



HAL
open science

Description sémantique de modèles 3D : application à la modélisation à partir d'exemples

Vincent Léon

► To cite this version:

Vincent Léon. Description sémantique de modèles 3D : application à la modélisation à partir d'exemples. Vision par ordinateur et reconnaissance de formes [cs.CV]. Université de Lille 1, Sciences et Technologies; CRISAL UMR 9189, 2016. Français. NNT : . tel-01720085

HAL Id: tel-01720085

<https://theses.hal.science/tel-01720085>

Submitted on 28 Feb 2018

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.



UNIVERSITÉ LILLE 1

École Doctorale Science Pour

l'Ingénieur

Laboratoire CRIStAL (UMR CNRS 9189)

THÈSE

pour obtenir le grade de

DOCTEUR,

SPÉCIALITÉ INFORMATIQUE

présentée et soutenue publiquement par

Vincent Léon

le 10/11/2016

**Description sémantique de modèles 3D - Application à la
modélisation à partir d'exemples**

COMPOSITION DU JURY

Mme. Géraldine Morin	Maître de conférences HDR, Université de Toulouse	Rapporteur
M. Laurent Lucas	Professeur, Université de Reims Champagne-Ardenne	Rapporteur
M. François Goulette	Professeur, Mines ParisTech	Examineur
M. Guillaume Lavoué	Maître de Conférences HDR, Université de Lyon	Directeur
M. Jean-Philippe Vandeborre	Professeur, Institut Mines-Télécom, Lille	Directeur
M. Nicolas Bonneel	Chargé de Recherche, CNRS, Université de Lyon	Encadrant

RÉSUMÉ

Description sémantique de modèles 3D - application à la modélisation à partir d'exemples

Bien que les applications utilisant les modèles 3D soient de plus en plus nombreuses, la création des modèles 3D en elle-même reste une tâche fastidieuse. La modélisation à partir d'exemples rend cette étape plus efficace en permettant à l'utilisateur d'exploiter des modèles existants pour les assembler afin d'obtenir un nouveau modèle. Pour exploiter ces modèles, il est nécessaire de pouvoir les retrouver rapidement dans la collection, par le biais d'une recherche partielle de maillages. La géométrie de chacune des parties doit ensuite être adaptée, avant de les assembler. Les méthodes les plus récentes de l'état de l'art se concentrent sur des mécanismes d'assemblages de segments : les modèles de la base sont préalablement segmentés et labellisés. Cette considération permet d'éviter à l'utilisateur plusieurs segmentations manuelles, mais elle limite le processus de création à une suite d'ajout ou d'échange de parties pré-déterminées. Dans cette thèse, la modélisation se fait par échange de parties depuis un modèle de base, mais ces parties ne sont pas directement segmentées. Nous présentons un descripteur sémantique continu calculé sur l'ensemble du maillage 3D, qui nous permet de retrouver parmi les modèles de la base les parties de modèles correspondant à une sélection de l'utilisateur sur le modèle de référence : si l'utilisateur sélectionne le coude d'un modèle humanoïde, il se verra proposé tous les coudes des modèles de la base même si ce label n'est pas explicitement utilisé pour labelliser la base. Cette description sémantique continue et ce mécanisme de correspondance de patches nous permettent d'obtenir l'ensemble de parties nécessaire à la création du nouveau modèle. Ces différentes parties sont alors assemblées par une méthode de recalage semi-rigide et un remaillage.

Mots-clés : Modélisation 3D, polygones, modélisation à partir d'exemples, labellisation, correspondance sémantique, adaptation géométrique.

REMERCIEMENTS

Comme toute grande aventure, la thèse ne se vit pas seul, et celle-ci n'aurait pu être la même sans l'implication de nombreuses personnes que je souhaite remercier ici.

En premier lieu, je remercie mes directeurs de thèses, Guillaume Lavoué et Jean-Philippe Vandeborre qui ont su conduire les trois années de recherches ayant conduit à ce manuscrit avec professionnalisme et patience. Je remercie également Nicolas Bonneel pour les conseils qu'il m'a apporté. À tous, merci encore pour la relecture et les corrections apportées à ce manuscrit.

Mes remerciements vont également à Géraldine Morin et Laurent Lucas pour avoir accepté d'être relecteurs de cette thèse, pour l'examen du manuscrit et pour vous être libérés afin de participer à la soutenance.

Je remercie également François Goulette d'avoir accepté d'être présent en qualité d'examineur de cette thèse.

Je tiens aussi à remercier l'ensemble de l'équipe 3DSAM du laboratoire CRISAL, avec lesquels j'ai partagé mon quotidien au cours de ces trois années : Boulbaba Ben Amor, Mohamed Daoudi, Hassen Drira, et Hazem Wannous.

Un grand merci à tous mes collègues doctorants, post-docs et ingénieurs, avec qui j'ai passé une bonne partie de mon temps durant ces trois années, à l'intérieur comme à l'extérieur du *labo* : Taleb Alashkar, Quentin De Smedt, Maxime Devanne, Vincent Itier, Nadia Hosni, Anis Kacem, Meng Meng, Sébastien Poulmane, Sarah Ribet, Rim Slama et Baiqian Xia.

Enfin, je tiens à remercier mes amis et ma famille pour le soutien qu'ils m'ont offert durant ces trois années. Je remercie mes parents pour leur bienveillance. Je remercie aussi Jessica, Audrey et Laurène, ainsi que ma nièce et mes neveux, pour leur présence et pour leurs sourires.

Lille, le 11 Septembre 2016

PUBLICATIONS

Revue internationale

Vincent Léon, Nicolas Bonneel, Guillaume Lavoué et Jean-Philippe Vandeborre. Continuous semantic description of 3D meshes. *Computers & Graphics*, volume 54, pages 47-56, 2016.

Conférence nationale

Vincent Léon, Nicolas Bonneel, Guillaume Lavoué et Jean-Philippe Vandeborre. Géodésiques sémantiques pour la description et la labélisation automatique de parties. *Actes des 27èmes journées de l'Association Française d'Informatique Graphique*, 2014.

TABLE DES MATIÈRES

LISTE DES FIGURES	viii
1 INTRODUCTION	1
1.1 CONTEXTE	3
1.2 CONTRIBUTIONS	5
1.3 ORGANISATION DU MÉMOIRE	5
2 ÉTAT DE L'ART	7
2.1 REPRÉSENTATION D'OBJETS 3D	8
2.2 OUTILS D'AIDE À LA MODÉLISATION	10
2.2.1 Outils de modélisation conventionnels	12
2.2.2 Outils à base d'esquisses 2D	14
2.2.3 Outils de génération procédurale	15
2.3 MODÉLISATION À PARTIR D'EXEMPLES	16
2.3.1 Modélisation à partir d'exemples : les premiers travaux .	17
2.3.2 Exploration et variation sur un ensemble de modèles . .	19
2.3.3 Méthode d'assemblage à partir d'esquisse 2D	22
2.3.4 Méthode d'assemblage avec suggestions de segments . .	24
2.4 CONCLUSION	28
3 DESCRIPTION SÉMANTIQUE	31
3.1 INTRODUCTION	32
3.1.1 État de l'art sur les descripteurs locaux de maillage 3D . .	33
3.1.2 Définition du descripteur	36
3.1.3 Propriétés	38
3.2 LABELLISATION DE MODÈLES 3D À L'AIDE DU DESCRIPTEUR SÉMANTIQUE	44
3.2.1 La labellisation comme un problème inverse	44

3.2.2	Apprentissage	45
3.2.3	Inférence	45
3.2.4	Résultats	46
3.3	TRAITEMENTS LOCAUX SUR LE MAILLAGE 3D	50
3.4	CONCLUSION	53
4	CORRESPONDANCE ET SÉLECTION À L'AIDE DU DESCRIPTEUR SÉMANTIQUE	55
4.1	INTRODUCTION	56
4.1.1	Outils de segmentation interactive	56
4.1.2	Recherche partielle de maillages	57
4.1.3	Détection de patches similaires	59
4.2	MÉTHODE	62
4.2.1	Vue d'ensemble	62
4.2.2	Initialisation	63
4.2.3	Minimisation d'énergie	65
4.2.4	Classification sur la base	66
4.3	EXPÉRIMENTATIONS	66
4.3.1	Sélection d'une zone à l'intérieur d'un segment	67
4.3.2	Sélection d'une zone entre plusieurs segments	68
4.3.3	Sélection de zones de tailles différentes	69
4.3.4	Influence de la résolution sur la sélection	70
4.3.5	Discussion	71
4.4	CONCLUSION	73
5	ADAPTATION ET ASSEMBLAGES DE MAILLAGES 3D	75
5.1	INTRODUCTION	76
5.1.1	Motivations	76
5.1.2	Méthodes de compositions de maillages	76
5.2	MÉTHODE	78
5.2.1	Principe général	78
5.2.2	Recalage Semi-rigide	79
5.2.3	Définition des régions de jonction S_a et S_b	81
5.2.4	Mise en correspondance des régions S_a et S_b	82

5.2.5	Définition des voisinages supports	82
5.2.6	Calcul et application de la transformation	83
5.3	RÉSULTATS	84
5.4	CONCLUSION	84
6	CONCLUSION	87
6.1	RÉSUMÉ	88
6.2	LIMITATIONS ET PERSPECTIVES	89
	BIBLIOGRAPHIE	91

LISTE DES FIGURES

2.1	Différentes représentation d'un objet 3D.	11
2.2	Interface de modélisation conventionnelle	13
2.3	Interface de modélisation par esquisse	15
2.4	Synthèse de modèles à partir d'exemple	16
2.5	Illustration de la méthode de Funkhouser <i>et al.</i> [26]	18
2.6	Résultats de mélanges entre deux modèles, extrait de Jain <i>et al.</i> [32].	20
2.7	Projection en 2D d'un ensemble de modèles, extrait de Averkiou <i>et al.</i> [4]	21
2.8	Capture d'écran de l'interface de modélisation par esquisse 2D, extrait de <i>Sketch to Design</i> [87].	23
2.9	Assemblage de segments de modèles à partir d'une labellisation cohérente, extrait de Kreavoy <i>et al.</i> [43].	25
2.10	Suggestions de segments à l'aide d'un modèle probabiliste, extrait de Chaudhuri <i>et al.</i> [12].	27
3.1	Principe du descripteur sémantique	34
3.2	Illustration des catégories de modèles considérées	37
3.3	Stratégie d'échantillonnage des sommets labellisés.	37
3.4	Visualisation du descripteur sémantique	39
3.5	Invariance du descripteur sémantique vis à vis de la résolution du maillage	40
3.6	Comparaison entre la distance géodésique et notre métrique sémantique	40
3.7	Correspondance sémantique entre plusieurs points sur différents modèles	41

3.8	Comparaison du descripteur sémantique et de caractéristiques géométriques	42
3.9	Positionnement multidimensionnel des signatures de segments	43
3.10	Influence de l'échantillonnage sur la comparaison des segments	44
3.11	Résultats de labélisation	49
3.12	Influence de la segmentation sur la labélisation	49
3.13	Application au traitement de maillages	51
3.14	Application du descripteur sémantique au traitement de maillages.	52
4.1	Interfaces de segmentation interactive	57
4.2	Illustration de l'extension de patches, extrait de <i>SimSelect</i> [29]	61
4.3	Procédé de sélection et de correspondance pour deux requêtes.	64
4.4	Présentation de la labellisation utilisée pour la correspondance.	67
4.5	Sélection d'une zone à l'intérieur d'un segment et correspondance.	67
4.6	Sélection d'une zone à l'intérieur d'un segment et correspondance.	68
4.7	Sélection d'une zone entre plusieurs segments et correspondance.	69
4.8	Sélection de zones de tailles différentes	70
4.9	Sélection de zones de tailles différentes	70
4.10	Correspondance entre une version basse résolution d'un modèle, et une version de plus haute résolution.	71
4.11	Classification avec un SVM, en utilisant les caractéristiques de <i>SimSelect</i> [29] et le descripteur sémantique.	72
5.1	Création d'une jonction entre deux surfaces par ajout de géométrie, extrait des travaux de Funkhouser <i>et al.</i> [26].	77
5.2	Exemple de deux modèles à adapter	78

5.3	Résultat après le recalage global par ICP.	79
5.4	Illustration des zones de jonction.	81
5.5	Évolution du recalage semi-rigide.	83
5.6	Exemples d'assemblages de parties par soft-ICP.	85

INTRODUCTION

1

SOMMAIRE

1.1	CONTEXTE	3
1.2	CONTRIBUTIONS	5
1.3	ORGANISATION DU MÉMOIRE	5

La représentation visuelle de l'environnement est un outil de communication primordial et l'apparition de l'informatique graphique 3D a permis des modes de visualisation jusqu'alors inédits. De nos jours, les applications exploitant la visualisation 3D sont de plus en plus nombreuses. Elles touchent des domaines de plus en plus larges (conception industrielle, domaine médical, formation, divertissement, etc.) mais surtout elles se mêlent de plus en plus au réel, par l'intermédiaire de nouvelles modalités telles que l'impression 3D ou la réalité augmentée.

Les avancées en matière de rendu photo-réaliste et d'animation ont démultiplié les possibilités de représentation. Les outils de création de contenu sont eux aussi plus accessibles : on peut donner comme exemple *Blender*¹ pour la création de films d'animation, ou *Unity*² pour la création d'applications interactives.

La modélisation se trouve en amont de toutes ces applications : il s'agit de donner une représentation aux objets que l'on souhaite visualiser en 3D. Les modes de représentations d'un objet en 3D sont nombreux, et sont souvent fonction de l'application. De même, ce mode de représentation conditionne la manière dont le modèle est créé.

La création d'un modèle est le plus souvent dirigée par un utilisateur, bien que certaines méthodes permettent de générer des modèles originaux de manière automatique. Il est primordial que ces outils de modélisation 3D se perfectionnent et intègrent de nouvelles fonctionnalités, afin de rendre la tâche de création plus efficace.

La modélisation à partir d'exemples permet cette efficacité en proposant d'utiliser des parties de modèles existants, et de les intégrer dans un nouveau modèle. Ce dispositif ne se substitue pas aux processus de création classiques, mais les complète, en permettant à l'utilisateur d'exploiter des créations antérieures afin de gagner du temps. Les problématiques dans la modélisation à partir d'exemples sont multiples : l'indexation et la recherche de contenus 3D sont nécessaires afin de retrouver rapidement les créations antérieures ; un traitement géométrique sur les données permet d'intégrer des modèles de sources différentes dans un seul modèle.

1. <https://www.blender.org/>

2. <https://unity3d.com/>

Dans la suite de ce chapitre, nous présentons le contexte du projet, les différentes contributions apportées dans cette thèse, et enfin l'organisation de ce mémoire.

1.1 CONTEXTE

Le projet CrABEx (Création de contenu graphique 3D Assistée par une Base d'Exemples, référence ANR-13-CORD-0013)³ est un projet de recherche français, financé par l'Agence Nationale de la Recherche (ANR), s'intéressant à des modalités de création et d'édition de contenus pour la modélisation 3D permettant d'exploiter des contenus existants.

Le projet est porté par cinq partenaires : le LIRIS (UMR 5205), le LTCI (UMR 5141) et CRISAL (UMR 9189) d'un point de vue académique, Gamagora (Université Lumière Lyon 2) pour la formation, et 3DDUO (entreprise basée à Tourcoing) en tant que partenaire industriel.

Le projet vise à proposer un système d'aide à la création et à l'édition de contenus 3D, utilisant une base de modèles existants. Ce système ne concerne pas seulement les données géométriques, mais aussi les méta-données telles que les textures, les squelettes d'animations, etc. Cette thèse se concentre sur la recherche et l'extraction de données géométriques 3D pour la création d'un maillage, ainsi que sur leur assemblage, dans le but d'obtenir un nouveau modèle à partir de modèles existants.

Cette collaboration a permis de recueillir les avis d'utilisateurs potentiels, à travers des réunions de travail et les rencontres effectuées. Nous avons notamment mis en place en amont de ce travail de recherche un sondage, à l'aide d'un formulaire en ligne⁴ auprès des salariés de 3DDUO et des étudiants de Gamagora afin d'avoir plus de détails sur leurs habitudes en terme de modélisation, et pour recueillir leurs avis sur une application de modélisation à partir d'exemples.

Sur les 36 personnes ayant répondu au sondage, 84% affirment ne jamais ou rarement modifier un modèle existant lors d'une tâche de modélisation. Ils sont 70% à affirmer toujours ou souvent démarrer une modéli-

3. <http://www.cristal.univ-lille.fr/crabex/>

4. <http://tinyurl.com/CrABEx>

sation de zéro. La modélisation à partir d'exemples ne fait donc pas partie des usages courants pour la majorité des personnes interrogées.

Cependant, après avoir décrit brièvement la modélisation à partir d'exemples, à la question "seriez-vous intéressé par un tel système?", ils sont 86% à être intéressés. Un utilisateur en particulier ajoute : "Cela pourrait être intéressant pour les environnements, mais semble moins pratique pour les personnages, mais généralement chaque designer a sa banque personnelle avec des éléments séparés qu'il peut réutiliser (mains, pieds, têtes, etc.)".

Les autres questions portent principalement sur le mode de requête préféré pour ajouter une nouvelle partie, ou remplacer une partie existante. Ici, les utilisateurs devaient noter sur une échelle de 1 à 5 les différents modes de requêtes, parmi la recherche par mots-clés, la sélection d'une partie de modèle, la réalisation d'une esquisse 2D, ou d'une esquisse 3D. Dans les deux tâches, la recherche traditionnelle par mots-clés et la sélection d'une partie de modèle 3D comme requête sont les modalités de requête les mieux notées. De manière générale, les utilisateurs préfèrent les requêtes classiques à base de mots-clés, et semblent réticents au bruit potentiel des autres méthodes. Les méthodes d'esquisse 2D et 3D sont bien moins populaires, car elles semblent plus fastidieuses à utiliser.

Seulement 42% des sondés seraient intéressés par un système procédural entièrement automatique, ce qui confirme notre orientation vers un outil d'aide à la création. D'une manière générale, les personnes ayant répondu au sondage sont majoritairement intéressées par l'idée d'un système de modélisation assisté par une base d'exemples. Une labellisation sémantique reste importante (permettant l'utilisation de mots-clés), et un système de requête par sélection de parties de modèles doit être privilégié.

En rencontrant les designers de 3DDUO, nous avons pu confirmer que les modèles sont rarement réutilisés. Parfois cependant, certaines parties sont réutilisées, notamment pour les parties qui prennent du temps à être modélisées, telle que la tête ou les mains. Mais cet usage est propre à chaque designer.

1.2 CONTRIBUTIONS

Nous considérons la modélisation à partir d'exemples comme une tâche de recherche et d'assemblage de parties de modèles existants dans une base. Plutôt que de demander à l'utilisateur d'effectuer une segmentation manuelle ou de considérer des segments à échanger, nous proposons un outil de correspondance sur des *patches* permettant à l'utilisateur d'identifier dans les modèles de la base des parties similaires à celles qu'il aura préalablement sélectionnées sur un modèle de référence. Les contributions principales de cette thèse sont les suivantes :

- *Descripteur sémantique continu* : à partir d'une labellisation du maillage, nous proposons un descripteur multidimensionnel calculé sur chaque point du maillage, pour représenter son contexte sémantique, d'une manière plus riche qu'une labellisation classique, en croisant les informations relatives aux différents labels. Ce descripteur est simple et rapide à calculer. Nous présentons aussi plusieurs applications de ce descripteur pour la labellisation automatique de modèles 3D, et pour l'édition de maillages.
- *Correspondance et sélection de patches par similarité* : afin de limiter l'intervention de l'utilisateur, et pour ne pas se limiter à une segmentation fixe, nous présentons un système permettant de retrouver, à partir d'un patch sélectionné par l'utilisateur sur un modèle de référence, les régions sémantiquement similaires dans les modèles de la base. Ce système de sélection repose sur le descripteur sémantique précédemment évoqué. Cet outil de correspondance et de sélection de patches par similarité sémantique est particulièrement utile pour la modélisation à partir d'exemples.

1.3 ORGANISATION DU MÉMOIRE

Ce mémoire de thèse est organisé de la manière suivante : dans le chapitre 2, nous présentons plus en détail le contexte du mémoire, en partant des différentes modalités de représentations des modèles 3D pour arriver à un état de l'art des méthodes de modélisation à partir d'exemples. Le

chapitre 3 introduit le nouveau descripteur sémantique continu que nous proposons ainsi que différentes applications. Le chapitre 4 présente la méthode que nous utilisons pour retrouver des patchs similaires à une sélection sur le maillage dans une collection de modèles 3D existants par similarité sémantique. Dans le chapitre 5, nous détaillons une méthode de recalage semi-rigide et de remaillage permettant d'adapter plusieurs maillages 3D malgré leurs différences géométriques. Enfin, nous concluons par le chapitre 6 où sont rappelées les différentes contributions et où sont proposées des perspectives de recherches futures.

ÉTAT DE L'ART

2

SOMMAIRE

2.1	REPRÉSENTATION D'OBJETS 3D	8
2.2	OUTILS D'AIDE À LA MODÉLISATION	10
2.2.1	Outils de modélisation conventionnels	12
2.2.2	Outils à base d'esquisses 2D	14
2.2.3	Outils de génération procédurale	15
2.3	MODÉLISATION À PARTIR D'EXEMPLES	16
2.3.1	Modélisation à partir d'exemples : les premiers travaux	17
2.3.2	Exploration et variation sur un ensemble de modèles	19
2.3.3	Méthode d'assemblage à partir d'esquisse 2D	22
2.3.4	Méthode d'assemblage avec suggestions de segments	24
2.4	CONCLUSION	28

Afin de mieux situer le contexte de ce travail de recherche, nous présentons dans ce chapitre les différents outils pour aboutir à la représentation d'un objet en 3D. En premier lieu, nous revenons dans la section 2.1 sur les différentes modalités de représentation d'un objet 3D. Par la suite, nous présentons dans la section 2.2 un ensemble d'outils de modélisation 3D dont le but est de simplifier la conception d'objets 3D faite selon une représentation polygonale. Enfin, nous terminons cet état de l'art en nous intéressant plus spécifiquement dans la section 2.3 aux méthodes de modélisation à partir d'exemples, grâce auxquelles il est possible de créer de nouveaux modèles, en exploitant des créations existantes.

2.1 REPRÉSENTATION D'OBJETS 3D

Les méthodes de modélisation d'objets 3D sont nombreuses, et sont souvent conditionnées par la représentation mathématique qui est faite de l'objet. Avant de présenter en détails différents outils de modélisation, il convient de présenter comment un modèle 3D (volumique ou surfacique) est représenté.

Les représentations volumiques permettent de représenter l'intégralité du volume 3D. Parmi ces représentations, on trouve le modèle CSG (*Constructive Solid Geometry*) : un objet est construit à l'aide d'opérations booléennes successives (union, intersection, différence) entre d'autres volumes. Les primitives de départ sont des volumes simples (cube, sphère, cylindre, etc.). Ainsi, différentes opérations binaires sont représentées sous forme arborescente pour permettre la représentation de formes complexes. Un exemple d'arborescence est présenté sur la figure 2.1a, page 11. Cette représentation offre des avantages en matière de coût de stockage, de rendu et de calcul de collision. En revanche, les possibilités en terme de modélisation sont limitées. Bien que ce mode de représentation est utilisé pour la conception assistée par ordinateur dans des applications industrielles, il reste moins adapté pour une création plus libre.

La représentation par voxel est un autre mode de représentation 3D volumique. De manière équivalente aux pixels dans l'espace 2D, un voxel est un élément volumique caractérisé par sa position sur une grille 3D.

Il est possible de lui assigner différentes caractéristiques, telle qu'une information de couleur, de niveau de gris, etc. Ce mode de représentation est particulièrement utilisé dans le milieu médical, puisqu'il est facile de construire une telle représentation en voxels à partir d'acquisitions par couches obtenues par un scanner IRM. Ce type de représentation permet aussi de calculer facilement une coupe d'un objet afin d'en visualiser l'intérieur, comme illustré dans la figure 2.1b. En revanche, le rendu de ces modèles reste coûteux.

Un autre moyen de représenter une forme en 3D est d'utiliser une représentation surfacique. D'un point de vue mathématique, la représentation surfacique la plus naturelle est la représentation par surface implicite. Une surface est alors définie par une équation dans un espace Euclidien. Ainsi, un tore de paramètres (R, a) sera définie par l'équation suivante :

$$(x^2 + y^2 + z^2 + R^2 - a^2)^2 - 4R^2(x^2 + y^2) = 0 \quad (2.1)$$

Il est alors possible de définir à partir de plusieurs surfaces implicites une nouvelle surface, à l'aide d'opérations simples (addition, multiplication). La figure 2.1c montre une surface obtenue par l'addition de trois tores. Cette représentation est intéressante en terme de stockage, et est particulièrement adaptée pour les techniques de rendu par lancer de rayons. Néanmoins, ce mode de représentation contraint les possibilités en terme de modélisation, ce qui fait qu'il n'est pas adapté pour la représentation d'objets réels.

La représentation utilisant les B-splines rationnelles non-uniformes (ou NURBS) permet une modélisation plus simple et plus souple à l'aide d'un réseau de points de contrôles. La surface est évaluée en fonction de la position des points de contrôle et du poids qui est associé à chacun d'eux. Les NURBS sont utilisées en infographie 3D pour la modélisation de surfaces lisses complexes. La manipulation des points de contrôle permet de définir facilement ces surfaces, et de les représenter de manière compacte. La figure 2.1d illustre une surface ainsi que les points de contrôle utilisés.

Enfin, le mode de représentation le plus couramment utilisé est la modélisation à partir de polygones. Cette représentation discrétise une sur-

face en un ensemble de sommets, reliés par des arêtes. La précision de la représentation dépend de la résolution. Ainsi, pour représenter une surface plus lisse, on augmentera le nombre de sommets. Il est possible de modifier l'apparence d'une surface en changeant directement la position des sommets. De nombreux algorithmes de traitement de maillages ont été développés pour changer l'aspect d'une surface (lissage, bruitage, subdivision, etc.). Ce mode de représentation permet une grande liberté en terme de modélisation et est adapté aux méthodes de rendu temps réel. Elle permet par ailleurs de modéliser tout type de surface, notamment les surfaces avec bord. Cette méthode de représentation étant la plus utilisée, et offrant une grande liberté en terme de modélisation, nous nous concentrerons sur cette dernière dans la suite. À noter qu'il est possible de passer d'un autre mode de représentation à une représentation par polygones, en utilisant par exemple une méthode de triangulation. C'est notamment le cas du modèleur 3D Z-Brush¹, qui utilise une représentation à base de *pixol* 2.5D, similaires aux voxels, pour permettre à l'utilisateur de créer un modèle 3D en utilisant des outils se rapprochant de la sculpture.

La figure 2.1 résume les différentes approches pour la représentation de forme en 3D.

Le reste de ce chapitre est organisé de la manière suivante : dans la section 2.2, nous présentons des outils d'aide à la modélisation, allant des modèleurs conventionnels tels que les modèleurs commerciaux aux méthodes rendant la modélisation plus accessible. Puis, dans la section 2.3, nous étudierons les méthodes de modélisation à partir d'exemples, qui permettent de créer de nouveaux modèles à partir d'un ensemble de maillages existants.

2.2 OUTILS D'AIDE À LA MODÉLISATION

Dans la section précédente, nous avons vu que les méthodes de représentation de modèles 3D sont nombreuses. Néanmoins, la représentation surfacique par maillage est simple à manipuler et permet de modéliser tout type de surface. Naturellement, la majorité des travaux en informa-

1. <https://pixologic.com/>

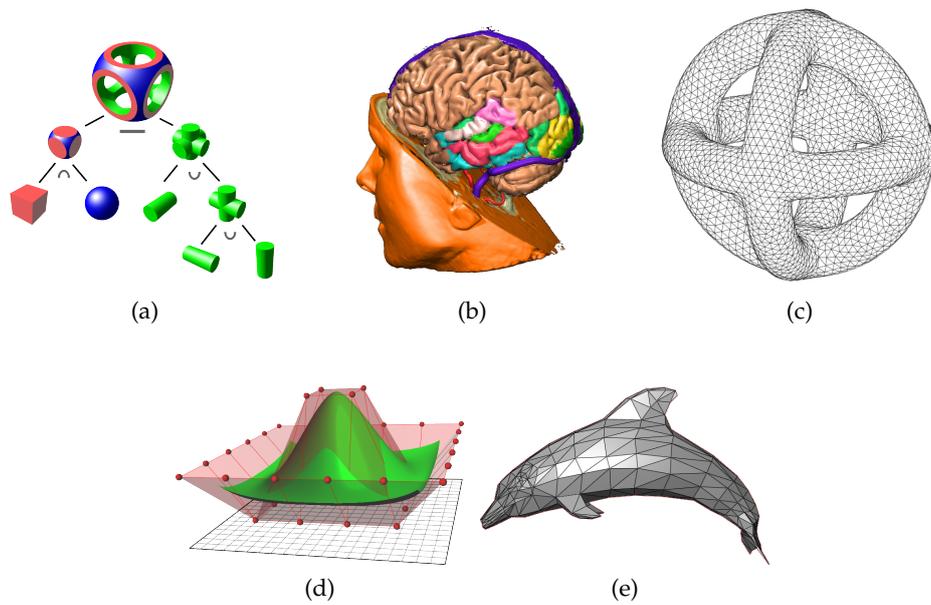


FIGURE 2.1 – Différentes représentations d'un objet 3D. -(a) géométrie CSG : arbre des opérations booléennes -(b) visualisation par voxels en couleur -(c) intersection de trois tores, la visualisation est faite par triangulation -(d) représentation d'une surface lisse par l'utilisation de NURBS -(e) représentation par maillage. Toutes ces figures sont issues de *wikimedia Commons* sous licence *Creative Commons ShareAlike*
commons.wikimedia.org

tique graphique, que ce soit pour le rendu ou la modélisation utilisent cette représentation. Nous nous concentrerons donc dans cet état de l'art sur les méthodes permettant de construire ces maillages.

Les outils permettant d'obtenir un maillage 3D sont aussi très nombreux. Dans un souci de rendre plus efficace la création de ces maillages, la communauté scientifique s'est efforcée de développer de nouvelles méthodes de modélisation, soit pour la rendre plus accessible à des utilisateurs novices, soit pour permettre une plus grande variété de création.

Nous présenterons par la suite les méthodes conventionnelles de modélisation par polygones dans la section 2.2.1, ceci afin d'observer les avantages et les inconvénients des outils de modélisation proposés dans les logiciels commerciaux. La section 2.2.2 présente des travaux de l'état de l'art cherchant à simplifier la tâche de modélisation à l'aide d'une interface de dessin en 2D. Enfin, dans la section 2.2.3, nous donnons un aperçu des outils de modélisation procédurale qui permettent d'obtenir une grande

variété de modèles 3D, soit par l'utilisation d'une grammaire générative, soit par l'exploitation de modèles existants.

2.2.1 Outils de modélisation conventionnels

Dans la modélisation polygonale, un modèle 3D est représenté sous la forme d'un graphe composé de sommets reliés par des arêtes. Bien qu'il soit envisageable de spécifier manuellement le graphe pour un polyèdre simple, des outils plus adaptés sont nécessaires afin de concevoir des maillages complexes.

Les outils de modélisation conventionnels proposent tous un ensemble d'outils communs. Ils permettent généralement d'importer un modèle existant, ou de créer des primitives géométriques simples (cubes, cylindre, cône, etc.). À partir de ces primitives, il est alors possible d'effectuer différentes opérations :

- subdivision des facettes en ensemble de facettes par insertion de sommets ;
- extrusion de facettes ;
- connexion de primitives ;
- opérations booléennes sur des surfaces fermées ;
- applications de transformations rigides sur des groupes de sommets ;
- etc.

De cette manière, il est possible de créer à partir de primitives simples des modèles plus complexes. Pour illustrer le processus de conception, nous prenons un exemple de modélisation d'un être humain². Le temps nécessaire pour la modélisation complète est d'environ 45 minutes. Différentes étapes de modélisation sont présentées dans la figure 2.2.

Pour cette modélisation, le designer utilise deux images 2D afin de se guider dans la modélisation. C'est une pratique couramment utilisée mais qui n'est pas obligatoire. Comme on peut le voir dans la figure 2.2a, le

2. url de la vidéo : https://www.youtube.com/watch?v=sHN1_GBwqcw - licence Creative Commons ShareAlike.

designer utilise pour débiter la modélisation trois cylindres : un pour le corps, un pour le bras et un pour la jambe.

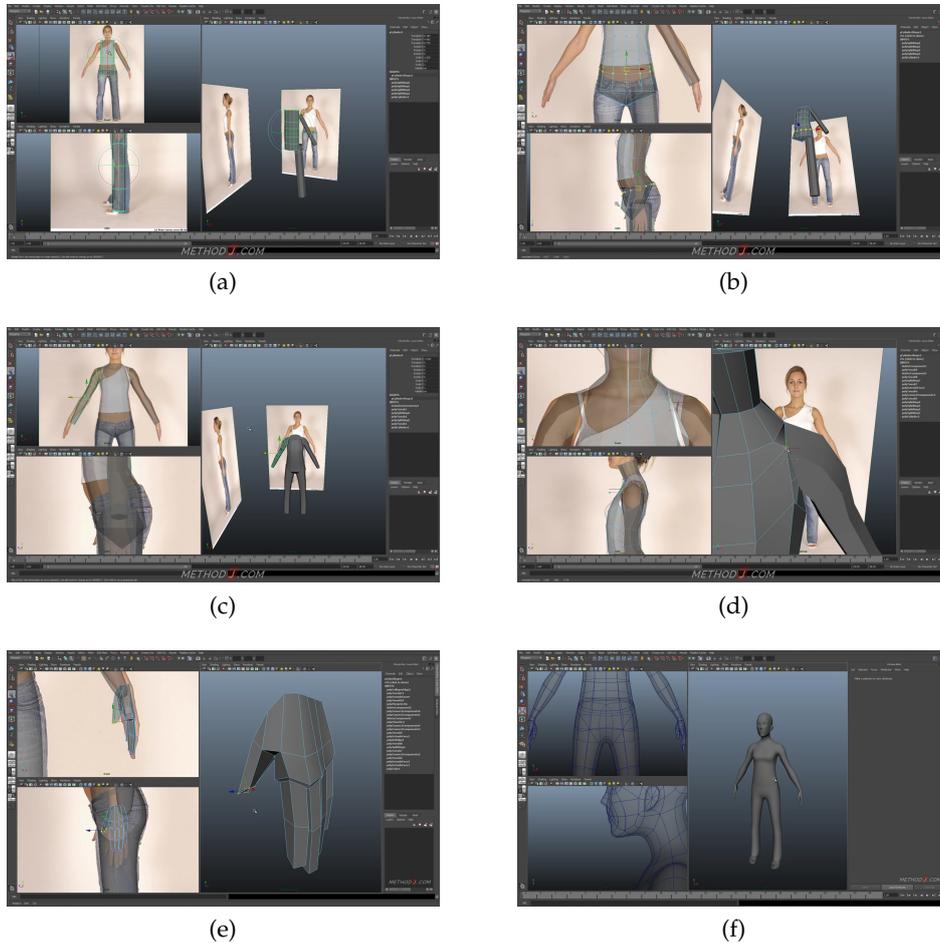


FIGURE 2.2 – Exemple de modélisation à l'aide d'un modèleur 3D conventionnel (Maya^a)
 -(a) Début de la modélisation à partir de trois cylindres -(b) Transformation des groupes de sommets pour modeler le buste -(c) Duplication du bras et de la jambe -(d) Création de facettes à la jointure entre les parties -(e) Création de la main à partir d'un cube par extrusion -(f) Modèle final complet. La modélisation complète prend environ 45 minutes.

La vidéo est disponible à l'URL suivante :

https://www.youtube.com/watch?v=sHN1_GBwqcw

a. <http://www.autodesk.fr/products/maya/overview>

Pour obtenir la silhouette du corps, les groupes de sommets du cylindre (en jaune dans la figure 2.2b) sont transformés indépendamment. Le même procédé est appliqué pour le bras et la jambe. Ces derniers sont ensuite dupliqués, afin d'obtenir l'autre jambe et l'autre bras, comme illustré par la figure 2.2c.

Les parties étant disjointes, il faut alors les associer. Pour cela, de nou-

velles facettes sont créées, à partir des arêtes au bord de chaque partie, comme illustré dans la figure 2.2d.

Les mains sont construites à partir d'un cube, par une succession de subdivisions de facette et d'extrusion. Le résultat est visible dans la figure 2.2e. Les pieds et la tête sont construits selon le même principe. La jonction entre les différents membres se fait par l'ajout de facettes entre les arêtes de bord, comme vu précédemment. Le résultat final est visible dans la figure 2.2f.

Les outils de modélisation conventionnels, tel que celui que nous avons présenté ici, permettent d'effectuer des modélisations complexes. Néanmoins, la modélisation en utilisant ces outils reste une tâche fastidieuse et complexe. Ces outils ne permettent pas de réutiliser facilement des réalisations existantes, puisqu'ils ont été conçus principalement pour démarrer la modélisation à partir de rien.

2.2.2 Outils à base d'esquisses 2D

Afin de rendre la modélisation 3D plus accessible, des méthodes de modélisation 3D ont été conçues pour être utilisées avec une interface de dessin en 2D [92, 31, 58, 70, 48]. Un état de l'art a été proposé par Olsen *et al.* [60]. Dans la méthode de Igarashi *et al.*[31], l'utilisateur peut créer une nouvelle forme en dessinant son contour 2D, extruder une partie de la surface, couper une partie du maillage ou le lisser. Ces opérations sont contrôlées par l'utilisateur en dessinant le contour de la forme à créer, en traçant un trait pour couper le maillage, ou en dessinant le nouveau contour de la partie à extruder. La figure 2.3 illustre ces fonctionnalités.

En revanche, cette simplicité d'utilisation limite le type de forme que l'on peut créer. Le système de Igarashi *et al.* par exemple, ne permet de créer que des surfaces topologiquement sphériques : il ne serait pas possible de créer un tore en utilisant une telle interface. De plus, ces outils sont conçus pour une modélisation complète, et ne permettent pas d'exploiter des modèles déjà existants. Nous verrons dans la section 2.3 que plusieurs méthodes utilisent une interface 2D pour composer de nouveaux modèles 3D à partir de modèles existants.

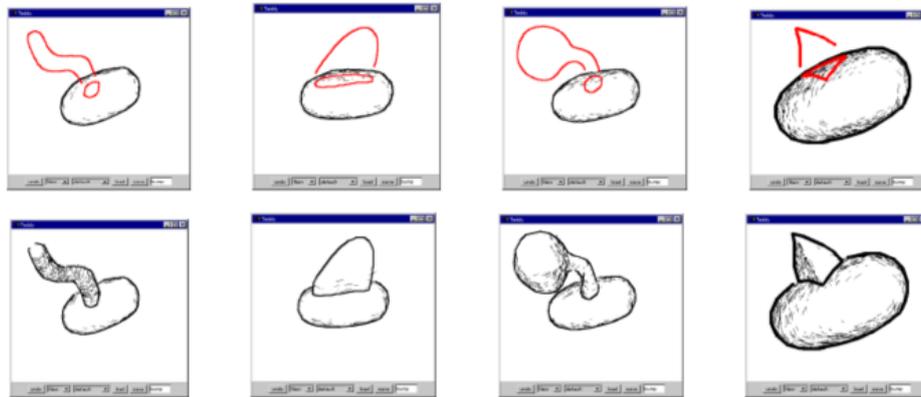


FIGURE 2.3 – Interface de modélisation par esquisse, extrait des travaux de Igarashi et al. [31]. Le contour de la partie à ajouter est dessiné (en haut), et le maillage 3D est mis à jour (en bas).

2.2.3 Outils de génération procédurale

Afin d'obtenir un grand nombre de maillages rapidement, les outils de génération procédurale proposent de remplacer l'intervention d'un utilisateur en générant des nouveaux modèles de manière algorithmique [66, 64, 57, 79, 56, 28].

Prusinkiewicz [66] utilise un système de Lindenmayer afin de décrire la structure géométrique des plantes. Il devient ainsi possible de générer une nouvelle plante en changeant les axiomes de départ et les règles de reconstruction. La création du modèle lui-même est alors vue comme un problème de réécriture. En utilisant une grammaire, il est aussi possible d'utiliser des primitives 3D complexes comme dans les travaux de Guo *et al.* [28]. Dans cette méthode, les règles de production sont spécifiques à la création de monstres humanoïdes, mais les primitives géométriques utilisées sont extraites d'une base de modèles existants. Ces systèmes sont cependant assez limités : la structure ou les règles de réécriture ne sont pas facilement généralisables à une autre famille d'objets.

La synthèse de modèles à partir d'exemples résoud ce problème en utilisant un ensemble de modèles 3D, en analysant leurs composants et les règles qui les lient, puis génère de nouveaux modèles à partir de ces composants et de ces règles [56]. La figure 2.4 présente les modèles exemples ainsi que les modèles générés à partir de contraintes de connectivité.

Ces méthodes sont bien adaptées pour analyser la structure d'un mo-



FIGURE 2.4 – Exemple de synthèse de modèles à partir d'exemple, extrait de Merrell *et al* [56]. À partir d'un ensemble de modèles (à gauche), de nouveaux modèles sont générés (à droite). Les sommets entourés en rouges définissent des contraintes de connectivité (ici, ils doivent être à l'intersection de 4 faces).

dèle complexe, et pour générer des foules, des villes ou des paysages contenant de nombreux modèles variés de manière automatique. Cependant, bien que l'utilisateur puisse définir un ensemble de contraintes pour la génération, il n'est pas possible d'avoir un contrôle plus fin pour la création.

L'édition de maillage à partir d'exemples permet de simplifier la création d'un modèle 3D, en exploitant un ensemble de modèles déjà existants, tout en laissant un contrôle fin à l'utilisateur.

2.3 MODÉLISATION À PARTIR D'EXEMPLES

Les méthodes de modélisation à partir d'exemples permettent de créer de nouveaux modèles en exploitant un ensemble de modèles existants, avec différents niveaux d'intervention de l'utilisateur. Les outils présentés dans la section 2.2 permettent de créer un modèle complet sans exploiter de données existantes, à l'exception notable des outils de génération procédurale (présentés dans la partie 2.2.3). Ces derniers ne sont pas considérés dans cette partie, puisque ce sont des outils automatiques de synthèse de modèle, et n'impliquent pas d'intervention de la part de l'utilisateur.

Les travaux de Funkhouser *et al.* [26] sont fondateurs dans le domaine de la modélisation à partir d'exemples. Ce sont les premiers travaux à exploiter des outils d'indexation de maillages et de segmentation manuelle pour créer un nouveau modèle par assemblage de segments. Étant donnée

l'importance de ces travaux dans ce domaine, ils sont exposés dans la section 2.3.1, de manière distincte des autres méthodes.

Nous présentons ensuite différentes méthodes de modélisation à base d'exemples selon une classification en trois familles. La section 2.3.2 s'intéresse aux méthodes de génération à partir d'un ensemble de modèles : ces méthodes permettent d'explorer un ensemble de modèles et de générer des variations intermédiaires entre des modèles choisis par l'utilisateur. Dans la section 2.3.3, nous étudions des méthodes d'assemblage de segments utilisant une interface d'esquisse 2D. Enfin, la section 2.3.4 s'intéresse en particulier aux méthodes de suggestions intelligentes pour l'assemblage de parties.

2.3.1 Modélisation à partir d'exemples : les premiers travaux

Les travaux de Funkhouser *et al.* [26] sont pionniers en matière de modélisation à partir d'exemples. À partir d'un ensemble de modèles, le système permet de créer de nouveaux modèles en recherchant des parties de modèles dans la base, et en les ajoutant à la création par assemblage.

Le scénario d'édition est le suivant : la modélisation s'effectue à partir d'un modèle de départ en sélectionnant les parties à échanger et en recherchant les modèles partiellement similaires dans la base.

À l'aide d'un système de découpage intelligent, l'utilisateur sélectionne sur le modèle de départ la partie qu'il souhaite remplacer. Les modèles contenant des parties similaires sont retrouvés dans la base à l'aide d'une méthode de recherche partielle basée sur une distance entre deux grilles de voxels sur les modèles. Les modèles présentés à l'utilisateur sont des modèles complets. L'utilisateur doit alors découper sur l'un des modèles retournés la partie à ajouter. En effet, la recherche partielle indique qu'un modèle contient une partie, mais elle ne donne pas d'information sur la position et l'étendue de cette partie correspondante dans l'objet. Une fois la nouvelle partie sélectionnée par l'utilisateur, elle est attachée au modèle de départ (auquel on a retiré la partie à remplacer) en connectant les deux surfaces. La figure 2.5 présente la modification d'un visage par remplacement de différentes parties.

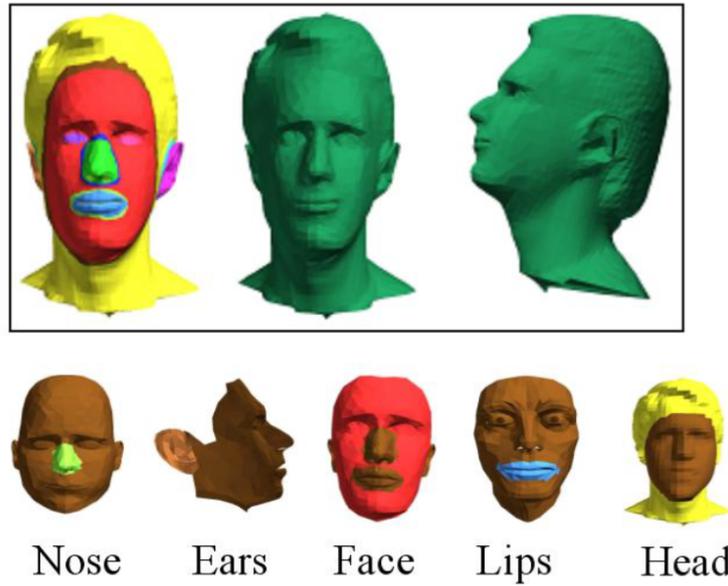


FIGURE 2.5 – Illustration de la méthode de Funkhouser [26]. En haut : visage obtenu par assemblage de parties ; chaque partie est colorée différemment. En bas : les modèles retrouvés dans la base de visages, dont les parties ont été sélectionnées par l'utilisateur.

Cette méthode permet d'exploiter un ensemble de modèles 3D afin de créer un nouveau modèle. Elle met en avant l'importance des mécanismes d'indexation partielle pour la modélisation à partir d'exemples. Les méthodes présentées dans la suite de cet état de l'art diffèrent par le mode de requête de la base et par le degré d'implication de l'utilisateur, mais conservent le principe de construire un nouveau modèle par assemblage de parties issues d'une collection.

2.3.2 Exploration et variation sur un ensemble de modèles

Il existe un grand nombre de collections de modèles 3D en ligne, spécifiquement pour les objets manufacturés. Ces objets ont la particularité d'avoir une grande variété géométrique, et de résulter naturellement de l'assemblage de plusieurs parties. Les méthodes présentées dans cette section envisagent la modélisation à partir d'exemple par une approche exploratoire : les modèles 3D sont considérés comme faisant partie d'un espace de formes. Il est alors possible d'explorer cet espace, et de générer de nouvelles formes à partir des formes existantes [90, 32, 95, 1, 4].

Ces méthodes génèrent de manière semi-automatique de nouveaux modèles en analysant un ensemble de formes. Certaines méthodes permettent de générer des variations entre deux modèles [32, 1]. L'objectif dans ces méthodes est d'éviter le plus possible les variations indésirables : les modèles sont synthétisés de façon à satisfaire un ensemble de contraintes déduit après une étape d'analyse. Dans la méthode présentée par Jain *et al.* [32], chaque modèle de la base est analysé préalablement : il est d'abord segmenté, recalé et mis à l'échelle. Les points de contacts (intersections) entre les différents segments sont localisés, et les éventuelles symmétries entre les différentes parties sont identifiées. Puis une décomposition hiérarchique du modèle en différentes parties est construite. Une fois que cette analyse a été effectuée sur chacun des modèles individuellement, il est possible de créer un nouveau modèle, en *mélangeant* deux modèles source. Cette synthèse est faite en appariant les hiérarchies obtenues pour les deux modèles, puis en échangeant les parties de modèles,

tout en s'assurant de respecter les contraintes, notamment sur les points de contacts entre les parties.



FIGURE 2.6 – Résultats de mélanges entre deux modèles, extrait de Jain et al. [32]. Les modèles présents sur la diagonales sont les modèles originaux, tandis que ceux présentés sur l'anti-diagonale correspondent à un mélange avec un poids de 0.5.

L'approche présentée par Alhashim *et al.* [1] considère qu'il existe différentes séquences de remplacement permettant de passer d'un assemblage à un autre. Dans cette méthode, plusieurs modèles sont créés à partir d'une source et d'une cible, selon plusieurs *chemins de mélange*. De façon analogue au travail de Jain *et al.*[32], les créations improbables sont filtrées selon des conditions de symétrie et de contact entre les parties. La différence principale est qu'il est possible dans cette approche de fusionner ou de diviser les parties, permettant ainsi des changements topologiques.

Plutôt que de considérer une source et une cible, d'autres méthodes [90, 95, 4] considèrent des ensembles de modèles plus grand. Ainsi,

Xu *et al.* [90] proposent de faire évoluer un ensemble de modèles en passant par des étapes de *croisements* (échange de parties) et de *mutations* (déformation de parties choisies aléatoirement). Dans cette approche, l'utilisateur choisit à chaque itération un sous-ensemble de modèles à conserver pour guider l'évolution. Dans *ShapeSynth* [4], les modèles sont conjointement segmentés et labellisés, puis chacun est représenté par un vecteur de configuration $X_i \in \mathbb{R}^{6t}$ qui caractérise les dimensions et le centre des boîtes englobantes de chaque partie, avec t le nombre de parties. Un sous-ensemble de ces modèles est ensuite sélectionné aléatoirement, puis représenté dans \mathbb{R}^2 à l'aide d'un positionnement multidimensionnel (MDS). L'utilisateur peut alors sélectionner un nouveau point dans le plan 2D, correspondant à un vecteur $X_i \in \mathbb{R}^{6t}$ dans l'espace de configuration. Les segments sont ensuite sélectionnés pour respecter les contraintes de dimensions des boîtes englobantes décrites dans X_i , ainsi que les contraintes de symétrie et de contact entre les parties. La figure 2.7 présente l'interface d'exploration : les points sont projetés dans un espace 2D, et les nouveaux modèles peuvent être générés à partir de la paramétrisation.

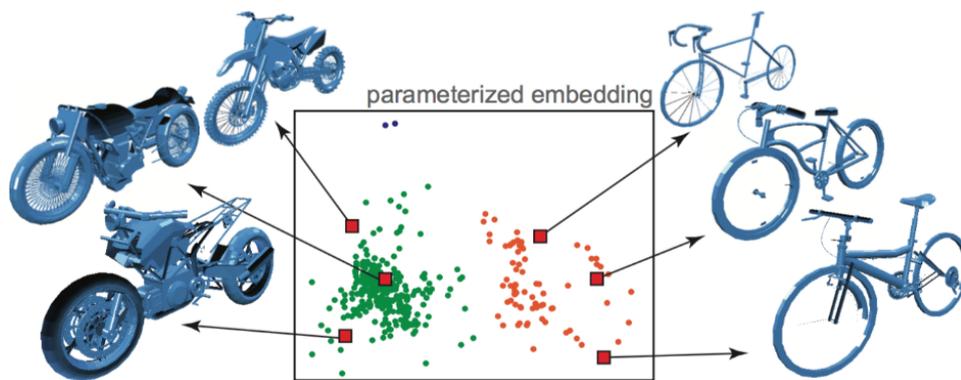


FIGURE 2.7 – Projection en 2D d'un ensemble de modèles, extrait de Averkiou *et al.* [4]. La projection permet d'explorer des ensemble de modèles (points rouges et verts) et de générer de nouveaux modèles (rectangles rouges).

Les méthodes que nous venons de présenter permettent d'obtenir de nouvelles variations à partir d'un sous-ensemble de modèles. Ces approches sont assez différentes de celle présentée par Funkhouser *et al.* [26], puisque l'utilisateur ne contrôle pas directement les parties de modèles qu'il va ajouter ou remplacer. Néanmoins, l'utilisateur participe à la création de ces nouveaux modèles, soit en sélectionnant un modèle source

et un modèle cible [32, 1], soit en explorant un vaste ensemble de modèles [90, 94, 4], ce qui distingue ces méthodes des approches de synthèses de modèles décrites dans la partie 2.2.3. Dans la suite de ce chapitre, nous verrons des approches impliquant plus directement l'utilisateur en lui proposant pour l'assemblage de segments une interface de dessin en section 2.3.3 ou un système de suggestion intelligent en section 2.3.4.

2.3.3 Méthode d'assemblage à partir d'esquisse 2D

Les méthodes que nous avons vues précédemment dans la section 2.3.2 permettent d'exploiter une base de modèles 3D pour créer de nouveaux modèles, mais l'intervention de l'utilisateur quant à la géométrie qu'il va réutiliser est indirecte : l'utilisateur ne peut pas désigner spécifiquement quel segment va apparaître dans la composition finale.

Les méthodes qui sont présentées dans cette section reprennent le principe des travaux de Funkhouser *et al.* [26] : la modélisation se fait par substitution de parties de modèles. La différence principale dans ces approches est double. Plutôt que de définir à l'aide d'un système de découpage intelligent les parties à échanger, les modèles sont pré-segmentés et labellisés : ce sont ces segments que l'on va pouvoir échanger. La deuxième différence se situe dans le mode de requête : là où Funkhouser *et al.* proposent un système de requête basé sur la géométrie 3D du maillage (une grille de voxels), la requête dans les méthodes que nous présentons maintenant est faite sous la forme d'une esquisse 2D.

Dans la méthode présentée par Xie *et al.* [87], les modèles 3D constituant la base d'exemples sont préalablement segmentés et labellisés dans une étape hors-ligne. Tous les modèles de la base appartiennent à une même catégorie (par exemple, la base ne contient que des modèles de chaises). Les contours 2D de chaque segment observé selon différentes orientations sont également pré-calculés. Dans l'étape de modélisation en ligne, l'utilisateur va dessiner par dessus la partie de modèle qu'il souhaite remplacer, comme illustré dans la figure 2.8. Les segments étant labellisés, la recherche est limitée aux segments portant le même label. Cette recherche se fait par comparaison du contour 2D de la requête et

du segment candidat par une approche par sac-de-mots utilisant un filtre de Gabor, réduisant le problème de correspondance 3D à un problème de comparaison de vues 2D. Cette recherche utilise aussi la similarité entre le contour du segment candidat et les contours des parties adjacentes, afin d'assurer une cohérence géométrique.



FIGURE 2.8 – Capture d'écran de l'interface de modélisation par esquisse 2D, extrait de Sketch to Design [87]. Le modèle est visible sur la gauche. Sur la droite, l'utilisateur dessine par dessus un segment, et les candidats pour le remplacement sont présentés dans la galerie dans la partie supérieure de l'interface.

L'approche de Fan *et al.* [23] repose également sur les contours de vues échantillonnées autour de chaque segment. La modélisation se fait non pas par remplacement mais par ajout de parties : l'utilisateur dessine un contour, et les segments correspondants sont proposés. Puisqu'il s'agit d'une méthode additive, il n'y a pas de connaissance sur le label ou la fonctionnalité de la partie que l'on souhaite ajouter. En revanche, cette méthode introduit un mécanisme de suggestion de contours pour guider l'utilisateur : ces contours sont issus des vues de segments dont les parties adjacentes sont déjà présentes dans le modèle.

Les méthodes d'assemblages à partir d'esquisse permettent de créer de nouveaux modèles à partir de modèles existants en dessinant le contour de la partie que l'on souhaite ajouter. Cette requête 2D a l'avantage de

grandement simplifier l'interaction par rapport à la méthode proposée par Funkhouser *et al.* [26]. En effet, les modèles sont pré-segmentés, ce qui facilite leur indexation et permet une correspondance sémantique simple. Dans le cas de Xie *et al.*, le dossier d'une chaise ne pourra être remplacé que par des segments portant le même label. De la même manière, la labellisation des segments dans les travaux de Fan *et al.* permet de guider les suggestions faites à l'utilisateur. Nous verrons dans la section suivante d'autres méthodes d'assemblage de segments qui utilisent directement la géométrie 3D comme requête pour suggérer des parties à ajouter ou à remplacer.

2.3.4 Méthode d'assemblage avec suggestions de segments

Dans la section 2.3.3, ont été présentées des méthodes de modélisation par l'exemple utilisant une requête 2D pour retrouver des segments à ajouter ou pour remplacer une partie déjà présente. Ce type d'interface nécessite d'avoir une idée des modèles présent dans la base. Nous allons exposer dans cette section des méthodes de modélisation exploitant des mécanismes permettant de suggérer des parties de modèles 3D à ajouter. Comme dans les sections 2.3.2 et 2.3.3, dans cette section, les modèles utilisés sont préalablement segmentés, ce qui permet de simplifier la recherche partielle et de minimiser l'intervention de l'utilisateur, contrairement à la méthode de Funkhouser *et al.*.

Le système proposé par Kreavoy *et al.* [43] se concentre sur l'obtention d'une segmentation et d'une labellisation cohérente sur l'ensemble des modèles de la base, pour limiter les manipulations géométriques telles que le découpage semi-automatique. Une segmentation hiérarchique est d'abord calculée pour chacun des modèles. Une correspondance entre ces différents segments est ensuite établie, groupant éventuellement les segments, en minimisant une fonction de coût prenant en compte la convexité des groupes de segments, mais aussi le contexte des différentes parties à partir d'un graphe d'adjacence. Une fois ces correspondances établies dans une étape hors-ligne, l'interface permet d'échanger des composants associés, et de composer un nouveau modèle. La figure 2.9 présente un modèle

créé à partir d'une segmentation cohérente sur un ensemble de modèles hétérogènes.

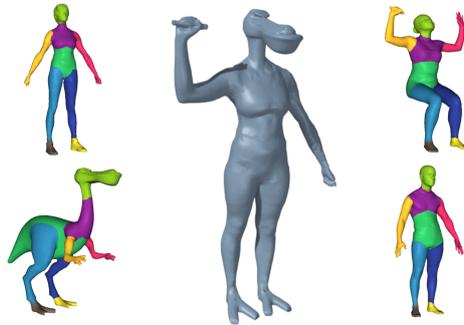


FIGURE 2.9 – Assemblages de segments de modèles à partir d'une labellisation cohérente, extrait de Kravovoy et al. [43]. Au centre, le modèle est composé de segments extraits des modèles disposés autour, labellisés de manière cohérente. Les couleurs correspondent aux labels.

L'approche de Chaudhuri et Koltun [13] diffère car elle introduit la notion de suggestion à partir des données. Il est alors possible d'ajouter un composant qui ne fait pas partie du modèle en cours de création, sans remplacement. Partant d'un modèle de départ, le système identifie un sous-ensemble des modèles de la base étant globalement similaire, en utilisant une métrique de similarité basée sur un histogramme 3D de distances paire-à-paire de valeurs du diamètre local (*Shape Diameter Function* [73]). Une fois ce sous-ensemble obtenu, il s'agit de présenter à l'utilisateur des segments qui ne sont pas présents dans le modèle en cours de création. Un score de correspondance est alors déterminé pour chaque segment du sous-ensemble de modèles retrouvé, basé sur des descripteurs locaux (*Shape Diameter Function* et *Spin Images* [34]). Les segments dont le score de correspondance se trouve en dessous d'un seuil sont ceux qui sont susceptibles de compléter le modèle en cours de création. Ce sont donc ceux-ci qui vont être suggérés à l'utilisateur. Ce score de correspondance est calculé à l'aide de descripteur purement géométrique.

Chaudhuri *et al.* [12] introduisent une information sémantique afin d'améliorer les suggestions proposées. À partir d'un ensemble de modèles segmentés et labellisés, une étape hors-ligne permet de construire un modèle probabiliste décrivant les interactions entre les segments des modèles

de la base. Cette connaissance est représentée sous la forme d'un réseau Bayésien. Ce modèle est appris à partir de la base de modèles et décrit les relations d'adjacences, de symétrie, la cardinalité des labels, mais aussi une description géométrique des segments appartenant à la base. Ces relations sont représentées sous la forme de variables aléatoires et les nœuds du graphe sont des tables de probabilités conditionnelles entre ces variables aléatoires. La structure et les probabilités du réseau sont apprises à partir des données, de manière à maximiser le critère d'information Bayésien.

Un modèle en cours de création va être représenté sous la forme d'observations des variables aléatoires précédemment discutées. Ces observations vont être utilisées ensuite pour inférer à partir du réseau Bayésien la probabilité d'ajout d'un segment provenant d'une catégorie sémantique donnée, étant données les relations observées entre les labels sur le modèle en cours.

Prenons par exemple un modèle composé de deux bras et d'un torse. La probabilité d'ajouter un bras est évaluée comme étant la probabilité conditionnelle $P(X|O)$ où O est l'ensemble des variables aléatoires observées (adjacences entre le bras et le torse, cardinalité de deux pour le bras, un seul torse, etc.), et X est la probabilité jointe que la cardinalité du label bras augmente (supérieure ou égale à trois), et que le nouveau bras soit adjacent à l'un des segments présents. Ici, si la base est constituée de modèles humanoïdes, alors cette probabilité sera plus faible (puisque aucun modèle de la base ne contiendrait trois bras) que d'ajouter un segment portant le label tête par exemple. La figure 2.10 résume les deux phases de l'approche, l'apprentissage du hors-ligne du réseau Bayésien, et l'inférence lors de la modélisation.

Une fois que la catégorie sémantique la plus probable a été déterminée, les segments sont classés au sein de chaque catégorie sémantique étant donné leur style géométrique (représenté par le cluster obtenu sur la signature géométrique), de manière à maintenir une cohérence stylistique : l'exemple donné par les auteurs est qu'il sera plus probable d'ajouter un bras d'alien à un modèle si le modèle en cours de création contient déjà

une tête d'alien (selon le descripteur géométrique). Un modèle probabiliste similaire est utilisé par Kalogerakis *et al.* [36] pour la synthèse de modèles.

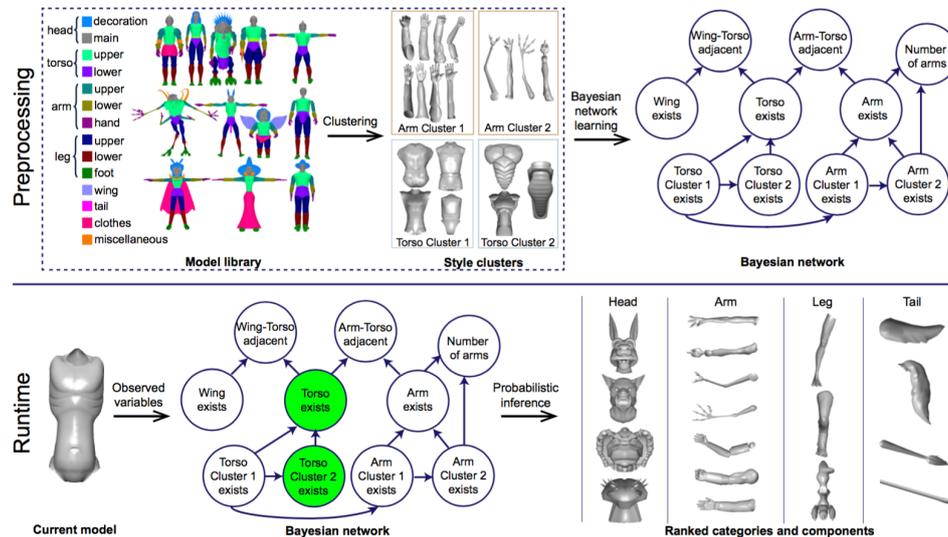


FIGURE 2.10 – Suggestions de segments à l'aide d'un modèle probabiliste, extrait de Chaudhuri *et al.* [12]. En haut : dans une phase hors-ligne, les segments sont catégorisés en fonction de leur label sémantique et de leur géométrie. Un réseau Bayésien construit afin de généraliser la connaissance sur la base, en fonction du nombre d'occurrences d'un label dans un modèle, des relations d'adjacences, de l'appartenance à un cluster géométrique, etc. En bas : lors de la création d'un nouveau modèle, les segments déjà assemblés permettent d'inférer sur le réseau bayésien la probabilité d'ajout d'un segment de la base, étant donné ceux déjà présents.

Enfin, Chaudhuri *et al.* [11] utilisent des caractéristiques sémantiques de plus haut niveau pour assister la création de nouveaux modèles. Plutôt que de considérer un label sémantique associé à une partie, (tête, bras, jambe, etc.), chaque segment se voit attribuer un classement pour plusieurs attributs, à l'aide d'une annotation manuelle sollicitant de nombreuses personnes. La nature de ces attributs est plus vaste que précédemment : chaque segment se voit attribuer un score selon différents critères (l'aérodynamisme, le caractère effrayant, la dangerosité, etc.). Un sous-ensemble des segments de la base ont été annotés manuellement, selon ces différents attributs. Ensuite, une fonction de classement est apprise à partir des données annotées, et des descripteurs géométriques, permet-

tant ainsi de généraliser le classement à tous les segments pour tous ces attributs.

Les méthodes d'assemblage présentées dans cette partie proposent des mécanismes intelligents de suggestion pour remplacer ou ajouter des segments, soit en utilisant des critères géométriques, ou une approche probabiliste. Ici encore, plusieurs méthodes exploitent une labellisation des modèles segmentés afin de mettre en correspondance les différentes parties du modèle et de les catégoriser simplement. Ces méthodes permettent d'obtenir une grande variété de modèles tout en laissant une grande liberté d'interaction à l'utilisateur.

2.4 CONCLUSION

Les méthodes de représentation des modèles 3D sont nombreuses. Cependant, la représentation sous forme de maillage est la plus utilisée, notamment parce qu'elle facilite le rendu et les manipulations géométriques. Nous nous concentrons donc sur cette représentation.

Les outils de modélisation par maillage 3D classiques demandent une grande expérience d'utilisation et un travail de manipulation parfois fastidieux. Pour cette raison, un grand nombre d'outils d'aide à la modélisation ont été proposés afin de faciliter cette tâche : les outils à base d'esquisse 2D, qui rendent la modélisation accessible mais qui limitent néanmoins les possibilités de créations, les méthodes de générations procédurales, qui permettent d'obtenir de grandes variations de modèles mais qui limitent l'intervention de l'utilisateur, et les outils de modélisation à partir d'exemples. Nous nous intéressons à ces méthodes en particulier, car elles s'intègrent facilement à un pipeline de modélisation classique, tout en permettant d'exploiter des créations existantes.

Parmi ces méthodes de modélisation à partir d'exemples, nous avons présenté en particulier l'exemple de Funkhouser *et al.* [26], puisqu'il s'agit du travail pionnier dans ce domaine. Cette méthode permet de rechercher les modèles 3D contenant une partie des modèles similaires à celle sélectionnée par l'utilisateur, en vue de les échanger. Cette méthode nécessite néanmoins plusieurs manipulations géométriques par l'utilisateur,

notamment le découpage de chacun des modèles. Les méthodes plus récentes de l'état de l'art ont répondu à cette problématique en travaillant à partir de modèles segmentés ou en générant une segmentation compatible entre les modèles. Les travaux effectués en matière de modélisation à partir d'exemples portent principalement sur les modes de recherche partielle de modèles, soit par une requête 2D [87, 23] ou une partie de maillage, ou sur des systèmes de suggestions, permettant de suggérer une partie manquante, soit par une approche géométrique [13] ou par une approche probabiliste [12].

Ces approches d'assemblages de segments permettent d'obtenir facilement des modèles convaincants en exploitant une collection de modèles existants. Les mécanismes proposés dans l'état de l'art rendent la modélisation accessible à tous. Néanmoins, la segmentation initiale peut contraindre l'utilisateur dans son travail de création. Le travail original de Funkhouser *et al.* n'utilise pas une approche basée sur des segments, mais demande à l'utilisateur de découper la requête, et de découper la partie remplaçante sur le modèle retrouvé. Cette tâche, bien que plus contraignante que l'assemblage de parties pré-segmentées, donne une certaine liberté quant à la définition de la requête. Nous présentons dans le chapitre 3 un nouveau descripteur permettant de représenter l'information sémantique de manière moins contrainte qu'une labellisation de segments.

Nous proposons dans le chapitre 4 une méthode utilisant ce descripteur sémantique permettant de sélectionner une sous-région d'un maillage 3D, et de trouver une région correspondante, rendant ainsi le procédé de création moins rigide, les parties à assembler n'étant pas pré-découpées. Enfin, le chapitre 5 présente les solutions aux problèmes d'adaptation géométrique liés à ces assemblages de parties.

DESCRIPTION SÉMANTIQUE

SOMMAIRE

3.1	INTRODUCTION	32
3.1.1	État de l'art sur les descripteurs locaux de maillage 3D	33
3.1.2	Définition du descripteur	36
3.1.3	Propriétés	38
3.2	LABELLISATION DE MODÈLES 3D À L'AIDE DU DESCRIPTEUR SÉ- MANTIQUE	44
3.2.1	La labellisation comme un problème inverse	44
3.2.2	Apprentissage	45
3.2.3	Inférence	45
3.2.4	Résultats	46
3.3	TRAITEMENTS LOCAUX SUR LE MAILLAGE 3D	50
3.4	CONCLUSION	53

3.1 INTRODUCTION

Dans le chapitre précédent, nous avons constaté que les méthodes de modélisation à partir d'exemples offrant le plus d'interactions avec l'utilisateur sont les méthodes par assemblage de segments. Ce fonctionnement facilite grandement les étapes de recherche de parties, de suggestion, ou même d'assemblage des différents segments. Dans ces méthodes, un label sémantique est associé à chaque segment [12, 36, 87, 23, 11] : ce qui permet de représenter une information de haut niveau sur chaque partie d'un objet qui va au delà de son aspect géométrique. Cette information sémantique est particulièrement utile pour déterminer la catégorie de la partie à remplacer [87, 23] ou à ajouter [12, 36, 11]. Il est intéressant d'étudier comment sont obtenus ces labellisations et comment elles peuvent être améliorées.

Un grand nombre de descripteurs géométriques ont été présentés par la communauté scientifique durant les 20 dernières années [80] jusqu'à récemment [19, 42]. Ils décrivent la géométrie de manière locale ou globale, avec différentes propriétés en matière d'invariance et de robustesse (vis-à-vis des isométries, de la résolution, *etc.*). Cette description géométrique est une première étape pour comprendre la structure et comparer des formes 3D. Cependant, cette description géométrique reste à un niveau de compréhension bas, et ne suffit pas pour des tâches de plus haut niveau, comme par exemple retrouver tous les dossiers de chaises dans une base hétérogène.

Plus récemment, la communauté s'est intéressée à inférer une description des modèles 3D de plus haut niveau sous la forme d'une labellisation. Par exemple, dans le cas d'un maillage 3D représentant une chaise, chacune des facettes porte un label dans l'ensemble suivant : $\{\text{dossier, pied, assise}\}$. Cette description de haut niveau peut être obtenue par une technique d'apprentissage supervisé [35] ou par l'utilisation d'une ontologie pré-définie [39]. Cette labellisation sémantique est très utile pour organiser un ensemble de modèles. Cependant, cette approche a l'inconvénient de contraindre les applications à un domaine sémantique fixe. Un seul label sémantique est affecté à chaque facette. Par exemple, dans le cas d'un

modèle humanoïde, chaque facette se voit attribuer un label parmi l'ensemble suivant : $\{bras, jambe, torse, tête\}$. La labellisation est alors contrainte par le niveau de granularité de la segmentation. Dans ce contexte, comment est-il possible de caractériser un sous-ensemble de la géométrie qui correspondrait à l'épaule ?

Nous proposons un nouveau descripteur sémantique continu (illustré dans la figure 3.1), qui encode à la fois les informations géométriques et sémantiques, sous la forme d'un vecteur de scalaires. Le principe est simple : chaque sommet est caractérisé par sa distance géodésique à chaque partie sémantique de l'objet. Cette représentation continue nous permet de mieux caractériser le contexte sémantique de chaque sommet du maillage, ainsi que les relations entre les parties. Dans la figure 3.1, on peut voir que les points p_4 et p_5 portent le même label selon une labellisation classique (ici, *patte*), mais ont des contextes sémantiques différents : p_4 est plus proche du label *tête*, ce qui indique qu'il est situé à l'avant tandis que p_5 en est plus éloigné mais est proche du label *corps*, ce qui indique qu'il est situé dans la partie supérieure de la jambe arrière. Nous montrons dans ce chapitre différentes applications directes de ce descripteur sémantique.

Ce chapitre est organisé de la manière suivante : dans le reste de cette section, nous présentons un bref état de l'art des différentes méthodes de description des maillages 3D (3.1.1) puis nous introduisons la méthode de calcul du descripteur (3.1.2) ainsi que ses propriétés notables (3.1.3)

3.1.1 État de l'art sur les descripteurs locaux de maillage 3D

De nombreux travaux ont été menés dans les 20 dernières années dans la description et la représentation de forme 3D. Les applications en informatique graphique sont nombreuses : indexation, segmentation, correspondance, *etc.*. Dans cette section, nous présentons les approches qui permettent d'obtenir une représentation de la sémantique d'un modèle 3D. Plusieurs études complètes ont été réalisées récemment [80, 86, 89].

Pour des applications de compréhension, représentation et d'indexation, un nombre important de descripteur de formes ont été présentés dans la littérature. Les premiers descripteurs de formes à avoir été uti-

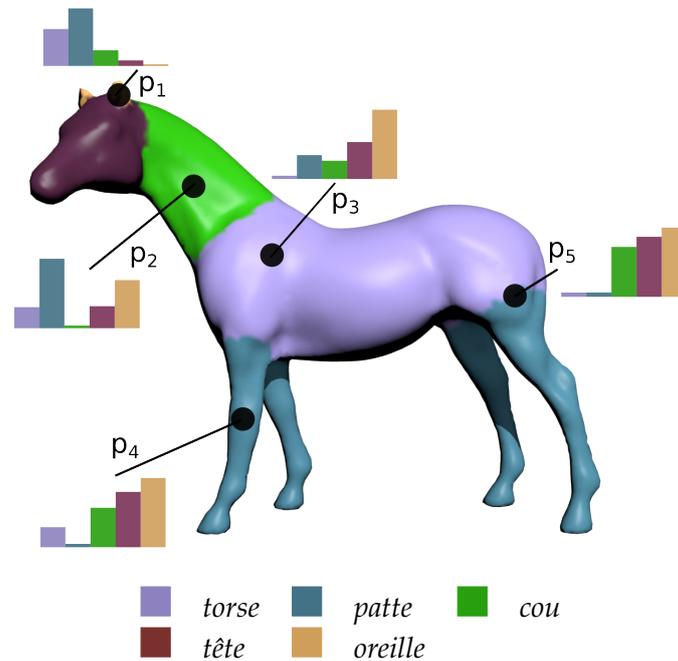


FIGURE 3.1 – Illustration de notre signature sémantique continue pour différents points. Cette signature est construite à partir d’une labellisation préalable du modèle et consiste en un ensemble de distances géodésiques à ces différentes parties sémantiques. Par exemple, le point p_1 est proche du label oreille et tête, mais éloigné des labels corps et patte. Cette signature permet de prendre en compte la nature continue de la sémantique du modèle.

lisés sont des descripteurs *globaux*. Ils représentent une forme complète par une unique signature. Les premiers descripteurs globaux n’étaient invariants qu’aux transformations rigides (par exemple les harmoniques sphériques [26]), tandis que les descripteurs plus récents sont aussi invariants aux transformations non-rigides. Parmi ces derniers, nous trouvons les plongements spectraux [67, 69], les histogrammes de descripteurs locaux [61, 27] et les approches basées sacs-de-mots [10, 45].

Les descripteurs locaux associent une signature par sommet, face ou région d’une forme 3D. Ils peuvent caractériser la courbure locale (*shape index*) [41], le diamètre local (*shape diameter*) [27], un histogramme de gradients [91], les fonctions de chaleur (*heat kernel signature* [77]), ou le *shape context* (caractérisant un champ scalaire par un histogramme en coordonnées polaires) [42]. En dehors de ces descripteurs géométriques, les représentations topologiques sont aussi utiles pour l’indexation de formes et leur interprétation [5]. Ces représentations topologiques sont principalement des représentations sous forme de graphes, telle que les graphes de

Reeb [7, 6, 82] ou des graphes d'adjacences obtenus à partir d'une segmentation [71].

Ces méthodes de description géométriques et topologiques ne donnent pas d'indication directe sur la sémantique d'une forme. Cependant, elles peuvent être utilisées pour obtenir une représentation plus haut niveau, telle qu'une labellisation sémantique. Pour passer de cette représentation bas-niveau vers un niveau de description plus élevé, il est possible d'utiliser une interaction manuelle, couplée à une ontologie décrivant le domaine sémantique, tel que proposé par Attene *et al.* [3]. Les interactions manuelles sont fastidieuses, et peuvent être source d'erreurs. Afin de les limiter, des méthodes récentes utilisent des techniques d'apprentissage, profitant de l'apparition de grandes collections de modèles 3D annotés sémantiquement [35, 51, 84, 88]. Les labels sémantiques pour chaque facettes sont inférés à partir d'un modèle appris sur un ensemble d'apprentissage annoté et de descripteurs géométriques.

Les méthodes de co-labellisation [30, 75, 85] permettent d'inférer des labels cohérents avec le reste d'une collection déjà anotée. D'une manière proche, Kim *et al.* [39] apprennent un *patron* de description pour une collection et l'adaptent afin de labelliser toute la collection. Laga *et al.* [44] ainsi que Zheng *et al.* [94] analysent la relation entre différentes parties d'un modèle pour labelliser la collection.

Obtenir une telle labellisation sémantique sur une collection est particulièrement adapté pour des application de haut niveau telle que l'exploration d'un ensemble de modèles [25] ou la modélisation par assemblage de parties [87, 12, 4]. Cependant, comme nous l'avons expliqué précédemment, ce type de description possède deux inconvénients majeurs : (1) la description sémantique est limitée a une ontologie pré-définie (c'est-à-dire un ensemble pré-déterminé de labels) et (2) cette description ne représente pas la nature continue de la sémantique. Par exemple, pour des formes organiques telles qu'un humanoïde ou un animal, il n'existe pas de limite sémantique stricte entre la jambe et le torse. Il existe cependant des régions qui appartiennent sémantiquement aux deux, dans des proportions différentes. C'est ce que nous cherchons à représenter ici.

3.1.2 Définition du descripteur

Nous souhaitons disposer d'une description sémantique plus souple qu'une labellisation classique qui associe à chaque segment un label unique, en représentant l'information sémantique sous la forme d'un ensemble de champs scalaires sur le maillage. Cette représentation continue de l'information sémantique est calculée à partir d'une labellisation classique : étant donné un maillage segmenté et labellisé, le descripteur sémantique continu consiste en un ensemble de distances géodésiques vers les différents labels. Cette section définit et présente la méthode pour évaluer ce descripteur.

Pour évaluer le descripteur en un sommet du maillage, la première étape est d'échantillonner un ensemble de sommets. Cette étape permet d'être plus robuste aux erreurs de segmentation inhérentes à la plupart des méthodes. Nous ne considérons pas une méthode de segmentation en particulier, mais nous disposons d'un maillage, d'une segmentation grossière et d'un label sémantique pour chaque segment. Cette labellisation est considérée comme étant cohérente pour tous les modèles de la base. De nombreux algorithmes permettent aujourd'hui de d'obtenir une telle labellisation sur un ensemble de modèles 3D [35, 84, 30, 75, 51, 85, 39, 88]. En pratique, dans les résultats présentés dans les sections 3.2 et 3.3, nous utilisons les modèles issus du *Princeton segmentation benchmark* [16], segmentés et labellisés grâce à la méthode de Kalogerakis *et al.* [35]. Des exemples de ces labellisations sont présentés dans la figure 3.2.

Nous utilisons une stratégie d'échantillonnage adaptatif : pour chaque label, nous déterminons un nombre de points proportionnel à l'aire occupée par ce label sur le maillage (avec un nombre d'échantillons minimum pour éviter les labels sous-représentés, comme la queue ou les oreilles). Nous choisissons ensuite les facettes aléatoirement selon leur aire et le label considéré. Nous utilisons $10 + 100 \times \text{area}(\ell) / \text{area}(\text{mesh})$ points pour chaque label, où $\text{area}(\ell)$ est l'aire totale occupée par le label ℓ sur le maillage. La figure 3.3 présente un exemple de ces points échantillonnés.

Pour chaque label ℓ , nous avons donc un ensemble de points S_ℓ . Notre descripteur sémantique, noté \mathbf{d} , dépend d'un ensemble de label



FIGURE 3.2 – Illustration de certaines catégories de modèles (labellisés par Kalogerakis et al. [35]) que nous considérons dans les exemples de ce chapitre : Human (8 labels), Armadillo (11 labels), FourLeg (6 labels) et Airplane (5 labels).

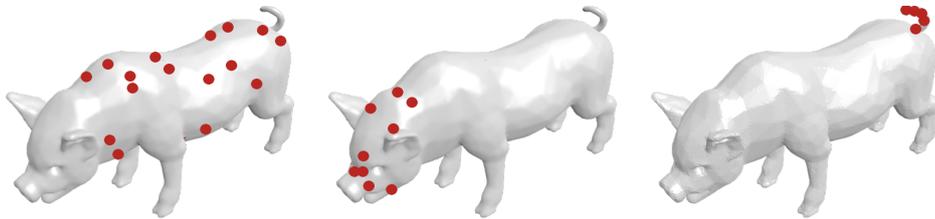


FIGURE 3.3 – Notre stratégie d'échantillonnage des sommets labellisés. De gauche à droite : body, head, tail.

pré-déterminé \mathcal{L} , qui correspond à l'ensemble des labels présents dans la totalité de la base. Pour un sommet p du maillage, le descripteur est calculé comme un vecteur de $|\mathcal{L}|$ éléments. Chaque élément du vecteur $\mathbf{d}(p)$, que l'on notera $d_\ell(p)$, décrit la relation entre ce sommet et le label ℓ . En pratique, un modèle particulier ne contiendra qu'un sous ensemble des labels de \mathcal{L} . Certaines valeurs du vecteur seront alors indéfinies. Le scalaire $d_\ell(p)$ est calculé comme étant la distance géodésique entre le point p et le point le plus proche de S_ℓ , notée $g(p, S_\ell)$. Les points de S_ℓ pouvant appartenir à différents segments de même label, notre descripteur est invariant au nombre d'occurrences d'un label dans un modèle. Par exemple, en considérant une base de modèles humanoïdes, notre descripteur contiendrait (entre autres) la relation géométrique entre les points du modèle et le bras le plus proche. De façon intuitive, le descripteur indique qu'un point sur l'épaule d'un humanoïde (le label *épaule* n'étant pas présent dans la labellisation de départ) serait proche des labels *bras* mais éloigné du label

jambe en terme de distance géodésique. De plus, l'utilisation de distances géodésiques rend notre descripteur robuste aux changements de pose et à la résolution du maillage, comme montré dans la figure 3.4.

Le descripteur est donc calculé tel que :

$$\begin{aligned} \mathbf{d}(p) &= \{d_\ell(p), \ell \in \mathcal{L}\} \\ d_\ell(p) &= \min_{s \in S_\ell} (g(p, s)) \end{aligned} \tag{3.1}$$

où g est la distance géodésique. Quand un label ℓ n'est pas présent dans un modèle, d_ℓ prend pour valeur $d_\ell = \infty$. En pratique, la distance géodésique exacte sur la surface est calculée, en utilisant l'algorithme proposé par Surazhsky et al. [78].

La figure 3.1 illustre le principe de notre descripteur pour quelques points sur un maillage. Il encode de manière continue l'information sémantique et l'information géométrique. À la différence d'une labellisation, il est possible de différencier deux points portant le même label mais ayant des positions géométriques différentes (par exemple les points p_4 et p_5).

La figure 3.4 représente le champ scalaire d_ℓ à la surface du maillage pour différents labels ℓ . Cette figure illustre la richesse de cette signature sémantique par rapport à une simple labellisation.

3.1.3 Propriétés

Notre descripteur sémantique présente plusieurs propriétés intéressantes en terme d'invariance. L'utilisation de la distance géodésique exacte à la surface du maillage, ainsi que l'échantillonnage adaptatif des points labellisés permet de rendre notre descripteur indépendant de la résolution du maillage, comme illustré par la figure 3.5. Pour la même raison, il est aussi invariant aux changements de poses tels que les déformations engendrées par l'utilisation d'un squelette d'animation, comme présenté dans la figure 3.4.

La distance L_2 dans l'espace de notre descripteur définit une nouvelle métrique qui prend en compte la sémantique. La figure 3.6 montre un ensemble d'isolignes du champ de distance à un point sur le genou droit.

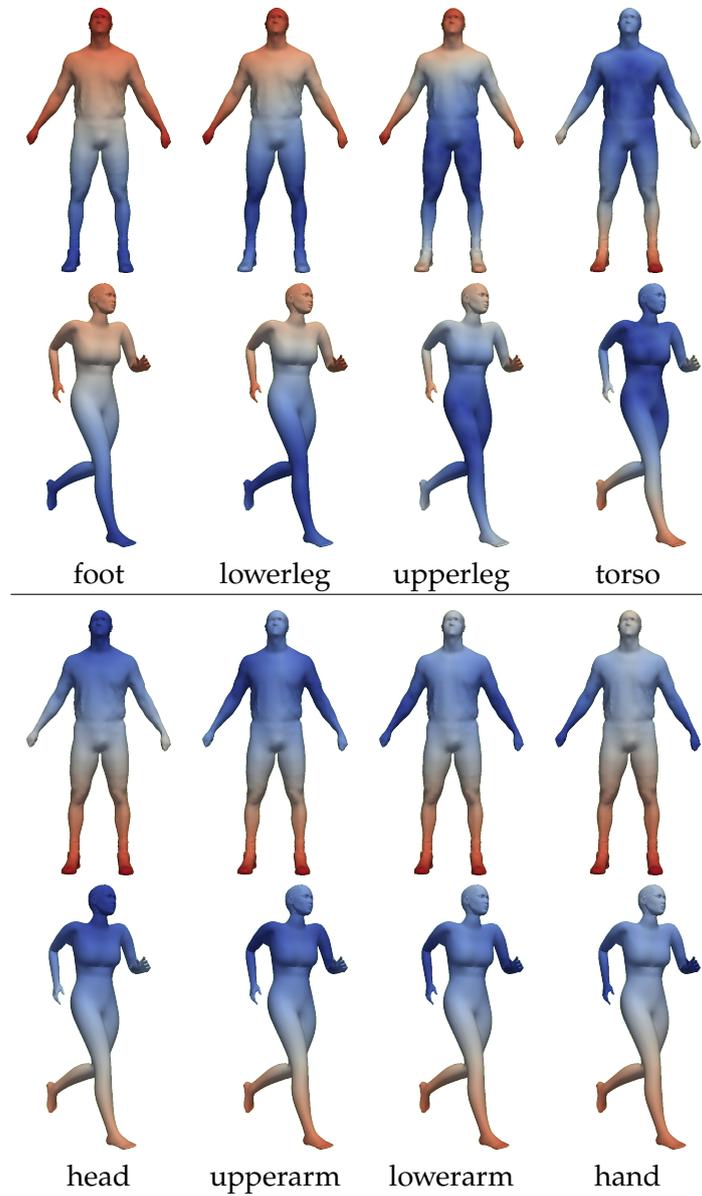


FIGURE 3.4 – Visualisation de $d_\ell(p)$ pour deux maillages de la classe Human du Princeton Shape Benchmark [16].

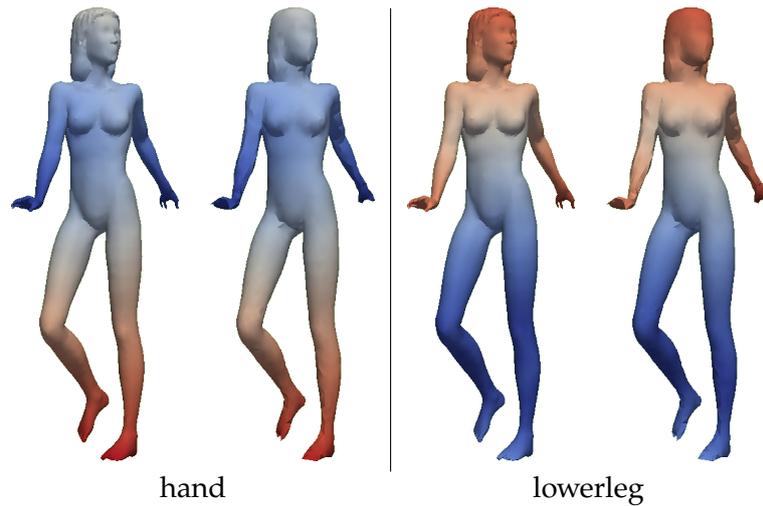
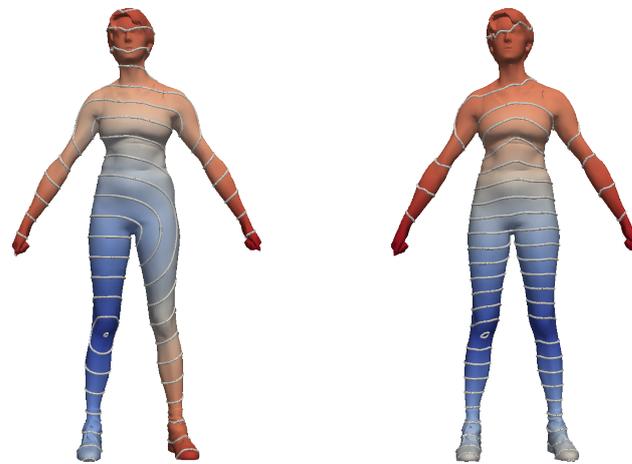


FIGURE 3.5 – Visualisation de $d_\ell(p)$ pour deux labels, avec différentes résolutions de maillages (11015 et 2639 sommets respectivement)

Nous présentons aussi la distance géodésique à ce point en chaque sommet du maillage pour comparaison. Cela permet de mettre en évidence la nature sémantique de la métrique. La distance L_2 utilisée sur un seul



Distance géodésique distance L_2 sur notre descripteur

FIGURE 3.6 – Champ de distance depuis un sommet du genou au reste du maillage, calculé avec une distance géodésique simple (gauche) et avec la distance L_2 dans l'espace du descripteur (droite). On observe que la métrique basée sur le descripteur sémantique continue intègre à la fois des informations géométriques et sémantiques, puisqu'elle capture les variations de distances géodésiques pour tous les labels.

modèle permet d'apprécier si deux points sur la surface du maillage sont *sémantiquement* proches ou non. Munis d'une base de modèles labellisés de manière cohérente, il devient possible d'identifier des similitudes entre des points de maillages différents.

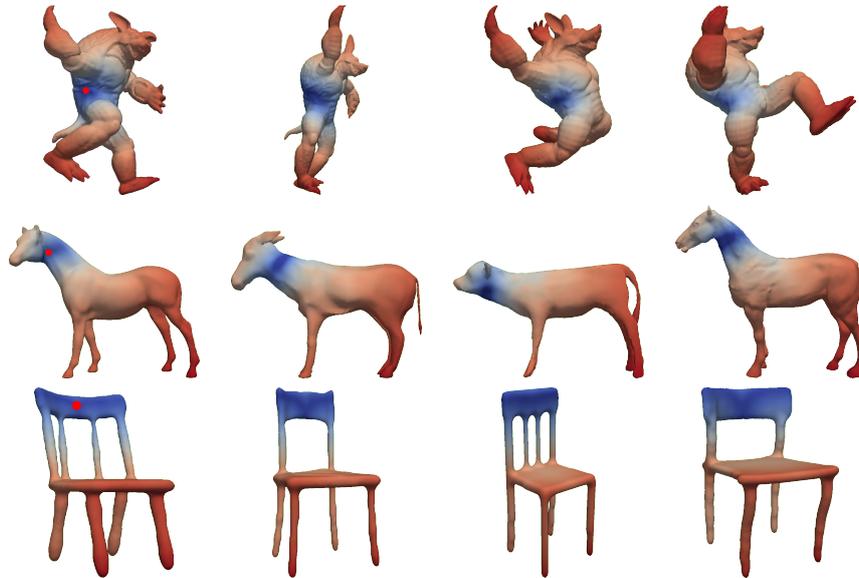


FIGURE 3.7 – Correspondance entre plusieurs modèles, calculé à partir de notre descripteur. Pour chaque classe (Armadillo, FourLeg and Chair), nous calculons la distance L_2 entre le descripteur du point rouge à chaque sommet de chaque modèle de la classe (illustré dans une échelle logarithmique allant du bleu au rouge). La similarité sémantique entre les régions bleues de chaque ligne illustre l'obtention d'une correspondance entre ces modèles.

La figure 3.7 illustre l'utilisation de la métrique précédemment définie pour calculer une correspondance entre différents modèles. Dans les trois exemples, un point (représenté en rouge) est sélectionné sur le premier modèle, et la distance L_2 entre le descripteur de ce point et les descripteurs de chacun des points de tous les modèles est évaluée et représentée sur une échelle de couleur allant du bleu (distance faible) au rouge (distance élevée). La similarité sémantique des zones bleues nous permet de constater qu'une bonne correspondance sémantique est trouvée entre les modèles, dans le cas de changement important de pose (*Armadillo*) ou même de forme (*FourLeg*, *Chair*).

La première rangée de la figure 3.8 illustre une nouvelle fois l'utilisation de la distance L_2 dans l'espace du descripteur sémantique pour des modèles de la classe *Human*. Le point sélectionné sur le coude du premier modèle est correctement positionné sur les autres. La deuxième rangée montre la même expérience, cette fois-ci en ne considérant plus le descripteur sémantique, mais les 608 caractéristiques géométriques extraites par Kalogerakis *et al.* [35]. Malgré la grande variété de caractéristiques

géométriques considérées, la correspondance obtenue n'est pas correcte en terme de sémantique.



FIGURE 3.8 – Correspondance entre les modèles de la classe Human, comparée avec les caractéristiques géométriques de Kalogerakis et al. [35]. En haut : distance L_2 dans l'espace du descripteur sémantique. En bas : distance L_2 sur l'ensemble de descripteur géométrique de Kalogerakis et al. [35] (608 dimensions).

Pour résumer, à partir d'une labellisation grossière (voir figure 3.2), notre descripteur permet d'obtenir une correspondance dense entre les modèles. Cette propriété nous est utile pour l'application de correspondance et de sélection, présenté dans le chapitre 4. D'autres méthodes de correspondance dense, telle que celle de Ovsjanikov et al. [62] sont basées sur des propriétés géométriques locales et ne prennent pas en compte l'information sémantique.

Ce descripteur peut aussi être utilisé pour décrire des segments. Dans ce cas, chaque segment est décrit comme une collection de $|\mathcal{L}|$ histogrammes, obtenus par un échantillonnage uniforme sur le segment et en accumulant le descripteur de chaque sommet échantillonné. Dans la figure 3.9, nous illustrons la répartition des segments pour deux catégories de modèles du *Princeton Benchmark* [16] labellisés par Kalogerakis et al.. Pour comparer les signatures de deux segments, leur distance est définie comme la moyenne des *Earth Mover's distances* entre les histogrammes.

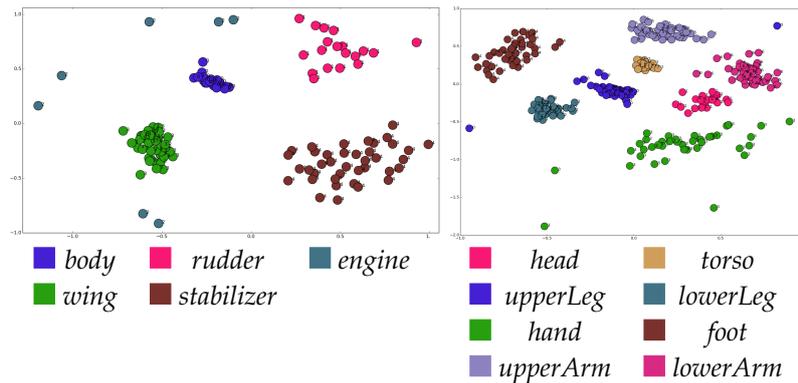


FIGURE 3.9 – Positionnement multidimensionnel des signatures de segments pour les catégories Airplane et Human du Princeton Benchmark [16]

Dans la figure 3.9, chaque point représente un segment, dont la couleur correspond à un label. Le regroupement des points selon leur label valide la qualité sémantique de la distance L_2 appliquée au descripteur.

Nous pouvons aussi faire d'autres observations sur la géométrie et la topologie des modèles considérés : dans la classe *Airplane*, les segments portant le label *engine* ne sont pas regroupés ensemble, parce que le contexte de ces segments sont différents : certains peuvent être attachés sur le devant du fuselage, sur le côté ou sur les ailes. Aussi, la position relative des différents groupements de segments donne des informations sur la relation entre les labels. Par exemple, dans la classe *Airplane*, le groupe correspondant au label *body* se trouve au centre des autres labels, tout comme le label *torso* pour la classe *Human*. De même, le groupe *lowerArm* se trouve à l'opposé du groupe *foot*.

Afin d'évaluer l'influence de la méthode de calcul du descripteur sur la description, nous avons réalisé l'expérimentation suivante : plutôt que d'échantillonner un ensemble de points pour chaque label, comme détaillé dans la section 3.1.2, nous considérons l'ensemble des points et visualisons, avec la méthode de positionnement multidimensionnel l'influence de ce changement. Comme nous pouvons le constater dans la figure 3.10b, en prenant tous les sommets de chaque segments, les regroupements sont plus compacts qu'avec la stratégie d'échantillonnage, comme on peut le voir dans la figure 3.10a. En revanche, la stratégie d'échantillonnage nous permet d'avoir une variabilité intra-classe, tout en conservant des groupes distincts.

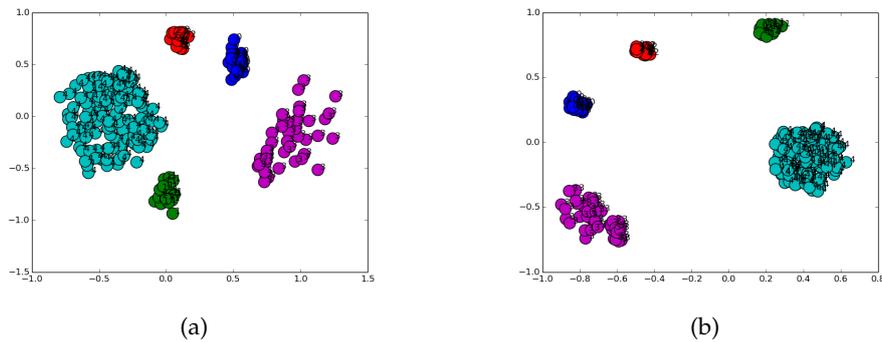


FIGURE 3.10 – Positionnement multidimensionnel des signatures de segments pour la catégorie Ant du Princeton Benchmark [16] - (a) En utilisant l'échantillonnage - (b) en utilisant tous les points labélisés.

Nous utilisons dans la section suivante ces signatures de segments afin de labelliser automatiquement un modèle en utilisant des techniques d'apprentissage supervisé.

3.2 LABELLISATION DE MODÈLES 3D À L'AIDE DU DESCRIPTEUR SÉMANTIQUE

Comme précisé dans la section 3.1.2, l'évaluation du descripteur nécessite une labellisation complète du maillage. Il paraît alors contre-intuitif d'utiliser ce descripteur pour une application de labellisation. Dans cette section, nous montrons qu'utiliser l'apprentissage supervisé ainsi qu'une technique d'optimisation combinatoire nous permet toutefois d'assigner un label à chaque partie d'un modèle segmenté (mais non labellisé).

Comme notre signature sémantique décrit la relation entre la géométrie et les labels dans un ensemble de modèles 3D, nous considérons la tâche de labellisation comme un problème inverse : plutôt que d'attribuer un label à chaque segment individuellement, nous cherchons la labellisation complète dont la probabilité étant donné l'ensemble des signatures de chaque segment est maximale.

3.2.1 La labellisation comme un problème inverse

Nous considérons un modèle 3D en entrée, noté \mathbf{Q} , composé de m segments non labélisés, obtenus à partir d'un algorithme de segmenta-

tion existant [93, 40]. En utilisant un ensemble de modèles labellisés et segmentés, nous pouvons inférer les labels des segments de \mathbf{Q} . La méthode est composée de deux étapes : l'apprentissage, pendant lequel un classifieur de type *random forest* est appris à partir de la base d'exemples déjà labellisés, et l'inférence, dans laquelle on estime la labélisation la plus probable à partir d'un modèle en entrée.

3.2.2 Apprentissage

Pour chaque modèle dans l'ensemble d'apprentissage, un ensemble de point S_ℓ est extrait pour chaque label $\ell \in \mathcal{L}$. Pour chaque point, le descripteur \mathbf{d} est calculé à partir de la labellisation vérité-terrain. Pour chaque segment, nous construisons un histogramme du descripteur. À partir de ces histogrammes, nous calculons la moyenne, la déviation standard, ainsi que les coefficients de *skewness* et *kurtosis*. Ces statistiques nous permettent de caractériser la distribution des distances géodésiques à chaque label pour un segment.

Nous obtenons donc un vecteur caractéristique pour chaque segment de taille $4 \times |\mathcal{L}|$. Si un label n'est pas représenté dans un modèle, les statistiques correspondantes dans le vecteur ne sont pas utilisées pour ce modèle.

À partir de cet ensemble d'apprentissage, nous entraînons un classifieur de type *random forest* multi-classe [8] qui donne en sortie pour un segment requête la probabilité d'appartenance à chacune des classes (les labels) étant donnés la signature du segment.

3.2.3 Inférence

Étant donné un modèle 3D composé de m segment, nous souhaitons le labelliser en utilisant le classifieur appris précédemment. Cependant, notre descripteur nécessite de définir un label pour chaque segment. Nous répondons au problème de labellisation en évaluant des *hypothèses* sur les labellisations possibles.

Pour chaque labellisation candidate, nous évaluons sa probabilité en utilisant le classifieur appris précédemment. Pour chaque segment s_i du

modèle en entrée, le classifieur nous donne la probabilité $P(\text{label}(s_i) = \ell)$ pour que le label ℓ soit assigné au segment s_i étant donné le descripteur du segment. Nous déterminons ensuite la probabilité de l'hypothèse complète de labellisation L pour tous les segments étant donnés leurs descripteurs, en multipliant ces probabilités.

$$P(L) = \prod_{i=0}^m P(\text{label}(s_i) = \ell) \quad (3.2)$$

Cette formulation considère que les labels sont indépendants. Cette hypothèse nous permet d'obtenir des résultats corrects. Si l'on ne considère pas que les labels soient indépendants, l'évaluation de la probabilité $P(L)$ ne pourrait pas être effectuée en un temps fini, puisque les descripteurs (et donc $P(\text{label}(s_i))$ pour chaque segment) dépendent des labels assignés aux autres segments.

Nous générons l'ensemble des labellisations possibles et nous conservons celle qui maximise la probabilité $P(L)$. Une approche de recuit simulé a été testée mais dégrade significativement le taux de labellisation correcte.

3.2.4 Résultats

Nous présentons les résultats de notre méthode de labellisation sur le *Princeton Segmentation Benchmark* [16] qui contient 19 catégories de 20 objets chacune, allant de modèles humanoïdes à des objets manufacturés. Nous utilisons la méthode de segmentation et la vérité terrain présentée par Kalogerakis *et al.* [35]. En pratique, il y a au total $|\mathcal{L}|^m$ labellisations possibles. Chaque catégorie contient en moyenne $|\mathcal{L}| = 5$ et $m = 7$ segments par objet, ce qui revient au total à environ 80000 évaluations). L'apprentissage se fait en 5 minutes sur une classe complète de 20 modèles. L'inférence prend entre 56 secondes (pour la classe *Glasses*) et 30 heures (pour la classe *Armadillos* qui a le plus grand nombre de labels et de segments), et le temps médian est de 18 minutes.

La parallélisation du code et de simples heuristiques sur la cardinalité des labels peuvent être utilisées pour réduire le temps d'inférence. Par exemple, nous observons un temps d'inférence divisé par trois pour

la classe *Bird* si l'on considère connue la cardinalité des labels (deux segments ailes, une queue, etc.), sans perdre en efficacité. Cette heuristique est valide si le maillage à labelliser est cohérent avec l'ensemble d'apprentissage.

Nous démontrons l'efficacité de la méthode en utilisant le protocole de validation croisée en *leave-one-out* pour chaque catégorie d'objet. Ces expérimentations supposent que la classe de l'objet est connue, mais cette restriction pourrait être levée, au détriment d'un temps d'inférence plus élevé. Dans le tableau 3.1, nous présentons le nombre de labels $|\mathcal{L}|$ considérés pour chaque catégorie et le pourcentage de facettes correctement labellisés par notre méthode. Nous présentons également pour comparaison, les résultats obtenus en considérant un descripteur géométrique simple (histogramme de *Shape Indices* [41] combiné avec l'aire du segment) et les résultats obtenus en considérant un grand vecteurs de caractéristiques géométriques, issu des travaux de Kalogerakis et al. [35]. Dans ce dernier cas, comme pour notre signature, nous fournissons en entrée du classifieur *random forest* la moyenne, la déviation standard, ainsi que les coefficients de *skewness* et de *kurtosis* des 608 caractéristiques unaires, ce qui résulte en un vecteur de taille 2432 pour chaque segment. Nous présentons aussi les résultats obtenus en combinant cet ensemble de caractéristiques géométriques et notre descripteur sémantique. Les résultats obtenus en utilisant cette combinaison sont les meilleurs ; en revanche, notre descripteur seul (qui a en moyenne 20 dimensions) offre de bons résultats malgré sa faible dimension.

Nous présentons dans la figure 3.11 les résultats de labélisation obtenus sur les classes *Airplane* et *Bird*. Sur chaque ligne, les derniers exemples ne sont pas labélisés correctement par rapport à la vérité terrain fournie par Kalogerakis *et al.* [35]. Pour l'un des avions, un segment est labélisé *stabilizer* plutôt que *rudder*, puisque ces deux éléments sont similaire d'un point de vue géométrique et d'un point de vue structurel.

En testant la labellisation sans échantillonner les points labélisés, mais en considérant tous les sommets, comme dans la figure 3.10, sur les classes *Bird*, *Fish*, *Plier* et *Teddy*, nous obtenons respectivement 82.3%, 78.5%, 59.0%

TABLE 3.1 – Résultats de labellisation sur le Princeton Segmentation Benchmark (vérité-terrain fournie par Kalogerakis et al. [35]). Pour chaque catégorie, nous montrons le nombre de labels et le pourcentage de labels corrects par facette en utilisant notre descripteur de dimension 20, puis des descripteurs géométriques simples (aire du segment et Shape Indices [41]), un vecteur de dimension 2432 contenant les moments des 608 caractéristiques géométriques de Kalogerakis et al. [35], et enfin en combinant notre descripteur avec ces caractéristiques géométriques. Combiner les informations géométriques et sémantiques offre les meilleurs résultats, mais notre descripteur seul offre des résultats raisonnables étant donné sa faible dimension.

Classe	# labels	notre méthode	[41]	[35]	notre méthode+[35]
Airplane	5	87.6	36.5	96.0	98.5
Ant	5	83.0	28.0	99.2	99.2
Armadillo	11	76.3	20.3	98.1	98.1
Bearing	6	57.1	2.0	85.2	84.1
Bird	5	87.5	3.8	95.4	94.0
Bust	8	65.4	2.6	68.1	67.7
Chair	4	59.0	1.7	99.7	99.7
Cup	2	92.1	87.5	95.1	95.1
Fish	3	82.4	78.2	100	90.5
FourLeg	6	80.0	25.0	97.8	97.7
Glasses	3	81.1	64.8	100	97.4
Hand	6	91.1	4.9	87.5	90.9
Human	8	79.3	13.2	92.1	92.0
Mech	5	75.9	14.9	84.7	87.6
Octopus	2	78.6	52.3	55.0	100
Plier	3	61.8	12.9	100	100
Table	2	84.3	89.2	97.8	97.8
Teddy	5	70.2	13.8	100	100
Vase	5	77.2	1.2	85.0	93.8
Moyenne	5	77.3	29.1	91.4	93.7

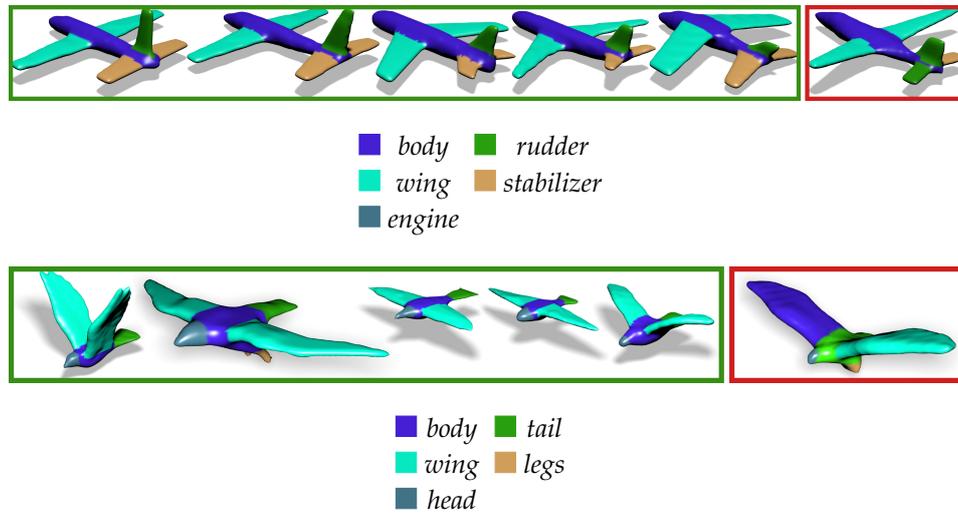


FIGURE 3.11 – Résultats de labélisation pour les classes Airplane et Bird du Princeton Benchmark. Les cadres verts (respectivement rouge) représentent les labellisations correctes (respectivement incorrectes)

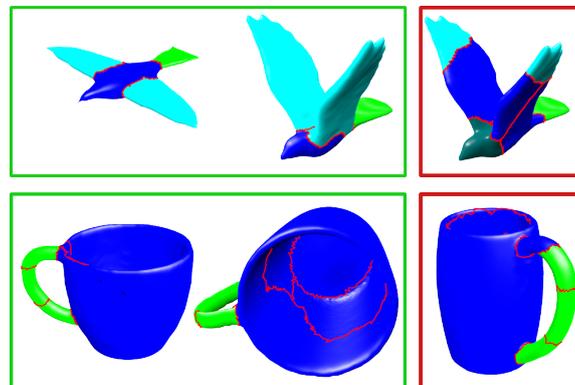


FIGURE 3.12 – Résultats de labélisation en utilisant une segmentation différente de l'ensemble d'apprentissage. Les cadres verts (respectivement rouge) représentent les labellisations correctes (respectivement incorrectes). En haut à gauche, les oiseaux sont sous-segmentés, la labellisation est correcte. En haut à droite, l'oiseau est sur-segmentés en 7 parties, ce qui a pour conséquence la mauvaise classification de certaines parties de l'aile. En bas, les tasses sont sur-segmentés en 7 parties, ce qui peut résulter en certaines parties de la poignée d'être mal labellisé. En général, la sous-segmentation ne pose pas de problèmes particuliers pour notre labellisation, tandis que certaines erreurs peuvent se produire avec la sur-segmentation.

et 68.1%. Ces résultats sont moindres qu'avec la stratégie d'échantillonnage, puisque notre descripteur n'est plus invariant à la résolution du maillage, et n'est pas adapté à l'aire de chaque segment.

Afin d'affirmer la robustesse de notre méthode par rapport aux erreurs de segmentation, nous utilisons notre technique pour labelliser des modèles sous et sur-segmentés par rapport à l'ensemble d'apprentissage (figure 3.12). Les segments labellisés *body* et *head* sont joints sur deux des oiseaux, et les deux ailes sont connectés sur les deux autres. Nous présentons aussi un autre modèle d'oiseau sur-segmenté, et 3 tasses segmentés en sept parties (uniquement deux segments dans l'ensemble d'apprentissage). Pour les maillages sous-segmentés, la labellisation automatique reste correcte dans la plupart des cas, tandis que la sur-segmentation provoque des erreurs de classification (colonne de droite).

3.3 TRAITEMENTS LOCAUX SUR LE MAILLAGE 3D

Dans cette section, nous présentons une utilisation de notre descripteur afin de modifier un maillage 3D selon sa sémantique, en y ajoutant des détails géométriques ou en le filtrant. La nature continue de notre descripteur et les qualités sémantiques de la métrique associée nous permettent d'obtenir des modifications convaincantes. Nous présentons ici trois scénarios d'usage :

1. L'utilisateur spécifie un label (par exemple *upper leg*) et les modifications géométriques sont appliquées sur la totalité de la région pour laquelle d_ℓ à la plus petite valeur dans le descripteur. Sur le bord de la région, les modifications sont moindres puisque la valeur du descripteur correspondant au label augmente.
2. L'utilisateur sélectionne un point sur le maillage, et les modifications géométriques sont appliquées autour de ce point et décroissent selon la distance L_2 dans l'espace du descripteur.
3. L'utilisateur sélectionne un point sur le maillage et une isoligne (de la métrique L_2). Les modifications sont alors appliquées sur toute la région bordée par cette isoligne. Une fois encore, l'amplitude des

modification diminue quand on s'éloigne de l'isoligne sélectionnée (dans l'espace du descripteur). Grâce aux isolignes, l'utilisateur peut facilement sélectionner une région sémantique, qui n'est pas définie préalablement dans la labélisation de départ, comme l'épaule dans le cas d'un modèle humanoïde.

Dans ces trois scénarios, nous avons utilisé une fonction Gaussienne $\mathcal{N}(\mu = 0, \sigma^2)$ pour l'atténuation des traitements, avec pour paramètre $\sigma^2 = 50$ dans le cas de la distance L^2 dans l'espace du descripteur et $\sigma^2 = 30$ lorsque l'on utilise qu'une seule dimension du descripteur. Dans tous ces scénarios, les distances sont normalisées selon la taille du maillage.

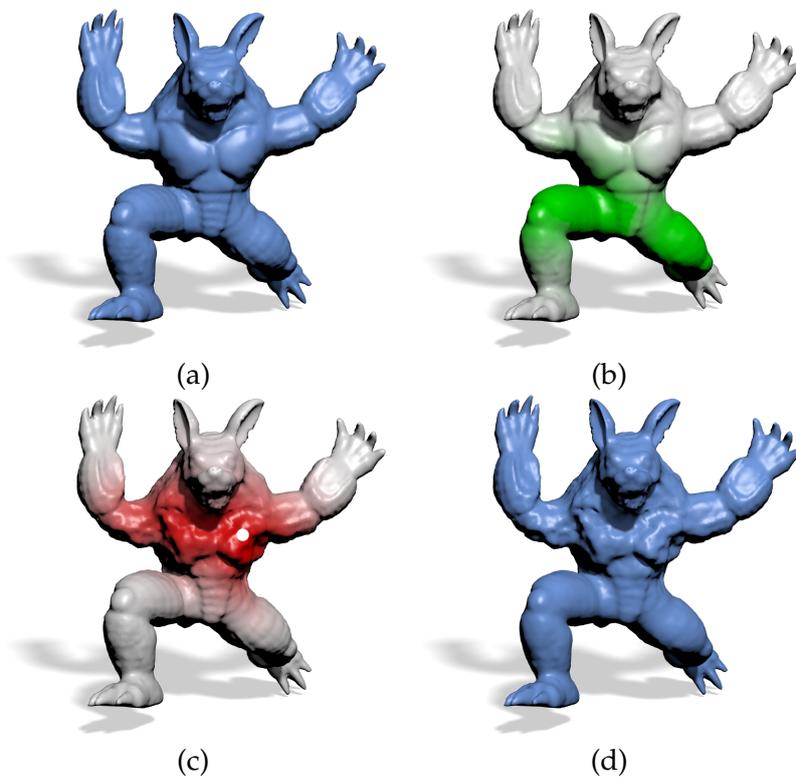


FIGURE 3.13 – Application à l'édition de maillages - (a) Le modèle Armadillo de référence - (b) Lissage Laplacien en sélectionnant le label upperleg - (c) Bruit procédural ajouté en sélectionnant un point (en blanc) sur le torse - (d) modèle final après lissage et addition d'un bruit procédural. L'augmentation de la distance dans l'espace du descripteur sémantique implique une diminution progressive des modifications géométriques (mis en évidence par la saturation de la couleur).

La figure 3.13 présente les scénarios 1 et 2. Dans le premier cas, un bruit de Perlin est ajouté sur le torse du modèle Armadillo, et la cuisse est lissée. Pour l'opération de lissage, nous sélectionnons le label *upper leg*

(scénario 1). Nous n'utilisons donc qu'une seule dimension de notre descripteur : nous lissons toute la zone pour laquelle la plus petite dimension correspond à ce label. L'amplitude du lissage est représentée par l'opacité de la couleur verte pour plus de visibilité.

Pour l'ajout de bruit, nous sélectionnons le point affiché en blanc (scenario 2). Le bruit est alors ajouté autour de ce point dans l'espace du descripteur selon la distance L_2 . Le bruit procédural est appliqué sur toute la région du torse. Les autres régions, telle que la tête ou les mains restent intactes. La figure 3.13 (d) illustre le modèles après ces deux modifications.

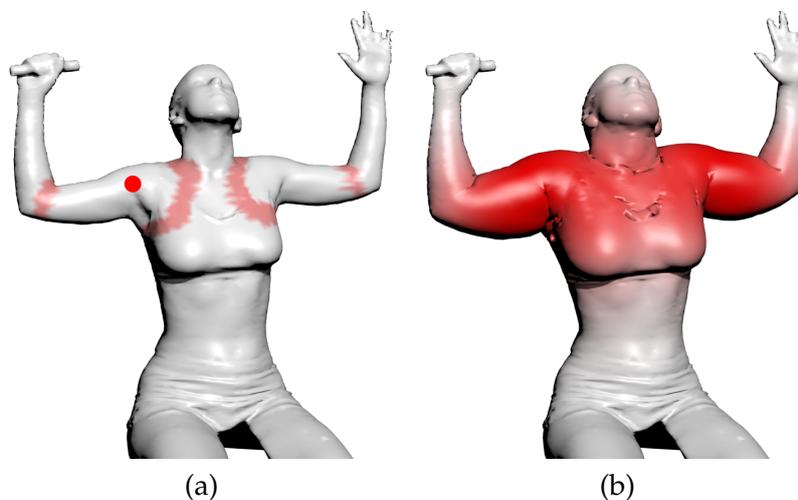


FIGURE 3.14 – Application à l'édition de maillages - (a) Modèle humanoïde de référence - un point (en rouge) est sélectionné sur l'épaule du modèle de référence, ainsi qu'une isoligne (en rouge) - (b) Les sommets à l'intérieur des isolignes sont extrudés avec une transition douce au bords (illustrée par la saturation de la couleur). L'édition est symétrique grâce à la symétrie de la labellisation.

Le scénario 3 est illustré dans la figure 3.14. Le point en rouge est sélectionné sur l'épaule du modèle humanoïde, ainsi que l'isoligne affichée en rouge. Nous extrudons ensuite tous les sommets à l'intérieur de cette isoligne. Puisque les distances sont calculées dans l'espace du descripteur, les sommets sur l'autre épaule sont automatiquement extrudés ; cette propriété de symétrie provient directement de l'utilisation d'information sémantique.

3.4 CONCLUSION

Dans ce chapitre, nous avons présenté un nouveau descripteur sémantique continu. Cette manière inédite de décrire la sémantique d'un modèle 3D se calcule simplement, en évaluant les distances géodésiques en chaque sommet à partir de points sources portant un label connu. Cette signature nous permet de décrire la nature continue de la sémantique d'un modèle ainsi que sa topologie (les relations entre les différentes parties du modèle). Nous avons illustré dans ce chapitre les bénéfices apportés par une telle représentation avec deux applications directes : la labellisation automatique de modèle et les traitements géométriques locaux.

La tâche de labellisation automatique est formulée comme un problème inverse en choisissant la labellisation de probabilité maximale, fournie par un classifieur *random forest* après une étape d'apprentissage sur un ensemble de modèle de labellisation connue.

L'application d'édition de maillage nous permet d'effectuer de manière automatique un traitement géométrique simple (telle que l'addition d'un bruit procédural ou le lissage Laplacien) d'une manière sémantiquement cohérente.

Dans le chapitre suivant, nous présenterons une application de ce descripteur pour la mise en correspondance de modèle. En effet, comme présenté dans la section 3.1.3, le descripteur sémantique nous permet d'identifier des zones sémantiquement similaires sur un modèle à partir d'une labellisation grossière du modèle 3D.

CORRESPONDANCE ET SÉLECTION

À L'AIDE DU DESCRIPTEUR

SÉMANTIQUE

SOMMAIRE

4.1	INTRODUCTION	56
4.1.1	Outils de segmentation interactive	56
4.1.2	Recherche partielle de maillages	57
4.1.3	Détection de patches similaires	59
4.2	MÉTHODE	62
4.2.1	Vue d'ensemble	62
4.2.2	Initialisation	63
4.2.3	Minimisation d'énergie	65
4.2.4	Classification sur la base	66
4.3	EXPÉRIMENTATIONS	66
4.3.1	Sélection d'une zone à l'intérieur d'un segment	67
4.3.2	Sélection d'une zone entre plusieurs segments	68
4.3.3	Sélection de zones de tailles différentes	69
4.3.4	Influence de la résolution sur la sélection	70
4.3.5	Discussion	71
4.4	CONCLUSION	73

4.1 INTRODUCTION

Dans la partie 2.3, nous avons présenté les méthodes de modélisation permettant d'exploiter une base de modèles exemples. Les méthodes présentées utilisent différents modes de requête afin de suggérer ou de retrouver des modèles complets ou des segments de modèles.

Nous avons présenté dans le chapitre précédent un descripteur sémantique pouvant être évalué en chaque point, définissant de manière continue l'appartenance d'un point d'une surface à un label. Ce descripteur multi-dimensionnel nous indique la distance (au sens de la distance géodésique) du point considéré à chaque ensemble de points labellisés, pour chaque label.

Nous présentons dans ce chapitre une application de ce descripteur pour la correspondance et la sélection de région dans une base de modèles. Dans un premier temps, nous revenons dans la section 4.1.1 sur les outils facilitant la définition d'une segmentation par l'utilisateur, puis nous donnons un aperçu des méthodes de recherche partielle de maillages dans la section 4.1.2. Enfin, nous détaillons les motivations pour une méthode permettant de définir une similarité par patches, et l'état de l'art dans ce domaine dans la section 4.1.3.

Dans le reste de ce chapitre, nous présentons la méthode de sélection dans la section 4.2 puis les résultats expérimentaux dans la section 4.3. Enfin nous concluons ce chapitre dans la section 4.4.

4.1.1 Outils de segmentation interactive

Les outils de sélection 3D permettent à un utilisateur de définir manuellement une segmentation binaire (intérieur/extérieur) d'un maillage. Ces outils sont particulièrement utiles pour des tâches d'édition, de traitement de maillages, ou pour définir une requête dans le cadre de la recherche partielle de maillage.

Dans le cadre d'images 2D, il est courant de voir des outils de sélection par *lasso* : l'utilisateur définit une courbe qui marque la frontière de la zone à sélectionner. Ce type d'interface est plus compliqué à mettre en œuvre dans le cas de maillages 3D, puisque cela nécessite de manipuler la ca-

méra, afin de pouvoir observer la surface dans son intégralité et résoudre les problèmes d'occlusions.

Différents outils permettent de faciliter cette tâche : l'utilisateur dessine alors une frontière grossière, incomplète, qui sera complétée par un algorithme. Nous avons déjà cité la méthode de découpage intelligent de Funkhouser *et al.* [26], et d'autres méthodes sont apparues plus récemment [96, 76] afin de découper un maillage à l'aide d'un trait permettant de découper la surface en deux parties. Une étude comparative de ces méthodes de découpage interactif de maillages a été réalisée par Fan *et al.* [22]. Ces méthodes diffèrent par le mode d'interaction utilisé : dans certaines méthodes, l'utilisateur doit tracer un trait pour définir deux régions (avant et arrière-plan), tandis que dans d'autres, il définit la frontière entre ces deux zones soit en traçant un trait le long de cette frontière, ou en travers, comme on peut le voir dans la figure 4.1.

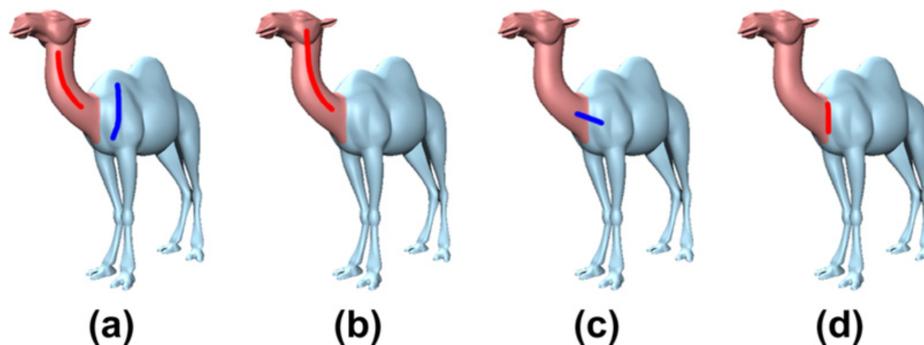


FIGURE 4.1 – Interfaces de segmentations interactives, extrait de Fan *et al.* [22] - (a) sélection avant/arrière plan [33] - (b) sélection de l'avant plan [21] - (c) dessin en travers de la frontière [96] - (d) dessin le long de la frontière [55].

Ces outils de segmentation interactive permettent de définir manuellement une zone à éditer sur le maillage, ou une requête pour un algorithme de recherche partielle, que nous verrons dans la section suivante.

4.1.2 Recherche partielle de maillages

Au delà de la segmentation interactive, afin de ne pas nécessiter d'intervention de l'utilisateur, il est intéressant d'avoir un outil permettant de retrouver les régions similaires à une requête dans une collection de maillages. Les méthodes de recherche partielle de maillages ont été très

grandement étudiées ces dernières années. Plusieurs études complètes sur ces méthodes ont été réalisées [50, 81].

L'étude de Liu *et al.* [50] définit le problème de recherche partielle de maillage et compare les différentes méthodes de l'état de l'art. Deux types d'applications sont possibles : la recherche de partie dans un ensemble, qui consiste à identifier si un maillage en requête est contenu dans un autre, et la mesure de similarité par partie, qui consiste à établir une mesure de similarité entre deux formes complètes, selon les parties qui les composent. Cette décomposition en parties n'est pas nécessairement connue.

Liu *et al.* classifient les méthodes de recherche partielle de maillages en trois catégories : les méthodes utilisant des descripteurs locaux, les méthodes utilisant une segmentation, et les méthodes basées sur des vues 2D. Ces méthodes diffèrent principalement selon la définition des parties (selon qu'il y ait une segmentation ou non), la correspondance entre ces parties, leur caractérisation, l'organisation de ces caractéristiques, et la mesure de similarité partielle.

Brièvement, les méthodes utilisant les descripteurs locaux utilisent une collection de descripteurs calculés sur un ensemble de points de la surface, chaque descripteur ne décrivant qu'un petit voisinage autour du point considéré. La similarité partielle est évaluée en estimant s'il existe un sous-ensemble de point de la surface possédant un ensemble de descripteurs similaires à la requête. Ce type de méthode a l'avantage de ne pas nécessiter une segmentation préalable. Les descripteurs locaux utilisés incluent les *Spin Images* [34], les harmoniques sphériques [38], les fonctions de chaleur [77] ou les mesures de courbure locale. Une fois ces descripteurs locaux calculés sur l'ensemble de la base, la similarité partielle entre deux formes peut-être évaluée de plusieurs façon : par comparaison au plus proche voisin [34, 53], par comparaison des points saillants uniquement [46], avec une stratégie multi-échelle [2], par une approche itérative type ICP [9], ou avec une approche de quantification vectorielle [72] ou basée sac-de-mots [24, 10, 45].

Les méthodes utilisant une segmentation quant à elles, décomposent en premier lieu chaque modèle en un ensemble de segments, qui ont un

sens sémantique. L'accent ici est mis sur la *fonctionnalité* de chaque partie, c'est-à-dire son contexte relativement aux autres segments. Typiquement, les segments sont organisés selon un graphe. Il peut s'agir alors d'un graphe de connectivité [52], d'un graphe de Reeb [82], d'un squelette [18], d'un arbre binaire [49] ou d'une structure hiérarchique [15]. La mesure de similarité partielle est alors évaluée par isomorphisme de sous-graphe. Afin d'éviter cette procédure qui peut-être coûteuse, certaines méthodes n'utilisent pas de graphe mais une structure de sac-de-mot pour collecter les signatures des segments [83].

Enfin, les méthodes basées sur des vues 2D utilisent un ensemble de vues du modèle 3D, uniformément distribuées. Ce type de méthode permet d'utiliser une requête 2D. Cela réduit la recherche de formes 3D à un problème de recherche sur des images 2D. Les descripteurs utilisés sont alors des descripteurs traditionnellement utilisés pour la recherche d'images : coefficients de Fourier [14, 63], descripteurs SIFT [59], filtres de Gabor [20], *etc.* Ce sont ces méthodes qui sont exploitées dans les méthodes de modélisation à partir d'esquisse, que nous avons déjà évoquées dans les sections 2.2.2 et 2.3.3.

L'étude de Liu *et al.* donne des détails supplémentaires sur les invariances de chaque méthode, la robustesse par rapport au bruit et les limitations diverses.

4.1.3 Détection de patches similaires

Les méthodes de recherche partielle permettent d'identifier les maillages *contenant* une requête mais ne permettent pas d'identifier les points similaires à la requête à la surface de ces maillages. Parallèlement, plusieurs méthodes s'attachent à étudier les similitudes à la surface d'un maillage unique. Maximo *et al.* [54] proposent une méthode permettant d'appliquer un traitement identique sur des sommets dont les voisinages sont similaires.

Leifman et Tal [47] répondent à un problème similaire de coloriage de maillage : étant donnée une zone sélectionnée à la surface du maillage par l'utilisateur, ce système permet de colorier tous les motifs similaires. Cette

recherche de motifs similaires est traitée par une approche de classification de sommets.

Rodola *et al.* [68] proposent une méthode de correspondance partielle basée sur une analyse spectrale de l'opérateur de Laplace-Beltrami. Cette méthode permet d'identifier dans un maillage complet les sommets correspondants à un maillage partiel. Cependant, cette méthode est évaluée uniquement sur des paires de maillages où l'un est une déformation quasi-rigide de l'autre. De plus, le temps de mise en correspondance est de cinq minutes entre une requête partielle et un modèle complet, ce qui rend cette méthode inadaptée pour mettre en correspondance une requête donnée avec l'ensemble des modèles de la base en temps réel.

Enfin, Guy *et al.* [29] proposent différents outils de sélection et d'extension adaptés. Ces outils permettent de simplifier la sélection de composantes connexes, de parties (définies par des traits à la surface du maillage), ou des patches (définis à l'aide d'un outil de *painting* adapté). L'outil d'extension permet de trouver les k sélections potentiellement similaires, où le nombre de correspondants k est donné par l'utilisateur. Brièvement, nous expliquons le principe d'extension, uniquement dans le cas des patches, puisque c'est le cas de sélection le plus général qui nous intéresse ici. Le principe d'extension des patches est illustré dans la figure 4.2.

Dans un premier temps, l'utilisateur sélectionne une zone de référence, visible dans la figure 4.2(a). Après avoir extrait le centre du patch, une carte de similarité, d'après une variante du *Shape Context* [42], est calculée. Les k points les plus similaires constituent l'ensemble des centres des patches candidats. Il reste alors à définir le bord de chaque patch candidat. Cette étape est traitée en classifiant les sommets voisins de chaque centre à l'aide d'un classifieur SVM binaire. Les sommets inclus dans la sélection de référence constituent l'ensemble d'apprentissage et chacun est caractérisé par des mesures définies dans un repère local au patch. Ce repère local étant essentiel pour la définition de ces caractéristiques, le repère de référence et les repères locaux sont recalés par ICP. Ces deux étapes (classification par SVM et recalage par ICP) sont toutes les deux complé-

mentaires et permettent d'avoir des voisinages correspondants de bonne qualité, comme illustré dans les figures 4.2(a,b,c).

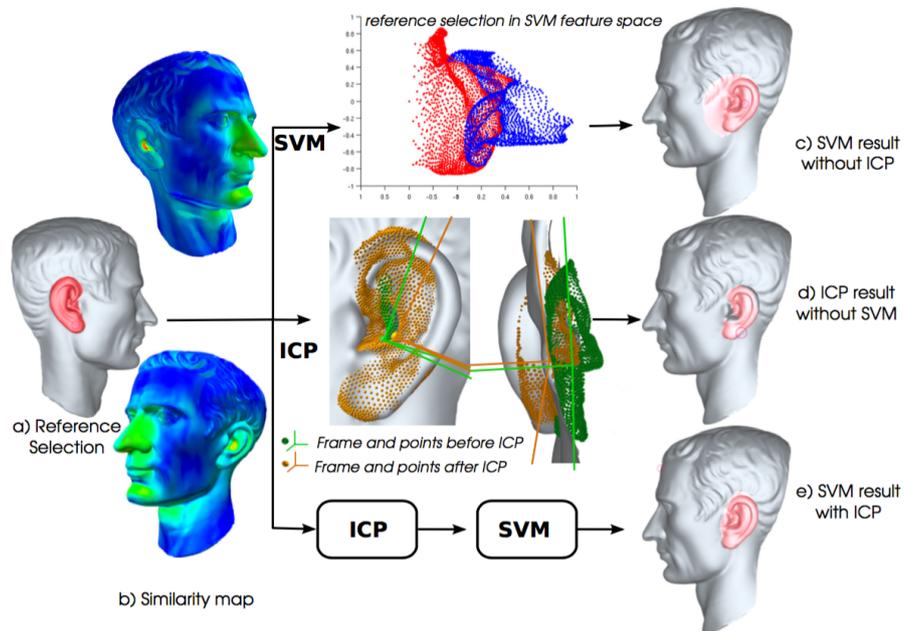


FIGURE 4.2 – Illustration de l'extension de patches, extrait de SimSelect [29] - (a) sélection du patch de référence - (b) carte de similarité, relativement au centre du patch - (c) résultat du SVM sans ICP - (d) résultat de ICP sans SVM - (e) SVM et ICP (résultat final).

Ces méthodes permettent de définir sur un maillage unique les zones présentant des similarités. En revanche, elles ne sont pas adaptées pour fonctionner sur une collection de maillages. Les méthodes de correspondances denses [62] permettent de définir un correspondant pour chaque sommet d'un maillage sur un autre. Cependant, ces méthodes prennent uniquement en compte l'information géométrique et pas l'information sémantique.

Notre but est d'exploiter l'information géométrique et sémantique à l'aide du descripteur introduit précédemment, afin de détecter les patches similaires à une requête sur l'ensemble de la base. Nous expliquons dans la section 4.2 le principe de la méthode.

4.2 MÉTHODE

4.2.1 Vue d'ensemble

L'objectif de la méthode est de pouvoir sélectionner un patch à la surface du maillage, et de retrouver les patches similaires, non seulement dans le maillage lui-même, de manière analogue à *SimSelect* [29], mais parmi une collection de modèles. Nous supposons que cette collection de modèles est correctement labélisée, en utilisant par exemple l'une des nombreuses méthodes de co-labellisation existantes [30, 75, 85].

Pour l'ensemble des modèles que nous considérons, nous calculons en chaque sommet le descripteur sémantique présenté dans le chapitre 3.

En entrée du processus, l'utilisateur choisit un modèle 3D et sélectionne une zone de référence à l'aide d'une brosse. Les sommets sélectionnés sont mis en évidence. Le système va alors automatiquement traiter l'ensemble des modèles de la base afin de mettre en évidence les zones correspondantes. La figure 4.3 illustre les différentes étapes du processus de sélection.

Notre objectif est de sélectionner des zones similaires dans un ensemble de modèles variés. Il s'agit ici de sélectionner les zones qui ont un contexte similaire à la partie de référence dans le maillage. Les méthodes telles que celles de Leifman et Tal [47] considèrent des détails géométriques fins sur un seul modèle. Ici, nous nous intéressons plus au contexte sémantique d'un sous-ensemble de sommets du maillage. Pour cette raison, nous considérons le descripteur sémantique présenté précédemment.

Pour un point p , le descripteur sémantique $\mathbf{d}(p)$ est multidimensionnel, et représente la distance géodésique vers un point de chaque label de référence. Nous modélisons le contexte sémantique d'une sélection comme l'étendue de ces distances selon chaque label. Cette représentation revient à rechercher une ellipse dans un espace de grande dimension (le nombre de labels de référence), qui décrit au mieux la sélection.

Afin de trouver les paramètres de cette ellipse, nous extrayons dans un premier temps la sélection de référence, et calculons une première ini-

tialisation. Puis, la minimisation d'une fonction d'énergie par descente de gradient permet de trouver les paramètres de l'ellipse.

Une fois ces paramètres obtenus, nous pouvons classifier chacun des sommets de chaque modèle pour retrouver les zones correspondantes.

4.2.2 Initialisation

Une ellipse dans un espace de grande dimension est définie par l'équation suivante :

$$\sum_{i=0}^n \frac{(x_i - p_i)^2}{a_i^2} = 1 \quad (4.1)$$

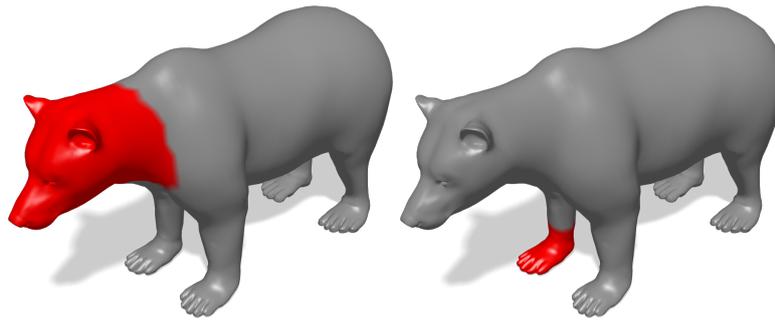
où $\{a_i\}_{i=1..n}$ sont les paramètres (axes) de l'ellipse et $p = \{p_i\}_{i=1..n}$ le centre.

Les paramètres de l'ellipse correspondant à la sélection sont calculés par minimisation d'une fonction d'énergie. Pour ce faire, nous utilisons une méthode de minimisation par descente de gradient, qui dépend fortement de l'initialisation. Il nous faut donc choisir des valeurs de départ pour les paramètres $\{a_i\}_{i=1..n}$ qui ne soient pas trop éloignés des valeurs finales.

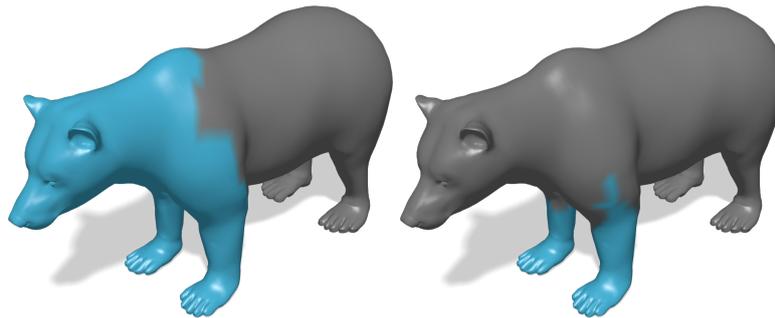
Nous utilisons pour l'initialisation une hypersphère H dans l'espace du descripteur, définie de la manière suivante :

$$\begin{aligned} \forall s \in H, \|\mathbf{d}(p) - \mathbf{d}(s)\|_2 &< r \\ r &= \|\mathbf{d}(p) - \mathbf{d}(q)\|_2 \end{aligned} \quad (4.2)$$

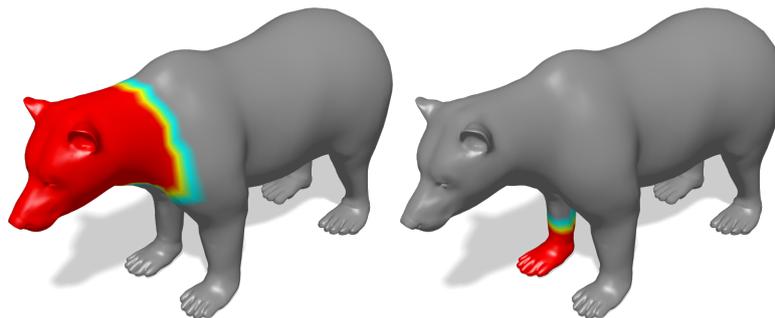
où s est un sommet du maillage, p est le centre de la sélection dans l'espace du descripteur, q est le sommet le plus éloigné de p parmi la sélection, et \mathbf{d} est le descripteur sémantique que nous avons présenté dans le chapitre 3. La figure 4.3a illustre cette initialisation. Les sommets dont le descripteur se situe à l'intérieur de l'hypersphère sont représentés en bleu. On voit sur la figure 4.3b que l'utilisation du descripteur nous permet de prendre en compte les symétries.



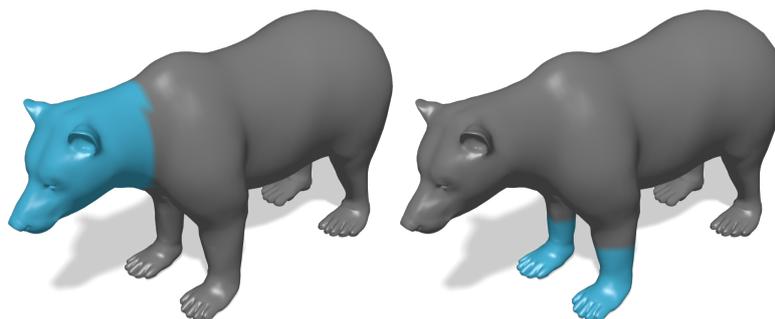
(a) Requête



(b) Initialisation



(c) K-voisinage



(d) Sélection finale

FIGURE 4.3 – Procédé de sélection et de correspondance pour deux requêtes. - (a) : spécification du patch de référence pour une partie de la tête, et une partie de la patte. - (b) : initialisation de la sélection à l'aide d'une hypersphère. Les sommets dont le descripteur se situe dans l'hypersphère sont en bleu clair, et font partie de la sélection initiale. - (c) : Le k -voisinage défini autour du patch de référence pour $k = 3$. La sélection initiale est rappelée en rouge, le 3-voisinage est représenté en jaune, vert et cyan. (d) : Classification finale sur le modèle de référence avec l'ellipse. Les sommets dont le descripteur se situe dans l'hypersphère sont en bleu clair, et font partie de la sélection finale.

4.2.3 Minimisation d'énergie

Afin de trouver les paramètres de l'ellipse qui correspondent au mieux à la sélection de référence S , nous minimisons la fonction d'énergie suivante :

$$e = \sum_{s \in \hat{S}} \mathbb{1}_s \left(\sqrt{\sum_{i=0}^n \frac{(d_i(p) - d_i(s))^2}{a_i^2}} - 1 \right)^2 \quad (4.3)$$

où $d_i(s)$ représente la i^{eme} composante du descripteur sémantique évalué sur le sommet s . Le terme d'énergie est évalué pour tous les sommets s à l'intérieur de la sélection de référence, et à l'intérieur du k -voisinage autour de ce patch, comme illustré dans la figure 4.3c. La fonction indicatrice $\mathbb{1}_s$ nous permet d'évaluer la concordance entre l'appartenance à la sélection et l'appartenance à l'intérieur de l'ellipse. Si un sommet est à la fois à l'intérieur du patch et si son descripteur se trouve à l'intérieur de l'ellipse, alors $\mathbb{1}_s = 1$. De même, si le sommet est en dehors du patch (mais dans le k -voisinage) et que son descripteur se situe en dehors de l'ellipse alors $\mathbb{1}_s = 0$. Dans tous les autres cas, $\mathbb{1}_s = 0$ et la contribution à la fonction d'énergie n'est pas évaluée pour ce sommet. Cette méthode nous permet d'éviter d'obtenir une ellipse trop large, qui contiendrait bien tous les sommets du patchs mais aussi un nombre important de faux-positifs. Le terme $\left(\sqrt{\sum_{i=0}^n \frac{(d_i(p) - d_i(s))^2}{a_i^2}} - 1 \right)^2$ donne plus d'importance aux sommets dont le descripteur est proche des bords de l'ellipse.

L'utilisation d'un k -voisinage nous permet d'avoir un ensemble d'exemples négatifs, ne faisant pas partie du patch pour évaluer les paramètres de l'ellipse. La figure 4.3c illustre ce voisinage pour les deux sélections considérées précédemment.

La minimisation de la fonction d'énergie est réalisée en utilisant l'algorithme COBYLA [65] qui supporte une minimisation par estimation de la dérivée, puisque la fonction d'énergie n'a pas d'expression analytique.

Les paramètres finaux de l'ellipse nous permettent de classifier les sommets selon leur descripteurs. Les figures 4.3d illustre cette classification à l'aide de l'ellipse. Les sommets dont le descripteur sémantique d se trouve à l'intérieur de l'ellipse sont illustrés en bleu. On remarque

dans la figure 4.3d que le descripteur nous permet de prendre en compte la symétrie. Par rapport à l'initialisation à l'aide de l'hypersphère, illustrée précédemment dans la figure 4.3b, l'ellipse nous permet d'avoir une sélection bien plus proche de la requête, illustrée dans la figure 4.3a (en prenant en compte la symétrie).

4.2.4 Classification sur la base

Une fois les paramètres de l'ellipse calculés, il devient possible de trouver les zones similaires à la sélection de référence en évaluant, pour chaque sommet de chaque modèle s'il se situe à l'intérieur de l'ellipse ou non. Le point est retenu si le descripteur \mathbf{d} est tel que $\sum_{i=0}^n \frac{(d_i - p_i)^2}{a_i^2} < 1$. Sinon, le point ne fait pas partie d'un patch correspondant. Les résultats de classification pour différents scénarios sont présentés dans la section 4.3.

4.3 EXPÉRIMENTATIONS

Dans cette section, nous présentons différents résultats de correspondance de sélection, obtenus de manière automatique grâce à notre méthode détaillée dans la section 4.2.

Pour présenter ces résultats, nous considérons un ensemble de modèles composé d'humanoïdes, obtenus grâce au logiciel *MakeHuman*¹ et un ensemble de quadrupèdes, obtenus sur le site *Turbosquid*². Pour chacun de ces modèles, le descripteur sémantique décrit dans le chapitre 3 est évalué en chacun des sommets. La labellisation de départ utilisée est constituée de quatre labels, correspondant à la tête, au corps, aux bras et aux jambes (pattes avant et arrière pour les quadrupèdes), comme illustré dans la figure 4.4.

Nous présentons maintenant différents scénarios de sélection et de correspondance afin d'illustrer les possibilités de correspondance offertes par la méthode décrite précédemment dans la section 4.2.

1. www.makehuman.org/

2. www.turbosquid.com/

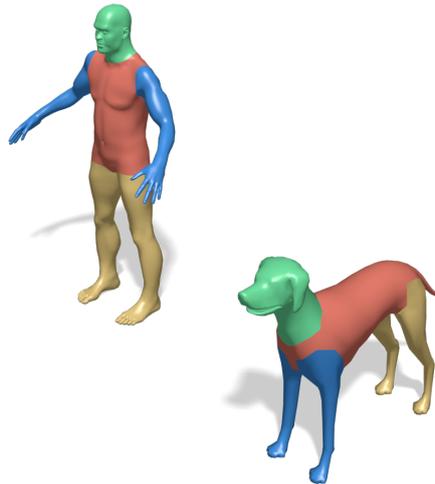


FIGURE 4.4 – Présentation de la labellisation utilisée pour la correspondance - tête (vert), corps (orange), bras (bleu), jambes (jaune).

4.3.1 Sélection d'une zone à l'intérieur d'un segment

La segmentation originale, utilisée pour calculer le descripteur sémantique d définit une zone pour le corps, la tête, les bras et les jambes. Dans le scénario suivant, on sélectionne la queue d'un modèle de chien, illustré dans la figure 4.5a. Si l'on compare avec la segmentation originale, présentée dans la figure 4.4, la queue n'est pas présente et correspond plutôt à une zone située à l'intérieur du segment labélisé *corps*.

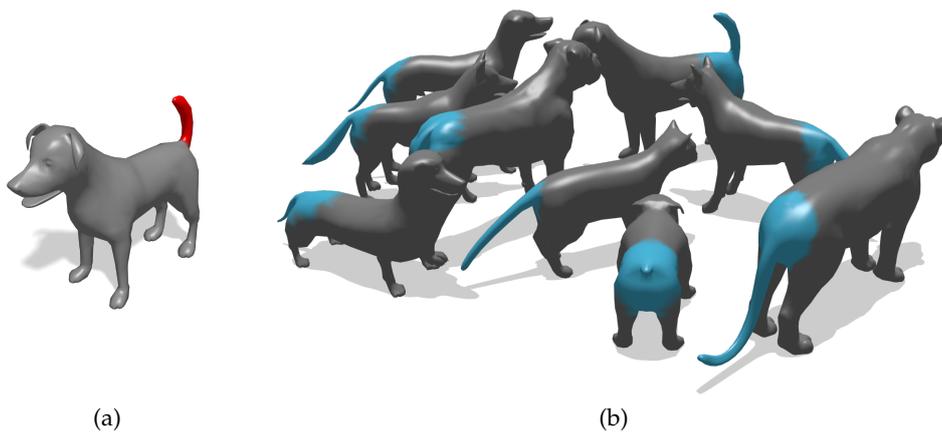


FIGURE 4.5 – (a) Sélection d'une zone à l'intérieur d'un segment, ici la queue du chien, en rouge - (b) Résultats de correspondance sur la queue de quadrupède. Les zones correspondantes sont en bleu.

La figure 4.5b présente les résultats de correspondance sur la queue de quadrupèdes, en bleu. L'utilisation du descripteur sémantique et la modélisation par une ellipse permettent d'identifier parmi ces différents

modèles les zones correspondant à la queue. Le descripteur sémantique permet de situer le contexte sémantique de la sélection, en mesurant la distance géodésique à chaque label considéré. On peut noter toutefois que sur certains modèles, la sélection est plus large et contient certains sommets appartenant au dos du quadrupède.

De la même manière, on sélectionne une zone correspondant au coude sur un modèle humanoïde. La labellisation d'origine ne dispose pas de ce niveau de sémantique, mais le descripteur d nous permet d'effectuer une telle sélection sur un ensemble de modèles. La sélection est présentée dans la figure 4.6a.

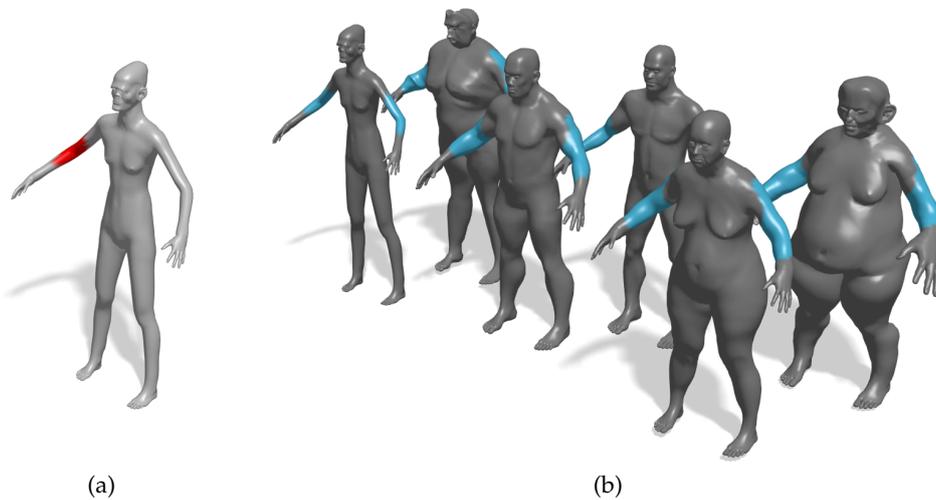


FIGURE 4.6 – (a) Sélection d'une zone à l'intérieur d'un segment, ici le coude, en rouge - (b) Résultats de correspondance sur le coude d'un modèle humanoïde. Les zones correspondantes sont en bleu.

La figure 4.6b présente les résultats de correspondance pour la sélection du coude, illustrée par la figure 4.6a. La zone sélectionnée sur l'ensemble des modèles correspond bien au coude et ce, malgré le fait que la labellisation de départ ne contienne pas ce niveau de description sémantique. À noter aussi que la description étant symétrique, les coudes droits et gauches sont sélectionnés, même avec une requête non-symétrique.

4.3.2 Sélection d'une zone entre plusieurs segments

Dans ce scénario, on souhaite sélectionner l'épaule d'un modèle humanoïde. Cette zone intermédiaire se situe entre trois segments dans la

labellisation originale (corps, bras, tête). La sélection est illustrée dans la figure 4.7a.

La requête présentée ici est située entre plusieurs labels. Dans le cas d'un système de modélisation à partir d'exemples tels que ceux présentés dans la section 2.3.4, la labellisation contraint la sélection. Dans notre cas, il ne serait possible de sélectionner que le bras, le corps, ou la tête, mais pas une zone située entre ces différents segments.

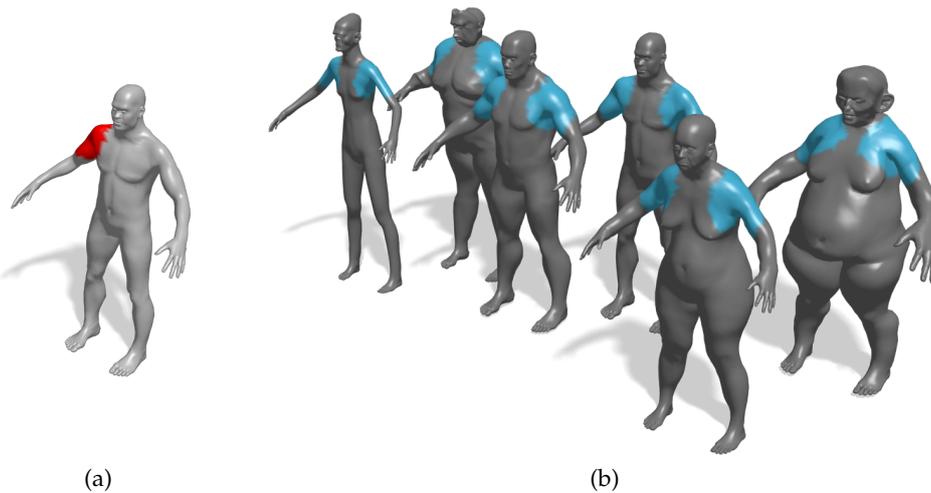


FIGURE 4.7 – (a) Sélection d'une zone correspondant à l'épaule représentée en rouge, se trouvant entre plusieurs labels - (b) Résultats de correspondance sur l'épaule d'un modèle humanoïde. Les zones correspondantes sont en bleu.

La figure 4.7b présente les résultats de correspondance pour la requête illustrée dans la figure 4.7a. Les zones sélectionnées sont bien situées dans une région intermédiaire entre le bras, le corps et la tête.

4.3.3 Sélection de zones de tailles différentes

Dans ce scénario, nous souhaitons comparer deux sélections, appartenant à des zones similaires mais de tailles différentes. Il s'agit ici de sélectionner la patte d'un animal, soit de manière partielle, soit de manière complète. Les sélections correspondant à ce scénario sont représentées dans la figure 4.8.

La figure 4.9 illustre les différents résultats de correspondance obtenus pour ces deux requêtes. On observe dans la figure 4.9a que les sélections sont faites uniquement sur la partie inférieure de la patte, tandis que toute

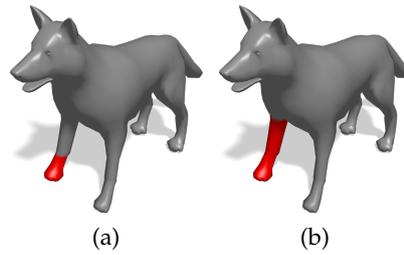


FIGURE 4.8 – Sélection de zones de tailles différentes - (a) sélection du pied - (b) sélection de la patte complète.

la patte est sélectionnée dans la figure 4.9b, ce qui montre que la méthode permet de contrôler l'étendue de la requête.

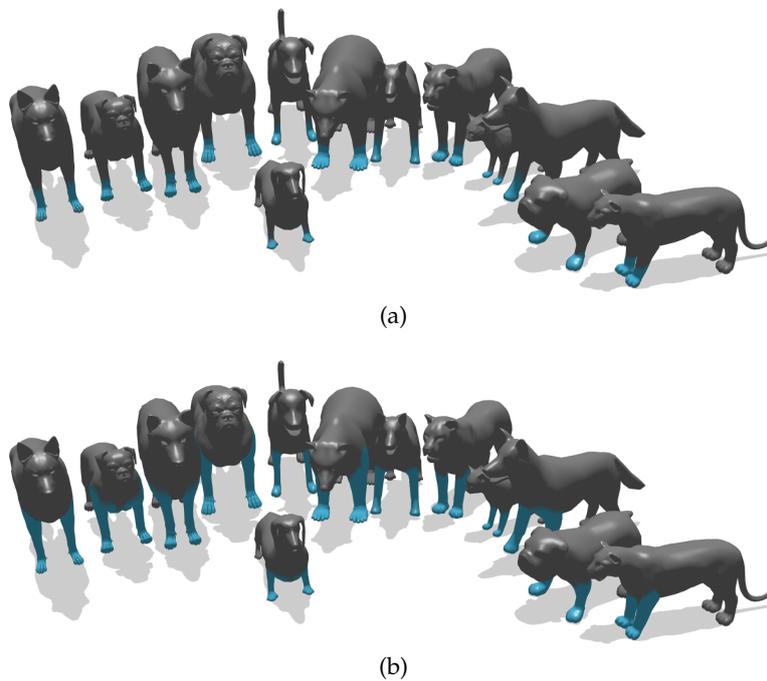


FIGURE 4.9 – Résultats de correspondance sur des zones de tailles différentes - (a) sélection du pied - (b) sélection de la patte complète. Les zones sélectionnées sont en bleu.

4.3.4 Influence de la résolution sur la sélection

Dans ce scénario, nous considérons un modèle, et nous souhaitons sélectionner une zone sur une version de plus haute résolution. En sélectionnant le centre de la face du modèle basse résolution, il est possible d'obtenir la zone correspondante sur le modèle haute résolution, comme illustré dans la figure 4.10. On observe qu'il est possible grâce à cette mé-

thode d'obtenir une correspondance entre un modèle basse résolution et sa version haute résolution.

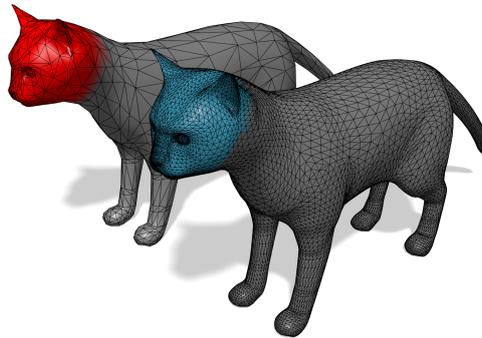


FIGURE 4.10 – La sélection est effectuée sur une version basse résolution du modèle (en rouge à gauche) et la correspondance est illustrée sur la version haute résolution (en bleu à droite).

4.3.5 Discussion

Afin de situer la méthode présentée dans ce chapitre par rapport à l'état de l'art, nous avons choisi de comparer notre méthode avec une approche similaire à celle utilisée pour l'expansion des patches dans *SimSelect* [29]. Une fois une première sélection effectuée, on recherche dans chaque modèle les k sommets les plus proches du centre du patch. Les sommets dans leur voisinage sont classifiés.

Nous utilisons un classifieur *Support Vector Machine* (SVM) avec un noyau Gaussien. Comme dans *SimSelect*, le paramètre γ du noyau Gaussien est choisi comme étant la médiane des distances entre un sommet et ses voisins. Les caractéristiques utilisées sont les mêmes. D'autres résultats sont présentés en combinant ces caractéristiques avec le descripteur sémantique dans l'apprentissage.

Comme nous pouvons le voir dans la figure 4.11a, les résultats obtenus par classification avec un SVM sont moins précis que ceux obtenus avec l'ellipse. Ce manque de précision est dû aux caractéristiques utilisées, qui sont moins discriminantes : dans le cas de *SimSelect* [29], les modèles utilisés sont des modèles de plus haute résolution, et la classification est effectuée sur un seul et même modèle. Aussi, le choix du paramètre γ du noyau Gaussien peut-être adapté pour un maillage, mais pas nécessairement pour tous les maillages de la base. L'ensemble d'apprentissage est

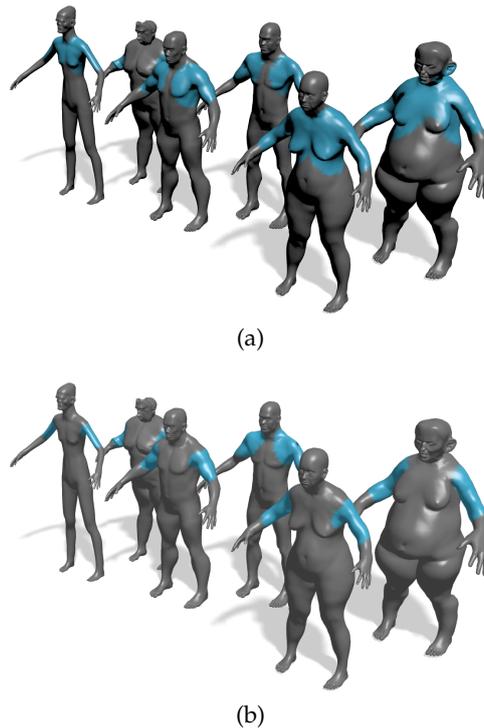


FIGURE 4.11 – Classification avec un SVM - (a) en utilisant les caractéristiques de *SimSelect* [29] - (b) en ajoutant le descripteur sémantique. Les facettes classifiées comme faisant partie de la sélection sont en bleu. La requête est celle présentée dans la figure 4.7a.

différent pour chaque requête, et une technique de recherche itérative de paramètres avec un ensemble de validation n'est pas compatible avec la contrainte de temps réel.

Les résultats obtenus en combinant les descripteurs de *SimSelect* [29] avec le descripteur sémantique d sont présentés dans la figure 4.11b. On constate que le descripteur sémantique améliore la précision de la sélection. La méthode de sélection par ellipse présente néanmoins d'autres avantages en matière de temps d'exécution, et de facilité d'utilisation : en utilisant la classification avec SVM, il est nécessaire de préciser un nombre k de candidats correspondants. Là où la spécification de k était adaptée pour la recherche dans un seul modèle, le nombre de correspondants a priori n'est pas connu lors de la recherche dans une base complète.

D'un point de vue du temps d'exécution, le tableau 4.1 revient sur les différents temps d'apprentissage et de tests pour les différentes requêtes présentées dans cette section.

L'apprentissage du SVM dans les expérimentations présentées dans les

TABLE 4.1 – Nom, nombre de sommets, temps d'apprentissage et temps moyen de tests pour les requêtes présentées dans cette section.

requête (figure)	sommets	apprentissage (ms)	moyenne test (ms)
Patte ours (4.3a)	11419	85	2
Tête ours (4.3a)	11419	924	2
Queue (4.5a)	1479	6	0.3
Coude (4.6a)	13426	3	2
Épaule (4.7a)	13459	7	3
Patte partielle (4.8a)	1246	7	0.2
Patte complète (4.8b)	1246	9	0.2
Resolution (4.10)	1086	6	0.2

figures 4.11a et 4.11b est réalisé dans les deux cas en 0.05 secondes et le temps de test moyen pour un modèle est de 0.2 secondes dans le cas où l'on considère les trois caractéristiques de *SimSelect*, et de 0.53 secondes dans le cas où l'on considère aussi le descripteur sémantique. Ces temps de tests nettement supérieurs peuvent s'expliquer, entre autre, par le fait que les trois caractéristiques (d, h, α) considérés par sommet dans *SimSelect* ne peuvent pas être pré-calculés, contrairement au descripteur sémantique d .

Les résultats obtenus dans *SimSelect* [29], sur des modèles de plus haute résolution, pour l'extension de patches varient entre 7.21 secondes pour le modèle Goro (82000 sommets) et 18.117 secondes pour le modèle Armchair (102000 sommets).

4.4 CONCLUSION

Dans cette section, nous avons présenté une nouvelle méthode de correspondance et de sélection dans une base de modèles 3D. Cette correspondance est basée sur le contexte sémantique d'un sommet du maillage vis-à-vis d'une labellisation connue.

Comme nous l'avons illustré dans différents scénarios dans la section 4.3, la sélection peut alors se faire indépendamment des labels considérés. Nous avons discuté dans la section 4.3.5 des avantages de notre approche par rapport à la méthode de *SimSelect* [29], permettant d'effectuer ce type de sélection sur un modèle unique, en terme de précision de sélection et de temps d'exécution.

La méthode présentée dans ce chapitre nous permet, en sélectionnant une partie d'un modèle de trouver dans une base de modèles un ensemble de régions similaires. Dans le contexte d'une application de modélisation à partir d'exemples, retrouver des parties partiellement similaires n'est qu'une des problématiques en jeu pour remplacer un segment par un autre. Il est nécessaire d'adapter la géométrie des différentes parties pour pouvoir obtenir un modèle final cohérent. Dans le prochain chapitre, nous étudierons les contraintes présentes en terme d'adaptation géométrique pour l'assemblage des parties.

ADAPTATION ET ASSEMBLAGES DE MAILLAGES 3D

SOMMAIRE

5.1	INTRODUCTION	76
5.1.1	Motivations	76
5.1.2	Méthodes de compositions de maillages	76
5.2	MÉTHODE	78
5.2.1	Principe général	78
5.2.2	Recalage Semi-rigide	79
5.2.3	Définition des régions de jonction S_a et S_b	81
5.2.4	Mise en correspondance des régions S_a et S_b	82
5.2.5	Définition des voisinages supports	82
5.2.6	Calcul et application de la transformation	83
5.3	RÉSULTATS	84
5.4	CONCLUSION	84

5.1 INTRODUCTION

5.1.1 Motivations

Nous avons présenté dans le chapitre 2 les méthodes de modélisation à partir d'exemples : ces méthodes visent à exploiter un ensemble hétérogène de modèles afin de créer de nouveaux modèles par assemblage de parties extraites de modèles existants.

Une première problématique est l'exploration des différentes parties composant les modèles de la base. Dans le chapitre 4, nous avons présenté une méthode permettant de retrouver les segments sémantiquement similaires à une requête. Pour une tâche de remplacement de segments, il est nécessaire cependant d'adapter les différentes géométries avant la tâche d'assemblage. Ces adaptations sont relatives au placement, à la mise à l'échelle, mais aussi à une déformation locale des deux géométries.

Ce chapitre est organisé de la manière suivante : dans un premier temps, nous reviendrons sur les différentes méthodes de l'état de l'art en matière de composition de maillages. Dans la suite de ce chapitre, nous nous intéresserons à une de ces méthodes, particulièrement adaptée aux formes organiques et aux modèles dont la segmentation n'est pas naturelle.

5.1.2 Méthodes de compositions de maillages

Les méthodes d'assemblage de maillages permettent de créer de nouveaux modèles par composition. Ce type d'outil est indispensable pour la modélisation à partir d'exemples. Dans de nombreux cas, la modélisation à partir d'exemples ne vise pas à obtenir une composante connexe unique, et l'assemblage est effectué en juxtaposant les points de contacts pré-calculés sur les parties [90, 32, 23, 95, 4, 1].

Dans le cas du mélange de modèles présenté par Alhashim *et al.* [1], un modèle généré peut être déterminé comme incohérent et supprimé si un contact présent dans les formes source et cible n'est pas conservé. Dans d'autres cas, le placement des différentes parties est obtenu par optimisation, soit à l'aide d'un système masse-ressort [32], soit par le calcul d'une

transformation rigide (translation, rotation, mise à l'échelle) permettant de mettre en correspondance les points de contacts pré-calculés sur les différentes composantes [23, 95].

Ce type d'assemblage est particulièrement adapté pour les modèles industriels directement segmentés en plusieurs composantes, mais pas pour les formes organiques telles que les humanoïdes, où une segmentation naturelle n'est pas directement disponible. Dans ce cas, plutôt que de considérer un modèle comme un assemblage de composants associés à des points de contacts, on considère une surface unique. Les différents composants, une fois segmentés sont alors des surfaces possédant un ou plusieurs bords, et il est nécessaire d'établir une stratégie afin d'obtenir une surface unique.

Dans le cas des méthodes d'assemblage de segments, différentes stratégies peuvent être utilisées. Kanai *et al.* [37] décrivent une méthode simple pour connecter deux surfaces ouvertes. Cette méthode est entre-autre utilisée par Funkhouser *et al.* [26] pour composer plusieurs maillages dans leur application de modélisation par l'exemple. Brièvement, le principe est le suivant : étant donné deux surfaces ouvertes, les deux contours C_1 et C_2 sont joints en mettant en correspondance leurs sommets, de proche en proche, selon la distance Euclidienne. Une fois les sommets mis en correspondance, un ensemble de nouvelles arrêtes est rajouté au maillage pour relier les deux surfaces. La figure 5.1 illustre ce principe.

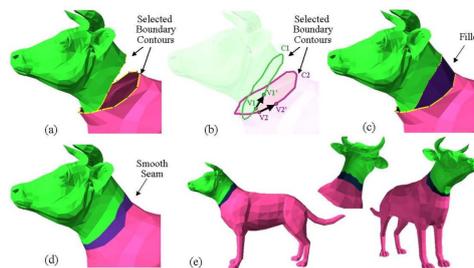


FIGURE 5.1 – Création d'une jonction entre deux surfaces par ajout de géométrie, extrait des travaux de Funkhouser *et al.* [26] - (a) Sélection des contours - (b) Sélection des deux points les plus proches V_1 et V_2 et orientation des contours - (c) Ajout de géométrie entre les sommets correspondants - (d) Lissage local (e) Résultat final.

Cette méthode permet de simplement connecter deux bords proches, mais ne remplit pas toute les contraintes propres à une application de

modélisation et nécessite une manipulation manuelle préalable. En effet, dans le cas général, il est nécessaire de positionner correctement les deux maillages l'un par rapport à l'autre avant leur assemblage, de les mettre à l'échelle, voire de les déformer localement, afin que la transition entre les deux surfaces soit la plus lisse possible.

La méthode *SnapPaste* de Sharf *et al.* [74] propose de fusionner deux maillages en obtenant ce type de transition lisse à l'aide d'un recalage semi-rigide.

Notre objectif est de fusionner deux maillages, de manière non rigide, sans avoir à remailler la totalité du maillage final, pour que le procédé soit rapide. Pour cela, seule une partie des deux maillages sera transformée de manière non-rigide, à la différence de la méthode présentée par Sharf *et al.* [74]. Dans la suite de ce chapitre, nous donnons plus de détails sur la méthode de recalage semi-rigide, et sur son application dans une méthode de modélisation à partir d'exemples.

5.2 MÉTHODE

5.2.1 Principe général

Les modèles présents dans une base utilisée pour la modélisation à partir d'exemples peuvent constituer un ensemble très hétérogène : ils peuvent présenter des différences significatives en matière d'échelle, de résolution ou d'orientation. Une méthode de recalage semi-rigide est donc plus appropriée pour adapter les deux géométries, qu'une méthode plus simple d'ajout de facettes.

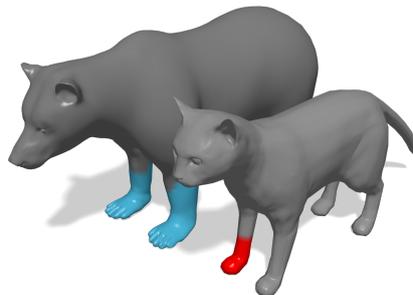


FIGURE 5.2 – Exemple de deux modèles à adapter - en rouge, la requête sur le modèle de chat, et à droite les zones correspondantes sur le modèle d'ours.

La figure 5.2 présente un problème type : la patte du modèle de chat est mise en correspondance avec l'ours. Les deux modèles possèdent une résolution différente (1086 sommets pour le modèle complet de chat, contre 11419 pour le modèle d'ours), et les échelles des deux parties sont aussi différentes.

Nous souhaitons minimiser l'intervention de l'utilisateur avant cette opération d'assemblage. Pour cette raison, le placement, l'orientation, et la mise à l'échelle de la partie à ajouter doit être faite de manière automatique. Pour cela, nous utilisons un recalage rigide par ICP (*Iterative Closest Point*) - dont nous reviendrons sur le principe dans la section 5.2.2 - et une mise à l'échelle entre la partie à remplacer et la nouvelle. Le résultat de cette opération est visible dans la figure 5.3.

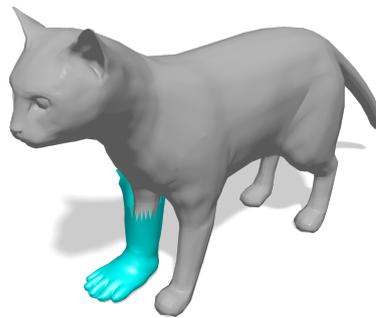


FIGURE 5.3 – Résultat après le recalage global par ICP. La patte de l'ours en bleu est recalée sur la patte du chat à remplacer.

Comme nous pouvons le constater, la position, l'orientation et l'échelle de la patte d'ours, recalée par ICP sur la patte du chat sont correctes. Néanmoins, il reste à déformer localement les deux géométries pour rapprocher les deux surfaces avant leur assemblage. Le recalage semi-rigide va nous permettre d'obtenir une zone de transition lisse entre ces deux surfaces.

5.2.2 Recalage Semi-rigide

L'algorithme ICP original permet de trouver une transformation rigide pour recalcr deux nuages de points, selon le critère des moindres carrés. La procédure pour recalcr deux nuages de points C_a et C_b est présentée par le pseudo-code 1.

À chaque itération de l'algorithme, une unique transformation T est

Algorithme 1 : ICP : recalage de C_a sur C_b

```

tant que  $erreur < seuil$  et  $iter < itermax$  faire
  pour  $p_a \in C_a$  faire
    | trouver le correspondant  $p_b \in C_b$ ,  $p_b = \Phi(p_a)$ 
    |  $T \leftarrow$  la transformation alignant au mieux  $C_b$  et  $\Phi(C_b)$  au sens
    | des moindres carrés pour  $p_a \in C_a$  faire
    | | Transformer  $p_a$  à l'aide de la transformation  $T$ 

```

calculée pour transformer tous les points du nuage C_a . Le processus itératif s'arrête quand l'erreur de recalage est en dessous d'un seuil déterminé ou que le nombre d'itération maximal est atteint.

Afin d'effectuer un recalage semi-rigide entre deux surfaces C_a et C_b , une transformation unique T_i est calculée pour chaque point p_i en recalant un support local dans le voisinage $N(p_i)$ du point sur son correspondant sur l'autre surface $\Phi(N(p_i))$. Il n'est pas nécessaire d'effectuer la mise en correspondance sur tous les points des deux maillages : en effet la jonction des deux surfaces ne se fait que localement au niveau des bords. Nous définissons donc dans un premier temps deux régions de jonction S_a et S_b . L'algorithme permettant de superposer ces deux parties de surfaces est présenté par le pseudo-code 2.

Algorithme 2 : soft-ICP : recalage semi-rigide de C_a sur C_b , selon [74]

```

tant que  $erreur < seuil$  et  $iter < itermax$  faire
  pour  $p_i \in S_a$  faire
    | trouver le point correspondant sur  $S_b$ ,  $\Phi(p_i)$ 
  pour  $p_a \in C_a$  faire
    | calculer le voisinage de  $p_i$ , noté  $N(p_i)$ 
    |  $T_i \leftarrow$  la transformation alignant au mieux  $N(p_i)$  et  $\Phi(N(p_i))$ 
  pour  $p_a \in C_a$  faire
    | appliquer la transformation  $T_i$  à  $p_i$ , de manière partielle, en
    | fonction de l'itération.
  À la fin de l'itération, échanger  $C_a$  et  $C_b$ .

```

Le principe est le suivant : les deux voisinages de deux points voisins s'intersectent. Ainsi, les transformations calculées pour ces points voisins seront proches. Un point éloigné aura un voisinage disjoint, et la transformation calculée ne sera donc pas nécessairement proche.

Les sections suivantes donnent les détails des différentes étapes de l'al-

gorithme tel qu'il est défini par Sharf *et al.* [74] ainsi que des modifications apportées pour notre application.

5.2.3 Définition des régions de jonction S_a et S_b

Afin de rapprocher les bords des deux surfaces, il n'est pas nécessaire de les mettre en correspondance dans leur intégralité, mais uniquement sur les deux sous-régions proches des bords que l'on souhaite rapprocher, que l'on note S_a et S_b . Ces sous régions sont représentées en rouge dans la figure 5.4.

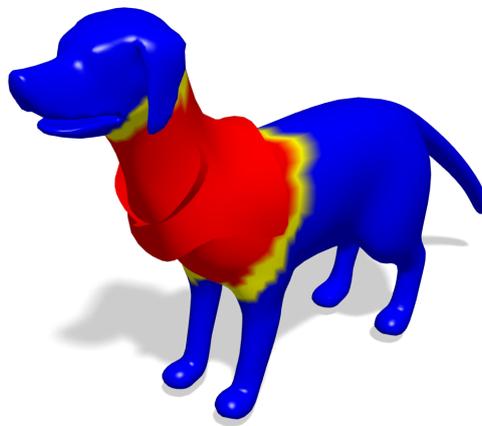


FIGURE 5.4 – Illustration des zones de jonction. Les deux régions de jonction sont représentées en rouge. Les sommets rouges et jaunes sont déplacés par le recalage semi-rigide. Les sommets en bleu sont déplacés par une transformation rigide à chaque itération.

Une fois le positionnement global des deux maillages obtenus par ICP, les deux régions de jonctions S_a et S_b sont identifiées comme étant les deux zones à une distance géodésique R du bord de chaque maillage. Si l'un des maillages présente plusieurs bords, les deux bords les plus proches sont considérés. La taille R de la région est définie comme étant la distance géodésique entre le point de S_a le plus proche du bord de S_b (au sens de la distance euclidienne), et le bord de S_a .

Dans la figure 5.4, les deux régions de jonction sont représentées en rouge. Les sommets rouges et jaunes sont déplacés par le recalage semi-rigide, tandis que les sommets en bleu sont déplacés par une transformation rigide à chaque itération.

5.2.4 Mise en correspondance des régions S_a et S_b

La position des points étant modifiée à chaque itération, la correspondance Φ entre les deux régions S_a et S_b est réévaluée. Pour chaque point $p_a \in S_a$, son correspondant est le point p_b de S_b minimisant la distance suivante :

$$D(p_a, p_b) = \omega_1 \|p_a - p_b\| + \omega_2 \text{Arccos}(N_a \cdot N_b) \quad (5.1)$$

Cette distance correspond à la somme pondérée de la distance euclidienne entre les deux points et à l'angle entre leurs normales respectives. Nous utilisons les poids suivants : $\omega_1 = 0.7$ et $\omega_2 = 0.3$.

En recalant S_a sur S_b , seulement les points de S_a sont déplacés. Pour que le processus soit symétrique, S_a et S_b sont échangés à chaque itération.

5.2.5 Définition des voisinages supports

Le principe du recalage semi-rigide est de calculer, à chaque itération une transformation par point. Le but est d'obtenir une déformation élastique du maillage. Pour cela, le calcul de la transformation n'est basé non pas sur l'intégralité du maillage comme dans l'algorithme ICP, mais sur un petit voisinage local défini pour chaque point p_i que l'on note $N(p_i)$. Nous définissons ce voisinage local autour de p_i comme son voisinage géodésique, contrairement à Sharf *et al.* [74] qui construisent un voisinage à l'aide d'une structure accélératrice arborescente (*GeoTree*). Comme nous utilisons uniquement une sous-région du maillage, il est moins coûteux dans notre cas de calculer directement le voisinage géodésique, plutôt que de construire la structure arborescente par clustering récursif.

Le rayon r_i dépend de l'itération et de la distance de p_i au bord : plus le point est éloigné du bord, plus le rayon sera grand et plus la transformation sera rigide. Pour un point donné, plus les itérations progressent, plus le rayon sera petit. La valeur du rayon est définie par l'équation suivante :

$$r_i = \begin{cases} R \cdot e^{[(k \cdot (t/t_{max})/d_i)^2]} & \text{si } p_i \in S_a \\ d(S_a, p_i) + R \cdot e^{[(k \cdot (t/t_{max})/d_i)^2]} & \text{sinon} \end{cases} \quad (5.2)$$

où t est le numéro de l'itération courante, t_{max} est le nombre maximal d'itérations. Le terme d_i est la distance géodésique entre p_i et le bord du maillage considéré. Dans le cas où p_i n'appartient pas à la zone de jonction S_a , le rayon est augmenté de $d(S_a, p_i)$, qui représente la distance géodésique de p_i à S_a . Ceci garantit un nombre suffisant de points inclus dans S_a dans le voisinage local $N(p_i)$, nécessaire pour calculer la transformation.

5.2.6 Calcul et application de la transformation

Une fois le voisinage géodésique de p_i défini, la transformation T_i recalant le voisinage support $N(p_i)$ sur son correspondant $\Phi(N(p_i))$ est calculée comme expliqué par la suite. La rotation est obtenue par une décomposition SVD : $\mathcal{R} = \text{argmin}(\sum_i \|R \cdot x_i - y_i\|)$, où x_i et y_i sont respectivement $N(p_i)$ et $\Phi(N(p_i))$ recentrés sur l'origine. Le facteur d'échelle s est calculé comme le rapport des normes de Frobenius des deux matrices x et y . Enfin, la translation est calculée par alignement des centres de masse des deux ensembles $N(p_i)$ et $\Phi(N(p_i))$: $t = \mu_N - s \cdot \mathcal{R} \cdot \mu_{\Phi(N)}$.

À chaque itération, une fois la transformation T_i calculée pour chaque point p_i , elle est appliquée au point de manière partielle avec le facteur $li = t/t_{max}$, de manière à ce que les deux surfaces se rapprochent l'une de l'autre progressivement avec les itérations. La figure 5.5 illustre l'évolution des deux maillages avec chaque itération.

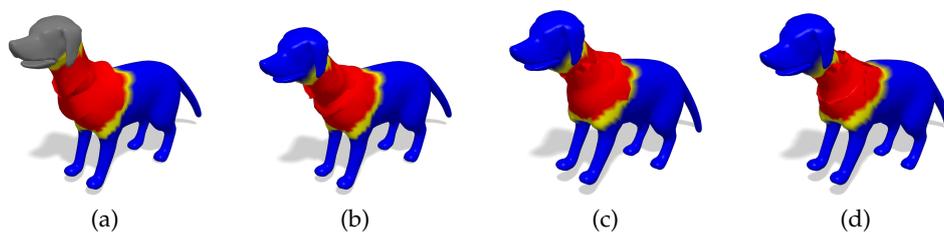


FIGURE 5.5 – Évolution du recalage semi-rigide, suivant 4 itérations. À chaque itération, seul l'un des deux maillages est modifié. Les deux surfaces sont plus proches lors de la quatrième itération (figure 5.5d), les deux zones de jonction seront ensuite remaillées pour obtenir une surface lisse, comme visible dans la figure 5.6b.

Une fois que les deux surfaces sont rapprochées à l'aide du recalage semi-rigide, les deux zones de jonction S_a et S_b sont remaillées à l'aide

d'un algorithme de reconstruction, sélectionnant les facettes de manière séquentielle à partir d'une segmentation de Delaunay [17].

5.3 RÉSULTATS

Nous présentons maintenant dans la figure 5.6 plusieurs exemples de modèles obtenus par assemblage de différentes parties d'autres modèles. L'assemblage final est obtenu après un recalage global par ICP, une déformation semi-rigide à l'aide de l'algorithme décrit précédemment, et un remaillage de la zone de jonction.

5.4 CONCLUSION

Dans ce chapitre, nous avons explicité les différentes contraintes en matière d'adaptation géométrique de maillage pour la modélisation à partir d'exemples. Les méthodes de recalage semi-rigide, telles que celle présentée par Sharf *et al.* [74] sont particulièrement adaptées car elles prennent en compte à la fois un recalage global (rotation, translation et mise à l'échelle), mais aussi une déformation élastique du maillage, permettant d'adapter les deux surfaces avant le remaillage. Ce type d'outil est primordial pour une application de modélisation à partir d'exemples impliquant des modèles organiques, où une optimisation de points de contacts ne serait pas adaptée. Avec l'outil de requête sur la base présenté dans le chapitre 4, nous avons ainsi un système complet d'aide à la création de modèles.

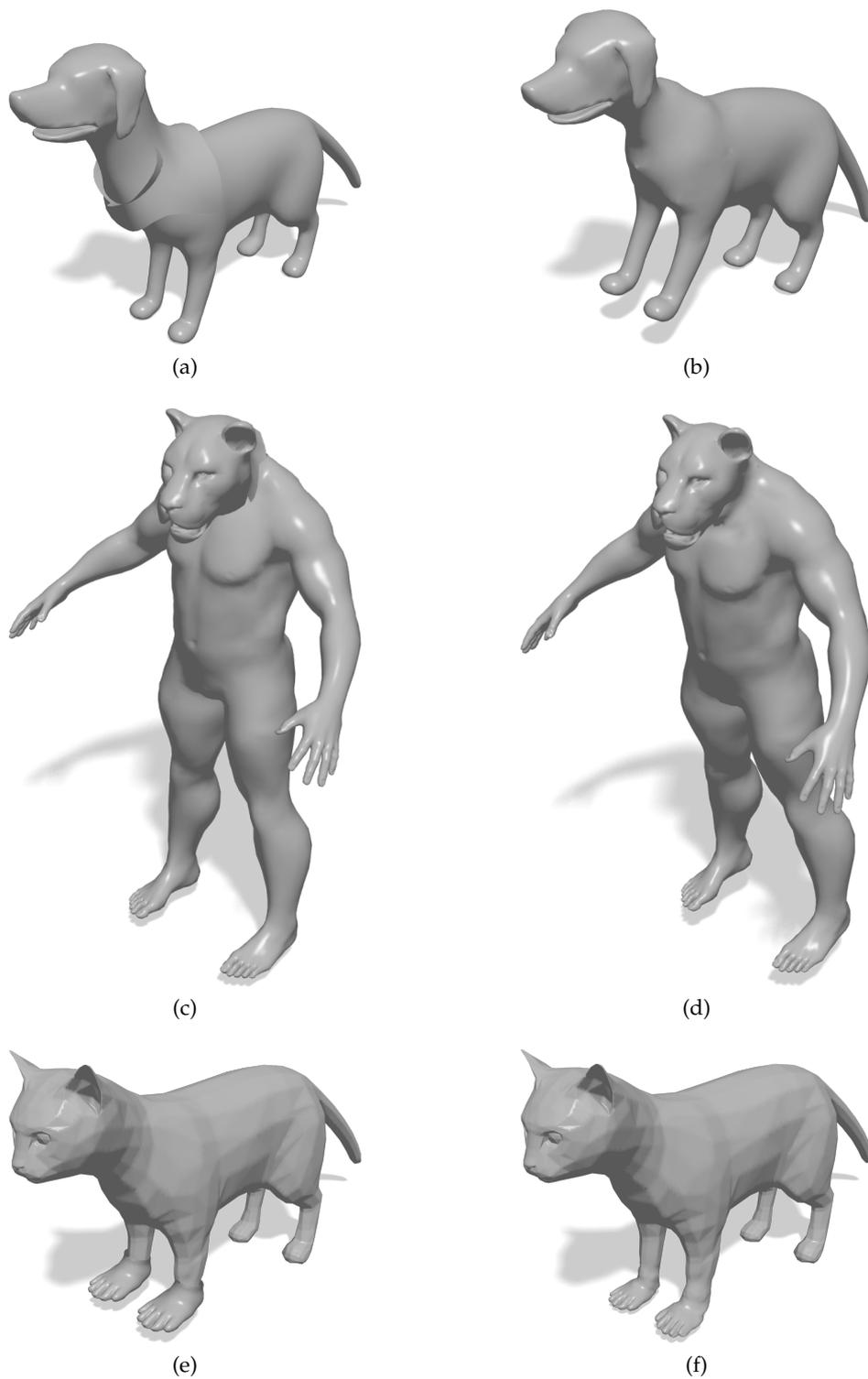


FIGURE 5.6 – Exemples d’assemblages de parties par soft-ICP. À gauche, les parties sont présentées avant remaillage, à droite, le modèle final après adaptation et remaillage de la zone de jonction.

CONCLUSION

6

SOMMAIRE

6.1	RÉSUMÉ	88
6.2	LIMITATIONS ET PERSPECTIVES	89

6.1 RÉSUMÉ

Dans cette thèse, nous avons proposé une nouvelle approche de modélisation 3D à partir d'une base d'exemples. Comme nous l'avons exposé, les méthodes les plus récentes de modélisation à partir d'exemples utilisent un paradigme d'assemblage de parties de modèles pré-segmentés. Ces méthodes permettent la création rapide par ajout ou échange de segments, mais sont contraintes par le domaine sémantique de la labellisation de la base.

Afin de se libérer de ces contraintes, plutôt que de considérer directement une segmentation avec une labellisation fixe, assignant un label unique à chaque segment, nous avons proposé une nouvelle description sémantique plus riche, basée sur une première labellisation, nous permettant de décrire le contexte sémantique de chaque point de la surface relativement à chacun des labels.

Cette description sémantique continue possède des propriétés intéressantes en matière d'invariance, et nous avons présenté différentes applications directes. Nous avons notamment présenté une méthode de labellisation automatique de segments exploitant le descripteur avec une approche d'apprentissage supervisé, utilisant un classifieur *Random Forest*.

Dans le contexte de la modélisation à partir d'exemples, nous avons utilisé ce descripteur afin de proposer une méthode de correspondance et de sélection de patches sémantiquement similaires à une région de référence, parmi une base de modèle. Cet outil permet de simplement échanger deux parties de modèles, indépendamment de la segmentation utilisée pour définir le descripteur. Nous avons présentés différents scénarios d'utilisation, avec des sélections de nature différente. Du point de vue du temps d'exécution, notre méthode basée sur l'optimisation des paramètres d'une hyper-ellipsoïde est compatible avec une utilisation temps réel, et nous obtenons des résultats satisfaisant à la fois en terme de qualité de la sélection et de temps d'exécution vis-à-vis de l'état de l'art.

Enfin, pour assembler les différentes parties de maillages, nous avons utilisée une méthode de recalage semi-rigide pour adapter les différentes géométries et obtenir un modèle complet à l'aide d'un remaillage local.

La recherche de parties sémantiquement similaires à une sélection et le système d'adaptation géométrique de ces parties nous permet d'obtenir un système complet d'aide à la modélisation.

6.2 LIMITATIONS ET PERSPECTIVES

Le descripteur sémantique que nous avons présenté offre de bonnes propriétés en terme d'invariance aux transformations rigides et aux changements de résolution. En revanche, en ce qui concerne l'application de labellisation de modèles, la méthode que nous proposons offre de meilleurs résultats en terme de taux de labellisation en combinant notre descripteur avec des propriétés géométriques. D'une manière générale, on peut considérer que cette description sémantique ne peut pas se substituer aux autres descripteurs géométriques, mais est complémentaire.

La méthode que nous avons proposée pour la correspondance et la sélection permet d'obtenir de bonnes correspondances et de s'abstraire de la labellisation de départ. En revanche, certaines requêtes peuvent être problématiques : en particulier si la labellisation est symétrique, et que la sélection de référence ne l'est pas. Afin de remédier à ce problème, il serait intéressant d'associer au descripteur sémantique un système de coordonnées locales sur le maillages, afin de paramétrer le contour de la sélection, et de reporter ce contour sur tous les modèles de la base.

En ce qui concerne l'adaptation géométrique, le point de départ de la mise en correspondance des deux parties de surface est la recherche de deux bords proches sur les deux surfaces. Le recalage des deux surfaces se fait ensuite en fonction de la distance de chaque point au bord. Il serait intéressant de pouvoir fusionner deux maillages de topologies différentes, permettant ainsi de fusionner un maillage à un bord avec un maillage à deux bords, ce qui n'est pas permis dans la méthode employée.

Enfin, d'un point de vue plus général, le paradigme utilisé permet l'assemblage par sélection et l'échange de parties. Certaines méthodes de l'état de l'art permettent d'assembler de nouveaux modèles par ajouts de parties pré-segmentées suggérées automatiquement. Il serait intéressant de considérer un système d'assemblage suggérant de nouvelles parties à

intégrer dans les modèles, dont la segmentation serait définie automatiquement de manière à s'intégrer au mieux au modèle en cours de création.

BIBLIOGRAPHIE

- [1] Ibraheem Alhashim, Honghua Li, Kai Xu, Junjie Cao, Rui Ma, and Hao Zhang. Topology-varying 3d shape creation via structural blending. *ACM Trans. Graph.*, 33(4) :158 :1–158 :10, July 2014. (Cited pages 19, 20, 22, and 76.)
- [2] Marco Attene, Simone Marini, Michela Spagnuolo, and Bianca Falcidieno. Part-in-whole 3d shape matching and docking. *The Visual Computer*, 27(11) :991–1004, 2011. (Cited page 58.)
- [3] Marco Attene, Francesco Robbiano, and Michela Spagnuolo. Characterization of 3D shape parts for semantic annotation. *Computer-Aided Design*, 41(10) :756–763, 2009. (Cited page 35.)
- [4] Melinos Averkiou, Vladimir G. Kim, Youyi Zheng, and Niloy J. Mitra. ShapeSynth : Parameterizing model collections for coupled shape exploration and synthesis. *Computer Graphics Forum*, 33(2) :125–134, May 2014. (Cited pages viii, 19, 20, 21, 22, 35, and 76.)
- [5] S Biasotti, L De Floriani, B. Falcidieno, P. Frosini, D. Giorgi, C. Landi, L. Papaleo, and M. Spagnuolo. Describing shapes by geometrical-topological properties of real functions. *ACM Computing Surveys*, 40(4), 2008. (Cited page 34.)
- [6] S. Biasotti, D. Giorgi, M. Spagnuolo, and B. Falcidieno. Reeb graphs for shape analysis and applications. *Theoretical Computer Science*, 392(1-3) :5–22, February 2008. (Cited page 35.)
- [7] Silvia Biasotti, Simone Marini, Michela Spagnuolo, and Bianca Falcidieno. Sub-part correspondence by structural descriptors of 3D shapes. *Computer-Aided Design*, 38(9) :1002–1019, September 2006. (Cited page 35.)

- [8] Leo Breiman. Random forests. *Machine Learning*, 45(1) :5–32, 2001. (Cited page 45.)
- [9] Alexander Bronstein and Michael Bronstein. Regularized partial matching of rigid shapes. *Computer Vision—ECCV 2008*, pages 143–154, 2008. (Cited page 58.)
- [10] Alexander M. Bronstein, Michael M. Bronstein, Leonidas J. Guibas, and Maks Ovsjanikov. Shape google : Geometric words and expressions for invariant shape retrieval. *ACM Transactions on Graphics*, 30(1) :1–20, January 2011. (Cited pages 34 and 58.)
- [11] Siddhartha Chaudhuri, Evangelos Kalogerakis, Stephen Giguere, and Thomas Funkhouser. Attribit : Content creation with semantic attributes. In *ACM Symposium on User Interface Software and Technology*, pages 193–202, 2013. (Cited pages 27 and 32.)
- [12] Siddhartha Chaudhuri, Evangelos Kalogerakis, Leonidas Guibas, and Vladlen Koltun. Probabilistic reasoning for assembly-based 3d modeling. *ACM Transactions on Graphics*, 30(4) :35 :1–35 :10, July 2011. (Cited pages viii, 25, 27, 29, 32, and 35.)
- [13] Siddhartha Chaudhuri and Vladlen Koltun. Data-driven suggestions for creativity support in 3d modeling. *ACM Trans. Graph.*, 29(6) :183 :1–183 :10, December 2010. (Cited pages 25 and 29.)
- [14] Ding-Yun Chen, Xiao-Pei Tian, Yu-Te Shen, and Ming Ouhyoung. On visual similarity based 3d model retrieval. In *Computer graphics forum*, volume 22, pages 223–232. Wiley Online Library, 2003. (Cited page 59.)
- [15] Xiang Chen, Shuming Gao, Song Guo, and Jing Bai. A flexible assembly retrieval approach for model reuse. *Computer-Aided Design*, 44(6) :554–574, 2012. (Cited page 59.)
- [16] Xiaobai Chen, Aleksey Golovinskiy, and Thomas Funkhouser. A benchmark for 3D mesh segmentation. *ACM Transactions on Graphics*, 28(3), 2009. (Cited pages 36, 39, 42, 43, 44, and 46.)

- [17] David Cohen-Steiner and Frank Da. A greedy delaunay-based surface reconstruction algorithm. *The visual computer*, 20(1) :4–16, 2004. (Cited page 84.)
- [18] Nicu D Cornea, M Faith Demirci, Deborah Silver, Ali Shokoufandeh, Sven J Dickinson, and Paul B Kantor. 3d object retrieval using many-to-many matching of curve skeletons. In *Shape Modeling and Applications, 2005 International Conference*, pages 366–371. IEEE, 2005. (Cited page 59.)
- [19] T Darom and Y Keller. Scale Invariant Features for 3D Mesh Models. *IEEE Transactions on Image Processing*, 21(5) :2758 – 2769, January 2012. (Cited page 32.)
- [20] Mathias Eitz, Ronald Richter, Tamy Boubekeur, Kristian Hildebrand, and Marc Alexa. Sketch-based shape retrieval. *ACM Trans. Graph.*, 31(4) :31–1, 2012. (Cited page 59.)
- [21] Lubin Fan, Kun Liu, et al. Paint mesh cutting. In *Computer graphics forum*, volume 30, pages 603–612. Wiley Online Library, 2011. (Cited page 57.)
- [22] Lubin Fan, Min Meng, and Ligang Liu. Sketch-based mesh cutting : A comparative study. *Graphical Models*, 74(6) :292–301, 2012. (Cited page 57.)
- [23] Lubin Fan, Ruimin Wang, Linlin Xu, Jiansong Deng, and Ligang Liu. Modeling by drawing with shadow guidance. *Computer Graphics Forum*, 32(7) :157–166, 2013. (Cited pages 23, 29, 32, 76, and 77.)
- [24] Janis Fehr, Alexander Streicher, and Hans Burkhardt. A bag of features approach for 3d shape retrieval. In *Proceedings of the 5th International Symposium on Advances in Visual Computing : Part I*, pages 34–43. Springer-Verlag, 2009. (Cited page 58.)
- [25] Noa Fish, Melinos Averkiou, Oliver van Kaick, Olga Sorkine-Hornung, Daniel Cohen-Or, and Niloy J. Mitra. Meta-representation of shape families. *ACM Transactions on Graphics*, 33(4) :1–11, July 2014. (Cited page 35.)

- [26] Thomas Funkhouser, Patrick Min, Michael Kazhdan, Joyce Chen, Alex Halderman, David Dobkin, and David Jacobs. A search engine for 3D models. *ACM Transactions on Graphics*, 22(1) :83–105, 2003. (Cited pages viii, ix, 16, 17, 18, 21, 22, 24, 28, 34, 57, and 77.)
- [27] Ran Gal, Ariel Shamir, and Daniel Cohen-Or. Pose-oblivious shape signature. *IEEE Transactions on Visualization and Computer Graphics*, 13(2) :261–71, 2007. (Cited page 34.)
- [28] Xuekun Guo, Juncong Lin, Kai Xu, and Xiaogang Jin. Creature grammar for creative modeling of 3D monsters. *Graphical Models*, 76(5) :376–389, September 2014. (Cited page 15.)
- [29] Emilie Guy, JM Thiery, and T Boubekeur. SimSelect : Similarity-based selection for 3D surfaces. *Computer Graphics Forum*, 33(2) :165–173, 2014. (Cited pages ix, 60, 61, 62, 71, 72, and 73.)
- [30] Qixing Huang, Vladlen Koltun, and Leonidas Guibas. Joint shape segmentation with linear programming. *ACM Transactions on Graphics*, 30(6), 2011. (Cited pages 35, 36, and 62.)
- [31] Takeo Igarashi, Satoshi Matsuoka, and Hidehiko Tanaka. Teddy : A sketching interface for 3d freeform design. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '99*, pages 409–416, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co. (Cited pages 14 and 15.)
- [32] Arjun Jain, Thorsten Thormählen, Tobias Ritschel, and Hans-Peter Seidel. Exploring shape variations by 3d-model decomposition and part-based recombination. *Computer Graphics Forum*, 31(2pt3) :631–640, May 2012. (Cited pages viii, 19, 20, 22, and 76.)
- [33] Zhongping Ji, Ligang Liu, Zhonggui Chen, and Guojin Wang. Easy mesh cutting. In *Computer Graphics Forum*, volume 25, pages 283–291. Wiley Online Library, 2006. (Cited page 57.)
- [34] Andrew Edie Johnson. *Spin-images : a representation for 3-D surface matching*. PhD thesis, 1997. (Cited pages 25 and 58.)

- [35] E Kalogerakis and Aaron Hertzmann. Learning 3D mesh segmentation and labeling. *ACM Transactions on Graphics*, 29(4), 2010. (Cited pages 32, 35, 36, 37, 41, 42, 46, 47, and 48.)
- [36] Evangelos Kalogerakis, Siddhartha Chaudhuri, and Daphne Koller. A Probabilistic Model for Component-Based Shape Synthesis. *ACM Transactions on Graphics*, 2012. (Cited pages 27 and 32.)
- [37] Takashi Kanai, Hiromasa Suzuki, Jun Mitani, and Fumihiko Kimura. Interactive mesh fusion based on local 3d metamorphosis. In *Graphics Interface*, volume 99, pages 148–156, 1999. (Cited page 77.)
- [38] Michael Kazhdan, Thomas Funkhouser, and Szymon Rusinkiewicz. Rotation invariant spherical harmonic representation of 3 d shape descriptors. In *Symposium on geometry processing*, volume 6, pages 156–164, 2003. (Cited page 58.)
- [39] Vladimir G. Kim, Wilmot Li, Niloy J. Mitra, Siddhartha Chaudhuri, Stephen DiVerdi, and Thomas Funkhouser. Learning part-based templates from large collections of 3D shapes. *ACM Transactions on Graphics*, 32(4), July 2013. (Cited pages 32, 35, and 36.)
- [40] Oscar Kin-Chung Au, Youyi Zheng, Menglin Chen, Pengfei Xu, and Chiew-Lan Tai. Mesh Segmentation with Concavity-Aware Fields. *IEEE Transactions on Visualization and Computer Graphics*, 18(7) :1125–1134, July 2011. (Cited page 45.)
- [41] Jan J. Koenderink and Andrea J. van Doorn. Surface shape and curvature scales. *Image Vision Comput.*, 10(8) :557–565, October 1992. (Cited pages 34, 47, and 48.)
- [42] I. Kokkinos, M. M. Bronstein, R. Litman, and a. M. Bronstein. Intrinsic shape context descriptors for deformable shapes. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 159–166, June 2012. (Cited pages 32, 34, and 60.)
- [43] Vladislav Kreavoy, Dan Julius, and Alla Sheffer. Model composition from interchangeable components. *Computer Graphics and Applications*, pages 129 – 138, 2007. (Cited pages viii, 24, and 25.)

- [44] Hamid Laga, M Mortara, and M Spagnuolo. Geometry and context for semantic correspondences and functionality recognition in man-made 3D shapes. *ACM Transactions on Graphics*, 32(5), 2013. (Cited page 35.)
- [45] G Lavoué. Combination of bag-of-words descriptors for robust partial shape retrieval. *The Visual Computer*, 28(9) :931–942, 2012. (Cited pages 34 and 58.)
- [46] Chang Ha Lee, Amitabh Varshney, and David W Jacobs. Mesh saliency. In *ACM transactions on graphics (TOG)*, volume 24, pages 659–666. ACM, 2005. (Cited page 58.)
- [47] George Leifman and Ayellet Tal. Pattern-driven colorization of 3d surfaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 241–248, 2013. (Cited pages 59 and 62.)
- [48] Y. J. Liu, C. X. Ma, and D. L. Zhang. Easytoy : Plush toy design using editable sketching curves. *IEEE Computer Graphics and Applications*, 31(2) :49–57, March 2011. (Cited page 14.)
- [49] Z Liu, S Bu, K Zhou, and X Sun. Geometrically attributed binary tree for 3d shape matching. In *Proc. the 25th International Conference on Computer Graphics*, 2011. (Cited page 59.)
- [50] Zhen-Bao Liu, Shu-Hui Bu, Kun Zhou, Shu-Ming Gao, Jun-Wei Han, and Jun Wu. A survey on partial retrieval of 3d shapes. *Journal of Computer Science and Technology*, 28(5) :836–851, 2013. (Cited page 58.)
- [51] J Lv, X Chen, J Huang, and H Bao. Semi-supervised Mesh Segmentation and Labeling. *Computer Graphics Forum*, 31(7) :2241–2248, 2012. (Cited pages 35 and 36.)
- [52] Athanasios Mademlis, Petros Daras, Apostolos Axenopoulos, Dimitrios Tzovaras, and Michael G Strintzis. Combining topological and geometrical features for global and partial 3-d shape retrieval. *Multimedia, IEEE Transactions on*, 10(5) :819–831, 2008. (Cited page 59.)

- [53] Sotiris Malassiotis and Michael G Strintzis. Snapshots : A novel local surface descriptor and matching algorithm for robust 3d surface alignment. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(7) :1285–1290, 2007. (Cited page 58.)
- [54] A. Maximo, R. Patro, Amitabh Varshney, and R. Farias. A robust and rotationally invariant local surface descriptor with applications to non-local mesh processing. *Graphical Models*, 73 :231 – 242, 2011/09//2011. (Cited page 59.)
- [55] Min Meng, Lubin Fan, and Ligang Liu. icutter : a direct cut-out tool for 3d shapes. *Computer Animation and Virtual Worlds*, 22(4) :335–342, 2011. (Cited page 57.)
- [56] P. Merrell and D. Manocha. Model synthesis : A general procedural modeling algorithm. *Visualization and Computer Graphics, IEEE Transactions on*, 17(6) :715–728, June 2011. (Cited pages 15 and 16.)
- [57] Pascal Müller, Peter Wonka, Simon Haegler, Andreas Ulmer, and Luc Van Gool. Procedural modeling of buildings. *ACM Trans. Graph.*, 25(3) :614–623, July 2006. (Cited page 15.)
- [58] Andrew Nealen, Olga Sorkine, Marc Alexa, and Daniel Cohen-Or. A sketch-based interface for detail-preserving mesh editing. *ACM Trans. Graph.*, 24(3) :1142–1147, July 2005. (Cited page 14.)
- [59] Ryutarou Ohbuchi, Kunio Osada, Takahiko Furuya, and Tomohisa Banno. Salient local visual features for shape-based 3d model retrieval. In *Shape Modeling and Applications, 2008. SMI 2008. IEEE International Conference on*, pages 93–102. IEEE, 2008. (Cited page 59.)
- [60] Luke Olsen, Faramarz F. Samavati, Mario Costa Sousa, and Joaquim A. Jorge. Technical section : Sketch-based modeling : A survey. *Comput. Graph.*, 33(1) :85–103, February 2009. (Cited page 14.)
- [61] R Osada and T Funkhouser. Shape distributions. *ACM Transactions on Graphics*, 21(4) :807–832, 2002. (Cited page 34.)

- [62] Maks Ovsjanikov, Mirela Ben-Chen, Justin Solomon, Adrian Butscher, and Leonidas Guibas. Functional maps : a flexible representation of maps between shapes. *ACM Transactions on Graphics*, 31(4) :30 :1–30 :11, 2012. (Cited pages 42 and 61.)
- [63] Panagiotis Papadakis, Ioannis Pratikakis, Theoharis Theoharis, and Stavros Perantonis. Panorama : A 3d shape descriptor based on panoramic views for unsupervised 3d object retrieval. *International Journal of Computer Vision*, 89(2-3) :177–192, 2010. (Cited page 59.)
- [64] Yoav I. H. Parish and Pascal Müller. Procedural modeling of cities. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '01*, pages 301–308, New York, NY, USA, 2001. ACM. (Cited page 15.)
- [65] Michael J. David Powell. *Advances in Optimization and Numerical Analysis*, chapter A Direct Search Optimization Method That Models the Objective and Constraint Functions by Linear Interpolation, pages 51–67. Springer Netherlands, Dordrecht, 1994. (Cited page 65.)
- [66] Przemyslaw Prusinkiewicz. Graphical applications of l-systems. In *Proceedings of graphics interface*, volume 86, pages 247–253, 1986. (Cited page 15.)
- [67] Martin Reuter, Franz-Erich Wolter, and Niklas Peinecke. Laplace-Beltrami spectra as Shape-DNA of surfaces and solids. *Computer-Aided Design*, 38(4) :342–366, April 2006. (Cited page 34.)
- [68] E. Rodolà, L. Cosmo, M. M. Bronstein, A. Torsello, and D. Cremers. Partial functional correspondence. *Computer Graphics Forum*, 2016. (Cited page 60.)
- [69] Raif M. Rustamov. Laplace-Beltrami eigenfunctions for deformation invariant shape representation. In *Eurographics symposium on Geometry processing*, pages 225 – 233, 2007. (Cited page 34.)
- [70] R. Schmidt, B. Wyvill, M. C. Sousa, and J. A. Jorge. Shapeshop : Sketch-based solid modeling with blobtrees. In *ACM SIGGRAPH*

- 2006 Courses, SIGGRAPH '06, New York, NY, USA, 2006. ACM. (Cited page 14.)
- [71] Konstantinos Sfikas, Theoharis Theoharis, and Ioannis Pratikakis. Non-rigid 3D object retrieval using topological information guided by conformal factors. *The Visual Computer*, 28(9) :943–955, April 2012. (Cited page 35.)
- [72] Ying Shan, Harpreet S Sawhney, Bogdan Matei, and Rakesh Kumar. Shapeme histogram projection and matching for partial object recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(4) :568–577, 2006. (Cited page 58.)
- [73] Lior Shapira, Ariel Shamir, and Daniel Cohen-Or. Consistent mesh partitioning and skeletonisation using the shape diameter function. *The Visual Computer*, 24(4) :249–259, 2008. (Cited page 25.)
- [74] Andrei Sharf, Marina Blumenkrants, Ariel Shamir, and Daniel Cohen-Or. Snappaste : an interactive technique for easy mesh composition. *The Visual Computer*, 22(9-11) :835–844, 2006. (Cited pages 78, 80, 81, 82, and 84.)
- [75] Oana Sidi, O. van Kaick, Y. Kleiman, H. Zhang, and D. Cohen-Or. Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering. *ACM Transactions on Graphics*, 30(6), 2011. (Cited pages 35, 36, and 62.)
- [76] Lucian Stanculescu, Raphaëlle Chaine, Marie-Paule Cani, and Karan Singh. Sculpting multi-dimensional nested structures. *Computers & Graphics*, 37(6) :753–763, 2013. (Cited page 57.)
- [77] Jian Sun, Maks Ovsjanikov, and Leonidas Guibas. A Concise and Provably Informative Multi-Scale Signature Based on Heat Diffusion. *Computer Graphics Forum*, 28(5) :1383–1392, July 2009. (Cited pages 34 and 58.)
- [78] Vitaly Surazhsky, Tatiana Surazhsky, Danil Kirsanov, Steven J. Gortler, and Hugues Hoppe. Fast exact and approximate geodesics on

- meshes. *ACM Transactions on Graphics*, 24(3) :553–560, July 2005. (Cited page 38.)
- [79] Jerry O. Talton, Yu Lou, Steve Lesser, Jared Duke, Radomír Měch, and Vladlen Koltun. Metropolis procedural modeling. *ACM Trans. Graph.*, 30(2) :11 :1–11 :14, April 2011. (Cited page 15.)
- [80] Johan W. H. Tangelder and Remco C. Veltkamp. A survey of content based 3D shape retrieval methods. *Multimedia Tools and Applications*, 39(3) :441–471, December 2008. (Cited pages 32 and 33.)
- [81] Johan WH Tangelder and Remco C Veltkamp. A survey of content based 3d shape retrieval methods. *Multimedia tools and applications*, 39(3) :441–471, 2008. (Cited page 58.)
- [82] Julien Tierny, Jean-Philippe Vandeborre, and Mohamed Daoudi. Partial 3D shape retrieval by Reeb pattern unfolding. *Computer Graphics Forum*, 28(1) :41–55, March 2009. (Cited pages 35 and 59.)
- [83] Roberto Toldo, Umberto Castellani, and Andrea Fusiello. A bag of words approach for 3d object categorization. In *Computer Vision/Computer Graphics Collaboration Techniques : 4th International Conference, MIRAGE 2009, Rocquencourt, France, May 4-6, 2009, Proceedings*, volume 5496, page 116. Springer, 2009. (Cited page 59.)
- [84] O. van Kaick, Andrea Tagliasacchi, Oana Sidi, Hao Zhang, D. Cohen-Or, Lior Wolf, and Ghassan Hamarneh. Prior Knowledge for Part Correspondence. *Computer Graphics Forum*, 30(2), 2011. (Cited pages 35 and 36.)
- [85] Oliver van Kaick, Kai Xu, Hao Zhang, Yanzhen Wang, Shuyang Sun, and Ariel Shamir. Co-Hierarchical Analysis of Shape Structures. *ACM Transactions on Graphics*, 32(4), 2013. (Cited pages 35, 36, and 62.)
- [86] Oliver van Kaick, Hao Zhang, Ghassan Hamarneh, and Daniel Cohen-Or. A Survey on Shape Correspondence. *Computer Graphics Forum*, 30(6) :1681–1707, June 2011. (Cited page 33.)

- [87] Xiaohua Xie, Kai Xu, Niloy J. Mitra, Daniel Cohen-Or, Wenyong Gong, Qi Su, and Baoquan Chen. Sketch-to-design : Context-based part assembly. *Computer Graphics Forum*, 32(8) :233–245, 2013. (Cited pages viii, 22, 23, 29, 32, and 35.)
- [88] Zhige Xie, K Xu, L Liu, and Y Xiong. 3D Shape Segmentation and Labeling via Extreme Learning Machine. *Computer Graphics Forum*, 33(5), 2014. (Cited pages 35 and 36.)
- [89] Kai Xu, Vladimir G. Kim, Qixing Huang, and Evangelos Kalogerakis. Data-Driven Shape Analysis and Processing. *arXiv :1502.06686*, February 2015. (Cited page 33.)
- [90] Kai Xu, Hao Zhang, Daniel Cohen-Or, and Baoquan Chen. Fit and diverse : Set evolution for inspiring 3d shape galleries. *ACM Transactions on Graphics*, 31(4) :57 :1–57 :10, July 2012. (Cited pages 19, 20, 21, 22, and 76.)
- [91] Andrei Zaharescu, Edmond Boyer, Kiran Varanasi, and Radu Horaud. Surface Feature Detection and Description with Applications to Mesh Matching. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 373–380, June 2009. (Cited page 34.)
- [92] Robert C. Zeleznik, Kenneth P. Herndon, and John F. Hughes. Sketch : An interface for sketching 3d scenes. In *Proceedings of SIGGRAPH 96*, Computer Graphics Proceedings, Annual Conference Series, pages 163–170, August 1996. (Cited page 14.)
- [93] Juyong Zhang, Jianmin Zheng, Chunlin Wu, and Jianfei Cai. Variational Mesh Decomposition. In *ACM Transactions on Graphics*, 2012. (Cited page 45.)
- [94] Youyi Zheng, Daniel Cohen-Or, Melinos Averkiou, and Niloy J. Mitra. Recurring part arrangements in shape collections. *Computer Graphics Forum*, 32(2) :115–124, 2014. (Cited pages 22 and 35.)
- [95] Youyi Zheng, Daniel Cohen-Or, and Niloy J. Mitra. Smart variations : Functional substructures for part compatibility. *Computer Graphics Forum*, 32(2pt2) :195–204, 2013. (Cited pages 19, 20, 76, and 77.)

- [96] Youyi Zheng and Chiew-Lan Tai. Mesh decomposition with cross-boundary brushes. In *Computer Graphics Forum*, volume 29, pages 527–535. Wiley Online Library, 2010. (Cited page 57.)