

Segmentation de maillages 3D de pièces manufacturées numérisées : Application à la rétro-conception

S. Gauthier^{1,2} R. Bénére² W. Puech¹ G. Subsol¹

¹ Equipe ICAR, LIRMM, Université de Montpellier / CNRS, 860 rue de St Priest, 34 095 Montpellier Cedex 5, France

² C4W, 219 rue le Titien 34 000 Montpellier, France

E-mail : silvere.gauthier@lirmm.fr.

Résumé

De nos jours, il est de plus en plus fréquent et facile de numériser en 3D la surface des objets réels. Les maillages issus de ces objets numérisés sont souvent imprécis et bruités. De plus, nombre d'entre eux sont de haute résolution, donc très lourds. Dans une chaîne de rétro-conception, bien souvent une étape de segmentation est nécessaire. Dans cet article, nous proposons une nouvelle approche de segmentation pour la rétro-conception utilisant les spécificités des maillages numérisés.

Mots clefs

Maillage 3D, Segmentation, Rétro-conception, Numérisation surfacique.

1 Introduction

La rétro-conception regroupe tous les procédés étudiant un objet afin d'en déterminer le fonctionnement interne ou la méthode de fabrication. Cela permet, par exemple, de pouvoir reconstituer numériquement un objet 3D composé d'un ensemble de primitives géométriques telles que des plans, des sphères ou des cylindres. De nombreuses méthodes de rétro-conception ont été proposées ces dernières années (voir par exemple [1], ou plus récemment [2]).

La plupart d'entre elles sont basées sur les paramètres de la courbure du maillage. Or, le calcul de ces paramètres sur un maillage numérisé discret peut être faussé par le bruit ou l'imprécision sur les coordonnées des sommets.

Ainsi, sur le maillage de la figure 1, nous avons calculé un paramètre de courbure, le shape index, par la méthode décrite dans [2] avec différentes valeurs de k -voisinage. Le k -voisinage d'un point P contient tous les sommets accessibles depuis P en parcourant au plus k arêtes.

Les points des parties convexes du maillage ont un shape-index entre 0 et 1 et sont représentés en bleu, les points des parties concaves entre -1 et 0 apparaissent en rouge et pour les points des parties localement planes, le shape-index n'est pas défini mais ils sont colorés en vert.

Avec un k -voisinage restreint (figure 1.a), bien que ce maillage soit issu d'un objet numérisé contenant des plans,

il n'y a aucun sommet détecté comme plan. Tandis qu'avec un k -voisinage étendu (figures 1.b), plusieurs points sont détectés plans mais uniquement au centre des larges parties planes. En augmentant encore le k -voisinage (figure 1.c), les plans étroits deviennent difficilement détectables.

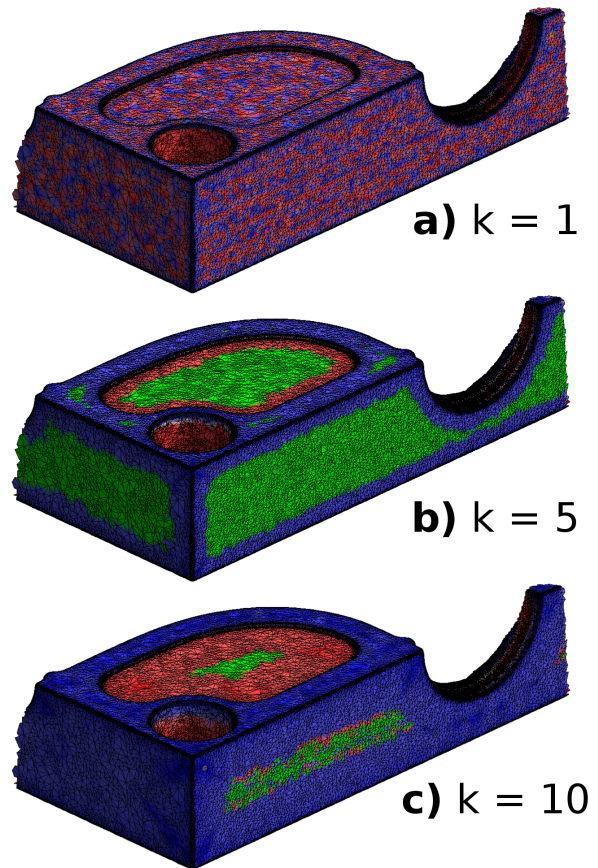


Figure 1 – Représentation des valeurs du shape-index d'un maillage 3D en fonction du k -voisinage (vert : plan, bleu : convexe, rouge : concave).

Nous voyons donc que l'utilisation directe du calcul des paramètres de courbure pour extraire des zones homogènes [3] ou des primitives [2] à partir de maillages 3D bruités n'est pas satisfaisante. Pour résoudre ce problème, nous

pouvons appliquer une segmentation avant de calculer la courbure, afin de limiter la portée du voisinage d'un sommet à d'autres sommets appartenant à la même zone homogène. Il devient alors possible d'étendre le k -voisinage afin de pallier au bruit, tout en évitant de déborder sur une zone voisine.

Pour valider cette approche, nous avons réalisé une segmentation manuelle de l'objet de la figure 1, en définissant les contours de sous-maillages "uniformes" (dans cet exemple, il s'agit de définir les arêtes). Les paramètres de courbure sont ensuite recalculés séparément pour chaque sous-maillage, et nous obtenons le résultat présenté dans la figure 2. Le calcul de courbure sur les sommets n'est plus perturbé par les zones voisines, et nous obtenons ainsi des paramètres de courbure plus cohérents.

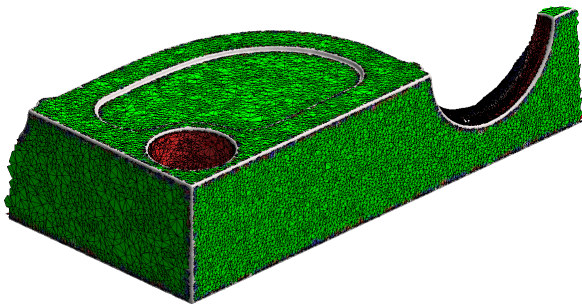


Figure 2 – Représentation des valeurs du shape-index avec un 5-voisinage après segmentation (vert : plan, rouge : concave).

Cette étape de segmentation va permettre d'améliorer la qualité du processus global de reconstruction et aussi de diminuer le temps de traitement.

La segmentation de maillages est un problème très courant en modélisation 3D (voir par exemple les revues de l'état de l'art [4] et [5]). Les méthodes peuvent être classées en trois catégories, à savoir les méthodes de croissance de régions [3][6], les méthodes utilisant des lignes caractéristiques [7], et enfin les méthodes de segmentation par patchs surfaciques [8].

Pour autant, les méthodes classiques ne permettent pas d'isoler les régions définissant les arêtes de l'objet, où le calcul de la courbure reste très instable car les valeurs sont souvent très élevées. De plus, détecter ces régions arêtes permettrait de les traiter différemment dans le processus de rétro-conception puisqu'elle vont définir de fait les relations de voisinage entre les primitives géométriques. Dans la suite, nous proposons donc une nouvelle approche de segmentation de maillages numérisés permettant de distinguer les arêtes du reste du maillage.

2 Présentation de la méthode

Cette segmentation se décompose en quatre parties, à savoir une détection des arêtes vives, une segmentation en régions basée sur les arêtes détectées, une fusion des petites régions et une fusion des régions non pertinentes. Un schéma récapitulatif est illustré figure 3, où les parties noires sont celles qui ne sont pas prises en compte.

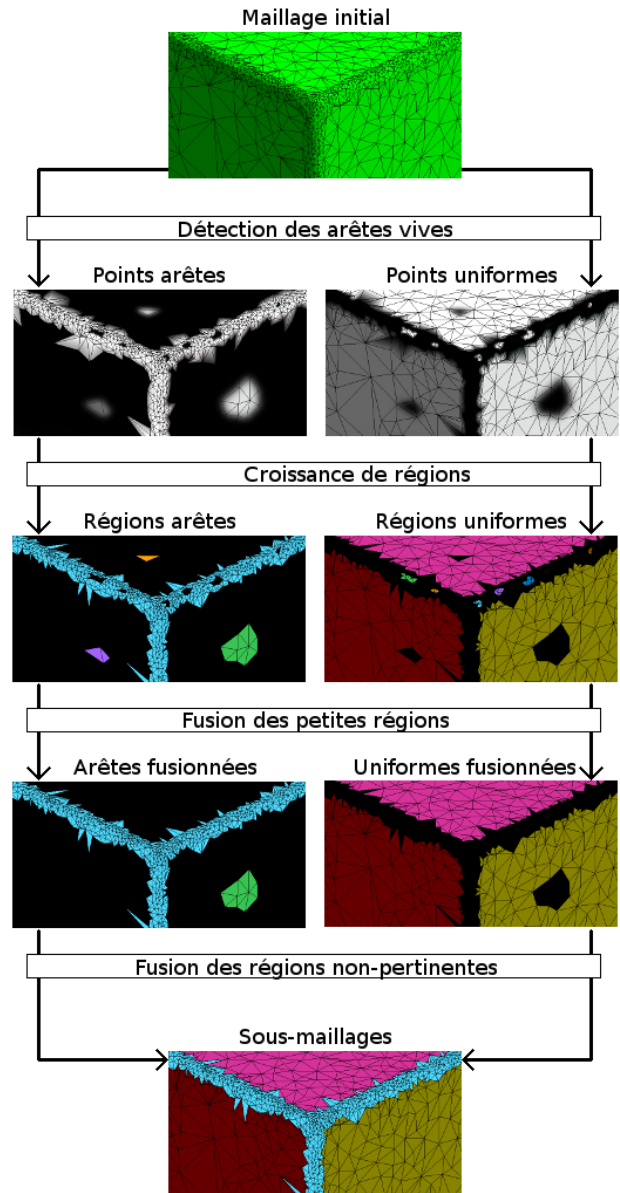


Figure 3 – Décomposition de la méthode proposée.

2.1 Détection des arêtes vives

L'enjeu de cette étape est de détecter les arêtes vives de l'objet. Pour cela, des valeurs de courbure moyenne (moyenne des courbures principales k_1 et k_2) sont calculées par la méthode de [2] sur les sommets, avec un voisinage de 2.

Les sommets sont ensuite catégorisés selon leur courbure, en tant que point arête ou uniforme, tel que :

$$|Uniforme| < Seuil < |Arete|. \quad (1)$$

Nous proposons d'utiliser un seuil basé sur les dimensions de l'objet :

$$Seuil = \sqrt[3]{\frac{NombreDePoints}{VolumeBoiteEnglobante}}. \quad (2)$$

Cette formule a été obtenue de manière empirique à partir de nombreux maillages de pièces manufacturées numérisées. Ainsi, nous obtenons une détection efficace des arêtes, comme illustré figure 4, où les points arêtes sont en noir.

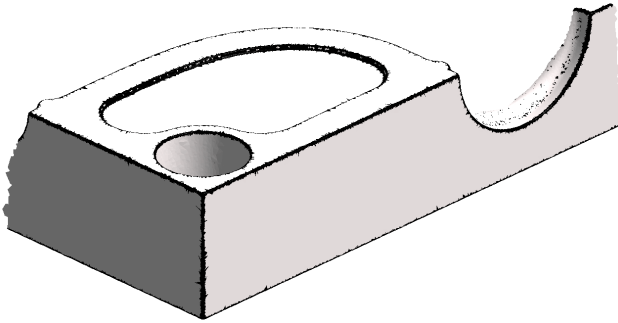


Figure 4 – Détection des points arêtes vives (illustrés en noir).

Nous avons constaté qu'un k -voisinage de 1 est un peu trop sensible au bruit. Nous avons donc choisi un k -voisinage de 2 car au delà, les résultats perdent en précision (voir figure 1).

De la même manière, nous avons choisi de seuiller la courbure moyenne qui fournit de meilleurs résultats que l'utilisation des courbures principales.

2.2 Croissance de régions

Nous avons obtenu, grâce à l'étape précédente, un ensemble de points arêtes ou uniformes. Nous proposons ici de regrouper ces points afin de former, cette fois-ci, des régions arêtes ou uniformes.

La segmentation va s'effectuer par une croissance de régions dont la connexité est définie par un voisinage "Triangle-Triangle".

La méthode initialise d'abord tous les triangles du maillage à une valeur signifiant qu'ils ne sont affectés à aucune région pour l'instant. Ensuite, un appel à la fonction de croissance, détaillée dans l'algorithme 1, est effectué sur chaque triangle de région indéfinie et ses voisins, jusqu'à ce que tous les triangles aient été traités.

Algorithme 1 Fonction de croissance

ENTRÉES: Le triangle de départ T

SORTIES: La région "arête" ou "uniforme" contenant T

Region \leftarrow List()

Region.PushBack(T)

$i \leftarrow 1$

tant que $i < \text{Region.Size}()$ **faire**

$T_i \leftarrow \text{Region}[i]$

Marquer T_i comme traité

pour tout triangle T_v voisin de T_i **faire**

si T_v non-traité ET ((T_i et T_v arêtes) OU (T_i et T_v uniformes)) **alors**

Region.PushBack(T_v)

fin si

fin pour

$i \leftarrow i + 1$

fin tantque

renvoyer Region

Après cette croissance de régions, nous obtenons un ensemble de sous-maillages de type uniforme ou arête, comme illustré figure 5, où chaque couleur correspond à un sous-maillage différent. Nous remarquons alors que le bruit des maillages numérisés a pour effet de créer de nombreuses petites zones, ayant très peu de triangles, comme illustré sur le zoom de la figure 5. Nous proposons alors une étape permettant de fusionner ces régions avec leur plus grande région voisine.

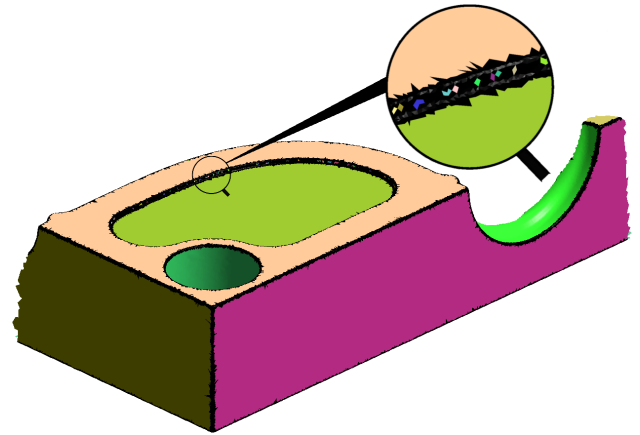


Figure 5 – Segmentation par croissance de régions après la détection des arêtes vives.

2.3 Fusion des petites régions

La fusion des régions utilise les prérequis de la segmentation, à savoir que le maillage ne peut comporter que deux types de région : "arête" ou "uniforme". Ainsi, nous savons qu'une région ne peut pas avoir une région de même type parmi ses voisines, conséquence de la croissance de régions. Néanmoins, il peut arriver que certaines régions aient plusieurs voisines différentes, toutes du type opposé.

Nous commençons donc par fusionner chaque petite région ayant une unique voisine avec cette même région voisine. Une fois que toutes les régions ayant une unique voisine sont fusionnées, nous pouvons faire de même avec les régions ayant plusieurs voisines, en prenant soin de fusionner les voisines entre elles également. Nous réitérons alors cette étape tant qu'il reste des régions à fusionner.

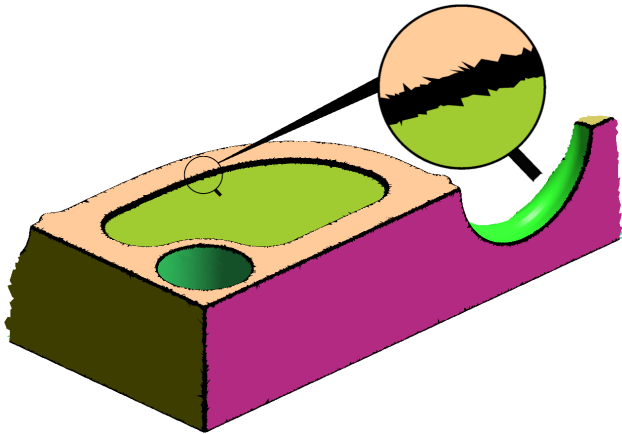


Figure 6 – Fusion des petites régions à partir de l'exemple illustré figure 5.

Après cette fusion de régions, nous obtenons un ensemble de sous-maillages, comme illustré figure 6, où chaque couleur correspond à un sous-maillage différent. Contrairement à la figure 5, nous remarquons que le maillage de la figure 6 ne contient plus de petite région.

2.4 Fusion des régions non pertinentes

Après la fusion des petites régions, il peut arriver que deux sous-maillages du même type soient voisins, ou qu'une région arête n'ait qu'un seul voisin alors que par définition, elle doit séparer deux régions uniformes. Nous devons donc fusionner ces régions non pertinentes.

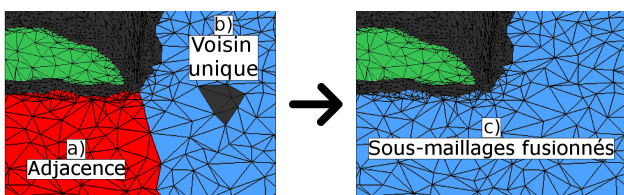


Figure 7 – Fusion des régions non pertinentes.

Dans le cas d'adjacence entre deux sous-maillages uniformes, illustré figure 7.a, nous fusionnons les régions en une seule. Dans le cas d'une région arête ayant un unique sous-maillage uniforme voisin, comme illustré figure 7.b, la région arête isolée est fusionnée avec le sous-maillage uniforme. Le résultat final est illustré figure 7.c.

3 Résultats expérimentaux

Dans cette section, des résultats obtenus avec notre méthode de segmentation sont étudiés. Les résultats présentés sont ceux obtenus sur trois maillages : *Shoe*, *Block* et *Mold*, illustrés respectivement figures 8.a, 8.b et 8.c.

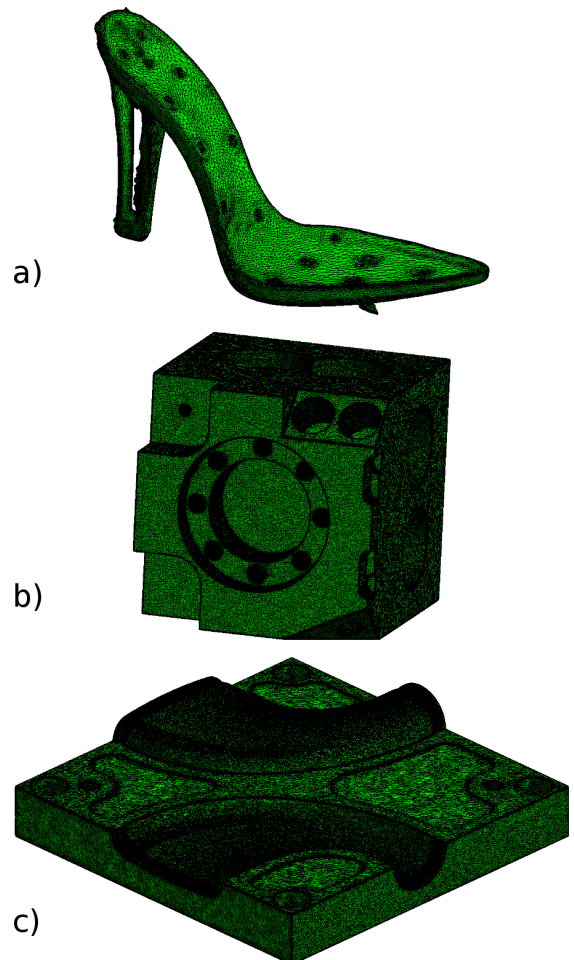


Figure 8 – Exemple de maillages numérisés : a) *Shoe* - 74 536 triangles, b) *Block* - 1 125 832 triangles, c) *Mold* - 851 194 triangles.

Ces trois maillages sont très différents par leur taille, leur forme et leur répartition des points. Ils sont respectivement composés de 74 536 triangles (37 252 sommets), 1 125 832 triangles (562 914 sommets) et 851 194 triangles (425 589 sommets).

Sachant que cette méthode est intégrée dans une application industrielle, il est à noter qu'aucun paramètre n'est demandé à l'utilisateur. Chaque paramètre utilisé lors de la segmentation est soit fixé, soit calculé, ce qui permet une grande simplification de son utilisation. Le but est ici de présenter une méthode pouvant être appréhendée par des utilisateurs non-experts.

3.1 Résultats de la segmentation

La première étape de notre segmentation consiste à détecter les arêtes vives du maillage. Les résultats de cette détection sur les trois maillages tests sont illustrés figure 9.

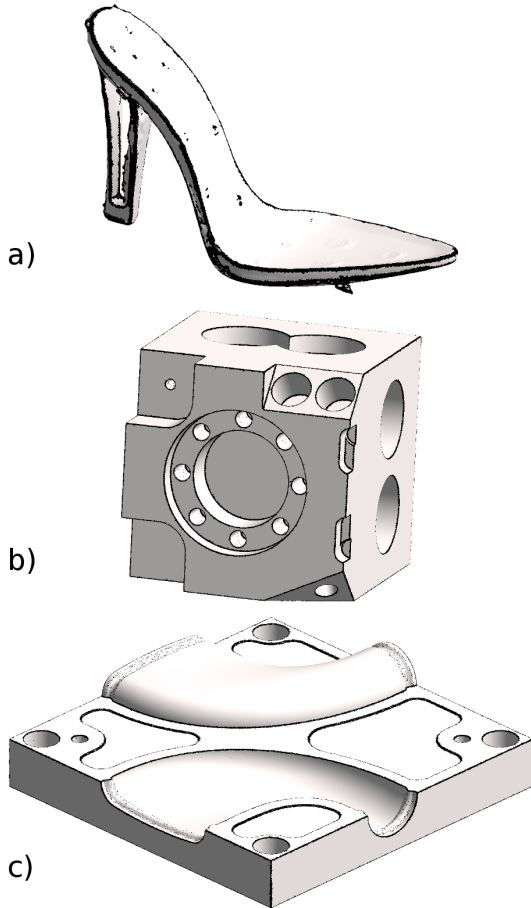


Figure 9 – Points arêtes détectés (illustrés en noir).

Ensuite, toutes les étapes de la segmentation ont été réalisées. Les résultats de cette segmentation sont illustrés figure 10 et détaillés dans le tableau 1. Chaque sous-maillage obtenu est colorié d'une couleur différente, et la partie noire restante du maillage correspond aux régions arêtes.

Maillage	a)	b)	c)
Régions avant fusion	207	224	592
Régions après fusion	12	71	32

Tableau 1 – Résultats de la segmentation avec fusion

Nous avons appliqué notre méthode sur 20 maillages très différents par la forme, la taille et la répartition des points, incluant ces trois exemples. Globalement, la segmentation obtenue correspond à une segmentation qu'un expert aurait pu faire manuellement.

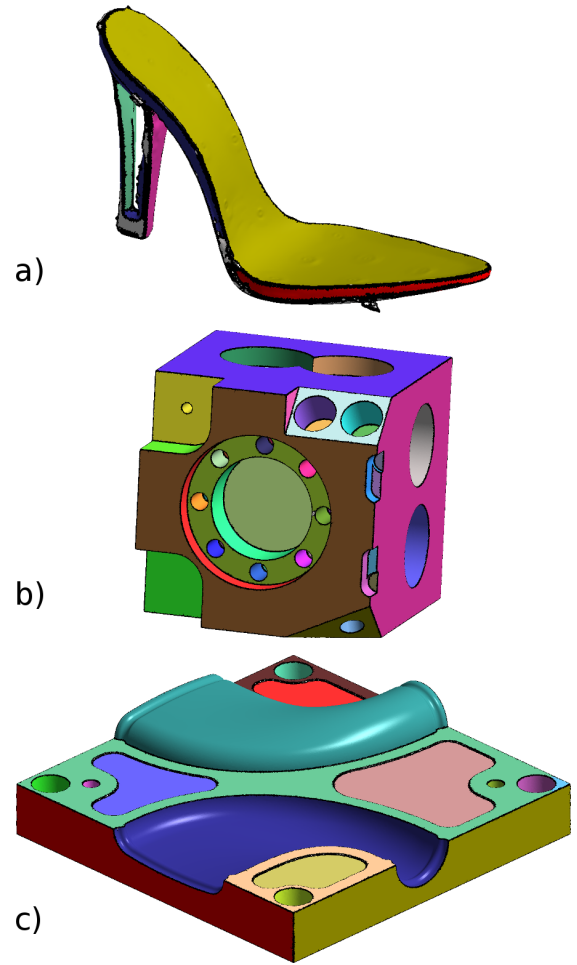


Figure 10 – Segmentation des maillages tests.

3.2 Application à la rétro-conception

Notre segmentation a été intégrée dans la solution de rétro-conception proposée par la société C4W.

Les résultats de l'extraction de primitives sur le maillage *Mold*, sans et avec la méthode de segmentation, sont illustrés figures 11.a et 11.b.

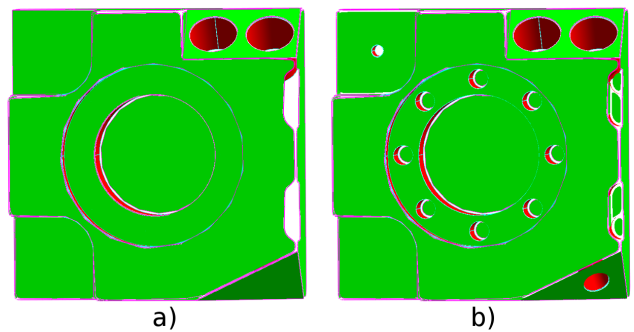


Figure 11 – Extraction de primitives, a) sans et b) avec segmentation préalable.

Nous avons alors pu vérifier expérimentalement notre hypothèse de départ, à savoir qu’une étape de segmentation permettait d’améliorer significativement l’extraction de primitives. De plus, nous avons observé un gain de temps non négligeable, se situant entre 40% et 75% sur la totalité de la solution, puisque la recherche des primitives se fait sur des régions plus cohérentes.

Des tests de performances ont été effectués, avec un processeur Intel® Core™ i5-4570 CPU @ 3.20GHz 3.20GHz.

Maillage	Triangles	Temps
a) Shoe	74 536	1s
b) Block	1 125 832	13s
c) Mold	851 194	17s

Tableau 2 – Temps de segmentation

Les résultats indiqués dans le tableau 2 montrent que la segmentation proposée s’exécute en quelques secondes. La méthode est donc adaptée à l’utilisation industrielle.

4 Conclusion et perspectives

Dans cet article, nous avons présenté une nouvelle méthode de segmentation des maillages numérisés. Aucun paramètre n’est demandé à l’utilisateur, ce qui est un avantage majeur dans le domaine industriel.

La méthode proposée se décompose en quatre parties, une détection des arêtes vives, une croissance de régions, ainsi que deux étapes de fusion de régions. Nous avons montré au travers de plusieurs résultats que notre méthode retournait des sous-maillages cohérents avec les maillages d’origine. De plus, après l’avoir intégrée dans le processus de rétro-conception de C4W, la segmentation s’est montrée très intéressante pour améliorer les résultats, que ce soit en terme de qualité ou de rapidité.

Un des plus gros avantages de notre méthode est de permettre l’extraction des arêtes du maillage. Ainsi, les sous-maillages sont plus homogènes, les temps de calculs sont plus courts et les calculs utilisés par la suite sont plus précis. De plus, la détection et l’extraction de ces arêtes apportent une information supplémentaire et nécessaire pour la reconstruction, à savoir la notion d’adjacence entre les sous-maillages et donc entre les primitives.

Notre méthode peut être améliorée, notamment au niveau du calcul des seuils. En effet, comme le propose [9], nous pouvons nous appuyer sur l’analyse de la distribution des valeurs de courbure sur un histogramme, illustré figure 12. Nous pouvons ainsi définir une meilleure valeur de seuil et donc améliorer nos résultats.

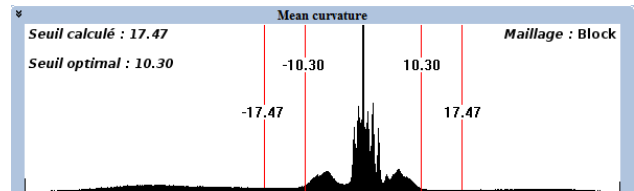


Figure 12 – Distribution de la courbure moyenne de Block.

Nous constatons que notre seuil est inférieur au seuil “optimal”, sélectionné manuellement par des experts. Une amélioration de notre segmentation utilisant cette distribution est en cours de développement.

Références

- [1] Pál Benkő, Ralph R. Martin, et Tamás Várady. Algorithms for reverse engineering boundary representation models. *Computer-Aided Design*, 33(11) :839 – 851, 2001.
- [2] R. Bénére, G. Subsol, G. Gesquière, F. Le Breton, et W. Puech. A comprehensive process of reverse engineering from 3d meshes to cad models. *Computer-Aided Design*, 45(11) :1382 – 1393, 2013.
- [3] G. Lavoué, F. Dupont, et A. Baskurt. A new CAD mesh segmentation method, based on curvature tensor analysis. *Computer-Aided Design*, 37 :975–987, 2004.
- [4] Ariel Shamir. A survey on mesh segmentation techniques. *Computer Graphics Forum*, 27(6) :1539–1556, 2008.
- [5] S. Petitjean. A Survey of Methods for Recovering Quadratics in Triangle Meshes. *ACM Computing Surveys*, 2(34) :1–61, july 2002.
- [6] L. Di Angelo et P. Di Stefano. Geometric segmentation of 3D scanned surfaces. *Computer-Aided Design*, 62 :44–56, 2015.
- [7] J. Digne, J.-M. Morel, N. Audfray, et C. Mehdi-Souzani. The level set tree on meshes. Dans *Proceedings of the Fifth International Symposium on. 3D Data Processing, Visualization and Transmission*, Paris, France, May 2010.
- [8] S. Delest, R. Boné, et H. Cardot. Etat de l’art de la segmentation de maillage 3D par patches surfaciques. *GTMG ’07 : Groupe de Travail en Modélisation Géométrique, Valenciennes : France*, mars 2007.
- [9] Jack Szu-Shen Chen et Hsi-Yung Feng. Automatic prismatic feature segmentation of scanning-derived meshes utilising mean curvature histograms. *Virtual and Physical Prototyping*, 9(1) :45–61, 2014.