



HAL
open science

Exploitation d'images numériques pour la simplification et la déformation de modèles polyédriques

Minica Panchetti

► **To cite this version:**

Minica Panchetti. Exploitation d'images numériques pour la simplification et la déformation de modèles polyédriques. Sciences de l'ingénieur [physics]. Arts et Métiers ParisTech, 2009. Français. NNT : 2009ENAM0004 . pastel-00005131

HAL Id: pastel-00005131

<https://pastel.hal.science/pastel-00005131>

Submitted on 23 Jun 2009

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

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



Ecole doctorale n° 432 : Sciences des Métiers de l'Ingénieur

THÈSE

pour obtenir le grade de

Docteur

de

l'École Nationale Supérieure d'Arts et Métiers

Spécialité "Conception"

*présentée et soutenue publiquement
par*

Minica PANCHETTI

le 10 avril 2009

**EXPLOITATION D'IMAGES NUMERIQUES
POUR LA SIMPLIFICATION ET LA DEFORMATION
DE MODELES POLYEDRIQUES**

Directeur de thèse : Philippe VERON

Co-encadrant de thèse : Jean-Philippe PERNOT

Jury :

Mme. Claire LARTIGUE, Professeur, LURPA, IUT de Cachan Présidente du jury
M. Jean-Claude LEON, Professeur, G-SCOP, INPG, Grenoble Rapporteur
M. Marc NEVEU, Professeur, LE2I, Dijon Rapporteur
M. Rémy BULOT, Professeur, I&M, Marseille Examineur
M. Philippe VERON, Professeur, IMS, Arts et Métiers ParisTech, Aix-en-Pce. Examineur
M. Jean-Philippe PERNOT, MdC., IMS, Arts et Métiers ParisTech, Aix-en-Pce Examineur

Laboratoire des Sciences de l'Information et des Systèmes
Arts et Métiers ParisTech, centre d'Aix-en-Provence

*A mes parents,
A William.*

Remerciements

Je voudrais remercier tous les membres du jury pour l'attention qu'ils ont bien voulu porter à ce travail :

- *Jean-Claude Léon et Marc Neveu pour avoir accepté la charge de rapporteur et pour leur relecture attentive,*
- *Claire Lartigue et Rémy Bulot pour m'avoir fait l'honneur de l'examiner,*
- *Philippe Véron qui a bien voulu diriger mes travaux pendant ces trois années. Qu'il trouve toute ma gratitude pour son suivi permanent, ses conseils avisés, la disponibilité et la bienveillance dont il a toujours fait preuve à mon égard.*
- *Jean-Philippe Pernot qui a bien voulu co-encadrer mes travaux. Qu'il trouve toute ma reconnaissance pour la qualité scientifique de ces remarques, la rigueur de ses raisonnements, ses encouragements et enfin pour tout le temps, considérable, qu'il a bien voulu me consacrer.*

Je remercie également Daniel Brun-Picard, pour m'avoir accueillie lors de mon master recherche au sein de l'équipe « Ingénierie, Mécanique, Systèmes » du laboratoire SIS.

Je n'oublie pas mes collègues et amis de toute l'équipe qui ont su créer une ambiance de travail chaleureuse et en particulier : Lionel, Cédric, Ruding, Anne, Thibaut, Fadel ainsi que Thierry, Nicolas, Michèle et Williams auxquels j'adresse une pensée bien amicale.

Enfin, un grand merci à mes parents pour leur affection et leurs encouragements, à William, mon mari, pour son soutien et sa compréhension tout au long de ces années.

Table des matières

I Représentations numériques et techniques de traitement associées : état de l'art	9
1 Représentations numériques multiples	11
1.1 Une variété de modèles géométriques et autres représentations	12
1.2 Courbes et surfaces paramétriques	17
1.3 Nuages de points	17
1.4 Modèles polyédriques	19
1.5 Images numériques	22
1.5.1 Images matricielles	22
1.5.2 Images vectorielles	25
1.6 Représentations symboliques ou schématiques	26
1.7 Conclusion	27
2 Techniques de traitement et d'exploitation d'images	31
2.1 Introduction	32
2.2 Techniques de traitement d'images	32
2.2.1 Extraction de contours	32
2.2.2 Segmentation	37
2.2.3 Reconnaissance d'objets	40
2.2.4 Bilan des techniques d'extraction des contours, de segmentation d'images et de reconnaissance d'objets	43
2.3 Calibration de caméras	43
2.3.1 Modèle de projection perspective (modèle sténopé linéaire)	44
2.3.2 Lancer de rayons	49
2.3.3 Bilan des méthodes de calibration de caméras et de lancer de rayons	49
2.4 Techniques de reconstruction 3D à l'aide d'images	49
2.4.1 Tomographie	49

2.4.2	Stéréoscopie	50
2.4.3	Shape From Shading	52
2.4.4	Bilan des techniques de reconstruction 3D à partir d'image(s)	55
2.5	Conclusion	57
3	Techniques de manipulation de modèles polyédriques	59
3.1	Manipulations avancées de modèles polyédriques	60
3.2	Techniques de simplification de modèles polyédriques	63
3.2.1	Mécanismes de simplification de modèles polyédriques	63
3.2.2	Simplifications géométriques	65
3.2.3	Simplifications perceptuelles	70
3.2.4	Synthèse des techniques de simplification de modèles polyédriques	73
3.3	Techniques de déformation de modèles polyédriques	74
3.3.1	Méthodes géométriques	74
3.3.2	Méthodes non géométriques	76
3.3.3	Synthèse des méthodes de déformation de modèles po- lyédriques	84
3.4	Conclusion	84
 II Nouvelle approche de traitement de modèles polyédriques basée sur l'exploitation d'images numériques		89
4	Simplification de polyèdres à l'aide d'images	91
4.1	Introduction	92
4.2	Schéma de traitement adapté à la simplification	93
4.3	Création de la carte de tolérances	96
4.3.1	Extraction et vectorisation des lignes de caractère	96
4.3.2	Projection des contours et calcul de la position des sphères d'erreur	97
4.3.3	Calcul des rayons des sphères d'erreur	102
4.4	Résultats et analyses	104
4.4.1	Critères d'analyse et métriques de comparaison des résultats	104
4.4.2	Résultats	105
4.5	Conclusion	113
5	Déformation de polyèdres à l'aide d'images	117

5.1	Introduction	118
5.2	Etat de l'art des méthodes de remplissage de trous	121
5.3	Prétraitements des trous	125
5.3.1	Identification et nettoyage du contour des trous	125
5.3.2	Remplissage topologique de trous	128
5.3.3	Calibration de la caméra	129
5.4	Définition des contraintes sur le maillage inséré	129
5.4.1	Définition de contraintes par stéréoscopie	129
5.4.2	Définition de contraintes à l'aide des méthodes de Shape From Shading	131
5.5	Spécification et résolution du problème d'optimisation numé- rique	138
5.5.1	Choix de la fonctionnelle	138
5.5.2	Linéarisation des contraintes	141
5.6	Résultats et comparaison des deux approches de remplissage de trous proposées	142
5.6.1	Résultats de la méthode de remplissage de trous à l'aide de contraintes obtenues par stéréoscopie	142
5.6.2	Résultats de la méthode de remplissage de trous à l'aide de contraintes obtenues par SFS	146
5.7	Conclusion	154
Synthèse et perspectives		159
	Synthèse	159
	Perspectives	160
	Améliorations du module de simplification de modèles poly- édriques à partir d'image(s)	160
	Améliorations du module de déformation de modèles poly- édriques à partir d'images	161
	Applications aux images réelles	162
	Mise à jour de modèles polyédriques	162
Bibliographie		163

Introduction

De nos jours, la **maquette numérique**¹ d'un produit (fig.1) est devenue un support d'informations privilégié dans un contexte d'ingénierie collaborative et de conception intégrée. En proposant des méthodes et des outils adaptés à leurs besoins et aux tâches qui leur sont spécifiques, cette architecture de travail commune facilite les interactions entre les différents acteurs pouvant intervenir dans les processus de conception, le plus souvent à distance, et ce à n'importe quel stade du cycle de vie du produit [Murphy *et al.* 2001].



FIGURE 1 – La maquette numérique du cockpit de l'Airbus A400 *courtoisie de la compagnie Airbus.*

En effet, de nombreuses activités requièrent un support de travail commun : le travail collaboratif, le développement et la gestion des processus de production, mais également la prise de décision, depuis l'étape de conception jusqu'à la maintenance du produit. Ce besoin peut concerner d'autres domaines que la conception mécanique, comme cela est notamment le cas pour le bâtiment, l'architecture et le patrimoine où l'utilisation des maquettes numériques tend à se développer considérablement.

1. DMU : Digital Mock-Up

Parmi les informations contenues et manipulées au sein des maquettes numériques, on retrouve les modèles géométriques, supports privilégiés permettant de véhiculer les informations entre les différents métiers (design, simulation, fabrication, etc.).

Différents types de **modèles géométriques** supportant la maquette numérique peuvent coexister au sein du même environnement de conception ; en effet, un même produit pourra, selon le métier considéré, être représenté par exemple :

- par un modèle CAO² (B-Rep³, CSG⁴, etc.) pour des besoins de conception mécanique,
- par un nuage de points issus d'un balayage au scanner laser de l'objet physique pour des besoins d'Ingénierie Inverse⁵, pour la rétro conception de pièces mécaniques ou la reconstruction de maquettes virtuelles d'édifices en architecture,
- par un modèle polyédrique obtenu par triangulation du nuage de points, pour des besoins de visualisation, de simulation numérique, ou de prototypage rapide,
- par des surfaces NURBS⁶, pour des besoins de conception de formes aérodynamiques, ergonomiques et esthétiques dans les domaines de l'automobile et l'aéronautique,
- par des courbes 3D pour réaliser des esquisses ou des trajectoires d'usinage par exemple,
- ...

La multitude des modèles géométriques existants et leur complémentarité mettent donc en exergue le besoin de **multi-représentations** et de **multi-résolutions** (modélisation d'un objet à plusieurs niveaux de détails⁷) dans la structuration de la maquette numérique. A titre d'exemple, la maquette numérique de l'Airbus A400M représente 15 téraoctets de données, dont 10 téraoctets pour les modèles géométriques.

Proposer de nouveaux modèles, méthodes et outils visant à faciliter le stockage, l'indexation, l'archivage, l'adaptation et la maintenance de ces modèles géométriques est donc devenu un véritable enjeu industriel. Les travaux de recherche présentés ici portent sur le **couplage d'images numériques et de modèles polyédriques 3D**. Des informations véhiculées par les images sont exploitées dans le but d'aider ou de contrôler le traitement des

2. CAO : Conception Assistée par Ordinateur

3. B-Rep : Boundary Representation

4. CSG : Constructive Solid Geometry

5. Reverse Engineering

6. NURBS : Non Uniform Rational B-Splines

7. LOD : Levels Of Details

modèles géométriques. D'un côté, de nombreuses techniques de traitement d'images, de photogrammétrie et reconstruction 3D à l'aide d'images 2D, révèlent la richesse de leur contenu et occupent une place centrale dans le domaine de l'information. De l'autre, diverses techniques de manipulation de polyèdres permettent leur visualisation et leur exploitation. Ici, l'idée est d'exploiter au mieux tous ces supports pour traiter efficacement la géométrie des modèles.

En **vision par ordinateur**, les images numériques correspondent à la projection bidimensionnelle sur la rétine, des objets tridimensionnels vus par l'oeil humain. S'en suit une étape d'interprétation (de reconnaissance) par le cerveau ou par un ordinateur dans le cas du processus de vision artificielle. Parce qu'elles sont une **représentation** de ce que l'oeil humain perçoit, les photographies sont très fréquemment utilisées comme textures dans les applications de réalité virtuelle notamment, car elles ajoutent un réalisme avancé aux modèles 3D sur lesquelles elles sont appliquées. Les algorithmes de filtrage des contours dans une image par exemple, révèlent les **lignes de caractère** (fig. 2) de l'objet physique qu'elle représente, ces lignes de caractère étant les courbes de l'objet qui impactent son aspect visuel [FIORES II]. Effectivement, les contours peuvent correspondre à une arête vive, une bordure ou plus généralement un changement quelconque de forme. Identifier ces contours de façon automatique par ordinateur correspond à une version artificielle du processus d'interprétation des images réalisé par le cerveau.

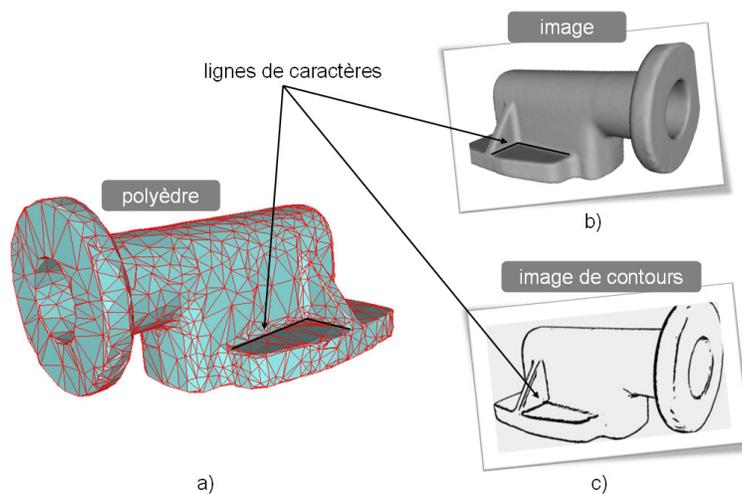


FIGURE 2 – Identification des lignes de caractère d'un objet : sur un modèle polyédrique (a), une image (b), une image des contours (c) obtenue par filtrage de l'image (b).

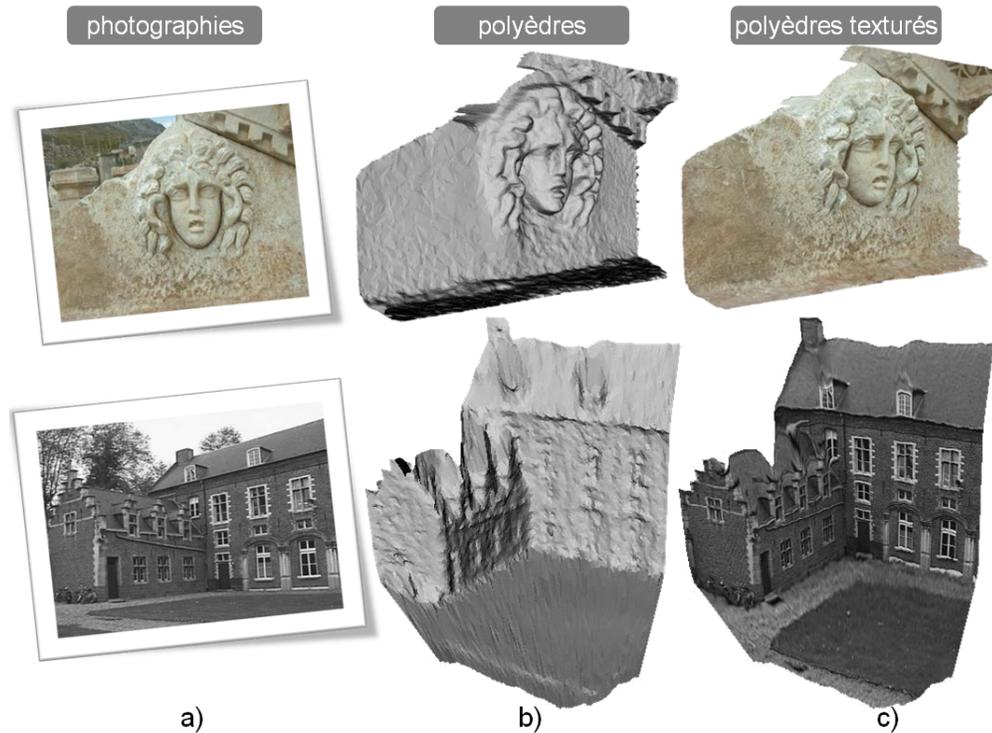


FIGURE 3 – Utilisation des images numériques (a) pour le placage de textures (c) sur des modèles polyédriques 3D (b) d'un bas relief (en haut) et des façades d'édifices (en bas) *courtoisie de [Pollefeys et al. 2004]*.

En tant que source de données permettant d'inférer des **informations tridimensionnelles**, les images numériques doivent être davantage exploitées afin de dépasser le simple stade de l'enrichissement de la maquette numérique, comme c'est le cas en architecture avec le placage de textures par exemple (fig.3). En effet, les informations contenues dans les images devraient être utilisées, non pas à des fins restreintes de rendu réaliste, mais pour le **pilotage direct des processus de manipulation de modèles géométriques**.

Que ce soit pour la simplification, la segmentation, la déformation, ou la mise en conformité des modèles géométriques, les techniques développées sont très souvent guidées par des **critères géométriques** liés par exemple à la notion de forme et dont l'évaluation nécessite le calcul de quantités discrètes (courbure, distance euclidienne, distance géodésique ...) sur l'**intégralité** des modèles. Afin d'éviter ces calculs, nous proposons ici d'utiliser des critères extraits non plus de la géométrie 3D du modèle, mais directement des images numériques.

En effet, alors que le nombre de *pixels*⁸ utilisés pour le codage des images explose (pouvant aller jusqu'à la dizaine de millions), et donc avec lui la quantité d'informations contenue, le stockage en mémoire des images numériques (de l'ordre de quelques Mo⁹) demeurera toujours moins volumineux que celui des modèles géométriques (plusieurs centaines de Mo).

La simplification de modèles polyédriques lourds en conception mécanique est une étape nécessaire pour permettre leur visualisation et leur manipulation. Les méthodes actuelles, fournissent des modèles simplifiés, plus simples et plus légers puisque contenant moins de faces. Dans la plupart des méthodes, les modèles résultants sont contraints à ressembler le plus fidèlement possible au modèle initial.

Cependant, ces techniques se préoccupent davantage d'obtenir des modèles fidèles "géométriquement" que "visuellement". En général, le processus de simplification est guidé par un critère lié à la distance géométrique entre le modèle simplifié et le modèle initial. Pourtant, bien que l'apparence d'un modèle soit en grande partie dictée par sa géométrie, d'autres sources d'informations peuvent être utilisées pour la décrire.

En effet, dans les phases amont d'un projet de lancement d'un nouveau produit, sa conception s'amorce en général par le biais d'esquisses 2D [Catalano *et al.* 2002, Pernot 2004] (fig. 4.a et fig. 4.b) qui traduisent la forme globale du produit.

Ces esquisses rassemblent les courbes structurelles comme les profils et les sections et plus globalement, toutes les courbes qui influent significativement sur l'apparence de l'objet, comme les lignes de caractère. Or, les contours dans les photographies filtrées, correspondent aussi aux lignes de caractère des objets représentés.

Contrôler l'étape de simplification directement à partir d'un critère ou d'une métrique lié aux informations contenues dans les images 2D est donc un axe de recherche à développer.

8. picture elements : nom donné aux points dont est composée une image numérique

9. Mo : méga octets

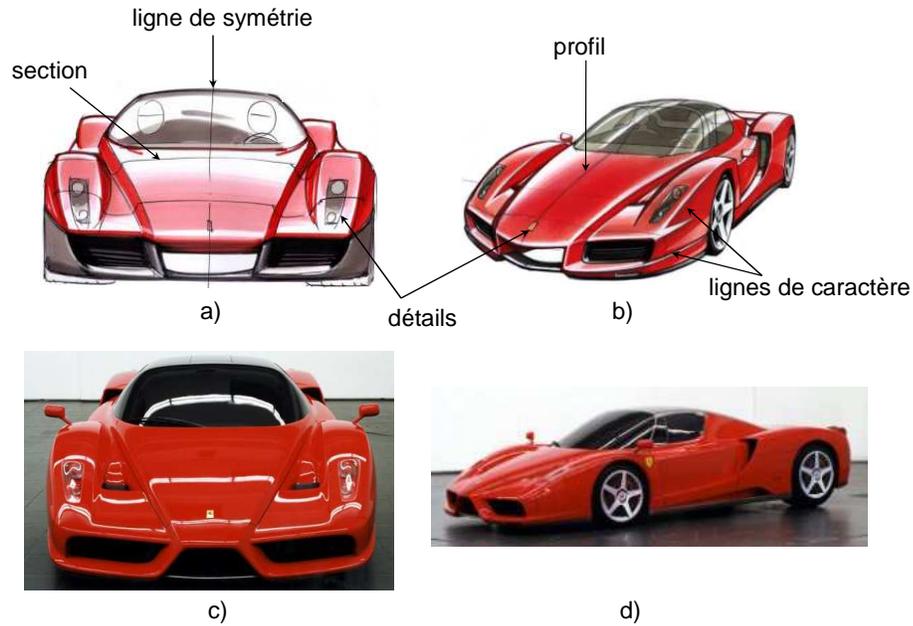


FIGURE 4 – Esquisses 2D tracées par les concepteurs (a,b) pour représenter un produit (c,d) *courtoisie du projet FIORES II, Character Preservation and Modelling in Aesthetic and Engineering Design [Pernot 2004]*.

De plus, l'acquisition des photographies avec les appareils numériques actuels se fait de manière de plus en plus rapide et aisée. Dans le cas précis de la manipulation de modèles incomplets obtenus à partir d'une numérisation de la surface de l'objet physique, suivant sa complexité, certaines zones ne pourront être atteintes alors qu'une photographie peut plus facilement être prise (fig.5). Une étape de remplissage de trous dans le modèle résultant de cette numérisation sera donc nécessaire.

En effet, la figure 5 montre le modèle de l'amphithéâtre de l'École d'Arts et Métiers ParisTech d'Aix en Provence (fig. 5.a.) relevé par un balayage au scanner laser à partir d'une station au sol : certaines zones du toit n'ont pu être atteintes par le laser (fig. 5.b.) alors qu'un hélicoptère radio commandé (fig. 5.c.) disposant d'un appareil photographique embarqué (fig. 5.d.) et géo-référencé (fig. 5.c.) peut facilement prendre des photographies de cette zone. Les photographies ainsi prises contiendraient des informations décrivant les formes manquantes sur le modèle.

Permettre la reconstruction des formes initiales en pilotant l'étape de remplissage de trous avec des informations ou des contraintes extraites d'images 2D est donc également un champ de recherche à investiguer.

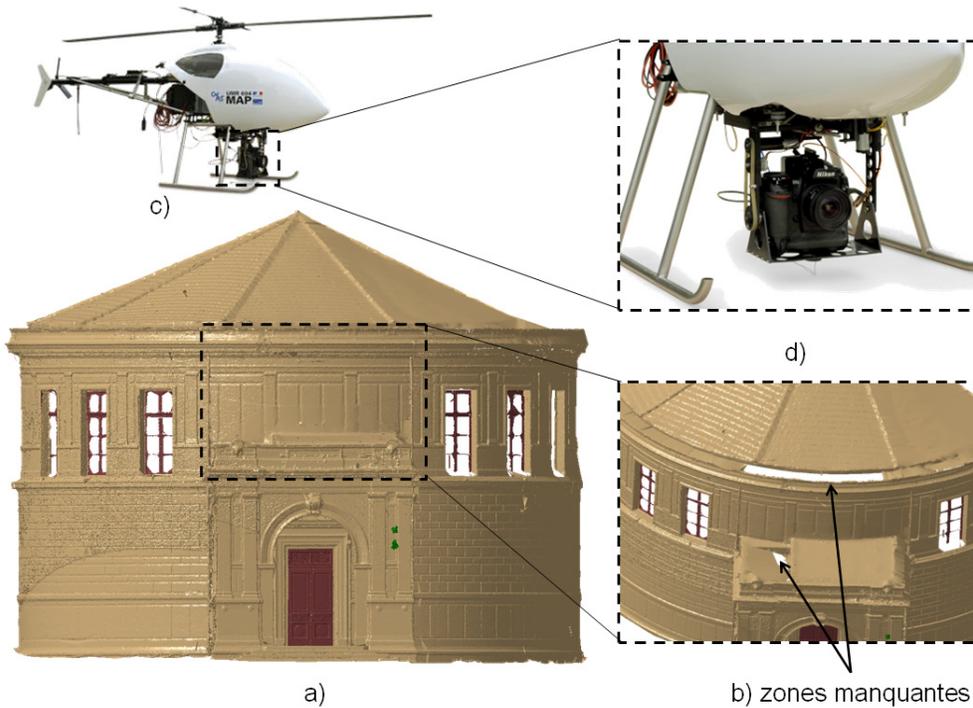


FIGURE 5 – Modèle scanné de l’amphithéâtre de l’École d’Arts et Métiers ParisTech d’Aix en Provence (a) comportant des trous (b). Une caméra géoréférencée et embarquée (d) dans un hélicoptère radio commandé (c) (*courtoisie* du laboratoire UMR CNRS/MCC 694 MAP - Gamsau) peut prendre des photographies des zones invisibles depuis le sol.

Au vu de ces diverses considérations, et en amont des travaux présentés ici, l’exploitation d’images pour le traitement et la manipulation des modèles 3D est prometteuse. Les champs d’application de l’approche proposée ont donc porté essentiellement sur le cas des modèles polyédriques ou maillages 3D et sur le développement de techniques mixtes, couplant images et modèles polyédriques, dédiées à la simplification de modèles lourds et à la déformation dans le cadre du remplissage de trous dans les modèles incomplets. Ce document est organisé de la façon suivante :

La première partie présente un état de l’art des différentes représentations possibles et de leurs méthodes de manipulation :

- les modèles géométriques 2D et 3D, les images (chapitre 1)
- les techniques de traitement et d’exploitation d’images (chapitre 2)
- les techniques de manipulation de modèles polyédriques : simplification et déformation (chapitre 3).

La deuxième partie est dédiée à la nouvelle approche développée :

- la méthode de simplification de modèles polyédriques pilotée à partir d'informations extraites d'images est décrite au chapitre 4,
- enfin deux méthodes de déformation de modèles polyédriques sous contraintes extraites d'images et appliquées au remplissage de trous sont présentées (chapitre 5).

- Partie I -

Représentations numériques et techniques de traitement associées : état de l'art

La première partie du présent mémoire propose une présentation générale des différents modèles géométriques et des différents types d'images qui peuvent potentiellement être présents au sein des maquettes numériques. Les différentes techniques d'exploitation d'images et de manipulation de polyèdres existantes sont aussi analysées.

Le premier chapitre débute par la description des modèles géométriques fréquemment utilisés tout au long du processus de conception de produits.

Le second chapitre présente les méthodes d'exploitation et de traitement d'images.

Enfin, le troisième chapitre est consacré aux méthodes de manipulation et de traitement des modèles polyédriques dans un contexte de développement numérique de produits.

Chapitre 1

Représentations numériques multiples

Ce chapitre synthétise les différents modèles géométriques et représentations qui peuvent être utilisés au cours des processus de conception de produits manufacturés. Ils sont tout d'abord classés selon leur dimension dans une première partie. Puis, les entités géométriques associées aux modèles B-Rep qui sont parmi les plus utilisées en conception sont présentées, ainsi que les types d'images numériques existants.

1.1 Une variété de modèles géométriques et autres représentations

On peut regrouper en différentes catégories les modèles géométriques selon leur dimension :

- **les éléments ponctuels** (dimension 0) sont les entités de base les plus simples. Ils servent à la construction d'éléments plus complexes comme les éléments linéiques et surfaciques.
- **les éléments linéiques** (dimension 1) correspondent aux segments de droites ou aux courbes. Ils sont utilisés pour la représentation filaire d'un objet ou servent à la définition d'éléments plus complexes comme les éléments surfaciques et volumiques.
- **les éléments surfaciques** (dimension 2) sont le plus souvent utilisés pour décrire des formes "lisses". On peut les regrouper en trois catégories :
 - **les surfaces implicites** [Bloomenthal *et al.* 1997, Cani 2007] qui servent à représenter des formes par une fonction implicite telle que (fig. 1.1) :

$$f(x, y, z) = 0. \tag{1.1}$$

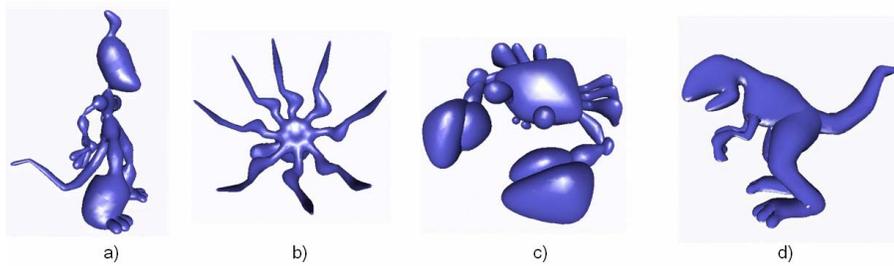


FIGURE 1.1 – Exemples de surfaces implicites [Angelidis 2001].

En ingénierie mécanique, ces fonctions correspondent à des primitives simples pour la définition de plans, cylindres, sphères (mais ne conviennent pas pour représenter les arêtes vives des objets) alors que dans le domaine de l’animation 3D, elles sont combinées pour représenter des formes plus complexes. Les arêtes vives, très présentes en conception de produits ne sont pas facilement modélisables avec ce genre de représentations.

- **les surfaces paramétriques** qui servent à représenter tout type de surfaces (plan, cylindre, surfaces gauches, etc...) grâce à l’utilisation de carreaux de Bézier [Bézier 1987] B-Spline et NURBS¹ [Léon 1991]. La définition paramétrique de la surface s’exprime à l’aide d’une équation dépendant de deux paramètres u et v et telle que :

$$\forall (u, v) \in \mathbb{N}_u \times \mathbb{N}_v \quad \mathbf{S}(u, v) = \frac{\sum_{i=0}^m \sum_{j=0}^n h_{ij} \cdot N_{ip}(u) \cdot N_{jq}(v) \cdot \mathbf{s}_{ij}}{\sum_{i=0}^m \sum_{j=0}^n h_{ij} \cdot N_{ip}(u) \cdot N_{jq}(v)} \quad (1.2)$$

avec $\mathbb{N}_u = [u_0, u_{m+p+1}]$ et $\mathbb{N}_v = [v_0, v_{m+p+1}]$ et où les (\mathbf{s}_{ij}) sont les points de contrôle, h_{ij} , les poids et (\mathbb{N}_u) , (\mathbb{N}_v) , les séquences nodales en u et v respectivement.

- **les surfaces de subdivision** [Sabin 2004] sont très utilisées dans le domaine de l’animation 3D et le deviennent de plus en plus au sein des applications CAO. Elles sont faciles à implémenter et ne requièrent que peu de mémoire pour leur stockage. Les sommets du maillage (triangulaire [Catmull *et al.* 1978] ou quadrangulaire [Loop 1987] ou mixte [Stam *et al.* 2003]) au niveau de détail $k+1$ sont donnés par l’équation :

$$P_i^{[k+1]} = SP_i^{[k]} \quad (1.3)$$

avec $P_i^{[k]}$ l’ensemble des sommets du maillage au niveau de détail k

1. Non Uniform Rational B-Spline

et S la matrice qui définit le schéma de subdivision caractérisé par les propriétés de convergence et de lissage de la surface (fig.1.2).

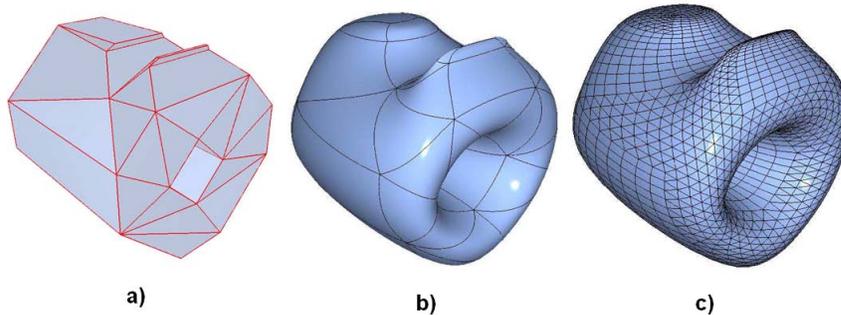


FIGURE 1.2 – Surfaces de subdivision : le maillage initial (a), après une première itération (b), et une deuxième itération du schéma de Loop [Stam *et al.* 2003] (c).

Les représentations surfaciques non fermées, ne possèdent pas suffisamment d'informations relatives à la notion d'intérieur/extérieur d'un solide puisqu'elles ne définissent que sa surface limite.

- **les éléments volumiques** (dimension 3), en revanche, intègrent la notion de topologie avec les conditions inhérentes de fermeture, d'orientation des surfaces bornant le domaine volumique et de non auto-intersection des surfaces.

Deux approches sont classiquement utilisées et donnent lieu à deux catégories de modèles :

- **les modèles CSG**² [Kumar *et al.* 1995] qui se basent sur des opérations topologiques élémentaires (union, soustraction, intersection...) appliquées à des primitives de base (plans, cylindres, sphères, solides obtenus par extrusion ...)(fig.1.3). La séquence d'opérations correspondante est appelée arbre CSG. Les principaux avantages des modèles CSG sont :

- qu'ils sont basés sur des structures de données simples,
- qu'ils ne requièrent que peu d'espace pour leur stockage (car seul l'arbre de conception est sauvegardé),
- qu'ils permettent une conception rapide et intuitive par le biais d'un paramétrage simple des primitives de base.

Leurs inconvénients majeurs résident dans le fait :

- qu'il n'y a pas unicité de l'arbre de conception,
- que les primitives disponibles ne permettent pas toujours de modéliser des formes gauches complexes,

- que la mise en jour d'une partie du modèle entraîne le recalcul entier de l'arbre,
- que pour être visualisé, un modèle CSG doit être converti en modèle polyédrique.

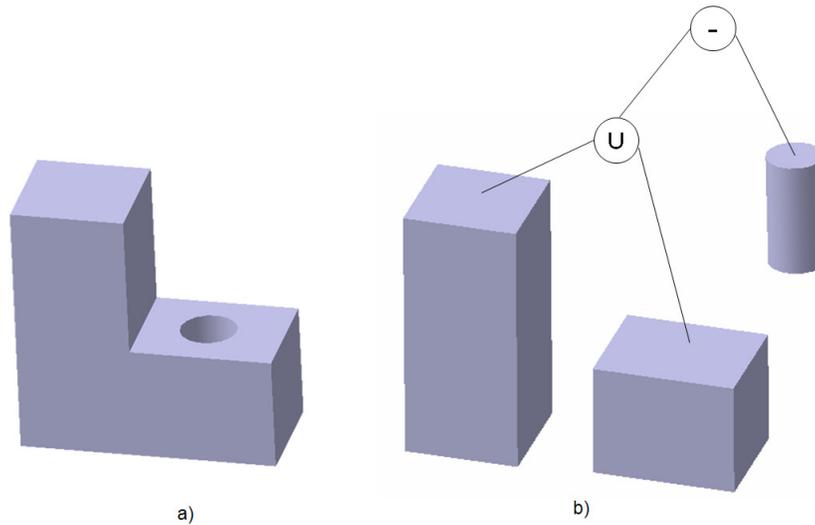


FIGURE 1.3 – Le solide (a) et un exemple d'arbre CSG servant à le modéliser (b) : \cup est l'union, et $-$ la différence

- les **modèles B-Rep**³ [Van et al. 1991] définissent un solide par l'ensemble fermé et orienté des surfaces (fig.1.4) représentant sa frontière (sa "peau"). La notion d'intérieur/extérieur est donc possible. Chaque face est limitée par ses contours bornés par leurs sommets et arêtes.

Les principaux avantages des modèles B-Rep sont :

- leur facilité de visualisation (car ils sont aisément facettisés),
- les modifications locales qu'ils permettent.

Leurs inconvénients majeurs résident dans le fait :

- qu'ils nécessitent une grande quantité d'informations (faces, arêtes, sommets),
- qu'ils ne représentent que la surface du solide.

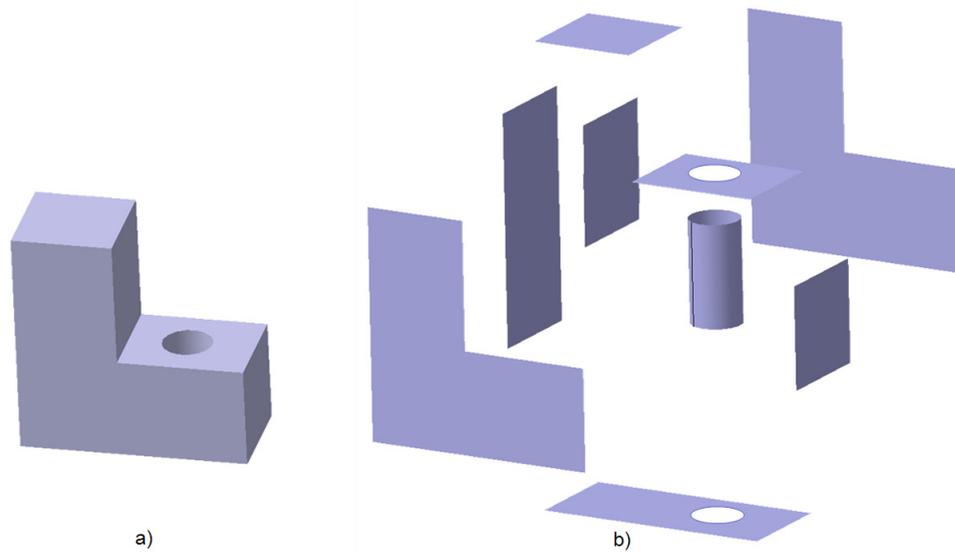


FIGURE 1.4 – Le solide (a) et la représentation B-Rep servant à le modéliser (b).

Le travail présenté ici s'intéresse plus particulièrement aux cas des modèles polyédriques ou maillages 3D, qui permettent la représentation de formes gauches complexes, utilisés dans de nombreux domaines tels la réalité virtuelle, l'architecture, la conception mécanique... Lorsqu'ils sont fermés, orientés et sans auto-intersection, les modèles polyédriques sont des modèles B-Rep. Ils sont détaillés en section 1.4.

D'autres entités géométriques associées aux modèles B-Rep sont brièvement présentées dans les sections suivantes. Elles entrent en jeu dans différentes tâches du processus de conception (*Reverse Engineering*⁴, activités de design, analyses mécaniques, etc.). Elles peuvent être différenciées selon leur degré de dérivabilité : modèles continûment dérivables (courbes et surfaces paramétriques, section 1.2), modèles discrets (nuages de points, section 1.3), modèles continûment dérivables par morceaux (polyèdres ou maillages, section 1.4).

Puis deux représentations de formes particulières que sont les images (section 1.5) et les représentations symboliques (section 1.6) sont également abordées. La définition d'un modèle hybride mixant deux représentations (polyèdre et image) est présentée.

4. Conception Inverse

1.2 Courbes et surfaces paramétriques

Les courbes et surfaces paramétriques utilisent une description à l'aide de formes polynomiales définies dans un espace bi-paramétrique (modèles de Bézier (fig.1.5), B-Spline et NURBS). Ces modèles permettent de définir, selon les contraintes fonctionnelles, une représentation continue de la surface d'un objet [Léon 1991]. La difficulté majeure réside dans l'établissement et le maintien des relations de connectivité et des conditions de continuité entre les différents carreaux.

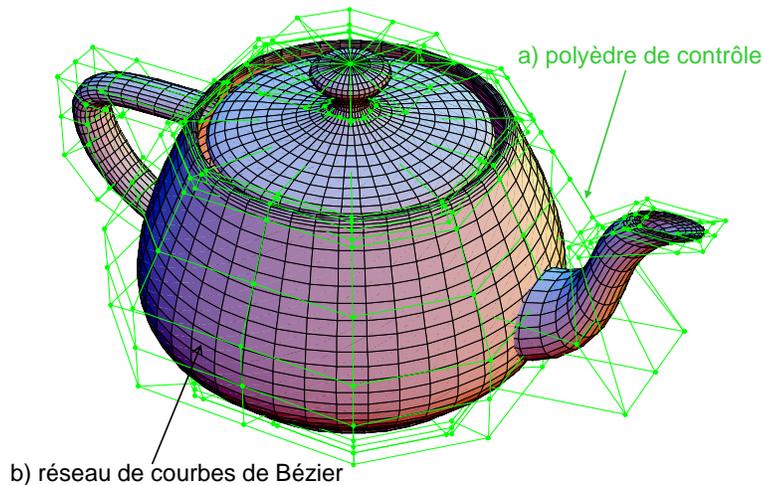


FIGURE 1.5 – Un exemple de surface de Bézier : le modèle de la théière (*Utah teapot*) formée de carreaux définis par des réseaux caractéristiques constitués de points de contrôle.

1.3 Nuages de points

En *Reverse Engineering*, le processus de construction de la maquette numérique d'un objet physique commence par l'acquisition d'un nuage de points (fig. 1.6.a.) représentatif de sa peau [Varady *et al.* 1997]. Un maillage triangulaire est ensuite créé à partir de cet ensemble non structuré de points (fig. 1.6.b). Des surfaces B-Spline/NURBS ou des surfaces de subdivision peuvent également être créées par approximation du nuage de points.

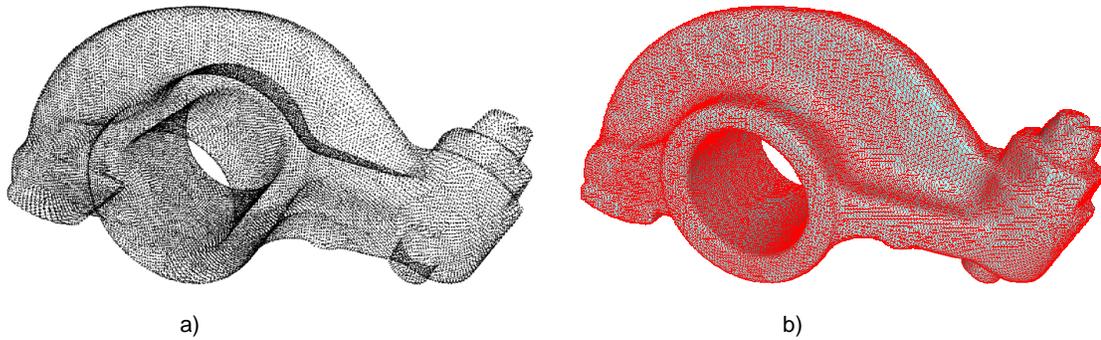


FIGURE 1.6 – Un nuage de points (a) acquis par balayage scanner d'un culbuteur (pièce de moteur thermique); un polyèdre (ou maillage) (b) du même culbuteur (80354 faces).

Le balayage au scanner laser consiste à projeter un rayon laser sur la surface de l'objet. Les trois techniques les plus utilisées sont :

- l'acquisition par **triangulation**, où le capteur est séparé de la station émettrice par une distance connue, créant ainsi une configuration en "triangle" avec l'objet scanné. La précision de cette technique dépend donc de la distance entre l'objet et la station de base et de la distance entre la source émettrice et le capteur. De plus, la station de base étant fixe, plus l'objet est proche ($<$ à 25 m), plus la précision est grande.
- l'acquisition par **différence de phase** procède par émission d'un signal modulé en phase, la différence de phase entre l'onde émise et le signal réfléchi permet de localiser la position de la surface de l'objet qui réfléchit le rayon.
- l'acquisition par **temps de vol** procède par mesure du temps que met le rayon pour parcourir le trajet de la source émettrice à la surface de l'objet, puis de la surface de l'objet au récepteur. La source émettrice et le récepteur sont placés à distance identique de la surface de l'objet. Ces deux dernières méthodes ont une précision meilleure pour des distances plus longues.

On parle de nuages de points colorés (fig. 1.7) lorsque chaque point 3D du nuage est coloré à partir de son pixel correspondant dans une image recalée dans le même repère de référence lié au nuage de points [De Luca 2006].



FIGURE 1.7 – Un nuage de points colorés [De Luca 2006].

1.4 Modèles polyédriques

Les modèles polyédriques (fig. 1.6.b.) sont constitués par un ensemble de faces, le plus souvent triangulaires, définies par des arêtes et sommets. Ces modèles sont particulièrement robustes grâce aux relations de connectivité simples établies entre les différentes faces. En effet, les approximations liées à la définition de courbes frontières pour les modèles paramétriques sont évitées, car la frontière entre deux faces voisines d'un modèle polyédrique est définie de manière exacte par une arête rectiligne. De plus, les faces sont triangulaires et possèdent donc toutes les propriétés relatives aux espaces plans bornés ainsi que la propriété fondamentale de convexité.

Les modèles polyédriques sont également nécessaires pour l'analyse des structures par éléments finis qui requièrent une discrétisation (sous la forme de maillage) de la surface des modèles manipulés. Dans ce cas, les triangles doivent satisfaire un certain nombre de critères de forme (ratio d'aspect, etc.). Ils permettent aussi le calcul de trajectoires pour l'usinage de pièces mécaniques par les *MOCN*⁵.

Enfin, la visualisation des modèles 3D de quelque nature qu'ils soient, exigent leur facettisation en modèles polyédriques : en effet, malgré leurs performances accrues en terme de capacité, le triangle demeure la seule primitive pouvant être tracée par les cartes graphiques.

5. Machines Outils à Commande Numérique

Deux aspects majeurs sont à prendre en compte lors de la manipulation des modèles polyédriques.

- Le premier concerne tout d'abord le nombre de faces nécessaire à la représentation d'une surface complexe. Compte tenu de la simplicité des entités géométriques utilisées, la description de ces surfaces entraîne l'utilisation d'un degré de détails important, c'est-à-dire d'un nombre important de faces. Pourtant dans le cadre de la conception collaborative, de nombreux échanges via Internet par exemple, sont nécessaires. Pour ce faire, ils exigent une adaptation de la maquette. La visualisation, le stockage, l'indexation et l'exploitation des modèles requièrent aussi une simplification des surfaces qui les composent. En effet, de nombreux modèles polyédriques avec un nombre très élevé (plusieurs millions) de faces comme celui d'une pale de turbine (fig. 1.8.a) ainsi que des images 2D (fig. 1.8.b) représentant le même produit sont utilisés au sein de la maquette numérique d'un produit.

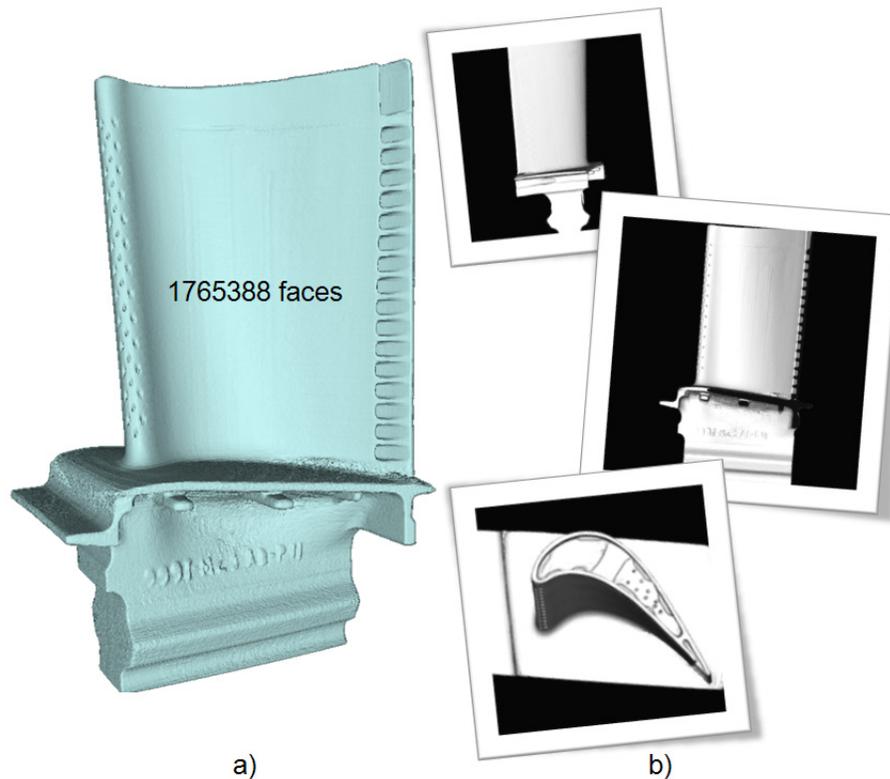


FIGURE 1.8 – Un modèle polyédrique lourd (a) d'une pale de turbine (*courtoisie de la Georgia Tech 3D large models archive*) et différentes images 2D du modèle (b).

- Le second critère est relatif au nombre de faces connectées à une arête de la surface d'un objet volumique (étanche). En effet, la surface ne doit pas comporter de trous comme cela peut se produire avec des modèles obtenus par triangulation d'un nuage de points acquis par balayage laser de la surface de l'objet physique. Certaines parties de la surface sont inaccessibles ce qui produit des zones manquantes dans le nuage de points puis des trous dans le modèle polyédrique reconstruit. Il existe donc également un besoin en remplissage de trous dans ces modèles. En effet, malgré 480 balayages laser, on note la présence de trous (fig. 1.9.e) dans le modèle polyédrique du "David de Michelange" (fig. 1.9.d) qui comporte près de 2 millions de faces. Parallèlement, 7000 photographies de la statue (fig. 1.9.c) ont été prises durant la session de scans (fig. 1.9.a et fig. 1.9.b).

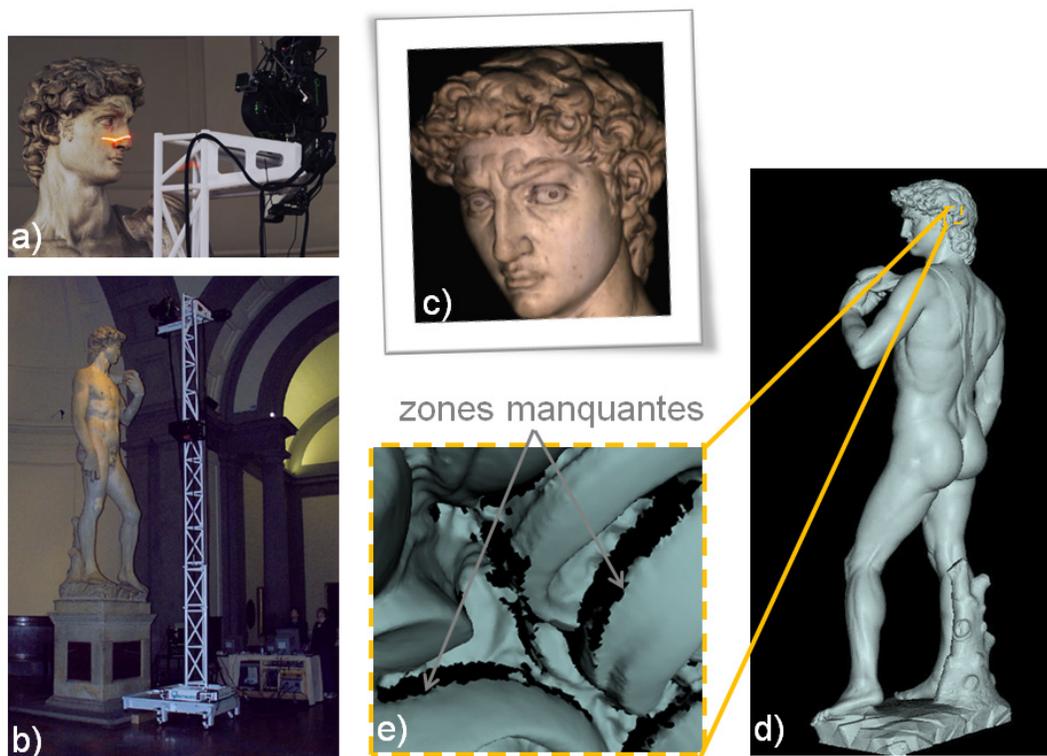


FIGURE 1.9 – Un modèle polyédrique (d) de la statue "le David de Michelange" a été obtenu par balayage au scanner laser (a) et (b) (*courtoisie de la Digital Michelangelo Project Archive*). Malgré plusieurs sessions de scans pris sous différents angles, une partie des cheveux du modèle comporte des trous (e).

1.5 Images numériques

En vision par ordinateur, les images numériques correspondent à la projection 2D sur la rétine des objets de la scène 3D observée. Les informations qu'elles contiennent sont donc semblables à celles traitées par le cerveau pour en extraire des caractéristiques 3D. Elles représentent donc une source de données qui participent à la perception humaine des formes 3D [Todd 2004]. Les informations de textures, de contours, de régions, de lignes de caractère, etc. contenues dans les images sont autant d'entités à deux dimensions caractérisant des formes tridimensionnelles.

Les images numériques peuvent être regroupées en deux catégories : les images matricielles et les images vectorielles.

1.5.1 Images matricielles

Les images matricielles (*bitmap* ou *raster* en anglais) sont des images numériques qui stockent les informations sous la forme d'un tableau ou matrice de pixels (fig. 1.10).

Images 2D

Dans le cas des images à deux dimensions, on considère l'image comme une fonction de \mathbb{R}^2 telle que pour un pixel de coordonnées (x,y) dans l'image, $I(x,y)$ est l'intensité lumineuse entière de ce pixel.

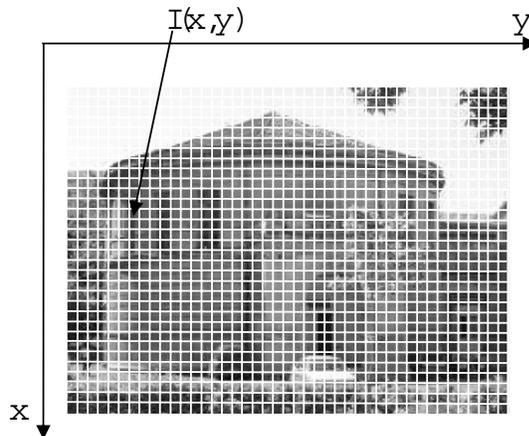


FIGURE 1.10 – Une image numérique stocke les intensités lumineuses des pixels dans un tableau.

On distingue deux modes de codage de l'intensité lumineuse : le mode en *niveaux de gris* et le mode *RVB*⁶.

Dans les images en mode RVB, la couleur d'un pixel est codée sur trois canaux différents, Rouge, Vert, Bleu. L'intensité lumineuse se calcule à partir d'une combinaison linéaire de ses trois composantes (section 2.2.1). Dans les images en niveaux de gris, la valeur d'un pixel est codée sur un seul canal et correspond à son intensité lumineuse.

Une image numérique 2D est une approximation de l'image réelle formée par la projection de la scène 3D sur le plan image (plan correspondant à la surface photosensible du capteur ayant acquis l'image). La qualité d'une image numérique dépend d'une part de sa résolution, i.e. du nombre de pixels dans l'image, ainsi que de la quantification de l'intensité lumineuse, i.e. du nombre de bits utilisés. Pour des images en niveaux de gris, l'intensité d'un pixel est codée sur un octet et il y a donc $2^{m=8bits} = 256$ nuances de gris possibles. Pour des images RVB, l'intensité est codée sur trois octets et il y a donc $2^{m=3*8=24bits} = 16777216$ couleurs possibles.

Les captures d'écran

Une capture d'écran (ou *screenshot*) est une image matricielle dont le contenu est celui de la scène qui a été affichée à un instant t sur un écran d'ordinateur. Dans le cas de la visualisation de modèles 3D, cette image est donc semblable à une photographie de la scène prise depuis la caméra (le point de vue de l'utilisateur dans le visualiseur 3D). La capture d'image est donc déjà référencée par l'ordinateur dans le même repère que le modèle géométrique lié à la scène.

Images 3D

On parle d'images 3D [De Luca 2006] lorsque l'on ajoute une troisième dimension aux images 2D, le plus souvent la troisième coordonnée indique une notion de *profondeur* (fig.1.11), et les pixels deviennent des *voxels* puisqu'ils forment un volume. En animation, la troisième composante ajoutée aux images 2D étant le *temps*, on obtient alors des séquences vidéo.

6. Rouge Vert Bleu ; RGB, Red, Green, Blue en anglais



FIGURE 1.11 – Exemple d’une image 3D ou carte de profondeur [De Luca 2006] calculée à partir d’un nuage de points : le niveau de gris d’un pixel est calculé en fonction de la distance métrique du point 3D correspondant dans le nuage de points, par rapport à un plan de référence.

Images stéréoscopiques

On appelle *images stéréoscopiques* deux images matricielles (fig. 1.5.1) représentant la même scène mais prises de deux points de vue légèrement différents afin de simuler la vision stéréoscopique humaine. Pour voir cette vision stéréoscopique, il suffit de loucher de façon à former une troisième image virtuelle, puis de patienter jusqu’à l’accommodation en se concentrant sur cette troisième image et le cerveau restituera la notion de relief. Un certain type de format d’images permet de stocker les intensités lumineuses des deux vues juxtaposées dans un même fichier (d’extensions .jps pour *jpeg stereoscopic*).

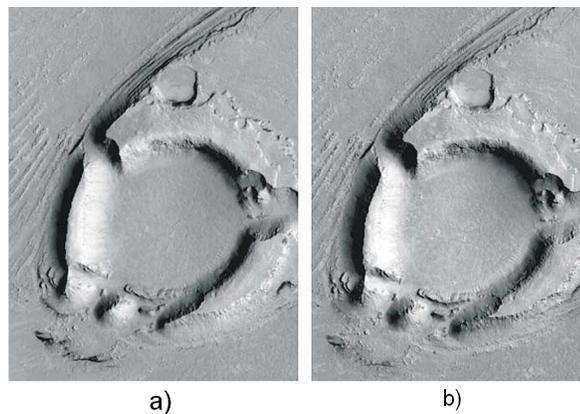


FIGURE 1.12 – Exemple de deux images stéréoscopiques de cratères martiens, vues de gauche (a) et de droite (b) (*courtoisie de la NASA et Malin Space Science Systems*).

1.5.2 Images vectorielles

Les images vectorielles [Michel-Duthel 2001], contrairement aux images matricielles, contiennent les primitives de dessin (formes, position, couleurs...) des objets géométriques qu'elles représentent (segments de droite, polygones, arcs de cercles...). Les inconvénients des images vectorielles sont qu'à chaque visualisation, elles sont converties en images matricielles (seules affichables sur un écran d'ordinateur) et qu'elles nécessitent des temps de calcul importants. La principale particularité des images vectorielles réside dans le fait que l'on peut les redimensionner sans limitation de résolution alors que cela entraîne des pertes de qualité avec des images matricielles (fig.1.13).



FIGURE 1.13 – Comparaisons de deux images numériques : le zoom dans l'image matricielle (a) fait apparaître les pixels qui la constituent et produit un effet de crénelage inexistant dans l'image vectorielle (b). *courtoisie de* <http://www.jeffkee.com/>

1.6 Représentations symboliques ou schématiques

Un type de représentation autre que les modèles géométriques consiste à représenter une forme ou un composant (liaisons mécaniques, composants électriques, hydrauliques...) par un symbole, une icône par exemple. Ainsi, une soudure en "V" peut être représentée par un schéma simple (fig.1.14.a) ou par un symbole (fig.1.14.b).

En dessin industriel [Boudet 1980], la lecture des plans d'ensemble de certains mécanismes (fig. 1.15.a) n'est pas toujours triviale. On utilise une représentation simplifiée et normalisée appelée schéma cinématique (fig. 1.15.b). Les groupes de pièces y sont représentés par des segments de droite. A chaque contact entre les groupes de pièces, une liaison est représentée par un symbole suivant la norme ISO 3952-2 :1981.

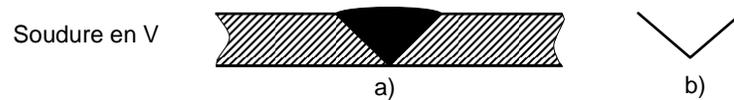


FIGURE 1.14 – Exemple de représentation symbolique d'un cordon de soudure : à gauche, une représentation simplifiée (a), à droite le symbole associé (b).

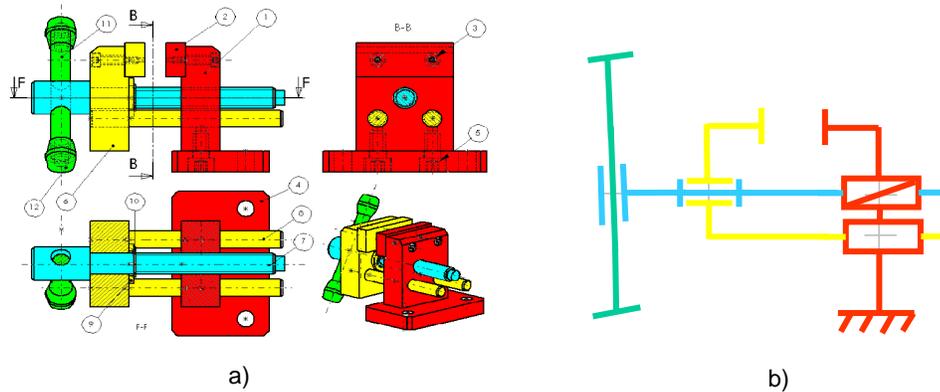


FIGURE 1.15 – Plan d'ensemble d'un étau de modéliste avec les classes d'équivalence (groupes) des pièces colorées différemment (a) et le schéma cinématique associé (b).

De même, afin d'alléger la représentation d'un grand édifice architectural, [De Luca 2006] propose de schématiser les différents éléments qui le composent ainsi que les relations qui les lient par des symboles simples

(fig. 1.16).

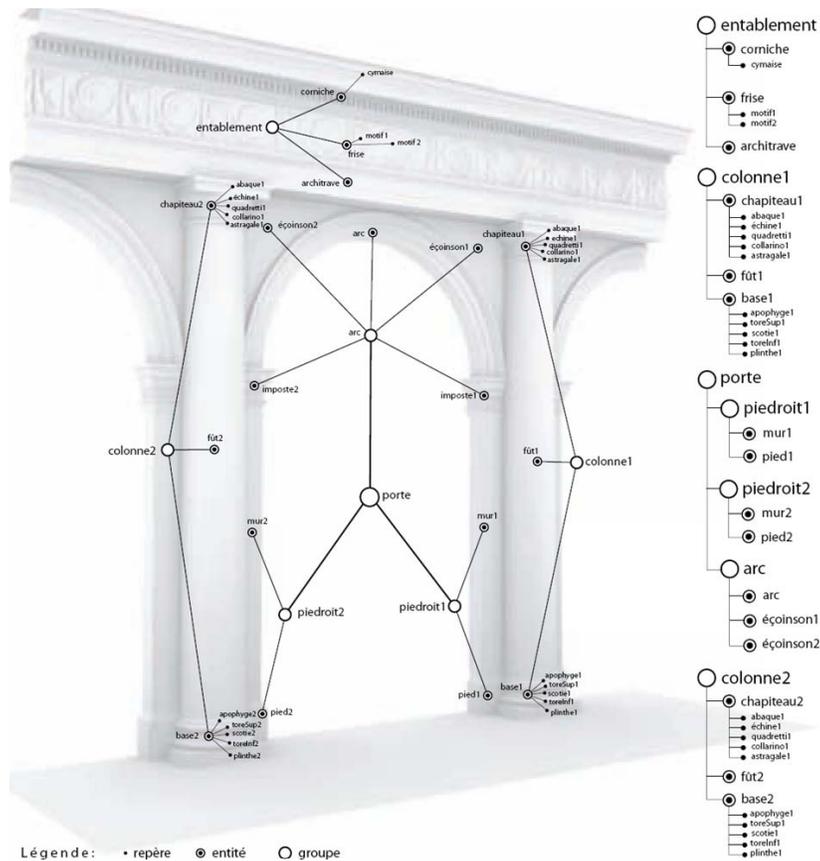


FIGURE 1.16 – Graphe de description sémantique d'un édifice [De Luca 2006] superposé sur une image du même édifice architectural : l'arborescence représente les relations hiérarchiques (partie/entité) entre les différents éléments architecturaux qui composent l'édifice.

1.7 Conclusion

Les différents types de représentations présentés précédemment possèdent des avantages et des inconvénients liés au domaine d'application qui leur est propre (tab. 1.1 et tab. 1.2). Parmi ces diverses représentations, deux types seront privilégiés :

- d'une part les modèles polyédriques qui permettent de définir les formes du produit et sont utilisés comme supports pour la majorité des mé-

- tiers intervenants dans le processus d'ingénierie du produit (e.g. design, conception, simulation, fabrication). Ils permettent entre autre, la visualisation des modèles (le rendu graphique ne pouvant se faire que sous la forme de modèles facettisés) et l'analyse par Éléments Finis qui requiert des représentations discrétisées (tab. 1.1);
- d'autre part, les images numériques qui contiennent des informations descriptives concernant les formes des objets qu'elles représentent. La détection de contours, la reconnaissance de formes, la segmentation, la mesure des propriétés de rugosité (par l'analyse de textures), sont autant de méthodes permettant d'extraire ces informations pour des applications dans de nombreux domaines, tels l'imagerie médicale, la robotique et l'architecture par exemple (tab. 1.2).

Dans le contexte de la conception mécanique et de l'architecture, l'exploitation d'informations contenues dans les images pour améliorer les techniques de manipulation de polyèdres est une activité émergente. En effet, des photographies de produits ou de bâtiments sont fréquemment prises durant les phases de conception et participent à la définition de leurs maquettes numériques. De fait, en devenant une nouvelle source de critères pouvant piloter des processus de traitement de polyèdres, les images numériques concourent à la définition d'un modèle multi-représentations associant modèles polyédriques et images numériques.

Partant de ce constat, les différents types d'informations qui peuvent être extraites des images ont tout d'abord été étudiés et analysés (chapitre 2). Ainsi, certaines méthodes de traitement d'images se sont révélées particulièrement intéressantes pour la définition du couplage polyèdre/images. Puis, l'état de l'art des principales techniques de manipulation de polyèdres (chapitre 3) a permis de mettre en exergue deux méthodes de simplification et de déformation. Ces méthodes, initialement pilotées par des critères liés à la géométrie du modèle, présentent l'avantage de pouvoir être adaptées à l'utilisation d'informations extraites d'images.

Élément de représentations	Modèles	Intérêts	Limitations
<i>ponctuel</i>	nuages de points	facilité d'acquisition [Varady <i>et al.</i> 1997]	pas de relation entre les points
<i>linéique</i>	courbes de Bézier, B-Splines, NURBS	permettent une représentation filaire du produit [Léon 1991]	<ul style="list-style-type: none"> - complément d'éléments plus complexes - la représentation filaire peut être ambiguë
<i>surfactive</i>	surfaces implicites et paramétriques	modification de la surface par la modification de l'équation [Turk <i>et al.</i> 2002]	pas de formes complexes car limitation à 1 ou 2 équations (sinon explosion des temps de modélisation)
	surfaces de Bézier, B-Splines, NURBS	modification de la surface par le biais de la manipulation des points de contrôle [Pernot 2004]	la gestion des jonctions entre les carreaux en terme de continuité n'est pas aisée
	surfaces de subdivision	<ul style="list-style-type: none"> - permettent une représentation multi-résolution - peu d'itérations suffisent pour obtenir un modèle lisse - peuvent s'appliquer à un polyèdre de topologie quelconque [Zorin 2006] 	l'augmentation importante du nombre de faces à chaque itération (qui est souvent quadruplé à chaque passe)
	modèles polyédriques	<ul style="list-style-type: none"> - adaptés à la visualisation (le triangle est la primitive capable d'être affichée par les cartes graphiques) - simples à mettre en oeuvre (par numérisation ou tessellation) - les arêtes vives sont faciles à modéliser - peuvent être volumineux pour les formes complexes [Véron 1997] 	<ul style="list-style-type: none"> - pas de représentation d'objets très lisses - approximations discrètes des calculs de normales et de courbures - grand volume de données à traiter
<i>volumique</i>	modèles CSG	faciles à concevoir par le biais des opérateurs booléens [Kumar <i>et al.</i> 1995]	respect de l'ordre chronologique de conception

TABLE 1.1 – Analyse des différents types de représentations [Badler *et al.* 1997, Hubeli *et al.* 2000].

Domaine	Application
Imagerie médicale	Téléchirurgie (ChAO)
	Aide au diagnostic
Télésurveillance et armement	Extraction automatique du relief à partir d'images satellite
	Surveillance du trafic routier
	Exploration de fonds marins
Robotique	Conduite automatique
	Suivi et préhension d'objets
Gestion de données multimedia	Indexation vidéo (ex : reconnaissance d'action dans un match de football)
	Restauration d'images (pour le patrimoine)
	Indexation d'images (requête à des bases de données)
Contrôle industriel	Détection de défaut

TABLE 1.2 – Les champs d'applications des techniques de traitement d'images.

Chapitre 2

Techniques de traitement et d'exploitation d'images

Ce chapitre rassemble un état de l'art des techniques de traitement d'images les plus fréquemment rencontrées dans la littérature. Dans une première partie, on s'intéresse aux méthodes de filtrage et d'analyse d'images dans le but d'extraire des informations descriptives (contours, formes, textures...) des objets représentés. La seconde partie porte sur le principe des méthodes de calibration de caméras qui permettent de faire le lien entre la géométrie 3D des objets et leur projection 2D dans les images. Enfin, les principales techniques de reconstruction 3D à partir d'images sont abordées.

2.1 Introduction

Les techniques de traitement d'images modifient la valeur de l'intensité des pixels des images numériques matricielles. Nous présentons ici les techniques d'**extraction d'informations** dans les images : le filtrage des contours, la reconnaissance d'objets ou la segmentation permettent d'obtenir des informations pertinentes sur les formes contenues dans les images. Notre but étant de rechercher des informations dans les images en vue de leur exploitation, nous n'aborderons pas les méthodes d'amélioration, de restauration et de compression d'images dont les objectifs sont d'améliorer la qualité des images ou de réduire la quantité de mémoire requise pour leur stockage.

2.2 Techniques de traitement d'images

2.2.1 Extraction de contours

Soient (x, y) les coordonnées d'un pixel dans le repère lié à l'image, $I(x, y)$ est la valeur de l'intensité lumineuse de ce pixel. On peut convertir une

image couleurs en une image en niveaux de gris en calculant une moyenne pondérée de ces trois composantes RVB comme proposé par la C.I.E¹ et en affectant cette valeur moyenne à chacune des trois composantes. Les images en niveaux de gris sont les images utilisées en traitement d'images car elles suffisent pour extraire des informations importantes. Par exemple, l'application de filtres de contours sur une image en couleurs multipliera par trois les temps de calcul puisque l'opération s'effectue sur chacune des trois composantes. On a donc utilisé des images en niveaux de gris dans les travaux présentés ici.

Dans une image supposée continue, un contour correspond à une forte variation de l'intensité lumineuse $I(x,y)$. Soit \mathbf{G} le gradient de I :

$$\mathbf{G} = \nabla I = \begin{bmatrix} \frac{\partial I}{\partial x}(x, y) \\ \frac{\partial I}{\partial y}(x, y) \end{bmatrix} \quad (2.1)$$

Un contour est défini comme la localisation des maxima du gradient. D'autres grandeurs différentielles peuvent être utilisées comme le *laplacien* de la fonction I . On recherche donc les extrema locaux du gradient (fig. 2.2.c.) avec les filtres de Prewitt [Prewitt 1970], Sobel [Sobel 1990] (fig. 2.1), Canny [Canny 1986] et Deriche [Deriche *et al.* 1993]) ou les passages par zéro du laplacien avec les filtres de Marr-Hildreth [Marr *et al.* 1980], [Roushdy 2007] (fig. 2.2.d.).

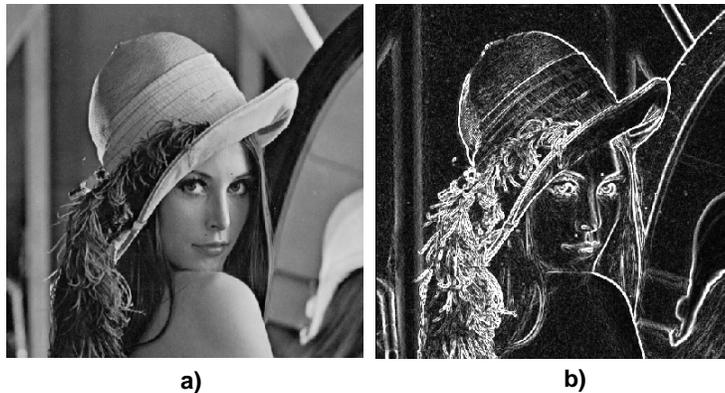


FIGURE 2.1 – L'image originale à gauche a) et l'image filtrée (l'image de contours) avec le filtre de Sobel [Sobel 1990] b) appliqué sur l'image de test standard *Lenna* (<http://www.lenna.org>).

La difficulté d'implémentation des méthodes à base des opérateurs gradient et laplacien réside dans l'approximation des grandeurs dans l'espace

1. Commission Internationale de l'Éclairage

discret des images numériques. Les approximations des dérivées se font par différences finies calculées par convolution.

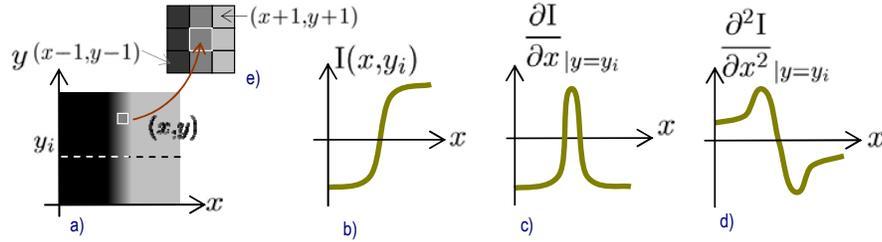


FIGURE 2.2 – Un contour dans une image (a), la fonction d'intensité correspondante (b) et l'évolution des dérivées partielles première (c) et seconde (d) de l'intensité par rapport à la coordonnée x . La zone d'influence d'un pixel (x, y) d'un filtre approché par une matrice de dimension 3×3 s'étend à ses huit pixels adjacents (e).

La convolution de la fonction intensité définie pour chaque pixel (x, y) de l'image, avec un filtre approché par exemple par une matrice F de dimension 3×3 , est définie ainsi :

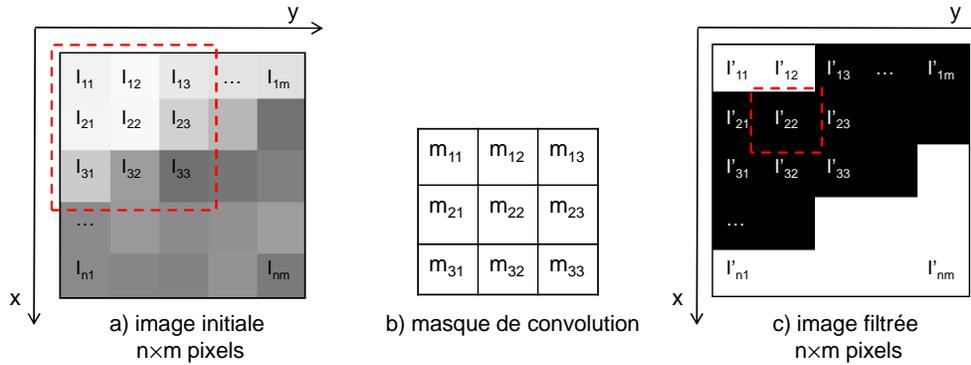
$$\text{conv}(I, F)(x, y) = \sum_{i=0}^2 \sum_{j=0}^2 I(x-1+i, y-1+j) \cdot F(i, j) \quad (2.2)$$

Le filtre de Sobel détecte en deux temps les changements d'intensité lumineuse horizontaux et verticaux. La magnitude ainsi que les directions de ces variations sont calculées par les masques suivants :

$$I_{Sobel}(x, y) = \sqrt{(\text{conv}(I_S, H)(x, y))^2 + (\text{conv}(I_S, V)(x, y))^2} \quad (2.3)$$

$$f_{Sobel}(x, y) = \tan^{-1}(\text{conv}(I, V)(x, y) / \text{conv}(I, H)(x, y)) \quad (2.4)$$

$$\text{avec } H = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix} \text{ et } V = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$$



$$l'_{22} = (l_{11} * m_{11}) + (l_{12} * m_{12}) + (l_{13} * m_{13}) + (l_{21} * m_{21}) + (l_{22} * m_{22}) + (l_{23} * m_{23}) + (l_{31} * m_{31}) + (l_{32} * m_{32}) + (l_{33} * m_{33})$$

FIGURE 2.3 – Convolution d'une image initiale (a) par un masque 3×3 (b) pour obtenir l'image filtrée (c).

Les masques horizontaux et verticaux H et V sont issus de l'approximation des dérivées du premier ordre avec un schéma centré aux différences finies du second ordre. Ici, l'axe x est orienté de haut en bas et l'axe y de la gauche vers la droite (fig. 2.3).

Les filtres basés sur l'opérateur gradient, comme le filtre de Sobel, semblent démontrer une meilleure corrélation entre les pics de la fonction d'intensité et les contours, que ne le font les filtres basés sur l'opérateur laplacien [Torre *et al.* 1986] (fig.2.4). Les résultats obtenus avec ces derniers sont plus sensibles à la présence de bruit (fig.2.4.c₂) alors que les filtres basés sur le gradient obtiennent des contours en général plus épais, mettant de fait davantage en exergue les contours (fig.2.4.b_i).

Une approche non basée sur des filtres dérivateurs appelée **contours actifs** ou **snakes** [Blake *et al.* 1998] a été proposée. Une série de points libres est répartie sur une courbe 2D placée par un utilisateur dans la zone d'intérêt de l'image. La courbe sera déformée sous l'action de plusieurs forces/énergies afin d'adopter la position qui minimise ces énergies. Initialement, le contour est disposé uniformément autour de l'objet, puis il va se déplacer pour se confondre au mieux avec le contour de l'objet (fig.2.5). A chaque itération, l'algorithme calcule un positionnement des points qui minimise des énergies dérivées. L'algorithme s'arrête lorsque la minimisation n'a plus de meilleure solution ou que le nombre maximum d'itérations est atteint. Les notions d'énergies interne et externe sont utilisées : l'énergie interne ne dépend que des points du contour (courbure du contour, régularité d'espacement des points) alors que l'énergie externe caractérise les zones d'intérêt de l'image (zones de fort gradient par exemple) vers lesquelles le contour actif doit se

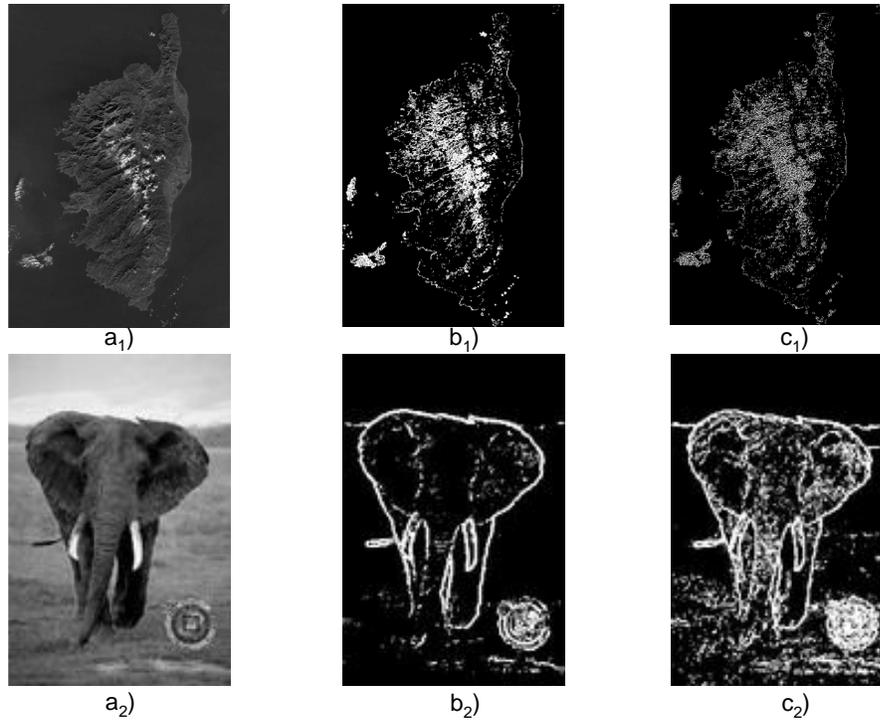


FIGURE 2.4 – Comparaison des filtres basés sur l'opérateur gradient (b_i) et sur l'opérateur laplacien (c_i) de deux images en niveaux de gris (a_i) (courtoisie de Julien Fauqueur, <http://docs.ufrmd.dauphine.fr/> et de <http://visibleearth.nasa.gov/>).

rapprocher.

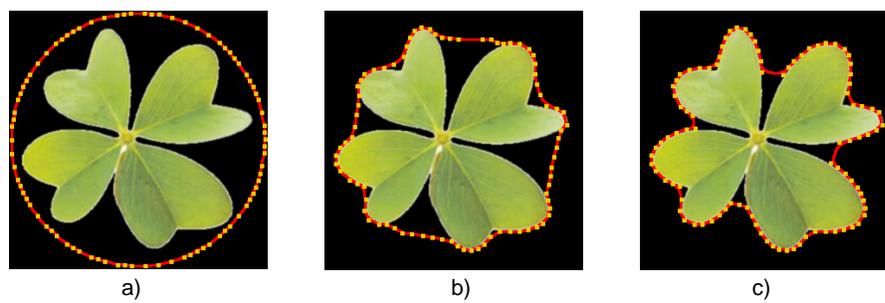


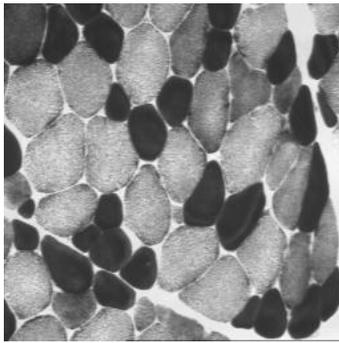
FIGURE 2.5 – Différentes itérations de l'algorithme de contour actif : initialement, le contour est positionné autour de l'objet (a) ; au milieu de l'exécution à une itération donnée, il commence à épouser le contour de l'objet (b) pour s'y confondre complètement (c) [Schwartz 2008].

Les inconvénients de la méthode des contours actifs résident dans sa faiblesse à détecter les contours internes de l'objet, ainsi que ceux des formes présentant de nombreuses et fortes concavités. En effet, le contour actif échouera à se positionner dans les concavités car cela pénalise son énergie interne en augmentant sa courbure. Sur la figure 2.5, on observe les divergences du contour actif par rapport aux contours de la base des feuilles du trèfle. En revanche, le contour épouse parfaitement les bords externes des feuilles, là où les contraintes de son énergie interne sont les mieux respectées.

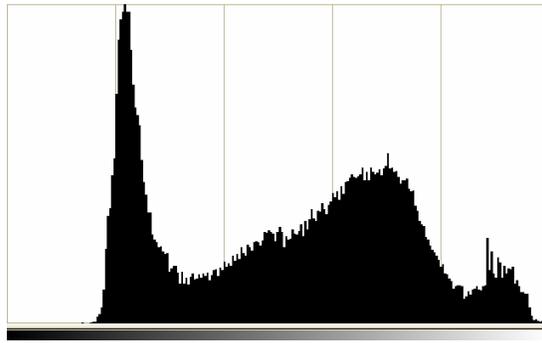
2.2.2 Segmentation

La segmentation d'une image consiste à la partitionner en distinguant différentes régions (ensemble de pixels connexes) présentant une certaine homogénéité selon un critère donné. Les techniques de segmentation peuvent être rassemblées en trois catégories :

- les **approches globales**, basées sur la notion d'*histogramme* [Monga *et al.* 1995, Olhander *et al.* 1978]. L'histogramme est la courbe de distribution des intensités des pixels d'une image, c'est-à-dire le nombre de pixels pour chaque intensité lumineuse (fig. 2.6.b). Les objets présents dans l'image (fig. 2.6.a) ont des valeurs bien distinctes et uniformes créant des "pics" dans l'histogramme. Le critère de segmentation utilisé est l'écart en intensité lumineuse des pixels. Dans cette méthode, les informations de proximité sont ignorées.



a)



b)

FIGURE 2.6 – Segmentation par histogramme (b) de l'image microscopique d'un muscle (a). L'histogramme (b) présente trois pics : à gauche le pic correspond aux cellules sombres, à droite au fond de l'image, et au milieu, le pic représente les cellules grises, prépondérantes dans l'image.

- les **approches locales** (accroissement de régions) [Monga *et al.* 1995, Bryant 1979] : en partant d'un état initial (pixel *amorce*) qui constitue une région élémentaire, cette région est étendue en y incluant les pixels du voisinage qui satisfont le critère local d'homogénéité (fig. 2.7). Le critère d'homogénéité local peut se baser sur l'écart-type $\gamma(x, y)$ (éq. 2.5) de l'intensité lumineuse qui décrit, pour chaque pixel de coordonnées (x, y) , le contraste de l'intensité dans son voisinage défini par une matrice de taille $d \times d$ [Cheng *et al.* 2000] :

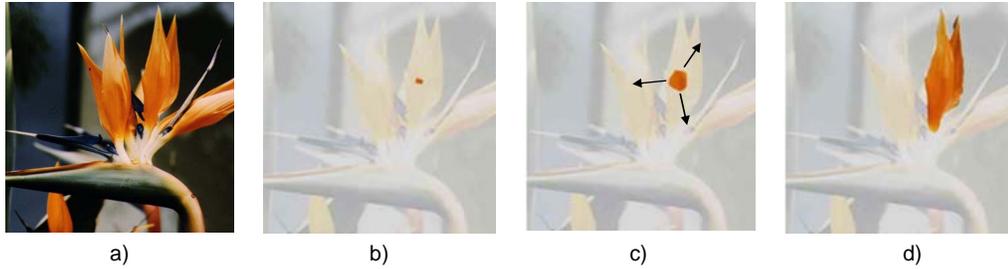


FIGURE 2.7 – A partir d'un point amorce (seed), on étend la région en ajoutant les points de la frontière qui satisfont le critère local d'homogénéité, ici, comparaison de la valeur du pixel candidat et du pixel de la frontière (*courtoisie de* <http://uei.ensta.fr/baillie/eng/index.html>).

$$\gamma(x, y) = \sqrt{\frac{1}{d^2} \sum_{p=x-\frac{d-1}{2}}^{x+\frac{d-1}{2}} \sum_{q=y-\frac{d-1}{2}}^{y+\frac{d-1}{2}} (I(p, q) - \mu(x, y))^2} \quad (2.5)$$

avec m la hauteur de l'image, n , sa largeur et $0 \leq x, p \leq m - 1$, $0 \leq y, q \leq n - 1$.

$\mu(x, y)$ est la moyenne des intensités lumineuses dans le même voisinage défini par la matrice de taille $d \times d$ autour du pixel de coordonnées (x, y) :

$$\mu(x, y) = \frac{1}{d^2} \sum_{p=x-\frac{d-1}{2}}^{x+\frac{d-1}{2}} \sum_{q=y-\frac{d-1}{2}}^{y+\frac{d-1}{2}} I(p, q) \quad (2.6)$$

- les **approches hybrides** (*split and merge*) [Monga *et al.* 1995, Horowitz *et al.* 1974] : l'image est considérée comme un arbre (*quadtrees*). Récursivement, chaque feuille est subdivisée en quatre si elle n'est pas assez homogène et les quatre sous images sont ajoutées comme des feuilles à l'arbre (fig. 2.8). La condition d'arrêt est atteinte lorsqu'il n'y a plus de feuilles non homogènes à subdiviser. On connecte ensuite

les régions adjacentes si leur différence d'homogénéité est inférieure à un seuil donné (fig. 2.9).

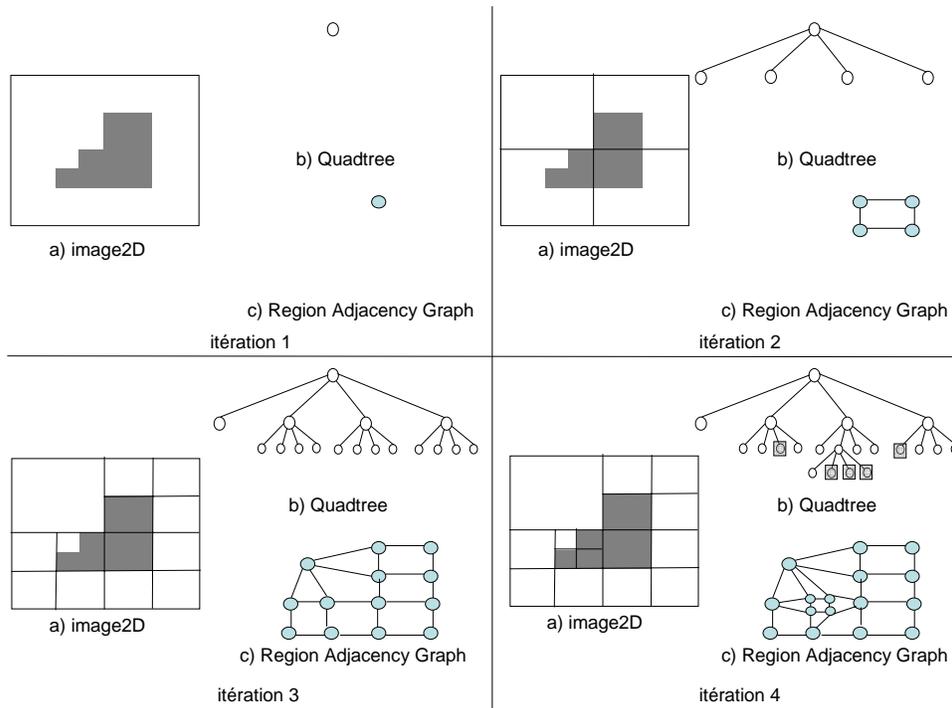


FIGURE 2.8 – Les différentes itérations de l'algorithme du *Split* avec la construction du quadtree (b) et du graphe de connectivité *Region Adjacency Graph* (RAG) (c) : on subdivise en quatre l'image (a) récursivement jusqu'à ce que les régions soient homogènes (b) ; les arêtes du RAG représentent les distances entre régions adjacentes en terme de différence d'homogénéité *courtoisie de* <http://uei.ensta.fr/baillie/eng/index.html>.

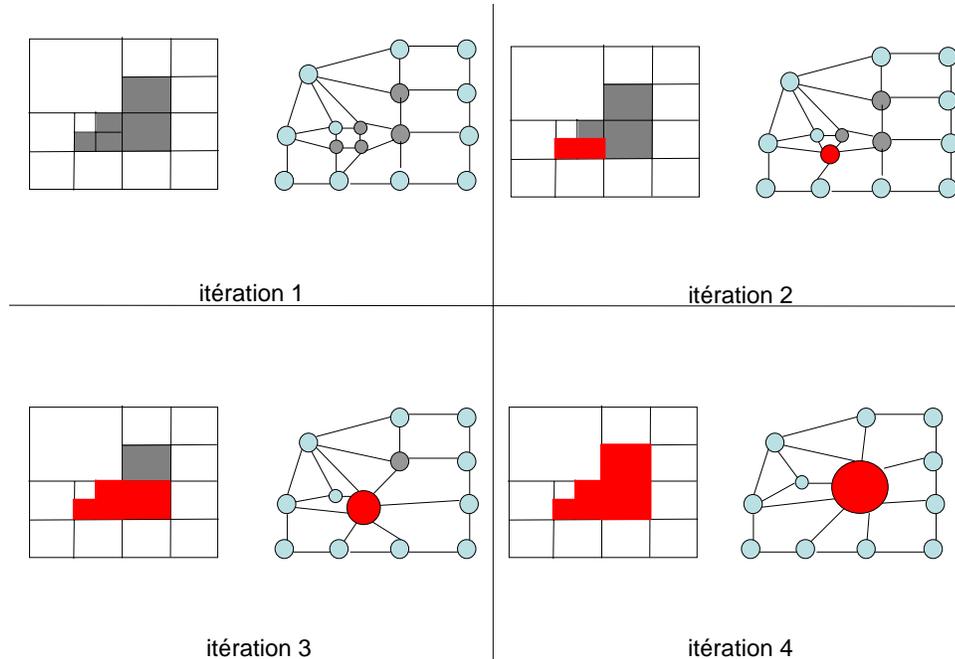


FIGURE 2.9 – Les différentes itérations de l'algorithme du *Merge* : pour chaque noeud du RAG, si un de ses voisins a une distance inférieure au seuil de regroupement, les deux noeuds fusionnent. On s'arrête lorsque aucun noeud ne peut plus fusionner avec l'un de ses voisins (*courtoisie de <http://uei.ensta.fr/baillie/eng/index.html>*).

2.2.3 Reconnaissance d'objets

La reconnaissance d'objets consiste à reconnaître des formes géométriques simples (droites, cercles, ellipses, courbes de Bézier...) ou plus complexes, comme des lettres, des chiffres, des empreintes digitales, dans les images numériques.

Parmi ces méthodes de reconnaissance d'objets, les méthodes de vectorisation s'intéressent à la reconnaissance d'ensemble de "barres" ou vecteurs (segments de droites). Les méthodes sont regroupées en deux catégories d'algorithmes de vectorisation : les méthodes aminçissantes et les méthodes non aminçissantes [Wenyin *et al.* 1999]. Les méthodes aminçissantes utilisent généralement un processus d'érosion itératif de suppression de pixels jusqu'à l'obtention d'un squelette dont l'épaisseur est égale à un pixel. Cependant, ces méthodes sont plus lentes que les méthodes non aminçissantes.

L'algorithme *Orthogonal Zig-Zig* (OZZ) [Chai *et al.* 1992], qui est une méthode de vectorisation non aminçissante, est très utilisé en raison de sa

simplicité et son temps d'exécution rapide qui est du à l'échantillonnage clairsemé de l'image. Il est basé sur le suivi de pixel à l'intérieur d'une zone de pixels noirs d'une image, poursuivant le parcours d'une manière orthogonale chaque fois que le chemin rencontre une frontière de la zone. Cet algorithme permet d'éviter partiellement les problèmes fréquemment rencontrés d'intersections et de jonctions entre les zones de contours. Il préserve également l'épaisseur de la ligne. Le principal inconvénient concerne son incapacité à approcher les arcs de cercle dans les images, les vecteurs voisins étant soit en intersection soit déconnectés.

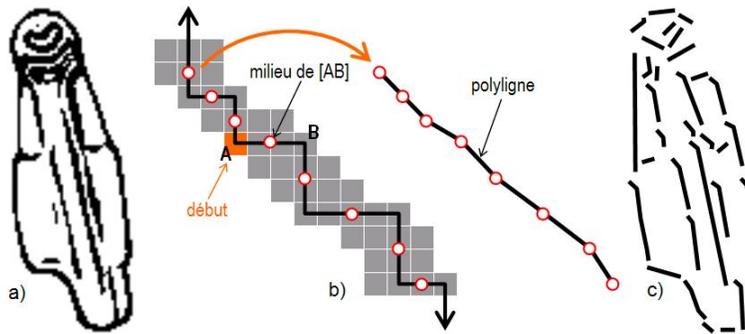


FIGURE 2.10 – Principe de l'algorithme OZZ (b) appliqué à une image des contours (a) produisant ainsi une image vectorisée (c).

L'algorithme *Sparse Pixel Vectorization* [Wenyin *et al.* 1999] est une optimisation de l'algorithme OZZ. Plutôt que de visiter tous les pixels de la zone de contours au moins une fois, seul un sous-ensemble sélectionné de points de l'axe médian est visité. Le résultat est une polyligne lissée par un algorithme d'approximation polygonale [Sklansky *et al.* 1980] pour enlever les points redondants.

La méthode de la transformée de Hough pour la détection de droites [Hough 1962] est également très utilisée. Dans l'espace de Hough, chaque droite est représentée par un vecteur de coordonnées paramétriques (fig .2.11) :

- Θ : son angle ou orientation de la droite
- ρ : la norme du vecteur (longueur du segment perpendiculaire à la droite d'angle Θ et passant par l'origine)

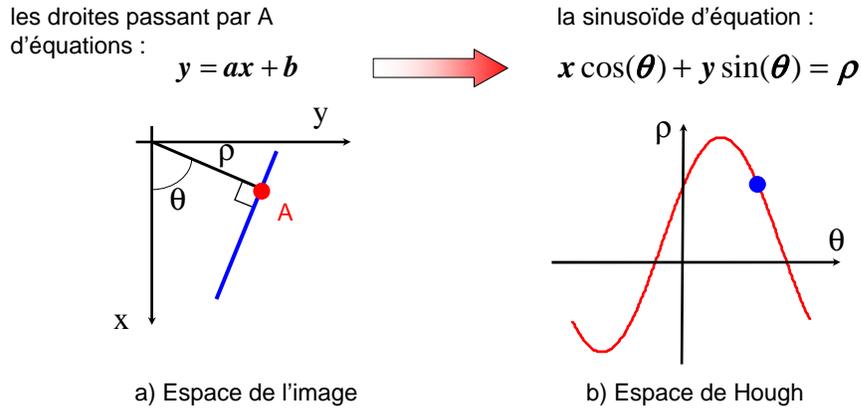


FIGURE 2.11 – Principe de la transformée de Hough : dans l'espace de Hough (b) une sinusoïde unique correspond à l'ensemble des droites passant par un point dans l'espace de l'image (a).

Dans l'espace de l'image (fig.2.11.a), une droite passant par un point correspond à un point (fig.2.11.b) de la sinusoïde dans l'espace de Hough. En calculant toutes les droites possibles passant par un point, c'est-à-dire en calculant la valeur de ρ pour chaque valeur de Θ , on obtient une sinusoïde unique dans l'espace de Hough. Afin de retrouver les paramètres de la droite passant par un groupe de pixels (fig.2.11.a.) d'une image, on calcule l'endroit où les sinusoïdes correspondant à ces pixels se coupent dans l'espace de la transformée de Hough.

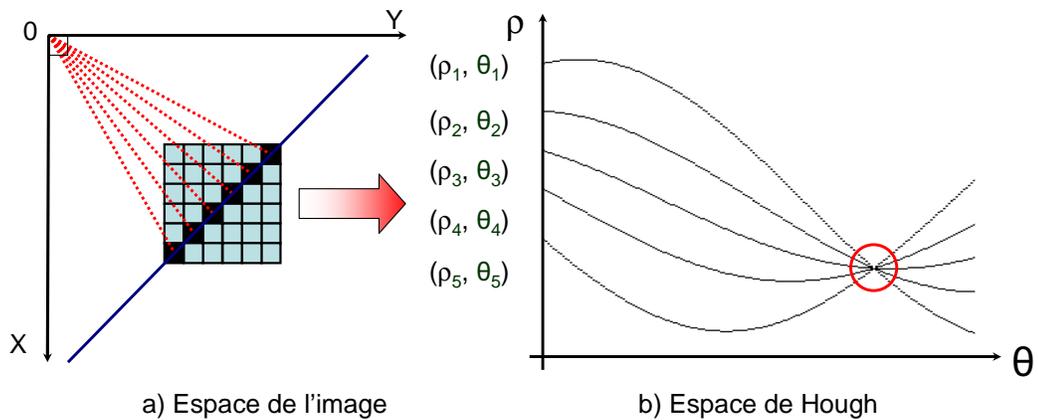


FIGURE 2.12 – La droite dans l'espace de Hough qui est à l'intersection de toutes les sinusoïdes (b) passe donc par tous les pixels correspondants aux sinusoïdes (a).

La transformée de Hough peut aussi être étendue à la détection d'objets d'équations paramétriques plus complexes comme les cercles et les ellipses [Yip *et al.* 1992, Le Troter *et al.* 2005].

2.2.4 Bilan des techniques d'extraction des contours, de segmentation d'images et de reconnaissance d'objets

Les méthodes d'extraction (section 2.2.1) permettent d'obtenir une image dont les pixels colorés correspondent aux pixels détectés comme appartenant à un contour de l'image initiale. L'information extraite fournit donc les coordonnées 2D dans le repère lié à l'image des lignes de caractère 3D des objets, ce qui est intéressant dans la cadre de nos travaux.

Les méthodes de segmentation d'images (cf. section 2.2.2) découpent l'image initiale en regroupant les pixels en régions distinctes. Le type d'informations fourni par les méthodes de segmentation correspond aux coordonnées 2D mais il comporte également une information relative au critère utilisé pour la segmentation. Par exemple, des informations sur la texture, le contraste, l'intensité lumineuse des pixels leur sont associées. Cette catégorie de techniques de traitement d'images est intéressante mais dépasse l'identification des lignes de caractère qui est l'objet de notre étude, en détectant des entités de plus haut niveau (les "régions").

Enfin, les méthodes de reconnaissance d'objets (section 2.2.3) sont pertinentes car elles produisent des informations relatives à différentes formes identifiées dans les images. En particulier, les techniques de vectorisation [Wenyin *et al.* 1999, Chai *et al.* 1992] qui permettent à partir d'une image de contours d'obtenir un ensemble de vecteurs, peuvent fournir les coordonnées 2D des lignes de caractère des objets.

2.3 Calibration de caméras

Dans cette section, nous décrivons le modèle géométrique associé au processus de formation des images numériques. En effet, ce modèle idéal (fig. 2.13) lie la caméra à partir de laquelle la photographie a été prise, l'image et l'objet 3D qu'elle représente [Monga *et al.* 1995].

2.3.1 Modèle de projection perspective (modèle sténopé linéaire)

Le modèle sténopé est un modèle idéal régi par les théorèmes de la géométrie projective centrale et qui formalise le processus physique d'acquisition d'image qui se produit dans une caméra (fig.2.13) sans tenir compte des phénomènes optiques réels tels que la distorsion [Debevec 1996].

Il est défini par un centre optique C , l'orientation du repère local lié à la caméra (C, C_x, C_y, C_z) dans le repère global de référence de la scène (O, x, y, z) , et une distance focale $f = Cc$ où c est le centre de l'image 2D (figure 2.13).

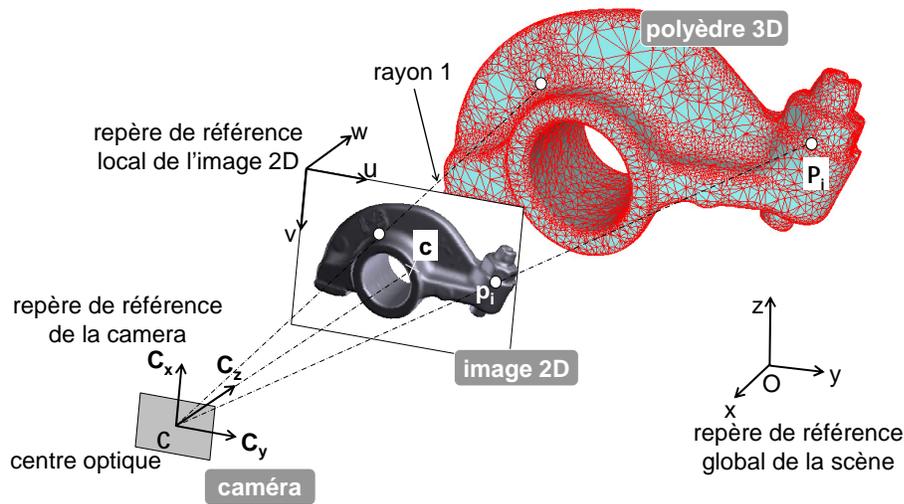


FIGURE 2.13 – Principe de la calibration de caméra avec un modèle sténopé linéaire.

On utilise la notation matricielle des *coordonnées homogènes* qui permettent la linéarisation des opérations (projection, rotation, translation, changement d'échelle). Par exemple, le système :

$$A \cdot X + B = Y \iff C \cdot \tilde{X} = \tilde{Y} \text{ avec } P = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \iff \tilde{P} = \begin{bmatrix} X' \\ Y' \\ Z' \\ W \end{bmatrix} \quad (2.7)$$

On passe des coordonnées cartésiennes aux coordonnées homogènes en posant $W = 1$. Soit (X_i, Y_i, Z_i) les coordonnées d'un point P_i de la scène

3D et (u_i, v_i, w_i) les coordonnées homogènes de sa projection p_i sur le plan image, correspondant à la surface photosensible (le capteur CDD) (figure 2.13). La calibration tend à calculer la matrice M telle que pour chaque couple de points (p_i, P_i) :

$$\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \mathbf{M} \cdot \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \quad (2.8)$$

avec

$$M = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} \quad (2.9)$$

Pour chaque couple de points (p_i, P_i) en posant [Monga *et al.* 1995] :

$$\begin{cases} U_i = m_{11}.X_i + m_{12}.Y_i + m_{13}.Z_i + m_{14} \\ V_i = m_{21}.X_i + m_{22}.Y_i + m_{23}.Z_i + m_{24} \\ W_i = m_{31}.X_i + m_{32}.Y_i + m_{33}.Z_i + m_{34} \end{cases} \quad (2.10)$$

on obtient :

$$u_i = \frac{U_i}{W_i} \quad (2.11)$$

$$v_i = \frac{V_i}{W_i} \quad (2.12)$$

et en multipliant 2.11 et 2.12 par W_i :

$$\begin{cases} X_i.m_{11} + Y_i.m_{12} + Z_i.m_{13} + m_{14} - X_i.m_{31}.u_i - Y_i.m_{32}.u_i - Z_i.m_{33}.u_i = u_i.m_{34} \\ X_i.m_{21} + Y_i.m_{22} + Z_i.m_{23} + m_{24} - X_i.m_{31}.v_i - Y_i.m_{32}.v_i - Z_i.m_{33}.v_i = v_i.m_{34} \end{cases} \quad (2.13)$$

Le système des équations (2.8) peut se mettre sous la forme :

$$A.p = b \quad (2.14)$$

Puis en développant :

$$Q = {}^t p' \cdot {}^t B \cdot B \cdot p' + {}^t p'' \cdot {}^t C \cdot C \cdot p'' + 2 \cdot {}^t p'' \cdot {}^t C \cdot B \cdot p' + \lambda \cdot (1 - {}^t p'' \cdot p'') \quad (2.29)$$

En imposant que les dérivées partielles par rapport à p' et p'' soient nulles, on obtient :

$${}^t B \cdot B \cdot p' + {}^t B \cdot C \cdot p'' = 0 \quad (2.30)$$

$${}^t C \cdot C \cdot p'' + {}^t C \cdot B \cdot p' - \lambda \cdot p'' = 0 \quad (2.31)$$

D'où :

$$p' = -({}^t B \cdot B)^{-1} \cdot {}^t B \cdot C \cdot p'' \quad (2.32)$$

$$D \cdot p'' = \lambda \cdot p'' \quad (2.33)$$

$$D = {}^t C \cdot C - {}^t C \cdot B \cdot ({}^t B \cdot B)^{-1} \cdot {}^t B \cdot C \quad (2.34)$$

Finalement, en substituant dans (2.29), on obtient pour Q :

$$Q = {}^t p'' \cdot D \cdot p'' = \lambda \cdot {}^t p'' \cdot p'' = \lambda \quad (2.35)$$

On remarque que D est une matrice symétrique et positive 3×3 . Elle a donc des valeurs propres réelles et positives. p'' est un vecteur propre de D associé à la valeur propre λ . Pour minimiser la fonctionnelle Q il faut calculer les valeurs propres de D , choisir la plus petite (car c'est celle qui minimise la fonctionnelle), calculer son vecteur propre associé, c'est-à-dire p'' , le normer, et finalement calculer p' . Les coefficients de la matrice M sont alors fournis par p' et p'' au signe près. On vérifie alors que l'on a bien m_{14} positif, ce qui assure que les correspondances de points (p_i, P_i) ont été faites à l'aide d'un objet 3D visible, c'est-à-dire, situé devant la caméra.

Finalement, on peut remarquer que l'étape de calibration de caméra est optionnelle quand on manipule des captures d'écran ou des images géoréférencées puisque dans ces deux cas les paramètres de la caméra sont déjà connus. De nouvelles méthodes d'auto calibration qui ne sont pas basées sur des correspondances de points mais sur l'utilisation de plusieurs images ont été proposées [Faugeras *et al.* 1992].

2.3.2 Lancer de rayons

Le lancer de rayon [O'Rourke 1998] simule le parcours inverse de la lumière allant de la scène vers l'oeil. En vision artificielle, cela consiste à "lancer un rayon" (tracer une droite) issu du point de vue (centre de la caméra) passant par chaque pixel de l'image et allant intersecter un objet de la scène 3D (fig. 2.13). Une fois l'image 2D et le modèle 3D calés dans un même repère de référence. Le lancer de rayon permet donc de faire le lien entre un pixel et son correspondant 3D sur le modèle et *vice-versa*.

2.3.3 Bilan des méthodes de calibration de caméras et de lancer de rayons

Les méthodes de calibration de caméras permettent de calculer dans le repère global de la scène, les coordonnées et l'orientation de la caméra à partir de laquelle une image a été prise. Les techniques de lancer de rayon quant à elles, permettent de calculer les coordonnées 3D des projections des pixels d'une image sur un modèle 3D, lorsque la calibration a été réalisée ou qu'elle n'est pas nécessaire comme dans le cas où l'image est une photographie géoréférencée ou une capture d'écran. Une fois les lignes de caractère identifiées par une des méthodes de traitement d'images présentées en section 2.2, c'est donc grâce à ces deux techniques que le lien entre les pixels identifiés et les points 3D correspondants est réalisé.

2.4 Techniques de reconstruction 3D à l'aide d'images

L'objectif de la section suivante n'est pas de faire une étude exhaustive des méthodes et approches de reconstruction 3D à l'aide d'images mais plutôt de donner un aperçu des trois techniques les plus utilisées.

2.4.1 Tomographie

La tomographie [Kak *et al.* 1988] est une technique d'imagerie, fréquemment utilisée dans le domaine médical, qui permet de reconstruire le volume d'un objet à partir d'une série de mesures effectuées par tranche depuis l'extérieur de cet objet. Plusieurs techniques sont possibles : tomographie à rayons X (fig. 2.14), à émission de positrons, de photons, par IRM (résonance magnétique), etc. . A partir d'une émission, de rayons X ou de photons, une

projection (image 2D ou *coupe*) de l'objet est obtenue à une hauteur donnée. Plusieurs émissions sont réalisées autour de l'objet selon sa taille et sa complexité. En général, une soixantaine de coupes minimum est nécessaire pour reproduire fidèlement l'objet. Le volume 3D est ensuite reconstruit par interpolation entre les coupes.

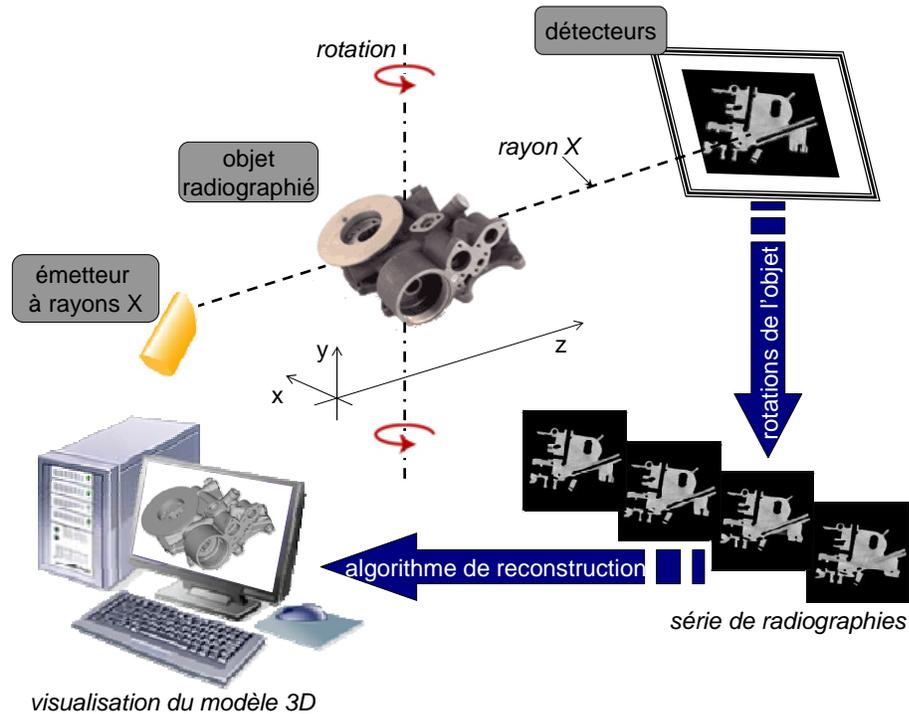


FIGURE 2.14 – Principe de la tomographie par émission de rayons X. 160 coupes ont été nécessaires pour reconstruire le modèle 3D d'une pièce mécanique (*courtoisie de Tomoadour*).

2.4.2 Stéréoscopie

La géométrie épipolaire [Monga *et al.* 1995, Chen *et al.* 2006] définit les relations qui existent entre les points 2D des images et leur point M 3D correspondant (fig. 2.15). Soient deux caméras de centres optiques C_1 et C_2 , M un point de l'espace se projetant en m_1 sur l'image 1 et m_2 sur l'image 2, alors M se situe à l'intersection de la droite (C_1m_1) avec la droite (C_2m_2) . Les correspondances de pixels sont réalisées par l'utilisateur. Chaque correspondance permet de retrouver un point 3D par intersection des deux rayons 3D. En effet, [Goldman 1990] montre que chaque rayon $(C_i m_i)$ peut être défini par son origine C_i et un vecteur unitaire directeur d_i . Les deux lignes

3D peuvent donc s'écrire :

$$\begin{cases} L_1(t_1) = C_1 + d_1.t_1 \\ L_2(t_2) = C_2 + d_2.t_2 \end{cases} \quad (2.36)$$

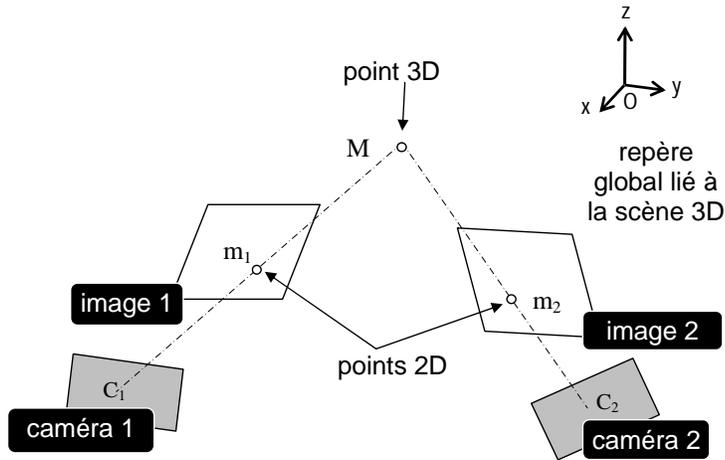


FIGURE 2.15 – Géométrie épipolaire de deux caméras

La solution (t_1, t_2) , correspondant à la position de M sur les deux droites, est donnée par :

$$\begin{cases} t_1 = \frac{|A_2|}{\|v\|^2} \\ t_2 = \frac{|A_1|}{\|v\|^2} \end{cases} \quad (2.37)$$

avec :

$$v = d_1 \wedge d_2 \quad (2.38)$$

$$A_1 = \begin{bmatrix} x_{O_2} - x_{O_1} & y_{O_2} - y_{O_1} & z_{O_2} - z_{O_1} \\ x_{d_1} & y_{d_1} & z_{d_1} \\ x_v & y_v & z_v \end{bmatrix} \quad (2.39)$$

et

$$A_2 = \begin{bmatrix} x_{O_2} - x_{O_2} & y_{O_2} - y_{O_2} & z_{O_2} - z_{O_2} \\ x_{d_2} & y_{d_2} & z_{d_2} \\ x_v & y_v & z_v \end{bmatrix} \quad (2.40)$$

l'opérateur \wedge désignant le produit vectoriel entre deux vecteurs, et $|A_i|$ étant le déterminant de la matrice A_i . (x_{di}, y_{di}, z_{di}) sont les coordonnées d'un vecteur d_i et (x_v, y_v, z_v) sont celles du vecteur v .

Si les rayons sont parallèles, le dénominateur $\|\mathbf{v}\|^2$ est nul. Si les rayons n'ont pas d'intersection, t_1 et t_2 sont les coordonnées paramétriques des points les plus proches sur chaque rayon. On prend alors le milieu de ces deux points comme solution.

2.4.3 Shape From Shading

Plusieurs techniques tentent de reconstruire l'information de profondeur à partir d'images en niveaux de gris. Elles portent le nom de "méthodes Shape From X", où X peut signifier "Textures", "Shadows"², "Contours", "Shading"³. Parmi ces différentes techniques, les méthodes de Shape From Texture ne fournissent que des indications sur la surface 3D représentée par la texture et requièrent qu'elle soit homogène [Criminisi *et al.* 2000]. Les travaux sur le Shape From Shadows analysent la formation du contour d'un objet connu (appelé *objet écran*) sur la surface à reconstruire [Yu *et al.* 2002]. Les techniques de Shape From Contours utilisent les contours des images d'un même objet vu sous différents angles pour en extraire une enveloppe englobante, puis une description volumétrique [Fromherz *et al.* 1994]. Finalement, seules les méthodes de Shape From Shading essaient de reconstruire des informations 3D à partir d'une et une seule image en niveaux de gris.

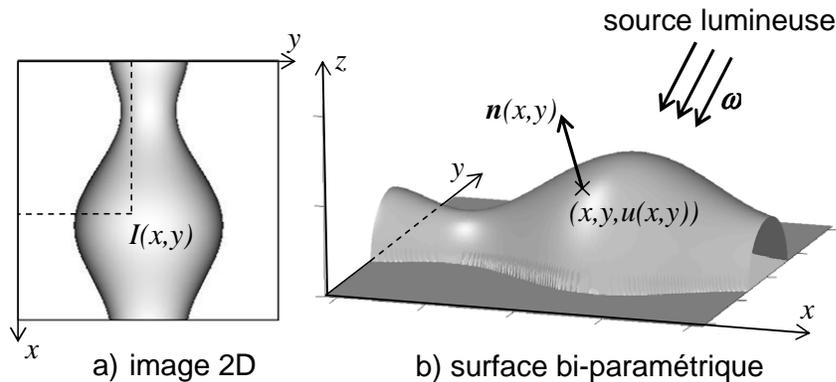


FIGURE 2.16 – Image (a) d'une surface (b) éclairée par une source de lumière située à l'infini.

2. Ombres
3. Ombrages

Les premières approches de Shape From Shading (*SFS*) ont été développées dans les années 70. Elles visent à reconstruire des modèles 3D à partir d'une ou de plusieurs images 2D. Par exemple, à partir de l'image 2D (fig.2.16.a), il est possible de reconstruire le maillage 3D (fig.2.16.b). Horn [Horn *et al.* 1989] formalise le problème du SFS comme la résolution de l'équation d'irradiance de l'image, une équation aux dérivées partielles du premier ordre non linéaire (éq. 2.41). Ensuite, plusieurs méthodes ont été proposées, fournissant des résultats de basse qualité et dont la convergence n'était pas assurée. De plus, il a été prouvé que la solution du problème du SFS n'est pas unique [Belhumeur *et al.* 1997, Prados *et al.* 2002, Rouy *et al.* 1992]. La principale difficulté est l'existence de l'ambiguïté concave/convexe qui est illustrée sur la figure 2.17 ; en effet, selon son orientation, une image 2D peut représenter à la fois une forme concave ou convexe.

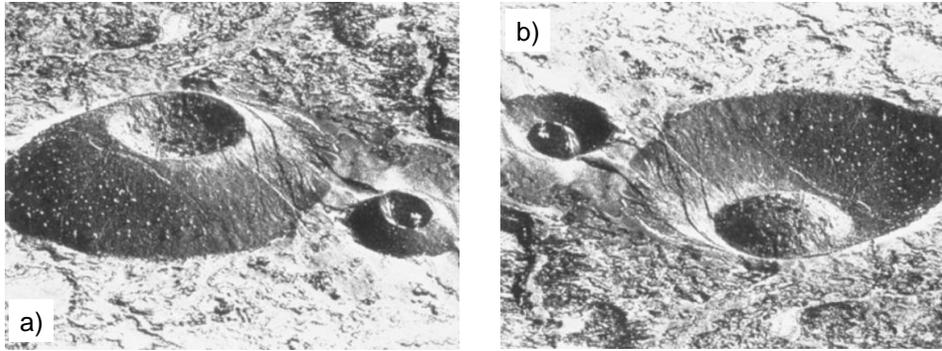


FIGURE 2.17 – Un exemple d'ambiguïté concave/convexe [Prados *et al.* 2005]. Selon son orientation, une image peut montrer des formes qui ressemblent à des cratères (b) ou à des volcans (a).

Soit $z = u(x, y)$, avec $(x, y) \in \Omega \subset \mathbb{R}^2$, la surface bi-paramétrique éclairée par une unique source lumineuse située à l'infini et ayant $\omega = (\omega_1, \omega_2, \omega_3) \in \mathbb{R}^3$ comme vecteur directeur unitaire (fig. 2.16.b), l'équation d'irradiance de l'image s'écrit comme suit [Horn *et al.* 1989, Durou *et al.* 2004] :

$$R(\mathbf{n}(x, y)) = I(x, y) \quad \forall (x, y) \in \Omega \quad (2.41)$$

avec I la fonction de l'intensité lumineuse de l'image (donnée) (fig. 2.16.a), et R la fonction de réflectance qui donne la valeur de la réflexion de la lumière en un point 3D $(x, y, u(x, y))$. $\mathbf{n}(x, y)$ est la normale unitaire à la surface telle que :

$$\mathbf{n}(x, y) = \frac{1}{\sqrt{1 + p^2 + q^2}}(-p, -q, 1) \quad \forall (x, y) \in \Omega \quad (2.42)$$

avec $p = \frac{\partial u}{\partial x}(x, y)$ et $q = \frac{\partial u}{\partial y}(x, y)$ les composants de $\nabla u(x, y) = \begin{pmatrix} p \\ q \end{pmatrix}$.

Si nous supposons que la surface est Lambertienne, i.e. la fonction de réflectance est directement proportionnelle au cosinus de l'angle entre la direction de la source lumineuse et la normale à la surface (fig.2.16.b), ce qui est le cas le plus fréquent concernant des objets réels, on peut ajouter l'équation suivante :

$$R(\mathbf{n}(x, y)) = \boldsymbol{\omega} \cdot \mathbf{n}(x, y) \quad \forall (x, y) \in \Omega \quad (2.43)$$

Ainsi, l'équation (2.41) devient l'équation non linéaire aux dérivées partielles du premier ordre suivante :

$$I(x, y)\sqrt{1 + \|\nabla u(x, y)\|^2} + (\omega_1, \omega_2) \cdot \nabla u(x, y) - \omega_3 = 0 \quad \forall (x, y) \in \Omega \quad (2.44)$$

En posant $\boldsymbol{\omega} = (0, 0, 1)$, l'équation (2.44) devient l'équation de l'Eikonal :

$$\left[\frac{\partial u}{\partial x}(x, y) \right]^2 + \left[\frac{\partial u}{\partial y}(x, y) \right]^2 = \frac{1}{I(x, y)^2} - 1 \quad \forall (x, y) \in \Omega \quad (2.45)$$

avec les conditions aux limites $u(x, y) = 0 \quad \forall (x, y) \in \partial\Omega$

Puisque l'image contient un ensemble fini de pixels (x_i, y_j) , la résolution de cette équation produit un ensemble de points 3D $(x_i, y_j, u(x_i, y_j))$ formant une surface discrète orientée le long des directions paramétriques x et y .

En pratique, des méthodes classiques de résolution des équations aux dérivées partielles ont donc été utilisées pour essayer de résoudre le problème du SFS : méthode des caractéristiques [Horn *et al.* 1989], méthode des courbes de niveau [Kimmel *et al.* 1995], méthode des solutions de viscosité, par des schémas aux différences finies [Lions *et al.* 1993, Prados *et al.* 2002], par des schémas semi-lagrangiens [Falcone *et al.* 1997], méthode du fast marching [Tankus *et al.* 2004], méthode de résolution par optimisation [Horn *et al.* 1989].

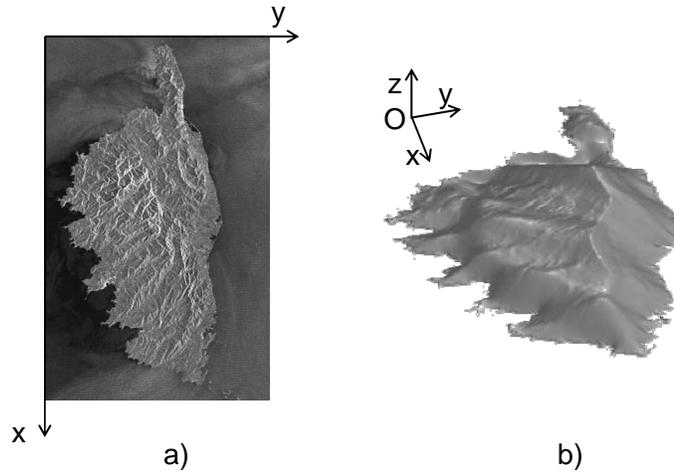


FIGURE 2.18 – Une image radar de la Corse (a) utilisée pour reconstruire un relief avec une méthode de SFS (b) [Daniel 2000] (courtoisie de la RADARSAT International Inc).

Parmi toutes ces méthodes, deux problèmes récurrents demeurent. Le premier concerne l'absence de conditions aux limites aux bords [Durou *et al.* 2000] qui est contourné en rajoutant une connaissance a priori sur l'altitude aux bords (condition de Dirichlet) ou sur son gradient (condition de Neumann). En effet, sur l'exemple de la figure 2.18, une image radar de la Corse a été utilisée. Le relief à reconstruire étant celui d'une île, l'élévation et son gradient des pixels du littoral sont fixés à 0, car l'altitude sur le littoral est nulle. Mais ce type d'information n'est généralement pas contenu dans des images réelles. Le deuxième problème majeur est le problème précédemment évoqué de l'ambiguïté concave/convexe qui reste à lever.

2.4.4 Bilan des techniques de reconstruction 3D à partir d'image(s)

La qualité de la reconstruction 3D obtenue par tomographie varie avec la technologie d'émission des rayons. Elle est en particulier liée au nombre de coupes réalisées et au bruit (dû aux phénomènes d'occlusion, d'atténuation de l'intensité des rayons...) présent dans les images utilisées. Ses applications sont en général limitées à l'imagerie médicale et la taille des objets à reconcevoir à l'échelle humaine (tab. 2.1). En retro conception, elle peut être intéressante car elle offre la possibilité de reconstruire les parties intérieures d'objets de formes complexes.

La reconstruction par stéréoscopie nécessite au moins deux images cali-

brées prises de deux points de vue différents. La géométrie épipolaire permet à partir de ces deux images, de calculer les coordonnées des points 3D correspondants aux couples de pixels contenus dans les images. La principale difficulté de la stéréoscopie réside dans le processus d'appariement des pixels homologues d'une image à l'autre (corrélation automatique) qui est un problème de traitement d'images à part entière (tab. 2.1). La stéréoscopie est très utilisée en architecture pour la reconstruction d'édifices.

Les méthodes de Shape From Shading doivent prendre en compte l'ambiguïté concave/convexe, les hypothèses usuelles (sur la source lumineuse, les propriétés de la surface de l'objet à reconstruire) et en général, la nécessité d'imposer des conditions aux limites (et donc d'utiliser des informations *a priori* sur l'altitude de certains points, tab. 2.1). Malgré l'émergence de méthodes qui formalisent ce problème inverse de manière plus réaliste (surface non lambertienne avec [Bakshi *et al.* 1994], projection perspective avec [Tankus *et al.* 2004]), l'utilisation d'images réelles ne produit pas de résultats probants.

Enfin, dans ces trois techniques, le phénomène d'occlusion restreint le champ de la reconstruction 3D.

Critère	Tomographie	Stéréoscopie	SFS
nombre d'images nécessaires	$\sim \geq 60$	≥ 2	1
calibration requise	oui	oui	non
sensibilité au bruit	oui	non	oui
ambiguïté de la reconstruction	non	non	oui

TABLE 2.1 – Comparatifs des méthodes de reconstruction 3D à partir d'images 2D.

Cependant la stéréoscopie peut fournir les coordonnées 3D des projections de pixels homologues de deux images. Ces coordonnées de points 3D, regroupées sous la forme de polygones, correspondent aux lignes de caractère 3D d'un objet, si les pixels utilisés sont les pixels noirs d'images de contours (lignes de caractère 2D) identifiés par une méthode de filtrage des contours (section 2.2.1). La stéréoscopie est donc intéressante pour notre étude, car elle fournit les lignes de caractère 3D d'un objet à partir d'images.

De plus, les techniques de Shape From Shading quant à elles, sont particulièrement intéressantes puisqu'à partir d'une unique image, elles retrouvent les élévations 3D des pixels à partir de leur intensité lumineuse. On obtient donc des informations sur les évolutions de formes d'un objet à partir d'une image. La tomographie n'a pas été employée car les images ("coupes") utilisées pour la reconstruction sont obtenues par émission de rayons X, ce

qui requiert des équipements adéquats.

2.5 Conclusion

Alors que les méthodes de filtrage, segmentation et de reconnaissance de formes, apportent des informations descriptives sur les formes représentées, les techniques de reconstruction 3D telles la tomographie, la stéréoscopie et le Shape From Shading, permettent à partir d'image(s) 2D de calculer des élévations 3D des pixels des images (fig.2.1).

Dans le cas de la manipulation de modèles polyédriques lourds (chapitre 1 section 1.4), une étape de simplification de la géométrie doit être effectuée. L'identification des lignes de caractère, afin d'assurer leur préservation pendant le processus de simplification est donc nécessaire. Une stratégie de simplification proposée consiste tout d'abord à détecter les lignes de caractère dans le repère 2D de l'image par extraction et vectorisation des contours (sections 2.2.1 et 2.2.3). Puis, par le biais des étapes de calibration de caméra et de lancer de rayon (sections 2.3 et 2.3.2), les éléments de la géométrie du modèle polyédrique correspondant à ces lignes de caractères 2D sont identifiés.

Dans le cas de modèles polyédriques scannés comportant des trous chapitre (1 section 1.4), cette stratégie ne peut être utilisée telle quelle car les entités du modèle appartenant à ses lignes de caractère sont manquantes.

Ainsi, deux nouvelles stratégies de déformation de polyèdres ont été élaborées. Après avoir rempli les trous par insertion d'un maillage venant se raccorder aux bords des trous, une déformation de ce maillage afin qu'il corresponde à la position des lignes de caractère 3D est réalisée.

- Dans la première approche, les lignes de caractère sont tout d'abord détectées dans deux images par extraction et vectorisation des contours (sections 2.2.1 et 2.2.3). Ensuite, les noeuds du maillage inséré sont contraints à se déplacer aux coordonnées 3D des lignes de caractère calculées par triangulation stéréoscopique (section 2.4.2) à partir de leurs coordonnées 2D.
- Dans la seconde approche, ces coordonnées 3D sont calculées directement en utilisant les équations du Shape From Shading (section 2.4.3).

Chapitre 3

Techniques de manipulation de modèles polyédriques

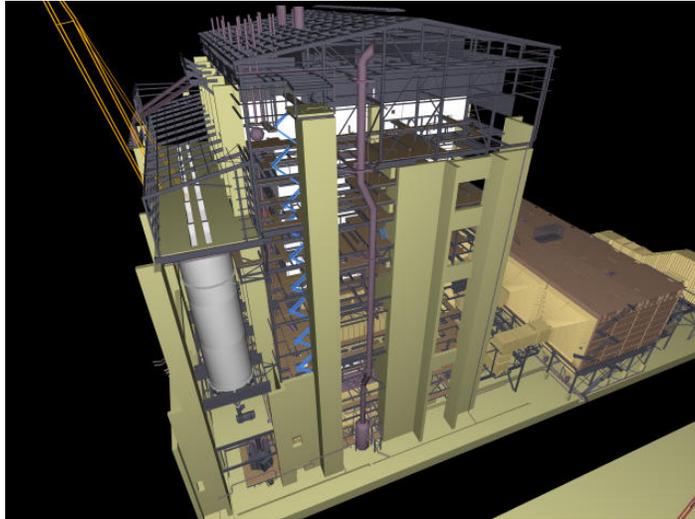
Ce chapitre est consacré à l'analyse des techniques de simplification et de déformation de modèles polyédriques. Les deux méthodes spécifiques qui ont été choisies pour réaliser le couplage polyèdres/images sont également présentées en détail.

3.1 Manipulations avancées de modèles polyédriques

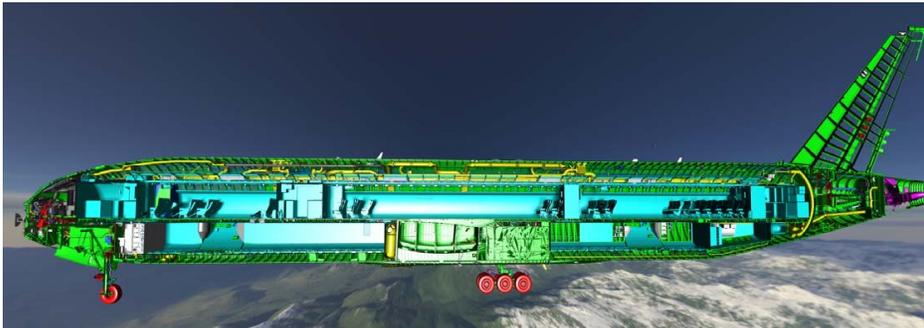
Fournir des méthodes de simplification et de déformation de modèles polyédriques les plus adaptées et performantes possibles est un enjeu important dans le domaine de l'ingénierie numérique de produits actuellement.

En effet, la simplification de modèles polyédriques permet la manipulation, le stockage et la transmission de modèles lourds qui sont utilisés au sein de maquettes numériques de produits. Par exemple, le modèle d'une centrale électrique [Wald *et al.* 2001] de la figure 3.1 comporte plus de 50 millions de faces et celui d'un modèle du Boeing 777 [Dietrich *et al.* 2004] en comptent plus de 350 millions. Ne serait-ce que pour leur visualisation, ce nombre élevé de faces est pénalisant. En effet, un taux de rafraîchissement de 3.4 images par seconde ou fps¹ est seulement atteint pour visualiser le modèle de la centrale électrique et de 5 fps pour celui du Boeing 777. Ceci est particulièrement faible quand on considère qu'un taux de rafraîchissement à partir duquel le cerveau humain qualifie un mouvement de fluide, se situe aux alentours de 10 fps. Le maximum est atteint avec un taux de 60 fps. Au delà de ce taux, le cerveau humain n'a alors plus le temps d'analyser l'image que l'oeil perçoit. La vitesse d'une projection cinématographique est en règle générale de l'ordre de 25 fps.

1. fps = *frame per second*, en anglais



a) 50 millions de faces, 3.4 fps



b) 350 millions de faces, 5 fps

FIGURE 3.1 – Deux modèles polyédriques lourds : un modèle de centrale électrique [Wald *et al.* 2001] et un modèle du Boeing 777 [Dietrich *et al.* 2004] (courtoisie de la Computers Graphics Group. Saarland University et de la compagnie Boeing).

La déformation de modèles polyédriques est également une étape fondamentale pour les activités de design, de définition et de modification des formes de pièces mécaniques. On admet habituellement qu'un concepteur consacre 20% de son activité à la création de formes et 80% à faire des modifications. Les processus de modification doivent donc être particulièrement efficaces. De plus, la déformation peut s'appliquer au remplissage de trous dans les modèles scannés (fig. 3.2.a et 3.2.b). La figure 3.2.c montre le modèle d'une prise d'escalade dans lequel une zone de la géométrie a été supprimée. Un maillage a été inséré pour remplir topologiquement le trou (fig. 3.2.d), puis déformé pour qu'il satisfasse des conditions de raccord en courbure avec le modèle initial [Pernot *et al.* 2006].

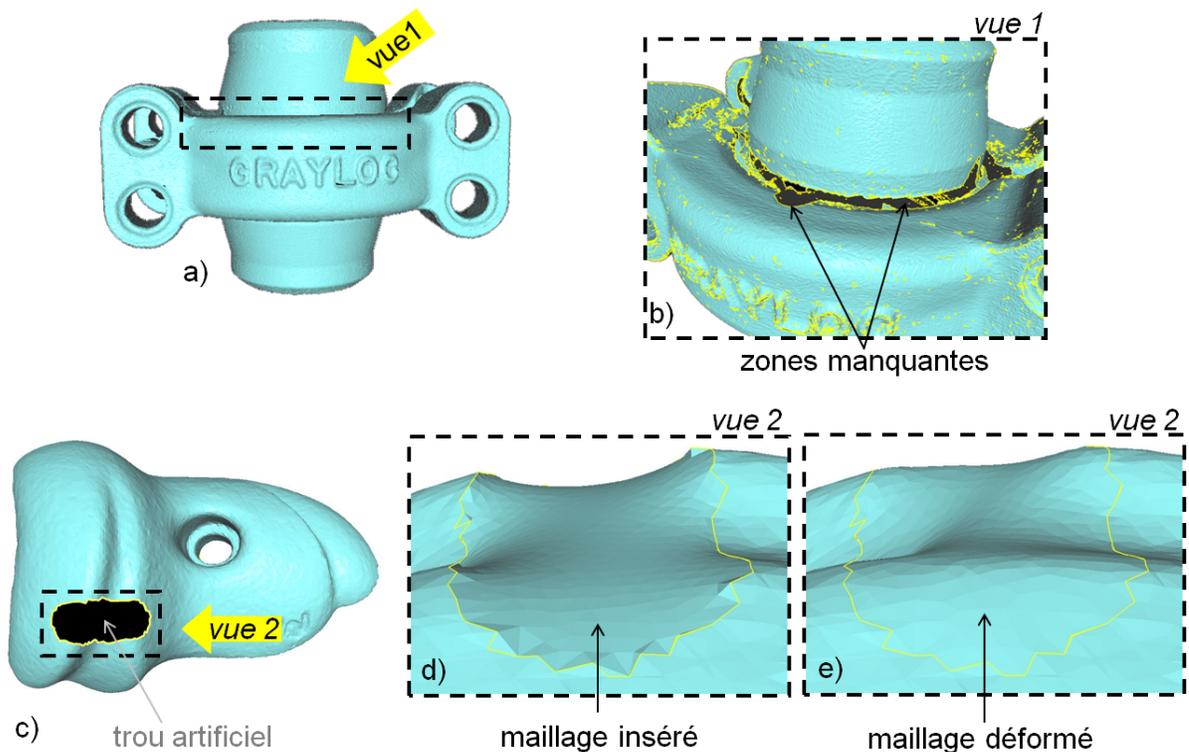


FIGURE 3.2 – Un modèle scanné (*courtoisie du CNR-IMATI et de l'INRIA*) d'une pièce mécanique (a) comporte des trous dus au bruit du laser (b). Un trou artificiel (c) a été créé dans un modèle de prise d'escalade (*courtoisie de Tomoadour*) qui a été rempli par un maillage inséré (d) et déformé pour satisfaire des conditions spécifiques de raccord (e).

3.2 Techniques de simplification de modèles polyédriques

La réduction du volume de données mise en jeu pour la représentation de produits est un véritable enjeu industriel et un défi scientifique. A titre d'exemple, la maquette numérique globale de l'Airbus A400M représente environ quinze téraoctets de données. Selon le champ d'application, pouvant aller de la visualisation des modèles polyédriques, aux simulations d'assemblages de pièces mécaniques ou d'analyse par Élément Finis, la maquette numérique doit être filtrée et adaptée aussi rapidement que possible. Ainsi, la simplification pertinente et adaptée des modèles polyédriques sous-jacents devient un véritable enjeu afin de permettre l'indexation, le stockage, la manipulation et la maintenance des informations. La simplification rend les modèles polyédriques exploitables pour un métier et/ou une tâche donnés.

La simplification d'un modèle géométrique consiste à réduire le nombre de données (nombre de faces d'un modèle polyédrique) nécessaires pour représenter l'objet selon un ou plusieurs critères. Ces critères sont en général spécifiés par l'utilisateur et ils dépendent de l'activité qui va être menée ensuite. Ainsi, il peut être intéressant de préserver la forme globale de l'objet, de minimiser la variation de son volume, de produire des faces les plus équilatérales possibles, de créer une enveloppe de l'objet, etc.... Concernant les méthodes préservant les formes de l'objet, certaines imposent le respect des **lignes de caractère** de l'objet. Ainsi, développer des techniques de simplification qui préservent ces éléments clés contribue à la définition d'une maquette numérique multi-résolution construite autour de la notion de *features* [Shah *et al.* 1995, Pernot 2004]. On entend par *features*, les formes (portions de la géométrie) et des caractéristiques du modèle auxquelles on peut associer certains attributs et connaissances utiles pour la conception, la fabrication, la simulation.

L'étude des techniques de simplification de modèles polyédriques peut être réalisée suivant deux directions principales : la première concerne les mécanismes de simplification et la seconde les techniques de simplification qui essaient de préserver les lignes de caractère [FIORES II, Catalano *et al.* 2002].

3.2.1 Mécanismes de simplification de modèles polyédriques

Classiquement on peut regrouper les mécanismes de simplification en trois catégories principales selon la manière dont la géométrie est simplifiée [Heckbert *et al.* 1997, Luebke 2001] :

- Les algorithmes de **subdivision adaptative** consistent à subdiviser récursivement un modèle de base simple jusqu'à ce qu'il approche le modèle original selon un critère défini par l'utilisateur [DeHaemer *et al.* 1991], [Eck *et al.* 1995], [Turk 1992]. Ces algorithmes sont bien adaptés lorsque le modèle de base est connu ou facile à construire. Par exemple, le modèle de base des modèles de terrain est un simple rectangle. Ils ne sont clairement pas adaptés à la simplification de pièces mécaniques dont les formes sont beaucoup plus complexes.
- Les algorithmes de **suppression de géométrie** enlèvent récursivement certaines faces et remaillent les zones concernées [Algorri *et al.* 1996], [Hinker *et al.* 1993], [Hoppe 1996], [Schroeder *et al.* 1992], [Véron *et al.* 1998]. Les opérations de suppression de faces peuvent provenir soit d'opérations de suppression de sommets, soit de contraction d'arêtes. Le processus est répété jusqu'à atteindre un critère d'arrêt, e.g. un ratio de décimation à atteindre ou un écart maximal entre la géométrie initiale et la géométrie simplifiée. Ces paramètres de contrôle peuvent être difficilement maîtrisables et les effets difficilement prévisibles. De tels algorithmes sont intéressants pour la simplification de modèles de formes très complexes qui requièrent une simplification drastique, ce qui est crucial dans le domaine de la conception mécanique. La plupart n'autorisent pas les opérations de suppression de sommets ou de faces générant un changement de topologie. Cette propriété est plus importante si le modèle simplifié sert de donnée d'entrée à une analyse Éléments Finis que pour un usage de visualisation pure.
- Les algorithmes d'**échantillonnage** extraient la géométrie du modèle original à partir de points aléatoirement distribués sur sa surface. Ils peuvent également utiliser une grille de voxels 3D [Garland *et al.* 1992, Rossignac *et al.* 1992]. Ces techniques comptent parmi les plus sophistiquées et difficiles à implémenter. Elles échouent à représenter efficacement les surfaces parsemées de détails ou de pics dont la fréquence d'apparition est supérieure à la fréquence d'échantillonnage. Elles conviennent en revanche, au cas des modèles organiques lisses sans arêtes vives. Cette catégorie d'algorithmes est donc moins appropriée aux besoins de la conception mécanique.

Différentes techniques de simplification tentent de réduire le volume de données tout en essayant de préserver les caractéristiques des formes. Ces algorithmes peuvent être classés en deux catégories : les approches géométriques et les approches perceptuelles.

3.2.2 Simplifications géométriques

Les approches *géométriques* requièrent le calcul de quantités géométriques locales sur l'ensemble du modèle polyédrique à simplifier qui peut être constitué de plusieurs centaines de millions de faces.

[Garland *et al.* 1992] utilise une métrique basée sur la mesure de l'erreur quadratique pour assigner un coût de simplification aux paires de sommets adjacents ou proches géométriquement. La contraction de paires de sommets qui ne correspondent pas nécessairement aux arêtes du maillage est donc permise.

La métrique d'erreur quadratique pour un sommet du modèle initial est calculée en mesurant de combien ce sommet "s'éloigne" des plans définis par les faces auxquelles il appartient. Ainsi, un sommet u qui se retrouve au point w , entraîne une erreur (fig. 3.3) :

$$e(u \rightarrow w) = \sum_F d^2(w, P(F)) \quad (3.1)$$

où F est l'ensemble des faces adjacentes à u et $P(F)$, les plans définis par les faces de F et $d(w, P)$, la distance euclidienne du point w au plan P .

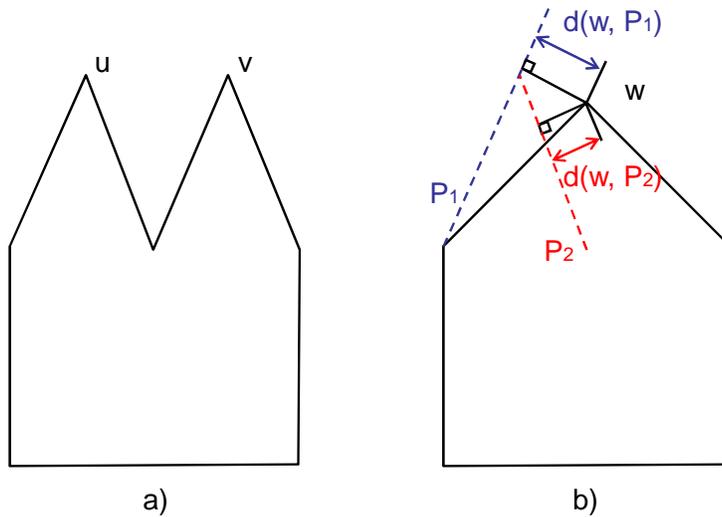


FIGURE 3.3 – L'erreur pour le sommet u (a) est la somme du carré des distances de u aux plans P_1 et P_2 (b).

Initialement, on calcule pour chaque arête le meilleur point où la contracter (c'est-à-dire celui qui minimise la métrique d'erreur), et c'est sa métrique associée qui est affectée à l'arête comme coût de contraction. Les coûts sont ensuite utilisés pour piloter un algorithme itératif. A chaque étape, l'arête de coût minimum est contractée.

[Yoshizawa *et al.* 2005] tente de préserver les lignes de caractère en utilisant une estimation d'un tenseur de courbure (matrice diagonalisée dont les valeurs propres sont une approximation des courbures principales et les vecteurs propres, une approximation des directions principales) pour détecter ce que [Yoshizawa *et al.* 2005] appelle les *lignes de crêtes*.

[Ha Lee *et al.* 2005] utilise un opérateur centré basé sur un critère de courbure gaussienne moyenne affectée d'un poids lié à la salience² inspiré de [Taubin 1995].

Simpoly©

Parmi les méthodes de simplification dites géométriques, la méthode de [Véron 1997] est basée sur un processus de suppression de sommet itératif mené par deux critères géométriques [Véron *et al.* 1997, Lesage *et al.* 2001] : le critère de zone d'erreur qui impose que le modèle simplifié reste à l'intérieur d'une enveloppe de sphères (fig. 3.4.b) associées aux sommets et définies par l'utilisateur et un critère optionnel de courbure discrète utilisé pour contrôler le remaillage à chaque suppression.

Les principales étapes de l'algorithme de simplification de Simpoly© sont les suivantes :

1. Initialisations

- initialisation des sphères d'erreur pour chaque sommet (fig. 3.5). Les rayons ne sont pas forcément égaux les uns aux autres pour permettre une simplification hétérogène du modèle. Il en résulte une "*carte de tolérances*".
- initialisation de la listes des dépendances de la carte de tolérances. Pour chaque face du modèle polyédrique, une liste est créée et contient les sphères d'erreur qui intersectent la face (fig. 3.5). Pour une face triangulaire, la liste contient initialement au moins trois sphères d'erreur.
- initialisation de la liste ordonnée des sommets L_v qui doivent être supprimés. Tous les sommets sont initialement étiquetés "suppri-

2. *Mesh saliency* = métrique d'importance régionale basée sur l'évaluation des courbures gaussiennes [Ha Lee *et al.* 2005]

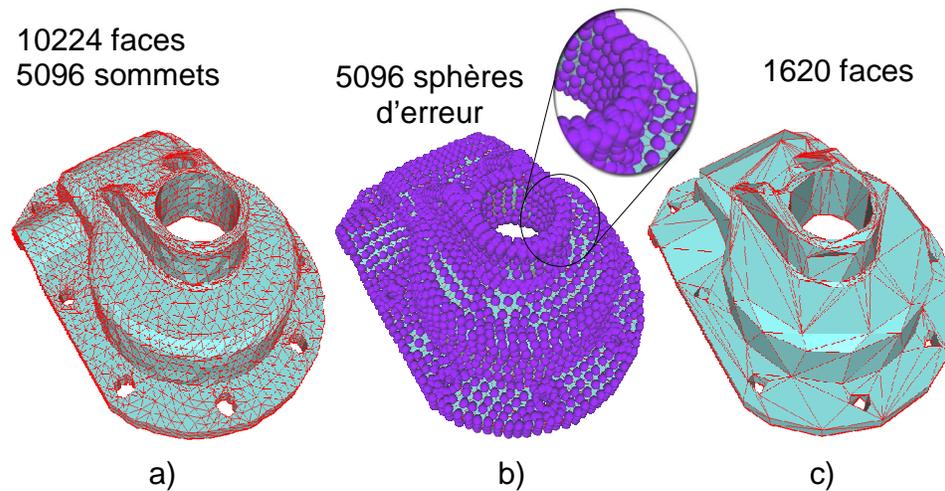


FIGURE 3.4 – Polyèdre initial (a), l'enveloppe formée des sphères d'erreur (b) et le polyèdre simplifié (c).

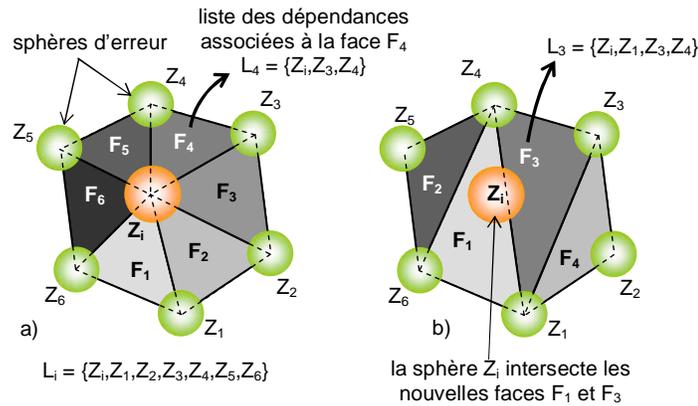
mables".

2. Boucle de traitement des sommets "supprimables" contenus dans la liste L_v - Simplification des V_i

- création de la liste L_i des sphères d'erreur qui concatène les listes des dépendances associées à toutes les faces connectées à V_i . Cette liste contient les sphères Z_i associées à V_i (fig. 3.5.a).
- suppression du sommet V_i et des faces qui lui sont connectées.
- remplissage du trou en utilisant le critère de courbure discrète [Véron *et al.* 1997] qui tend à préserver la distribution de courbure du modèle initial (fig. 3.5.b). Ce faisant, le processus de remplissage connecte en premier les sommets qui ont les plus grandes courbures discrètes.
- test si la simplification est conforme (fig. 3.5.b) :
 - toutes les sphères d'erreur L_i intersectent au moins une des faces nouvellement insérées, la simplification est conforme au vu du respect du critère de zone d'erreur. De nouvelles listes de dépendances sont assignées aux nouvelles faces. Puisque la nouvelle configuration peut influencer sur les sommets précédemment testés, l'ensemble des sommets entourant V_i sont à nouveau étiquetés "supprimables".
 - sinon, la simplification est non conforme et le sommet V_i est étiqueté "non supprimable". Dans ce cas, le modèle polyédrique et les listes de dépendances de l'itération précédente sont rétablis.

3. Fin de la simplification.

Le processus s'achève quand il n'y a plus

FIGURE 3.5 – Principe de la simplification d'un sommet V_i .

aucun sommet étiqueté "supprimable".

Avantages de Simpoly©

La méthode de simplification de Simpoly© possède les avantages suivants :

- parmi les approches présentées dans [Heckbert *et al.* 1997], cette méthode de simplification possède l'avantage d'être pilotée par un critère géométrique (une sphère d'erreur affectée aux sommets) ayant une signification pour un concepteur et/ou une précision donnée (fig. 3.6). Ceci est aussi le cas pour [DeHaemer *et al.* 1991], [Hinker *et al.* 1993], [Varshney 1994]. Au contraire, les méthodes développées dans [Eck *et al.* 1995], [Turk 1992], [Hoppe 1996], [Algorri *et al.* 1996], [Schroeder *et al.* 1992], [Rossignac *et al.* 1992] essaient de trouver un modèle polyédrique avec un nombre prérequis de faces (ou de sommets) et qui préserve autant que possible la géométrie du modèle original. Un tel paramètre de contrôle est moins adapté aux besoins actuels en conception mécanique. De plus, [DeHaemer *et al.* 1991] ne peuvent simplifier des modèles arbitraires car ils sont dédiés aux modèles de DEM³ dont ils exploitent les propriétés pour évaluer l'erreur entre le modèle initial et le modèle simplifié. En effet, chaque sommet du modèle peut être localisé par deux coordonnées définies dans un plan de référence et par une valeur d'élévation ou de hauteur par rapport à ce plan. La méthode de [Hinker *et al.* 1993] requiert le calcul et le stockage de modèles po-

3. Digital Elevation Models, modèles de terrains en français

lyédriques parallèles, tout comme celle de [Varshney 1994] qui est en outre mieux adaptée aux modèles de faible courbure. A contrario, la méthode de [Véron 1997] peut simplifier des modèles arbitraires et à forte courbure sans utilisation de modèles intermédiaires.

- les sphères d'erreur représentent une incertitude sur la position 3D des sommets auxquels elles sont associées. L'ensemble formé par les sphères d'erreur représente une discrétisation de l'enveloppe du modèle simplifié dans lequel il est restreint. Le mécanisme d'héritage des sphères d'erreur durant le processus de simplification permet le contrôle de la préservation des formes. Il consiste à vérifier que chaque sphère d'erreur a bien une intersection avec le modèle simplifié.
- la combinaison de différentes techniques de simplification basées sur un critère spécifique fournit des paramètres de contrôle qui peuvent être utilisés pour améliorer la forme du modèle résultant.

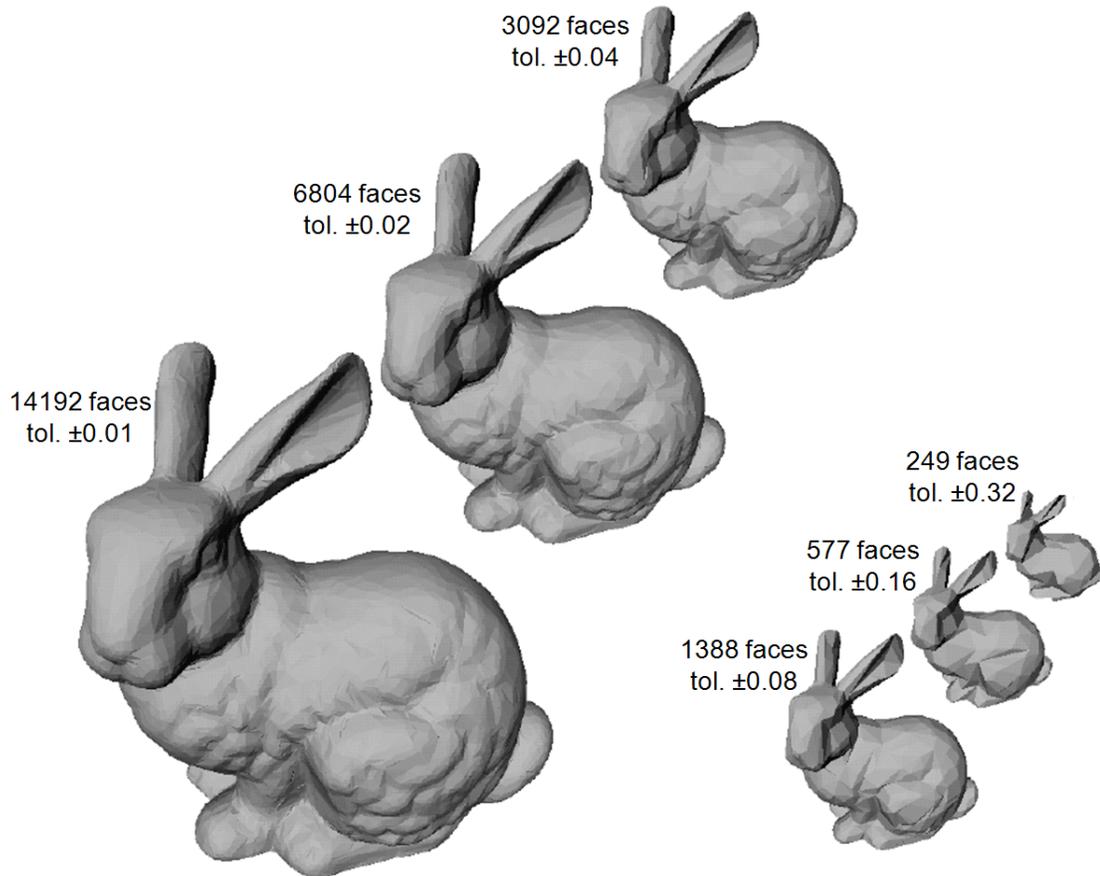


FIGURE 3.6 – Modèles simplifiés du bunny de Stanford obtenus en faisant varier une tolérance homogène de simplification (qui correspond à la taille des sphères d’erreur) et affichés avec un facteur d’échelle adapté [Véron 1997].

3.2.3 Simplifications perceptuelles

Les approches *perceptuelles* sélectionnent les éléments géométriques à supprimer (sommets, arêtes, faces) en utilisant un critère comme les variations de visibilité [Zhang *et al.* 2002], de luminance [Lindstrom *et al.* 2000], de texture [Williams *et al.* 2003] entre des captures d’écran du modèle original et du modèle simplifié. Certaines techniques utilisent des images et des capteurs optiques pour calculer les modèles simplifiés.

L’algorithme de [Lindstrom *et al.* 2000] est le premier à piloter un processus de simplification directement à partir d’une métrique liée à des images. De la même manière que l’algorithme de [Garland *et al.* 1992], la simplifica-

tion est basée sur un processus itératif de suppression d'arêtes. Le coût d'une suppression d'arête est déterminé en supprimant l'arête et en mesurant de combien le modèle simplifié ainsi obtenu varie visuellement du modèle initial.

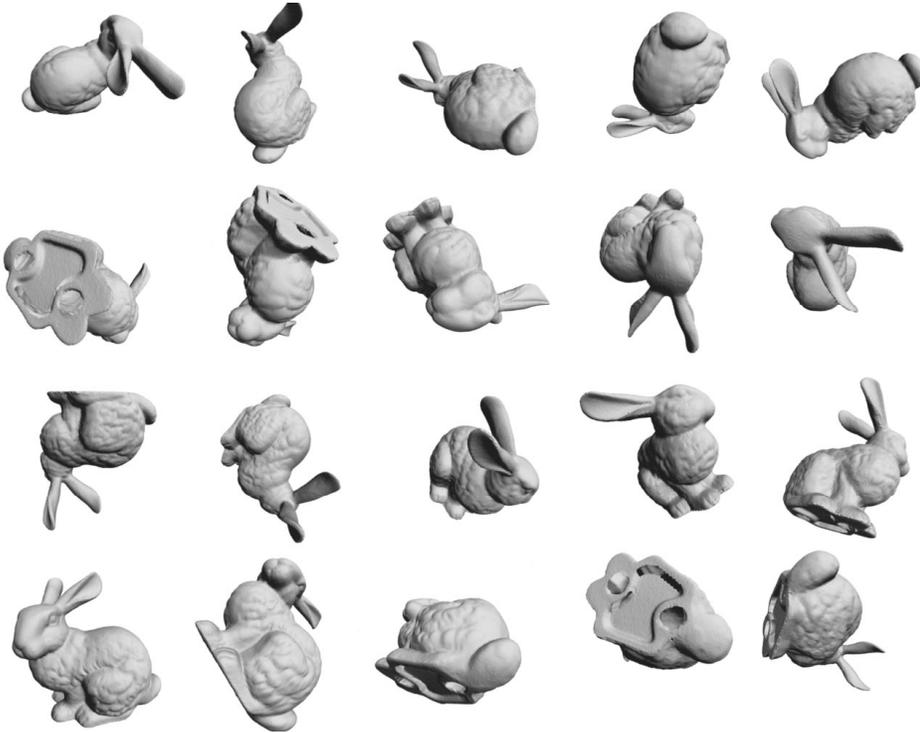


FIGURE 3.7 – Vingt captures d'écran correspondant à vingt points de vue différents du modèle initial du bunny de Stanford, utilisées durant le processus de simplification de [Lindstrom *et al.* 2000].

Soit $Init$ l'ensemble des k captures d'écran du modèle initial (fig. 3.7) et $Simp$ l'ensemble des k captures d'écran du modèle simplifié, tels que $Init = \{I^1, I^2, I^3, \dots, I^k\}$ et $Simp = \{S^1, S^2, S^3, \dots, S^k\}$, avec $I^h = \{I_{ij}^h\}$ la $h^{ième}$ image du modèle initial et $S^h = \{S_{ij}^h\}$, la $h^{ième}$ image du modèle simplifié. Chaque capture d'écran est composée de $n \times m$ pixels, et $i \in [1..n]$ et $j \in [1..m]$.

Le coût de contraction d'une arête est égal à la *déviatio*n visuelle entre le modèle initial et le modèle simplifié obtenu en supprimant cette arête. Pour calculer cette déviation, l'algorithme mesure la somme des erreurs moyennes au carré de la luminance entre les pixels telle que :

$$d_{RMS}(Init, Simp) = \sqrt{\frac{1}{kmn} \sum_{h=1}^k \sum_{i=1}^n \sum_{j=1}^m (I_{ij}^h - S_{ij}^h)^2} \quad (3.2)$$

Les arêtes sont triées dans l'ordre croissant de leur coût de contraction. Le calcul du coût des arêtes ralentit clairement le processus, à cause du temps d'affichage des différents modèles simplifiés et requiert une taille en mémoire pour stocker les modèles 3D et les images.

[Zhang *et al.* 2002] pilote également un processus itératif de suppression d'arête dont le coût de contraction est calculé en combinant la métrique d'erreur quadratique de [Garland *et al.* 1992] avec une métrique dite de *visibilité*. L'objet est visible depuis un ensemble de caméras dont le champ de visibilité définit "l'espace des caméras". La métrique de visibilité représente le pourcentage de l'espace des caméras depuis lequel un point de la surface de l'objet est visible, ce pourcentage étant pondéré par le produit scalaire de la normale à la surface au point considéré avec la direction de visualisation de la caméra. Cette métrique combinée à la métrique d'erreur quadratique autorise la suppression d'arêtes produisant une erreur géométrique importante si elles sont par ailleurs situées dans des régions très peu visibles, à l'intérieur de l'objet ou dans des zones complexes de type "replis" de la surface.

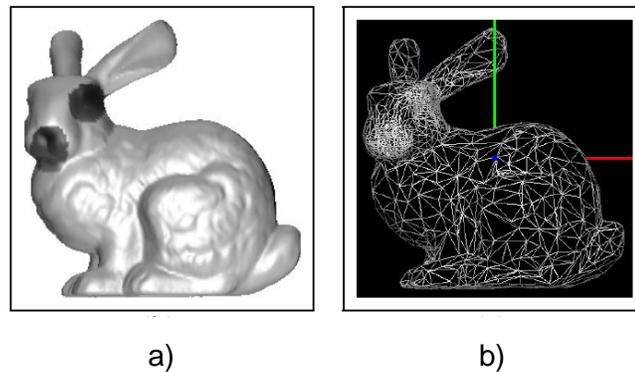


FIGURE 3.8 – Les zones de grande importance perceptuelle (en gris sombre) sur le modèle du bunny de Stanford (a) et le modèle simplifié (b) avec la méthode de [Mata *et al.* 2007].

Enfin, des méthodes semi-automatiques ont été proposées [Ho *et al.* 2006, Mata *et al.* 2007]. Dans [Ho *et al.* 2006], l'utilisateur peut raffiner localement une simplification réalisée par des méthodes automatiques [Garland

et al. 1992]. Une pondération calculée à partir de ce raffinement est utilisée pour directement réordonner les arêtes : le critère d'ordonnement n'est plus le coût de la contraction de l'arête candidate à la suppression, mais un poids extrait des raffinements locaux réalisés par l'utilisateur. Dans [Mata *et al.* 2007], les zones d'intérêt du modèle sur lesquelles les yeux de l'utilisateur se sont focalisés le plus longtemps, sont affectées d'une "importance perceptuelle" haute (fig. 3.8.a.) qui est ensuite utilisée comme critère pour piloter un processus itératif de suppression de sommets.

3.2.4 Synthèse des techniques de simplification de modèles polyédriques

Toutes les approches présentées dans la section 3.2 comportent des inconvénients, notamment en terme de temps d'exécution. En effet, les méthodes basées sur des critères liés à la notion de courbure souffrent de leur lenteur due au calcul de la courbure discrète en chaque sommet du maillage à simplifier. De plus, les estimateurs de courbure ne permettent pas facilement de détecter les principales formes caractéristiques d'un polyèdre. En effet, [Turk 1992] a ainsi prouvé que la courbure ne prenait pas en compte l'importance "régionale". Ainsi, une fine excroissance de forte courbure à l'intérieur d'une zone à faible courbure sera systématiquement considérée comme une forme caractéristique importante. Les méthodes perceptuelles, qui sont basées sur la comparaison de captures d'écran du modèle original et des modèles simplifiés ont des temps d'exécution importants. En effet, elles doivent :

- calculer itérativement la simplification,
- afficher les modèles simplifiés obtenus,
- effectuer la comparaison entre les images originales et finales,
- réaliser itérativement la simplification en tenant compte du nouveau coût de contraction des arêtes.

Ces temps d'exécution sont d'autant plus importants quand on manipule des modèles polyédriques constitués de plusieurs centaines de millions de faces. La méthode de [Véron 1997] implémentée dans le logiciel Simpoly© est donc adaptée à notre étude de part ses avantages généraux (section 3.2.2) et en particulier la possibilité qu'elle offre de piloter le processus de simplification avec un critère extrait d'images par le biais du concept de carte de tolérances.

3.3 Techniques de déformation de modèles polyédriques

Les techniques de déformation de modèles polyédriques peuvent être regroupées en trois grandes catégories. Cette sous-section vise à présenter les grandes catégories d'approches pour introduire celle qui sera utilisée et améliorée par la suite (voir [Pernot 2004] pour un état de l'art complet des méthodes de déformation de modèles géométriques) :

- les méthodes **géométriques** permettent la modification des modèles à l'aide d'une fonction de déformation, suivant un vecteur de déplacement, le long de directions privilégiées ou par le biais d'un volume d'influence,
- les méthodes **physiques** qui permettent la modification des modèles à partir d'un modèle physique qui simule le comportement d'objets réels (loi de comportement de matériaux, etc.),
- les méthodes **mixtes** qui utilisent à la fois des méthodes géométriques et des approches physiques de déformation. Le but est de simuler un comportement physique en utilisant une fonction d'énergie ou un modèle mécanique couplé aux sommets du modèle polyédrique.

3.3.1 Méthodes géométriques

Les méthodes géométriques déforment le modèle polyédrique par le biais de la déformation de l'espace (ou une partie de l'espace) dans lequel il est immergé. Dans cette catégorie de méthodes, un vecteur de déplacement et une fonction de déformation spécifiés par l'utilisateur définissent le comportement de la déformation autour des points déplacés [Borrel *et al.* 1991]. Soit un point P de l'espace 3D dans lequel le modèle est immergé, le déplacement $d(P)$ de P est donné par l'équation :

$$d(P) = \sum_{i=1}^{n_c} M_i \cdot f_i(P) \quad (3.3)$$

où n_c représente le nombre de contraintes, chaque contrainte i étant caractérisée par sa direction M_i et sa fonction intensité (de déformation) f_i .

Soit D_j le vecteur de déplacement du point contraint C_j , l'équation 3.3 devient :

$$d(C_j) = D_j = \sum_{i=1}^{n_c} M_i \cdot f_i(C_j), \forall j \in 1..n_c \quad (3.4)$$

Les fonctions intensité f_i étant spécifiées pour chaque point de contraintes, les directions \mathbf{M}_i sont calculées en résolvant le système formé des $(3 \cdot n_c \times 3 \cdot n_c)$ équations 3.4. Ainsi, la déformation du modèle est obtenue à partir des équations 3.3 pour chaque point du modèle. Selon la nature des fonctions de déformation utilisées (B-Spline, polynomiales etc...), des comportements locaux ou globaux différents sont obtenus.

Des améliorations [Borrel *et al.* 1994] ont été proposées en ajoutant le concept de *rayon d'action* R_i associé à chaque contrainte C_i , la fonction d'intensité étant nulle pour les points situés en dehors de la sphère définie par ce rayon d'équation (3.5), (fig. 3.3.1) :

$$f_i(\mathbf{P}) = B_i \cdot \left(\frac{\|\mathbf{P} - C_i\|}{R_i} \right) \quad (3.5)$$

B_i est une fonction de base BSpline centrée en 0, décroissante, maximale et égale à 1 en 0.

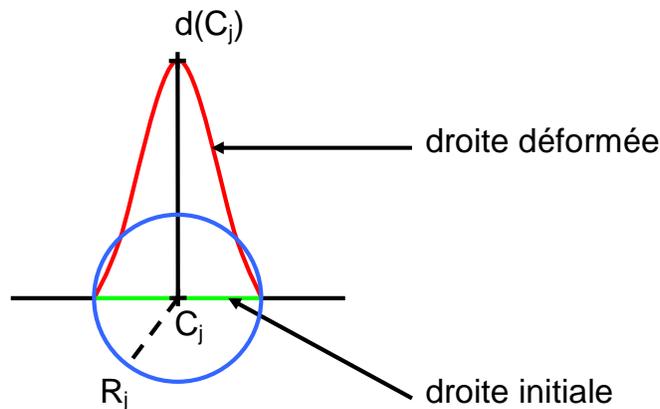
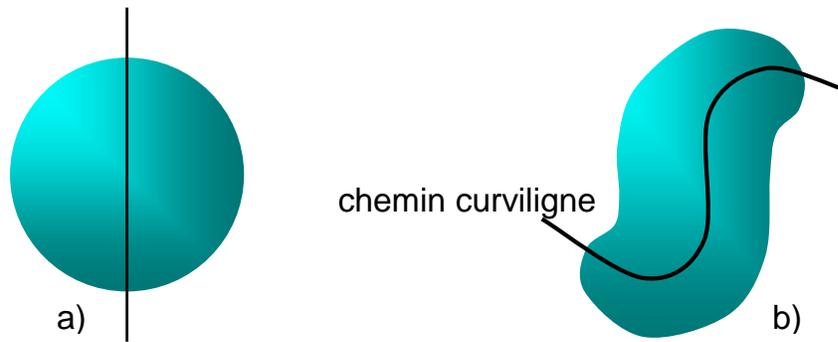


FIGURE 3.9 – Exemple de déformation d'une droite [Borrel *et al.* 1994].

D'autres méthodes ont étendu ce concept à la notion d'enveloppe d'influence superquadrique définissant un volume d'influence par une fonction implicite [Neveu *et al.* 1995, Raffin *et al.* 1998].

[Lazarus *et al.* 1994] propose une méthode consistant à définir un chemin curviligne (courbe B-Spline, coïncidant par exemple avec l'axe central du modèle). En modifiant la forme du chemin, l'espace cylindrique l'entourant ainsi que le modèle ou une partie du modèle immergé sont modifiés (fig.3.3.1). Des approches ont étendu le concept de chemin de déformation [Feng *et al.* 1996, Ma *et al.* 1997] à l'utilisation de surfaces de déformation.

FIGURE 3.10 – Exemple de déformation axiale [Lazarus *et al.* 1994].

Enfin, [Sederberg *et al.* 1986] propose une méthode de Déformation de Formes Gauches (*Free Form Deformation*, FFD) qui permet de déformer des modèles immergés dans un parallélépipède élastique appelé "bloc FFD", représenté par un volume de Bézier. Les manipulations du volume entraînent la modification des modèles. Des extensions de cette méthode ont été développées permettant la définition de volumes englobants de formes complexes [Coquillart 1990, MacCracken *et al.* 1996]. Une méthode définie par [Kobayashi *et al.* 2003] permet la déformation d'un modèle polyédrique par le biais de la modification d'un modèle polyédrique de contrôle où chaque point du modèle original est paramétré dans l'espace défini par chaque face du polyèdre de contrôle.

Pour permettre une déformation globale et complexe préservant des détails du modèle initial, de nouvelles méthodes proposent de considérer le processus de déformation comme un problème d'optimisation numérique avec la minimisation d'une fonction d'énergie de déformation ([Sorkine *et al.* 2004, Zhou *et al.* 2005, Shi *et al.* 2007, Lipman *et al.* 2007]). La fonction d'énergie de déformation utilisée contient des termes permettant la préservation de détails et des termes permettant une manipulation directe sous la forme de positions *cibles* (fig.3.12). L'ajout de contraintes relatives à l'équilibre, la longueur, la rigidité, etc. du modèle original permet un meilleur contrôle de la déformation au détriment des temps de résolution du problème d'optimisation qui augmentent.

3.3.2 Méthodes non géométriques

Les **approches physiques** incorporent des principes physiques au sein des modèles géométriques et ont été initialement introduites par

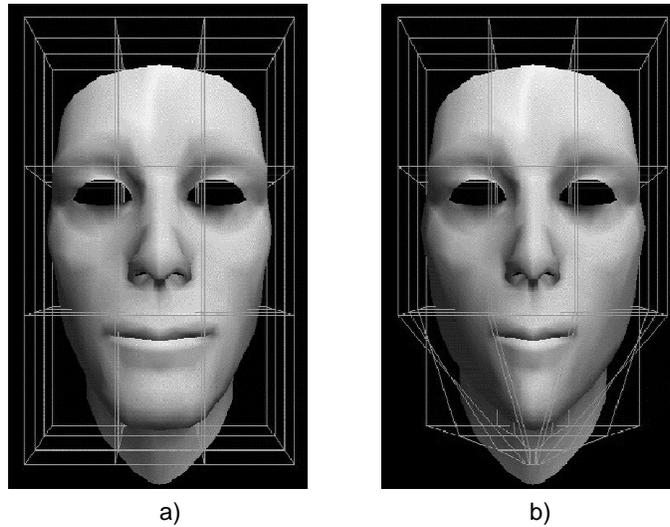


FIGURE 3.11 – Exemple de déformation de forme gauche incluse dans un parallélépipède : l’objet est déformé par déplacement des points de contrôle du volume englobant [Sederberg *et al.* 1986].

[Terzopoulos *et al.* 1987]. Les principes physiques régissent la manipulation de forces, de couples, d’énergies et autres quantités physiques. Des comportements élastiques, isotropiques ou anisotropiques sont possibles.

Une première catégorie d’approches rassemble les méthodes dédiées aux B-Spline [Bloor *et al.* 1990], Nurbs [Qin *et al.* 1996], surfaces de subdivision [Lanquetin *et al.* 2005](fig.3.13), déformables. [Terzopoulos *et al.* 1994] étend le principe au cas de la Déformation de Formes Gauches où le volume de contrôle n’est plus modifié par le déplacement de ses points mais par l’application de forces extérieures sur le volume global.

Enfin, les **approches hybrides** utilisent un modèle mécanique afin de réduire les manipulations et les temps d’exécution. Une première catégorie de méthodes, [Sarraga 2004], se réduit à la minimisation d’une fonction d’énergie. La dernière catégorie rassemble les méthodes *masse-ressort* et les méthodes relatives à l’utilisation d’une *densité de force*, qui couplent un modèle mécanique simplifié au modèle polyédrique à déformer. Les méthodes à base de modèle *masse-ressort* (fig.3.14) [Thingvold *et al.* 1990, Hirota *et al.* 1999] obtiennent la déformation du modèle soit en appliquant des forces extérieures aux sommets du réseau soit en faisant varier la raideur des ressorts.

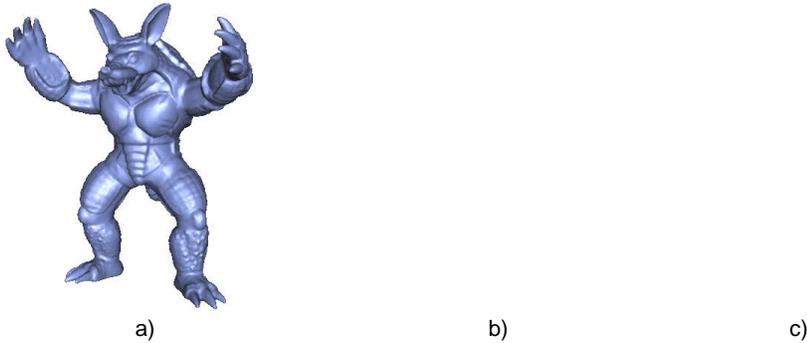


FIGURE 3.12 – Exemple de déformation obtenu par minimisation d'énergie de déformation utilisée par [Shi *et al.* 2007] et de contrainte de contrôle d'équilibre.

Méthode des densités de forces

La méthode de densité de force (*Force Density Method*), a été développée par [Schek 1974] afin de calculer l'équilibre statique d'un réseau quelconque.

Modèle mécanique de réseau de barres

Pour modifier la forme d'un réseau, une solution consiste à déplacer un par un les noeuds libres. La figure 3.15.a montre comment la forme d'un réseau peut être modifiée en déplaçant un seul noeud numéroté 2. Ces manipulations basiques prennent malheureusement beaucoup de temps puisqu'elles nécessitent le déplacement de tous les noeuds libres avant d'atteindre la forme désirée.

Pour surmonter ces problèmes, un modèle mécanique de réseau de barres est couplé au modèle géométrique. Ainsi, les sommets du modèle géométrique et les connexions entre eux sont associés respectivement aux noeuds et barres du réseau de barres. Chaque barre peut être assimilée à un ressort qui aurait une longueur à vide nulle et une raideur q_i (plus précisément une densité de force). Pour préserver l'équilibre statique de la structure, des efforts extérieurs f_i doivent être appliqués aux noeuds (fig.3.15.b). Si ces forces n'étaient pas appliquées, tous les noeuds seraient confondus en un seul point. La linéarité des relations entre les efforts et les positions des noeuds permet des modifications intuitives de la forme par l'intermédiaire de manipulations des efforts. Sur l'exemple de la figure 3.15.c, seule la force extérieure appliquée au noeud numéro 2 est modifiée pour produire le déplacement de tous les autres noeuds libres.

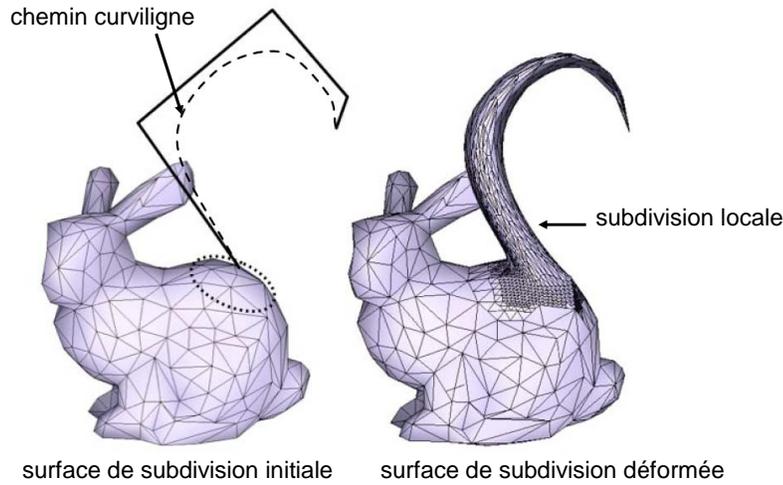


FIGURE 3.13 – Déformation d’une surface de subdivision du modèle Stanford bunny, suivant un chemin curviligne et avec une subdivision locale du maillage [Lanquetin *et al.* 2005].

Formalisation :

Étant donnés \mathbf{x} , \mathbf{y} et \mathbf{z} , les trois vecteurs contenant les composantes des coordonnées 3D des N_n noeuds du réseau de barres couplé aux sommets du modèle géométrique, les composantes f_x , f_y et f_z des efforts extérieurs appliqués à ces noeuds peuvent être obtenues à l’aide des $(3 \times N_n)$ équations suivantes qui expriment l’équilibre statique du réseau de barres :

$$\begin{aligned} \mathbf{f}_x &= ({}^t\mathbf{C} \cdot \mathbf{Q} \cdot \mathbf{C}) \cdot \mathbf{x}, \\ \mathbf{f}_y &= ({}^t\mathbf{C} \cdot \mathbf{Q} \cdot \mathbf{C}) \cdot \mathbf{y}, \\ \mathbf{f}_z &= ({}^t\mathbf{C} \cdot \mathbf{Q} \cdot \mathbf{C}) \cdot \mathbf{z}, \end{aligned} \quad (3.6)$$

où \mathbf{Q} est la matrice de densité de force de taille $(N_b \times N_b)$, N_b étant le nombre de barres. $Q_{ij} = q_j \cdot \delta_{ij}$ avec δ_{ij} le symbole de Kroenecker et $q_j = f_j/l_j$ la densité de force dans la $j^{\text{ème}}$ barre de longueur l_j . \mathbf{C} est une matrice de taille $(N_b \times N_n)$ exprimant la connexion des noeuds [Pernot 2004]. En suivant [Pernot *et al.* 2005], une distinction entre les noeuds libres et les noeuds bloqués peut être effectuée au sein de l’équation 3.6. Ceci mène à l’ensemble d’équations :

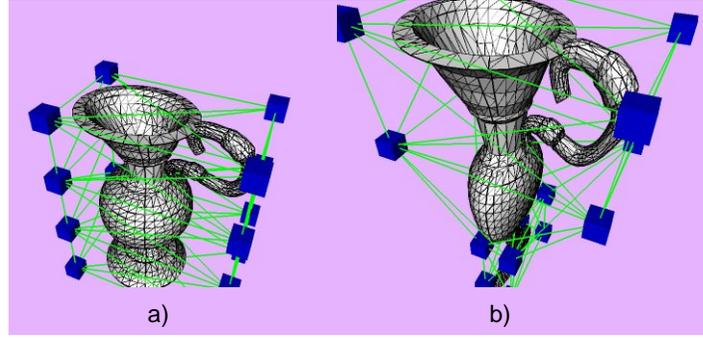


FIGURE 3.14 – Le modèle déformé (b) par la méthode de [Hirota *et al.* 1999] basée sur un système masse/ressort préservant le volume du modèle d'un pichet d'eau (a). Les ressorts correspondent aux segments verts, les masses affectées aux sommets correspondent aux cubes bleus.

$$\begin{aligned}
 \mathbf{f}_{fn_x} &= \mathbf{D}_f \cdot \mathbf{x}_{fn} + \mathbf{D}_{bf} \cdot \mathbf{x}_{bn}, & \mathbf{f}_{bn_x} &= {}^t\mathbf{D}_{bf} \cdot \mathbf{x}_{fn} + \mathbf{D}_b \cdot \mathbf{x}_{bn}, \\
 \mathbf{f}_{fn_y} &= \mathbf{D}_f \cdot \mathbf{y}_{fn} + \mathbf{D}_{bf} \cdot \mathbf{y}_{bn}, & \mathbf{f}_{bn_y} &= {}^t\mathbf{D}_{bf} \cdot \mathbf{y}_{fn} + \mathbf{D}_b \cdot \mathbf{y}_{bn}, \\
 \mathbf{f}_{fn_z} &= \mathbf{D}_f \cdot \mathbf{z}_{fn} + \mathbf{D}_{bf} \cdot \mathbf{z}_{bn}, & \mathbf{f}_{bn_z} &= {}^t\mathbf{D}_{bf} \cdot \mathbf{z}_{fn} + \mathbf{D}_b \cdot \mathbf{z}_{bn},
 \end{aligned} \quad (3.7)$$

Les équations (3.7) permettent de lier les efforts aux noeuds libres f_{fn} et aux noeuds fixes f_{bn} à la position des noeuds libres et des noeuds fixes. Les différentes matrices D_i sont obtenues par décomposition de la matrice tCQC .

Inversement, étant donné un ensemble d'efforts extérieurs appliqués aux noeuds du réseau de barres, la position des noeuds libres s'obtient à l'aide des relations :

$$\begin{aligned}
 \mathbf{x}_{fn} &= (\mathbf{D}_f)^{-1} \cdot (\mathbf{f}_{fn_x} - \mathbf{D}_{bf} \cdot \mathbf{x}_{bn}), \\
 \mathbf{y}_{fn} &= (\mathbf{D}_f)^{-1} \cdot (\mathbf{f}_{fn_y} - \mathbf{D}_{bf} \cdot \mathbf{y}_{bn}), \\
 \mathbf{z}_{fn} &= (\mathbf{D}_f)^{-1} \cdot (\mathbf{f}_{fn_z} - \mathbf{D}_{bf} \cdot \mathbf{z}_{bn}).
 \end{aligned} \quad (3.8)$$

Ces dernières équations montrent comment il est possible de modifier la position des noeuds par l'intermédiaire de manipulations des efforts extérieurs (voir [Pernot 2004] pour le traitement des configurations où la matrice D_f est singulière). Les inconnues du processus de déformation ne sont plus les positions des noeuds mais les efforts extérieurs eux-mêmes.

Certaines forces appliquées sur des noeuds bloqués connectés à au moins une barre libre, *i.e.* une barre dont la longueur change durant le processus, sont susceptibles de varier durant la déformation. Elles peuvent donc prendre part à la définition de la fonction objectif à minimiser. Elles sont

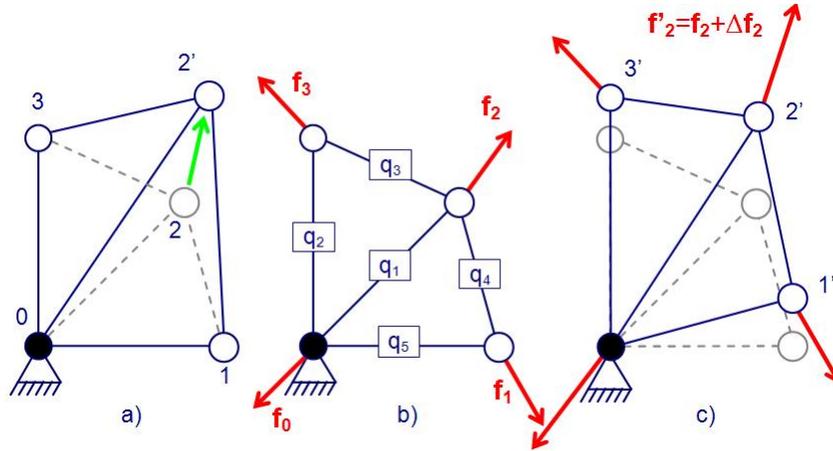


FIGURE 3.15 – Déformation d'un réseau avec (c) et sans (a) le couplage du modèle mécanique (b).

linéairement dépendantes des efforts extérieurs appliqués aux noeuds libres et les équations correspondantes peuvent être obtenues par combinaison des équations (3.7) et (3.8).

Enfin, ce modèle mécanique de réseau de barres est indépendant du type de modèle géométrique sous-jacent, sous réserve que l'on puisse le coupler à des éléments de dimension 0 (points) et 1 (segments). Il est utilisé pour la définition et la manipulation des Entités de Déformation de Formes Totalment Gauches [Pernot 2004, Pernot *et al.* 2005]. Plusieurs réseaux sont alors simultanément couplés aux polygones et réseaux caractéristiques de courbes et surfaces NURBS.

Formulation du problème d'optimisation

Pour permettre la déformation d'un modèle polyédrique, un réseau de barres est couplé au modèle complet. Les efforts extérieurs appliqués aux noeuds libres sont autant d'inconnues que l'utilisateur doit encore ajuster pour trouver la position appropriée des noeuds libres. En fait, ces manipulations doivent être transparentes pour un utilisateur néophyte certainement plus intéressé par la spécification de contraintes géométriques et de comportements de déformation. Pour cela, un problème d'optimisation est défini :

$$\begin{cases} \mathbf{G}(\mathbf{F}) = \mathbf{0} \\ \min \phi(\mathbf{F}) \end{cases} \quad (3.9)$$

où le vecteur inconnu \mathbf{F} contient les composantes de tous les efforts extérieurs appliqués aux noeuds libres.

Une telle formulation montre clairement le découplage qui existe entre :

- les **contraintes géométriques** qui peuvent être appliquées sur le modèle polyédrique (*e.g.* contraintes de position, de tangence ou de courbure) afin de spécifier sa forme (fig. 3.16.a). Ces contraintes produisent un ensemble d'équations potentiellement non linéaires qui lient entre elles les positions des noeuds libres. Le vecteur contrainte \mathbf{G} , contenant l'ensemble de ces équations, peut s'exprimer en fonction des efforts extérieurs à l'aide de l'équation (3.8).
- et la **fonction objectif à minimiser**. Il s'agit là d'un paramètre de plus haut niveau sémantique permettant la spécification de divers comportements de déformation par combinaison de plusieurs quantités géométriques et/ou mécaniques relatives au modèle de réseau de barres (fig. 3.16.b à fig. 3.16.e).

Les différentes fonctionnelles possibles sont décrites au chapitre 6.

Avantages de la méthode des densités de forces

La méthode de densité de force offre un bon compromis entre la préservation de la géométrie du modèle, la rapidité du processus de déformation et la facilité de sa mise en oeuvre par le biais de manipulations simples [Guillet 1999]. En particulier elle présente les avantages suivants [Pernot 2004] :

- Elle permet la manipulation directe des formes gauches à travers la manipulation de forces extérieures appliquées aux sommets d'un modèle mécanique de réseau de barres couplé aux sommets du modèle géométrique. Elle permet de s'affranchir du déplacement des sommets du modèle géométrique.
- Le réseau de barres étant formé de variétés de dimension zéro (noeuds) et un (barres), il peut être couplé à des modèles géométriques de diverses variétés. Ainsi, la déformation de tout type de modèle géométrique (polyligne, polyèdre, courbe et surface paramétriques) est rendue possible ainsi que celle des modèles hybrides.
- La minimisation de diverses fonctionnelles permet une plus grande variété de formes possibles et d'autoriser des comportements hétérogènes locaux (fig. 3.16).
- Enfin, le concept de lignes *objectifs* et de lignes de *limitation*⁴ [Pernot 2004] a permis de contrôler la déformation le long de courbes définies

4. *Target and limiting lines*

par l'utilisateur. En effet, les lignes *objectifs* expriment les directions globales de la déformation, tandis que les lignes de *limitations* spécifient les frontières de la zone du modèle à déformer.

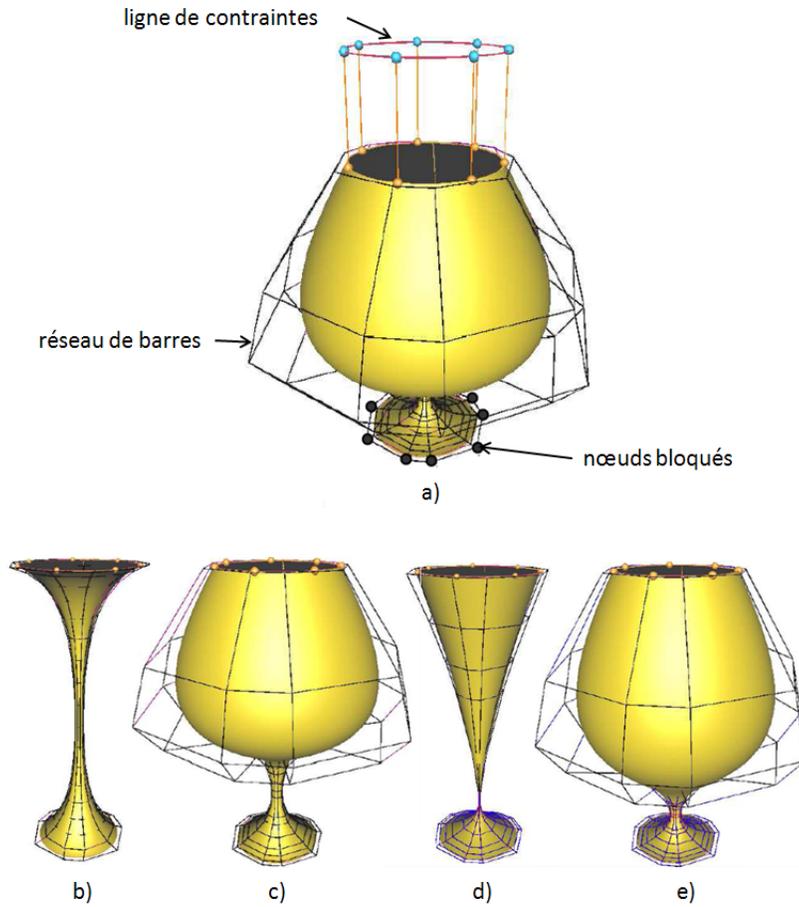


FIGURE 3.16 – Déformation d'un verre avec une ligne de contraintes (a) et avec la minimisation des forces extérieures (b,d) et la minimisation de la variation des forces extérieures (c,e) [Pernot 2004]

3.3.3 Synthèse des méthodes de déformation de modèles polyédriques

Les méthodes de déformation géométriques requièrent en général en entrée du processus un modèle polyédrique. Dans le cas de surfaces paramétriques, une étape de discrétisation et une étape de reconstruction seront donc nécessaires. Les principales limites des méthodes physiques concernent la manipulation de modèles géométriques lourds ainsi que la résolution itérative de systèmes d'équations différentielles qui ralentit considérablement les temps d'exécution. Les deux catégories de méthodes échouent le plus souvent à déformer des modèles composés de plusieurs carreaux devant rester connectés avec des conditions de continuité spécifiques. Enfin, les méthodes dites mixtes utilisent un modèle mécanique simplifié couplé aux sommets de contrôle afin d'améliorer la vitesse du processus de déformation.

3.4 Conclusion

Ce chapitre a présenté les principales méthodes de simplification et de déformation de modèles polyédriques existant dans la littérature. Cet état de l'art a permis de mettre en exergue deux techniques particulièrement intéressantes dans le cadre du couplage polyèdre/images.

En effet, la méthode de simplification de polyèdre de [Véron 1997] est basée sur un processus de suppression de sommet itératif mené principalement par le critère de sphères d'erreur. Ce critère impose que le modèle simplifié soit contenu dans l'enveloppe de toutes les sphères attachées aux noeuds du polyèdre et ce, tout au long du processus de simplification. Il est donc possible de piloter l'algorithme de simplification grâce à ce critère. Une méthode de simplification proposée est présentée au chapitre 4. Elle consiste à calculer la taille des sphères d'erreur à partir d'informations extraites d'images.

L'idée est tout d'abord de détecter les lignes de caractère dans les images, puis sur le modèle par le biais des étapes de calibration de caméras et de lancer de rayon (chapitre 2, sections 2.2.1, 2.2.3 et 2.3). Puis une carte hétérogène des tailles des sphères d'erreur est créée. Cette affectation automatique est calculée par un algorithme de filtrage d'images spécifiquement développé tel que le rayon des sphères d'erreur croisse avec la distance du noeud à la ligne de caractère identifiée sur le modèle polyédrique (chapitre 4, section 4.3).

La méthode de déformation initialement développée dans [Schek 1974] et optimisée dans [Pernot 2004] permet bien la déformation de modèles po-

lyédriques. Elle consiste à résoudre un problème d'optimisation numérique avec une fonctionnelle à minimiser et un jeu de contraintes à satisfaire. [Pernot *et al.* 2006] a montré que cette déformation pouvait s'appliquer au cas du remplissage de trous dans les modèles polyédriques où la fonctionnelle choisie permet d'obtenir des conditions de raccord en courbure entre le modèle initial et le maillage inséré pour remplir la zone manquante. Le jeu de contraintes additionnelles offre la possibilité de faire adopter des formes diverses au maillage inséré.

Deux méthodes de déformation de maillages appliquées au remplissage de trous sont présentées au chapitre 5. Dans les deux techniques, l'objectif est la génération de contraintes traduisant l'évolution de formes initialement présentes à la place du maillage inséré. Ces contraintes sont extraites des images. Dans la première méthode, les lignes de caractère sont détectées dans une paire d'images avec les méthodes de traitement d'images du chapitre 2, sections 2.2.1 et 2.2.3. Puis, les lignes de caractère 3D sont obtenues par triangulation stéréoscopiques de ces lignes de caractère 2D (chapitre 2, section 2.4.2). Le maillage inséré est contraint d'adopter la position de ces lignes de caractère 3D. Dans la deuxième méthode, les contraintes sont issues des équations du Shape From Shading (chapitre 2, section 2.4.3) et traduisent l'évolution de formes des surfaces représentées par l'image. Elles dépendent directement de l'intensité lumineuse des pixels et ne nécessitent pas de filtrage d'images pour leur extraction. De même, le maillage inséré est contraint à se déformer pour satisfaire ces contraintes.

Un couplage polyèdre/images est donc possible et est proposé sous la forme de ces trois nouvelles méthodes de manipulation de modèles polyédriques à l'aide de critères ou de contraintes issues d'images.

Utilisation conjointe d'images et de polyèdres

Une nouvelle approche de manipulations avancées de modèles polyédriques est ici proposée, sur la base de l'exploitation conjointe des modèles polyédriques et des images numériques.

L'utilisation d'images numériques comme enrichissement de données 3D a fait l'objet de plusieurs études. [Dias *et al.* 2003] utilisent des images pour détecter des zones d'intérêt par filtrage des contours pour aller ensuite enrichir par stéréoscopie un nuage de points. [Baltzakis *et al.* 2002] proposent d'exploiter des images numériques afin d'estimer la qualité d'un modèle 3D reconstruit par triangulation d'un nuage de points acquis par balayage laser.

Les images sont parfois utilisées pour identifier les zones où la reconstruction 3D est inconsistante et pour insérer des nouveaux points par stéréoscopie. [Torres *et al.* 2004] évaluent les coordonnées de points 3D appartenant à des zones incomplètes du modèle 3D en utilisant un processus stochastique (chaîne de Markov) à partir de la connaissance de coordonnées de points 3D et de la valeur de l'intensité des pixels 2D correspondants. [Deveau 2006] propose un graphe automatique de description de la scène 3D, qui réunit des informations 2D (image numérique, image d'intensité laser, image de distance, etc.) et des informations 3D (nuage de points, normales, modèles géométriques, etc.) afin de visualiser différents niveaux de description et de sélectionner le meilleur pour chaque objet.

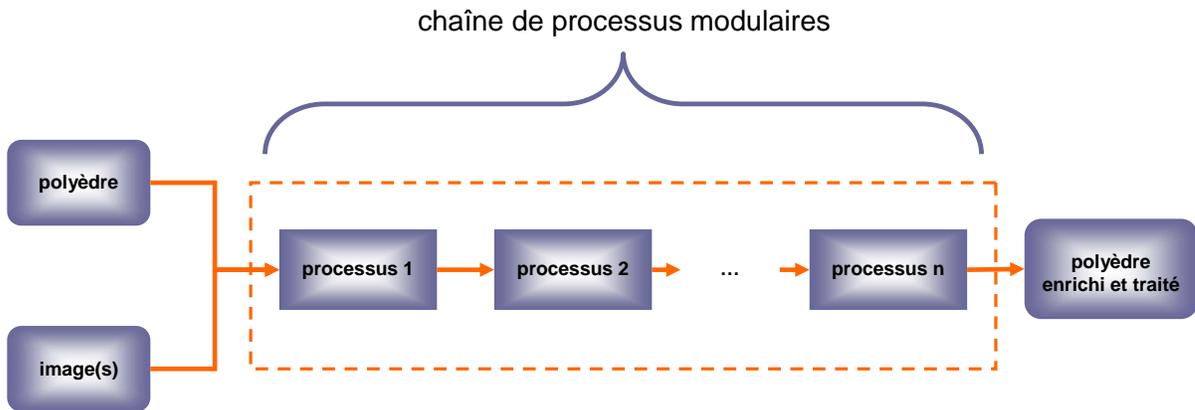


FIGURE 3.17 – Architecture de l'approche proposée de manipulation de modèles polyédriques à l'aide d'informations extraites d'images numériques.

Partant donc du constat que les images numériques peuvent contenir des données pertinentes pour l'analyse et la manipulation modèles géométriques, une nouvelle approche de couplage de ces deux représentations est proposée au sein d'une architecture de processus modulaire (figure 3.17). Dans cette approche, les informations extraites des images ne sont plus uniquement des sources complémentaires d'informations servant à enrichir les modèles 3D. Ici, les informations contenues dans les images sont utilisées pour définir des critères et des contraintes permettant de piloter directement des méthodes de traitement de polyèdres, telles la simplification et la déformation.

Ainsi, la stratégie adoptée consiste à utiliser des images pour piloter les deux méthodes de manipulation de modèles polyédriques présentées au chapitre 3 : la simplification et le remplissage de trous. Une chaîne de modules indépendants est proposée avec en données d'entrée une ou plusieurs image(s) et un polyèdre. En sortie, on obtient le polyèdre simplifié ou complété (trous remplis). Les images utilisées sont soit des images réelles ou des captures d'écran (images de synthèse), soit des images de contours (obtenues après filtrage de photographies, comme décrit en section 2.2). Les polyèdres sont obtenus soit par triangulation d'un nuage de points acquis par balayage laser, soit par tessellation face par face de modèles CAO.

- Partie II -

Nouvelle approche de traitement de modèles polyédriques basée sur l'exploitation d'images numériques

La deuxième partie du présent mémoire présente la nouvelle approche d'exploitation d'images numériques pour le traitement avancé de modèles polyédriques.

Le quatrième chapitre est consacré à la méthode de simplification de polyèdres contrôlée par un critère basé sur l'exploitation d'informations extraites d'images numériques et lié à la notion de *lignes de caractère*.

Le cinquième chapitre présente les deux méthodes de déformation de modèles polyédriques. L'une utilise des contraintes obtenues par triangulation stéréoscopique, et l'autre des contraintes issues des techniques de Shape From Shading.

Enfin, le présent mémoire s'achève sur le bilan de l'apport de ces nouvelles méthodes et les perspectives de recherche des travaux présentés.

Chapitre 4

Simplification de modèles polyédriques pilotée à partir d'informations extraites d'images numériques

Les algorithmes de simplification de modèles polyédriques visent à réduire le nombre de faces d'un modèle en satisfaisant divers critères, notamment en matière de préservation de formes. Dans ce chapitre, une méthode de simplification pilotée à partir d'informations extraites d'images numériques est ainsi présentée. Elle repose sur l'extraction des lignes de caractère du produit pour identifier sur le modèle 3D, les zones qui doivent être plus ou moins simplifiées. Cette méthode s'affranchit du calcul de courbures discrètes ou autres quantités issues de la courbure. Elle utilise des techniques de filtrage permettant l'extraction de contours dans les images. Les images 2D utilisées sont soit des photographies de l'objet réel, soit des captures d'écran du modèle géométrique reconstruit. L'algorithme de simplification est basé sur le processus de suppression de sommets présenté à la section 3.2.2. Il utilise une carte de tolérances pour contraindre la façon dont les noeuds du modèle peuvent être supprimés.

4.1 Introduction

Les besoins en terme de simplification de polyèdres ont été présentés à la section 3.2. En pratique, simplifier un modèle géométrique consiste à réduire le nombre de faces qu'il comporte en respectant un ensemble de critères. Par exemple, il peut être intéressant d'utiliser un critère relatif à la préservation de la forme globale de l'objet, ou un critère visant à minimiser sa variation de volume, à produire des faces triangulaires les plus équilatérales possibles, ou à créer une enveloppe englobante du modèle initial, ou toute autre combinaison de ces différents critères.

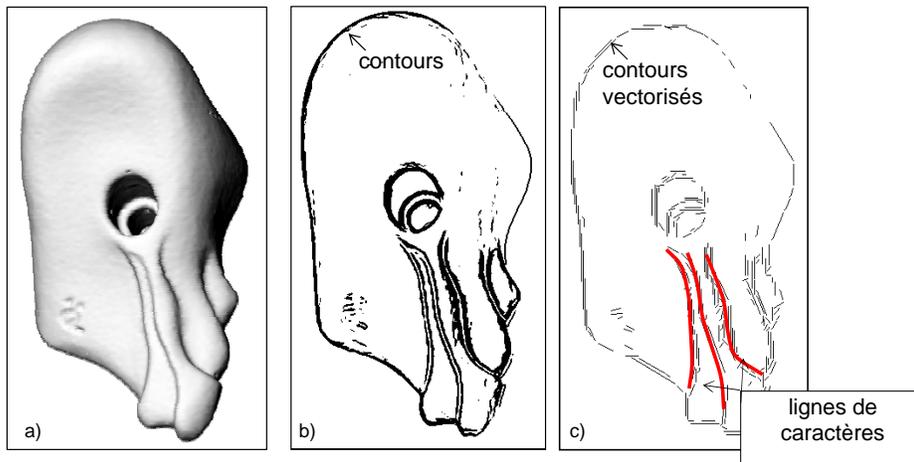


FIGURE 4.1 – Les lignes de caractère identifiées dans une image (a) après filtrage (b) et vectorisation (c) des contours (modèle de la prise d’escalade *courtoisie de Tomoadour*).

Lorsque l’on s’intéresse aux algorithmes de simplification préservant les formes de l’objet, on s’aperçoit que les *lignes de caractère* jouent un rôle primordial dans l’apparence visuelle qu’elles dégagent [Fontana *et al.* 2000, FIORES II]. Elles représentent une sorte de signature visuelle permettant de distinguer et d’identifier les formes d’un objet et peuvent être identifiées sur un modèle (fig. 4.1.a), sur une image de contours (fig. 4.1.b) ou une image de contours vectorisés (fig. 4.1.c). Ainsi, une méthode de simplification capable de préserver ces éléments clés aiderait à la définition et à la maintenance de la maquette numérique structurée autour de la notion de *features*. Ici les *features* correspondent à des entités de haut niveau reposant sur les lignes de caractère et donnant une signification à un ensemble de faces [Pernot 2004].

4.2 Schéma de traitement adapté à la simplification

L’idée sous jacente de la méthode proposée concerne la détection des lignes de caractère à partir d’images. En effet, les images numériques des objets sont de taille en mémoire inférieure de l’ordre de quelques Mo à celles des modèles géométriques correspondants (nécessitant plusieurs Go). Permettre l’identification des lignes de caractère à partir d’images et non plus à partir de la géométrie du modèle est donc pertinent.

Les principales étapes de la méthode de simplification de polyèdres à l'aide d'images sont illustrées sur la figure 4.2. Ici, la méthode a été testée sur un modèle polyédrique issu de la tessellation face par face d'un modèle CAO de piston de compresseur. En entrée de la chaîne de processus, on trouve le modèle polyédrique ainsi qu'une ou plusieurs images le représentant. Les images peuvent être de simples captures d'écran (exemple de la figure 4.2.a) comme des photographies réelles de l'objet physique 4.2.b).

La première étape est relative à l'extraction et à la vectorisation des contours des images (figure 4.2). Cette étape a pour but de révéler sur les images 2D les lignes de caractère de l'objet.

Afin de permettre une correspondance entre les lignes de caractère identifiées et le modèle 3D, une étape de calibration de caméra est ensuite nécessaire. Une ou plusieurs images du polyèdre sont requises. Une fois l'image et le polyèdre calés dans le même repère de référence, les contours sont projetés sur le polyèdre.

Une carte de tolérances (voir section 3.2.2 pour un rappel de l'algorithme de simplification adopté) est générée pour piloter de manière hétérogène l'algorithme de simplification. Dans ce chapitre seules les nouvelles contributions sont présentées. La chaîne de processus étant complètement modulaire, chaque module peut être optimisé indépendamment voir même être remplacé par une autre technique plus performante.

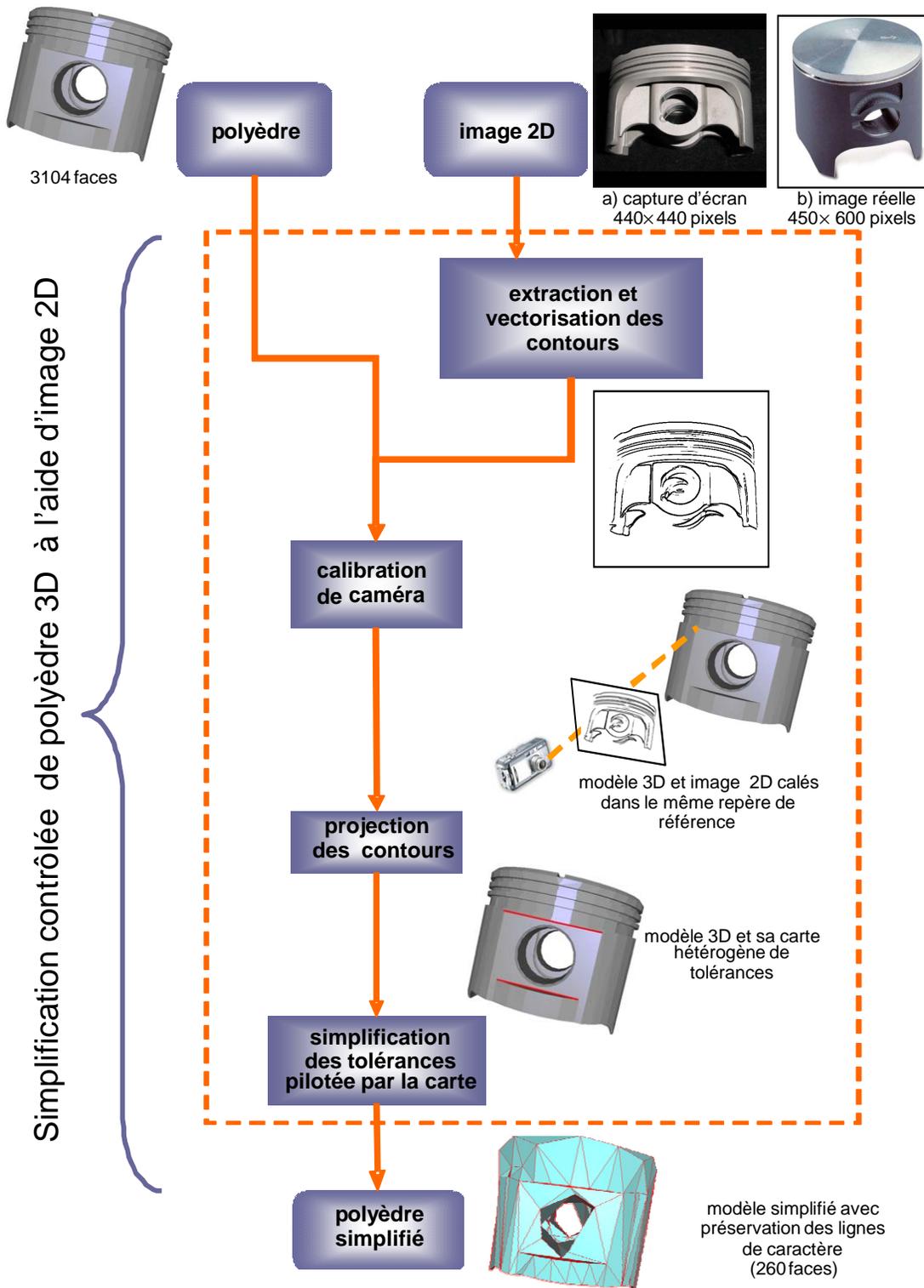


FIGURE 4.2 – Principe de l'algorithme de simplification de polyèdres à l'aide d'images numériques. Il est à noter sur cet exemple illustratif que les lignes de caractère spécifiées sont bien préservées. La forme du contour trou se trouve quant à elle dégradée car il n'a pas été détecté comme une ligne de caractère dans cet exemple.

4.3 Création de la carte de tolérances

4.3.1 Extraction et vectorisation des lignes de caractère

Afin d'identifier les lignes de caractère d'un polyèdre en se servant d'une image représentant le même objet, une méthode de détection de contours, le filtre de Sobel [Sobel 1990], a été choisie dans la littérature (voir section 2.2.1).

Un prétraitement visant à réduire le bruit dans l'image est tout d'abord réalisé : il s'agit d'un filtre moyen défini par une matrice M de dimension 3×3 avec ses neuf coefficients égaux à $\frac{1}{9}$. Ce filtre a pour effet d'assigner à chaque pixel la valeur moyenne des intensités lumineuses des neuf pixels voisins. La fonction intensité lissée devient :

$$I_S(x, y) = conv(I, M)(x, y) \quad (4.1)$$

où la fonction *conv* représente la convolution de la fonction intensité définie pour chaque pixel (x, y) de l'image avec le filtre défini par la matrice M (section 2.2.1).

Afin de finaliser le filtrage et d'obtenir une image binaire en noir et blanc, un seuillage simple est appliqué à l'image des contours selon l'algorithme suivant :

Algorithme 1 Binarisation d'une image I de dimension $n \times m$ pixels

```

pour  $x = 1$  to  $n$  faire
  pour  $y = 1$  to  $m$  faire
    si  $I(x, y) \geq seuil$  alors
       $I(x, y) \leftarrow 0$ ;
    sinon
       $I(x, y) \leftarrow 255$ ;
    finsi
  fin pour
fin pour

```

Afin d'éviter les effets d'*aliasing* présentés à la section 2.2.3, une étape de vectorisation est nécessaire. La méthode de [Chai *et al.* 1992] qui propose l'algorithme *OZZ* a été choisie car elle permet d'obtenir un ensemble de vecteurs ayant tous un début et une fin (voir section 2.2.3).

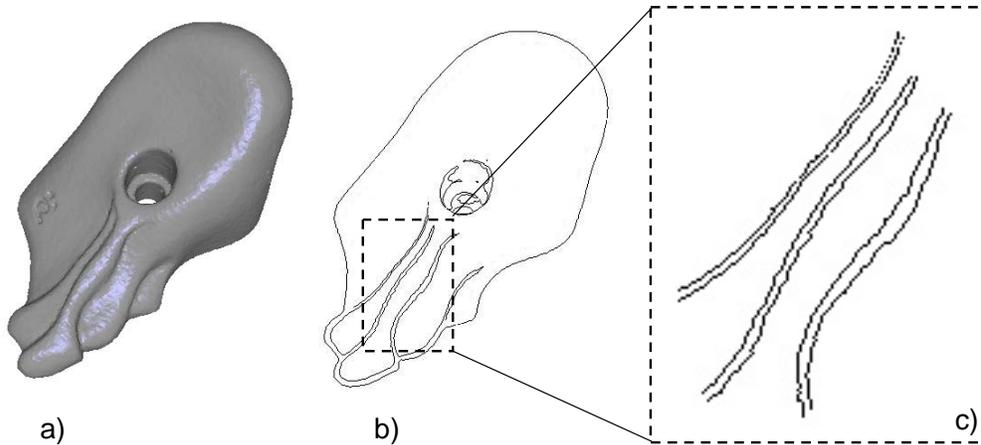


FIGURE 4.3 – Extraction des lignes de caractère 2D (c) à partir de l’image des contours (b) obtenue par filtrage d’une image (a) *courtoisie de Tomoadour*).

4.3.2 Projection des contours et calcul de la position des sphères d’erreur

Une fois les contours identifiés dans l’image 2D, une étape de calibration de caméra est réalisée afin de caler le polyèdre 3D et l’image 2D dans le même repère de référence. On utilise la méthode de [Faugeras *et al.* 1987] décrite en section 2.3. Les contours peuvent ensuite être projetés sur le polyèdre.

La projection d’un contour proprement dite est réalisée en deux étapes :

Première étape

Les points extrémités d’un contour (fig.4.4) p_1 et p_2 sont projetés en deux points P_1 et P_2 sur le polyèdre. Ceci est réalisé grâce à un lancer de rayon en calculant les intersections des rayons (Cp_1) et (Cp_2) , passant par le centre optique C et les points extrémités du contour, avec les faces du polyèdre [O’Rourke 1998].

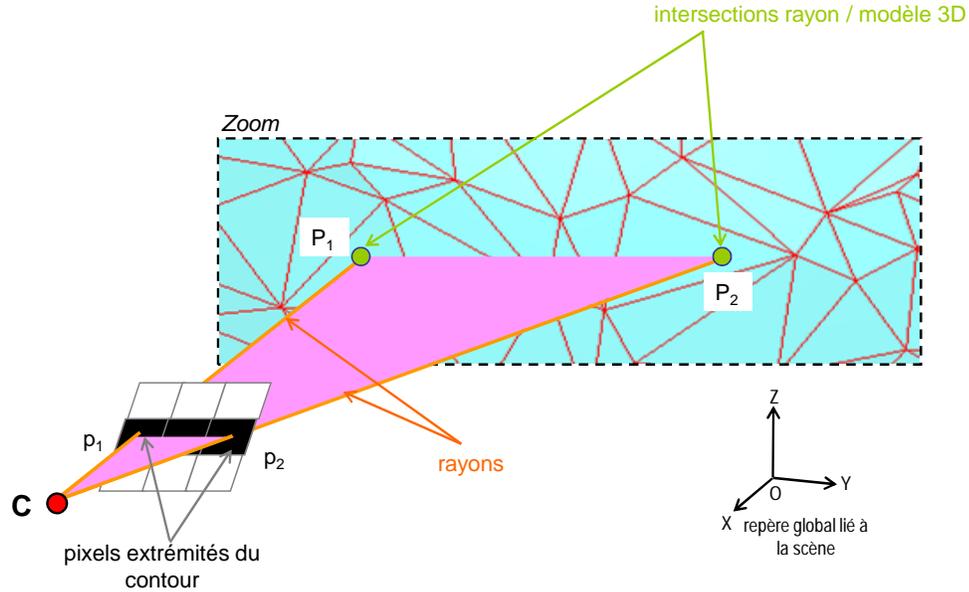


FIGURE 4.4 – Projection des pixels des extrémités d'un contour.

Deuxième étape

Le calcul de l'intersection du plan (Cp_1p_2) avec le polyèdre [O'Rourke 1998] est réalisé. Si nécessaire, l'insertion de nouvelles arêtes est réalisée en produisant une triangulation locale de type Delaunay [Barry 1991].

Certains contours ne sont toutefois pas projetés. C'est le cas des contours identifiés appartenant aux frontières de l'objet. Une autre image prise d'un point de vue plus adapté sera alors nécessaire.

Enfin, il faut signaler que l'étape d'insertion est optionnelle. Dans la plupart des cas, une sphère d'erreur est simplement créée aux coordonnées 3D de l'intersection de la projection de la ligne de caractère sur le polyèdre. Ainsi, le modèle simplifié étant contenu dans l'enveloppe des sphères d'erreur, les lignes de caractère seront préservées.

Selon l'origine du modèle polyédrique utilisé, deux types de projections de contours sont envisageables.

Dans le cas d'un polyèdre issu de la tessellation face par face d'un modèle CAO, les arêtes vives existent déjà sur le modèle et il s'agit d'identifier celles qui correspondent aux contours des images (fig. 4.5) sans insérer de nouveaux sommets. Pour éviter des effets de crénelage, du à la finesse de la tessellation, certaines méthodes de lissage pourraient être mises en oeuvre [Zhao *et al.* 2006].

Le critère actuellement utilisé se base sur un calcul de distance euclidienne aux sommets voisins des noeuds projetés (alg. 2). On pourrait également imaginer d'utiliser la courbure discrète dans un voisinage proche de la projection, le noeud identifié étant le noeud de plus forte courbure discrète calculée dans la face. Ces idées n'ont pas encore été implémentées.

Algorithme 2 Identification des noeuds du maillage appartenant à une polyligne

```

n ← le nombre de projections 3D de la polyligne
pour  $i = 1$  to  $n$  faire
   $P_i \cap$  la face  $F_k$ 
   $N_k \leftarrow (N_{1k}, N_{2k}, N_{3k})$  l'ensemble des trois noeuds de  $F_k$ 
   $d_{min} \leftarrow \min_{N_m \in N_k} d(N_m, P_i)$ 
  si  $d(N_1, P_i) \leq d_{min}$  alors
     $N_1 \in$  la polyligne ;
  sinon
    si  $d(N_2, P_i) \leq d_{min}$  alors
       $N_2 \in$  la polyligne ;
    sinon
       $N_3 \in$  la polyligne ;
    finsi
  finsi
finsi
fin pour

```

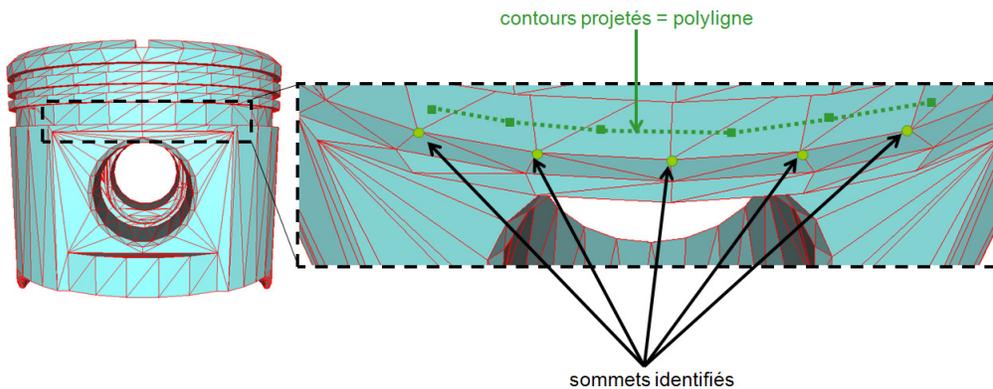


FIGURE 4.5 – Projection des contours et identification des sommets pour un modèle de piston de compresseur tessellé.

Dans le cas d'un modèle scanné, les arêtes du modèle reconstruit peuvent également ne pas correspondre exactement aux contours projetés (fig.4.6). Ainsi, l'insertion de nouveaux sommets (fig. 4.6.c) est requise afin d'éviter des effets de crénelage (effet d'*aliasing*, fig. 4.6.b.) (alg. 3) :

Algorithme 3 Localisation de la projection P_i dans la face F_k

```

 $\varepsilon_d \leftarrow$  rayon de la sphère d'un sommet ou d'une zone autour d'une arête
 $d(P_i, V_k) \leftarrow$  distance d'un sommet  $P_i$  au sommet  $V_k$ 
 $d(P_i, E_k) \leftarrow$  distance d'un sommet  $P_i$  à l'arête  $E_k$ 
si  $\exists k \setminus d(P_i, V_k) \leq \varepsilon_d$  alors
     $P_i \leftarrow V_k$ 
sinon
    si  $\exists k \setminus d(P_i, E_k) \leq \varepsilon_d$  alors
         $E_k$  est subdivisée en 2 en  $P_i$ 
    sinon
         $F_k$  est subdivisée en 3 en  $P_i$ 
    finsi
finsi

```

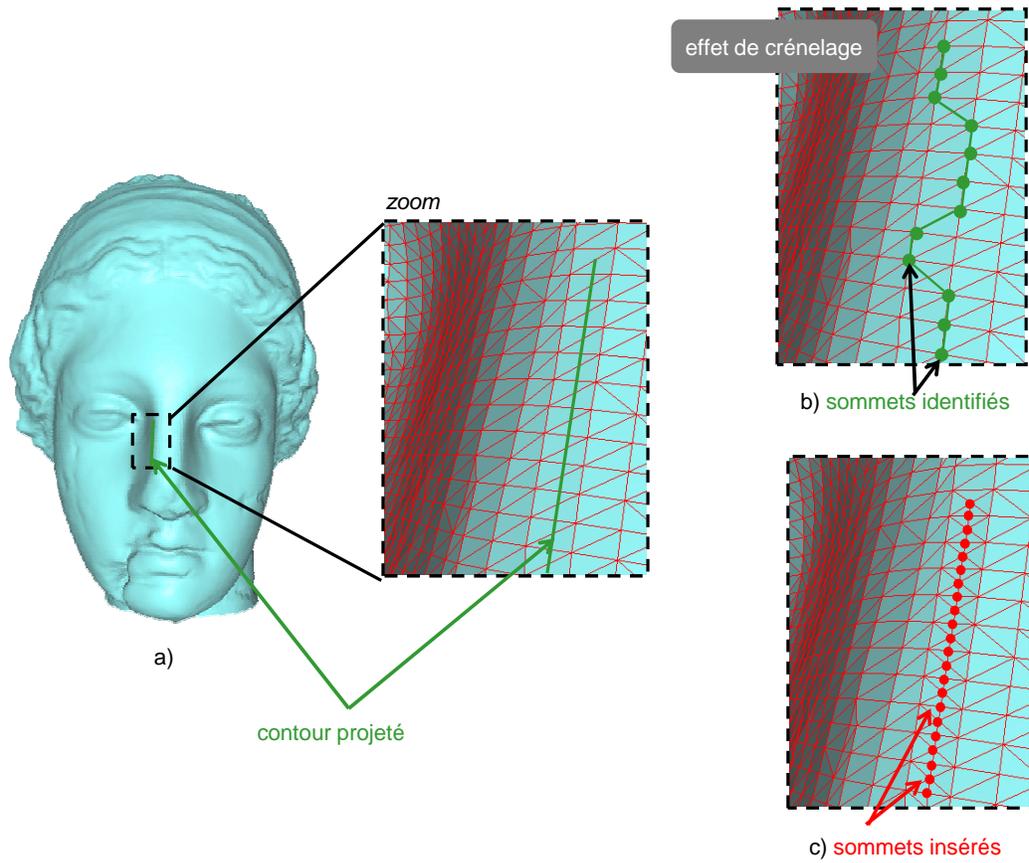


FIGURE 4.6 – Projection des contours sur un modèle scanné (a) (*courtoisie de l'INRIA*) : les sommets du modèle les plus proches du contour projeté forme une polygone crénelée (b) ; de nouveaux sommets sont donc insérés au modèle pour éviter cet effet de crénelage (c).

4.3.3 Calcul des rayons des sphères d'erreur

Avant de réaliser l'étape de simplification à proprement parler, la carte de sphères d'erreur hétérogène doit être construite de telle manière que les sommets les plus éloignés des lignes de caractère soient les plus décimés. L'idée est de réaliser une distribution des sphères d'erreur telle que les diamètres décroissent avec la distance géodésique entre le sommet et la ligne de caractère (fig.4.7.g.).

Afin de simuler un tel comportement et d'ainsi éviter le calcul coûteux des distances géodésiques, un algorithme de filtrage spécifique a été élaboré. L'idée est de commencer par filtrer n fois l'image 2D pour obtenir n images des contours $I_1, I_3, I_5, \dots, I_{2(n-1)-1}, I_{2(n-1)+1}$ ayant des épaisseurs de contour augmentant respectivement de "1" à " $2(n-1)+1$ " pixels (fig.4.7.b à 4.7.d). Chaque image filtrée I_k est ensuite soustraite à l'image filtrée I_{k-2} produisant ainsi une nouvelle image de contours $I'_{(k-1)/2}$ pour $k \in [3, \dots, 2(n-1)+1]$ et $k' \in [1, \dots, (n-1)]$ (figures 4.7.e et 4.7.f).

Puis l'étape de projection est réalisée pour chaque image $I'_{k'}$ (fig. 4.7.b, fig. 4.7.e et fig. 4.7.f). Pour chaque point obtenu par projection d'un pixel noir contenu dans l'image $I'_{k'}$, une sphère d'erreur de rayon $r(k')$ est créée (alg. 4). L'évolution de $r(k')$ suit l'évolution de la fonction sigmoïde définie par quatre paramètres de contrôle : le rayon de la plus petite sphère r_{min} , le nombre de filtrages n qui correspond à la largeur de la sigmoïde, la différence entre le rayon de la plus grande sphère r_{max} et r_{min} qui représente la hauteur de la sigmoïde, s la valeur de l'extrema de la dérivée $r'(k')$ qui représente la rapidité d'évolution entre la plus petite et la plus grande sphères (fig. 4.7.h) :

Algorithme 4 Algorithme de calcul de la taille des sphères d'erreur

```

 $r_{min} \leftarrow$  rayon de la plus petite sphère
 $r_{max} \leftarrow$  rayon de la plus grande sphère
 $n \leftarrow$  nombre de filtrages réalisés à partir d'une même image
 $s \leftarrow$  rapidité d'évolution des rayons entre  $r_{min}$  et  $r_{max}$ 
pour  $k' = 1$  to  $n - 1$  faire
     $r(k') = r_{min} + \frac{r_{max} - r_{min}}{1 + s \cdot e^{\frac{n}{2} - k' - 1}}$ 
fin pour

```

Dans l'implémentation actuelle, ces quatre paramètres sont définis par l'utilisateur. Une fois la carte de tolérances calculée, l'algorithme de simplification du logiciel Simpoly© (section 3.2.2) est lancé. La figure 4.8 montre le résultat de la simplification du modèle d'un cube avec une carte hétérogène de sphère d'erreur. La carte utilisée a été calculée suivant l'algorithme 4 à l'aide de quatre filtrages de l'image de la figure 4.7.a₁.

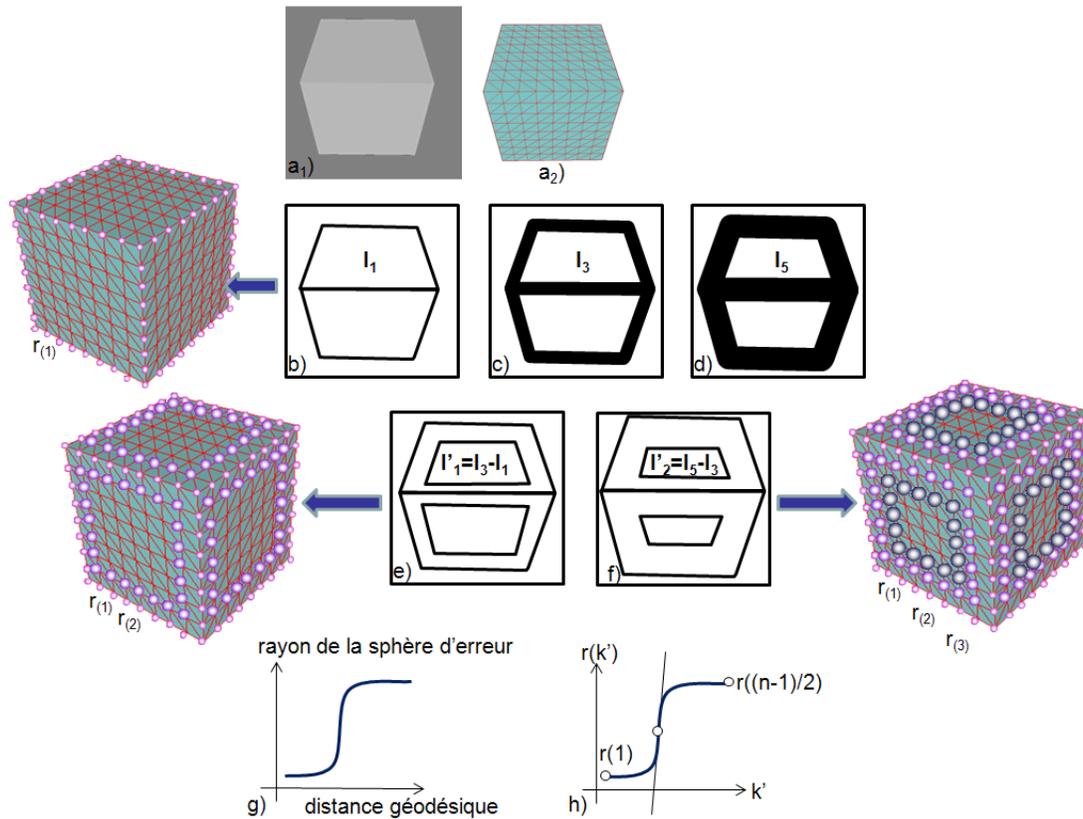


FIGURE 4.7 – Algorithme de filtrage de l'image initiale (a_1) spécifique conçu pour calculer la taille des sphères d'erreur.

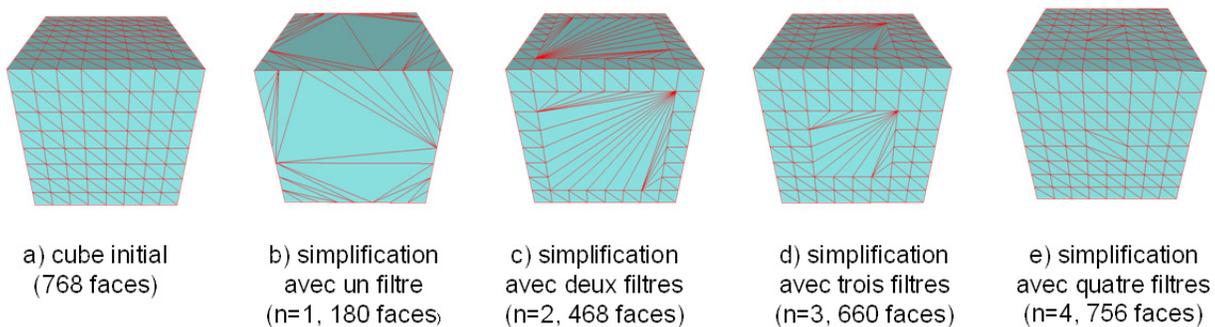


FIGURE 4.8 – Exemple d'application de l'algorithme de filtrage approximant les distances géodésiques : quatre filtrages ont été testés sur une capture d'écran du modèle d'un cube de dimension $1 \times 1 \times 1$, avec $r_{min} = 0.000001$, $r_{max} = 0.001$, $s = 0.00036$.

Conclusion

L'algorithme de filtrage développé (alg. 4) permet d'approcher le calcul de distances géodésiques des noeuds du polyèdre aux lignes de caractère détectées à l'aide du filtrage des contours [Sobel 1990]. En effet, l'obtention des images des contours et leur soustraction est facile à mettre en oeuvre. En revanche, le calcul des distances géodésiques sur un polyèdre, c'est-à-dire une surface plane par morceau, n'est pas aussi aisé. Il aurait fallu mettre en oeuvre des techniques basées sur la théorie des graphes, ne s'appliquant pas toujours au cas de polyèdres de topologie quelconque, ayant des temps d'exécution et une complexité importants [Bommes *et al.* 2007].

4.4 Résultats et analyses

4.4.1 Critères d'analyse et métriques de comparaison des résultats

Afin de pouvoir comparer les résultats obtenus par les différents algorithmes de simplification testés, deux métriques ont été adoptées :

- la première métrique est un facteur d'aspect ratio permettant d'évaluer la qualité des maillages Éléments Finis. Il permet de distinguer les faces "dégénérées", i.e. les faces ayant un ou deux angles très faibles, des faces plus équilatérales. Selon [Ciarlet 1978], le facteur Q est défini tel que :

$$Q = \frac{\alpha S}{hp} \quad (4.2)$$

où α est égal à $2\sqrt{3}$, h est la longueur de sa plus longue arête, S son aire et p son demi périmètre. Cet indicateur vaut 1 pour un triangle équilatéral. Il est compris dans l'intervalle [0,1] avec la limite en zéro correspondant aux triangles "plats". En analyse Elements Finis, il est communément admis qu'une triangulation est acceptable si le facteur Q de son triangle le plus allongé est supérieur à 0.5.

- la seconde métrique est la distance géométrique (la distance de Hausdorff) entre le modèle initial et le modèle simplifié. Nous avons utilisé le logiciel libre METRO afin de calculer cette distance [Cignoni *et al.* 1998]. Soit $d(\mathbf{P}, M')$ la distance entre un point $\mathbf{P} \in$ au maillage M et le maillage M' telle que :

$$d(\mathbf{P}, M') = \min_{\mathbf{P}' \in M'} d(\mathbf{P}, \mathbf{P}') \quad (4.3)$$

avec $d(\mathbf{P}, \mathbf{P}')$ la distance euclidienne entre \mathbf{P} et \mathbf{P}' .

La distance de Hausdorff entre le maillage M et le maillage M' est alors définie comme :

$$d(M, M') = \max_{P \in M} d(P, M') \quad (4.4)$$

4.4.2 Résultats

Le processus global de simplification présenté à la figure 4.2 a été testé sur le modèle polyédrique d'une prise d'escalade (*courtoisie de Tomoadour*). Les données d'entrée de la chaîne de processus sont un modèle numérisé de 89 038 faces (fig. 4.9.a) et deux images en niveaux de gris ayant une résolution de 1060×723 pixels (fig.4.9.b et fig.4.9.e). Ces deux captures d'écran ont été obtenues avec un modèle de rendu de Gouraud [Gouraud 1971]. En effet, cette méthode permet d'obtenir le rendu d'une face d'un polyèdre sans brusque variation de luminosité entre deux faces adjacentes.

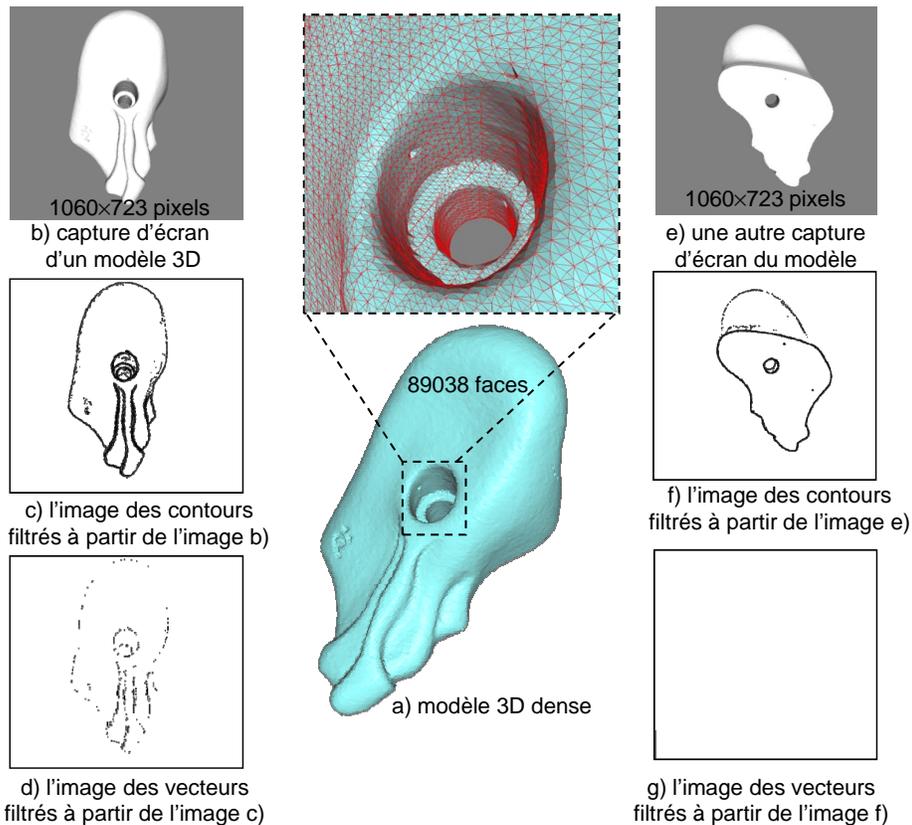


FIGURE 4.9 – Captures d'écran (b,e) du modèle polyédrique de la prise d'escalade (a), les images des contours (c,f), les images vectorisées (d,g) identifiant les lignes de caractères (*courtoisie de Tomoadour*).

Afin d'éviter le calcul de courbures discrètes et de permettre la simplification à l'aide de lignes de caractères extraites d'images, trois contributions majeures ont été proposées :

1. La taille des sphères d'erreur n'est plus spécifiée de manière homogène par l'utilisateur mais de manière hétérogène à partir d'une loi utilisant la position des sommets par rapport aux lignes de caractère (contribution 1). Les calculs de distances géodésiques sont approximés en utilisant une technique de filtrage spécifiquement développée.
2. L'ordonnement des sommets L_v candidats à la suppression n'est plus obtenu à partir des courbures discrètes absolues mais en fonction de la taille des sphères d'erreur. Ainsi, les sommets les plus éloignés des lignes de caractère seront supprimés en premier (contribution 2).
3. L'algorithme de simplification n'entraîne plus le calcul de la courbure discrète absolue durant l'étape de remaillage local puisque le critère retenu n'est plus le critère de préservation de la variation de courbure mais celui de l'équilatéralité des faces produites (contribution 3).

Comparaison 1 : méthode de simplification de Simpoly© à l'aide d'images vs. méthode de simplification classique de Simpoly©

La méthode de simplification à l'aide de lignes de caractère extraites d'images a été dans un premier temps comparée à la méthode de simplification classique de [Véron 1997]. L'exécution commence par le filtrage des images en niveaux de gris (fig. 4.9.b et 4.9.e) afin d'en extraire les contours (fig. 4.9.c et 4.9.f) avec un filtre de Sobel. Les images résultantes sont ensuite vectorisées en utilisant l'algorithme OZZ (fig. 4.9.d et 4.9.g). Une fois les lignes de caractère identifiées, ces dernières sont projetées sur le modèle 3D. Ici l'étape de calibration des images n'est pas nécessaire puisqu'il s'agit de captures d'écran déjà calibrées. Chaque image produit une carte de tolérances qui recouvre localement le modèle. L'union de ces cartes forme une carte hétérogène de sphères d'erreur qui est utilisée pour piloter l'algorithme de simplification.

La figure 4.10.b montre le résultat d'une simplification avec un ordonnancement des sommets candidats à la suppression selon la taille de leur sphère d'erreur et avec un remaillage local qui utilise le critère d'équilatéralité. Le résultat est un polyèdre simplifié composé de 2628 faces. On peut remarquer que les sommets préservés sont localisés autour des lignes de caractère et que l'aspect des faces est très acceptable au regard du facteur de qualité $Q_{moyen} = 0.58$.

La génération automatique de la carte des tolérances est beaucoup plus rapide que si elle avait été construite manuellement et les temps de calcul coûteux des courbures discrètes ont été évités. Néanmoins, elle pourrait l'être davantage avec l'utilisation de volumes englobants (*bounding boxes*) pour réaliser la projection des contours.

En comparaison, la simplification qui utilise un ordonnancement basé sur le critère de courbure [Fine 2001] et un remaillage local préservant la variation de courbure [Véron 1997] produit un polyèdre dont les 2628 faces sont réparties de façon plus homogène (fig. 4.10.c). Tous les rayons des sphères d'erreur ont été initialisés avec une unique valeur telle que le polyèdre résultant possède précisément 2628 faces. Les triangles produits sont davantage dégénérés dans le sens où ils sont plus fins et allongés ce qui est pénalisant vis-à-vis du critère d'aspect ratio Q .

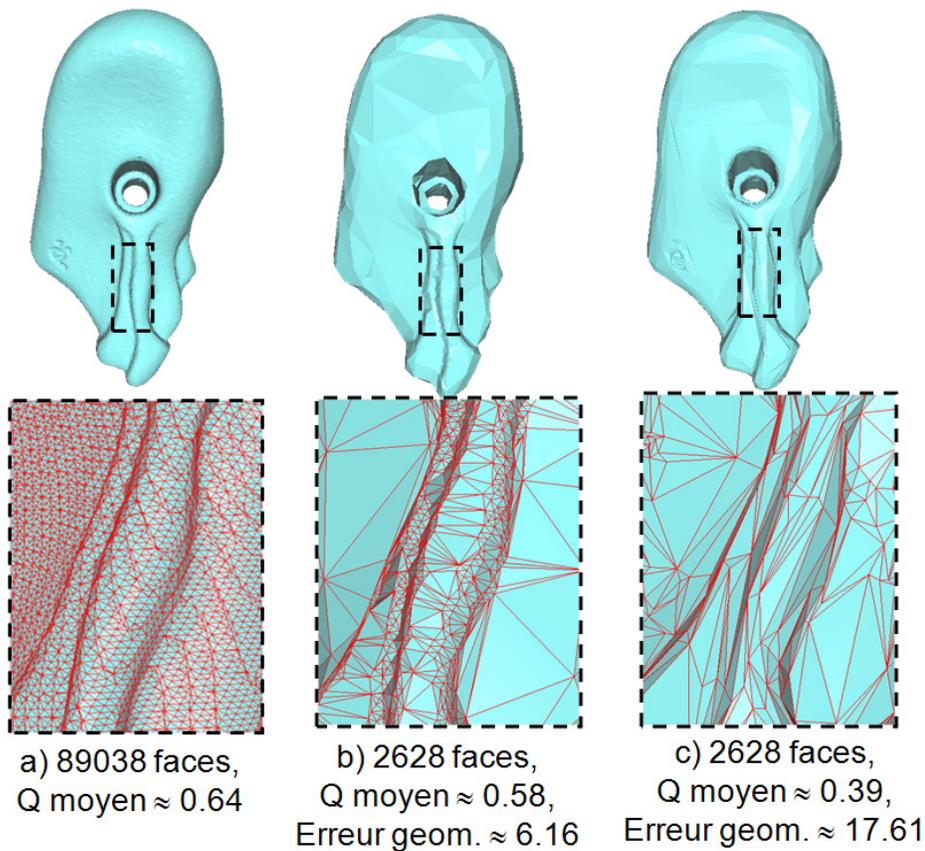


FIGURE 4.10 – Modèle initial (a) (*courtoisie de Tomoadour*) et les résultats de simplifications utilisant un remaillage local selon le critère d'équilatéralité (b) ou le critère de préservation de la variation de courbure (c) .

Sur cet exemple, on peut remarquer que pour un même nombre de faces restantes, la qualité de la préservation des lignes de caractère est meilleure en utilisant le processus de simplification à l'aide d'images. Le facteur Q_{moyen} de la zone entourant les lignes de caractère est supérieur à 0.5 et l'erreur géométrique est inférieure à celle du modèle obtenu avec le remaillage local préservant la variation de courbure.

D'autres tests ont été réalisés afin de comparer la nouvelle méthode à base d'images avec la méthode classique (figures 4.11).

Pour un nombre donné de faces restantes, et quelque soit le critère de remaillage local utilisé, les lignes de caractère sont beaucoup mieux préservées avec la nouvelle méthode. En effet, dans le cas du remaillage préservant la variation de courbure, le facteur Q_{moyen} et l'erreur géométrique sont meilleurs lors de l'utilisation d'une carte de tolérances hétérogène (fig. 4.11.a et fig. 4.11.b).

Dans le cas du remaillage avec le critère d'équilatéralité des faces, la différence au niveau du facteur Q_{moyen} n'est pas aussi significative. Par contre, l'erreur géométrique est beaucoup plus élevée avec l'utilisation d'une carte de tailles homogène (fig. 4.11.d) qu'avec une carte hétérogène (fig. 4.11.c).

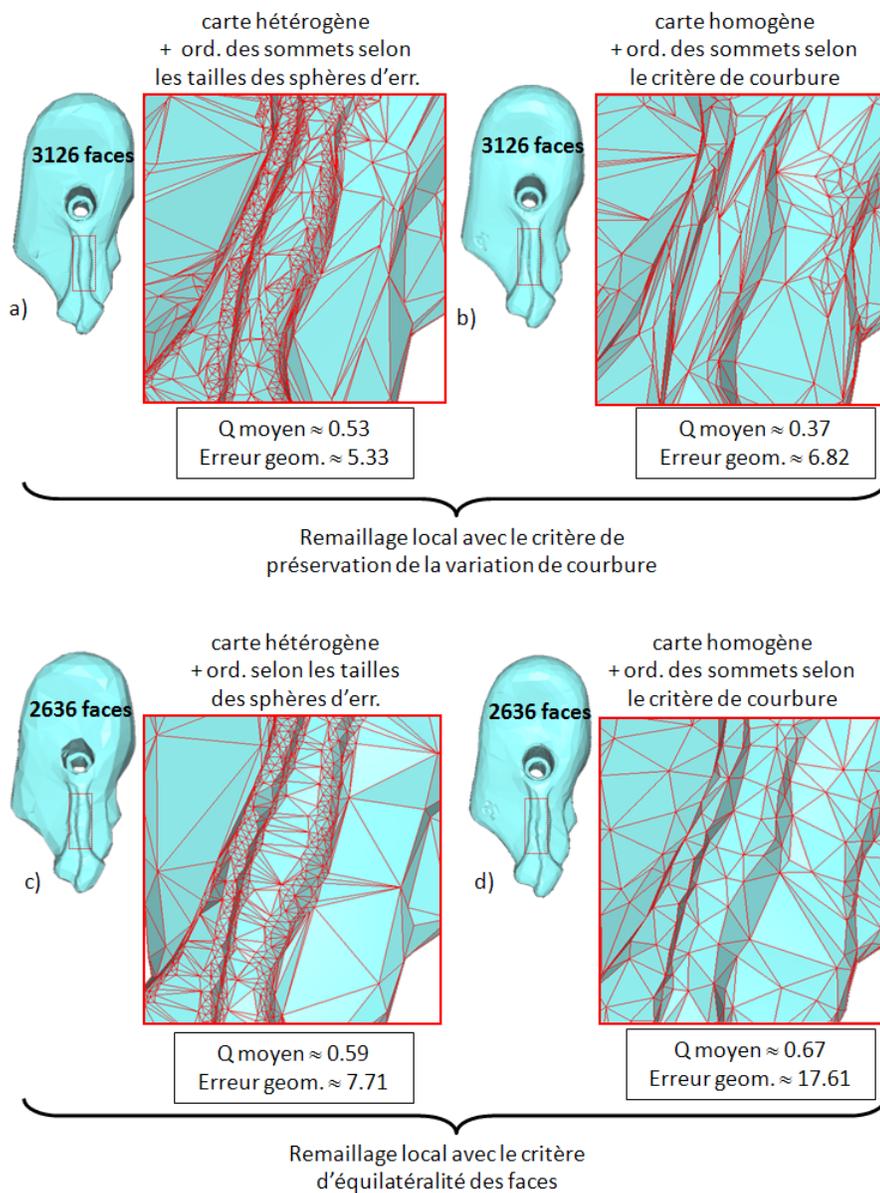


FIGURE 4.11 – Comparaisons des résultats des méthodes de simplification sans ou avec notre approche sur le modèle de la prise d'escalade artificielle (*courtoisie de Tomoadour*).

Comparaison 2 : méthode de simplification de Simpoly© à l'aide d'images vs. méthode de simplification du logiciel QSlim

Nous avons comparé ensuite, la méthode proposée avec celle de [Garland *et al.* 1992] implémentée dans le logiciel QSlim [Garland 1999]. Pour quatre taux de simplification différents, nous avons testé les deux méthodes. La figure 4.12 montre les modèles obtenus avec une simplification drastique (90% des faces ont été supprimées).

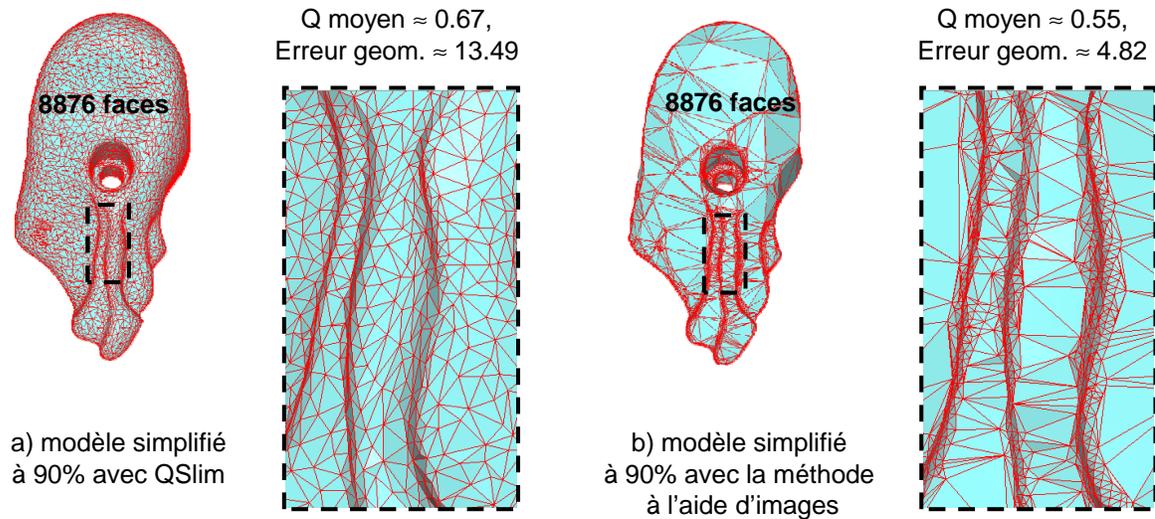


FIGURE 4.12 – Modèles simplifiés obtenus avec la méthode basée sur les quadriques (a) et avec notre méthode qui utilise des images (b).

Quelque soit le taux de simplification, la distance de Hausdorff des modèles obtenus avec la méthode de simplification à l'aide d'images est inférieure à celle des modèles obtenus avec QSlim (fig.4.13).

Le facteur de qualité de maille Q_{moyen} est quant à lui sensiblement le même (fig.4.14). Ceci est dû au fait que QSlim simplifie uniformément toutes les faces du modèle initial. Par conséquent, il simplifie de la même manière les faces visibles de l'extérieur du modèle et celles qui ne le sont pas (faces appartenant à des parties et cavités internes de l'objet, par exemple sur la paroi interne d'un trou débouchant). Au contraire, la méthode que nous proposons simplifie davantage les faces invisibles depuis l'extérieur de l'objet puisqu'elles n'apparaissent pas sur les images. Les lignes de caractère sont également mieux préservées.

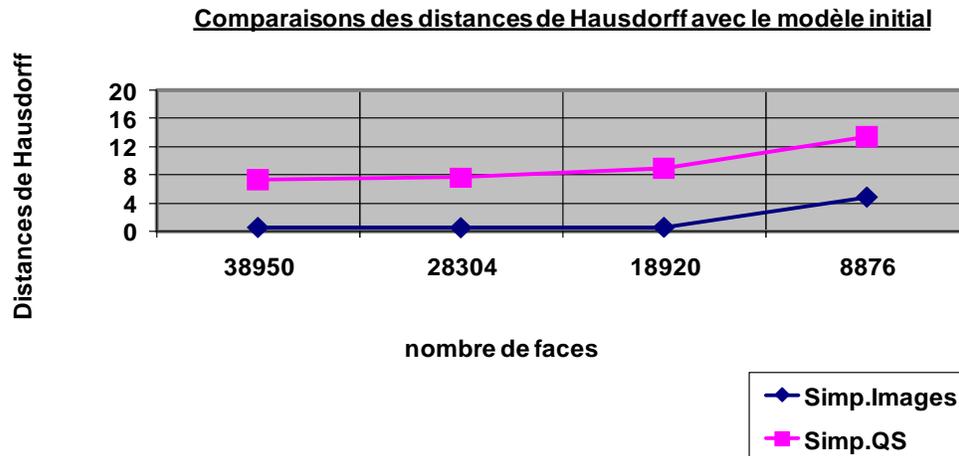


FIGURE 4.13 – Comparaison en terme de *distances de Hausdorff* entre le modèle initial et le modèle simplifié avec QSlim (Simp.QS.) et avec notre méthode à base d’images (Simp.Images) sur le modèle de la prise d’escalade artificielle (*courtoisie de Tomoadour*).

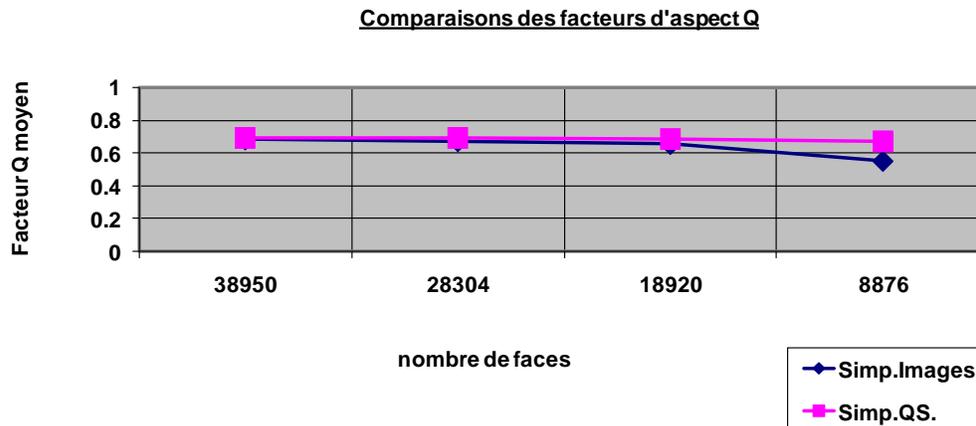


FIGURE 4.14 – Comparaison en terme de *facteur de qualité des faces Q_{moyen}* entre le modèle initial et le modèle simplifié avec QSlim (Simp.QS.) et avec notre méthode à base d’images (Simp.Images) sur le modèle de la prise d’escalade artificielle (*courtoisie de Tomoadour*).

En terme de temps d’exécution, QSlim demeure plus rapide que notre méthode (environ 10 fois plus rapide en moyenne pour le modèle de la prise d’escalade composé de 89038 faces).

Les temps de calcul de la méthode à base d’images pourraient être diminués en utilisant une méthode de lancer de rayon optimisée. L’utilisation

de la méthode des volumes englobants permettrait par exemple d'éviter les calculs d'intersection du rayon issu de la caméra et passant par un pixel à projeter, avec toutes les faces du modèle. Une décomposition de la scène 3D avec une représentation octree permettrait d'englober le modèle dans des volumes hiérarchisés, chaque volume possédant la liste des faces du modèle polyédrique qu'il contient (fig.4.15).

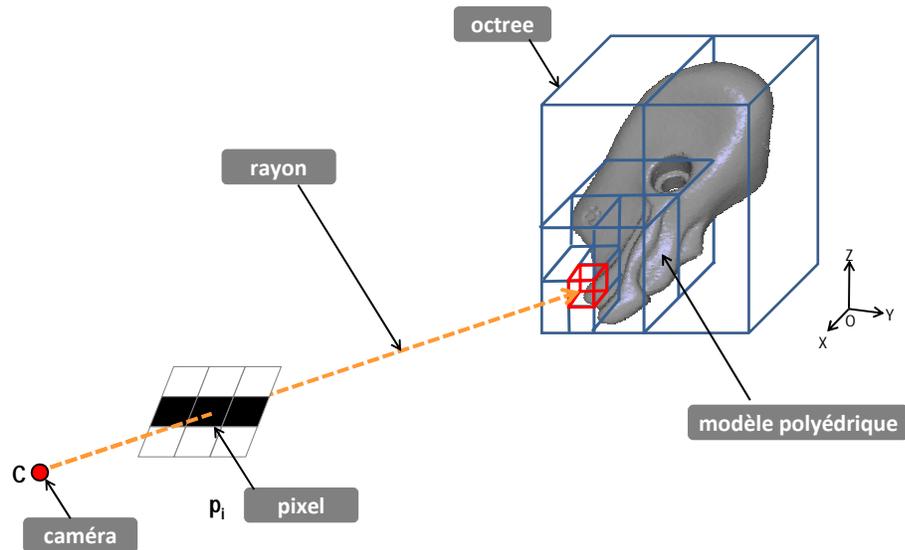


FIGURE 4.15 – Le principe du lancer de rayon avec l'optimisation des volumes englobants.

Finalement, la méthode proposée a été également comparée à celle de QSlim sur le modèle très dense de la *pale de turbine* (courtoisie du *Georgia Institute of Technology*) composé de 1 765 388 faces (fig. 4.16.a). Il a été simplifié à l'aide de 8 images de contours avec un total de 41 442 pixels projetés. Pour un nombre similaire de faces résultantes, alors que le facteur de qualité des faces Q_{moyen} est identique, le modèle simplifié avec la méthode de simplification à l'aide d'images (fig.4.16.c) engendre une erreur géométrique inférieure (13.90 contre 18.62 pour le modèle simplifié avec QSlim fig.4.16.b).

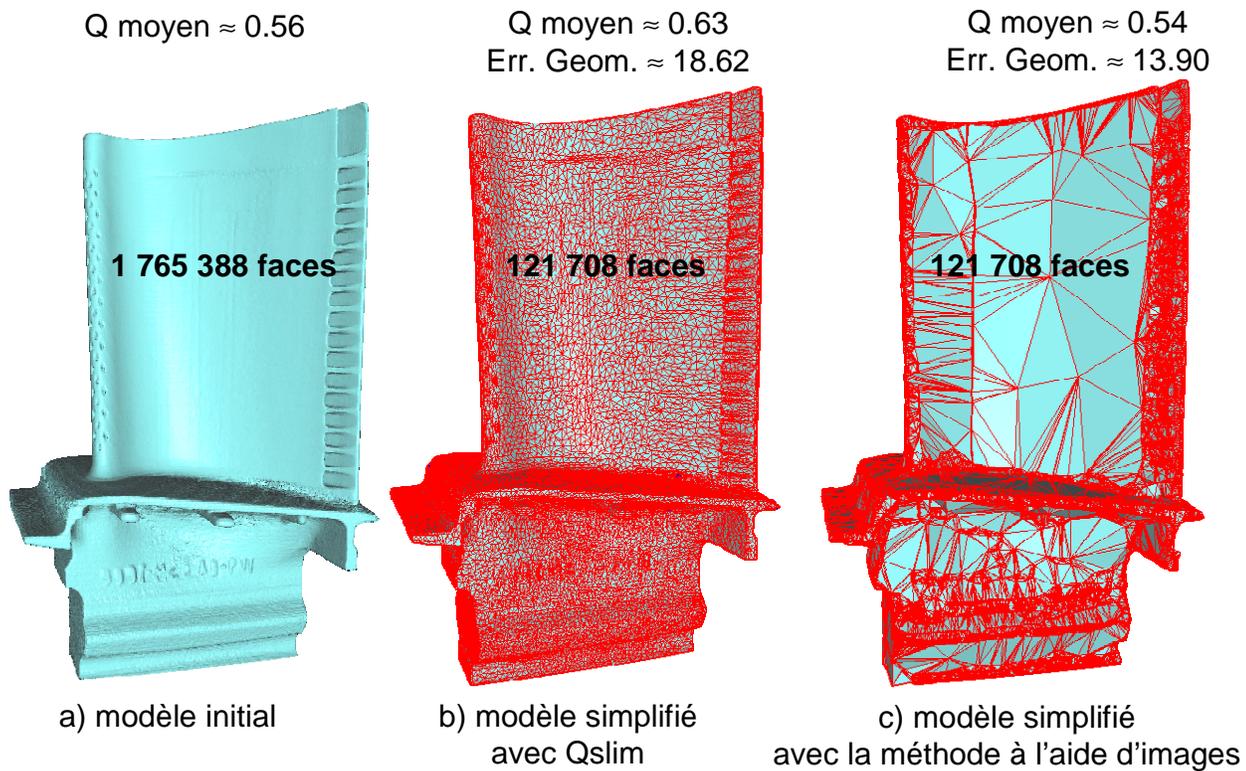


FIGURE 4.16 – Modèle initial de la pale de turbine (1 765 388 faces, *courtoisie de Georgia Institute of Technology*) (a) et les modèles simplifiés avec Qslim (b) et avec la méthode à l'aide d'images proposée (c).

4.5 Conclusion

Dans ce chapitre, une méthode de simplification de modèles polyédriques à l'aide d'informations extraites d'images est présentée. Elle est constituée d'une chaîne globale de processus modulaires qui permet de préserver les lignes de caractère d'un modèle durant le processus de simplification.

Ce dernier est basé sur un algorithme itératif de suppression de sommets piloté par une carte hétérogène de sphères d'erreur qui définissent l'enveloppe avec laquelle le modèle simplifié doit être en intersection. La taille des sphères est calculée automatiquement en utilisant une méthode de filtrage d'images qui approxime la distance géodésique entre les sommets et les lignes de caractère.

Ces dernières sont extraites par filtrage des contours des images, puis par projection de ces contours sur le modèle polyédrique. Plus les sommets sont

éloignés des lignes de caractère identifiées, plus les sphères qui leur sont affectées sont grandes, et plus la simplification est importante. On s'affranchit ainsi des calculs de courbures discrètes. Dans le cas de l'utilisation de plusieurs images, chaque image produit une carte de tailles de sphères d'erreurs qui est ensuite fusionnée avec les autres.

La méthode de simplification à l'aide d'images proposée répond également aux besoins de gestion des niveaux de détails (LOD¹) des maquettes numériques volumineuses. En effet dans ce contexte, l'enjeu est de pouvoir trouver un équilibre entre un modèle qui soit assez précis et complet pour permettre des simulations numériques fiables, et un modèle peu volumineux pour répondre aux besoins de visualisation rapide et interactive [Souffez *et al.* 2006]. Les modèles simplifiés avec notre méthode satisfont à ces deux critères. L'identification et la préservation des lignes de caractère ont pour conséquence une meilleure préservation de l'apparence du modèle original. Au niveau des lignes de caractère, un bon compromis est proposé puisque le nombre de faces est suffisamment raffiné et de bonne qualité (au regard du facteur Q par exemple) pour permettre une analyse locale et le nombre total de faces du modèle est réduit afin d'en améliorer les temps de visualisation.

Si les résultats obtenus valident la chaîne de processus proposée, plusieurs optimisations sont envisagées. Concernant le module de calibration de caméra, les imprécisions dues aux correspondances des points/pixels réalisées par l'utilisateur pourraient être corrigées en utilisant une méthode de calibration automatique [Faugeras *et al.* 1986, Faugeras *et al.* 1992].

Le module de projection des contours peut également être amélioré pour diminuer les temps d'exécution en utilisant une méthode de boîtes englobantes [Revelles *et al.* 2000]. La technique de filtrage développée afin de calculer les diamètres des sphères pourrait être remplacée par une méthode exploitant les intensités lumineuses des pixels de l'image utilisée. Ainsi, en s'appuyant sur des techniques de Shape From Shading (voir section 2.4.3), on pourrait piloter plus finement la simplification entre les lignes de caractère. Ici, l'image ne serait plus filtrée mais analysée à partir d'une équation aux dérivées partielles traduisant la façon dont la lumière se réfléchit sur la surface de l'objet. Une telle approche pourrait même aller remplacer/simplifier une partie cylindrique (identifiée dans les images) par un nombre restreint de triangles bien positionnés.

Jusqu'à récemment, en ingénierie mécanique, les images n'étaient exploitées qu'en vue de compléter la description des objets ou d'enrichir la maquette numérique mais rarement comme une source d'information pouvant piloter un processus de manipulation de modèles géométriques. En ce sens,

1. Levels Of Details

la méthode de simplification à l'aide d'images ouvre la voie à de nouvelles applications.

De tels résultats et perspectives concordent avec les objectifs du réseau d'excellence européen Aim@Shape [[Aim@Shape 2004-08](#)]. Dans ce contexte, la méthode proposée peut contribuer à la définition d'un modèle multi-représentations composé d'un polyèdre et d'images ainsi qu'à l'enrichissement du modèle géométrique avec des informations sémantiques extraites des images.

Chapitre 5

Déformation de modèles polyédriques à l'aide d'images numériques : application au remplissage de trous en Conception Inverse

Le processus de conception inverse¹ consiste à créer une représentation numérique d'un objet physique. En fonction du processus d'acquisition adopté et de la complexité de l'objet à digitaliser, certaines zones peuvent ne jamais être atteintes. Il en résulte des déficiences dans le nuage de points et un ensemble de trous dans le modèle polyédrique reconstruit. Dans ce chapitre, deux processus complets de remplissage de trous dans les modèles polyédriques sont proposés. Ils utilisent des informations extraites d'images 2D de l'objet physique. Dans les deux cas, le trou est tout d'abord identifié, nettoyé et rempli grossièrement. La première méthode déforme le maillage inséré pour qu'il respecte des lignes de caractère 3D reconstruites par triangulation stéréoscopique à l'aide de deux images. Dans la seconde méthode, le maillage est déformé pour qu'il satisfasse à des contraintes calculées à partir de l'intensité lumineuse de pixels. Ces contraintes décrivent la manière dont la lumière est réfléchiée par la surface de l'objet dont on cherche à reconstruire la géométrie.

5.1 Introduction

La conception inverse est une technique utilisée pour créer une représentation numérique d'un objet physique. Le but final est généralement de permettre l'analyse et/ou l'amélioration de produits dont on ne possède

1. ou rétro-conception

pas les modèles géométriques. Le processus de reconstruction commence par l'acquisition d'un nuage de points représentatif de la peau de l'objet physique. Un polyèdre est ensuite créé à partir de cet ensemble non structuré de points. Des surfaces B-Spline/NURBS peuvent également être créées, soit directement à partir du nuage de points, soit à partir de la triangulation. Plus récemment, l'utilisation des surfaces de subdivision a également été explorée. Là encore, la plupart des approches fonctionnent sur une triangulation du nuage de points. Par conséquent, il est crucial de se concentrer sur la recherche d'un modèle polyédrique qui représente au mieux la surface extérieure de l'objet.

En fonction de la complexité de l'objet à digitaliser et du type de technologie du système d'acquisition adopté (*e.g.* machine à mesurer tridimensionnelle ou scanner laser), certaines zones de la surface extérieure peuvent ne jamais être accessibles. La figure 5.1 montre le résultat de l'acquisition laser d'un haut de colonne depuis un seul point de vue (vue 1).

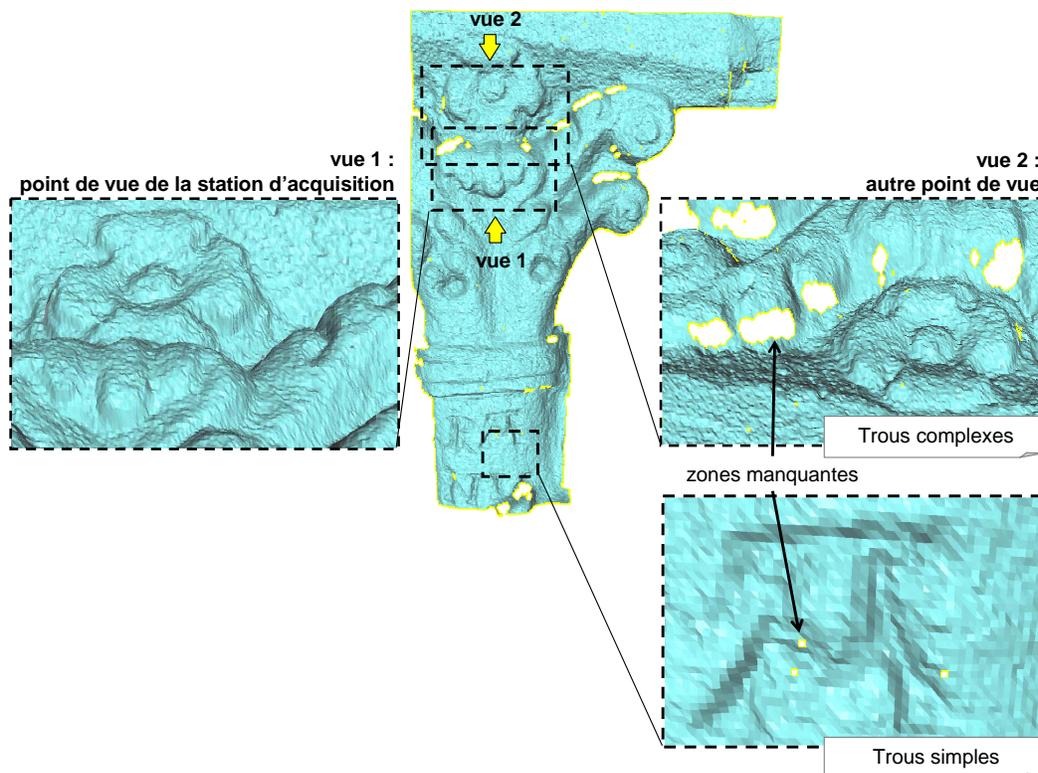


FIGURE 5.1 – A partir du point de vue 1, le laser est incapable d'atteindre les portions de l'objet visibles d'un autre point de vue 2.

Ces limites se traduisent par des déficiences dans le nuage de points et par un ensemble de trous dans le modèle polyédrique reconstruit. La présence de trous indésirables et inacceptables peut induire des résultats inattendus si le maillage triangulaire sert de support pour le prototypage rapide ou pour une analyse Éléments Finis.

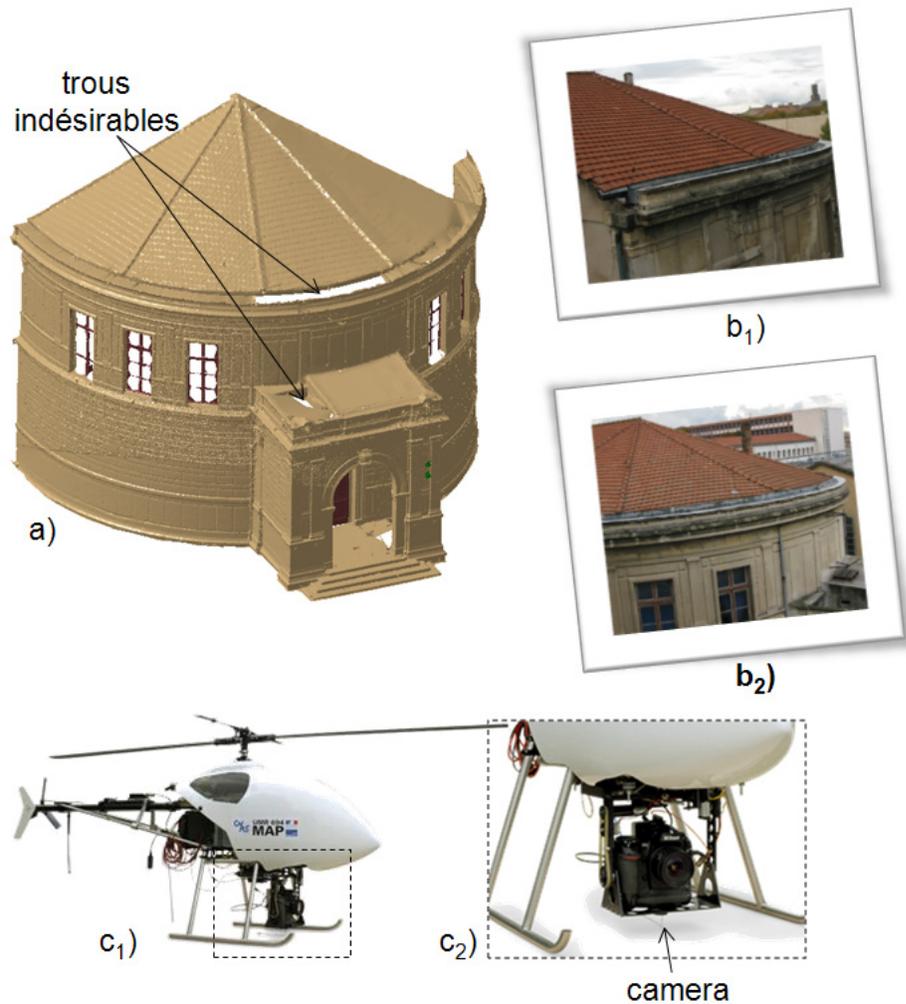


FIGURE 5.2 – L’Amphithéâtre du CER Arts et Métiers ParisTech d’Aix-en-Provence avec des trous (a) qui pourraient être remplis en utilisant des images réelles (b₁ et b₂) prises à l’aide d’un hélicoptère radio-commandé (c₁) (courtoisie of Gamsau) embarquant un appareil photographique (c₂).

Comme montré sur la figure 5.2, des zones de l’objet à retro concevoir (l’amphithéâtre des Arts et Métiers ParisTech à Aix-en-Provence ici) ne peuvent pas être atteintes par un scanner laser. Sur cet exemple, le modèle

digitalisé de l'amphithéâtre a été obtenu par fusion et triangulation de trois nuages de points acquis avec un scanner laser localisé en trois différentes positions en face de l'édifice. Le modèle compte plus d'un million de triangles. Ainsi, certaines zones du monument invisibles depuis le sol génèrent des trous indésirables (fig. 5.2.a) qui pourraient être reconstruites à l'aide d'images prises sur le dessus (fig. 5.2.b₁ et 5.2.b₂).

Dans l'exemple de la numérisation du Château Comtal de la ville de Carcassonne, 45 millions de points 3D et 540 photographies ont été générées produisant ainsi un total de 60 giga octets de données (*courtesy MAP GAM-SAU laboratory [De Luca 2006]*). Ici, les photographies sont déjà calibrées ce qui signifie que la position et l'orientation de la caméra qui les a prises sont connues. Finalement, il est clair que l'exploitation d'images n'est pas nécessaire pour remplir des trous dont la forme manquante n'est pas complexe (fig. 5.1.b), c'est-à-dire des trous dont le contour comporte peu (une dizaine) d'arêtes.

5.2 Etat de l'art des méthodes de remplissage de trous

Diverses techniques ont été proposées pour remplir les trous indésirables présents dans les polyèdres. Parmi elles, on peut distinguer les approches dites **géométriques**, qui remplissent les zones manquantes en interpolant la géométrie adjacente, des autres approches dites **non géométriques** principalement basées sur une représentation implicite de la surface du modèle.

Parmi les **approches non géométriques**, [Curless *et al.* 1996] utilisent une représentation volumétrique pour détecter les zones qui doivent être remplies. [Davis *et al.* 2002] appliquent un processus de diffusion permettant d'étendre une fonction de distance signée au travers de leur représentation volumétrique et ceci jusqu'à ce que les zéros de la fonction créent des ponts au travers des trous (fig. 5.3). Cette approche itérative est particulièrement bien adaptée aux bouchages de trous de géométrie et/ou de topologie complexe.

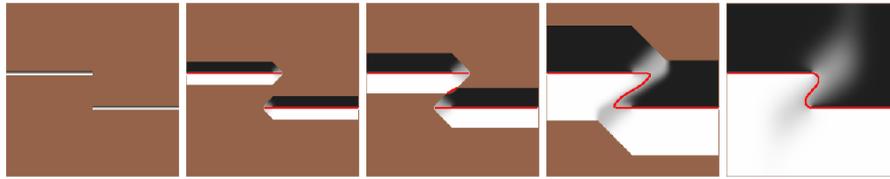


FIGURE 5.3 – *Le processus de diffusion volumétrique de [Davis et al. 2002] : une fonction de distance signée est étendue par convolution jusqu'au remplissage du trou ; la surface est ensuite obtenue par l'algorithme de marching cubes [Lorensen et al. 1987].*

Une approche similaire a été développée par [Nooruddin et al. 2003] pour la simplification et la réparation de modèles polyédriques. [Verdera et al. 2003] représentent également la surface de façon implicite et bouchent les trous à l'aide d'un système d'équations différentielles issues d'algorithmes de réparation d'images. Ces équations utilisent les caractéristiques géométriques des parties connues (*e.g.* la courbure) et sont appliquées uniquement sur le trou ainsi que son voisinage proche. Ces équations étant non-linéaires, elles conduisent à un algorithme plus lent que celui de [Davis et al. 2002].

[Clarenz et al. 2004] utilisent quant à eux une méthode éléments finis pour minimiser l'intégrale du carré de la courbure moyenne (énergie de Willmore) du trou bouché. Leur processus est itératif et permet seulement d'assurer une continuité en tangence avec le maillage adjacent.

Concernant les **approches géométriques**, la méthode de [Liepa 2003] commence par détecter les trous (fig. 5.4.a) et les remplit à l'aide d'une triangulation d'aire minimale (fig. 5.4.b). La triangulation est raffinée de telle sorte que la densité de triangles s'accorde avec celle du maillage adjacent et lissée (fig. 5.4.c) à l'aide d'un opérateur "parapluie" [Kobbelt et al. 1998] basé sur la résolution d'un système d'équations linéaires. Il satisfait des conditions de raccord en tangence avec le modèle initial.

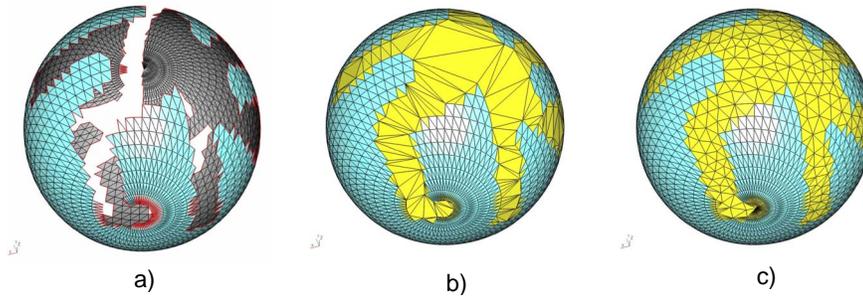


FIGURE 5.4 – Etapes du remplissage de trou de [Liepa 2003] : identification et nettoyage du contour du trou (a), remplissage grossier minimisant l'aire de la triangulation (b), lissage et raffinement du maillage inséré (c).

[Schneider *et al.* 2001] proposent une technique de lissage basée sur la résolution d'une équation différentielle du quatrième ordre non linéaire. Une continuité G^1 est atteinte mais la résolution est itérative et plus longue. Les travaux de [Desbrun *et al.* 1999] nécessitent également un processus de résolution itératif. D'autres approches sont basées sur la projection des moindres carrés induisant un processus de minimisation non linéaire [Wang *et al.* 2003, Tekumalla *et al.* 2004].

Outre les comparaisons en terme de complexité algorithmique, toutes ces méthodes ne se valent pas quant à la qualité de la forme reconstruite et notamment en ce qui concerne les critères suivants :

- la **forme des triangles** du maillage inséré. En effet, la plupart des logiciels de CAO ne permettent qu'un remplissage bord à bord, le maillage inséré étant très facilement distingué du reste du polyèdre, soit parce que ses arêtes ne sont pas de la même taille (fig. 5.5.a₁) soit parce qu'il résulte d'un remplissage itératif paramétrique privilégiant une direction d'alignement des arêtes (fig. 5.5.a₂).
- la **satisfaction ou non des conditions au raccord en courbure**. Le plus souvent, le raccord entre le maillage inséré et le reste du modèle n'est pas un raccord en courbure (fig. 5.5.b₂.) mais en position ou tangence (fig. 5.5.b₁).
- la **reconstruction de formes complexes**. La plupart des méthodes n'offrent pas la possibilité de prescrire des contraintes supplémentaires pour sculpter la forme de la partie manquante, ce qui est fortement pénalisant dans le cas où la forme présente originellement dans le trou est complexe (fig. 5.5.c).

Ce sont ces limites que nous tentons de surmonter. Dans ce chapitre, nous supposerons que les polyèdres traités sont triangulaires, orientés, manifolds

et connectés. En particulier, deux trous séparés n'auront aucun sommet en commun et un trou n'aura pas d'îlots (sinon le modèle polyédrique ne serait pas connecté).

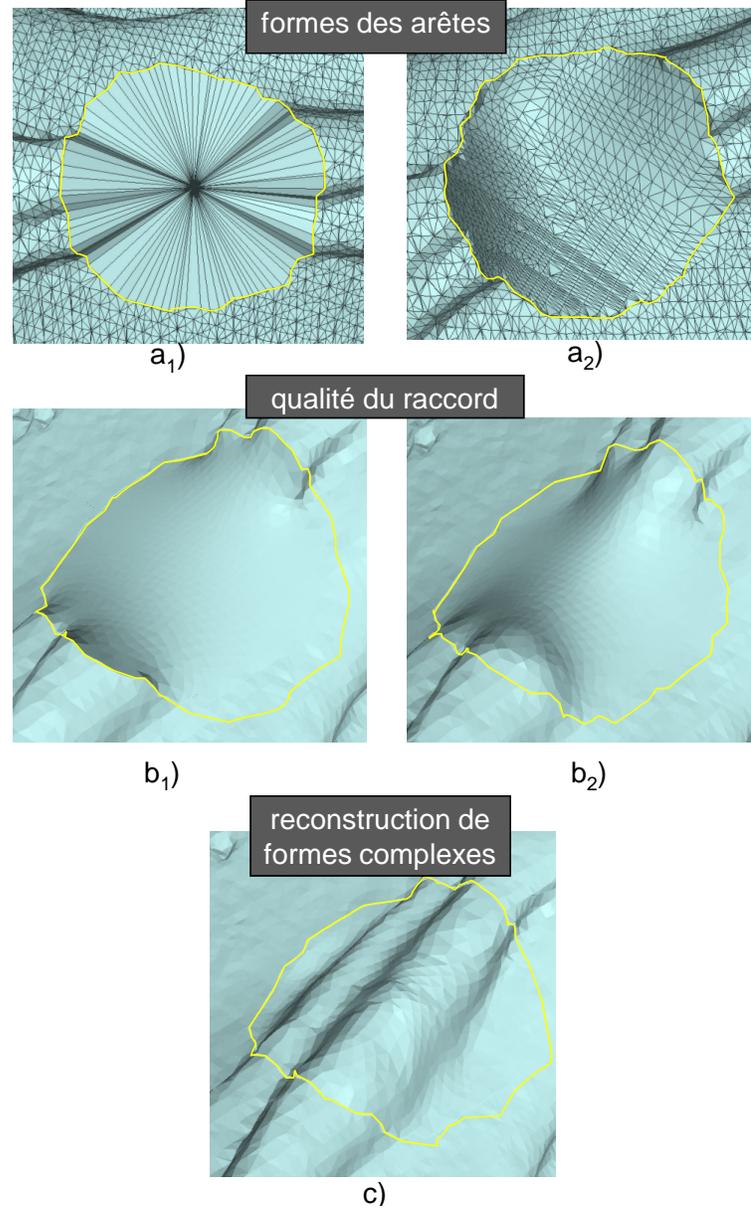


FIGURE 5.5 – Les différentes problématiques du remplissage de trou dans les modèles polyédriques portent sur la forme du maillage inséré (a), la qualité du raccord entre le maillage inséré et le modèle original (b) et sur la reconstruction de formes complexes (c)

Deux méthodes de remplissage de trous à l'aide d'images sont proposées :

- la première méthode utilise deux images de l'objet complet afin d'extraire par stéréoscopie (section 5.4.1) les lignes de caractère définissant la forme présente dans le trou. Ces lignes de caractère 3D sont ensuite utilisées pour contraindre un maillage grossier inséré pour remplir le trou.
- La deuxième méthode consiste à extraire des contraintes directement à partir de l'intensité des pixels d'une image de l'objet complet (section 5.4.2).

5.3 Prétraitements des trous

5.3.1 Identification et nettoyage du contour des trous

Dans les deux méthodes de remplissage de trous développées, le trou est tout d'abord identifié puis nettoyé selon la méthode présentée dans [Pernot *et al.* 2006]. Les modèles polyédriques manipulés étant supposés triangulaires, orientés, manifold et connectés, l'identification du contour du trou est réalisée automatiquement en considérant les arêtes de frontières, i.e. les arêtes qui ne sont connectées qu'à une seule face. A cause du bruit de mesure du scanner, le contour du trou présente des faces de formes indésirables (fig. 5.6.c).

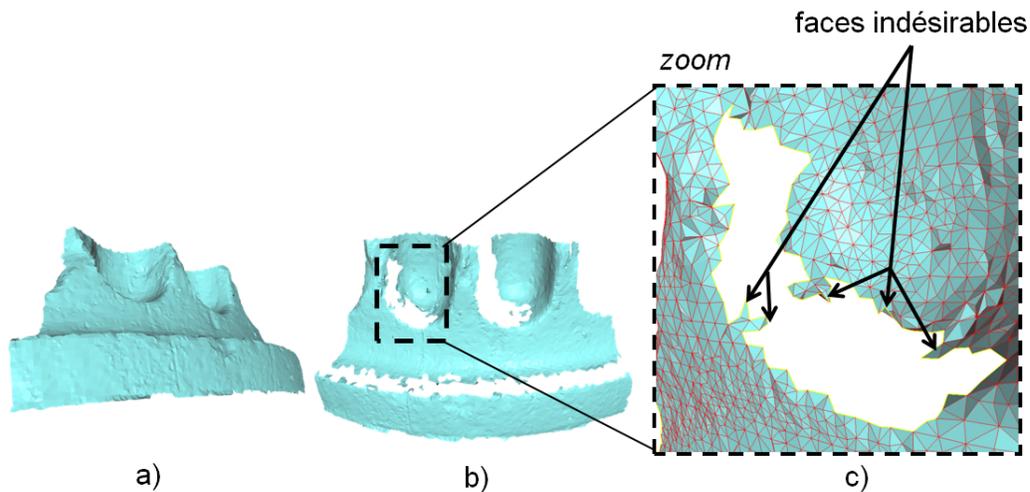


FIGURE 5.6 – *Modèle scanné de la base d'une colonne vue de la station d'acquisition (a), et vue d'une autre position (b) : le contour des trous doit être nettoyé des faces de formes indésirables dues au bruit du laser (c).*

Le contour est donc nettoyé en supprimant tout d'abord les faces dites *dégénérées*, i.e. les faces qui ont un ou deux angles très faibles (fig. 5.7.a et b.) pouvant engendrer des problèmes d'orientation.

Plutôt que de calculer les angles pour chaque face du maillage, on détecte les faces dégénérées en calculant le facteur Q de [Ciarlet 1978] introduit à la section 4.4.1. Ce facteur de qualité est compris dans l'intervalle $[0, 1]$. Il donne une indication sur la forme des faces : plus le facteur Q est proche de 0, plus la face considérée est allongée.

On considère qu'une face est dégénérée lorsque son facteur Q est inférieur à 0.25. Ce seuil a été choisi expérimentalement en analysant l'évolution du facteur Q en fonction des valeurs des angles entre les faces.

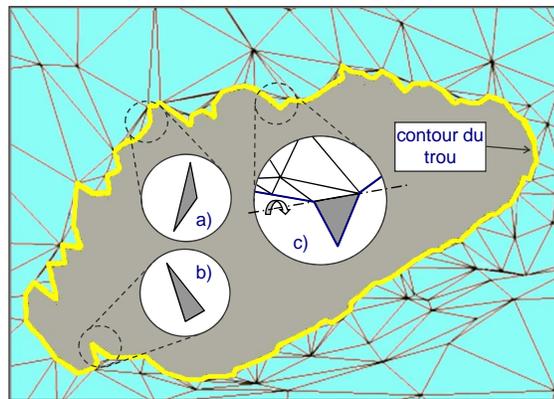


FIGURE 5.7 – Identification du contour du trou et les faces indésirables dues au bruit du scanner laser (a), (b) et (c)

On distingue trois types de faces dégénérées qui sont traitées comme suit :

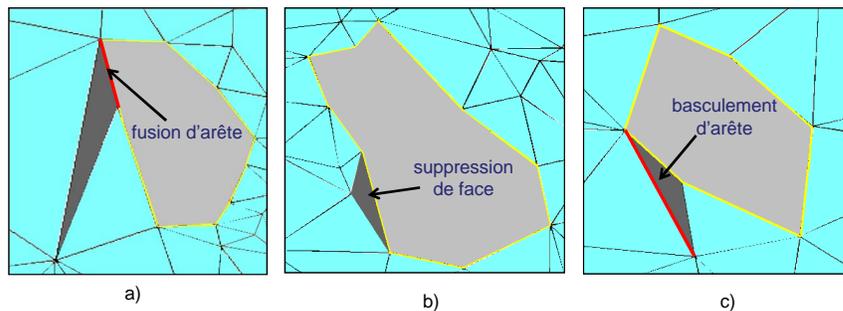


FIGURE 5.8 – Les trois différents types de faces dégénérées.

Algorithme 5 Traitement des faces dégénérées

```
si l'arête de plus petite taille est sur la frontière alors
    cette arête est fusionnée (fig. 5.8.a)
sinon
    si la plus grande des arêtes correspond à l'arête de frontière de la face
    considérée alors
        la face est supprimée (fig. 5.8.b)
    sinon
        si la plus grande des arêtes ne correspond pas à l'arête de frontière
        de la face considérée alors
            cette plus grande arête est permutée par basculement (fig. 5.8.c)
            si les projetés orthogonaux de ces sommets appartiennent à l'arête
            une fois permutée.
        finsi
    finsi
finsi
```

Le contour du trou est ensuite nettoyé des faces dites *mal orientées*, *i.e.* ces faces qui possèdent deux arêtes de frontières (fig. 5.7.c.). Une face définie par deux arêtes de frontière peut "tourner" autour de son arête intérieure sans affecter les autres faces environnantes. Ce degré de liberté, combiné aux incertitudes de calcul des positions des sommets connectés aux arêtes de frontières produit des faces dégénérées du genre de celles de la figure 5.9.a. Pour supprimer itérativement ce genre de faces tout en préservant un certain niveau de détails du contour du trou, une face dégénérée est donc supprimée si et seulement si elle a deux arêtes de frontière et que sa face adjacente ne présente pas elle-même d'arête de frontière (fig. 5.9.b.).

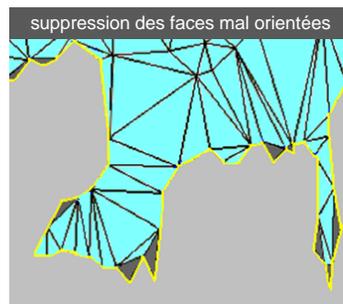


FIGURE 5.9 – *Suppression des faces mal orientées.*

5.3.2 Remplissage topologique de trous

La seconde étape de pré-traitement consiste à remplir le trou par un maillage grossier (fig. 5.10) qui sera ensuite déformé sous diverses contraintes définies aux sections 5.4.1 et 5.4.2.

La technique de remplissage est soit celle développée par [Liepa 2003] et qui produit une triangulation qui minimise l'angle maximum entre deux faces, soit la méthode de [Lou *et al.* 2008] qui est une amélioration de la méthode précédente et qui maximise le facteur d'aspect ratio Q [Ciarlet 1978] présenté en section 4.4.1.

Dans le cas de trous dont les faces du contour sont majoritairement coplanaires, la méthode de [Lou *et al.* 2008] est à privilégier puisque l'aire de la triangulation sera sensiblement la même que celle générée par l'algorithme de [Liepa 2003] tout en produisant des faces de meilleure qualité au regard du facteur Q , c'est-à-dire des faces les plus équilatérales possibles. La figure 5.10.b montre l'application de la méthode de [Liepa 2003] et la figure 5.10.c celle de la méthode de [Lou *et al.* 2008] sur le modèle d'un cube troué 5.10.a.

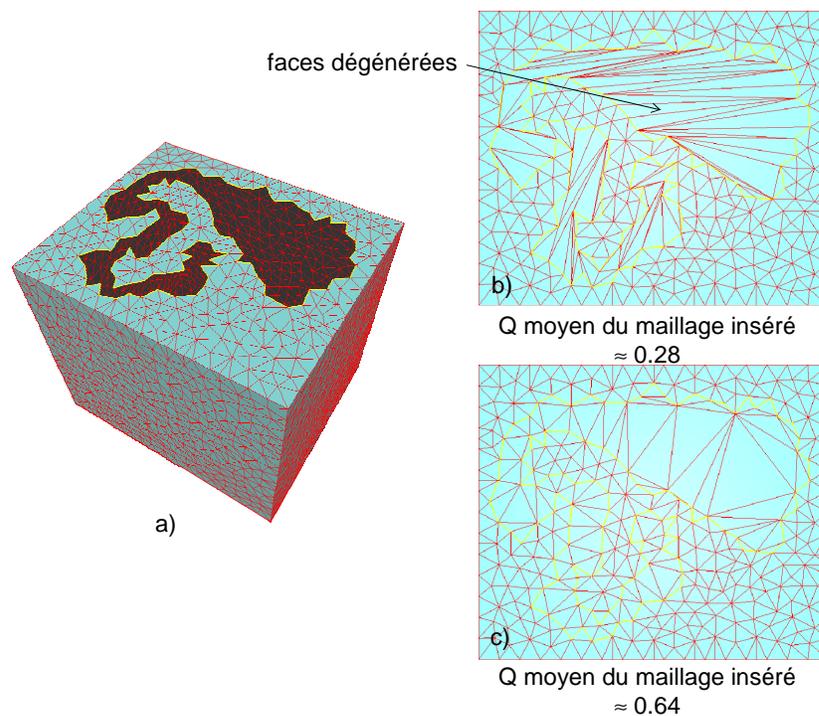


FIGURE 5.10 – Remplissage de trou dans un cube (a) par la méthode [Liepa 2003] (b) et celle de [Lou *et al.* 2008] (c).

5.3.3 Calibration de la caméra

Les deux méthodes de remplissage de trous proposées utilisent toutes les deux des images calibrées (photographies ou captures d'écran). Il faut donc que le plan image 3D correspondant à l'image 2D et le modèle 3D soient calés dans un même repère global de référence lié à la scène. Un module de calibration de caméra a été développé suivant l'algorithme de [Faugeras *et al.* 1987] présenté au chapitre 2 (section 2.3).

Deux types de contraintes peuvent ensuite être imposées sur le maillage inséré. Dans le premier cas, c'est-à-dire lorsque l'on dispose de deux images numériques 2D du produit dont le modèle 3D est troué, des contraintes sur les coordonnées 3D des noeuds du maillage inséré peuvent être écrites en utilisant les équations issues de la triangulation stéréoscopique.

Dans le second cas, seule une image du modèle initial est disponible. Des contraintes de déformation directement liées à l'intensité des pixels correspondant à la zone du trou peuvent être imposées en utilisant les équations du Shape From Shading.

5.4 Définition des contraintes sur le maillage inséré

5.4.1 Définition de contraintes par stéréoscopie

En entrée de la chaîne des processus (fig. 5.11) de la première méthode de remplissage de trous [Panchetti *et al.* 2007a], deux images ou captures d'écran de l'objet sont tout d'abord filtrées avec le filtre de Sobel [Sobel 1990] pour en révéler les contours 2D.

Puis, par triangulation stéréoscopique [Goldman 1990], les lignes de caractère 3D de la forme manquante sont reconstruites (voir section 2.4.2). Ces lignes de caractère 3D sont ensuite discrétisées en polygones composées de N_s points géométriques \mathbf{P}_s^g de coordonnées (X_s, Y_s, Z_s) (fig.5.12).

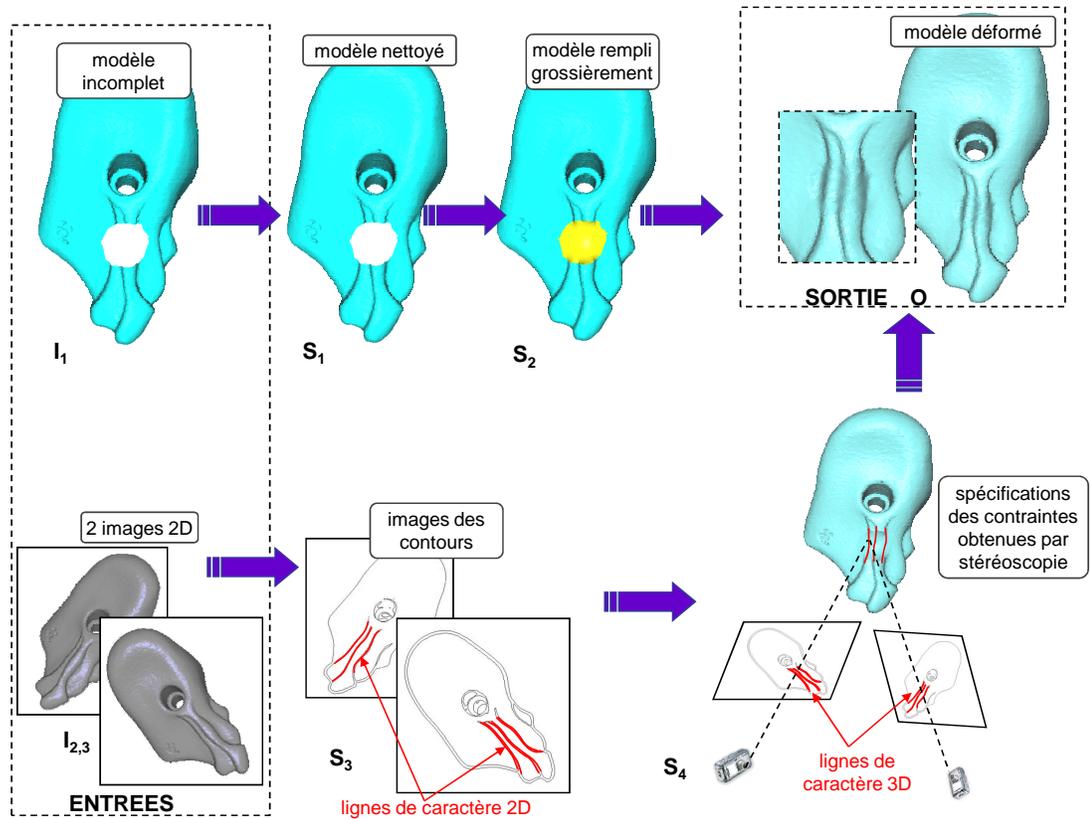


FIGURE 5.11 – Chaîne de processus de la méthode de remplissage de trous à l'aide d'informations obtenues par stéréoscopie.

Chaque point \mathbf{P}_s^g est projeté en un point paramétrique \mathbf{P}_s sur le maillage inséré, ce qui a pour résultat l'obtention d'une contrainte ponctuelle liant les deux entités [Pernot 2004]. Chaque point paramétrique \mathbf{P}_s est défini par ses coordonnées barycentriques $(\alpha_s, \beta_s, \gamma_s)$ dans la face composée des sommets $(\mathbf{V}_s^m, \mathbf{V}_s^n, \mathbf{V}_s^k)$ telles que (fig.5.12) :

$$\begin{cases} X_s = \alpha_s \cdot X_m + \beta_s \cdot X_n + \gamma_s \cdot X_k \\ Y_s = \alpha_s \cdot Y_m + \beta_s \cdot Y_n + \gamma_s \cdot Y_k \\ Z_s = \alpha_s \cdot Z_m + \beta_s \cdot Z_n + \gamma_s \cdot Z_k \end{cases} \quad (5.1)$$

Pour chaque point géométrique d'une polyligne contraignant un point paramétrique du maillage inséré, la contrainte correspondante peut s'écrire sous la forme suivante :

$$P_s = P_s^g \tag{5.2}$$

En imposant que les coordonnées barycentriques restent constantes durant le processus de déformation (5.1) et les N_s contraintes (5.2) pour les N_s points paramétriques P_s , l'ensemble de ces équations linéaires définit le vecteur G tel que :

$$\{ G(\mathbf{X}) = 0 \tag{5.3}$$

avec \mathbf{X} , le vecteur inconnu des positions des noeuds libres du maillage inséré.

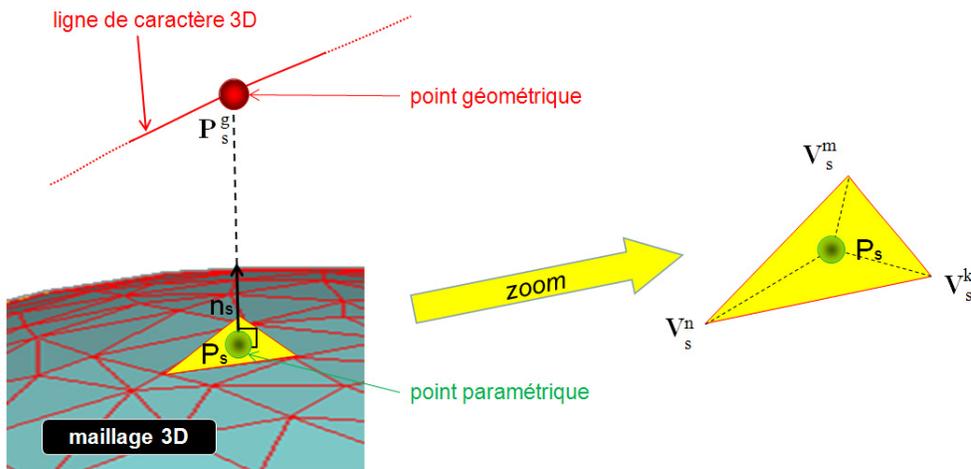


FIGURE 5.12 – Contrainte imposée par un point géométrique P_s^g de la polyligne d'une ligne de caractère sur un point paramétrique appartenant à la face du maillage définie par les trois sommets (V_s^m, V_s^n, V_s^k) .

5.4.2 Définition de contraintes à l'aide des méthodes de Shape From Shading

En entrée de la chaîne des processus (fig. 5.13) de la deuxième méthode de remplissage de trous [Panchetti *et al.* 2008, Panchetti *et al.* 2009], une seule image ou capture d'écran de l'objet est nécessaire pour en extraire des informations relatives à la forme manquante.

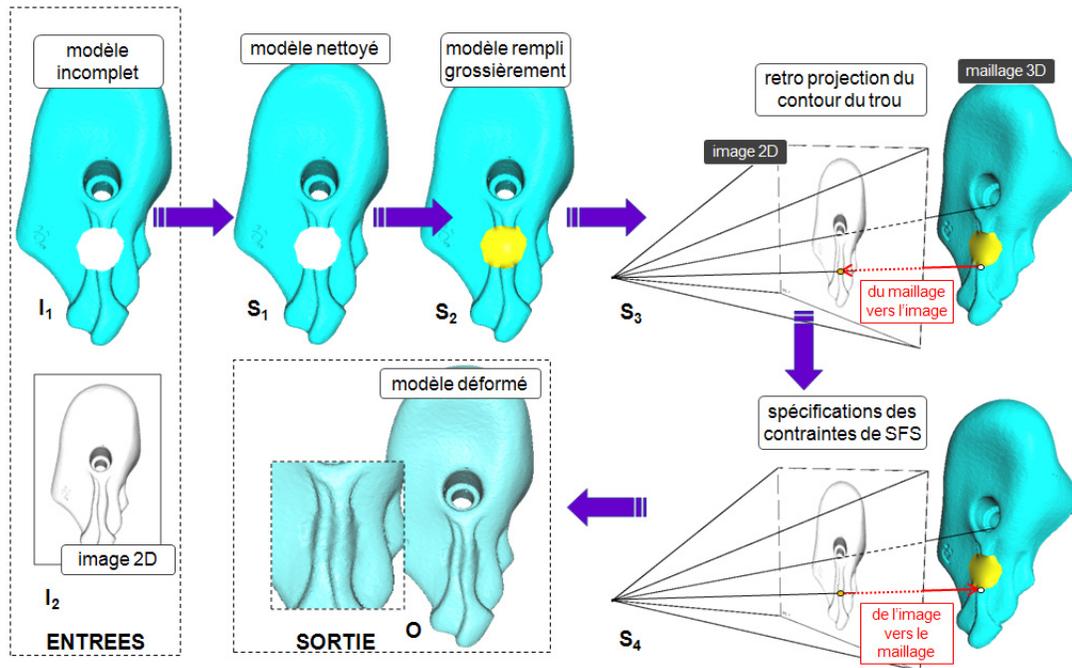


FIGURE 5.13 – Chaîne de processus de la méthode de remplissage de trous à l'aide d'informations issues des méthodes de Shape From Shading.

L'étape de calibration ayant été réalisée (section 5.3.3), on suppose la matrice fondamentale de projection M connue. Elle lie les coordonnées (X_{ij}, Y_{ij}, Z_{ij}) d'un point P_{ij} aux coordonnées $(x_i, y_j, u(x_i, y_j))$ du point correspondant p_{ij} dans l'image utilisée en entrée de la chaîne de processus (fig. 5.14). Cette matrice est connue soit parce que la position et les paramètres de la caméra qui a pris la photographie ont été enregistrés par un GPS, soit parce que les images ont été calibrées *a posteriori* [Horaud *et al.* 1995, Panchetti *et al.* 2007]. Les méthodes de calibration de caméras calculent les paramètres de la matrice M tels que :

$$\begin{bmatrix} x_i \\ y_j \\ u(x_i, y_j) \end{bmatrix} = M \cdot \begin{bmatrix} X_{ij} \\ Y_{ij} \\ Z_{ij} \\ 1 \end{bmatrix} \quad (5.4)$$

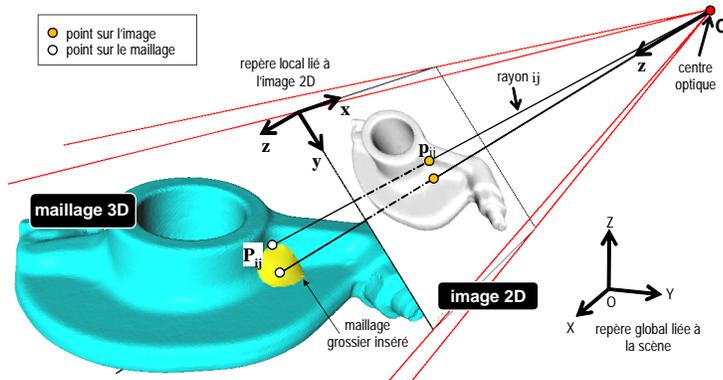


FIGURE 5.14 – Lien entre le modèle 3D et l'image 2D réalisé par l'étape de calibration de caméra.

L'image 2D étant calibrée, l'étape de rétro-projection des sommets du modèle 3D appartenant au contour du trou dans le plan image peut commencer [Haines 1994]. On identifie ainsi dans l'image, la zone de pixels (délimitée par une polygone fermée) définissant la forme manquante dans le modèle 3D (fig. 5.15).

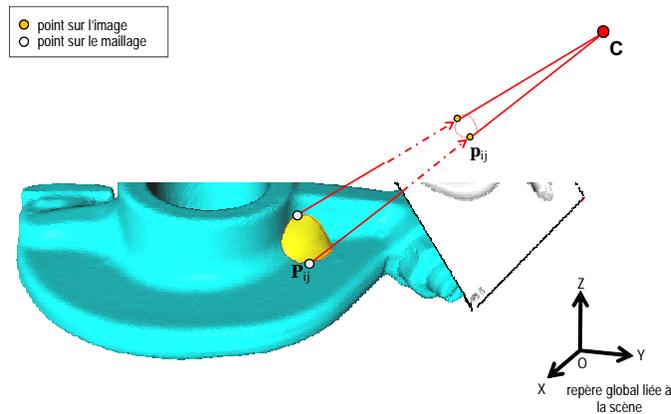


FIGURE 5.15 – Identification du contour du trou dans l'image par rétro-projection des sommets du polyèdre.

Pour ne pas surcontraindre le maillage à déformer, seuls certains pixels de cette zone vont être utilisés lors de la spécification des contraintes. Pour cela, on crée une grille 2D dont le pas peut varier en fonction du nombre d'inconnues, *i.e.*, du nombre de sommets du maillage inséré. En effet, plus le pas de la grille sera faible, plus le maillage sera contraint localement.

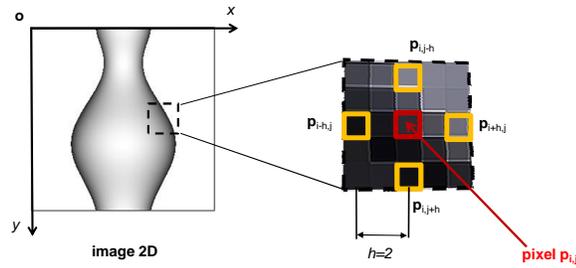


FIGURE 5.16 – Découpage de la zone des pixels à projeter sur le polyèdre par une grille 2D à pas h réglable. Ici, le pas est de 2 pixels, donc un pixel sur 2 de la zone sera projeté sur le modèle 3D.

Trois types de contraintes sont alors imposées sur le maillage inséré.

Contrainte de localisation sur la face d'origine : coordonnées barycentriques constantes

Le premier type de contrainte imposé aux noeuds libres du maillage inséré concerne la localisation de la projection sur une face du modèle. Ici nous faisons l'hypothèse qu'elle doit être maintenue durant l'étape de déformation. En fait, cette hypothèse pourrait être levée en laissant les points glisser sur la surface du polyèdre durant la déformation. Ceci aurait pour conséquence de relaxer la zone contrainte. Cette option n'a pas été implémentée et testée.

Ainsi, pour chaque pixel (x_i, y_j) de la zone identifiée, une projection sur le maillage inséré est réalisée afin d'obtenir les coordonnées barycentriques sur la face (fig. 5.17) :

$$\begin{cases} X_{ij} = \alpha_{ij} \cdot X_m + \beta_{ij} \cdot X_n + \gamma_{ij} \cdot X_k \\ Y_{ij} = \alpha_{ij} \cdot Y_m + \beta_{ij} \cdot Y_n + \gamma_{ij} \cdot Y_k \\ Z_{ij} = \alpha_{ij} \cdot Z_m + \beta_{ij} \cdot Z_n + \gamma_{ij} \cdot Z_k \end{cases} \quad (5.5)$$

avec $(\alpha_{ij}, \beta_{ij}, \gamma_{ij})$ les coefficients de la position barycentrique dans une face définie par ses trois sommets $(\mathbf{V}_m, \mathbf{V}_n, \mathbf{V}_k)$ de coordonnées (X_m, Y_m, Z_m) , (X_n, Y_n, Z_n) et (X_k, Y_k, Z_k) .

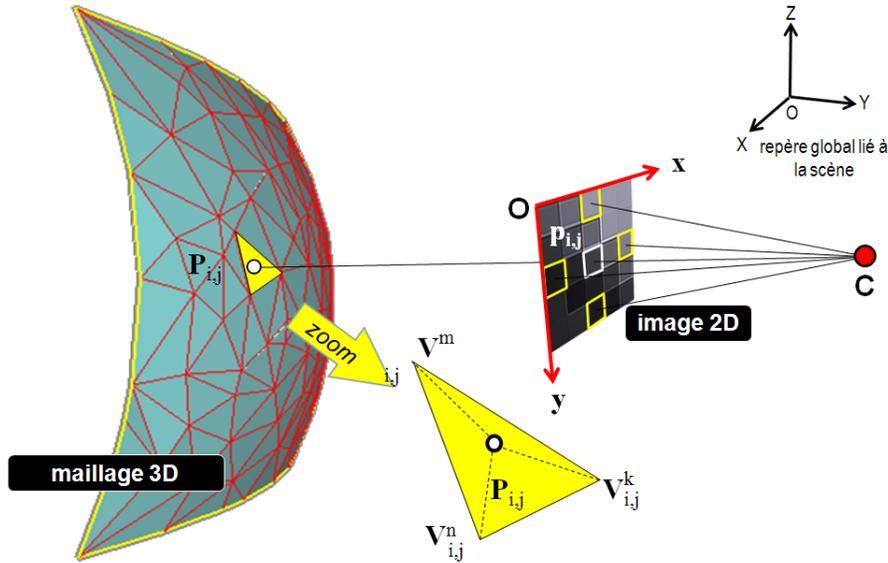


FIGURE 5.17 – Contrainte de localisation : les coordonnées barycentriques du point P_{ij} sur la face définie par les trois sommets (V_m, V_n, V_k) à laquelle il appartient demeurent constantes durant le processus de déformation.

Contrainte d'élévation : discrétisation de l'équation de l'Eikonal

Le second type de contrainte à imposer aux noeuds libres du maillage inséré porte sur l'élévation u_{ij} d'un pixel p_{ij} par rapport au repère local (O, x, y, z) (fig. 5.18).

Ainsi, étant donné un pixel p_{ij} de la grille 2D de pas h réglable, dont l'intensité lumineuse $I(x_i, y_j)$ est donc connue, l'équation de l'Eikonal (5.6) s'écrit :

$$\left[\frac{\partial u}{\partial x}(x_i, y_j) \right]^2 + \left[\frac{\partial u}{\partial y}(x_i, y_j) \right]^2 = \frac{1}{I(x_i, y_j)^2} \quad (5.6)$$

où $I(x_i, y_j)$ est l'intensité du pixel connue. Cette équation est discrétisée et approchée en utilisant le schéma aux différences finies centré suivant :

$$\left[\frac{u_{i+h,j} - u_{i-h,j}}{\Delta x_{ij,h}} \right]^2 + \left[\frac{u_{i,j+h} - u_{i,j-h}}{\Delta y_{ij,h}} \right]^2 = \frac{1}{I(x_i, y_j)^2} - 1 \quad (5.7)$$

où h est le pas de la grille utilisée (fig. 5.16) qui correspond aussi à l'intervalle du schéma aux différences finies. Les quantités $\Delta x_{ij,h}$ et $\Delta y_{ij,h}$ sont définies ci-après (éq. 5.8).

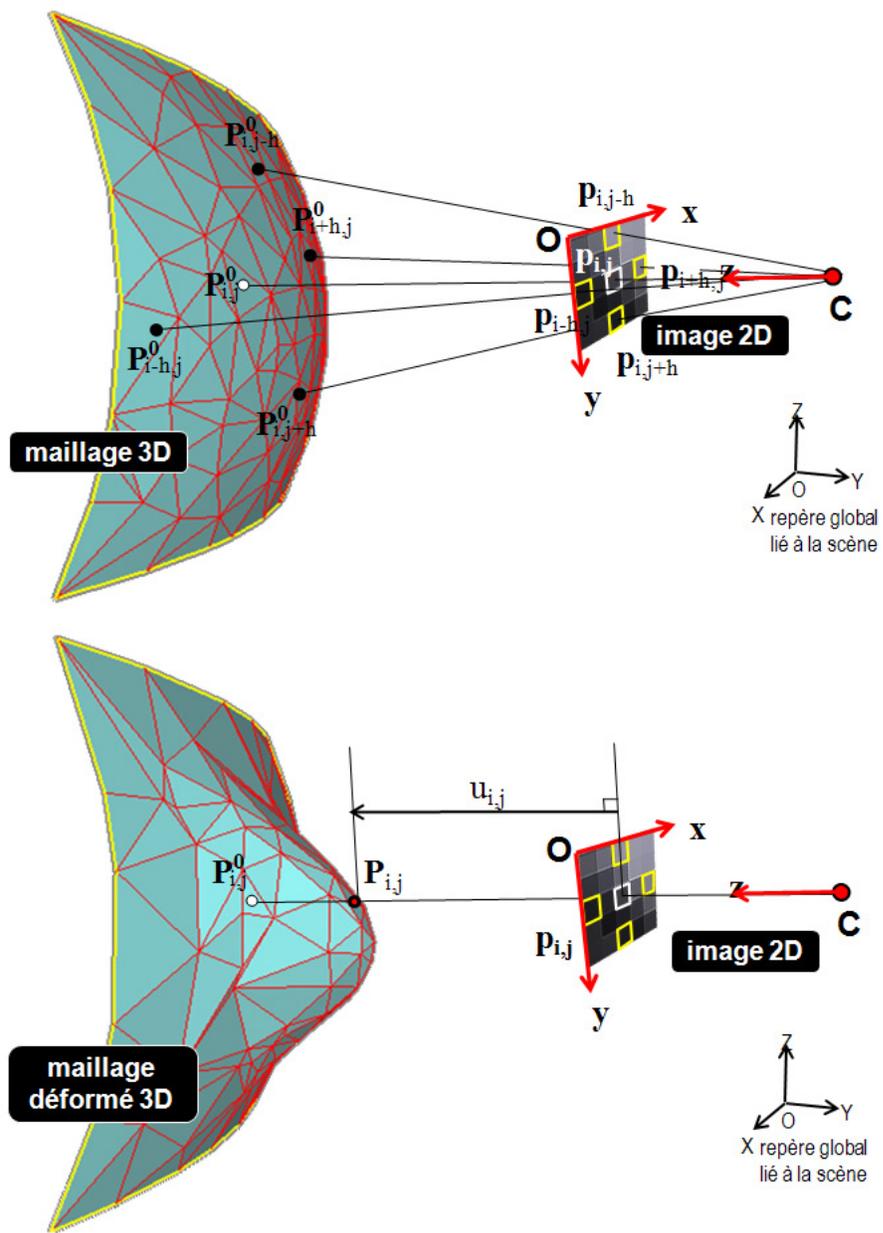


FIGURE 5.18 – Contrainte d'élévation imposée à la projection P_{ij} d'un pixel p_{ij} selon l'axe optique (orthogonale au plan image).

Contrainte directionnelle de déplacement suivant l'axe optique

Le troisième type de contrainte imposé aux noeuds libres du maillage inséré concerne la direction de déplacement des noeuds le long d'axes parallèles à l'axe optique (fig.5.18).

Cette hypothèse pourrait être revue en forçant les noeuds à se déplacer le long d'axes définis par les directions de projection des pixels et passant tous par le centre optique C . Cette option n'a pas été implémentée et testée.

Le scalaire u_{ij} caractérise l'élévation du point 3D \mathbf{P}_{ij} avec \mathbf{z} l'axe optique. $\Delta x_{ij,h}$ et $\Delta y_{ij,h}$ correspondent aux pas de la grille 3D sur le maillage inséré, suivant les directions \mathbf{x} et \mathbf{y} , respectivement. $\Delta x_{ij,h}$ et $\Delta y_{ij,h}$ sont constantes durant le processus de déformation.

Afin d'établir le lien entre les quantités u_{ij} et les coordonnées des points \mathbf{P}_{ij} , les contraintes suivantes sont spécifiées :

$$\left\{ \begin{array}{l} \mathbf{P}_{ij}^0 \mathbf{P}_{ij} \cdot \mathbf{x} = 0 \\ \mathbf{P}_{ij}^0 \mathbf{P}_{ij} \cdot \mathbf{y} = 0 \\ \mathbf{OP}_{ij} \cdot \mathbf{z} = u_{ij} \end{array} \right. \text{ et } \left\{ \begin{array}{l} \Delta x_{ij,h} = \mathbf{P}_{i-h,j}^0 \mathbf{P}_{i+h,j}^0 \cdot \mathbf{x} \\ \Delta y_{ij,h} = \mathbf{P}_{i,j-h}^0 \mathbf{P}_{i,j+h}^0 \cdot \mathbf{y} \end{array} \right. \quad (5.8)$$

\mathbf{P}_{ij}^0 correspond à la position initiale de \mathbf{P}_{ij} sur le maillage inséré, i.e. avant la déformation.

En écrivant les équations (5.6), (5.7), et (5.8) pour chaque pixel de la grille 2D de la zone considérée dans l'image, nous obtenons un système d'équations non-linéaires contraignant les coordonnées du maillage inséré.

Dans ces équations, certains points projetés \mathbf{P}_{ij} correspondent à des points à l'extérieur du trou. Leurs coordonnées sont donc constantes et permettent de tenir compte de la forme environnant le trou.

Ces contraintes sont rajoutées dans un vecteur G tel que :

$$\left\{ \mathbf{G}(\mathbf{X}) = \mathbf{0} \right. \quad (5.9)$$

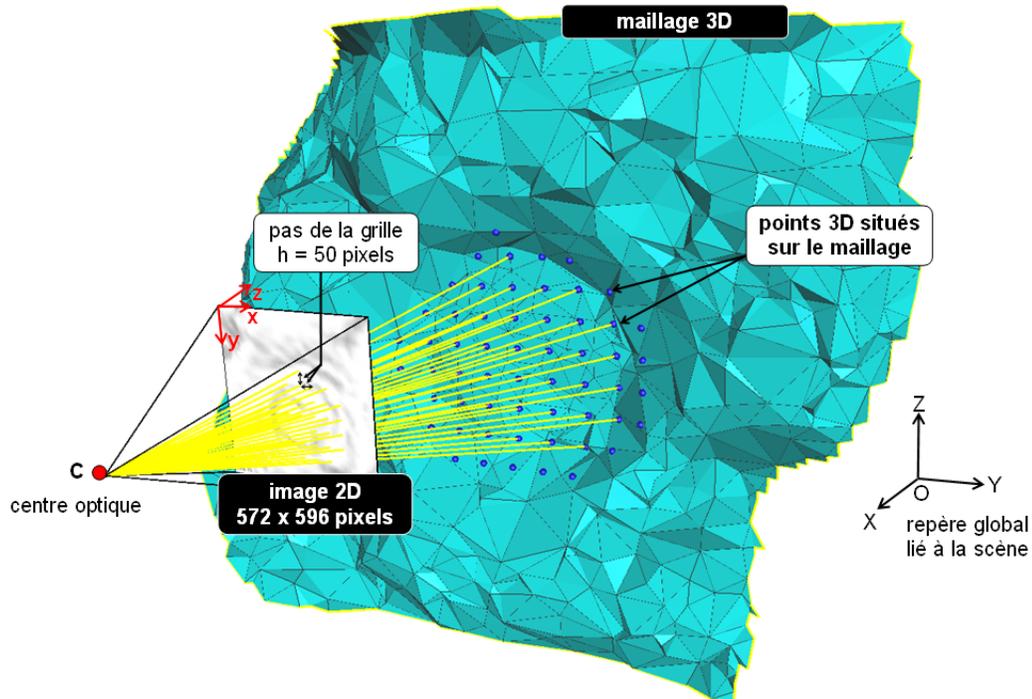


FIGURE 5.19 – Contraintes de SFS directement exprimées sur le maillage 3D par un ensemble de pixels situés sur une grille cartésienne de l’image 2D (pas de la grille $h = 50$ pixels entre les lignes et les colonnes).

5.5 Spécification et résolution du problème d’optimisation numérique

5.5.1 Choix de la fonctionnelle

Comme expliqué au chapitre 3, section 3.3.2, le moteur de déformation utilisé est basé sur un modèle mécanique de réseau de barres [Pernot 2004]. La déformation du modèle polyédrique revient à résoudre un problème d’optimisation :

$$\begin{cases} \mathbf{G}(\mathbf{F}) = \mathbf{0} \\ \min \phi(\mathbf{F}) \end{cases} \quad (5.10)$$

où le vecteur inconnu \mathbf{F} contient les composantes de tous les efforts extérieurs appliqués aux noeuds libres. $\phi(\mathbf{F})$ est une fonctionnelle permettant de choisir une solution parmi toutes celles satisfaisant le jeu de contraintes $\mathbf{G}(\mathbf{F}) = \mathbf{0}$.

Les contraintes géométriques ainsi spécifiées sur le modèle polyédrique

permettent de *sculpter* l'intérieur du maillage inséré. Afin d'assurer un bon raccord avec le modèle initial, il faut choisir une fonctionnelle à minimiser. En théorie, n'importe quelle combinaison de quantités s'exprimant comme une forme quadratique du vecteur inconnu peut être utilisée pour définir la fonctionnelle $\phi(\mathbf{F})$.

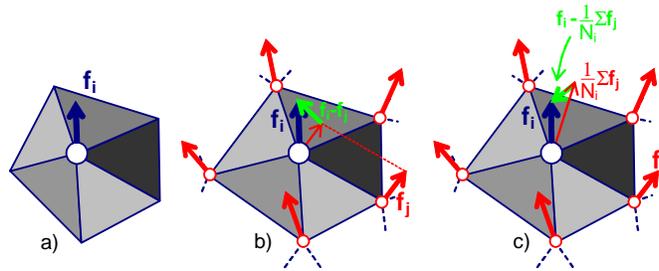


FIGURE 5.20 – Forces extérieures prises en compte lors de la minimisation des efforts extérieurs (b) appliqués au ième noeud (a) ou celle minimisant les variations de cet effort par rapport aux efforts voisins (c).

Parmi les différentes quantités qui ont déjà été imaginées et qui pourraient être utilisées dans le contexte du bouchage de trous, il y a la minimisation des efforts extérieurs [Pernot *et al.* 2006] :

$$\phi_F = \sum_i \mathbf{f}_i^2 = \sum_i f_{x_i}^2 + f_{y_i}^2 + f_{z_i}^2 \quad (5.11)$$

Les efforts extérieurs appliqués aux noeuds libres étant tous indépendants les uns des autres (fig. 5.20.a), la minimisation de cette somme sur tous les noeuds donnera toujours un vecteur inconnu nul (pas de termes croisés dans la forme quadratique). Cette fonctionnelle aura donc tendance à minimiser la courbure de la géométrie sous-jacente tout en produisant des zones les plus plates possibles (fig. 5.21.a). De plus, on peut remarquer que chacun des efforts extérieurs ne dépend que de la position du noeud sur lequel il est appliqué et de la position des noeuds directement connectés à celui-ci.

Par conséquent, seules les informations relatives à la position des noeuds du contour du trou sont considérées. Ceci ne permet pas de satisfaire nos exigences puisque, dans ce cas, ni l'évolution de la tangence ni l'évolution de la courbure n'est préservée au passage du raccord entre les deux maillages. Pour prendre en compte plus d'informations sur le maillage adjacent, on peut rajouter dans la fonction objectif les efforts appliqués aux noeuds bloqués formant le contour du trou. En utilisant une combinaison des équations

liant les positions de noeuds aux efforts appliqués, on obtient une forme quadratique avec des termes carrés et des termes croisés.

La zone d'influence est ainsi plus grande, vu qu'elle utilise des informations géométriques relatives à la fois aux noeuds libres, aux noeuds du contour du trou et au premier anneau du maillage adjacent, *i.e.* aux noeuds qui sont directement connectés aux noeuds du contour du trou (fig. 5.21.b). Malheureusement, une telle minimisation ne peut prétendre plus que la préservation de la tangence au travers du raccord du bord de trou.

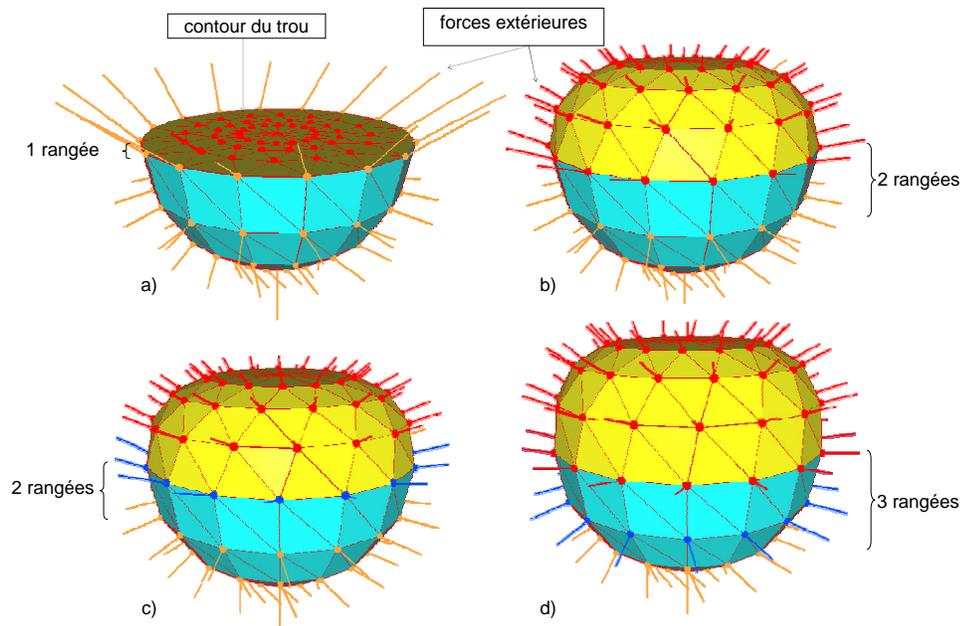


FIGURE 5.21 – Minimisation de ϕ_F appliquée aux noeuds libres (a) et aux noeuds du contour (b). Minimisation de $\phi_{\Delta_r F}$ appliquée aux noeuds libres (c) et aux noeuds du contour (d).

Pour atteindre le niveau de préservation de la courbure, il faudrait utiliser les informations géométriques relatives au deuxième anneau du maillage adjacent. En remarquant que la variation entre deux efforts adjacents représente en quelque sorte l'évolution de la courbure entre les noeuds adjacents, la minimisation suivante a été proposée [Pernot *et al.* 2006] :

$$\phi_{\Delta_r F} = \sum_i \sum_j [\mathbf{f}_i - \mathbf{f}_j]^2 \quad (5.12)$$

Cette fonctionnelle permet la minimisation des variations entre des couples d'efforts extérieurs appliqués à des noeuds connectés entre eux. La zone d'influence de ces quantités de base est plus large que dans le cas précédent (fig. 5.21.c). Ceci est dû à la fois au fait que tous les efforts appliqués aux noeuds j , directement connectés à un noeud i donné, sont également pris en compte (fig. 5.20.b) et au fait que chaque force extérieure est influencée à la fois par la position du noeud sur lequel elle agit et par la position des noeuds directement connectés à ce dernier. Mais nos exigences ne sont toujours pas satisfaites si cette minimisation est appliquée sur tous les noeuds libres (fig. 5.21.c). Comme précédemment, seul le premier anneau du maillage adjacent est utilisé. Afin de prendre en compte le deuxième anneau, les quantités de base relatives aux noeuds du contour sont également minimisées (fig. 5.21.d). Dans ce cas, l'évolution de la courbure est préservée au niveau du raccord entre les deux maillages. En fait, la minimisation de la variation de courbure est approximée par la minimisation de la variation relative des efforts extérieurs. Ici, on peut remarquer que les deux premiers anneaux du maillage intérieur semblent être obtenus par symétrie des deux premiers anneaux du maillage adjacent par rapport à un plan qui contiendrait le contour du trou.

Pour finir, il n'y a pas d'intérêt à utiliser plus d'anneaux du maillage adjacent vu que les termes supplémentaires seraient des constantes qui n'affecteraient pas le résultat de la minimisation. Être capable de minimiser la variation de courbure entre les deux maillages est par ailleurs suffisant pour la plupart des applications de conception mécanique.

5.5.2 Linéarisation des contraintes

Dans le cas de l'utilisation des contraintes de SFS (section 5.4.2), une étape de linéarisation du vecteur \mathbf{G} au premier ordre est nécessaire. Cette linéarisation n'est pas nécessaire dans le cas des contraintes issues de la stéréoscopie (section 5.4.1). En posant $\mathbf{G}^{[0]} = \mathbf{G}(\mathbf{F}^{[0]})$ le vecteur des contraintes initial, $\mathbf{F}^{[0]}$ le vecteur initial des inconnues, les forces extérieures à l'itération k on peut écrire :

$$\mathbf{F}^{[k]} = \mathbf{F}^{[k-1]} + \Delta \mathbf{F}^{[k]} \quad (5.13)$$

avec $\mathbf{F}^{[k]}$ spécifié par l'équation $\mathbf{G}^{[k]} = \mathbf{G}(\mathbf{F}^{[k]}) = \mathbf{0}$. La linéarisation du vecteur des contraintes au premier ordre produit le système d'équations suivant :

$$\mathbf{G}^{[k-1]} + {}^t \mathbf{G}^{[k-1]} \cdot \Delta \mathbf{F}^{[k]} = \mathbf{0} \quad (5.14)$$

où la matrice ${}^t\mathbf{G}^{[k-1]}$ contient les dérivées partielles des équations non-linéaires par rapport aux forces extérieures.

La fonctionnelle à minimiser étant une forme quadratique des forces extérieures, on peut l'écrire sous la forme :

$$\phi(\mathbf{F}^{[k]}) = \frac{1}{2} {}^t\Delta\mathbf{F}^{[k]} \mathbf{C}^{[k-1]} \Delta\mathbf{F}^{[k]} - {}^t\mathbf{D}^{[k-1]} \Delta\mathbf{F}^{[k]} + cst \quad (5.15)$$

Afin de résoudre le problème d'optimisation numérique ainsi obtenu (5.14 et 5.15), un ensemble additionnel d'équations est défini :

$$\mathbf{C}^{[k-1]} \Delta\mathbf{F}^{[k]} + \mathbf{A}^{[k-1]} \boldsymbol{\lambda}^{[k]} = \mathbf{B}^{[k-1]} \quad (5.16)$$

où $\boldsymbol{\lambda}^{[k]}$ est le vecteur Lagrangien à l'itération k , $\mathbf{B}^{[k-1]} = -\mathbf{G}^{[k-1]}$ et $\mathbf{A}^{[k-1]} = {}^t\mathbf{G}^{[k-1]}$. On obtient alors le système suivant à résoudre à chaque itération :

$$\begin{bmatrix} \mathbf{C}^{[k-1]} & \mathbf{A}^{[k-1]} \\ {}^t\mathbf{A}^{[k-1]} & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta\mathbf{F}^{[k]} \\ \boldsymbol{\lambda}^{[k]} \end{bmatrix} = \begin{bmatrix} \mathbf{D}^{[k-1]} \\ \mathbf{B}^{[k-1]} \end{bmatrix} \quad (5.17)$$

A la fin de ce processus itératif de résolution, on obtient les composantes des forces extérieures appliquées aux noeuds du maillage inséré. A l'aide des relations linéaires liant les positions des noeuds aux forces extérieures, on retrouve les nouvelles coordonnées des noeuds après déformation (éq. 3.8).

5.6 Résultats et comparaison des deux approches de remplissage de trous proposées

5.6.1 Résultats de la méthode de remplissage de trous à l'aide de contraintes obtenues par stéréoscopie

L'exemple proposé porte sur une configuration pour laquelle le modèle polyédrique initial d'une prise d'escalade artificielle nécessite une mise à jour après une opération de polissage (fig. 5.22.a).

Pour tester les algorithmes, un trou est créé dans une zone complexe. Une triangulation grossière de 232 noeuds obtenue avec la méthode de [Lou *et al.* 2008] est utilisée pour remplir le trou : les positions des 76 noeuds du contour du trou sont fixées alors que celles des 156 noeuds internes sont libres de varier pendant le processus de déformation.

Le raccord avec le maillage initial peut être obtenu en minimisant soit les efforts extérieurs soit les variations relatives de ces efforts (fig. 5.22. b_i et 5.22. c_i).

Même si ces minimisations produisent une forme qui se raccorde bien avec le maillage initial, la partie ainsi réparée demeure facilement identifiable. Pour retrouver la forme initiale, des ajustements de forme sont réalisés en spécifiant un jeu de contraintes supplémentaire obtenu à partir de six lignes de caractère 3D extraites des images 2D (fig. 5.23).

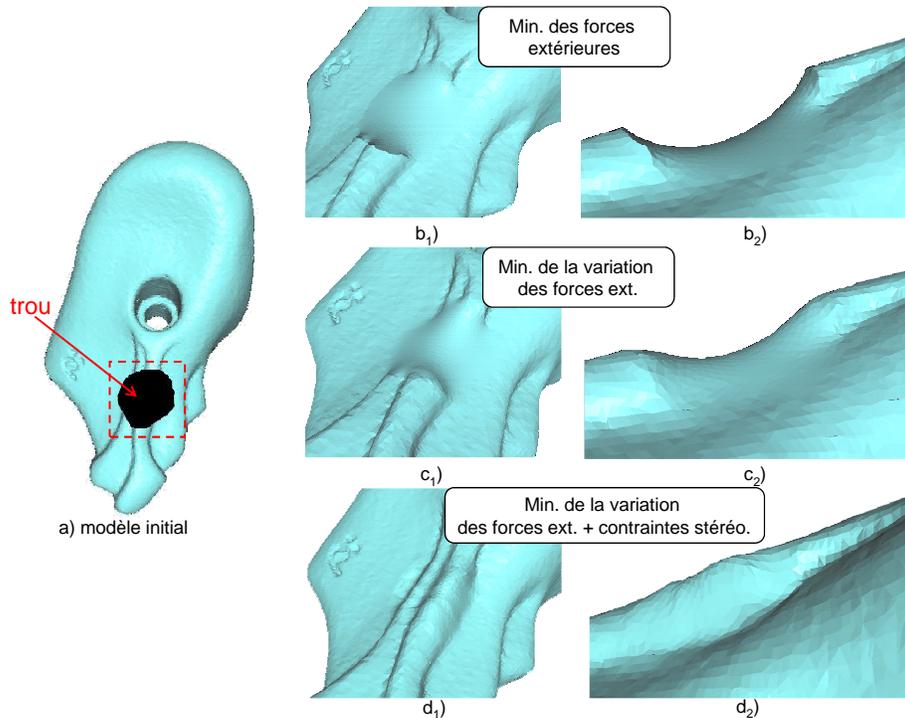


FIGURE 5.22 – Remplissage de trou à l'aide de contraintes issues de la stéréoscopie.

Deux images 2D calibrées du modèle 3D sont utilisées, constituées chacune de 26244 pixels. Elles sont filtrées avec l'algorithme de [Sobel 1990], puis vectorisées par l'algorithme de [Chai *et al.* 1992] en 6 courbes 2D (fig. 5.23.c). Dix correspondances de pixels réalisées par l'utilisateur pour chaque courbe permettent de calculer les coordonnées du point 3D correspondant par la méthode décrite au chapitre 2 (section 2.4.2).

L'ensemble de points 3D résultant sert à la définition de courbes *spline* qui sont ensuite discrétisées et projetées afin d'obtenir une répartition homogène des contraintes imposées sur le maillage inséré. 24 projections de

points appartenant aux lignes de caractère ont été utilisées pour contraindre le maillage inséré (fig. 5.23.d).

Ces contraintes plus la minimisation de la variation de courbure approximée forment un problème d'optimisation dont la résolution donne une forme proche de celle initialement supprimée (fig. 5.22.d_i) en une itération.

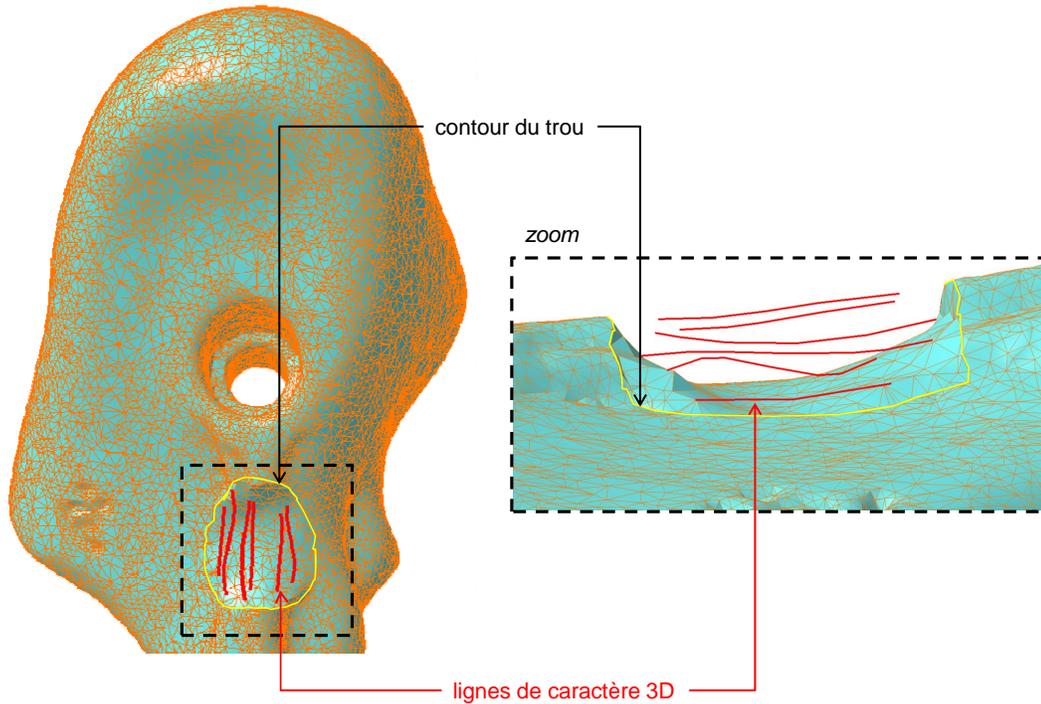


FIGURE 5.23 – Modèle polyédrique de la prise d’escalade artificielle (*courtoisie de Tomoadour*) rempli grossièrement et les lignes de caractère 3D obtenue par triangulation stéréoscopique.

Pour comparer la qualité et la précision de la reconstruction, nous utilisons la distance de Hausdorff implémentée dans le logiciel METRO [Cignoni *et al.* 1998]. Premièrement, la distance de Hausdorff est mesurée entre le maillage initial et le maillage obtenu en minimisant uniquement la variation de courbure (fig. 5.22.c_i).

Le résultat obtenu avec notre algorithme soumis aux contraintes de stéréoscopie (fig. 5.22.d_i) est ensuite comparé au modèle 3D initial. Comme décrit par le tableau 5.1, le résultat obtenu avec notre méthode est plus

proche du modèle initial que celui obtenu par simple minimisation de la variation de courbure. La figure 5.24 montre la distribution de la déviation locale entre les deux modèles résultants.

Modèle de la prise d'escalade	Variation de la courbure vs. modèle initial	Variation de la courbure + contraintes de stéréoscopie vs. modèle initial
distance de Hausdorff	6.546	1.099

TABLE 5.1 – Comparaison quantitative entre les résultats obtenus soit en minimisant seulement la variation de courbure soit en minimisant la variation de courbure et en satisfaisant les contraintes de stéréoscopie sur le modèle de la prise d'escalade.

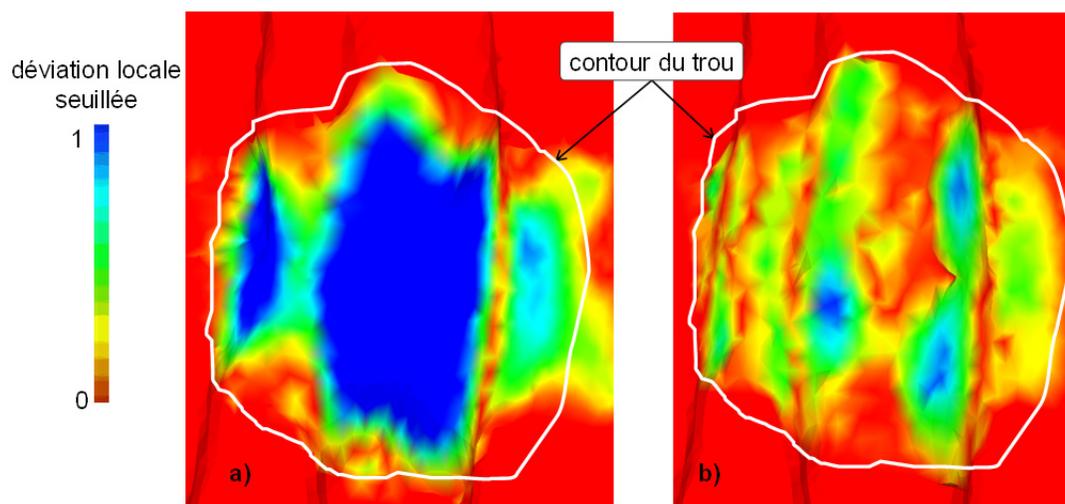


FIGURE 5.24 – Comparaison de la déviation locale entre le modèle initial de la prise d'escalade et le modèle obtenu avec la méthode des densités de forces (a), et entre le modèle initial et la méthode des densités de forces avec l'ajout des contraintes obtenues par stéréoscopie (b).

5.6.2 Résultats de la méthode de remplissage de trous à l'aide de contraintes obtenues par SFS

Remplissage de trou pour des courbes 2D

Afin d'illustrer les capacités de la méthode de remplissage de trous à l'aide d'informations extraites par SFS, nous avons tout d'abord considéré un exemple en 2D d'une courbe continue et dérivable dont la fonction d'intensité lumineuse peut être calculée analytiquement (fig. 5.25) :

$$u(x) = \frac{1}{\sqrt{1 + (1.3x - 0.35x^3)^2}} \text{ avec } I(x) = \sqrt{\frac{1}{1 + u'(x)^2}} \quad (5.18)$$

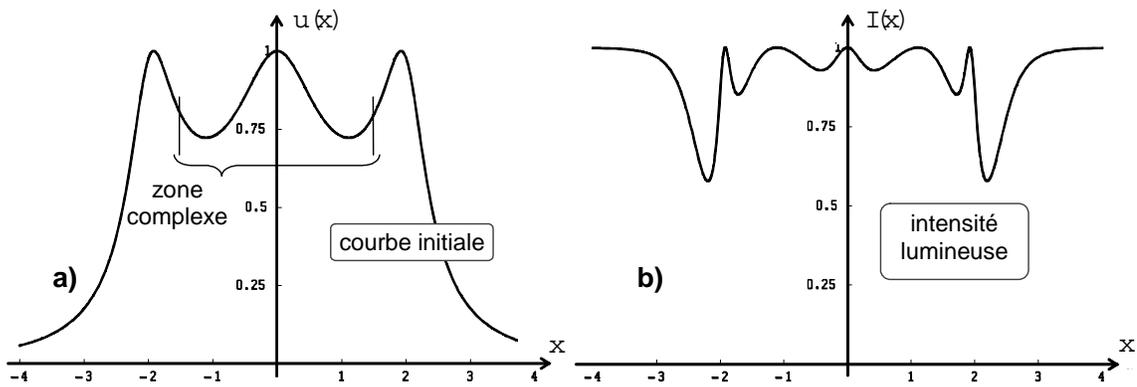


FIGURE 5.25 – Une courbe 2D analytique initiale (a) et l'évolution de sa fonction d'intensité lumineuse (b).

Un trou artificiel est réalisé dans une zone de forme complexe en supprimant l'intervalle $[-1.5, 1.5]$ de la courbe (fig. 5.26.a).

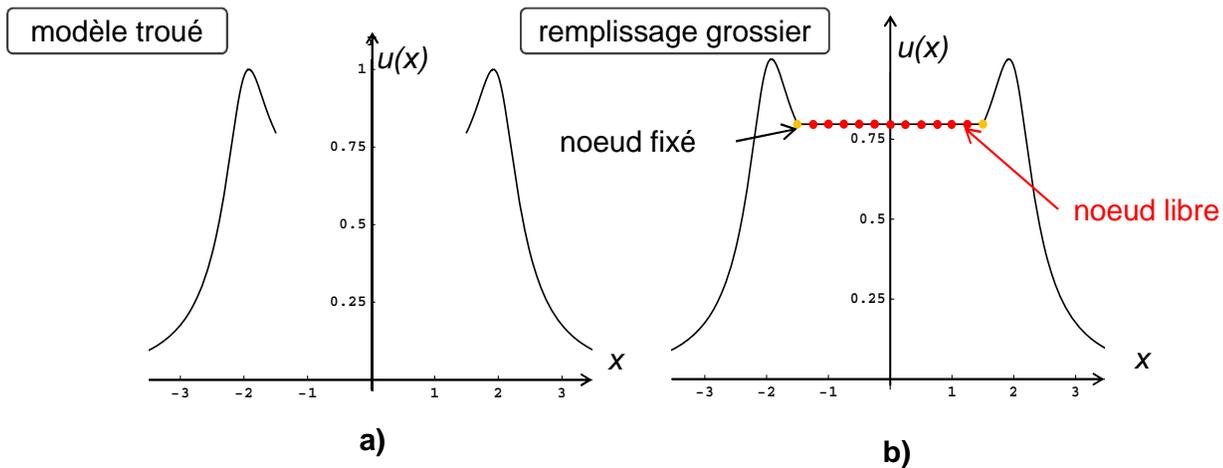


FIGURE 5.26 – La courbe initiale trouée (a) remplie avec une polyligne rectiligne (b) : les noeuds libres sont les 11 noeuds internes et les 2 noeuds du contour des extrémités sont fixés.

Le trou est ensuite rempli avec une polyligne droite définie par onze sommets internes (fig. 5.26.b). Afin de reconstruire la zone manquante, trois tests ont été réalisés :

1. trente contraintes SFS sont imposées sur les onze sommets internes de la polyligne et sur les deux sommets extrémités. Puisqu'il y a une infinité de solutions, nous avons imposé la position de certains points et avons choisi arbitrairement des solutions possibles. Ici, trois solutions possibles sont montrées (fig. 5.27.a₁ à 5.27.a₃). Ces trois solutions satisfont le même jeu de contraintes SFS. Il n'y a aucune garantie d'obtenir une solution qui reconstruise bien la courbe initiale et le processus de résolution n'est pas simple.
2. La variation relative des forces extérieures appliquées au réseau de barres couplé à la polyligne est minimisée. Cela tend à produire une forme dont la variation de courbure est minimale. Cela produit un raccord lisse entre la courbe et la polyligne insérée, mais cela ne suffit pas à reconstruire la zone de forme complexe (fig. 5.27.b).
3. Les trente contraintes SFS et la minimisation de la variation relative des forces extérieures sont utilisées pour obtenir une solution (fig. 5.27.c) dont la forme est très proche de la forme de la courbe initiale (fig. 5.26.a). Ainsi, en utilisant l'approche proposée qui combine les contraintes SFS et une minimisation qui est basée sur la variation de courbures au rac-

cord, les inconvénients de chacune des méthodes prises séparément sont surpassés.

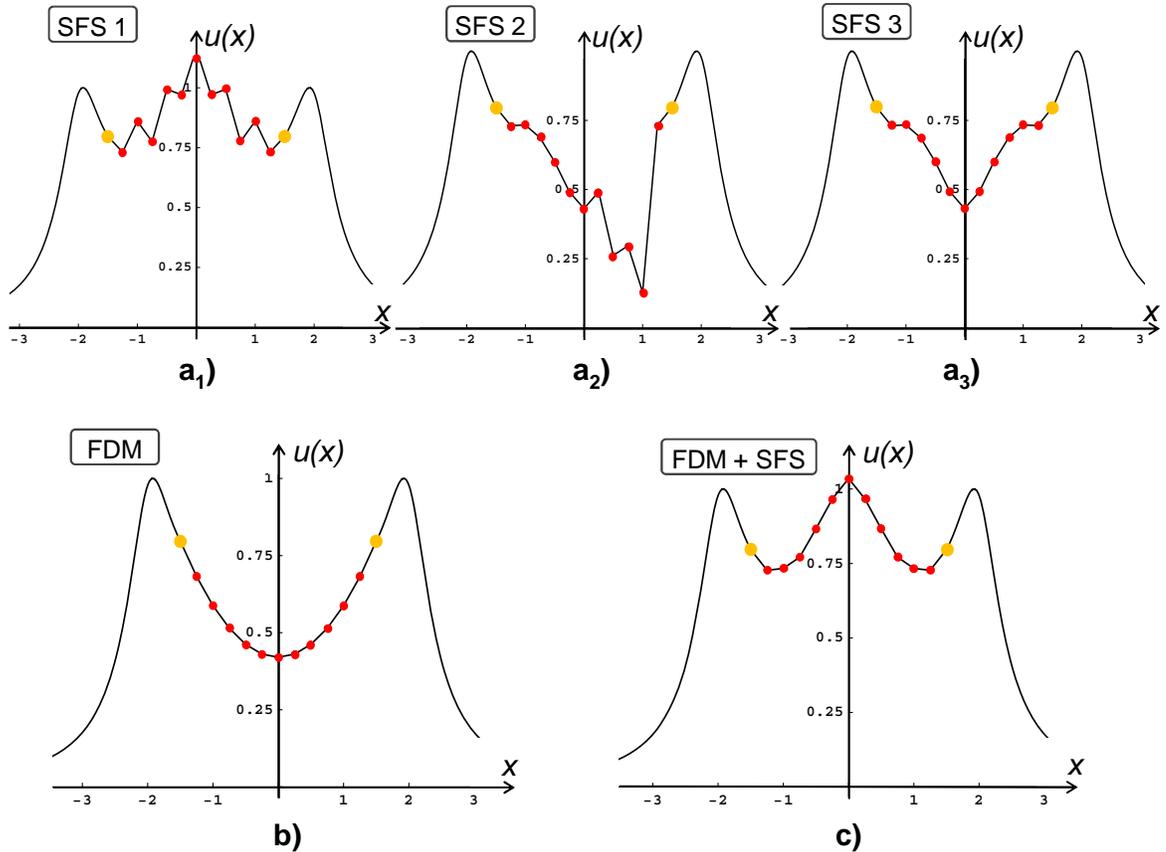


FIGURE 5.27 – La polygône est déformée afin de satisfaire soit les contraintes de SFS (a_i), soit la minimisation de la variation relative des forces extérieures (b), soit les deux à la fois (c).

Afin de comparer la qualité et la précision de la reconstruction de la courbe 2D, nous utilisons comme métrique la racine carrée des moyennes des déviations² entre la position des noeuds libres v_i de la courbe initiale et leur position w_i après déformation sur la courbe reconstruite telle que :

$$DRMS = \sqrt{\frac{1}{11} \sum_{i=1}^{11} (v_{ix} - w_{ix})^2 + (v_{iy} - w_{iy})^2} \quad (5.19)$$

2. Root Mean Square Deviation

Les remplissages obtenus en minimisant la variation de courbure seule (fig. 5.27.b) est comparé à la courbe initiale (fig. 5.25.a) ou la variation de courbure et les contraintes SFS (fig. 5.25.c) sont comparés à la courbe initiale (fig. 5.25.a).

Comme décrit dans le tableau 5.2, le résultat obtenu prenant en compte les contraintes de SFS est bien plus proche de la forme initiale que celui obtenu par simple minimisation de la variation de courbure.

Comparatif sur la courbe 2D $u(x)$	Variation de courbure <i>vs.</i> modèle initial	Variation de courbure + contraintes SFS <i>vs.</i> modèle initial
Déviations RMS (DRMS)	0.389	0.012

TABLE 5.2 – Comparaison quantitative des résultats obtenus en minimisant soit uniquement la variation de courbure seule soit la variation de courbure et la satisfaction des contraintes de SFS.

Remplissage de trou dans des polyèdres 3D

Le deuxième exemple utilise un modèle polyédrique d’un édifice architectural (5.28.a). Quarante six trous simples ont été détectés et ont pu être traités par l’algorithme de remplissage de trous de [Lou *et al.* 2008] et la minimisation de la variation de courbure (5.28.b).

Un trou est considéré comme simple lorsque le nombre d’arêtes du contour est inférieur à une dizaine d’arêtes et que leur taille est négligeable en regard des dimensions du modèle.

Afin de tester la méthode de remplissage de trous à l’aide d’informations issues du SFS, un trou a été créé (fig. 5.30.b) à l’intérieur d’une zone de forme plus complexe (fig. 5.28.b et 5.30.a).

Le contour du trou étant identifié et nettoyé, le trou est tout d’abord rempli (fig. 5.30.a) suivant le critère d’équilatéralité de [Lou *et al.* 2008]. Les tailles des triangles insérés sont homogènes par rapport à celles des triangles du maillage initial et n’ont pas de directions privilégiées. Ainsi, le premier critère de qualité est satisfait (voir section 5.2).

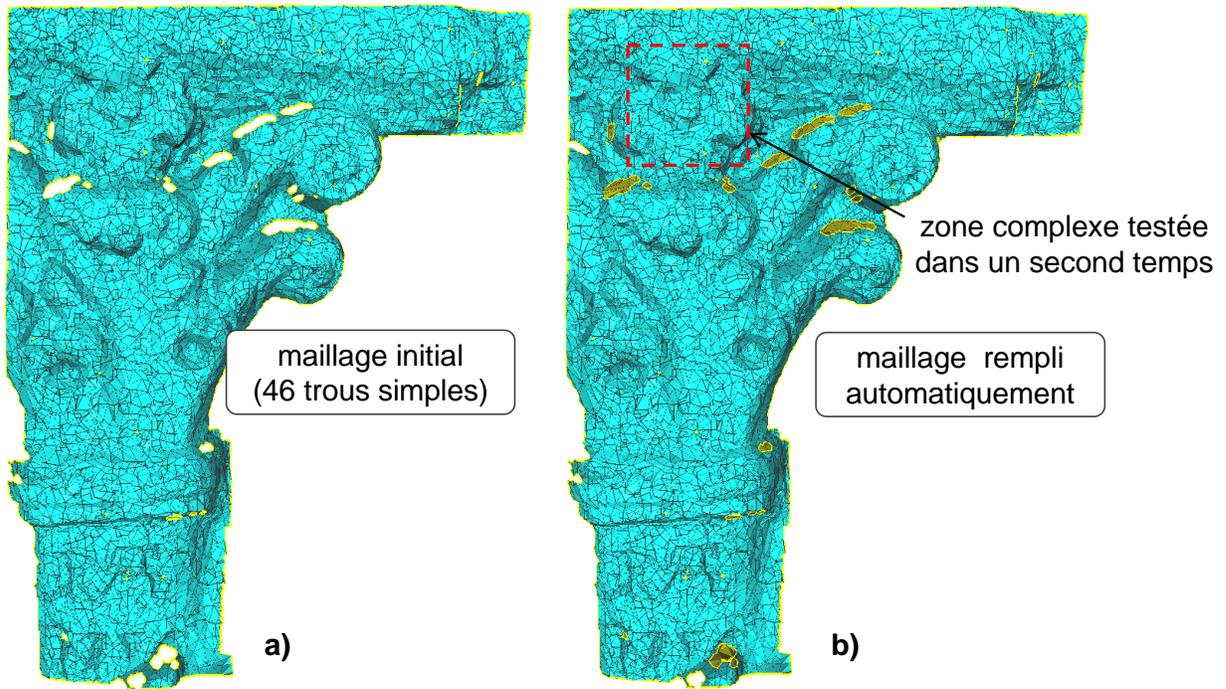


FIGURE 5.28 – Modèles polyédriques d'un édifice architectural avant (a) et après (b) le remplissage de quarante six trous utilisant simplement la minimisation de la variation des forces extérieures.

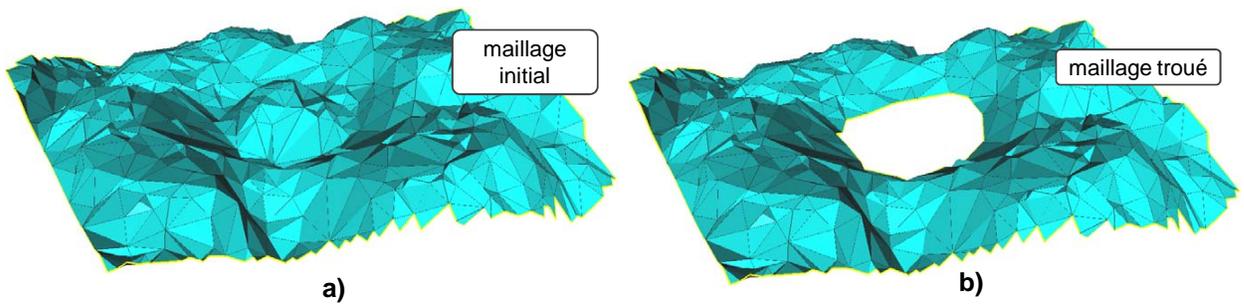


FIGURE 5.29 – Trou artificiel (b) créé dans une zone complexe.

Un réseau de barres est couplé aux sommets et arêtes du maillage à déformer (section 3.3.2). Tous les noeuds sont bloqués sauf les 44 noeuds insérés. Les coordonnées de ces derniers sont les inconnues du problème de déformation.

La figure 5.30.b montre le résultat de la minimisation de la variation relative des forces extérieures. Le maillage inséré tend à lisser le raccord

avec le maillage initial, ce qui satisfait le second critère de qualité introduit en section 5.2. La forme complexe n'est toutefois pas reconstruite.

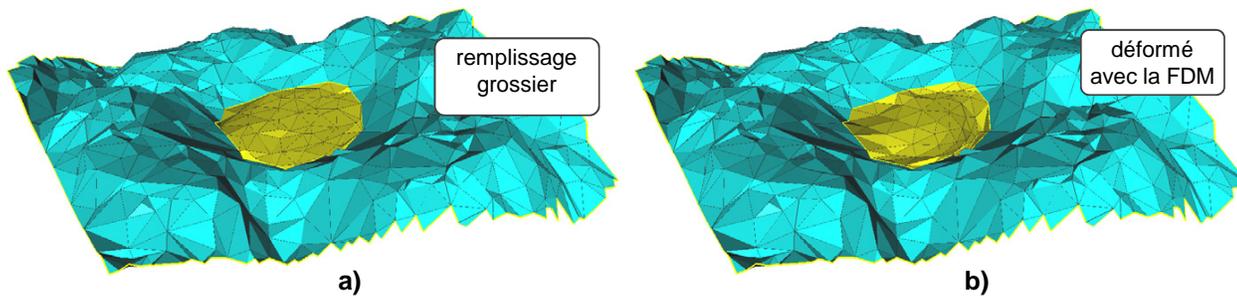


FIGURE 5.30 – Remplissage de trou dans une zone complexe utilisant la méthode de densité de force (FDM) qui tend à minimiser l'évolution de la courbure du maillage.

L'image utilisée afin de retrouver la forme initiale est une capture d'écran du maillage initial (fig. 5.31.b). Ici, le modèle est éclairé avec une unique source de lumière située à l'infini et dont la direction connue est orthogonale à l'écran. Ainsi, le calage de l'image dans le même repère que le maillage (étape de calibration de la caméra) est automatique (fig. 5.31.a).

L'étape de rétro-projection permet l'identification des 263 pixels faisant partie du contour du trou dans l'image (section 5.4.2). Une grille ayant un pas $h = 150$ pixels est utilisée pour écrire les contraintes SFS sur le maillage inséré (fig. 5.31.c). Le pas de la grille est déterminé de manière expérimentale afin d'obtenir un nombre de contraintes suffisant et homogène avec le nombre de noeuds libres. Cela conduit à la spécification de 48 contraintes (10 contraintes non linéaires et 38 contraintes linéaires) contraignant la position (x, y, z) des 44 noeuds libres.

La résolution du problème d'optimisation, *i.e.* minimisation de la variation relative des forces extérieures sous les contraintes SFS, produit en une vingtaine d'itérations le maillage de la figure 5.31.d qui satisfait tous les critères énoncés en section 5.2.

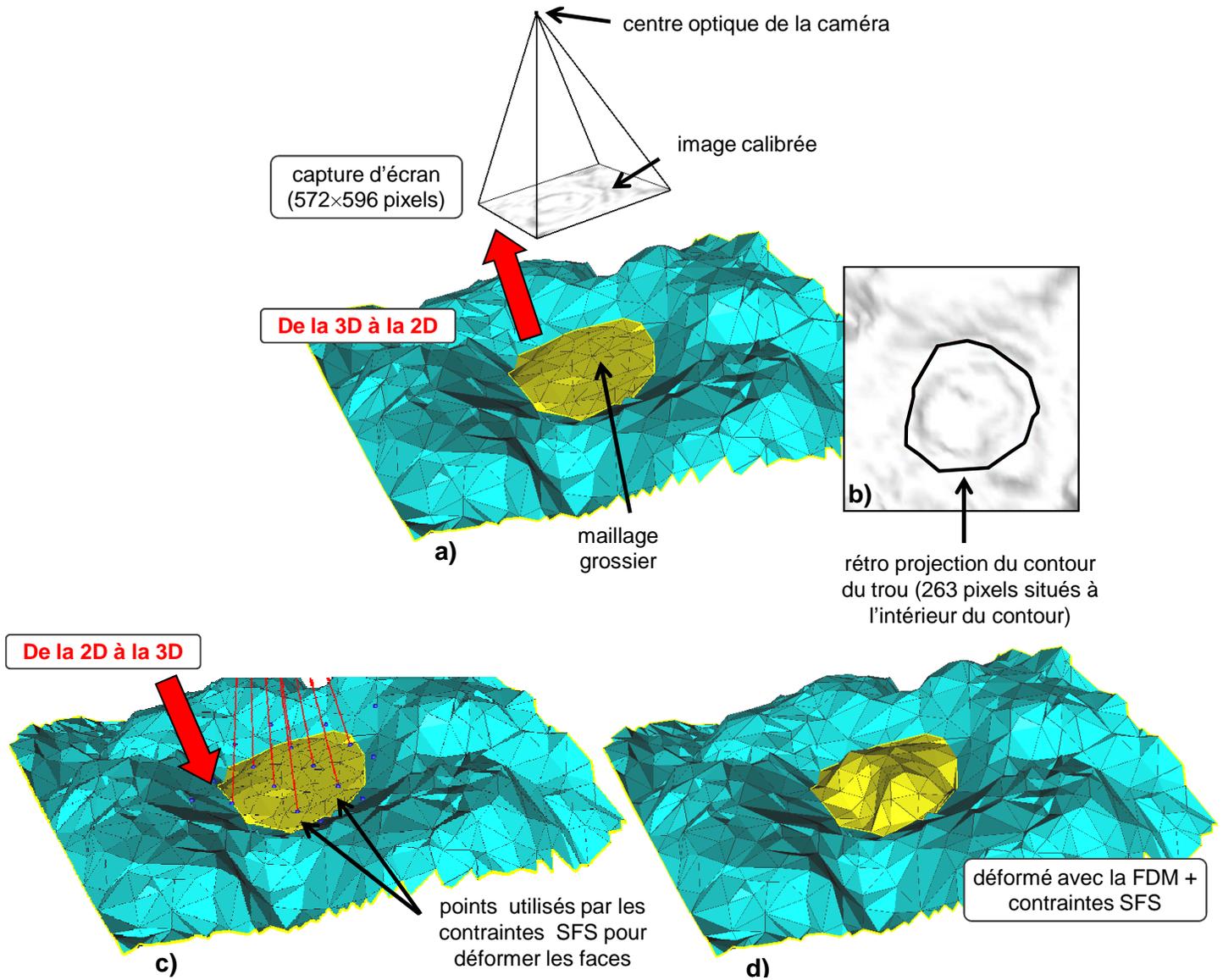


FIGURE 5.31 – Remplissage de trou mixant FDM et contraintes SFS.

Pour comparer la qualité et la précision de la reconstruction, nous calculons les distances de Hausdorff entre le modèle résultant et le modèle initial.

Premièrement, la distance de Hausdorff est mesurée entre le maillage initial (fig. 5.30.a) et le maillage obtenu en minimisant uniquement la variation de courbure (fig. 5.30.d).

Le résultat obtenu avec notre algorithme de satisfaction des contraintes SFS (fig. 5.31.d) est ensuite comparé au modèle 3D initial (fig. 5.30.a). Comme décrit par le tableau 5.3, le résultat obtenu avec notre méthode est plus proche du modèle initial que celui obtenu par simple minimisation de la variation de courbure. La figure 5.32 montre la distribution de la déviations locale entre les deux modèles résultants.

Comparatif sur le modèle de l'édifice architectural	Variation de la courbure vs. modèle initial	Variation de la courbure + contraintes SFS vs. modèle initial
distance de Hausdorff	3.045	1.883

TABLE 5.3 – Comparaison quantitative entre les résultats obtenus soit en minimisant seulement la variation de courbure soit en minimisant la variation de courbure et en satisfaisant les contraintes SFS sur le modèle de l'édifice architectural.

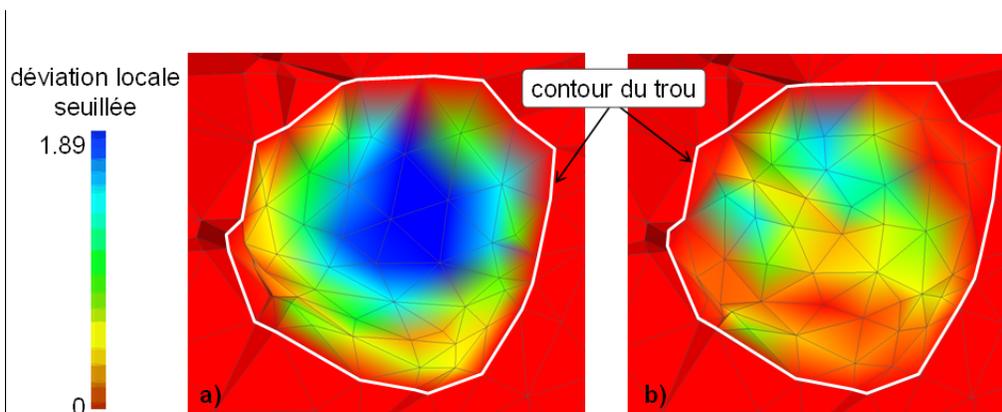


FIGURE 5.32 – Comparaison de la déviation locale entre : le modèle initial et le résultat de la minimisation de la variation de courbure (a), ou le modèle initial et le résultat de la minimisation de la variation de courbure avec la satisfaction des contraintes SFS (b).

L'algorithme a été testé sur un CPU Intel Core 2 duo, à 3 GHz et avec 8 GB RAM. Dans notre implémentation actuelle, nous utilisons Mathematica 6 pour résoudre le problème d'optimisation avec la méthode de l'Evolution Différentielle [Storn *et al.* 1997].

Puisque le langage de programmation du logiciel Mathematica est un langage interprété au moment de l'exécution, le processus de résolution est lent et affecte significativement les temps d'exécution. Sur l'exemple de l'édifice architectural, pour une erreur relative sur la position des sommets de

10^{-2} , la méthode de résolution converge au bout de 20 itérations prenant au total 49s. Dans le futur, nous prévoyons d'intégrer des modules optimisés directement implémentés en C/C++ dans notre logiciel prototype.

De bons résultats sont obtenus au regard des divers critères considérés. Toutefois, plusieurs améliorations sont envisagées. Parmi elles, nous prévoyons :

- l'utilisation de **contraintes SFS perspectives** [Tankus *et al.* 2003] dans le cas de déformations plus importantes. En effet, les contraintes SFS spécifiées contraignent les sommets insérés à se déplacer le long d'une direction orthogonale à l'axe z de l'image calibrée. L'utilisation de contraintes SFS perspectives, c'est-à-dire, des contraintes forçant le déplacement des noeuds libres le long d'axes passant tous par le centre optique mais aussi par les pixels de la grille 2D utilisée pourrait être intéressante.
- la **relaxation des contraintes**. Dans la version actuellement implémentée, les coordonnées barycentriques des sommets projetés sur le maillage restent constantes durant le processus de déformation. Cela peut entraîner des comportements indésirables particulièrement dans le cas de fortes déformations qui requièrent un grand nombre d'itérations. Puisque le processus est itératif (contraintes non linéaires), nous envisageons de relaxer les coordonnées barycentriques en reprojettant les pixels sur le maillage obtenu à chaque itération.
- la capacité de prendre en compte des **contraintes provenant de plusieurs images numériques** : cela devrait aider à reconstruire des formes plus complexes telles des formes concaves de type "champignons" puisque l'architecture proposée repose sur un problème d'optimisation dont les inconnues sont les coordonnées des noeuds libres et dont les contraintes lient directement ces inconnues aux valeurs de l'intensité lumineuse dans chaque image.

5.7 Conclusion

Dans ce chapitre, deux processus complets ont été proposés et permettent le remplissage de trous dans les polyèdres provenant d'une digitalisation. La force de ces approches réside dans l'utilisation simultanée et réciproque de la géométrie 3D et d'informations extraites d'images 2D.

Le premier processus modulaire comporte des phases d'identification des lignes de caractère 2D par filtrage des contours, de construction de lignes de caractère 3D par stéréoscopie ainsi que des phases de détection et de

nettoyage des trous pour finir sur de la déformation de grille topologique construite à partir des noeuds du bord du trou. La déformation proprement dite est le résultat de la résolution d'un problème d'optimisation numérique où les contraintes permettent la spécification partielle de la forme intérieure. Ces contraintes sont obtenues par discrétisation et projection sur le maillage inséré, à partir des lignes de caractère 3D.

Dans la deuxième méthode, la solution est obtenue en résolvant le problème d'optimisation formé d'un système d'équations non linéaires (provenant des techniques du SFS) et d'une fonctionnelle à minimiser (afin de préserver l'évolution de courbure entre le maillage inséré et le maillage initial). Ainsi, les limites des techniques du SFS et de la déformation de maillages sont surpassées : le maillage résultant satisfait le raccord en courbure et donc l'ambiguïté concave/convexe est levée.

Par rapport aux critères énoncés en section 5.2, dans les deux méthodes, le maillage inséré imite l'évolution de la forme complexe qui est extraite d'une image 2D. Le maillage résultant se raccorde en courbure avec le maillage initial. Il ne présente pas de directions privilégiées et le nombre de triangles le composant ne correspond pas forcément au nombre de pixels présents dans l'image (possibilité de faire varier le pas de la sélection des pixels utilisés pour reconstruire les lignes de caractère par stéréoscopie, et le pas de la grille des pixels projetés pour la méthode issue du SFS).

Cependant, des améliorations peuvent encore être apportées. Au niveau de l'étape de projection des points des lignes de caractère 3D sur le maillage inséré. En fait, la projection peut s'avérer inefficace lors de la reconstruction de formes complexes ou lorsque les projections de deux lignes de caractère sont voisines. Dans ce cas, il y a risque d'avoir des sur-contraintes locales altérant la qualité du résultat (fig. 5.33).

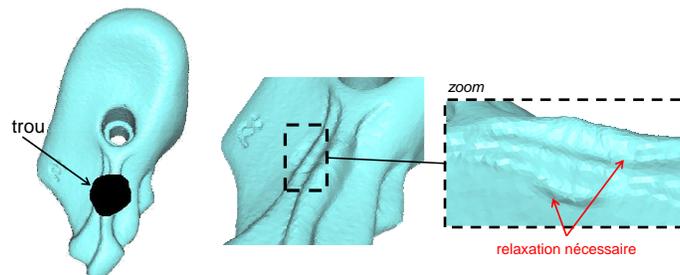


FIGURE 5.33 – Zones du modèle polyédrique surcontraintes durant le processus de déformation.

Pour pallier ces problèmes, une méthode de relaxation de contraintes

pourra être envisagée et permettre le déplacement des points projetés durant le processus de déformation. Il est également envisagé de remplacer le remplissage grossier par une approche basée sur la segmentation naturelle de l'image par les lignes de caractère. La figure 5.34 montre comment on pourrait construire un remplissage grossier à partir de la segmentation d'une partie de l'image.

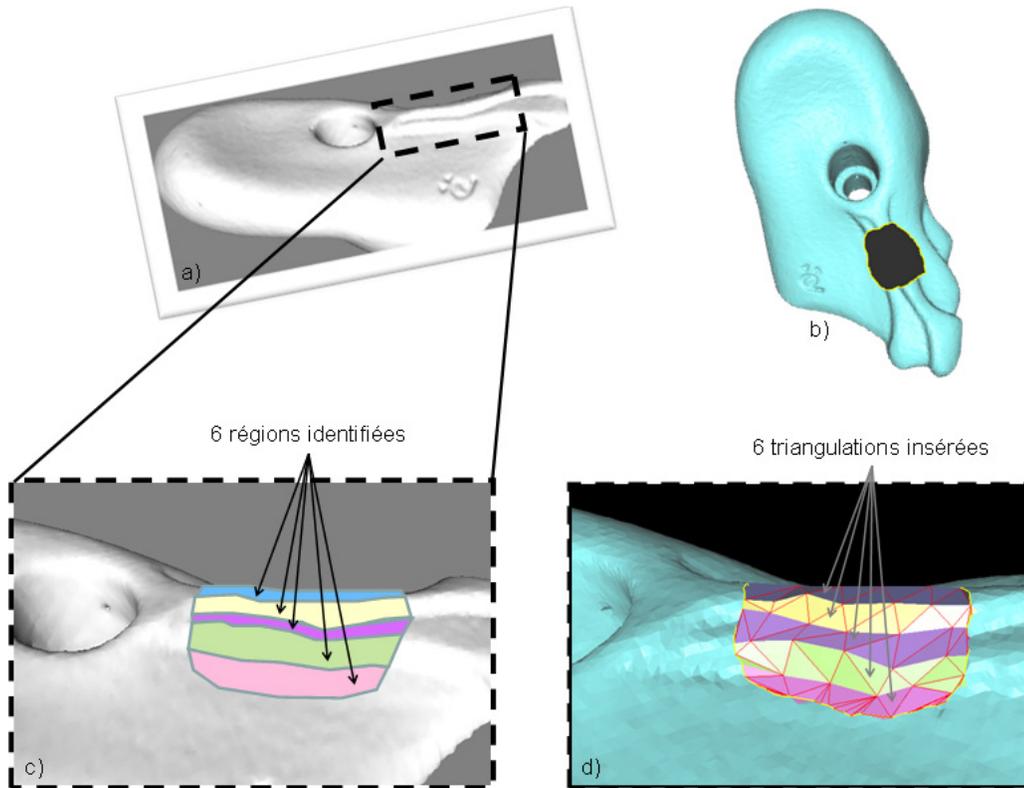


FIGURE 5.34 – Une image 2D (a) d'un modèle 3D incomplet (b) est segmentée (c) pour pouvoir remplir le trou avec autant de triangulations grossières que de régions identifiées dans l'image (d)

De même que pour la méthode de simplification proposée au chapitre 4, les deux méthodes de remplissage de trous qui exploitent des informations extraites d'images sont complètement modulaires, l'adaptation de nouvelles méthodes ou l'utilisation d'autres outils logiciels est parfaitement possible.

Enfin, un des défis les plus importants de l'approche proposée reste l'utilisation d'images réelles, et donc bruitées, dont on ne possède aucune information concernant la source de lumière.

En effet, l'approche est prometteuse pour compléter la définition du mo-

dèle digitalisé à l'aide de photographies prises du dessus de l'édifice (fig. 5.2.b₁ and 5.2.b₂) avec un appareil photographique embarqué sur un hélicoptère radio-commandé (fig. 5.2.c₁ et 5.2.c₂). Plus généralement, au cours de l'acquisition de grands édifices, l'utilisation d'un scanner laser et d'une caméra prenant des photographies aériennes peut être réellement utile et peu onéreuse par rapport au déplacement du laser vers ces zones hors de portée depuis le sol. Ceci est d'autant plus vrai pour le cas de sessions de scan très volumineuses en terme de données collectées.

Une application intéressante serait également de pouvoir réaliser des actions de type "copier/coller" d'une forme d'un maillage sur un autre maillage, ou bien, d'une forme contenue dans une image sur un maillage.

Synthèse et Perspectives

Synthèse

Les travaux de recherche présentés dans ce document ont permis de spécifier et valider des modèles et des méthodes d'exploitation d'informations contenues dans les images numériques pour le traitement de modèles polyédriques de produits.

Dans le premier chapitre, un aperçu de la variété des modèles utilisés en conception mécanique a permis de révéler le rôle prépondérant que jouent les modèles polyédriques en conception. Les images numériques souvent utilisées comme source complémentaire d'informations, peuvent également être un support pour la description des formes des objets qu'elles capturent.

Le deuxième chapitre a permis de dresser un état de l'art des techniques usuelles de traitement d'images numériques, visant à extraire des informations de type contours, formes, etc. Les méthodes de reconstruction 3D à partir d'images ont également été étudiées. Elles sont particulièrement intéressantes car permettant de passer des informations d'intensité lumineuse des pixels des images à une information tridimensionnelle de l'objet représenté.

Le troisième chapitre rassemble un état de l'art des principales méthodes de manipulation de modèles polyédriques que sont la simplification et la déformation. En effet, la complexité des formes des produits à concevoir devenant de plus en plus complexes, les modèles polyédriques associés deviennent de plus en plus volumineux et nécessitent une étape de simplification. La déformation de modèles polyédriques est également un enjeu important tant pour des fins de conception esthétique que pour la simulation de comportements dynamiques. En particulier, la présence de trous dans les modèles polyédriques acquis par numérisation laser nécessite l'insertion d'une triangulation qui peut être déformée afin d'épouser les formes de l'objet physique qui sont manquantes sur le modèle polyédrique.

Deux techniques spécifiques, une méthode de simplification et une méthode de déformation ont été choisies en raison de leur capacité à pouvoir être piloter par des critères non pas liés à la géométrie du modèle mais extraits des images.

Le quatrième chapitre introduit la nouvelle méthode de simplification à partir d'images. Elle est basée sur un algorithme itératif de suppression de sommets et est pilotée par un critère d'erreur calculé en fonction d'une approximation de la distance géodésique aux *lignes de caractère*.

Le cinquième chapitre est consacré aux deux techniques de déformation proposées et appliquées au cas de remplissage de trous. Dans les deux cas, le processus consiste à insérer une triangulation grossière puis à la déformer pour qu'elle respecte des contraintes de formes. Dans la première technique, la triangulation est déformée afin que ses sommets respectent la position 3D des lignes de caractère obtenues par triangulation stéréoscopique. Dans la seconde, c'est directement à partir de l'intensité lumineuse des pixels d'une seule image que la triangulation est déformée pour satisfaire des contraintes spécifiques découlant du problème général du Shape From Shading.

Perspectives

Plusieurs perspectives ont émergé des différentes méthodes et outils développés.

Améliorations du module de simplification de modèles polyédriques à partir d'image(s)

Deux perspectives découlant directement de la méthode de simplification de polyèdres à partir d'informations extraites d'images sont envisagées :

- la première amélioration concerne la génération automatique de la carte de tolérances. Dans l'implémentation actuelle, la carte est obtenue à partir de la projection sur le modèle polyédrique 3D, de contours 2D filtrés et appartenant à une image 2D calibrée. Il est en fait imaginable d'affecter aux zones du modèle 3D correspondant aux *lignes de caractère 3D* de l'objet des sphères d'erreur dont la localisation serait obtenues par triangulation stéréoscopique. On s'affranchirait ainsi, de l'étape de projection (lancer de rayons) qui peut ralentir considérablement l'algorithme.
- la seconde amélioration consisterait à affecter des tolérances de simplification directement calculées à partir de l'intensité lumineuse. En

effet, on pourrait envisager d'imposer des tolérances très faibles aux sommets correspondant aux zones de fort gradient d'intensité lumineuse dans l'image par exemple.

- enfin, le calcul de la carte de tolérances de simplification à partir de la reconnaissance de formes identifiées dans les images est à considérer. En effet, le partitionnement du modèle polyédrique en régions identifiées dans les images, pourrait permettre d'imposer une simplification plus importante pour les zones détectées comme faisant partie d'une région *plane* alors, que des zones *cylindriques* ou *sphériques* seraient moins simplifiées.

Améliorations du module de déformation de modèles polyédriques à partir d'images

Concernant le module de déformation de polyèdres sous contraintes issues d'images numériques, trois améliorations sont envisagées :

- la **relaxation des contraintes** de la méthode utilisant la stéréoscopie. En effet, certaines zones sont trop contraintes ce qui produit un comportement de déformation non conforme à ces endroits.
- le **calcul itératif des contraintes** issues du SFS : les contraintes imposées aux sommets de la triangulation insérée pour remplir le trou sont dans l'implémentation actuelle supposées constantes durant le processus de déformation. Il est envisagé de reprojeter les sommets obtenus à l'itération précédente dans l'image, pour recalculer les contraintes de SFS à chaque itération.
- enfin, la **forme de la triangulation** insérée pourrait dépendre de la reconnaissance de formes dans les images. Ainsi, une bibliothèque de primitives de différentes formes serait disponible et le patch 3D choisi correspondrait à la forme 2D détectée dans l'image. Le maillage 3D serait alors "cousu" au bord du trou par remplissage d'un trou topologiquement égal à un anneau.
- l'utilisation de **plusieurs images** pourrait compléter la définition des contraintes de déformation. Elle aurait pour conséquence l'ajout d'un jeu de contraintes de déformation additionnel. La gestion des problèmes de surcontraintes notamment, sera nécessaire si trop d'images sont utilisées.

Applications aux images réelles

Dans les deux types de méthodes (simplification et déformation), l'application à l'utilisation de photographies réelles, et donc bruitées, et surtout ne respectant pas totalement les formalismes idéaux décrits dans le chapitre 5 en particulier, reste une des perspectives dont l'enjeu est le plus important.

Mise à jour de modèles polyédriques

Une perspective prometteuse des travaux présentés ici consiste à appliquer et à adapter le module de déformation de polyèdres à la mise-à-jour de maquettes numériques par rapport à la géométrie réelle de structures existantes.

En effet, durant leur exploitation, les installations réelles subissent souvent des modifications qui ne sont pas répercutées sur les modèles CAO originaux. Il s'agirait d'utiliser les informations numériques contenues dans des photographies, prises directement *in situ* afin d'aller modifier localement les modèles polyédriques correspondants.

Des travaux dans ce domaine sont actuellement en cours avec le département d'études et de recherche et Développement "Analyses Mécaniques et Acoustique" (AMA) d'EDF.

Bibliographie

- [Aim@Shape 2004-08] *Advanced and Innovative Models And Tools for the development of Semantic-based systems for Handling, Acquiring, and Processing knowledge Embedded in multidimensional digital objects*, European Network of Excellence, Key Action : 2.3.1.7 Semantic-based knowledge systems, VI Framework, URL : <http://www.aimatshape.net>. 115
- [Algorri *et al.* 1996] M. E. Algorri and F. Schmitt, *Mesh Simplification*, In proceedings of Eurographics'96, vol. **15**, pp. 77–86. 64, 68
- [Angelidis 2001] A. Angelidis and M-P. Cani, *Surfaces implicites multi-résolutions*, mémoire de DEA, INPG, juin 2001. 13
- [Badler *et al.* 1997] N.-I. Badler, A.-S. Glassner, *3D object modeling*, In SIGGRAPH 97 Introduction to Computer Graphics Course Notes. ACM SIGGRAPH, August 1997. 29
- [Bakshi *et al.* 1994] S. Bakshi and Y.-H. Yang, *Shape from Shading for Non-Lambertian Surfaces*, ICIP (2), vol. **8**(2), pp. 130–134, 1994. 56
- [Baltzakis *et al.* 2002] H. Baltzakis and A. Trahanias, *Fusion of range and visual data for the extraction of scene structure information*, ICPR2002, vol. **IV**, pp. 7–11, 2002. 87
- [Barry 1991] J. Barry, *Construction of three-dimensional Delaunay triangulations using local transformations*, Computer Aided Geometric Design, vol. **8**(2), pp. 123–142, 1991. 98
- [Belhumeur *et al.* 1997] P. N. Belhumeur, D. J. Kriegman, and A. L. Yuille, *The Bas-Relief Ambiguity*, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1060–1066, 1997. 53
- [Bézier 1987] P. Bézier, *Courbes et surfaces*, Mathématiques et CAO, vol. **4**, Edition Hermès, 1987. 13
- [Blake *et al.* 1998] A. Blake and M. Isard, *Active Contours : The Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion*, Inc., Springer-Verlag New York, 1998. 35

- [Bloomenthal *et al.* 1997] J. Bloomenthal and B. Wyvill, *Introduction to Implicit Surfaces*, Morgan Kaufmann Publishers Inc., 1997. 12
- [Bloor *et al.* 1990] M. Bloor and M. Wilson, *Using partial differential equations to generate free-form surfaces*, Computer Aided Design, vol. **22**(4), pp. 202–212, 1990. 77
- [Bommes *et al.* 2007] D. Bommes, L. Kobbelt, *Accurate Computation of Geodesic Distance Fields for Polygonal Curves on Triangle Meshes*, VMV'07, pp. 151–160, 2007. 104
- [Borrel *et al.* 1991] P. Borrel and A. Bechmann, *Deformation of n -dimensional objets*, International Journal of Computational Geometry & Applications, vol. **1**(4), pp. 427–453, 1991. 74
- [Borrel *et al.* 1994] P. Borrel and A. Rappoport, *Simple constrained deformation for geometric modeling and interactive design*, ACM Transactions on Graphics, vol. **13**(2), pp. 137–155, 1994. 75
- [Boudet 1980] R. Boudet, *Présentation des mécanismes*, Techniques de l'ingénieur. Génie mécanique B 600, 600-600, Techniques de l'ingénieur, 1980. 26
- [Bryant 1979] J. Bryant, *On the clustering of multidimensional pictorial data*, In Pattern Recognition, vol. **11**, pp. 115–125, 1979. 38
- [Cani 2007] M-P. Cani, E. Galin, *Surfaces implicites*, Informatique Graphique, modélisation géométrique et animation, vol. **5**, ed. Hermès-Lavoisier, pp. 163–181, 2007. 12
- [Canny 1986] J. Canny, *A Computational Approach to Edge Detection*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. **8**, pp. 679–698, 1986. 33
- [Catalano *et al.* 2002] C.E. Catalano, B. Falcidieno, F. Giannini, M. Monti, *A survey of computer-aided modeling tools for aesthetic design*, Journal of Computing and Information Science in Engineering, **2**(1), pp. 11–20, 2002. 5, 63
- [Catmull *et al.* 1978] E. Catmull, and J. Clark, *Recursively generated B-spline surfaces on arbitrary topological meshes*, Computer-Aided Design, vol. **10**, pp. 350–355, 1978. 13
- [Chai *et al.* 1992] I. Chai, D. Dori, *Orthogonal Zig-Zag : An Efficient Method for Extracting Lines From Engineering Drawings*, Plenum Press, pp. 127–136, 1992. 40, 43, 96, 143
- [Chen *et al.* 2006] Y-S. Chen, C-H. Teng and W-H. Hsu, *Constructing a 3D trunk model from two images*, Graphical Models, vol. **69**(1), pp. 33–56, 2006. 50

- [Cheng *et al.* 2000] H. Cheng and Y. Sun, *A hierarchical approach to color image segmentation using homogeneity*, IEEE Transactions on Image Processing, vol. **9**(12), pp. 2071–2082, 2000. [38](#)
- [Ciarlet 1978] P. Ciarlet, *The finite element method for elliptic problem*, Classics in Applied Mathematics, 1978. [104](#), [126](#), [128](#)
- [Cignoni *et al.* 1998] P. Cignoni, C. Rocchini, and R. Scopigno, *Metro : measuring error on simplified surfaces*, Computer Graphics Forum, vol. **17**(2), pp. 167–174, 1998. [104](#), [144](#)
- [Clarenz *et al.* 2004] U. Clarenz and U. Diewald and G. Dziuk and M. Rumpf and R. Rusu, *A finite element method for surface restoration with smooth boundary conditions*, JCAD, vol. **21**(5), pp. 427–445, 2004. [122](#)
- [Coquillart 1990] S. Coquillart, *Extended free-form deformation : a sculpturing tool for 3D geometric modelling*, Computer Graphics, pp. 187–196, 1990. [76](#)
- [Criminisi *et al.* 2000] A. Criminisi and A. Zisserman, *Shape from texture : homogeneity revisited*, In Proc. 11th British Machine Vision Conference, pp. 82–91, 2000. [52](#)
- [Curless *et al.* 1996] B. Curless and M. Levoy, *A volumetric method for building complex models from range images*, Processing of SIGGRAPH'96, pp. 303–312, 1996. [121](#)
- [Daniel 2000] P. Daniel, *Peut-on reconstruire le relief d'une seule image ?*, mémoire de thèse, janvier 2000, France. [55](#)
- [Davis *et al.* 2002] J. Davis, S. R. Marschner, M. Garr, M. Levoy, *Filling holes in complex surfaces using volumetric diffusion*, Processing of the first International Symposium on 3D Data, Visualization and Transmission, 2002. [121](#), [122](#)
- [Debevec 1996] P. E. Debevec, *Modeling and rendering architecture from photographs*, Ph.D. Thesis, Univ. of California at Berkeley, 1996. [44](#)
- [DeHaemer *et al.* 1991] Jr. DeHaemer and M. Zyda, *Simplification of objects rendered by Polygonal Approximations*, Computers and Graphics, vol. **15**(2), pp. 175–184, 1991. [64](#), [68](#)
- [De Luca 2006] L. De Luca, *Relevé et multi-représentations du patrimoine architectural : définition d'une approche hybride de reconstruction 3D d'édifices*, mémoire de thèse, Mars 2006, France. [18](#), [19](#), [23](#), [24](#), [26](#), [27](#), [121](#)
- [Deriche *et al.* 1993] R. Deriche and G. Giraudon, *A Computational Approach for Corner and Vertex Detection*, International Journal of Computer Vision, vol. **10**, pp. 101–124, 1993. [33](#)

- [Desbrun *et al.* 1999] M. Desbrun and M. Meyer and P. Schroder and A.H. Barr, *Implicit fairing of irregular meshes using diffusion and curvature flow*, SIGGRAPH '99 : Proceedings of the 26th annual conference on Computer graphics and interactive techniques, pp. 317–324, 1999. 123
- [Deveau 2006] M. Deveau, *Utilisation conjointe d'images optiques et d'images de distance pour la modélisation de scènes terrestres*, Thèse de doctorat de l'Université Paris V - René Descartes, 2006. 87
- [Dias *et al.* 2003] P. Dias, V. Sequeira, F. Vaz, and J.G.M. Goncalves, *Combining intensity and range images for 3D modelling*, ICIP03, vol. I, pp. 417–420, 2003. 87
- [Dietrich *et al.* 2004] A. Dietrich, I. Wald and P. Slusallek, *Interactive Visualization of Exceptionally Complex Industrial CAD Datasets*, ACM SIGGRAPH'04, Sketches and Applications, 2004. 60, 61
- [Durou *et al.* 2000] J-D. Durou and D. Piau, *Ambiguous Shape from Shading with Critical Points*, JMIV, vol. 12(2), pp. 99–108, 2000. 55
- [Durou *et al.* 2004] J-D. Durou, M. Falcone and M. Sagona, *A Survey of Numerical Methods for Shape from Shading*, Technical Report, IRIT, 2004. 53
- [Eck *et al.* 1995] M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbury and W. Stuetzle, *Multiresolution Analysis of Arbitrary Meshes*, In Proceeding of SIGGRAPH'95, 1995. 64, 68
- [Falcone *et al.* 1997] M. Falcone and M. Sagona, *An algorithm for the global solution of the Shape-from-Shading model*, ICIAP 1997 (vol I), volume 1310 de LNCS, pp. 596–603, Florence, Italie, 1997. 54
- [Faugeras *et al.* 1986] O. Faugeras , G. Toscani, *The calibration problem for stereo*, in Proceeding of IEEE Int. Conf. on Computer Vision and Pattern Recognition, pp. 15-20, 1986. 114
- [Faugeras *et al.* 1987] O. Faugeras , G. Toscani, *Camera Calibration for 3D Computer Vision*, Int. Workshop on Machine Vision and Machine Intelligence, pp. 240–247, 1987. 47, 97, 129
- [Faugeras *et al.* 1992] O. Faugeras , Q. Luong, S. Maybank, *Camera self calibration : theory and experiments*, In Proceedings of the 2nd European Conference on Computer Vision, Sandini G. editor, Santa Margherita Ligure, Springer-Verlag Italy, pp. 321-334, 1992. 48, 114
- [Fontana *et al.* 2000] M. Fontana, F. Giannini, F. Meirana, *Free Form Features for Aesthetic Design*, International Journal of Shape Modelling, vol. 6(2), pp. 273–302, 2000. 93
- [Feng *et al.* 1996] J. Feng, L. Ma. and Q. Peng, *A new free-form deformation through the control of parametric surfaces*, Computer & Graphics, vol. 20(4), pp. 531–539, 1996. 75

- [Fine 2001] L. Fine, *Processus et méthodes d'adaptation et d'idéalisation de modèles dédiés à l'analyse de structures mécaniques*, Ph. D. thesis, INP-Grenoble, 2001. 107
- [FIORES II] *FIORES II : Character Preservation and Modelling in Aesthetic and Engineering Design*, project funded by the European Commission working in the field of Computer Aided Aesthetic Design (CAAD), Growth-Project, GRD1-1999-10385, URL : <http://www.fiores.com/FIORES2/FIORES2.html>. 3, 63, 93
- [Fromherz et al. 1994] T. Fromherz and M. Bichsel, *Shape from Contours as Initial Step in Shape From Multiple Cues*, Proceedings of the ISPRS Commission III Symposium on Spatial Information from Digital Photogrammetry and Computer Vision, pp. 249–256, 1994. 52
- [Garland et al. 1992] M. Garland, P. Heckbert, *Surface Simplification Using Quadric Error Metrics*, In proceedings of SIGGRAPH'92, 1992. 64, 65, 70, 72, 73, 110
- [Garland 1999] M. Garland, *QSlim software*, <http://graphics.cs.uiuc.edu/garland/software/qslim.html> 110
- [Goldman 1990] R. Goldman, *Intersection of Two Lines in Three-Space*, Graphics Gems, Academic Press Professional, Inc., USA, p. 304, 1990. 50, 129
- [Gouraud 1971] H. Gouraud, *Continuous shading of curved surfaces*, IEEE Transactions on Computers, vol. 20(6), pp. 623–628, 1971. 105
- [Guillet 1999] S. Guillet, *Modification et Construction de formes gauches soumises à des contraintes de conception*, Ph. D. thesis, INP-Grenoble, 1999. 82
- [Haines 1994] E. Haines, *Point in polygon strategies*, Graphics Gems IV, ed. Paul Heckbert, Academic Press, pp. 24–46, 1994. 133
- [Ha Lee et al. 2005] C. Ha Lee, A. Varshney, D. Jacobs, *Mesh Saliency*, ACM Trans. Graph., vol. 24(3), pp. 659–666, 2005. 66
- [Heckbert et al. 1997] P. Heckbert and M. Garland, *Survey of polygonal surface simplification algorithms*, SIG-GRAPH Course Notes : Multiresolution Surface Modeling, 1997. 63, 68
- [Hinker et al. 1993] P. Hinker and C. Hansen, *Geometric optimization*, In proceedings of Visualization'93, pp. 189–195, 1993. 64, 68
- [Hirota et al. 1999] G. Hirota, R. Maheshwari and M.-C. Lin, *Fast volume-preserving free form deformation using multi-level optimization*, IProc. Symposium on Solid Modeling and Applications, pp. 234–245, 1999. 77, 80

- [Ho *et al.* 2006] T.-C. Ho, Y.-C. Lin, J.-H. Chuang, C.-H. Peng and Y.-J. Cheng, *User-assisted mesh simplification*, In VRCIA'06 : Proceedings of the 2006 ACM international conference on Virtual reality continuum and its applications, pp. 59–66, 2006. 72
- [Hoppe 1996] H. Hoppe, *Progressive meshes*, In proceedings of SIGGRAPH'96, 1996. 64, 68
- [Horaud *et al.* 1995] R. Horaud and O. Monga, *Géométrie et calibration*, Vision par ordinateur, Outils fondamentaux, Hermès, chapter 5, 1995. 132
- [Horn *et al.* 1989] B.K.P. Horn and M.J. Brooks, *Shape from Shading*, MIT Press, 1989. 53, 54
- [Horowitz *et al.* 1974] S.L. Horowitz and T. Pavlidis, *Picture segmentation by a directed split-and-merge procedure*, in Proceedings of the Second International Joint Conference on Pattern Recognition, pp. 424–433, 1974. 38
- [Hough 1962] P.V.C. Hough, *Method and means for recognising complex patterns*, Technical report, U.S. Patent No. 3069654, 1962. 41
- [Hubeli *et al.* 2000] A. Hubeli, M. Gross, *A survey of surface representations for geometric modeling*, Technical Report 335, ETH Zurich, Institute of Scientific Computing, March 2000. 29
- [Kak *et al.* 1988] A.C. Kak and M. Slaney, *Principles of Computerized Tomographic Imaging*, IEEE Press, 1988. 49
- [Kimmel *et al.* 1995] R. Kimmel and A.M. Bruckstein, *Tracking Level Sets by Level Sets : A Method for Solving the Shape from Shading Problem*, CVIU, vol. 62(1), pp. 47–58, 1995. 54
- [Kobayashi *et al.* 2003] K.G. Kobayashi and K. Ootsubo, *t-FFD : Free-Form Deformation by using Triangular Mesh*, ACM Symposium on Solid Modelling and Applications (SM'03), Seattle, USA, pp. 226–234, 2003. 76
- [Kobbelt *et al.* 1998] L. Kobbelt and S. Campagna and J. Vorsatz and H-P. Seidel, *Interactive Multi-Resolution Modeling on Arbitrary Meshes*, Proceedings of SIGGRAPH'98, pp. 105–114, 1998. 122
- [Kumar *et al.* 1995] S. Kumar, S. Krishnan, D. Manocha and A. Narkhede, *Representation and Fast Display of Complex CSG Models*, Technical Report : TR95-019, University of North Carolina at Chapel Hill Chapel Hill, NC, USA, 1995. 14, 29
- [Lanquetin *et al.* 2005] S. Lanquetin, R. Raffin, and M. Neveu, *Constrained free form deformation on subdivision surfaces*, Advances in Computational Methods in Sciences and Engineering, ISSN 1573-4196, pp. 311–314, 2005. 77, 79

- [Lazarus *et al.* 1994] F. Lazarus, S. Coquillart and P. Jancène, *Axial deformation : an intuitive technique*, Computer Aided Design, vol. **26**, pp. 607–613, 1994. [75](#), [76](#)
- [Léon 1991] J.C. Léon, *Modélisation et construction de surfaces pour la CFAO*, Editions Hermès, 1991. [13](#), [17](#), [29](#)
- [Lesage *et al.* 2001] D. Lesage, J-C. Léon, P. Véron, *Discrete curvature approximations for the segmentation of polyhedral surfaces*, In Proceedings of the International ASME Design Automation Conference, Pittsburg, USA, 2001. [66](#)
- [Le Troter *et al.* 2005] A. Le Troter and R. Bulot and J-M. Boï and J. Sequeira, *Arc of ellipse detection for video image registration*, in : Proceedings of Signal Processing Systems Design and Implementation, pp. 365–367, 2005. [43](#)
- [Liepa 2003] P. Liepa, *Filling Holes in Meshes*, Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry Processing (SGP'03), pp. 200–205, 2003. [122](#), [123](#), [128](#)
- [Lindstrom *et al.* 2000] P. Lindstrom, G. Turk, *Image-Driven Simplification*, In ACM Transactions on Graphics, vol. **19**(3), pp. 204–241, 2000. [70](#), [71](#)
- [Lions *et al.* 1993] P.-L. Lions, E. Rouy et A. Tourin, *Shape-from-Shading, viscosity solutions and edges*, Numerische Mathematik, vol **64**(3), pp. 323–353, 1993. [54](#)
- [Lipman *et al.* 2007] Y. Lipman, D. Cohen-Or, R. Gal and D. Levin, *Volume and shape preservation via moving frame manipulation*, In ACM Trans. Graph., vol. **26**(1), 2007. [76](#)
- [Loop 1987] C. Loop, *Smooth subdivision surfaces based on triangles*, Master's thesis, 1987. [13](#)
- [Lorenson *et al.* 1987] W. Lorenson, H. Cline, *Marching cubes : a high resolution 3D surface construction algorithm*, in Proceedings of the SIGGRAPH'87, pp. 163–169, 1987. [122](#)
- [Lou *et al.* 2008] R. Lou, A. Mikchevitch, J.-P. Pernot and P. Véron, *Direct merging of triangle meshes preserving simulation semantics for fast modification of numerical models*, Proceeding of the Tools and Methods for Competitive Engineering (TMCE'08), vol. **1**, pp. 119–131, 2008. [128](#), [142](#), [149](#)
- [Luebke 2001] D. Luebke, *A developer's survey of polygonal simplification algorithms*, IEEE Computer Graphics and Applications, 2001. [63](#)
- [Ma *et al.* 1997] L. Ma, R.W.H. Lau, J. Feng, Q. Peng and J. Wong, *Surface deformation using the sensor glove*, ACM VRST'97, Lausanne, Switzerland, pp. 189–196, 1997. [75](#)

- [MacCracken *et al.* 1996] R. MacCracken and K.I. Joy, *Free-form deformation with lattices of arbitrary topology*, Computer Graphics, pp. 181–188, 1996. [76](#)
- [Marr *et al.* 1980] D. Marr and E. Hildreth, *Theory of edge detection*, Proceedings of the Royal Society of London, Series B, vol. **207**, pp. 187–217, 1980. [33](#)
- [Mata *et al.* 2007] S. Mata, L. Pastor, J.-J. Aliaga, A. Rodríguez, *Incorporating visual attention into mesh simplification techniques*, APGV '07 : Proceedings of the 4th symposium on applied perception in graphics and visualization, pp. 134–134, 2007. [72](#), [73](#)
- [Michel-Duthel 2001] G. Michel-Duthel, *Images numériques et formats graphiques*, collection : Studio Graphique, éd. CampusPress, 2001. [25](#)
- [Monga *et al.* 1995] O. Monga, R. Horaud., *Vision stéréoscopique*, Hermès edition, 1995. [37](#), [38](#), [43](#), [45](#), [50](#)
- [Murphy *et al.* 2001] C. A. Murphy and T. D. Perera, *Role of simulation in industries : the definition and potential role of simulation within an aerospace company*, in Proceedings of the 33nd conference on Winter simulation, WSC'01, Arlington, Virginia, 2001. [1](#)
- [Neveu *et al.* 1995] M. Neveu, D. Faudot, B. Derdouri, *Superquadriques B-déformables pour la reconstruction 3D*, Numéro Spécial. RAIRO. Techniques et Sciences Informatiques, vol **14**(10), pp. 1291–1314, 1995. [75](#)
- [Nooruddin *et al.* 2003] F. S. Nooruddin, G. Turk, *Simplification and Repair of Polygonal Models Using Volumetric Techniques*, IEEE Transactions on Visualization and Computer Graphics, vol **9**(2), pp. 191–205, 2003. [122](#)
- [Olhander *et al.* 1978] R. Olhander, K. Price and D.R. Reddy, *Picture segmentation using a recursive region splitting method*, In Computer Graphics and Image Processing, vol **8**, pp. 313–333, 1978. [37](#)
- [O'Rourke 1998] J. O'Rourke, *Computational Geometry in C*, 2nd Edition, Cambridge Tracts in Theoretical Computer Science, 1998. [49](#), [97](#), [98](#)
- [Panchetti *et al.* 2007] M. Panchetti, J-P. Pernot and P. Véron, *Polyhedral simplifications preserving character lines extracted from images*, Proceedings of the IEEE Shape Modelling International Conference (SMI'07), pp. 81–90, 2007. [132](#)
- [Panchetti *et al.* 2007a] M. Panchetti, J-P. Pernot and P. Véron, *Remplissage de trous dans les maillages à l'aide de lignes de caractère extraites d'images*, in : 10ème colloque national AIP-PRIMECA, La Plagne, France, ISBN 978-2-86272-451-5, 2007. [129](#)
- [Panchetti *et al.* 2008] M. Panchetti, J-P. Pernot and P. Véron, *Towards recovery of complex shapes in meshes using shaded images*, in : Tools and

- Methods for Competitive Engineering (TMCE'08), vol. 1, pp. 187–198, ISBN 978-90-5155-044-3, 2008. 131
- [Panchetti *et al.* 2009] M. Panchetti, J-P. Pernot and P. Véron, *Towards recovery of complex shapes in meshes using digital images for reverse engineering applications*, submitted to the Journal of Computer-Aided, Special Issue on Shape design and manipulation for improved product modeling, 2009. 131
- [Pernot 2004] J.P. Pernot, *Fully Free Form Deformation Features for Aesthetic and Engineering Designs*, mémoire de thèse, en cotutelle entre l'INP Grenoble en France et Università di Genova en Italie, 2004. 5, 6, 29, 63, 74, 79, 80, 81, 82, 83, 84, 93, 130, 138
- [Pernot *et al.* 2005] J.P. Pernot, S. Guillet, J-C. Léon, B. Falcidieno and F. Giannini, *Shape Tuning in Fully Free Form Deformation Features*, Journal of Computing and Information Science in Engineering (JCISE), vol. 5(1), pp. 95–103, 2005. 79, 81
- [Pernot *et al.* 2006] J-P. Pernot, G. Moraru and P. Véron, *Filling holes in meshes for efficient Reverse Engineering of products*, Proceedings of the TMCE 2006, 2006. 62, 85, 125, 139, 140
- [Pollefeys *et al.* 2004] M. Pollefeys, L. Van Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, R. Koch, *Visual modeling with a hand-held camera*, International Journal of Computer Vision, vol. 59(3), pp. 207–232, 2004. 4
- [Prados *et al.* 2002] E. Prados, O. Faugeras, and E. Rouy, *Shape from Shading and Viscosity Solutions*, in Proceedings of the 7th European Conference on Computer Vision (volume II), volume 2351 of Lecture Notes in Computer Science, pp. 790–804, 2002. 53, 54
- [Prados *et al.* 2005] E. Prados and O. Faugeras, *Shape From Shading*, In N. Paragios, Y. Chen, and O. Faugeras, editors, Handbook of Mathematical Models in Computer Vision, chapter 23, Springer-Verlag, pp. 377–390, 2005. 53
- [Prewitt 1970] J.M. Prewitt, *Object enhancement and extraction*, Picture Processing and Psychopictorics, B.S. Lipkin and A. Rosenfeld, Eds, Academic Press, 1970. 33
- [Qin *et al.* 1996] H. Qin and D. Terzopoulos, *D-NURBS : A Physics-Based Framework for Geometric Design*, IEEE Transactions on Visualization and Computer Graphics, vol. 2(1), pp. 85–96, 1996. 77
- [Raffin *et al.* 1998] R. Raffin, M. Neveu and B. Derdouri, *Constrained deformation for geometric modelling and object reconstruction*, WSCG'98, vol. 2, pp. 299–306, 1998. 75

- [Revelles *et al.* 2000] J. Revelles, C. Urena and M. Lastra, *An Efficient Parametric Algorithm for Octree Traversal*, In the 8th International Conference in Central Europe on Computer Graphics, Visualization and Interactive Media, 2000. 114
- [Rossignac *et al.* 1992] J.R. Rossignac, P. Borrel, *Multi-resolution 3D approximations for rendering complex scenes*, In Falcidieno, B. and Kunii, T. L., editors, *Geometric Modeling in Computer Graphics*, Genova, Italy, Springer-Verlag, pp. 455–465, 1992. 64, 68
- [Roushdy 2007] M. Roushdy, *Comparative Study of Edge Detection Algorithms Applying on the Grayscale Noisy Image Using Morphological Filter*, ICGST International Journal on Graphics, Vision and Image Processing, vol. 6(4), pp. 17–23, 2007. 33
- [Rouy *et al.* 1992] E. Rouy and A. Tourin, *A Viscosity Solutions Approach to Shape-From-Shading*, *SIAM Journal on Numerical Analysis*, vol. 29(3), pp. 867–884, 1992. 53
- [Sabin 2004] M. Sabin, *Recent progress in subdivision : a survey*, in *Advances in Multiresolution for Geometric Modelling*, Dogdson N., Floater M., Sabin M., (Eds.). Springer-Verlag, pp. 203–230, 2004. 13
- [Sarraga 2004] R.F. Sarraga, *Modifying CAD/CAM surfaces according to displacements prescribed at a finite set of points*, *Computer Aided Design*, vol. 36, pp. 343–349, 2004. 77
- [Schek 1974] H.J. Schek, *The force density method for form finding and computation of general networks*, *Comp. Meth. in App. Mech. and Eng.*, vol. 3(2), pp. 115–134, 1974. 78, 84
- [Schneider *et al.* 2001] R. Schneider and L. Kobbelt, *Geometric fairing of irregular meshes for free-form surface design*, *Journal of Computer-Aided Geometric Design*, vol. 18(4), pp. 359–379, 2001. 123
- [Schroeder *et al.* 1992] W. J. Schroeder, J.A. Zarge, W.E. Lorensen, *Decimation of triangle meshes*, In proceedings of SIGGRAPH'92, pp. 65–70., 1992. 64, 68
- [Schwartz 2008] P. Schwartz, *Méthode des contours actifs*, <http://khayyam.developpez.com/articles/algo/contours-actifs/>, 2008. 36
- [Sederberg *et al.* 1986] T.W. Sederberg and S.R. Parry, *Free-form deformation solid geometric models*, *Computer Graphics*, vol. 20(4), pp. 18–22, 1986. 76, 77
- [Shah *et al.* 1995] J. Shah, M. Mantyla, *Parametric and feature-based CAD/CAM*, Wiley-Interscience Publication, John Wiley & Sons, Inc. (1995) 63

- [Shi *et al.* 2007] X. Shi, K. Zhou, Y. Tong, M. Desbrun, H. Bao and B. Guo, *Mesh puppetry : cascading optimization of mesh deformation with inverse kinematics*, ACM Trans. Graph., vol. **26**(3), pp. 81, 2007. [76](#), [78](#)
- [Sklansky *et al.* 1980] J. Sklansky, V. Gonzalez, *Fast Polygonal Approximation of Digitized Curves*, Pattern Recognition, vol. **12**, pp. 327–331, 1980. [41](#)
- [Sobel 1990] I. Sobel, *Chapter "An Isotropic 3D Images Gradient Operator"*, Machine Vision of Three-Dimensional Scenes, Academic Press, pp. 376–379, 1990. [33](#), [96](#), [104](#), [129](#), [143](#)
- [Sorkine *et al.* 2004] O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C. Rössl, H.-P. Seidel, *Laplacian surface editing*, SGP'04 : Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing, pp. 175–184, 2004. [76](#)
- [Stam *et al.* 2003] J. Stam and C. T. Loop, *Quad/Triangle Subdivision*, Comput. Graph. Forum, vol. **22**, pp. 79–86, 2003. [13](#), [14](#)
- [Storn *et al.* 1997] R. Storn and K. Price, *Differential Evolution : A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces*, Journal of Global Optimization, vol. **11**, pp. 341–359, 1997. [153](#)
- [Souffez *et al.* 2006] J.-M. Souffez., *Partitionnement et simplification de maillages pour l'analyse temps-réel de maquettes numériques*, Ph. D. Thesis, Université de Rennes 1, 2006. [114](#)
- [Tankus *et al.* 2003] A. Tankus, N. Sochen and Y. Yeshhurun, *A New Perspective [on] Shape-from-Shading*, ICCV'03 : Proceedings of the Ninth IEEE International Conference on Computer Vision, USA, pp. 862, 2003. [154](#)
- [Tankus *et al.* 2004] A. Tankus, N. Sochen and Y. Yeshhurun, *Perspective Shape-from-Shading by Fast Marching*, CVPR 2004 (vol I), pp. 43–49, Washington,D.C., USA, 2004. [54](#), [56](#)
- [Taubin 1995] G. Taubin, *Estimating the tensor of curvature of a surface from a polyhedral approximation*, In Proceedings of IEEE International Conference on Computer Vision, pp. 902–907, 1995. [66](#)
- [Tekumalla *et al.* 2004] L.S. Tekumalla and E. Cohen, *A Hole-Filling Algorithm for Triangular Meshes*, Technical report UUCS-04-019, School of Computing, University of Utah, USA, <http://www.cs.utah.edu/techreports/>, 2004. [123](#)
- [Terzopoulos *et al.* 1987] D. Terzopoulos, J. Platt, A. Barr and K. Fleischer, *Elastically deformable models*, Computer Graphics (SIGGRAPH'87), vol. **21**, pp. 205–214, 1987. [77](#)

- [Terzopoulos *et al.* 1994] D. Terzopoulos, H. Qin, *Dynamic NURBS with geometric constraints for interactive sculpting*, ACM Transactions on Graphics, vol. **13**(2), pp. 103–136, 1994. [77](#)
- [Thingvold *et al.* 1990] J.A. Thingvold and E. Cohen, *Physical Modeling with B-spline Surfaces for Interactive Design and Animation*, Computer Graphics (Interactive 3D Graphics'90), vol. **24**, pp. 129–137, 1990. [77](#)
- [Todd 2004] J. T. Todd, *The visual perception of 3D shape*, Trends in Cognitive Sciences vol. **9**(3), pp. 115–121, 2004. [22](#)
- [Torre *et al.* 1986] V. Torre and T. Poggio, *On edge detection*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. **8**, pp. 147–163, 1986. [35](#)
- [Torres *et al.* 2004] L. A. Torres-Méndez and G. Dudek, *Reconstruction of 3D models from intensity images and partial depth*, Proc. American Association for Artificial Intelligence (AAAI), pp. 476–481, 2004. [87](#)
- [Turk 1992] G. Turk, *Retiling polygonal surfaces*, In proceeding of SIGGRAPH'92, pp. 55–64, 1992. [64](#), [68](#), [73](#)
- [Turk *et al.* 2002] G. Turk, J.-F. O'brien, *Modelling with implicit surfaces that interpolate*, In ACM Trans. Graph., vol. **21**(4), pp. 855–873, 2002. [29](#)
- [Van *et al.* 1991] G. Van and E. Cek, *BRep-Index : a multidimensional space partitioning tree*, SMA '91 : Proceedings of the first ACM symposium on Solid modeling foundations and CAD/CAM applications, ISBN 0-89791-427-9, Austin, Texas, United States, pp. 35–44, 1991. [15](#)
- [Varady *et al.* 1997] T. Varady, R. Martin, J. Coxt, *Reverse engineering of geometric models - An introduction*, Computer Aided Design, vol. **29**(4), pp. 255–268, 1997. [17](#), [29](#)
- [Varshney 1994] A. Varshney, *Hierarchical Geometric Approximations*, PhD thesis, Dept. of CS, U. of North Carolina, Chapel Hill, TR-050, 1994. [68](#), [69](#)
- [Verdera *et al.* 2003] J. Verdera and V. Caselles and M. Bertalmio and G. Sapiro, *Inpainting surface holes*, International Conference on Image Processing (ICIP), 2003. [122](#)
- [Véron *et al.* 1997] P. Véron and J.C. Léon, *Static polyhedron simplification using an error measurement criterion*, Computer Aided Design, vol. **29**(4), pp. 287–298, 1997. [66](#), [67](#)
- [Véron 1997] P. Véron, *Techniques de simplification de modèles polyédriques pour un environnement de conception mécanique*, mémoire de thèse, INPG, Grenoble, 1997. [29](#), [66](#), [69](#), [70](#), [73](#), [84](#), [106](#), [107](#)
- [Véron *et al.* 1998] P. Véron and J.C. Léon, *Shape preserving polyhedral simplification with bounded error*, Computer & Graphics, vol. **22**(5), pp. 565–585, 1998. [64](#)

- [Wald *et al.* 2001] I. Wald, P. Slusallek and C. Benthin, *Interactive Distributed Ray-Tracing of Highly Complex Models*, in Rendering Techniques 2001/Proceedings of the EUROGRAPHICS Workshop on Rendering 2001, pp. 274–285, 2001. [60](#), [61](#)
- [Wang *et al.* 2003] J. Wang and M.M. Oliveira, *Hole-Filling Strategy for Reconstruction of Smooth Surfaces in Range Images*, Proceedings of SIBGRAP'03, 2003. [123](#)
- [Williams *et al.* 2003] N. Williams, D. Luebke, J.-D. Cohen, M. Kelley and B. Schubert, *Perceptually guided simplification of lit, textured meshes*, In I3D'03 : Proceedings of the 2003 symposium on Interactive 3D Graphics, pp. 113–121, 2003. [70](#)
- [Wenyin *et al.* 1999] L. Wenyin, D. Dori, *Geometric fairing of irregular meshes for free-form surface design*, Pattern Analysis and Applications, vol. **2**(2), pp. 10–21, 1999. [40](#), [41](#), [43](#)
- [Yip *et al.* 1992] R.K.K. Yip, P.K.S. Tam and D.N.K. Leung, *Modification of Hough Transform for Circles and Ellipses Detection Using a 2-Dimensional Array* PR, vol. **25**(9), pp. 1007–1022, 1992. [43](#)
- [Yoshizawa *et al.* 2005] S. Yoshizawa, A. Belyaev and H. Seidel, *Fast and Robust Detection of Crest Lines on Meshes*, In Proceeding of ACM Symposium on Solid and Physical Modelling (SPM), pp. 227–232, 2005. [66](#)
- [Yu *et al.* 2002] Y. Yu and J.-T. Chang, *Shadow graphs and surface reconstruction*, In European Conference on Computer Vision, pp. 31–45, Springer-Verlag, 2002. [52](#)
- [Zhang *et al.* 2002] E. Zhang, G. Turk, *Visibility-Guided Simplification*, In Proceedings of IEEE Visualization'02, pp. 267–274, 2002. [70](#), [72](#)
- [Zhao *et al.* 2006] H. Zhao, G. Xu, *Triangular surface mesh fairing via Gaussian curvature flow*, In J. Comput. Appl. Math., vol. **195**(1), pp. 300–311, 2006. [98](#)
- [Zhou *et al.* 2005] K. Zhou, J. Huang, J. Snyder, X. Liu, H. Bao, B. Guo and H.-Y. Shum, *Large mesh deformation using the volumetric graph Laplacian*, ACM Trans. Graph., vol. **24**(3), pp. 496–503, 2005. [76](#)
- [Zorin 2006] D. Zorin, *Modeling with multiresolution subdivision surfaces*, SIGGRAPH '06 : ACM SIGGRAPH 2006 Courses, pp. 30–50, ISBN 1-59593-364-6, New York, NY, USA, 2006. [29](#)

EXPLOITATION D'IMAGES NUMERIQUES POUR LA SIMPLIFICATION ET LA DEFORMATION DE MODELES POLYEDRIQUES

RESUME: Les modèles polyédriques, très utilisés dans les processus d'ingénierie, constituent une représentation privilégiée au sein des maquettes numériques de produits. Les approches et méthodes de manipulation et d'exploitation de ces modèles, sont le plus souvent pilotées par un critère géométrique lié à la forme des objets (normale, courbure ...) mais très peu prennent en compte des informations de type perceptuelles. Parallèlement, les techniques de traitement d'images existantes extraient des données (contours, primitives géométriques, textures ...) relatives à la description des formes des objets qu'elles représentent. Ainsi, le but de cette thèse est de réaliser un couplage modèle polyédrique 3D / images numériques 2D pour manipuler les modèles 3D avec des critères extraits d'image(s). Une méthode de simplification et deux méthodes de déformation de polyèdres sont présentées. Dans la première méthode, la simplification est basée sur un processus itératif de suppression de sommets piloté par un critère de tolérance de simplification. Cette tolérance est liée au filtrage de contours d'image(s) qui sont projetés sur le modèle 3D afin d'identifier les zones plus ou moins proches de ces *lignes de caractère*. Les méthodes de déformation de polyèdres sont appliquées au cas du remplissage de trous. Une triangulation est insérée au modèle puis déformée par la résolution d'un problème d'optimisation numérique sous contraintes. La fonctionnelle à minimiser simule la variation de courbure entre le maillage inséré et le modèle initial. La solution est obtenue par un algorithme itératif basé sur un modèle mécanique de réseau de barres. Les contraintes imposent le respect de *lignes de caractère 3D* obtenues par triangulation stéréoscopique dans la première méthode, ou bien calculées en fonction de l'intensité lumineuse des pixels et qui imposent le déplacement des sommets correspondants suivant une certaine élévation (problème inverse du Shape From Shading). Les trois méthodes implémentées sont complètement modulaires.

Mots-clés: *conception mécanique, modèles polyédriques, images numériques, simplification, déformation, remplissage de trous dans les maillages, stéréoscopie, Shape From Shading, reconstruction 3D*

TOWARDS USING IMAGES FOR POLYHEDRONS SIMPLIFICATION AND DEFORMATION

ABSTRACT: Polyhedral models which are very often used in mechanical design, constitute a privileged representation among the digital mock-ups of products. On the one hand, the approaches and methods of manipulation and exploitation of these models are usually led by one geometric criterion linked to the objects' shape (normal, curvature ...) but only a few of them pay attention to perceptual information. On the other hand, the existing image processing techniques extract data (contours, geometric features, textures...) related to the objects' shapes description they represent. Thus, the goal of this Ph.D. is to perform a coupling between polyhedral 3D models and 2D digitalized image(s) in order to handle these models with image-based criteria. One simplification and two deformation methods are presented. In the first one, the simplification is based on a vertex removal iterative process led by a tolerance criterion: it is linked to images contours filtering which are projected onto the 3D model to identify areas which are more or less close to those *character lines*. The polyhedrons deformation methods are applied to the hole-filling case. A triangulation is inserted and deformed by an optimization problem under constraints resolution. The function to minimize simulates the curvature variation between the inserted patch and the initial model. The solution is obtained by an iterative algorithm based on a mechanical model of bars network. The constraints come either from the respect of 3D character lines obtained by stereoscopic triangulation or directly from the intensity of the pixels to constrain the vertices displacement to a certain elevation (inverse Shape From Shading problem). The three developed methods are completely modular.

Keywords: *mechanical design, polyhedrons, images, simplification, deformation, filling holes, stereoscopy, Shape From Shading, 3D recovery*

