

Table des matières

Liste des figures	5
Introduction générale	7
 Chapitre I : Modélisation d'objets 3d	
I. Introduction.....	8
II. Différents modes de modélisation.....	8
II.1.Modèles frontière.....	8
II.1.1.Fil de fer.....	9
II.1.2. Modèles à partir de surfaces frontière (Brep).....	10
II.2. Modèles volumiques.....	11
II.2.1. Modèles de composition (constructifs)	11
II.2.2. Modèle de décomposition (descriptifs).....	12
II.3. Modèles par balayage (sweeping).....	14
II.3.1. Balayages translationnels	15
II.3.2 Balayages rotationnels	15
II.3.3. Balayages coniques	16
II.4. Modèle fractale	16
II.4.1 Courbes fractales.....	16
II.4.2 Surfaces fractales	17
II.5. Système de particules	19

II.6. Modèles polygonaux	19
II.7. Courbes et surfaces paramétriques	20
II.7.1. Courbes de Bézier.....	20
II.7.2. Courbes et surfaces B-splines.....	21
II.7. 3. Courbes et surfaces NURBS	22
II.8. Modélisation par subdivision de surface	22
III. Conclusion	23

Chapitre II : Maillage triangulaire

I. Introduction	24
II. Maillage triangulaire	24
II.1 Méthodes de génération des maillages de surfaces 3D	24
II.1.1 Méthode de Hoppe	25
II.1.2. Approches basées sur le diagramme de Voronoi	26
II.2. Adaptation de maillages	27
II.2.1. Raffinement	27
II.2.2. Déplacements de nœuds	28
II.2.3. Simplification de maillage	28
III. Format d'objets 3D	28
III .1.les différents formats des objets 3D	29
III.1.1. DXF (Drawing eXchange Format)	29
III.1.2. Open Inventor	29
III.1.3. Le Virtual Reality Modeling Language (VRML):.....	30

III.1.4. 3D Studio Max (ou 3ds max).....	31
III.1.5. OBJ.....	31
III.1.6. Blender	31
III.1.7. Initial Graphics Exchange Specification (IGES).....	32
III.1.8. VTK.....	32
IV. Conclusion	33

Chapitre III : Méthodes de segmentation d'objets 3D

I. Introduction	34
II. Définition de la segmentation	34
III. Méthodes de segmentation	35
III.1 Segmentation de maillages triangulaires et rectification directionnelle de frontières basées sur les tenseurs de courbure.....	35
III.1.1 Algorithme générale	35
III.2 Ligne de Partage des Eaux (LPE)	37
III.2.1 Adaptation à la 3D	38
III.2.2 Sur-segmentation	39
III.2.3 Critère de courbure.....	40
III.2.4. Génération des marqueurs.....	41
III.3 Croissance de régions.....	43
III.3.1 Points de départ (seeds).....	43
III.3.2 Croissance (growing)	43
III.4 Système Multi-Agents	44

IV. Conclusion	45
-----------------------------	-----------

Chapitre IV : Implémentation

I. Introduction.....	46
-----------------------------	-----------

II. Choix techniques	46
-----------------------------------	-----------

II.1 Le langage de programmation	46
--	----

II.2 L'environnement de développement	46
---	----

III. Format des données d'entrée	46
---	-----------

IV. Fonctionnement du logiciel	47
---	-----------

IV.1. Présentation	47
--------------------------	----

IV.2. Expérimentation	53
-----------------------------	----

V. Conclusion	54
----------------------------	-----------

Conclusion générale	55
----------------------------------	-----------

Références bibliographiques	56
--	-----------

Annexe	58
---------------------	-----------

Liste des figures

Figure I.1 : Plusieurs interprétations d'un même objet	9
Figure I.2 : Modèle impossible.....	10
Figure I.3 le modèle Brep.....	10
Figure I.4: Plusieurs bâtiments définis avec primitive instancing.....	12
Figure I.5 : L'octree.....	14
Figure I.6 : Balayage translationnel.....	15
Figure I.7 : Balayage rotationnel.....	15
Figure I.8 : Balayage conique.....	16
Figure I.9 : Le flocon de Von Koch.....	17
Figure I.10 : Exemple de courbe fractale.....	17
Figure I.11 : Montagnes fractales.....	18
Figure I.12 : Le motif utilisé pour la modélisation du ciel.....	18
Figure I.13 : La surface générée pour réaliser le ciel et l'image résultat.....	18
Figure I.14 : Exemple de modèle polygonal.....	19
Figure I.15 : Une courbe de Bézier cubique avec les points de contrôle	21
Figure II.1 : Exemple de reconstruction par la méthode de Hoppe	26
Figure II.2 : Diagramme de Voronoi et sa triangulation de Delaunay.....	27
Figure II.3 : Exemple d'images 3D enVRML	30
Figure II.4: Exemple d'image 3D en format VTK	33
Figure III .1 : Les deux étapes de l'algorithme : la segmentation en régions et la rectification de frontières.....	36
Figure III.2 : Un sommet et les triangles entourant ce dernier	37
Figure III.3 : Minimum et plateaux.....	38

Figure III.4 : Les deux approches de la ligne de partage des eaux.....	38
Figure III.5 : Segmentation hiérarchique à partir de la ligne de partage des eaux.....	39
Figure III.6 : Marqueurs et résultat de la LPE.....	40
Figure III.7 : Structure de stockage des voxels.....	41

Résumé

Le développement continu des techniques multimédias et de la réalité virtuelle a suscité un intérêt croissant pour l'utilisation de contenus tridimensionnels (3D), et afin de faciliter le traitement et le transfert de ces objets, de nombreux algorithmes de décomposition ont été proposés. La décomposition de maillage triangulaire correspond au découpage du maillage en régions, c'est un outil nécessaire à de nombreuses applications citons la compression et la simplification. Nous avons présenter dans ce travail une méthode de décomposition de mailles triangulaires en se basant sur la critère de la courbure discrète, qui permet de déterminer la forme de l'objet (un plan, une sphère), et utilise tous ce que peut contenir une maille triangulaire, en occurrence les vertex et les triangles.

Introduction Générale

Introduction générale

Le domaine de l'informatique est en constante évolution, et a connu des avancées technologiques importants ces quinze dernières années.

Le matériel informatique et les données numériques ont évolués parallèlement, faisant apparaitre de nouveaux besoins et de nouvelles applications.

Les données tridimensionnelles 3D, constituent actuellement un contenu multimédia émergeant.

La représentation tridimensionnelle d'objets est utilisée dans des applications de plus en plus nombreuses, notamment la médecine, la conception assistée par ordinateur, les jeux et le cinéma.

Nous traitons dans ce mémoire la décomposition de maillage triangulaire, nous présentons dans le prochain chapitre les techniques de modélisation d'objets 3D.

Nous discutons au chapitre deux du maillage triangulaire, ainsi que des méthodes de génération et d'adaptation de ce dernier.

Au chapitre trois nous abordons plusieurs méthodes de segmentation d'objets 3D tel que la ligne de partage des eaux, la croissance de régions et la segmentation par système multi-agents qui est une méthode apparue récemment.

Nous terminons ce document par une conclusion et des perspectives.

Chapitre 1:
Modélisation d'objets 3D

I. Introduction

Parmi les techniques infographiques¹ il y a la modélisation 3D qui vise à créer des objets en trois dimension, à partir ou non d'objets réels.

La modélisation 3D est une partie récente de l'informatique et qui ne cesse de se développer grâce aux progrès scientifiques dont l'informatique bénéficie, elle repose essentiellement sur des lois mathématiques complexes, qui sont appliquées à des contextes précis de la physique (et en particulier à l'optique, qui est une partie de la physique prédominante dans la modélisation 3D). [7]

Un modèle d'un objet est une représentation simplifiée d'un objet qui permet de l'observer plus facilement. La construction d'un modèle pour représenter la structure géométrique d'un objet est intéressante à plusieurs points de vue: certaines caractéristiques du modèle peuvent être étudiées plus facilement que celles de l'objet lui-même; l'objet peut ne pas exister; il ne peut être observé directement; comme il ne peut être observé sans engendrer des coûts déraisonnables ou sans contrôle de l'expérience.

II. Différents modes de modélisation

La modélisation d'objets 3D peut être traitée par de nombreuses méthodes, chacune d'entre elles étant bien adaptée à un domaine d'application bien précis.

Il existe deux principaux types de modèles, on a les modèles par frontières et modèles volumiques.

II.1.Modèles frontières

Comme leur nom l'indique les modèles frontières représentent les frontières des objets.

Il existe deux types de modèles frontières : le fil de fer et le Boundary Representation (BR ou Brep).

II.1.1.Fil de fer

Ce type de modèle permet de représenter uniquement le contour des objets. Les objets sont représentés à partir d'un ensemble de segments de droite reliés éventuellement par leurs extrémités.

¹ Outils pour l'acquisition, la création, la transmission d'objets 3D.

Ce modèle est simple, il facilite les transformations de base et de visualisation. Permet à peu près d'avoir une représentation géométrique globale de l'objet. Cependant sa capacité de modélisation est très limitée (faible degré de réalisme), et le calcul de certaines mesures de l'objet peut être difficile ou impossible par manque d'informations sur l'objet. [1]

Il peut y avoir ambiguïté-dans l'interprétation de la représentation ou donner lieu à un modèle impossible.

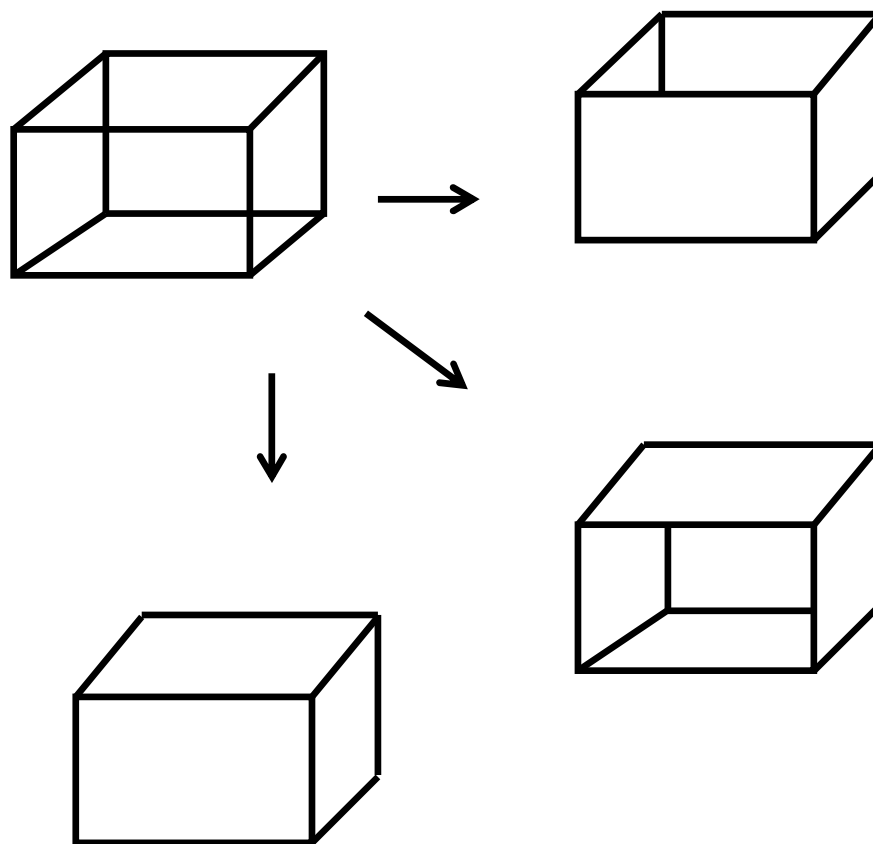


Figure I.1 : plusieurs interprétations d'un même objet.

Bien que toujours utilisé à des fins architecturales, pour des objets simples, il présente le problème de poser des ambiguïtés en représentation 3D. [6]

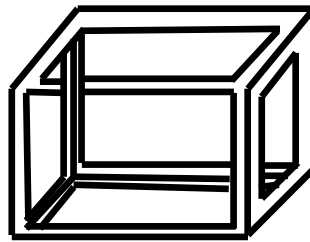


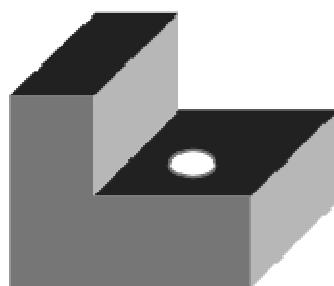
Figure I.2 : Modèle impossible.

II.1.2. Modèles à partir de surfaces frontières (Brep)

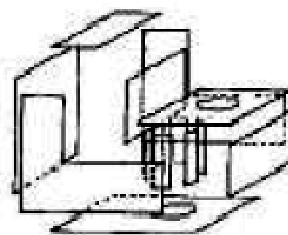
Ce modèle est une extension du modèle Fil de fer, il a été introduit afin de régler les problèmes d'ambiguïté dans le modèle filaire en introduisant la notion de face.

Il ne fournit pas d'information quant à l'intérieur de l'objet. L'ensemble des facettes forme une figure fermée. Ces facettes ne s'interceptent pas sauf à des arêtes ou sommets communs. Le nombre de facettes peut être élevé.

Le modèle Brep permet d'effectuer facilement une visualisation du solide et les opérations complexes tel que le calcul du volume d'un solide, appartenance d'un point au Solide. Cependant il ne fournit qu'une approximation de la surface du solide, et la construction de ce dernier est longue et fastidieuse car il faut Construire et agencer manuellement tous les éléments : Faces, Arêtes, Sommets, en plus du risque d'erreur topologique (ex. : oubli d'une arête) et la difficulté de revenir en arrière dans la conception du solide sans oublier une grande quantité d'informations à stocker.



Objet à modéliser



Brep correspondant

Figure I.3 : le modèle Brep.

II.2. Modèles volumiques

Il existe deux types de modèles volumiques:

- Les modèles de composition.
- Les modèles de décomposition.

II.2.1. Modèles de composition (constructifs)

Ils considèrent les solides comme des ensembles de points 3D.

Débutent avec des ensembles simples qui peuvent être représentés directement à l'aide de primitives (quadriques, polyèdres, etc.). Des objets plus complexes sont obtenus en combinant des ensembles simples entre eux à l'aide d'opérations ensemblistes.

■ Modèles CSG (*constructive solid geometry*)

Il s'agit de manipuler des objets élémentaires bornés : polyèdre, cylindre, sphère, etc.

Pour ce qui est des primitives non élémentaires, l'utilisateur doit spécifier uniquement le système de référence de l'objet. On met à la disposition de l'utilisateur un nombre fini de primitives simples dont la grandeur, la forme, la position et l'orientation sont définies par ce dernier.

L'opération de modélisation est simplifiée; l'animateur doit spécifier seulement l'arbre de construction. Aucune ambiguïté dans le modèle; cela donne lieu à des objets valides.

Cependant il est incomplet car il nécessite des algorithmes pour évaluer l'arbre de construction. Lorsque l'arbre CSG est mal balancé, les algorithmes proposés sont inefficaces. Il est difficile de construire un modèle CSG pour décrire des objets complexes n'intégrant aucune propriété géométrique, par exemple : la représentation d'un visage humain, L'évaluation de l'arbre de construction peut exiger des temps de calculs importants.

■ *primitive instancing*

La modélisation appelée « primitive instancing » utilise des primitives complexes paramétrables.

C'est une modélisation qui peut être employée dans des domaines bien précis où les objets peuvent se regrouper par type. Dans ce cas, c'est une alternative très intéressante

au CSG par exemple la figure ci-dessous présente plusieurs bâtiments définis par leur longueur, le nombre de cheminées et de fenêtres.



Figure I.4 : plusieurs bâtiments définis avec primitive instancing.

II.2.2. Modèle de décomposition (descriptifs)

Ces modèles décrivent l'espace utilisé au moyen d'un ensemble de volumes contigus et sans intersection.

C'est un mode de représentation de solides par partitionnement spatial où chaque solide est décomposé en un ensemble de solides adjacents, sans intersection, qui sont plus primitifs que le solide original, bien que nécessairement du même type.

Les objets sont représentés comme un ensemble d'objets ou cellules élémentaires : nous en distinguons deux méthodes : l'énumération exhaustive des cellules élémentaires, et la subdivision irrégulière d'une région dont l'objet à modéliser fait partie.

■ Énumération exhaustive des cellules élémentaires

- *Model Voxel*

Les voxels² sont des cellules spatiales occupées par le solide : en général des cubes de taille fixe repartis selon une grille, le solide est défini par la liste des coordonnées des centres des cellules.

Chaque cube peut être défini uniquement à partir d'un de ses sommets. En associant à chaque objet une région régulière dont l'objet fait partie, l'objet est représenté à partir d'un tableau binaire 3D.

Un élément du tableau $\equiv$ $\begin{cases} 1 \Leftrightarrow \text{le cube associé représente une sous-région du solide,} \\ 0 \Leftrightarrow \text{le cube ne fait pas partie du solide.} \end{cases}$

² Volumetric pixel : désigne les pixels d'images 3D.

Le modèle Voxel permet de construire facilement et efficacement de nouveaux objets à l'aide d'opérations ensemblistes en se basant sur ces tableaux binaires, et d'évaluer efficacement différentes mesures de l'objet.

Ce modèle est une représentation approximative de l'objet à moins que celui-ci ne coïncide exactement avec la grille ce qui est rarement le cas. On peut réduire la taille des voxels afin d'augmenter la précision de la représentation du solide.

Cela exige une quantité d'espace mémoire imposante. Cette approche ne peut être exploitée directement : pour construire un objet, on ne peut exiger l'énumération de l'ensemble de ces cellules.

■ Subdivision irrégulière d'une région dont l'objet à modéliser fait partie

Les techniques d'énumération spatiale sont simples, générales et permettent de développer une grande variété d'algorithmes. Mais elles exigent une grande quantité d'espaces mémoires et une représentation très approximative des solides.

Les cubes élémentaires voisins d'un cube faisant partie de l'objet ont de bonnes chances d'en faire partie aussi. Pour remédier à cet inconvénient, nous nous basons sur le principe suivant:

Le nombre de cubes nécessaires pour représenter un objet devrait dépendre de la surface de l'objet et non du volume de l'objet.

La principale méthode basée sur ce principe est l'Octree qui est une généralisation des Quadrees.

- **QUADTREE**

Un quadtree est une représentation d'un objet 2D basée sur la subdivision récursive d'un carré en quatre carrés de même taille (quadrants). A chaque stade du processus récursif, deux cas sont possibles :

- ❖ L'objet ne recouvre pas complètement le quadrant; il est alors subdivisé en quatre nouveaux carrés
- ❖ Le quadrant est plein ou vide; la subdivision est alors stoppée et le quadrant est marqué comme "plein" (1) ou "vide" (0).

- **OCTREE**

L'Octree est donc une extension du quadtree à trois dimensions pour encoder des solides.

Une région cubique est récursivement subdivisée en huit octants (régions cubiques).

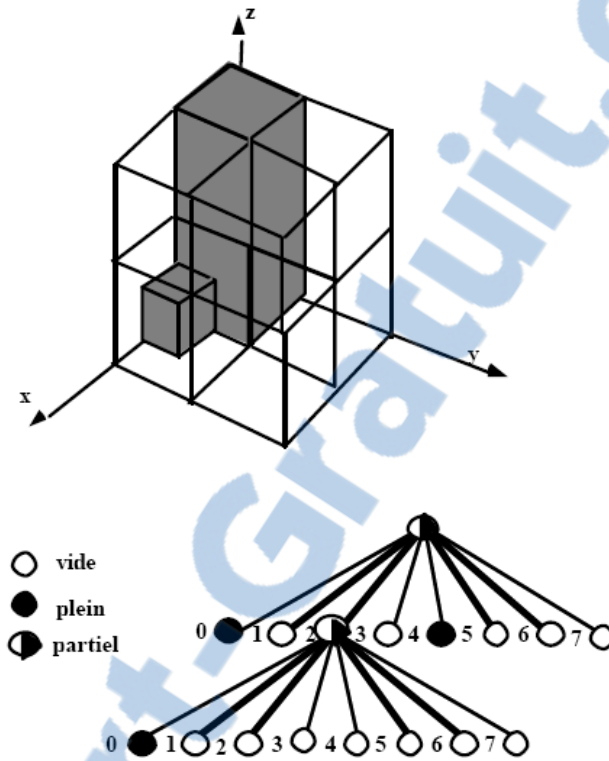


Figure I.5 : L'Octree.

Dans les sections précédentes nous avons présenté les principaux modes de modélisation, cependant il existe d'autres que nous allons voir dans ce qui suit.

II.3.Modèles par balayage (sweeping)

Ces représentations sont basées sur le déplacement d'une région (génératrice) le long d'une courbe (directrice). On distingue trois principales représentations de ce type: les surfaces par balayage translationnel, les surfaces par balayage rotationnel et les surfaces par balayage conique.

II.3.1. Balayages translationnels

Ils sont à l'origine des formations de cylindres. Un objet défini par balayage translationnel s'obtient en translatant une région 2D « C » le long d'un vecteur « T » et selon une distance « d », comme le montre la Figure I.6. Le plan « P0 » de la courbe est appelé le plan de base et le plan « P1 » à une distance « d » est le plan du sommet.

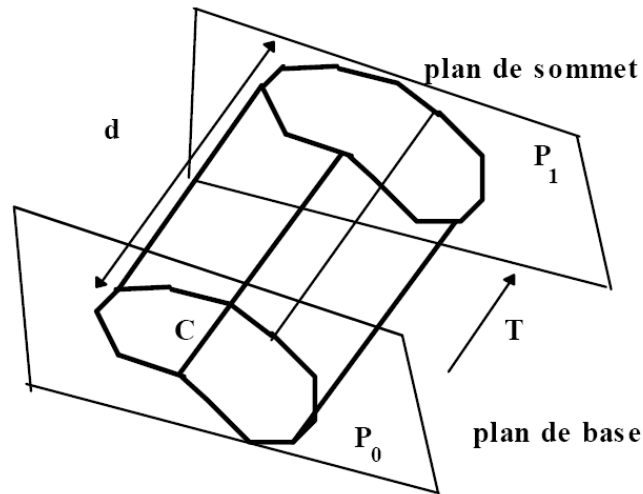


Figure I.6: Balayage translationnel.

II.3.2 Balayages rotationnels

Les objets générés par balayage rotationnel sont obtenus en faisant tourner une région 2D « C » autour d'un axe « A ». Deux éléments sont importants dans une telle surface: le point de base « B » et la fonction radiale « r(s) ».

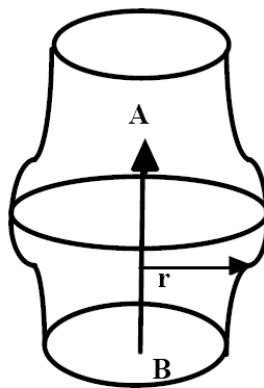


Figure I.7 : Balayage rotationnel.

II.3.3. Balayages coniques

Ces balayages à l'origine des cônes consistent en un déplacement d'une région 2D « C » avec transformation d'échelle.

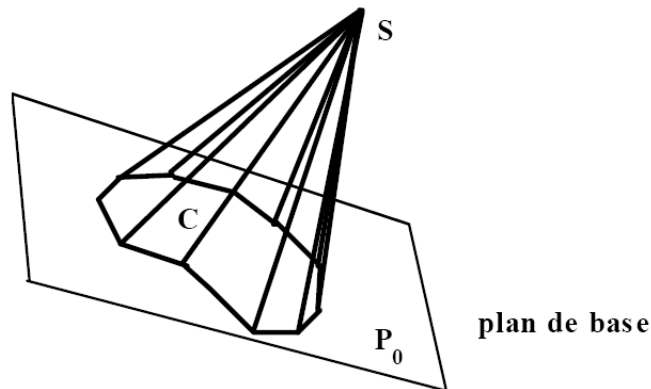


Figure I.8 : Balayage conique.

II.4. Modèle fractale

Pour modéliser des environnements naturels tels que les montagnes, les feuillages ou un ciel nuageux, il est nécessaire de disposer de modèles autres que ceux présentés précédemment. Ces derniers sont, en effet, inefficaces pour représenter des motifs répétés un grand nombre de fois comme c'est le cas pour les décors cités.

Les modèles fractals apportent une solution efficace à ce problème. Le terme fractal, issu du latin « frangere » et originalement proposé par MandelBrot [1982], est utilisé par la communauté graphique pour décrire tout modèle irrégulier et fragmenté pour lequel les irrégularités sont conservées (on parle d'auto-similarité). Il s'agit généralement de fonctions récursives utilisant un motif initial et un motif de remplacement.

Un objet est dit fractal lorsque le processus de remplacement est appliqué à l'infini pour la création de l'objet. [3]

II.4.1 Courbes fractales

Pour générer une courbe fractale, on remplace à l'étape « i » les segments du motif de l'étape (i-1) par le motif de remplacement. Lorsque le processus est répété à l'infini, la



courbe est dite auto-similaire : la courbe entière est similaire à une sous-partie d'elle-même.



Figure I.9: Le flocon de Von Koch.

La dimension fractale d de la courbe est définie par : $n^{1/d} = f$,

Où n est le nombre de parties du motif ($n = 4$ ici) et f est le facteur de diminution du motif ($f = 3$ ici).

La dimension est donc : $d = \log 4 / \log 3 = 1,26$.

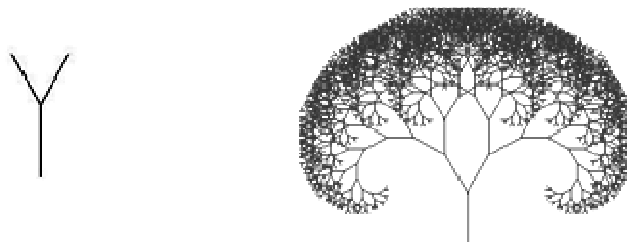


Figure I.10: Autre exemple de courbe fractale.

II.4.2 Surfaces fractales

Les surfaces sont générées de manières similaires aux courbes. Par exemple, l'algorithme de Fournier-Russel-Carpenter permet de générer des montagnes fractales. Le principe est, à partir d'un triangle initial, de subdiviser chaque segment du triangle en deux puis de modifier la hauteur du point milieu de chaque segment de manière aléatoire (voir figure I.10).

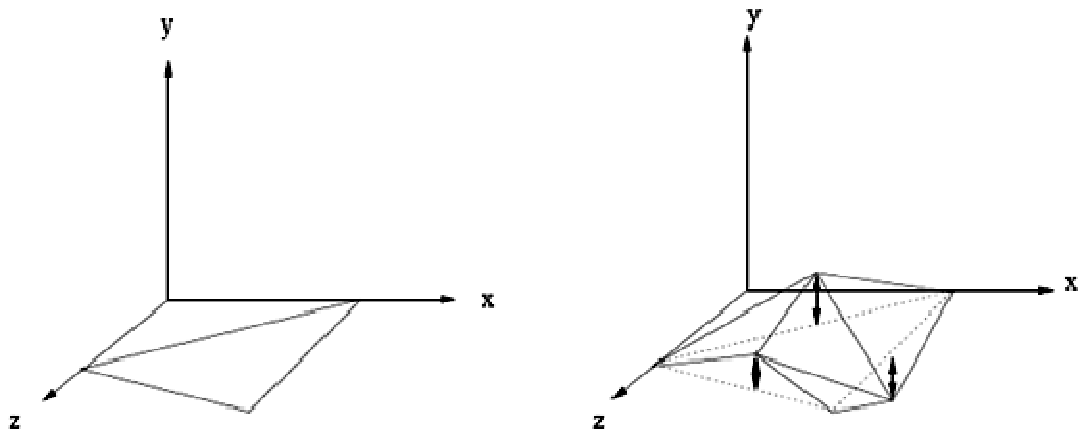


Figure I.11 : Montagnes fractales : chaque segment est subdivisé en deux et la hauteur du point milieu de chaque segment est modifiée de manière aléatoire.

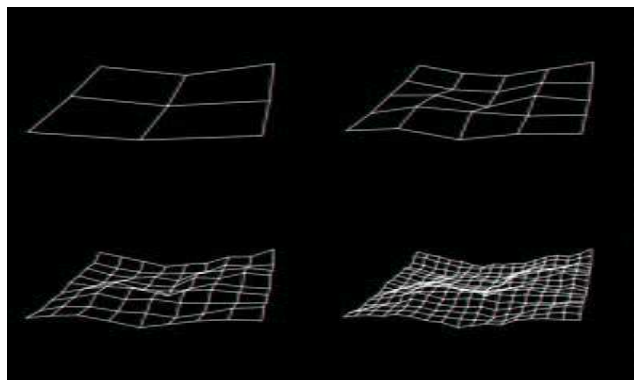


Figure I.12: Le motif utilisé pour la modélisation du ciel.

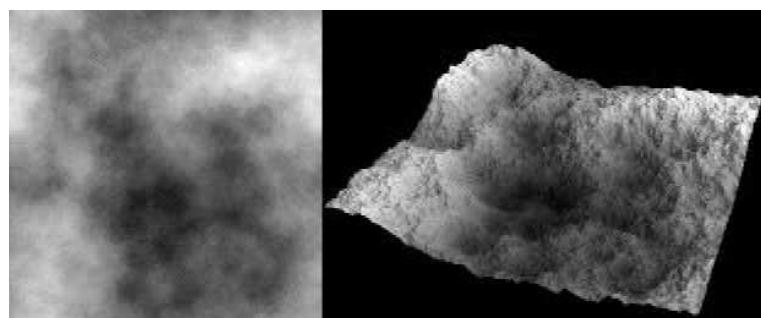


Figure I.13: La surface générée pour réaliser le ciel et l'image résultat.

II.5. Système de particules

Ils permettent de modéliser des objets flous (fuzzy) tels que feu, nuage, eau, . . . Ce type de modélisation prend naturellement en compte les contraintes d'animation bien qu'il décrive parfaitement les modèles purement statiques.

Un objet est représenté par un ensemble de particules, chacune d'elles naissant, évoluant dans l'espace et le temps et disparaissant. Des attributs tels que la couleur, la transparence, la vitesse et la taille sont fonctions du temps. [5]

II.6. Modèles polygonaux

Le modèle est assimilé à un ensemble de polygones : ce polyèdre³ est donc décrit par la liste des sommets et des arêtes. Si les polygones sont orientés, on pourra différencier l'extérieur et l'intérieur du modèle. Si l'on veut lever l'ambiguïté apportée par des polygones non plans, on utilisera uniquement des triangles.

Ces représentations sont extrêmement répandues en raison de leurs simplicités.

Dans le cas de surface, on parle de maillage polygonale : une collection de sommets, côtés et polygone telle qu'un côté est partagé par deux polygones au plus. Le problème qui se pose au programmeur ici est de choisir une représentation adéquate, pour le stockage externe (dans un fichier) et interne (en mémoire). [3]

Quelles informations stockées et comment ?

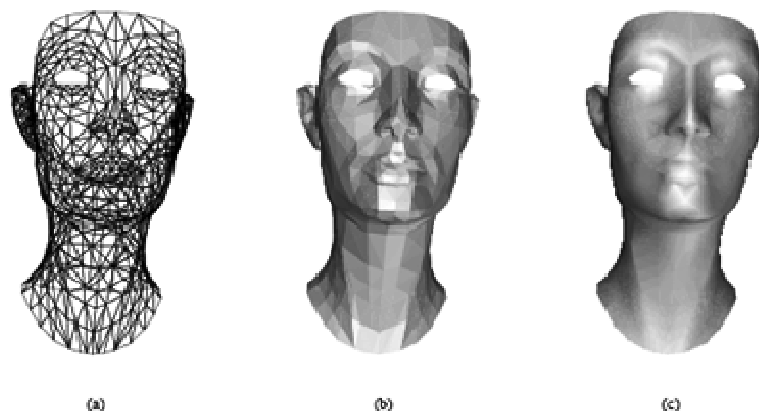


Figure I.14 : Exemple de modèle polygonal (facettes triangulaires) :
(b) ombrage plat, (c) ombrage de Gouraud.

³ Un polyèdre est une forme géométrique à trois dimensions ayant des faces planes polygonales qui se rencontrent selon des segments de droite qu'on appelle arêtes.

Plusieurs représentations sont possibles, le choix se fait en fonction des besoins et considérant les coûts en espace (représentations externes et internes) et en temps (représentations internes).

Les opérations typiques sur les maillages sont :

- trouver les côtés incidents à un sommet.
- trouver les voisins d'un sommet.
- trouver les polygones partageant un côté ou un sommet.
- trouver les côtés d'un polygone.

L'idéal consiste à stocker en interne le maximum d'informations pour éviter des calculs répétitifs, mais cela nécessite un espace mémoire important.

II.7. Courbes et surfaces paramétriques

Les approximations polygonales sont des approximations linéaires par morceaux (1er degré). Elles nécessitent un nombre de points importants pour obtenir une bonne précision. Une autre approche consiste donc à utiliser des modèles d'ordre supérieurs pour représenter les courbes et surfaces. Les représentations paramétriques répondent à ce besoin et apportent un gain en espace mémoire ainsi qu'en possibilité de manipulations.

On utilise en général des courbes de degré 3 (cubiques). Dans le contexte du traitement ou de la synthèse d'images, nous cherchons à interpoler ou approximer une courbe à partir d'un nombre fini de points.

On utilise pour cela les courbes de la forme : $P(t) = (x(t), y(t), z(t)) = \sum_{i=0}^n f_i(t) P_i$.

$x_i=0$

Où $f_i(t)$ sont des fonctions de pondérations, $p_i = (x_i; y_i; z_i)$ sont les points de données.[3]

II.7.1. Courbes de Bézier

Du nom d'un ingénieur de Renault (Pierre Bézier) qui développa ce modèle de courbes pour la conception de carrosseries de voitures.

Une telle courbe commence par le premier, finit par le dernier point et ne passe pas obligatoirement par les autres points, elle n'est pas déformée par translation et rotation, mais par changement des paramètres de mise en perspective.[14]

Quand on modifie la position d'un seul point de contrôle (sommet), la courbe est intégralement modifiée ce qui entraîne sa réévaluation intégrale. Il n'y a donc pas de contrôle local sur la forme de la courbe.

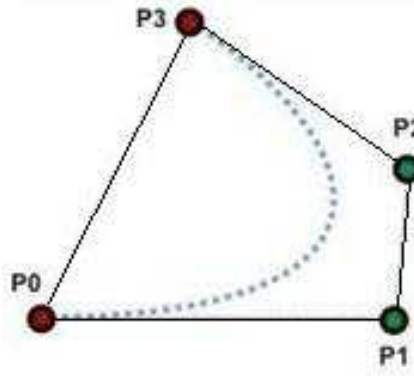


Figure I.15: Une courbe de Bézier cubique avec les points de contrôle $P_0, \dots, P_3$.

II.7.2. Courbes et surfaces B-splines

Les courbes et les surfaces spline sont des modèles mathématiques qui permettent d'associer une représentation continue (courbe ou surface) à un ensemble discret de points d'un espace affine (habituellement $\mathbb{R}^3$). [4]

▣ Courbes B-splines

Les B-splines cubiques (splines = lattes de jardinier) sont des courbes polynomiales cubiques de continuité C^2 . Elles approximent un ensemble de points de contrôle P_i , $i \in [0;m]$ avec une courbe constituée de $m-2$ segments de courbes polynomiales Q_i , $i \in [3;m]$. Chaque segment de courbes est en fonction d'un paramètre t variant de t_i à t_{i+1} et de 4 points de contrôle. [3]

Le segment Q_i est donc défini par : $p_{i-3}, p_{i-2}, p_{i-1}, p_i$ pour t variant de t_i à t_{i+1} .

➤ Un point de contrôle affecte 4 segments de courbes.

On distingue deux types de courbes B-splines :

- Les B-splines uniformes.
- Les B-splines non-uniformes.

▣ Surfaces B-splines

Surfaces remédiant en grande partie aux défauts des surfaces de Bézier.

Les propriétés des surfaces splines découlent des propriétés des courbes splines.

Parmi les avantages surfaces B-splines on cite:

- Control local la modification d'un point de contrôle ne change pas toute la courbe.
- « K » et « l » contrôlent l'ordre des B-splines, donc pour la même grille de points de contrôle, on peut obtenir une surface qui adhère plus ou moins à la grille en faisant varier « k » et « l ».

Cependant lorsqu'on augmente « k » ou « l », on augmente aussi considérablement la complexité des calculs, k et l sont entiers et empêchent toute variation continue. [2]

II.7. 3. Courbes et surfaces NURBS

Les NURBS sont utilisées pour représenter mathématiquement des objets géométriques. Elles généralisent la représentation par les B-splines des courbes et des surfaces en ajoutant un dénominateur.

Une B-spline ressemble à une représentation polynomiale par morceaux, alors qu'une NURBS est une représentation par fractions rationnelles par morceaux.

Elles présentent de nombreux avantages comme la facilité et précision pour évaluer une forme, la capacité pour approximer des formes complexes et la simplicité de construction et d'implémentation. [8]

II.8. Modélisation par subdivision de surface

Cette méthode regroupe un peu des deux méthodes classiques (polygonale et NURBS). Elle consiste à accélérer le processus grâce à la subdivision automatique d'une partie de la surface. Ceci permet d'ajouter des détails à certains endroits uniquement, sans se soucier du nombre de faces comprises sur la globalité de l'objet.

Elle se rapproche de la modélisation polygonale par les techniques employées lors de la création de la forme, et de la modélisation par NURBS en ce qui concerne le rendu de la surface, c'est-à-dire son arrondi.

III. Conclusion

Dans ce chapitre nous avons vu les différentes techniques de modélisation des objets 3D, la modélisation surfacique (modèles par balayage) qui offre une représentation plus ou moins complexe d'un objet et qui est très adaptée à la conception de formes, la modélisation filaire qui permet une description simple de la forme mais n'apporte pas d'information de surface, et la modélisation volumique qui représente un objet comme un volume avec des informations sur l'intérieur de la forme.

Chapitre 2:
Maillage triangulaire

I. Introduction

Les maillages 3D sont composés d'un ensemble de faces comprenant des sommets (vertex) ainsi que des arêtes. Le type de face est variable et les maillages triangulaires, quadrangulaires ou simplexes sont assez courants. Les maillages triangulaires sont cependant les plus répandus car ils représentent une solution privilégiée pour l'industrie de par leur simplicité et leur flexibilité.

II. Maillage triangulaire

La représentation la plus simple d'une forme est de décrire les points caractérisant cette forme. Une représentation plus complète consiste à représenter les formes par maillage.

Un maillage est un ensemble d'éléments ou mailles dont la dimension dépend du domaine à mailler et de l'espace dans lequel se trouve ce domaine :

- Linéique lorsque le domaine est une courbe ou une droite ; il peut être plongé dans un espace de dimension un, deux ou trois. Les mailles sont des segments.
- Plan lorsque le domaine et l'espace sont en dimension deux ; les mailles sont des polygones.
- Surfactive lorsque le domaine est en dimension deux et l'espace en dimension trois ; les mailles sont des "polygones" non forcément planaires (par exemple les quadrangles).
- Volumique lorsque le domaine et l'espace sont en dimension trois : les mailles sont des polyèdres, très souvent des tétraèdre⁴ ou des hexaèdres⁵. [11]

II.1 Méthodes de génération des maillages de surfaces 3D

Les surfaces d'objets tridimensionnels ont un rôle très important dans les applications graphiques actuelles. Elles sont représentées par un maillage polyédrique.

La représentation de ces surfaces par des maillages permet de faire la compression, la visualisation et le transfert des objets 3D efficacement, pour aboutir à un tel maillage triangulaire plusieurs méthodes peuvent être utilisées.

⁴ Le tétraèdre est un polyèdre composé de quatre triangles

⁵ Un hexaèdre est un polyèdre à six faces

II.1.1 Méthode de Hoppe

A travers une série de trois articles, Hoppe et Al, développèrent la première méthode de reconstruction de surface arbitraire à partir d'un nuage de points non structuré.

Cette approche se résume en trois phases :

■ Reconstruction

Cette phase de reconstruction de maillage primaire à partir d'un nuage de points se découpe en deux étapes :

- Création d'une iso-surface approximant le nuage de points : pour cela un plan tangent est estimé pour chaque point selon son voisinage, et l'iso valeur calculée en tout point de l'espace est la distance signée au plan tangent du point du nuage le plus proche.
- Génération d'un maillage à partir de cette surface via une variante de l'algorithme de générations des maillages.

■ Optimisation du maillage

Cette phase consiste à réduire le nombre de triangles issus de la première phase ; en équilibrant le rapport entre une représentation minimale (en terme de complexité mémoire, et donc de facettes) et une bonne approximation du nuage de points en se basant sur les techniques de compression.

■ Génération d'une surface de subdivision

Cette dernière phase consiste à générer une surface de subdivision, minimisant l'énergie avec le maillage issu de la deuxième phase. Une modification du schéma de subdivision est proposée, permettant de diminuer localement l'ordre de continuité de la surface, et de représenter les arêtes vives.

Si l'angle entre deux faces adjacentes est inférieur à un seuil prédéfini, l'arête commune est marquée comme vive, et un masque spécial de subdivision sera appliqué. Cette approche constitue le point d'entrée de la reconstruction de surface à partir d'un nuage de point non structuré.

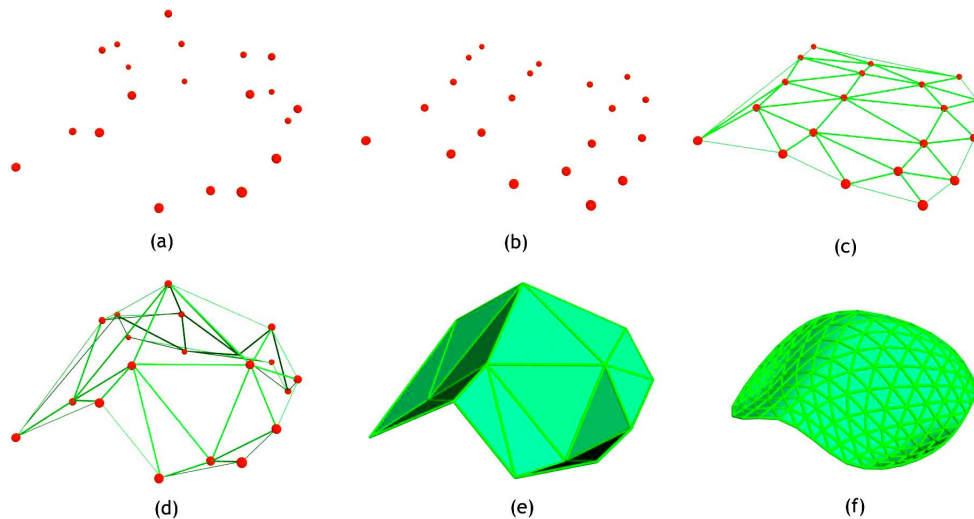


Figure II.1 : Exemple de reconstruction par la méthode de Hoppe. [5]

II.1.2. Approches basées sur le diagramme de Voronoï

Il existe toute une famille d'algorithmes générant des surfaces explicites (le plus souvent une collection de polygones structurée) à partir de nuages de points.

Afin de générer le maillage, la plupart de ces approches commencent par estimer la topologie de l'ensemble de points à l'aide de son diagramme de Voronoï.

Le diagramme de Voronoï est un partitionnement de l'espace en cellules. Soit p un point de l'ensemble P . La cellule associée à p représente l'ensemble des points de l'espace qui sont plus proches de p que de tout autre point de P .

Le diagramme de Voronoï est habituellement utilisé en deux ou trois dimensions. Il est la structure topologique d'un nuage de points, et notamment fournit la notion de voisinage d'un point : p est voisin d'un point q de P si et seulement si leur cellules de Voronoï sont adjacentes.

Une fois le diagramme de Voronoï établie, on peut construire son dual géométrique, la Triangulation de Delaunay, comme montré sur la figure II.2. Il s'agit d'un ensemble de triangles en 2D, ou de tétraèdre en 3D. [5]

Le noyau de Delaunay est une méthode permettant d'insérer un point dans une triangulation en modifiant localement la topologie de la triangulation existante tout en maintenant le critère de Delaunay. C'est pourquoi les méthodes utilisant le noyau de Delaunay sont qualifiées d'incrémentales.

Une triangulation de Delaunay a les propriétés suivantes :

- le maillage de Delaunay interpole l'ensemble des points de P ;
- une arête entre deux points de P existe si et seulement si leurs cellules de Voronoï sont adjacentes ;
- elle maximise l'angle minimal entre deux arêtes ayant un sommet en commun (bon équilibre du maillage) ;
- le cercle circonscrit d'un triangle de Delaunay ne contient aucun autre point de P.

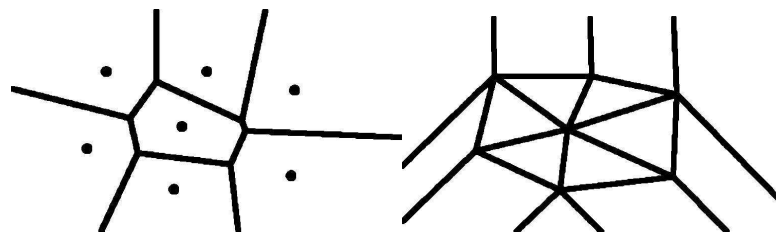


Figure II.2 : Diagramme de Voronoï et sa triangulation de Delaunay

II.2. Adaptation de maillages

Il arrive qu'on ne sache pas construire un maillage avec les propriétés optimales ou qu'on souhaite réutiliser un maillage dans un autre contexte que celui prévu au départ. Dans ces cas, une adaptation est appliquée au maillage existant de manière à améliorer ses propriétés.

Il peut s'agir d'affiner le maillage là où une concentration de nœuds donnée est nécessaire, de le simplifier, par exemple pour l'adapter au point de vue, ou de modifier la position des nœuds pour améliorer la forme des éléments.

II.2.1. Raffinement

Le raffinement de maillage n'est autre que l'ajout de nœuds dans le maillage de manière à rendre le traitement sur le maillage plus précis.

Pour les maillages triangulaires, les méthodes de raffinement les plus connues et les plus utilisées dans le domaine de la simulation exploitent le noyau de Delaunay.

II.2.2. Déplacements de nœuds

La construction et le raffinement d'un maillage nécessitent la plupart du temps un post traitement car les algorithmes produisent malgré tous des défauts que les méthodes d'optimisation réduisent considérablement en peu de temps.

Ces défauts proviennent le plus souvent de la difficulté à construire de nouveaux points de manière à satisfaire au mieux des propriétés parfois contradictoires sur le maillage. Pour améliorer le maillage, un déplacement des nœuds est alors appliqué de manière à rendre plus régulières la répartition des nœuds et, par suite, la forme des éléments.

La méthode la plus simple pour rendre la forme des éléments plus régulière, est de chercher à placer chaque sommet au barycentre de ses voisins. Chaque sommet est traité une fois, puis le processus recommence jusqu'à ce que le déplacement de chaque sommet soit inférieur à un seuil donné ou qu'un nombre d'itérations fixé à l'avance soit atteint. Le plus souvent, les points ne sont déplacés que d'une fraction de la distance entre la position actuelle et le barycentre.

Cette méthode dite du Laplacien, possède des inconvénients, dont celui de produire des éléments de mauvaise qualité voire des éléments inversés dans les régions concaves.

II.2.3. Simplification de maillage

Les travaux de simplification de maillage concernent majoritairement le domaine de la visualisation, où les scènes peuvent comporter de nombreux objets plus ou moins complexes.

Pour afficher rapidement une scène et permettre de s'y déplacer, le maillage est simplifié par réduction du nombre de triangles. Il existe différentes manières de simplifier un maillage. La plupart dégradent la précision d'approximation du maillage afin de réduire significativement le nombre de triangles.

Ces méthodes d'adaptation de maillage ont pour vocation de modifier un maillage existant pour qu'il satisfasse mieux un certain nombre de contraintes.

III. Format d'objets 3D

Pour représenter un objet 3D il faut au minimum une description :

- De la topologie de l'objet, sa forme, sa taille et sa complexité.

- Des attributs de représentation : couleurs, textures (nature et position), qualité photométrique de sa surface, transparence.
- De ses attributs dynamiques s'il est animé : capacité de collision avec d'autres objets, articulations et contraintes, etc.

Les premiers formats standard de fait ont été des formats adaptés à la CAO⁶ : l'objet est défini à l'aide de facettes ou de surfaces analytiques. Il suffit de définir son origine puis les coordonnées caractéristiques des éléments dans l'espace à trois dimensions.

Actuellement, dans le monde professionnel, il n'y a pas de format unique mais plutôt des formats plus ou moins utilisés selon le type d'application.

III .1.les différents formats des objets 3D

La plupart des modeleurs 3D savent plus ou moins bien lire (Import) et créer (Export) plusieurs formats : c'est un critère de choix important. On va citer dans ce qui suit les formats les plus répandus.

III.1.1. DXF (Drawing eXchange Format)

Il a été créé par la société Autodesk servant à échanger des fichiers DAO⁷ ou CAO entre systèmes CAO n'utilisant pas le même format de fichier natif. Il a été conçu à l'origine pour représenter les modèles 3D créés avec AutoCAD.

DXF est un format de fichier utilisé pour le transfert de données du type vecteur. Il contient de l'information pour la visualisation des données graphiques et est supporté par presque tous les logiciels graphiques. Il y a beaucoup de méthodes pour enregistrer les données des attributs des objets graphiques par le format DXF et aussi pour lier des objets DXF à des attributs externes. [17]

Dans ce format, un objet est une suite d'entités nommées et constituées de liste de points X, Y, Z. Par indexation, on constitue des facettes triangles ou des lignes qui s'appuient sur ces points.

III.1.2. Open Inventor

Si le format DXF était suffisant pour du dessin technique il était totalement inadapté à la réalité virtuelle. Dans les années 1990, la société Silicon Graphics (constructrice de

⁶ CAO : Conception Assistée par Ordinateur.

⁷ DAO : Dessin Assisté par Ordinateur, discipline permettant de produire des dessins techniques avec un logiciel informatique.

station graphique 3D) publia le format Inventor qui comportait la majorité des éléments nécessaires.

C'est une bibliothèque logicielle développée à l'origine par la société Silicon Graphics (IRIS Inventor) pour permettre de gérer des graphes de scènes en trois dimensions. Cette bibliothèque, orientée objet et écrite en C++, est capable de gérer son propre format natif de description des scènes : le format Inventor considéré comme étant l'ancêtre de VRML (leurs syntaxes respectives sont d'ailleurs similaires).

Ce format évolua vers le format VRML qui a été normalisé.

III.1.3. Le Virtual Reality Modeling Language (VRML):

Le Virtual Reality Modeling Language ou « Virtual Reality Markup Language » est un langage de description d'univers virtuels en trois dimensions. Les fichiers VRML ont habituellement pour extension « .wrl ».

En fait, c'est à proprement parler un langage de présentation et non de programmation, puisque comme pour le langage HTML par exemple, un fichier VRML ne contient généralement pas une suite d'instructions mais plutôt les informations permettant au visionneur d'afficher ensuite les éléments (formes, senseurs, lumières, etc.).

Présenté lors de la World Wide Web Conférence de 1994, VRML n'est pas l'œuvre d'un unique programmeur, mais plutôt le résultat de la collaboration de plusieurs professionnels de la 3D, dont entre autres Mark Pesce, Tony Parisi, Gavin Bell (Silicon Graphics) et Paul Strauss (Silicon Graphics).

Les programmes VRML peuvent décrire des formes simples (points, lignes, polygones) ou complexes (sphères, cubes, cônes, cylindres...), du texte, des images, des animations, des éclairages, des sons.

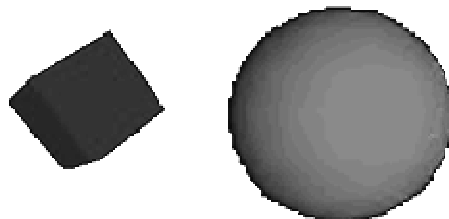


Figure II.3 Exemple d'images 3D enVRML

III.1.4. 3D Studio Max (ou 3ds max)

3ds max est un logiciel de modélisation et d'animation 3D, développé par la société Autodesk. Avec Maya, Softimage XSI, Lightwave, Houdini et Blender, il est l'un des logiciels de référence dans le domaine de l'infographie 3D.

Le logiciel est issu du programme 3D Studio qui a tourné sous DOS jusqu'à sa version 4. Les programmeurs de Kinetix⁸ ont développé un logiciel entièrement nouveau et repensé.

3dsmax est ainsi conçu sur une architecture modulaire et supporte des plug-ins (extensions), ainsi que les scripts écrits dans un langage propriétaire « maxscript ».

Le logiciel s'est développé rapidement, en étant utilisé principalement dans le cadre du jeu vidéo. 3D Studio Max est également utilisé dans un nombre croissant de films dont : X-Men II et Final Destination II.

Le logiciel est actuellement en version 2011 (3ds max 2011) et est développé par Autodesk (anciennement appelé Discreet, qui développe aussi Combustion et d'autres logiciels professionnels), et intègre de très nombreuses fonctionnalités, comme le moteur de rendu Mental Ray, ainsi que le plugin Shave and HairCut de Joe Alter pour les rendu de cheveux et de poil, nouveautés de la version 8 ou le moteur d'animation de personnage Character Studio, associé avec 3D Studio Max sous la forme d'un plugin. Gmax en est une version simplifiée et gratuite.

III.1.5. OBJ

C'est un format de fichier contenant la description d'une géométrie 3D. Il a été défini par la société Wavefront Technologies dans le cadre du développement de son logiciel d'animation « Advanced Visualizer ».

Ce format de fichier est ouvert et a été adopté par d'autres logiciels 3D (tels que Poser de e-frontier, Maya de Autodesk, Blender, MeshLab, 3D Studio Max, GLC Player etc.) pour des traitements d'import / export de données.

III.1.6. Blender

Blender, aussi appelé Blender3D, est un logiciel libre et gratuit d'animation, de modélisation et de rendu 3D. Il dispose de fonctions avancées de modélisation, de

⁸ Une division d'Autodesk rebaptisée maintenant Autodesk Media and Entertainment

sculpture 3D, de dépliage UV, de texturage, de gréage, d'armaturage, d'animation 3D, et de rendu. Il gère aussi l'édition non linéaire, les compositions, la création nodale de matériaux, la création d'applications 3D interactives, ainsi que diverses simulations physiques telles que les particules, les corps rigides, les corps souples et les fluides. Disponible dans de nombreuses langues, Blender est également disponible sur plusieurs plates-formes telles que Microsoft Windows, Mac OS X, GNU/Linux, IRIX, Solaris, FreeBSD, SkyOS, MorphOS et Pocket PC. De plus, c'est un programme extensible (ajout de scripts) à l'aide du langage Python. [15]

III.1.7. Initial Graphics Exchange Specification (IGES)

C'est un format d'exportation de données graphiques; IGES est directement importable dans les logiciels de CAO. Ce type de format d'export présente une universalité assez forte dans le domaine des CAO 3D orientées surfacique et filaire. L'export porte ici sur la définition mathématique de la surface qui conserve ainsi toutes ses propriétés initiales.

III.1.8. VTK

Ce sont des types de fichiers utilisés dans la librairie VTK (The Visualization Toolkit) c'est une librairie graphique gratuite, portable, orientée objet pour développer des logiciels d'infographie, de visualisation ou de traitement d'images.

Elle fournit une grande variété de structures de données codant des ensembles utiles en visualisation comme les ensembles de points.

De plus elle apporte une grande quantité de procédures de lecture/écriture en de très nombreux formats pour permettre l'importation et l'exportation des formats de codages usuels (PLOT3D, IV, VRML, ...). Afin de traiter efficacement ces données, des filtres sont disponibles agissant sur la géométrie ou les données numériques des maillages. Cette partie de la librairie est particulièrement bien fournie apportant un choix considérable et surtout efficace de méthodes.



Figure II.4: Exemple d'image 3D en format VTK.

Actuellement VTK est distribuée comme une surcouche d'OpenGL pour se plier au standard et assurer une grande portabilité. [16]

Conclusion

Dans ce chapitre nous avons présenté deux méthodes qui permettent la reconstruction d'un maillage à partir d'un nuage de points dispersés, et les méthodes d'adaptation d'un maillage.

Ensuite nous avons fait un tour sur les différents formats d'objets tridimensionnels. Il existe une palette impressionnante des types de fichier concernés par la 3D, cependant nous avons vu juste une partie.

Chapitre 3: *Segmentation d'objets 3D*

I. Introduction

La segmentation de maillage 3D est apparue comme une problématique importante du fait de sa complexité de mise en œuvre et de la multitude d'applications qui en sont dépendantes.

Un maillage, et plus généralement une forme, peut être considérée d'un point de vue purement géométrique ou bien d'une manière plus orientée sémantique.

La façon d'interpréter un maillage est liée à l'application visée et amène à une segmentation en patches surfaciques ou bien en parties significatives. Dans le premier cas, le maillage est découpé en patches caractéristiques qui peuvent respecter des contraintes de courbures aux frontières, de planéité, de convexité, etc.

Les algorithmes se basent sur l'information géométrique de bas niveau, telle que la courbure pour définir des segments c'est-à-dire des régions⁹ qui sont géométriquement homogènes.

Dans le deuxième cas, l'objectif est de détecter des parties significatives de l'objet. Les algorithmes dits sémantiques visent à distinguer les régions d'un point de vue perceptuel, en suivant des notions de haut niveau définies dans la théorie de la perception humaine. [1]

Ce type d'approche est particulièrement utile pour les applications de déformation, et d'animation d'objets où la décomposition doit être significative.

La segmentation de maillage consiste à subdiviser une surface polygonale en différentes parties présentant des caractéristiques identiques. Cette opération est devenue un outil de prétraitement nécessaire à de nombreuses applications telles que la reconnaissance de forme, compression, etc.

Il existe dans la littérature un grand nombre d'algorithmes de segmentation permettant le traitement des modèles 3D.

II. Définition de la segmentation

Classiquement, on peut définir la segmentation comme étant une partition de l'image (I) en un nombre (K) de sous ensembles (R_i), appelés régions.

La détermination automatique du nombre de régions K est une vraie problématique.

⁹ Une région est un ensemble connexe de pixels ayant des propriétés communes (intensité, texture...).

Nous pouvons adopter la définition suivante pour la segmentation :

« La segmentation est un traitement de bas niveau qui consiste à créer une partition de l'image A en sous-ensembles R_i , appelés *régions* tels qu'aucune région ne soit vide, l'intersection entre deux régions soit vide et l'ensemble des régions recouvre toute l'image. Une *région* est un ensemble de pixels connexes ayant des propriétés communes qui les différencient des pixels des régions voisines. » [Cocquerez et Philipp, 1995].

III. Méthodes de segmentation

Comme cité précédemment la segmentation de maillage 3D est apparue comme une problématique importante du fait de sa complexité de mise en œuvre et de la multitude d'applications qui en sont dépendantes.

La façon d'interpréter un maillage est liée à l'application visée et amène à une segmentation en patches surfaciques ou bien en parties significatives.

Pour chaque catégorie de segmentation, de nombreuses méthodes ont été développées. Nous discutons dans ce qui suit des méthodes apparues récemment et qui bénéficient d'un rayonnement important dans la communauté.

III.1 Segmentation de maillages triangulaires et rectification directionnelle de frontières basées sur les tenseurs de courbure

Cette méthode proposée par « G. Lavoué, F. Dupont, A. Baskurt » utilise tous ce que peut contenir une maille triangulaire, en occurrence, les vertex et les triangles. Ceci permet d'avoir une source d'information assez fiable et assez complète pour pouvoir atteindre une segmentation qui soit correcte de l'objet 3D. [21]

III.1.1 Algorithme générale

Décomposition de maillages triangulaire en patches surfacique de courbure quasi-constante et connue délimitée par des frontières propres et régulière. L'algorithme est particulièrement adapté aux pièces issues de la CAO.

Notre approche se décompose en deux étapes (voir Fig.III.1):

- **La segmentation en régions surfaciques, basée sur la courbure** : Une étape de prétraitement identifie les arêtes et les sommets aigus.

Cette information est nécessaire pour la suite de l'algorithme, en particulier dans le cas des maillages CAO triangulés de façon optimale. Les sommets sont classifiés en

clusters selon leurs valeurs de courbure principale K_{min} et K_{max} . Un algorithme de croissance de régions est ensuite effectué réunissant les triangles en régions connexes à partir des clusters de courbure des sommets.

Un graphe d'adjacence des régions est alors construit et réduit afin de fusionner les régions semblables selon plusieurs critères (similitude de courbure, taille et périmètre commun).

- **La rectification de frontières:** Les arêtes frontières sont extraites à partir des régions surfaciques issues de l'étape précédente. Pour chacune de ces arêtes, un score de justesse est calculé qui notifie un degré de cohérence de l'arête frontière par rapport aux directions principales de courbure.

A partir de ce score, les arêtes estimées correctes sont marquées et servent de base pour un algorithme de suivi de contour qui extrait les frontières correctes finales de l'objet.

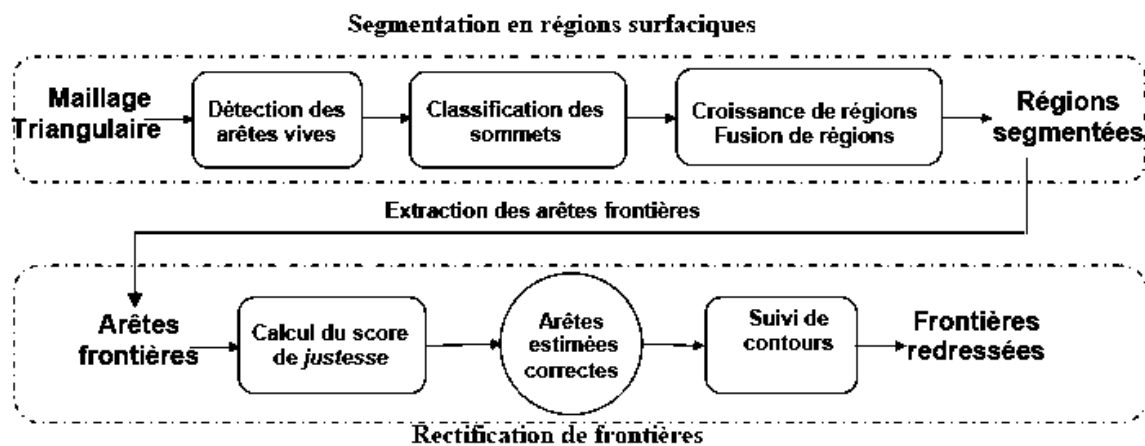


Figure III .1. Les deux étapes de l'algorithme : la segmentation en régions et la rectification de frontières.

C'est une méthode simple et efficace de classification, détecte les transitions de courbure et ne découpe pas seulement l'objet le long de ses arêtes vives ou de forte courbure.

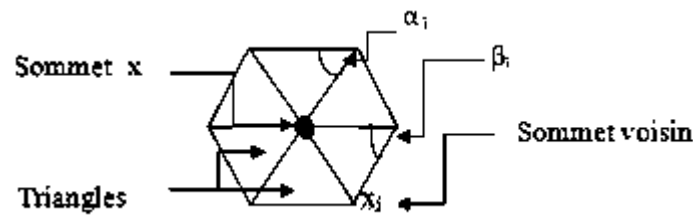


Figure III.2 : Un sommet et les triangles entourant ce dernier.

Les valeurs de courbures discrètes d'un sommet sont calculées comme suit :

- o La courbure discrète moyenne d'un sommet X est défini par :

$$K(X) = \frac{1}{2A} \sum_{j=1} (\cot(\alpha_j) + \cot(\beta_j)) (x_j - x)$$

Le mécanisme de croissance de régions extrait des régions de triangles homogènes en termes de courbure de manière très efficace, même pour les objets CAO, que peu de méthodes existantes sont capable de traiter.

La méthode de rectification de frontières basée sur l'orientation des tenseurs de courbure, permet de supprimer les artefacts généralement produits par les algorithmes de segmentation.

La méthode de rectification est indépendante de la segmentation en région et peut être employée comme post-processus à n'importe quel algorithme de décomposition ou détection de contours.

III.2 Ligne de Partage des Eaux (LPE)

La ligne de partage des eaux est une méthode de segmentation largement utilisée sur les images 2D.

Elle se place dans un contexte où le graphe à partitionner est considéré comme un relief et les sommets sont positionnés à une certaine hauteur.

Ce relief est plongé progressivement dans de l'eau et des inondations sont simulées à partir des minima (vertex ou plateaux n'ayant pas de voisin de niveau inférieur, voir figure III.1). Des bassins d'eau se forment et lorsque deux bassins se rencontrent, une LPE est créée. [20]



Figure III.3 : Minimum et plateaux.

II.2.1 Adaptation à la 3D

La contribution de Mangan et Whitaker sur la segmentation de maillage 3D par LPE a servie de base à de nombreux travaux ultérieurs.

Ces deux auteurs ont généralisé le principe de la LPE aux maillages 3D et ont proposé deux stratégies pour appliquer la LPE (Figure III.4) :

- L'approche ascendante, qui consiste à inonder les minima jusqu'à ce que les bassins voisins se rencontrent.
- L'approche descendante, qui correspond au positionnement d'un jeton en un point et à son déplacement le long de la pente la plus forte pour atteindre un minimum.

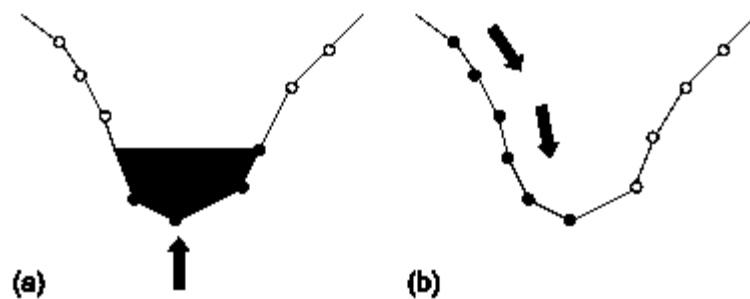


Figure III.4 : Les deux approches de la ligne de partage des eaux : (a) l'approche ascendante et (b) l'approche descendante.

Le résultat de la LPE souffre généralement de sur-segmentation.

Il est possible de contrôler ce phénomène à l'aide de marqueurs, d'un seuillage des régions ou encore de la segmentation hiérarchique.

II.2.2 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.

◆ 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 III.3 propose deux segmentations avec des niveaux de fusions différents. Le modèle de gauche contient 208 régions et celui de droite 57 régions.



Figure III.5 : Segmentation hiérarchique à partir de la ligne de partage des eaux. Modèle Cow avec 208 régions à gauche et 57 régions à droite.

◆ 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 .

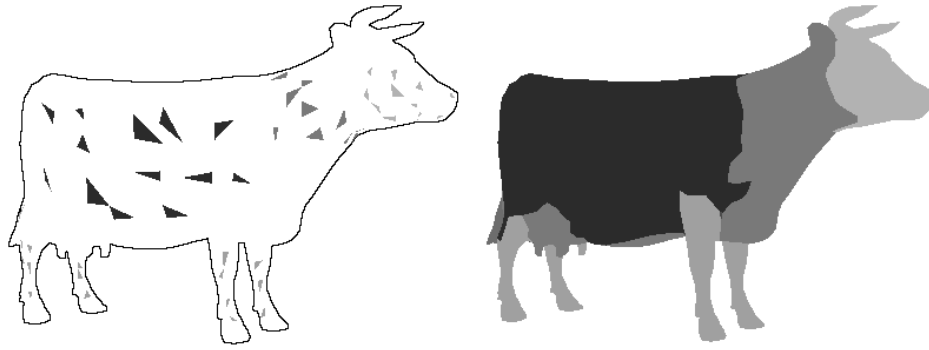


Figure III.6 : marqueurs et résultat de la LPE.

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.

II.2.3 Critère de courbure

Plusieurs approches ont été explorées pour obtenir une estimation de la courbure ; Meyer et al, 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. 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.



La matrice de covariance

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 vertex 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 . [26]

II.2.4. 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.



La voxelisation

Brunner et Brunnet 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 (figure III.7).[26]

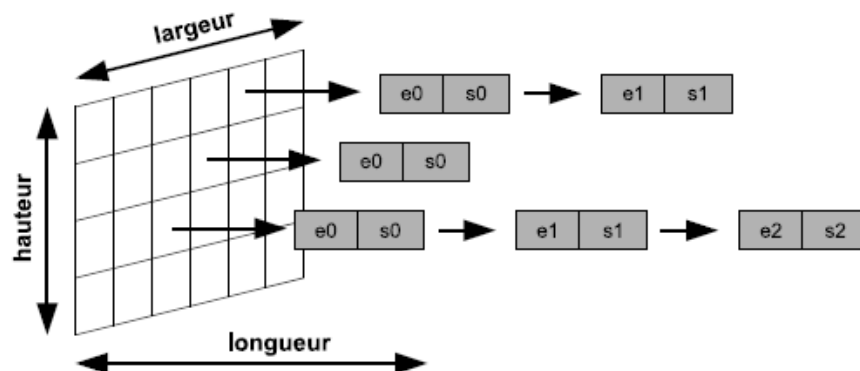


Figure III.7 : Structure de stockage des voxels. Seuls les couples de voxels (entrée/sortie) sont référencés.

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, 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.



La squelettisation

Pour extraire le squelette de voxels, l'algorithme de squelettisation de « Palàgyi » est utilisé 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.

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.

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.



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.

Le marquage peut commencer par l'attribution d'un label différent à chaque arc. 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

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. Les faces qui sont liés à des voxels ambigus ne seront pas labellisées à cette étape mais le seront après la segmentation par ligne de partage des eaux.

III.3 Croissance de régions

La croissance de régions représente une des méthodes les plus intuitives pour segmenter un maillage.

Cette technique consiste à faire progressivement grossir les régions autour de leur point de départ.

L'algorithme se compose de deux étapes:

1. Trouver les points de départ des régions.
2. Faire grossir les régions par agglomération des pixels voisins.

III.3.1 Points de départ (seeds)

Le choix des points de départ est la partie critique de l'algorithme. En effet, l'étape de croissance va utiliser une mesure de similarité pour choisir les pixels à agglomérer.

Si le point de départ est situé dans une zone non homogène, la mesure de similarité va produire de fortes variations et la croissance va s'arrêter très tôt.

Par conséquent, il convient de choisir les points de départs dans des zones les plus homogènes possibles.

III.3.2 Croissance (growing)

Cette étape a pour objectif de faire grossir une région en agglomérant des pixels voisins. Les pixels sont choisis afin de maintenir l'homogénéité de la région. Pour cela, nous devons définir un indicateur d'homogénéité. Les pixels voisins sont ajoutés à la région si l'indicateur d'homogénéité reste vrai.

La croissance s'arrête lorsqu'on ne peut plus ajouter de pixels sans briser l'homogénéité. [23]

III.4 Système Multi-Agents

Le système multi-agents (SMA) a pour objet de réaliser le processus d'extraction par la fusion de comportements de coopération et de compétition associés à des agents spécifiques, ces agents étant liés à différents objets de la scène. Par comparaison aux autres systèmes existants en segmentation d'image, l'originalité de l'approche proposée porte sur la détection simultanée de plusieurs objets et dans des temps de calcul rapides face à de grands volumes de données à traiter. Cette approche est caractérisée par un couplage de processus de croissance de régions 3D avec un apprentissage supervisé au sein d'un système multi-agents.[27]

Le système multi-agents produit une segmentation d'images 3D basée sur une étape initiale de sélection, durant laquelle l'utilisateur pointe interactivement un ou plusieurs points germes à l'intérieur de chacun des objets d'intérêt.

Le SMA est composé :

- d'un ensemble d'agents purement situés appelés agents « Workers », qui coopèrent et entrent en compétition de manière à, par croissance de région, maximiser leur « territoire image » (selon la vérification d'un critère de texture et d'intensité).
- d'un agent purement communicant (appelé agent « Contrôleur ») qui coordonne les comportements de tous les agents « Workers ».
- L'architecture adoptée correspond à une organisation micro-sociale puisque la société est seulement composée de quelques agents. L'environnement est défini par le volume de voxels (l'image 3D) à segmenter et les comportements des différents agents introduits sont complètement déterministes.

La segmentation par SMA est une nouvelle méthode de segmentation 3D multi-objets basée sur un schéma multi-agents combiné avec une approche de classification supervisée et un procédé de croissance de région.

L'approche multi-objets autorise un contexte multi-agents et une implémentation compétitive qui conduisent à une amélioration qualitative et quantitative des résultats, en particulier dans les situations ambiguës.

IV. Conclusion

La segmentation d'images est l'une des opérations les plus importantes dans le domaine de l'analyse d'images. L'objectif de cette opération est de diviser l'image en zones homogènes (régions).

Ainsi, la segmentation d'une image permet de retrouver des formes ou des zones ayant un sens pour l'utilisateur ou pour une autre opération de traitement d'images.

La segmentation d'une image cause encore aujourd'hui beaucoup de problèmes, aucune méthode ne fonctionne pour toutes les images, il n'y a pas de garantie, pas de recette miracle !

Chapitre 4: *Implémentation*

I. Introduction

Le but de notre travail est de réaliser une application qui permet de faire la décomposition d'un objet 3D en régions, afin d'être utilisées dans les domaines qui interagissent avec la 3D, tel que le traitement des données tridimensionnelles.

II. Choix techniques

II.1 Le langage de programmation

Ce projet étant essentiellement centré autour de la réalisation d'une interface graphique, notre choix s'est naturellement porté vers des langages orientés objet : Java et C++. Il est donc nécessaire de faire un choix.

Notre projet en plus de l'interface graphique, nécessite une bibliothèque graphique qui permet de développer des applications 3D. Il existe plusieurs bibliothèques en C++ notamment OpenGL.

Mais l'existence de la bibliothèque graphique VTK en java en plus du fait que l'un des principaux avantages de Java est sa portabilité native, contrairement à C++, a finalement confirmé notre choix.

II.2 L'environnement de développement

Il existe une multitude d'environnements de développement (IDE) qui prennent en charge les langages Java et C++. Parmi eux, seul un nombre plus restreint permet une utilisation plus rapide et efficace, Eclipse et Visual Studio offrent tous deux des outils de qualité professionnelle.

Cependant, Eclipse a l'avantage d'offrir des fonctionnalités telles que la vérification de la syntaxe en cours de frappe, ainsi que le « «refactoring¹⁰ ». et en plus de sa portabilité c'est un logiciel gratuit contrairement à Visual Studio, donc on a écarté ce dernier comme outil de développement.

III. Format des données d'entrée

Il existe un grand nombre de type de fichiers concernés par la 3D, nous avons eu l'occasion d'en décrire quelques uns dans le chapitre II, et après avoir fait un tours sur

¹⁰ Le refactoring est une opération qui consiste en la maintenance du code d'un programme. Pour un langage de programmation orientée objet le refactoring peut par exemple permettre de renommer des membres de classe et de répercuter les changements dans tout le code source (commentaires inclus). Le refactoring est couramment employé pour nettoyer le code.

ces types, notre choix s'est porté sur les fichiers de type VTK a cause de la simplicité de la structure des points 3D et des faces qui constituent l'objet tridimensionnel .

IV. Fonctionnement du logiciel

IV.1. Présentation

Notre application est conçue pour décomposer une image 3D en mailles triangulaires elle se présente comme suit :

Premièrement on a la figure d'accueil qui est montrée ci-dessous (figure IV.1)

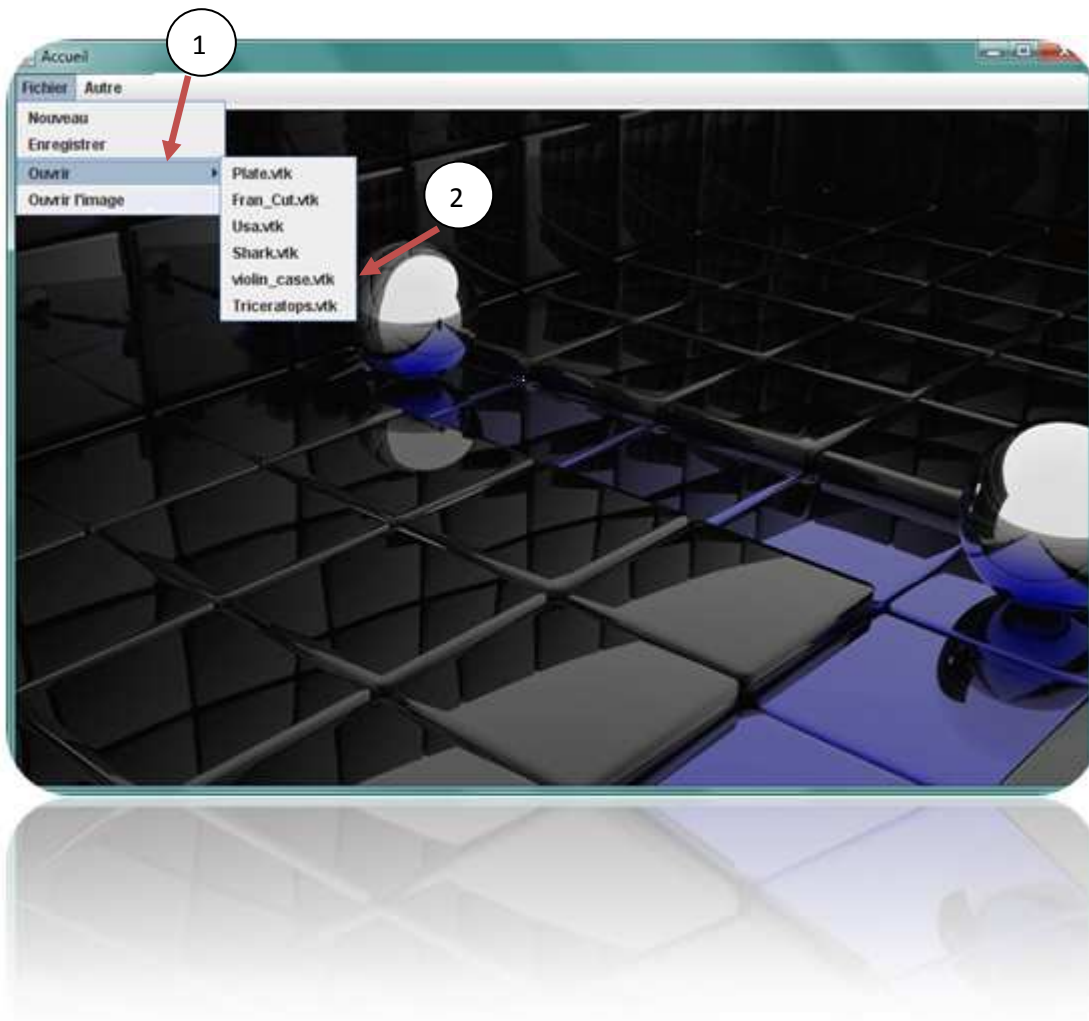


Figure IV.1 : Accueil de l'application

Comme on peut l'apercevoir dans la figure IV.1 on a le menu qui nous permet d'ouvrir une image 3D d'extension .vtk en cliquant sur le bouton (1) ce qui fait apparaître la liste des images 3D (2) de notre base de données.

Après que l'utilisateur ait choisi une image il clique sur celle-ci pour l'afficher ce qui donne la figure IV.2.

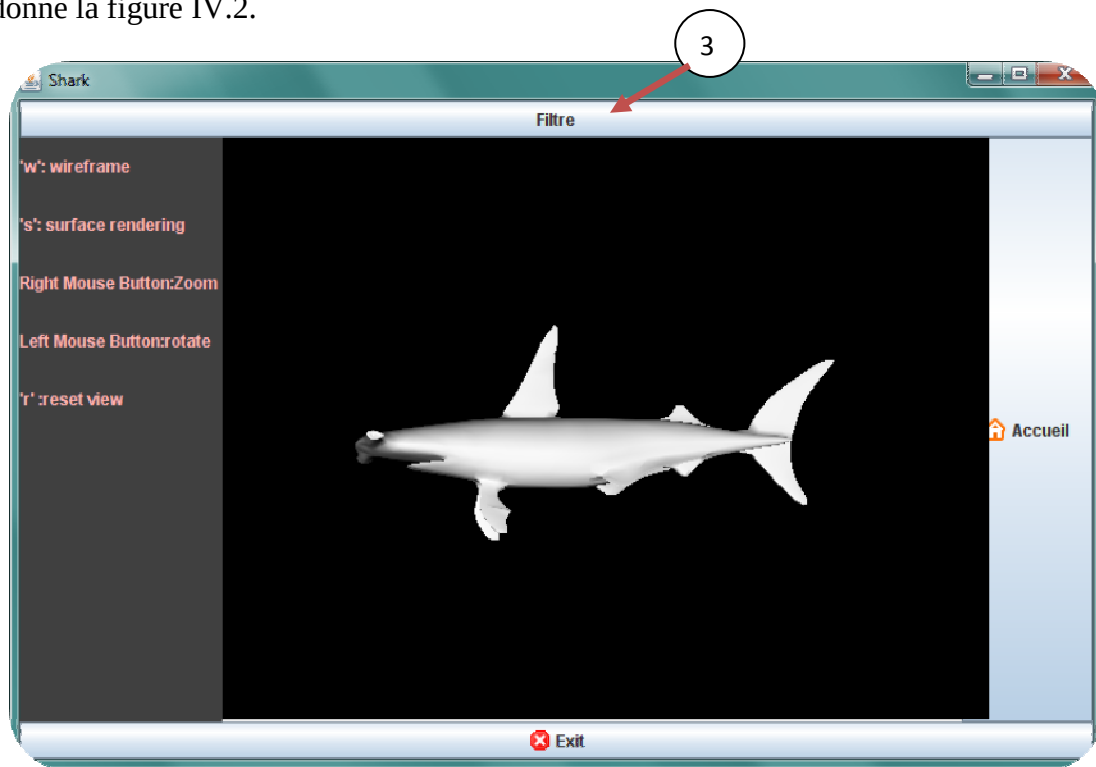


Figure VI.2 : Image 3D .vtk

Après avoir visualisé l'image, on lui applique le filtre triangulaire en cliquant sur le bouton 3 .

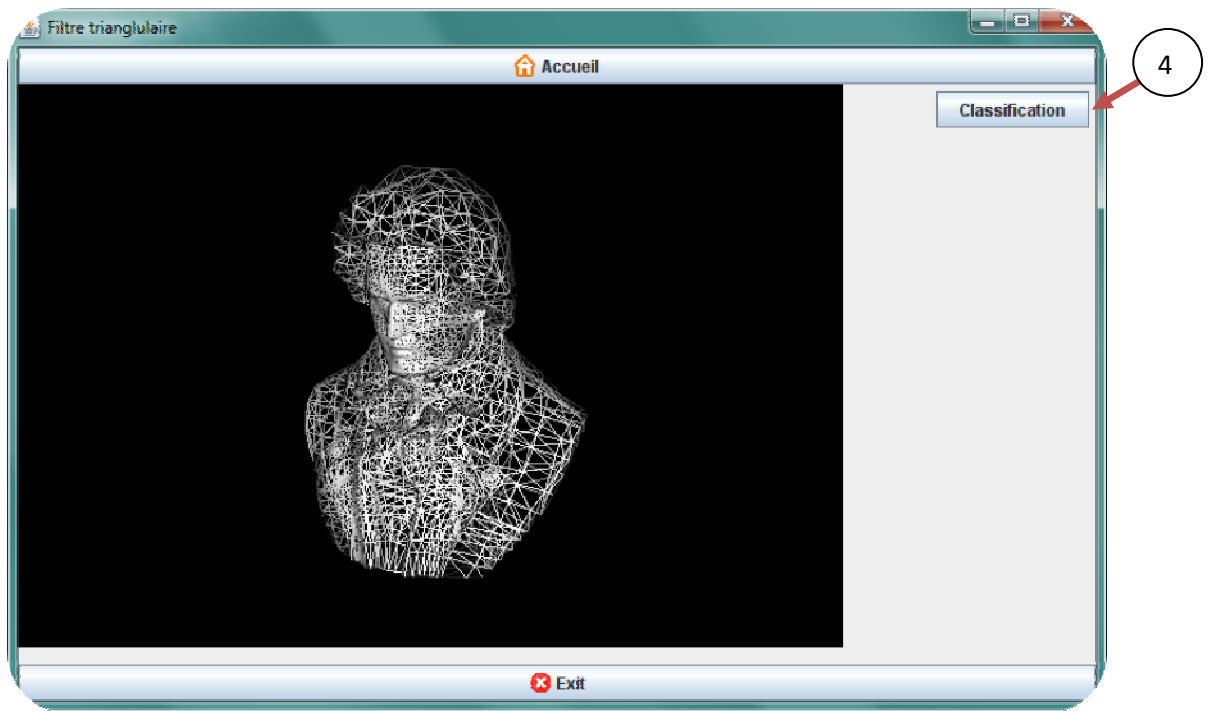


Figure IV .3 : Image 3D décomposée.

Comme on peut le voir sur la figure VI.3 après qu'on est appliqué le filtre sur l'image 3D, on a comme résultat une image décomposée en mailles triangulaires.

L'étape suivante est la classification des régions en cliquant sur le bouton

4

Qui nous donne la forme suivante (Figure VI.4).

The screenshot shows a window titled "Choix" with a standard Windows-style title bar. Inside the window, there are two main sections. The first section is titled "Cocher le type de classification que vous voulez appliquer" and contains four radio button options: "Type Gaussien", "Type Mean" (which is selected), "Type Minimum", and "Type Maximum". The second section is titled "Saisir le nombre de classes" and features a text input field containing the number "5". Below the input field, there is a label "Nombre de classes". At the bottom of the window, there are two buttons: "Valider" (with a green checkmark icon) and "Annuler" (with an orange circular arrow icon). A red arrow points from a circled number "5" to the "Valider" button.

Figure IV.4 : Choix d'algorithme à appliquer

L'utilisateur doit cocher le type d'algorithme à appliquer, une fois son choix est fait, il lui suffit de préciser le nombre de classes et de cliquer sur le bouton **5** et on a le résultat final attendu par l'utilisateur dans la figure (Figure IV.5).

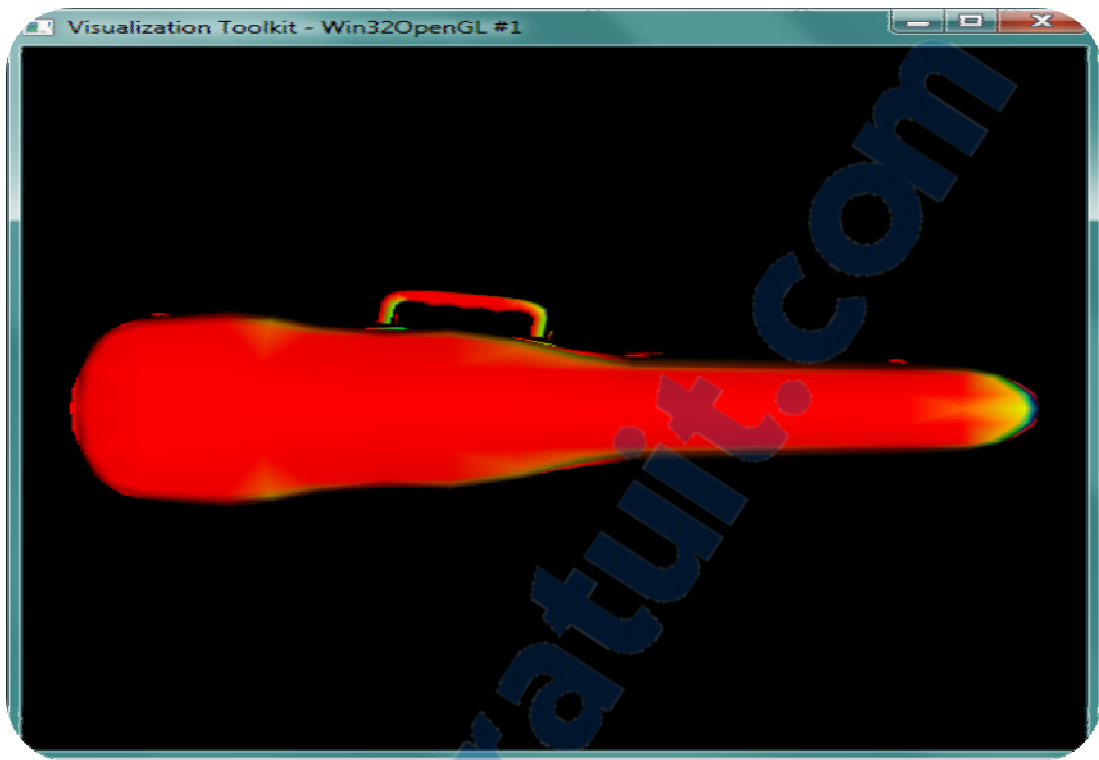


Figure IV.5 : Image 3D résultat

Programme :

Pour pouvoir ouvrir une image 3D, la librairie VTK nous offre la classe **vtkPolyDataReader**, quand à l'enregistrement de l'image résultat on a utilisé la classe **vtkPolyDataWriter**.

Pour appliquer le filtre triangulaire sur l'image 3D en utilisant la librairie VTK, on a la classe **vtkTriangleFilter**.

Exemple :

```
vtkSphereSource sphere =new vtkSphereSource(); /**Source***/

    sphere.SetRadius(5);

    sphere.SetThetaResolution (36);
    sphere.SetPhiResolution(18);

vtkPolyDataMapper isoMapper=new vtkPolyDataMapper(); /**Mapper***/
    isoMapper.SetInput(sphere.GetOutput);
    isoMapper.ScalarVisibilityOn;

    vtkActor isoActor=new vtkActor() ; /**Actor***/
    isoActor.SetMapper(isoMapper) ;

    vtkRenderer ren1=new vtkRenderer();
    ren1.AddActor(isoActor) ;
    ren1.SetBackground(1,1,1) ;

    vtkRenderWindow renWin=new vtkRenderWindow(); /**Renderer***/
    renWin.AddRenderer(ren1) ;
    renWin.SetSize(500,500) ;

    vtkRenderWindowInteractor iren= new
    vtkRenderWindowInteractor() ; /**Render Window***/

    iren.SetRenderWindow(renWin) ;
    iren.Initialize ;
```

IV.2 Expérimentation :

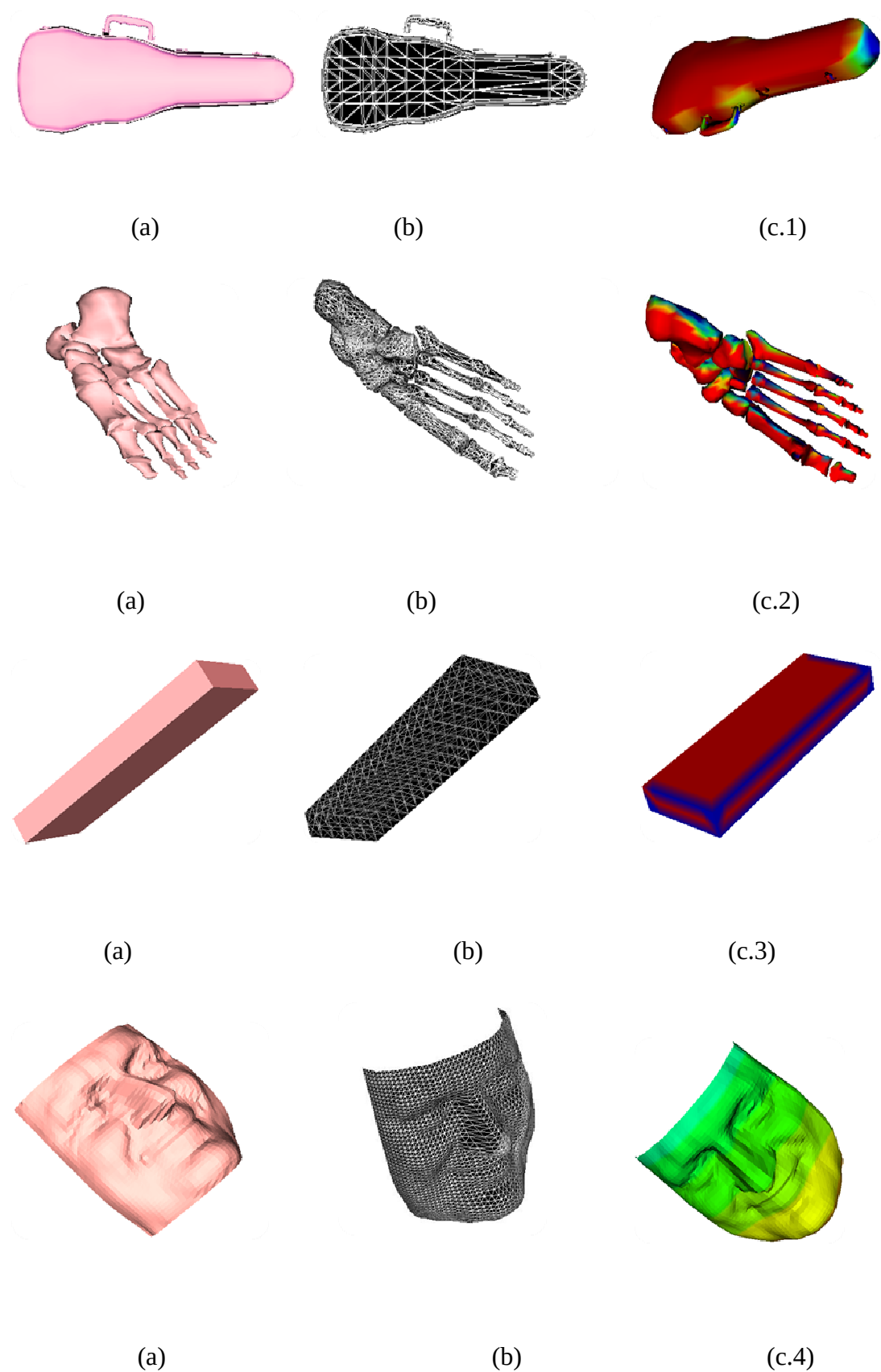


Figure IV.6 : (a) Affichage de l'objet 3D, (b) Visualisation de maillage triangulaire, (c) : classification, (c.1): *type minimum*, (c.2): *gaussien*, (c.3) *maximum*, (c.4) : *mean*.

V. Conclusion

Dans ce chapitre nous avons présenté notre application dont le but est la décomposition de maillage triangulaire, en spécifiant les différentes étapes qui constituent notre travail.

Conclusion générale

Conclusion générale

Au cours de ce projet nous avons eu l'occasion de travailler avec les objets 3D, et de mesurer leurs importance dans les applications actuelles, qui visent à donner aux machines la capacité de voir et de détecter afin de pouvoir aider, ou même remplacer l'être humain dans de nombreuses applications, notamment dans le domaine de l'imagerie médicale. De ce fait la décomposition de maillage triangulaire est devenue une étape de prétraitement très importante pour des traitements ultérieurs sur ces objets.

Nous avons présenté dans ce modeste travail, une méthode de décomposition de mailles triangulaires, en se basant sur le critère de la courbure discrète, et en utilisant les triangles pour générer les différentes régions du modèle 3d.

Notre travail s'inscrit dans un objectif plus modeste de segmentation. Un bon nombre d'améliorations sont à envisager :

- Effectuer un algorithme de croissance de régions réunissant les triangles en régions connexes à partir des clusters de courbure des sommets.
- Construction et réduction du graphe d'adjacence des régions afin de fusionner les régions semblables.

Références bibliographiques

Références bibliographiques

- [1] Severine DUBUISSON, Modélisation des objets solides : « *Algorithmique Graphique Et Modélisation* » Mars 2007.
- [2] « *courbes et surface* » consulté le 23/04/2010, disponible sur : <http://vrlab.epfl.ch/~thalmann/infographie/part3.pdf>
- [3] « *représentation des courbes et des surfaces* » consulté le 06/05/2010, disponible sur : <http://perception.inrialpes.fr/people/Boyer/Teaching/RICM/c5.pdf>
- [4] « *les courbes et surfaces splines* » consulté le 12/03/2010, disponible sur: http://www2.lifl.fr/~grisoni/IMA5/IRV/cours1_Splines.pdf
- [5] Christian NGUYEN «*Modélisation 3D*».
- [6] Delphine QUESTE «*Modélisation et Simulation des Espaces Bâtis* » Mémoire de Master Design Global, Octobre 2006.
- [7] Denis DIDEROT « *La modélisation 3D* »,2010.
- [8] «Wikipedia, the free encyclopedia » consulté le 24/03/2011, disponible sur : <http://fr.wikipedia.org/wiki/NURBS>
- [9] François SILLION « *Simplification de maillage 3D* ».
- [10] «*Subdivision de surfaces*» consulté le 24/03/2011, disponible sur : http://iparla.inria.fr/publications/2004/Bou04/surface_reconstruction_with_subdivision_master_thesis.pdf
- [11] Frédéric HESCHUNG « *Vers une construction et une adaptation automatiques des Maillages* », 2005.
- [12] Fabien VIVODTZEV « *Simplification et visualisation de gros ensembles volumiques* » juin 2002.
- [13] Marc TOUBIN « *Caractérisation et simplification de modèles numériques de scènes réelles par approche multi-résolution dans un contexte multi-capteurs* » Novembre 2000.
- [14] Claudia NEGULESCU « *Interpolation et approximation : Courbes de Bézier* », 2008
- [15] « Wikipedia, free encyclopedia », consulté le 11/04/2011, disponible sur : http://fr.wikipedia.org/wiki/Format_de_donn%C3%A9es

- [16] Fabrice Aubert « *Courbes et surfaces, Modélisation 3D et Synthèse* », 2010.
- [17] « Wikipedia, free encyclopedia », consulté le 14 /04/2011, disponible sur :
<http://fr.wikipedia.org/wiki/DXF>
- [18] « Wikipedia, free encyclopedia », consulté le 14 /04/2011, disponible sur :
<http://www.vtk.org/VTK/img/file-formats.pdf>
- [19] Tamy BOUBEKEUR « *Reconstruction de surface à l'aide de surfaces de Subdivision* ».Juin 2004.
- [20] Halim BENHABILES, Jean-Philippe VANDEBORRE, Guillaume LAVOUÉ, et Mohamed DAOUDI « *Une collection de modèles 3D avec vérité-terrain pour L'évaluation objective des algorithmes de segmentation* ».
- [21] Guillaume LAVOUÉ, F. DUPONT, A. BASKURT « *Segmentation de maillages triangulaires et rectification directionnelle de frontières basées sur les tenseurs de courbure* » 2004.
- [22] S. DELEST, R. BONÉ et H. CARDOT « *Etat de l'art de la segmentation de maillage par patchs surfaciques* », 31 mai 2007.
- [23] Xavier PHILIPPEAU « *Segmentation en régions* ». Date de publication : 05/01/2008.
- [24] Alain BOUCHER « *TRAITEMENT D'IMAGE, Segmentation* »
- [25] AMIR Samir « *évaluation de la segmentation d'images sans références étude et Implémentation* »2005.
- [26] S.Delest. G. Pageot. R. Boné. H. Cardot. « *Génération automatique de marqueurs pour la ligne de partage des eaux 3D* ».
- [27] Guilhelm SAVIN« *Segmentation 2D et 3D par Systèmes Multi-Agents*», juin 2008.

- [28] J. Fleureau, M. Garreau, D. Boulmier, C. Leclercq et A. Hernandez,
« *Segmentation 3D multi-objets* », 4 novembre 2009.

Annexe

Annexe

I. La bibliothèque VTK (Visualisation ToolKit)

La librairie VTK est une puissante bibliothèque graphique permettant de visualiser de gros volumes de données 2D ou 3D.

Très fréquemment utilisée dans de nombreux domaines (physique, chimie, médecine, mathématiques...), elle bénéficie d'une évolutivité très importante et de développeurs très réactifs.

VTK est une librairie écrite en C++, elle peut être utilisée soit directement via les langages C++, Python, Tcl ou Java, soit indirectement via des interfaces graphiques telles que Paraview (écrite en Tcl) ou Mayavi (écrite en Python).

II. Avantages et inconvénients

Son grand avantage est son large spectre d'applications (du traitement de l'image à la visualisation de données volumiques).

VTK a aussi un système de picking et d'interaction avec les objets 3D très avancé et utilisant divers algorithmes (sélection par frustum, pick color, etc.). Vous pouvez sélectionner des objets entiers ou des primitives graphiques (triangles, points, lignes, etc.).

Son principal défaut est que les performances de VTK s'effondrent s'il contient plus de 1000 objets 3D. Si votre application dépasse ce nombre on vous déconseille son utilisation. Ceci est dû à l'architecture de VTK qui donne une très grande généricité d'utilisation mais détruit les performances pour un très grand nombre d'objets.

Néanmoins avec peu d'acteurs, les performances de VTK sont très correctes (utilisant des « display list » donc l'affichage d'objets de plusieurs millions de triangles est possible).

Résumé

Le développement continu des techniques multimédias et de la réalité virtuelle a suscité un intérêt croissant pour l'utilisation de contenus tridimensionnels (3D), et afin de faciliter le traitement et le transfert de ces objets, de nombreux algorithmes de décomposition ont été proposés. La décomposition de maillage triangulaire correspond au découpage du maillage en régions, c'est un outil nécessaire à de nombreuses applications citons la compression et la simplification. Nous avons présenter dans ce travail une méthode de décomposition de mailles triangulaires en se basant sur la critère de la courbure discrète, qui permet de déterminer la forme de l'objet (un plan, une sphère), et utilise tous ce que peut contenir une maille triangulaire, en occurrence les vertex et les triangles.

Abstract

The continuous development of multimedia technologies and virtual reality has generated growing interest in use of content dimensional (3D), and to facilitate the processing and transfer of these objects, many decomposition algorithms have been proposed. The decomposition of triangular mesh corresponds to cutting the mesh into regions. This is a necessary tool for many applications include compression and simplification. We presented in this work a method for decomposing triangular meshes based on the criterion of discrete curvature, which determines the shape of the object (a plane, a sphere), and use all that can contain a triangular mesh, in this case the vertices and triangles.

ملخص

التطوير المستمر لتقنيات الوسائط المتعددة والواقع الافتراضي ولدت اهتماما متزايدا في استخدام المحتوى الأبعاد (D3) ، ولتسهيل معالجة ونقل هذه الأجسام قد تم اقتراح العديد من خوارزميات التفكيك. تفكيك الشبكات يناظر قطع الشبكة إلى مناطق , وهو أداة ضرورية للعديد من التطبيقات التي تشمل الضغط والتبسيط. نقدم في هذا العمل وسيلة لتفكيك الشبكات الثلاثية على أساس معيار الانحناء المنفصل، الذي يحدد شكل الجسم (مسطح، كروي)، ويستخدم كل ما قد تحتوي عليه الشبكة الثلاثية , أي القمم والمثلثات.