marquage offrent d'excellents marqueurs pour la ligne de partage des eaux. Les principales parties des modèles sont repérées et la finesse du marquage peut être paramétrée à partir de la résolution de voxels.

Modèles	Nombre de vertex	Volume en voxels	Temps de calcul (s.)	
			Marquage	LPE
Cow	2903	38321	15.6	0.062
Triceratops	4152	22282	6.9	0.125
Dinosaur	42146	15582	5.2	9.25

Tableau 1 – Temps de calcul du processus de marquage et de la segmentation par ligne de partage des eaux.

Le nombre de vertex d'un modèle va directement influencer le temps de calcul de la LPE. Le temps de calcul du marquage (qui inclut la squelettisation) dépend fortement du volume en voxels du modèle. Le tableau 1 fait apparaître une comparaison des différentes caractéristiques des modèles. Le modèle "Cow" contient le plus petit nombre de vertex et le calcul de la ligne de partage des eaux sera pour cela le plus rapide (62ms), cependant son volume est 1.7 fois plus important que celui du "Triceratops" et 2.4 fois plus grand que celui du "Dinosaur"; le temps de calcul du marquage sera donc assez conséquent. Le modèle "Dinosaur" est avantagé par son faible volume mais pénalisé par son nombre important de vertex, il en résultera une squelettisation rapide mais un temps de calcul plus grand pour la LPE.

5 Conclusions

Dans cet article, nous avons présenté une nouvelle méthode pour calculer automatiquement les marqueurs pour la segmentation par ligne de partage des eaux sur des maillages triangulaires. Les marqueurs permettent de faire apparaître les régions importantes du modèle et les régions incertaines sont laissées au processus de segmentation de surface. Les marqueurs générés automatiquement sont bien positionnés et permettent ainsi de s'affranchir de la lourde tâche du marquage manuel. La méthode offre de bons résultats et les principales parties des modèles sont détectées. Nos expérimentations ont montré qu'une résolution d'une centaine de voxels, pour la dimension la plus grande l'objet, était suffisante au processus de segmentation pour repérer les principales parties du modèle.

Remerciements

Les modèles utilisés proviennent des sites Internet de Cyberware (www.cyberware.com) et Avalon.

Références

[1] Sagi Katz et Ayellet Tal. Hierarchical mesh decomposition using fuzzy clustering and cuts. *ACM Transactions on Graphics*, 22(3) :954–961, July 2003.

[2] Kenong Wu et Martin D. Levine. 3d part segmentation using simulated electrical charge distributions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(11) :1223–1235, 1997.

[3] D. L. Page, A. Koschan, et M. Abidi. Perception-based 3d triangle mesh segmentation using fast marching watersheds. Dans *Proceedings of the Computer Vision and Pattern Recognition Conference*, volume II, pages 27–32, June 2003.

[4] Guillaume Lavoué, Florent Dupont, et Atilla Baskurt. A new cad mesh segmentation method, based on curvature tensor analysis. *Computer-Aided Design*, 37(10) :975–987, September 2004.

[5] David Brunner et Guido Brunnett. Mesh segmentation using the object skeleton graph. Dans *International Conference on Computer Graphics and Imaging*, pages 48–55, Kauai, Hawaii, USA, August 2004.

[6] Kálmán Palágyi, E. Sorantin, E. Balogh, A. Kuba, Cs. Halmai, B. Erdöhelyi, et K. Hausegger. A sequential 3d thinning algorithm and its medical applications. Dans *in Proc. 17th Int. Conf. Information Processing in Medical Imaging*, 2001.

[7] H. Digabel et C. Lantuéjoul. Iterative algorithms. Dans *2nd European Symp. Quantitative Analysis of Microstructures in Material Science*, pages 85–99, Caen, France, 1978. Biology and Medicine.

[8] Serge Beucher. Watershed, hierarchical segmentation and waterfall algorithm. Dans *Mathematical morphology and its applications to image processing*, pages 69–76. Kluwer Academic Publishers, 1994.

[9] Serge Beucher. Algorithmes sans biais de ligne de partage des eaux. Rapport technique, Centre de Morphologie Mathématique de l'École des Mines de Paris, Avril 2004.

[10] Jonathan Betsier, Sébastien Delest, et Romuald Boné. Segmentation 3d hiérarchique par ligne de partage des eaux sans biais. Dans *Actes de 15ème Congrès Francophone de Reconnaissance des Formes et Intelligence Artificielle (RFIA 2006)*, Tours, Janvier 2006.

[11] Mark Meyer, Mathieu Desbrun, Peter Schröder, et Alan H. Barr. Discrete differential-geometry operators for triangulated 2-manifolds. Dans Hans-Christian Hege et Konrad Polthier, éditeurs, *Visualization and Mathematics III*, pages 35–57. Springer-Verlag, Heidelberg, 2003.

[12] Alan P. Mangan et Ross T. Whitaker. Partitioning 3d surface meshes using watershed segmentation. *IEEE Transactions On Visualization And Computer Graphics*, 5(4) :308–321, October-December 1999.

[13] Evaggelia-Aggeliki Karabassi, Georgios Papaioannou, et Theoharis Theoharis. A fast depth-buffer-based voxelization algorithm. *J. Graph. Tools*, 4(4) :5–10, 1999.