HAL
open science

Séquences de maillages : classification et méthodes de segmentation

Romain Arcila

To cite this version:

Romain Arcila. Séquences de maillages : classification et méthodes de segmentation. Multimédia [cs.MM]. Université Claude Bernard - Lyon I, 2011. Français. NNT : . tel-00653542v1

HAL Id: tel-00653542

https://theses.hal.science/tel-00653542v1

Submitted on 21 Dec 2011 (v1), last revised 26 Jun 2013 (v3)

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.

N° d'ordre XXX-XXXX

Année 2011

THESE DE L'UNIVERSITE DE LYON

delivrée par

L'UNIVERSITE CLAUDE BERNARD LYON 1

ECOLE DOCTORALE INFORMATIQUE ET MATHEMATIQUES

préparée au

LABORATOIRE LIRIS UMR 5205 CNRS

pour l'obtention du

DIPLOME DE DOCTORAT
(arrêté du 7 aout 2006)

SPECIALITE INFORMATIQUE

soutenue publiquement le

par

Romain Arcila

Séquences de maillages : classification et méthodes de segmentation

Jury

Mme. Dominique Bechmann	Professeure, Université de Strasbourg	Rapporteure
M. Bruno Lévy	Directeur de Recherche, INRIA	Rapporteur
Mr Christophe Garcia	Professeur, Université Lyon 2	Examinateur
M. Julien Tierny	Maître de Conférences, Telecom ParisTech	Examinateur
M. Florent Dupont	Professeur, Université Lyon 1	Directeur de thèse
M. Franck Hétroy	Maître de Conférences, Grenoble INP	Co-directeur de thèse

Résumé

Les séquences de maillages sont de plus en plus utilisées, comme on peut le voir avec les jeux vidéos ou l'imagerie médicale. Cette augmentation des besoins entraîne un développement des méthodes de génération de séquences de maillages, parmi lesquelles les méthodes fondées sur un modeleur et les acquisitions multi-caméras. Ces méthodes de générations peuvent produire des séquences de maillages de natures différentes. Le nombre d'applications utilisant ces séquences s'est également accru, avec par exemple la compression et le transfert de pose. Ces applications nécessitent souvent dans un premier temps de calculer une partition de la séquence.

Dans cette thèse, nous nous intéressons plus particulièrement à la segmentation en composantes rigides de séquences de maillages. Ainsi dans un premier temps, nous formalisons la notion de séquence de maillages. En effet, les applications utilisant les séquences de maillages se sont développées de manière indépendante les unes des autres et il n'existe pas de terminologie commune. Nous proposons donc une classification et une nomenclature permettant de désigner quelles sont les propriétés attachées à un type de séquence, et ainsi de décrire précisément quel type de séquence est nécessaire pour une application donnée.

Dans un second temps, nous formalisons la notion de segmentation de séquence de maillages en posant différentes définitions. Nous présentons également l'état de l'art des méthodes de segmentation sur les séquences de maillages, pour les différents types de segmentations définies.

Ensuite, nous proposons une première méthode de segmentation de type globale pour les séquences stables de maillages. Cette méthode est fondée sur la fusion de régions.

Par la suite, nous présentons deux autres méthodes, toutes deux reposant sur la classification spectrale, et qui acceptent en entrée des séquences non contraintes de maillages. La première produit un ensemble de segmentations globales, tandis que la seconde génère, soit une segmentation globale, soit une segmentation temporellement variable.

Nous mettons également en place un système d'évaluation quantitative des segmentations.

Enfin, nous présentons les différentes perspectives liées à la segmentation de séquences de maillages.

Abstract

Mesh sequences are becoming a common tool in computer graphics as can be seen in video games or medical imaging. As a consequence, mesh sequences generation's methods have been developed. Sequences can be created using a modeler or by multi-cameras acquisition. These methods can generate sequences which have different properties. Moreover, different applications using mesh sequences have been proposed such as compression or pose transfer. Most of these applications require to compute a segmentation as a pre-processing step.

This thesis deals with mesh sequence segmentation into rigid components. First, we formalize the notion of mesh sequence, and propose a nomenclature, allowing us to design the properties attached to each kind of mesh sequence, and to describe precisely what kind of sequence an algorithm requires as input.

Next, we propose a formalization of mesh sequence segmentation by providing different definitions of segmentation. We then present mesh sequences segmentation state of the art.

We then propose a first segmentation method, producing a global segmentation and taking as input a stable mesh sequence. This method is based on a region-growing process.

Next, we present two segmentation methods, both based on spectral clustering and taking as input an unconstrained mesh sequence. The former method produces a set of global segmentations, while the latter generate either a global segmentation or a time-varying segmentation.

We also propose a quantitative segmentation evaluation system.

Finally, we provide future work leads on mesh sequence segmentation.

Notations

$\mathrm{Ng}(v_i^{(k)})$	1-voisinage du sommet $v_i^{(k)}$, page 11
F_k, $0 \leqslant k \leqslant N-1$	$k^{ième}$ image, pouvant contenir un maillage avec plusieurs composantes connexes, page 11
i, j	indices des sommets du maillage dans l'image k, page 11
k, l	indices des images, page 11
N	nombre d'images composant la séquence, page 11
$nv(F_k)$	nombre de sommets dans F_k, indexés de 0 à $nv(F_k)-1$, page 11
$v_i^{(k)}$, $0 \leqslant i \leqslant nv(F_k)-1$	sommet d'index i dans F_k, page 11

Table des matières

Résumé iii
Abstract iv
Notations v

1 Introduction **1**
1.1 Contexte 1
1.1.1 Génération de séquences de maillages 1
1.1.2 Traitements 2
1.2 Objectifs de la thèse 4
1.3 Plan 5

2 Définition et classification de séquences de maillages **7**
2.1 Introduction 7
2.2 Rappels et travaux antérieurs 8
2.2.1 Maillage 8
2.2.1.1 Définition 8
2.2.1.2 Propriétés sur les maillages 8
2.2.2 Séquence de maillage : nomenclature existante 10
2.3 Séquences de maillages : formalisme 11
2.3.1 Définition générale d'une séquence de maillages 11
2.3.2 Maillage dynamique 12
2.3.2.1 Définition 12
2.3.2.2 Génération 13
2.3.3 Séquence stable de maillages 13
2.3.3.1 Définition 13
2.3.3.2 Génération 13
2.3.4 Séquence non contrainte de maillages 15
2.3.4.1 Définition 15
2.3.4.2 Génération 15
2.3.5 Limitations de la classification courante 15
2.3.5.1 UMS avec Séparation 15
2.3.5.2 UMS avec Fusion 17

2.3.5.3 UMS avec Apparition/Disparition de géométrie . 17
2.3.6 Hiérarchie 17
2.3.6.1 Imbrication des classes. 17
2.3.6.2 Décomposition en sous-séquences. 19
2.4 Conclusion 20

3 Classifications des méthodes de segmentation 21
3.1 Introduction 21
3.2 Classification des méthodes de segmentation 23
3.2.1 Etat de l'art 23
3.2.1.1 Segmentation statique 25
3.2.1.2 Segmentation de séquence de maillages 26
3.2.2 Formalisation et raffinement de la classification 27
3.2.2.1 Segmentation globale 27
3.2.2.2 Segmentation par instances 27
3.2.2.3 Segmentation temporellement variable 29
3.2.2.4 Segmentation temporelle 30
3.2.2.5 Combinaison de segmentations 31
3.2.2.6 Critère de segmentation 31
3.3 Etat de l'art des méthodes de segmentation 32
3.3.1 Segmentation de maillages dynamiques 33
3.3.2 Segmentation de séquences stables et non contraintes de maillages 34
3.3.2.1 Difficultés 34
3.3.2.2 Etat de l'art des méthodes de segmentation 36
3.3.2.3 Segmentation temporelle 36

4 Segmentation globale fondée sur la fusion de régions 39
4.1 Nature des clusters 40
4.2 Chaîne d'exécution 40
4.3 Mise en correspondance 42
4.3.1 Maillages dynamiques 42
4.3.2 Recalage fondé sur CPD 42
4.3.3 Projection 43
4.3.4 Mise en correspondance et vecteurs de déplacement . . . 44
4.3.5 Mise en correspondance d'un maillage avec le reste de la séquence 45
4.4 Segmentation 46
4.4.1 Vue d'ensemble 46
4.4.2 Erreurs 47
4.4.3 Segmentation initiale 47
4.4.3.1 Non-Rigid Registration Under Isometric Deformation [35] 48
4.4.3.2 Version modifiée 48
4.4.4 Raffinement 51
4.4.5 Transfert de la segmentation sur la séquence 52

4.5 Discussion 53
4.5.1 Mise en correspondance 53
4.5.2 Segmentation 55
4.5.2.1 Résultats 55
4.5.2.2 Entrée utilisateur 58
4.5.2.3 Frontière et résistance au bruit 60
4.5.2.4 Mise en correspondance et segmentation 61
4.5.2.5 Pile de mouvement 61
4.5.3 Limitations 62
4.5.3.1 Dépendance aux résultats de la mise en correspondance 62
4.5.3.2 Modèle et changement de topologie 62
4.5.3.3 Goulot d'étranglement 62
4.5.3.4 Correspondance unidirectionnelle et croissance de clusters 62
4.5.3.5 Mouvement lent et subdivision erronée 63
4.5.3.6 Pile de mouvement 63
4.5.3.7 Dérive de la segmentation 63

5 Classification spectrale **65**
5.1 Segmentation globale. 65
5.1.1 Amélioration visée par rapport à la méthode par fusion de clusters. 66
5.1.2 Mise en correspondance 66
5.1.2.1 Recalage 67
5.1.2.2 Correspondance et vecteurs de déplacement . . . 68
5.1.3 Segmentation 68
5.1.3.1 Vue d'ensemble 68
5.1.3.2 Principe de la classification spectrale 69
5.1.3.3 Application à la segmentation de séquences de maillages 70
5.1.3.4 Etape de raffinement 71
5.1.3.5 Transfert de segmentation et changement de topologie 72
5.1.4 Résultats et discussion 73
5.1.4.1 Résultats 73
5.1.4.2 Discussion 76
5.2 Segmentation temporellement variable 78
5.2.1 Introduction 78
5.2.1.1 But de la méthode 78
5.2.1.2 Chaîne d'exécution 79
5.2.2 Mise en correspondance 79
5.2.2.1 Recalage 79
5.2.2.2 Mise en correspondance 80
5.2.3 Segmentation 80
5.2.3.1 Segmentation spectrale 80

5.2.3.2 Transfert de segmentation sur la séquence 81
5.2.4 Résultats et discussion 81
5.2.4.1 Résultats 81
5.2.4.2 Segmentation entre images distantes 82
5.2.4.3 Discussion 82

6 Evaluation quantitative des méthodes de segmentation 87
6.1 Segmentation optimale : problème et définition 87
6.2 Critères d'erreurs 88
6.3 Résultats sur les méthodes de segmentation présentées 88
6.3.1 Segmentation globale 88
6.3.2 Segmentation temporellement variable 90
6.4 Conclusion 91

7 Conclusion 93
7.1 Bilan 93
7.2 Perspectives 94
7.2.1 Segmentation 94
7.2.1.1 Améliorations des méthodes présentées 94
7.2.1.2 Perspectives plus générales 95
7.2.2 Protocole d'évaluation quantitative 96

Bibliographie 97

Lexique 105

Table des figures

1.1 Extraction de squelette. 3
1.2 Exemple de transfert de poses. 3

2.1 Exemple de maillage quadrangulaire. 9
2.2 2-variété et orientabilité. 10
2.3 Illustration de la méthode d'animation par squelette. 14
2.4 Exemple de SMS. 14
2.5 Reconstruction multicaméras. Exemple de maillages reconstruits. 16
2.6 Illustration des classes intégrant les changements de topologie et de genre. 18
2.7 Imbrication et hiérarchie des classes. 19

3.1 Exemple d'images segmentées. 22
3.2 Exemple de segmentation. 24
3.3 Exemple de segmentation hiérarchique. 24
3.4 Exemple de segmentation de maillage statique. 25
3.5 Exemple de segmentations hiérarchiques de maillage statiques. . 26
3.6 Illustration de la segmentation globale. 28
3.7 Exemples [46] de segmentations globales sur différents maillages dynamiques. 28
3.8 Illustration d'une segmentation par instances. 29
3.9 Illustration d'une segmentation temporelle variable. 30
3.10 Exemple de segmentation temporellement variable. 30
3.11 Illustration d'une segmentation temporelle. 31
3.12 Segmentations obtenues par [37]. 33
3.13 Segmentations obtenues par [29]. 33
3.14 Résultats des segmentations globales dans l'état de l'art. 35
3.15 Résultats des segmentations globales de l'état de l'art sur des séquences de maillages. 37

4.1 Chaîne d'exécution du processus de segmentation. 41
4.2 Recalage de deux maillages. 43

4.3 Projection de maillage. 44
4.4 Correspondance et vecteur de déplacement. 45
4.5 Chaîne d'exécution du processus de segmentation. 46
4.6 Segmentation avec [35] 49
4.7 Segmentation Rigide. 50
4.8 Clusters statiques, étirés, rigides et sous-maillage. 51
4.9 Exemple de raffinement pour la méthode par fusion de clusters. . 52
4.10 Transfert de la séquence. 53
4.11 Résultat de la mise en correspondance. 54
4.12 Erreur de mise en correspondance 54
4.13 Segmentation globale du maillage "horse 55
4.14 Gestion des étirements 56
4.15 Segmentation sur une séquence stable. 57
4.16 Paramètres et raffinement de la segmentation. 58
4.17 Comparaisons de Segmentation, bruit et erreur 59
4.18 Positionnement des frontières d'un cluster. 60
4.19 Erreurs de mise en correspondance groupées 61

5.1 Chaîne d'exécution 67
5.2 Mise en correspondance. 67
5.3 Utilisation de la fenêtre temporelle par cluster. 72
5.4 Résultat de la mise en correspondance. (a) Complet. (b) Partiel. . 73
5.5 Segmentation sur un maillage dynamique 74
5.6 Raffinement de la segmentation sur la séquence dynamique "Horse". (a) la value de l'eigen-gap est de 0.7. (b) la valeur de l'eigen-gap est de 0.5 74
5.7 Résultat sur la séquence stable de maillages 75
5.8 Segmentation d'une séquence reconstruite par multi-caméra . . . 75
5.9 Segmentation sur la séquence "Walk". 76
5.10 Chaîne d'exécution 79
5.11 Processus de mise en correspondance. 80
5.12 Segmentation temporellement variable. 83
5.13 Segmentation globale sur la séquence "Balloon". 84
5.14 Segmentation entre images distantes. 84

6.1 Evaluation sur la séquence "Cat" 89

Liste des tableaux

2.1 Séquences de maillages : propriétés possibles et noms associés. . 12
2.2 Séquences de maillages invalides. 12
2.3 Nouvelle classification des séquences de maillages. 17

6.1 Résultat de l'évaluation qualitative sur la séquence "Cat" avec 5165 sommets 90

Liste des Algorithmes

1 Corps de la méthode de segmentation. 41
2 Mise En Correspondance 42
3 Projection 44
4 Correspondance. 45
5 Segmentation Rigide. 50

6 Mise en correspondance. 67
7 Correspondance 68
8 Segmentation spectrale normalisée selon Shi et Malik [62] 70

Chapitre 1

Introduction

Dans cette thèse, nous nous intéressons à la segmentation des maillages évoluant au cours du temps, communément appelés séquences de maillages. L'utilisation de ces séquences de maillages n'est pas nouvelle, mais des avancées technologiques récentes dans la génération de telles séquences, ainsi que les évolutions matérielles et l'augmentation des débits des réseaux ont provoqué un regain d'intérêt pour celles-ci ces dernières années.

1.1 Contexte

1.1.1 Génération de séquences de maillages

Comme mentionné précédemment, l'utilisation de la 3D et de la 3D+t s'est fortement accrue au cours de ces dernières années. On peut citer par exemple les films et les jeux vidéos dans les applications visibles, ainsi que d'autres domaines moins grand public comme l'imagerie médicale et la CAO [1]. Ainsi le nombre de modèles de représentation d'objet 3D a augmenté. Les modèles surfaciques en particulier, comme les NURBS [2] ou les surfaces paramétriques, ont une place prépondérante. Parmi les modèles surfaciques, les maillages sont les plus utilisés, cela est en partie dû au support matériel et aux accélérations matérielles par la carte graphique, couramment présents dans les ordinateurs actuels. L'utilisation prépondérante des maillages nécessite la mise en place de nombreux traitements, qu'ils soient de haut niveau (déformation de maillages et transfert de pose par exemple) ou de bas niveau (remaillage par exemple). Ces maillages sont ensuite animés. Un graphiste crée dans un premier temps un maillage (avec un modeleur par exemple), puis un animateur est chargé de mettre en mouvement ce maillage. Nous présentons ici les deux principales familles d'animation de maillages :

1. Conception assistée par ordinateur
2. pour Non Uniform Rational B-Spline

- animation par images clés : l'animateur déforme le maillage, ou une représentation simplifiée, à différents instants, par exemple aux images k et $k+l$. Les images intermédiaires, entre $k+1$ et $k+l-1$, sont "interpolées" (par différentes techniques) à partir de ces deux images (voir la section 2.3.2.2). Les méthodes par images clés sont omniprésentes dans les modeleurs ;
- animation procédurale : le maillage est déformé en suivant un ensemble de règles, reposant généralement sur des lois physiques. Il s'agit par exemple d'animations physiques (simulation de vêtement) ou de simulations numériques.

Parallèlement les travaux d'analyse d'images et de vidéos progressent et des applications complexes et interactives deviennent possibles. Ainsi il est possible à partir d'une installation multi-caméras de reconstruire un maillage de l'objet filmé. En appliquant ce principe sur toutes les images d'une vidéo, des séquences de maillages peuvent donc être créées.

Ces séquences sont, comme nous le verrons dans le chapitre 2, d'une nature différente de celles présentées ci-dessus.

1.1.2 Traitements

Les séquences de maillages sont de plus en plus utilisées dans l'industrie. Les principaux traitements appliqués à ces séquences sont les suivants :

- la compression de séquences de maillages : il s'agit actuellement de l'application la plus importante et de nombreux travaux ont été publiés ces dernières années. La majeure partie de ces travaux porte sur les séquences générées à partir d'un modeleur [6, 18, 23, 47, 57]. Il existe également des travaux sur les séquences reconstruites par acquisition multicaméras[31, 73]. Le but premier de la compression est de réduire l'espace de stockage requis pour une séquence de maillages contenant des informations redondantes. Les méthodes de compression peuvent avoir également d'autres critères tels que la transmission interactive permettant de visualiser une séquence durant son téléchargement ou le placement arbitraire à un instant de la vidéo.
- L'extraction et paramétrisation de squelette d'une séquence générée avec un modeleur [25]. Il s'agit ici principalement de rétro-animation, et synthèse de nouveau mouvement sur un maillage déjà existant, voir la figure 1.1.
- Le transfert de pose [11, 46] : on dispose d'un maillage animé M_1 et d'un maillage qui peut être non animé (appelé statique) ou animé M_2. Le but est de transférer les mouvements du maillage M_1 sur le maillage M_2, voir la figure 1.2.
- La simplification de séquences de maillages [53] : il s'agit ici de produire une séquence géométriquement proche de la séquence d'origine où chaque maillage de la séquence contient moins de sommets que les maillages de la séquence d'origine.
- Le découpage en sous séquences [71] : il s'agit de découper la séquence en sous-séquences, où chaque sous-séquence correspond à une action.

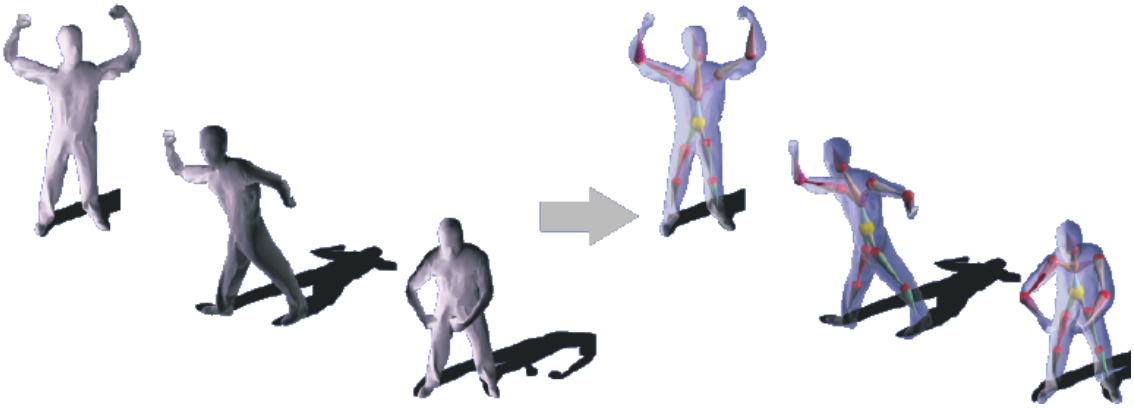
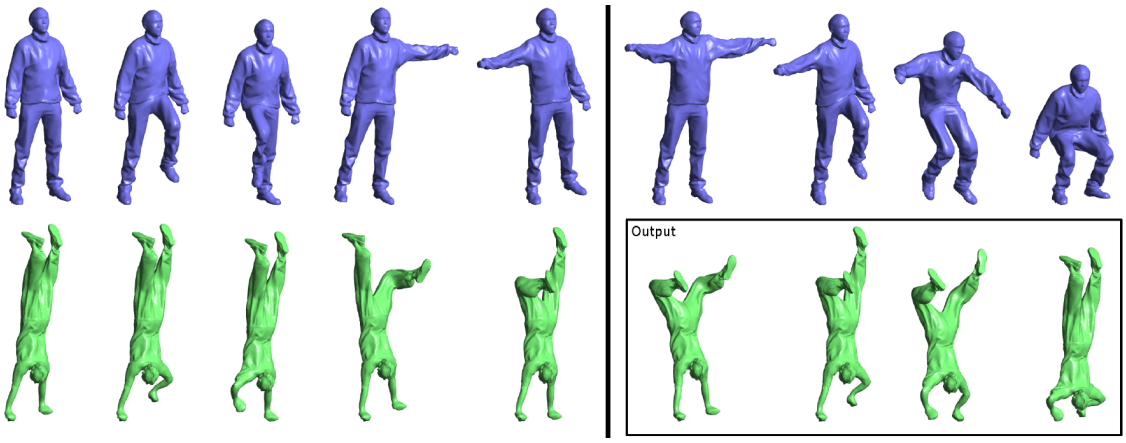


FIGURE 1.1 – Extraction de squelette [25]. A partir d'une séquence de maillage existante (à gauche), un squelette d'animation ainsi que sa paramétrisation est calculé et l'animation peut être reconstruite (à droite).

(a)



(b)

FIGURE 1.2 – Exemple de transfert de poses. (a) [46] : la ligne du haut est l'animation source, et le maillage avant la flèche correspond à la première image de la séquence. Le maillage de la deuxième ligne avant la flèche est le maillage cible dans la pose de référence. La deuxième ligne correspond au transfert de pose sur le maillage cible. (b) [11] : Etablissement de 5 correspondances entre deux maillages sur la partie gauche (phase d'apprentissage). A droite, transfert de pose d'une séquence sur une autre séquence avec en haut la source et en bas la cible.

– La recherche et l'indexation de sous-séquences [70].
– Le résumé de séquence de maillage [32] : le but est d'identifier quels sont les maillages représentant le mieux les actions présentes dans la séquence.

Enfin, une application prometteuse est l'analyse de mouvement sur les séquences de maillages, proposant des interactions complexes, comme le propose Microsoft avec sa technologie Kinect en 2D plus profondeur.

1.2 Objectifs de la thèse

Comme le montre la section précédente, l'usage des séquences de maillages est important et les applications se développent. Les projets de recherche utilisant de telles séquences sont de plus en plus nombreux comme les projets 3DTV [1] et Morpho [3] [3]. L'étude de ces séquences va se développer et les applications mentionnées auparavant vont être perfectionnées, de nouvelles applications vont être proposées. Cependant, les séquences générées avec un modeleur et celles générées par acquisition multicaméras par exemple sont de natures différentes comme nous le verrons dans le chapitre 2. Les algorithmes traitant des séquences de maillages ne sont pas forcément à même de prendre en entrée les deux types de séquences, ou bien ces algorithmes peuvent être optimisés pour un type de séquence donné. Le premier problème lorsque l'on travaille avec des séquences de maillages est donc un problème de nomenclature : il n'existe pas de terme pour désigner de manière précise les propriétés intrinsèques d'une séquence qui sont nécessaires pour le bon fonctionnement d'un algorithme. De plus, de nombreuses applications parmi celles citées ci-dessus peuvent utiliser un prétraitement similaire : la segmentation. Ce prétraitement consiste à découper une surface en fonction de certaines propriétés (voir la section 3 pour une définition rigoureuse).

Dans cette thèse, nous proposons deux contributions majeures :

– comme pour les séquences de segmentation, il n'y a pas unicité du concept de segmentation. La segmentation est un pré-traitement commun à un grand nombre d'applications, qui ont des besoins différents : il en découle que le résultat de la segmentation est différent en fonction de l'application visée. Nous formalisons les différents types de segmentation, et posons une nomenclature précise permettant de désigner aisément la sortie d'une méthode de segmentation.
– Nous proposons également plusieurs méthodes de segmentation de natures différentes.

Cette thèse est réalisée dans le cadre du projet ANR Madras [4] [2], projet visant à proposer de nouvelles méthodes de segmentation des maillages statiques et des séquences de maillages ainsi que des critères d'évaluation de celles-ci.

3. Projet ANR numéro ANR-10-BLAN-0206
4. Projet ANR numéro ANR-07-MDCO-015

1.3 Plan

Cette thèse est divisée en 7 chapitres. Dans le chapitre 2 nous définissons ce qu'est une séquence de maillages, quelles en sont les propriétés et posons les notations utilisées dans les parties suivantes (rappelées dans la section Notations). Nous y présentons donc les différents types de séquences de maillages, définis rigoureusement, ainsi que les principales méthodes de génération pour chaque type de séquence. Le chapitre 3 est consacré à la formalisation des différentes segmentations possibles de séquences de maillages. Nous posons les concepts liés aux différents types de segmentations. Le chapitre 4 introduit une première méthode de segmentation globale, fondée sur la fusion de régions. Le chapitre 5 présente deux méthodes de segmentation fondées sur la l'analyse spectrale. La première produit une segmentation globale, tandis que la seconde génère soit une segmentation globale, soit une segmentation temporellement variable. Le chapitre 6 présente un système d'évaluation quantitative des méthodes de segmentation. Finalement, le dernier chapitre (7) clôt cette thèse en présentant les perspectives sur les méthodes de segmentation et sur les applications de la segmentation.

Chapitre 2

Définition et classification de séquences de maillages

Sommaire

2.1 Introduction **7**
2.2 Rappels et travaux antérieurs **8**
2.2.1 Maillage 8
2.2.2 Séquence de maillage : nomenclature existante 10
2.3 Séquences de maillages : formalisme **11**
2.3.1 Définition générale d'une séquence de maillages 11
2.3.2 Maillage dynamique 12
2.3.3 Séquence stable de maillages 13
2.3.4 Séquence non contrainte de maillages 15
2.3.5 Limitations de la classification courante 15
2.3.6 Hiérarchie 17
2.4 Conclusion **20**

2.1 Introduction

Les séquences de maillages, comme il est mentionné dans le chapitre 1, sont utilisées couramment dans les domaines du numérique, comme le multimédia ou la simulation. Leur importante utilisation implique donc que de telles séquences puissent être facilement générées. Ces dernières années, les méthodes de création se sont multipliées, comme nous le verrons dans ce chapitre.

Les séquences de maillages peuvent être crées de différentes manières. Nous présentons ici rapidement trois méthodes de création de séquences de maillages. La méthode la plus courante actuellement est l'utilisation d'un *modeleur* tel que *Maya*[1], *3DS Max*[2] ou *Blender*[3], où un maillage de base est déformé (voir la section 2.3.2.2). La deuxième méthode de création de séquences de maillages est l'animation procédurale (simulation de fluide, d'animation de tissu), où chaque maillage de la séquence est généré, parfois de manière indépendante, à partir d'une formulation mathématique du problème. Enfin, la troisième méthode permettant de produire des séquences de maillages est l'utilisation de la reconstruction multicaméras : les objets d'intérêt sont filmés par plusieurs caméras. A partir des vidéos, un maillage est reconstruit à chaque pas de temps.

Ces différentes méthodes de création produisent des séquences qui sont de natures différentes. Les objectifs de ce chapitre sont donc d'analyser les caractéristiques intrinsèques de ces séquences et de proposer une nomenclature. Celle-ci doit permettre de désigner aisément quels types de séquences un algorithme est capable de traiter, et ainsi de préciser ses contraintes.

2.2 Rappels et travaux antérieurs

Dans cette section, nous rappelons dans un premier temps la définition de maillage, ainsi que des propriétés que nous utilisons dans la suite de ce mémoire. Nous présentons ensuite la classification actuellement utilisée.

2.2.1 Maillage

2.2.1.1 Définition

Dans le reste de ce manuscrit, nous utilisons la définition suivante de maillage (présentée par Shamir [58]) :

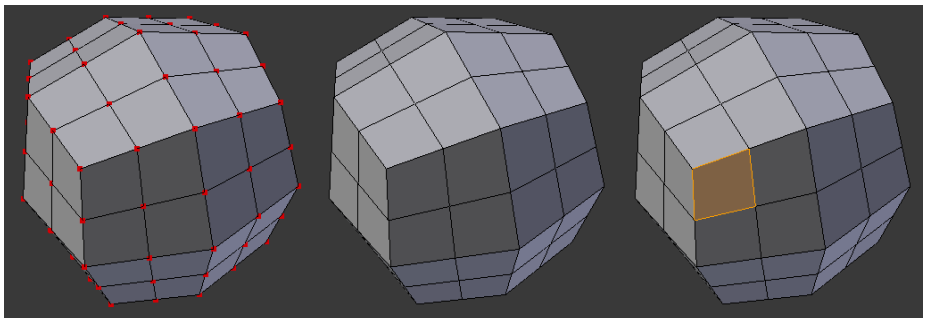
Définition 2.1. *Un maillage M avec m sommets est un triplet (V, E, F), avec :*
- *V : les sommets $= \{P_i \mid P_i \in \mathbb{R}^3, 1 \leqslant i \leqslant m\}$;*
- *E : les arêtes$= \{(P_i, P_j) \mid P_i, P_j \in V\}$;*
- *F : les faces$= \{< P_i, P_j, P_k, \ldots > \mid P_i, P_j, P_k, \ldots \in V\}$.*

Un maillage est donc défini à partir d'un ensemble de sommets. Deux sommets sont reliés par une arête. Une face est un cycle d'arêtes. L'image 2.1 présente un maillage.

2.2.1.2 Propriétés sur les maillages

Les ensembles E et F permettent de définir la connectivité du maillage. Il est important de noter que les informations fournies par E et F sont redondantes,

1. http://www.autodesk.com/maya
2. http://www.autodesk.com/3dsmax
3. http://www.blender.org



(a) (b) (c)

FIGURE 2.1 – Exemple de maillage quadrangulaire. (a) les sommets sont les éléments rouges. (b) les arêtes sont les traits noirs reliant les sommets. (c) le carré jaune est une face.

ou avec un large recouvrement si un maillage est un complexe [13, 54] (voir la définition 2.3).

Définition 2.2. *La connectivité d'un maillage est l'ensemble des relations de voisinage entre les sommets du maillage.*

La connectivité d'un maillage est donc définie à partir de ses arêtes et de ses faces. Les faces peuvent être de taille quelconque, mais les maillages triangulaires (trois sommets par face) et quadrangulaires (quatre sommets par face) sont les plus courants. Le nombre d'arêtes issues d'un sommet définit sa valence.

De plus, les sommets peuvent avoir des propriétés qui leur sont attachées. Parmi les propriétés les plus couramment utilisées, on trouve les normales, les couleurs et les coordonnées de texture. Notre but est, pour les différents traitements et applications proposés, d'utiliser uniquement la position des sommets (géométrie) et les normales (qui peuvent être calculées à partir de la géométrie et de la connectivité). Ces informations sont en effet les seules qui sont toujours disponibles, contrairement à la couleur par exemple.

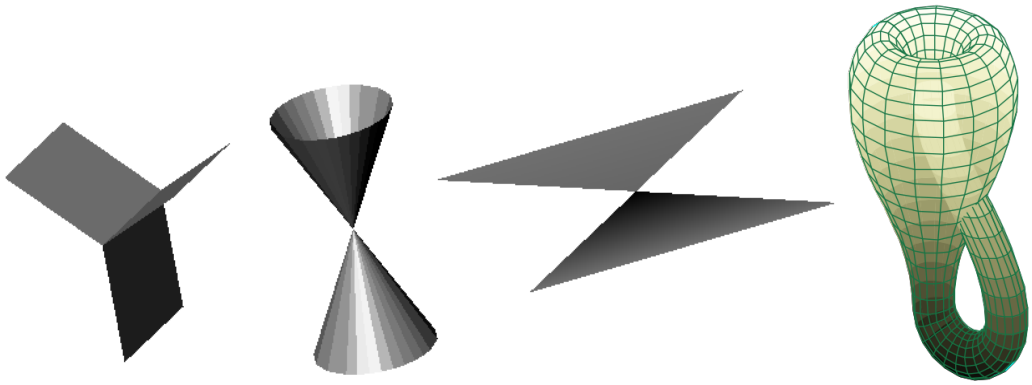
Une propriété souvent prépondérante sur les maillages est la propriété de 2-variété à bord, voir la figure 2.2 :

Définition 2.3 (Variété)**.** *Un maillage est une 2-variété à bord si le voisinage de tout sommet est homéomorphe à :*
- *soit un demi-disque pour les sommets dits du bord ;*
- *soit un disque pour les autres sommets.*

Une variété à bord ne possédant pas de sommet homéomorphe à un demi-disque est appelée 2-variété, ou 2-variété sans bord. Un maillage qui n'est pas une 2-variété à bord est dit complexe ou non-variété.

Une deuxième propriété importante sur les maillages est la notion d'orientabilité, voir la figure 2.2 :

Définition 2.4. *Un maillage M est orientable s'il sépare $\mathbb{R}^3 \setminus M$ en deux sous-espaces disjoints : l'intérieur et l'extérieur.*



(a) (b)

FIGURE 2.2 – 2-variété et orientabilité. (a) 3 exemples de maillages complexes. (b) Maillage variété non orientable (Wikipedia).

Cette définition d'orientabilité est valide dans le cas de maillage 2-variété. Une définition plus générale, valide pour les complexes est :

Définition 2.5. *Une variété topologique nD est dite orientable s'il est possible de définir une direction gauche et droite globalement en tout point de la variété.*

Nous ne posons pas de contraintes spéciales sur ces maillages (variété avec/sans bord ou complexes) hormis qu'ils soient triangulaires. Lorsque des contraintes supplémentaires sont nécessaires, nous le signalons explicitement. Nous utilisons également les définitions suivantes :

Définition 2.6. *Un maillage est dit dense par rapport à un autre maillage s'il décrit la même surface sous-jacente avec un plus grand nombre de sommets par unité de surface.*

Définition 2.7. *Un maillage est dit régulier si tous les sommets ont la même valence. Un maillage ne présentant pas cette propriété est appelé maillage irrégulier.*

Définition 2.8. *Un maillage est dit uniforme si la distribution des sommets sur la surface est homogène, i.e. il y a la même densité de sommets par unité de surface. Un maillage non-uniforme peut-être adaptatif : la distribution des sommets peut alors être adaptée à la courbure par exemple.*

Définition 2.9. *Le n-voisinage d'un sommet s est l'ensemble des sommets s_i tel que la longueur du plus court chemin entre les sommets s et s_i soit inférieure ou égale à n arêtes.*

2.2.2 Séquence de maillage : nomenclature existante

La nomenclature existante prévoit principalement deux cas :
- maillage statique : il s'agit d'un seul maillage ;

– maillage dynamique : collection ordonnée de maillages.

L'appellation maillage dynamique implique historiquement une connectivité constante, car il s'agit des premières séquences traitées. Cependant, il est souvent précisé qu'il s'agit d'une connectivité constante. Les séquences de maillages ne présentant pas cette caractéristique n'ont pas réellement de termes associés, provoquant des difficultés d'utilisation de cette nomenclature.

2.3 Séquences de maillages : formalisme

2.3.1 Définition générale d'une séquence de maillages

Le terme "séquence de maillages" est proposé comme un terme générique pour désigner une succession de maillages ordonnée dans le temps. Nous posons comme unique contrainte que le ou les objets représentés soient animés avec une cohérence temporelle minimum : il n'y a pas de discontinuité dans l'animation. La notion de continuité implique :

– l'échantillonnage temporel lors de l'acquisition est suffisamment élevé pour avoir une animation fluide ;
– l'animation est cohérente (un maillage représentant un homme à un instant k ne devient pas un chien à l'instant $k+1$ par exemple).

Cette définition est bien évidemment insuffisante pour désigner précisément les propriétés d'une séquence. Nous proposons donc de nouvelles définitions. Notons :

– N : nombre d'images composant la séquence ;
– F_k, $0 \leqslant k \leqslant N-1$: $k^{ième}$ image, pouvant contenir un maillage avec plusieurs composantes connexes ;
– $nv(F_k)$: nombre de sommets dans F_k, indexés de 0 à $nv(F_k)-1$;
– $v_i^{(k)}$, $0 \leqslant i \leqslant nv(F_k)-1$: sommet d'index i dans F_k ;
– Ng$(v_i^{(k)})$: 1-voisinage du sommet $v_i^{(k)}$.

Dans la suite de cette thèse, nous réservons l'utilisation des indices k et l pour les images tandis que les indices i et j dénotent les sommets dans les images.

Les principaux paramètres pouvant varier dans la séquence sont le nombre de sommets, la connectivité des maillages et la topologie des objets sous-jacents aux maillages. Une topologie constante implique que la séquence ne présente pas de changement de topologie des maillages de la séquence : il ne peut donc pas y avoir de séparation, fusion, d'apparition ou disparition de géometrie. Ces paramètres sont corrélés comme nous le verrons lorsque nous présentons les différentes combinaisons. Après avoir retiré les combinaisons impossibles ou n'ayant pas de sens, il reste trois types de séquences, dont les caractéristiques sont résumées dans le tableau 2.1.

Les combinaisons non retenues sont présentées dans le tableau 2.2. Les trois premières lignes du tableau sont impossibles pour la même raison : la connectivité constante implique que le nombre de sommets et la topologie soient constants. Les derniers cas sont des séquences jugées atypiques. Nous pensons que, dans

Nombre de sommets	Connectivité	Topologie	Nom
constant	constante	constante	Maillage dynamique
variable	variable	constante	Séquence stable de maillages
variable	variable	variable	Séquence non contrainte de maillages

TABLE 2.1 – Séquences de maillages : propriétés possibles et noms associés.

ce cas là, il est plus opportun de se ramener à un cas courant :

- pour le quatrième cas : soit considérer la séquence comme une séquence non contrainte de maillages, soit effectuer un remaillage sur chaque maillage de la séquence pour obtenir un maillage dynamique ;
- pour le dernier cas : considérer la séquence comme une séquence non contrainte de maillages.

Nous utilisons le terme *image* pour désigner le maillages (qui peut être non connexe comme dans le cas de plusieurs personnes en interactions) présents dans l'image.

2.3.2 Maillage dynamique

2.3.2.1 Définition

Le premier type de séquences que nous définissons est le *maillage dynamique*, noté DM (pour Dynamic Mesh) par la suite. Une séquence de maillages est appelée maillage dynamique si et seulement si la connectivité du maillage est constante au cours de la séquence. Cette définition implique :

- $\forall k, l \in [0, N[, nv(F_k) = nv(F_l)$ (strictement équivalent à $\forall k \in [0, N[, nv(F_k) = nv(F_0)$) ;

Nombre de sommets	Connectivité	Topologie	Raisons
constant	constante	variable	Impossible : un changement de topologie implique un changement de connectivité
variable	constante	constante	Impossible : une variation du nombre de sommets implique un changement de connectivité
variable	constante	variable	Impossible : une variation du nombre de sommets implique un changement de connectivité
constant	variable	constante	Atypique : se ramener à un maillage dynamique ou une séquence non contrainte de maillages
constant	variable	variable	Atypique : Se ramener à une séquence non contrainte de maillages

TABLE 2.2 – Séquences de maillages invalides.

- $\forall k, l \in [0, N[,\ \forall i \in [0, nv(F_k)[,\ \mathrm{Ng}(v_i^{(k)}) = \mathrm{Ng}(v_i^{(l)})$ (strictement équivalent à $\forall k \in [0, N[,\ \forall i \in [0, nv(F_k)[,\ \mathrm{Ng}(v_i^{(k)}) = \mathrm{Ng}(v_i^{(0)})$) ;
- Il ne peut pas y avoir de changement de topologie ;
- Seule la position des sommets varie au cours du temps.

2.3.2.2 Génération

Certaines méthodes d'animation procédurale telles que la simulation de tissu produisent des maillages dynamiques. La méthode de création de maillage dynamique la plus utilisée, l'animation par squelette, passe par un modeleur : elle fait partie des méthodes d'animation par images clés, où l'animateur déforme le maillage à différents instants k et l, les images intermédiaires étant *interpolées* à partir de ces deux images.

Son principe est simple, elle nécessite un maillage et un squelette composé d'os, voir la figure 2.3. L'animation par squelette se fait en 2 étapes :

1. Attachement du squelette à la surface : le squelette est attaché au maillage. Un poids $p_{i,j}$ est affecté à chaque paire (Sommet v_i, os O_j), ce poids représente l'influence de l'os O_j sur le sommet v_i : plus le poids est important, plus le mouvement de l'os modifie la position du sommet. La position d'un sommet est donc dépendante de l'ensemble des os du squelette. Cette étape peut être en grande partie automatisée [33].
2. Afin d'animer le maillage, l'animateur manipule directement le squelette. Celui-ci est donc une représentation simplifiée du maillage. La position de chaque sommet v_i est alors une combinaison des transformations de chaque os O_j du squelette pondérées par le poids $p_{i,j}$. Ainsi à chaque déplacement d'un os, le maillage est déformé.

En posant les images clés sur les différentes positions du squelette, le maillage est déformé par interpolation des transformations des os du squelette entre chaque image de référence.

2.3.3 Séquence stable de maillages

2.3.3.1 Définition

Une séquence stable de maillages, notée SMS (pour Stable Mesh Sequence) par la suite, peut avoir un nombre de sommets ainsi qu'une connectivité variables entre les maillages de chaque image. La seule restriction porte sur la topologie des objets sous-jacents : celle-ci doit rester constante tout au long de la séquence. La figure 2.4 montre deux maillages successifs d'une SMS.

2.3.3.2 Génération

Les séquences stables de maillages peuvent être produites par des méthodes d'animation procédurale ou de simulation numérique. Ce type de séquences peut être obtenu par une segmentation temporelle sur une séquence non contrainte, comme nous le verrons dans le chapitre 3.

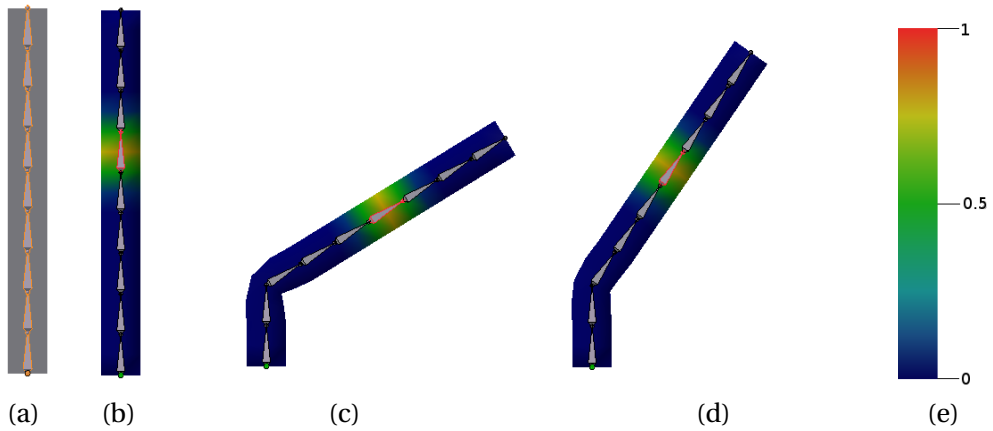


FIGURE 2.3 – Illustration de la méthode d'animation par squelette. (a) Maillage de base et son squelette. (b) Poids associés à un os. La couleur indique la pondération des sommets par rapport à l'os sélectionné en orange : plus la couleur tend vers le rouge, plus le poids est important. (c) Image clé à l'image 10. (d) Image intermédiaire interpolée. (e) Echelle de couleur : pour un poids de 0, l'os n'as pas d'influence sur le sommet, pour un poids de 1, l'os à une influence maximale sur le sommet.

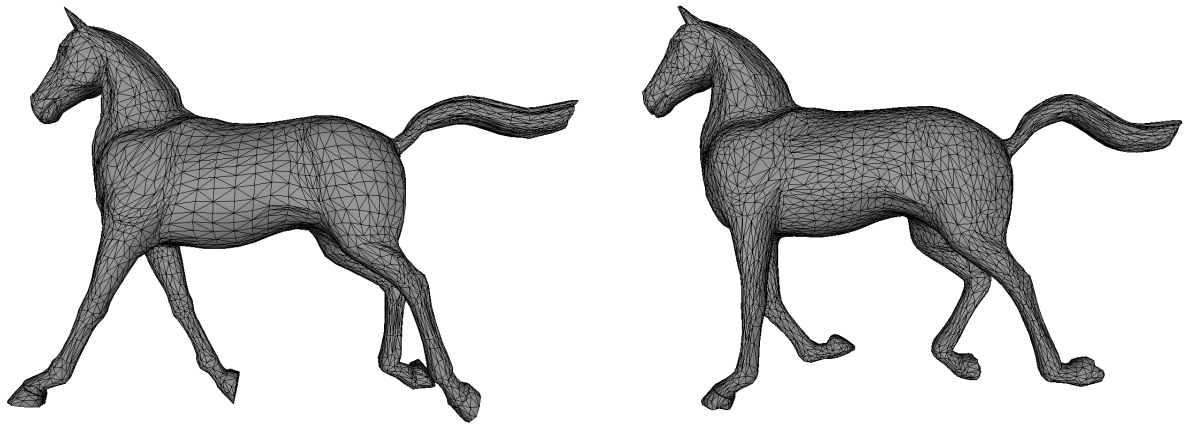


FIGURE 2.4 – Exemple de SMS.

2.3.4 Séquence non contrainte de maillages

2.3.4.1 Définition

Une séquence non contrainte de maillages, notée UMS (pour Unconstrained Mesh Sequence) par la suite, n'impose aucune restriction sur le nombre de sommets, la connectivité ou la topologie du maillage au cours de la séquence : l'animation représente un ensemble de maillages sans aucune relation a priori entre eux.

2.3.4.2 Génération

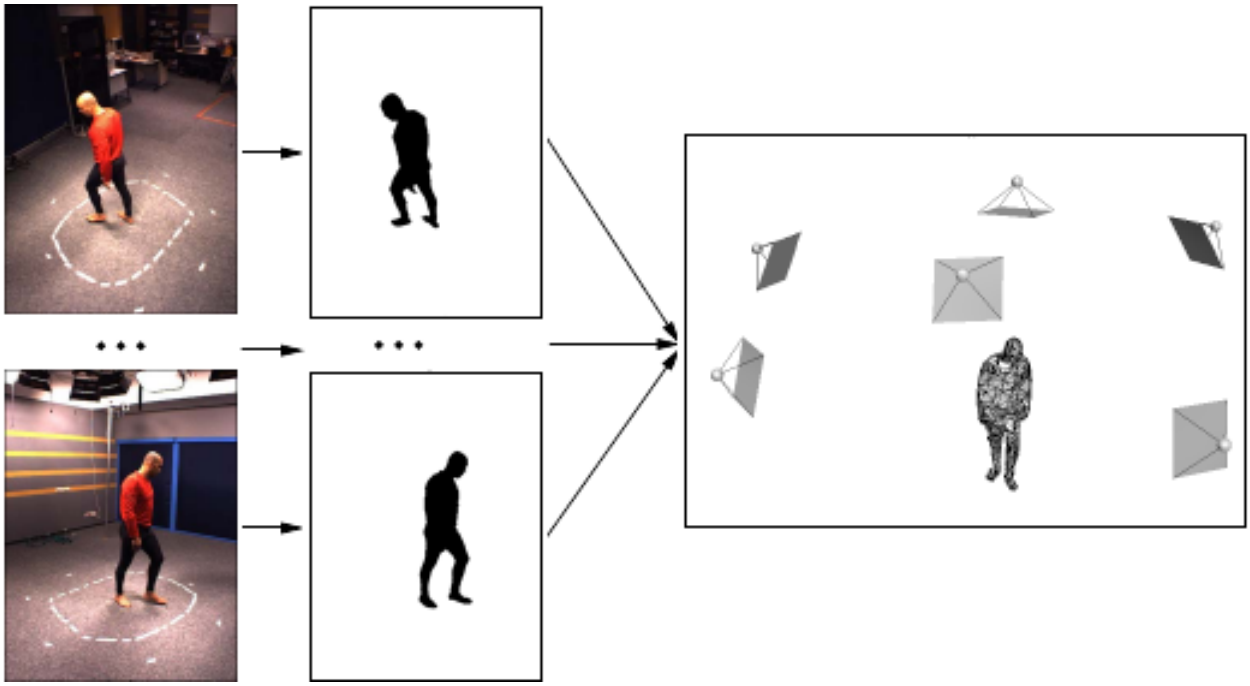
Ces séquences peuvent être créées en utilisant les méthodes d'animations procédurales telles que la simulation de fluides. Elles sont cependant le plus souvent générées par des systèmes d'acquisition multicaméras. Nous détaillons ici le principe des systèmes de reconstruction multicaméras, voir la Figure 2.5a pour un exemple. Le but de ces méthodes est de reconstruire un maillage en procédant en deux étapes. La première est l'acquisition : plusieurs caméras fixes filment le sujet et une soustraction de fond sur chaque caméra est appliquée pour isoler le sujet. On obtient donc une silhouette pour chaque caméra. Ensuite commence l'étape de modélisation : on utilise une méthode de shape-from-silhouette, telle que l'algorithme du *Visual Hull* [12, 42], pour obtenir une modélisation 3D à partir des différentes silhouettes. Il existe plusieurs méthodes de reconstruction multicaméras avec différents compromis, notamment vitesse de calcul/qualité de reconstruction. Il est important de noter que ces méthodes produisent encore des maillages qui ne sont pas forcément des 2-variétés, ils peuvent avoir des auto-intersections ainsi que d'autres artefacts. De plus, la séquence reconstruite contient également des artefacts dus aux occultations comme illustré avec les figures 2.5b et 2.5c.

2.3.5 Limitations de la classification courante

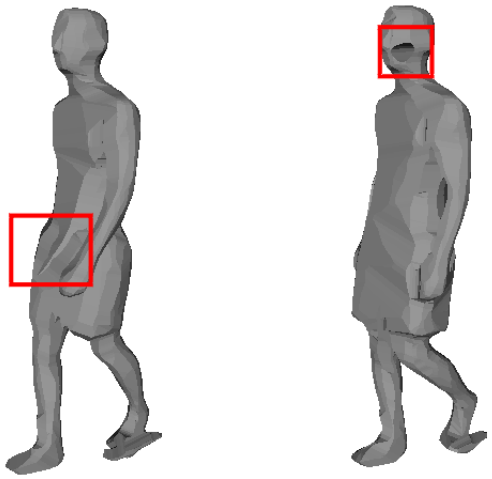
La classification proposée est insuffisante car elle est fondée uniquement sur des critères géométriques et ne prend donc pas en compte les problèmes de contenus, nous le verrons dans le chapitre 3. En effet, elle ne fait pas de différence entre les types de changement de topologie : séparation, fusion, et apparition/disparition de géométrie. Nous gardons le terme UMS pour désigner les séquences de maillages dans lesquelles des changements de topologie interviennent.

2.3.5.1 UMS avec Séparation

Cette classe contient les séquences dans lesquelles :
- la séparation en plusieurs composantes connexes d'un objet a lieu, illustrée sur la figure 2.6b ;
- la séparation provoque un changement de genre, mais pas du nombre de composantes connexes, illustrée sur la figure 2.6g



(a)



(b) (c)

FIGURE 2.5 – (a) Reconstruction multicaméras [5]. (b) et (c) Exemple de maillages reconstruits (avec la plateforme Grimage) avec des erreurs dues aux occultations : pic au niveau du bassin (b) et trou au niveau du visage (c).

Nombre de sommets	Connectivité	Topologie	Nom
constant	constante	constante	DM
variable	variable	constante	SMS
variable	variable	variable	UMS
variable	variable	fusion	UMS avec fusion
variable	variable	séparation	UMS avec séparation
variable	variable	apparition et disparition	UMS avec apparition/disparition

TABLE 2.3 – Nouvelles classification des séquences de maillages : propriétés possibles et noms associés.

2.3.5.2 UMS avec Fusion

Cette classe contient les séquences dans lesquelles :
- la fusion de plusieurs objets (i.e. composantes connexes) a lieu, illustrée sur la figure 2.6c ;
- la fusion provoque un changement de genre, mais pas du nombre de composantes connexes, illustrée sur la figure 2.6f.

2.3.5.3 UMS avec Apparition/Disparition de géométrie

Cette classe contient les séquences dans lesquelles un ou plusieurs objets, précédemment absents de la scène, apparaissent, et les séquences dans lesquelles un ou plusieurs objets initialement présents dans la scène disparaissent, comme dans la figure 2.6d.

Nous avons choisi de différencier les cas de séparation et de fusion, car ces deux changements de topologie se traitent différemment dans les algorithmes de segmentation comme nous le verrons dans la partie 3.

Il est important de noter que l'appartenance à la dernière classe proposée, apparition/disparition, ne peut être déterminée automatiquement contrairement aux autres classes. Il s'agit d'un critère de plus haut niveau d'un point de vue géométrique. En effet, l'apparition/disparition de géométrie (un nouvel objet) implique un changement de topologie, alors que l'inverse n'est pas vrai. Un opérateur humain est donc nécessaire pour distinguer ces classes.

Le tableau 2.3 présente la classification finale.

2.3.6 Hiérarchie

2.3.6.1 Imbrication des classes.

Cette classification, avant la subdivision de la classe UMS, présente un raffinement strict. En effet, on a $DM \subseteq SMS \subseteq UMS$. Avec les modifications effectuées, une séquence peut appartenir à 2 classes de même niveau. La nouvelle

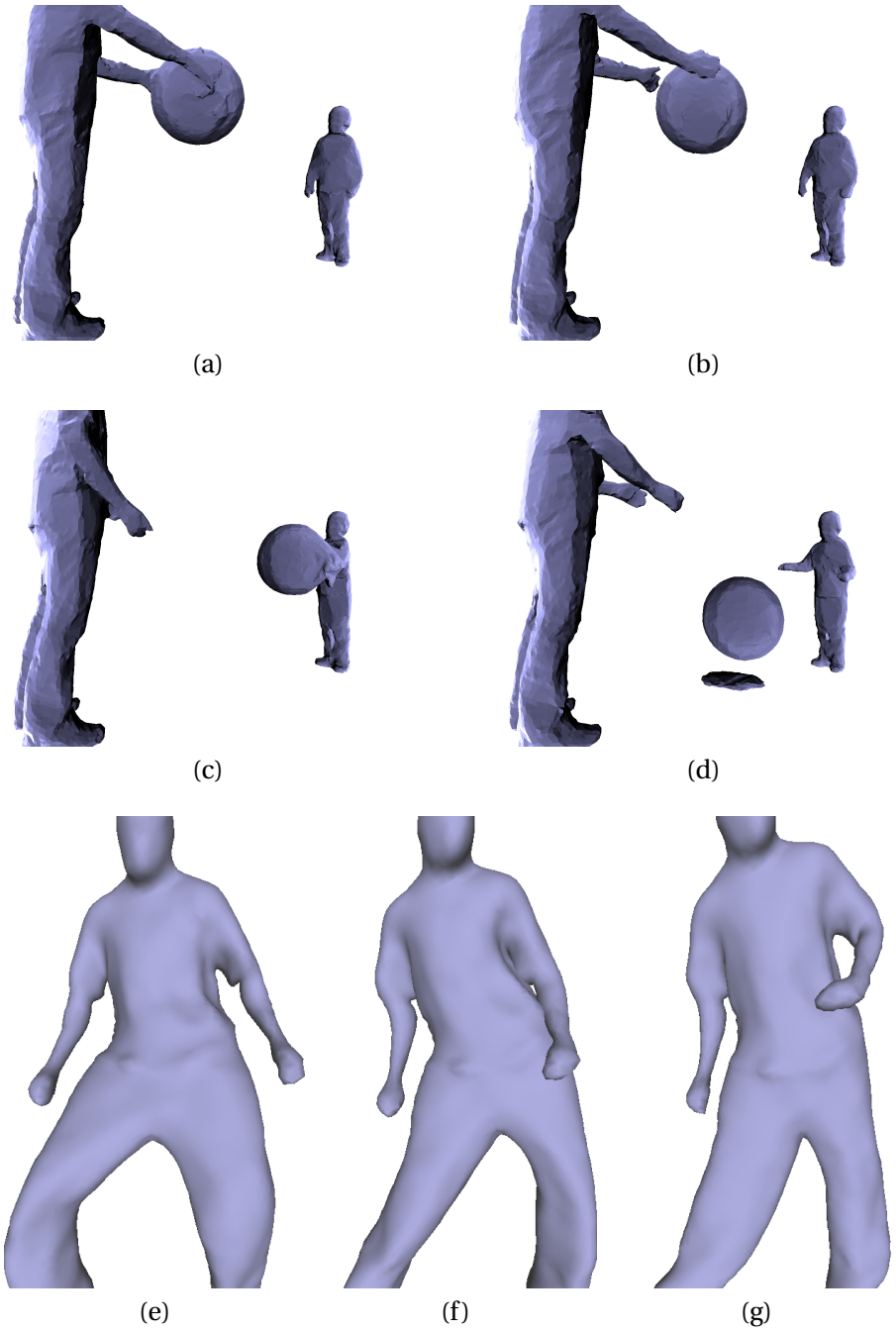


FIGURE 2.6 – Illustration des différentes classes intégrant les changements de topologie sur la séquence "Balloon" [36] et de genre sur la séquence Head [63] (a) Maillage de référence. (b) Séparation au niveau du ballon et de la main de l'homme. (c) Fusion au niveau du ballon et de la main de l'enfant. (d) Apparition et disparition de l'ombre du ballon. (e) Maillage de référence. (f) Fusion au niveau du contact bras-hanche sans changement du nombre de composantes connexes. (g) Séparation au niveau du contact bras-hanche sans changement du nombre de composantes connexes.

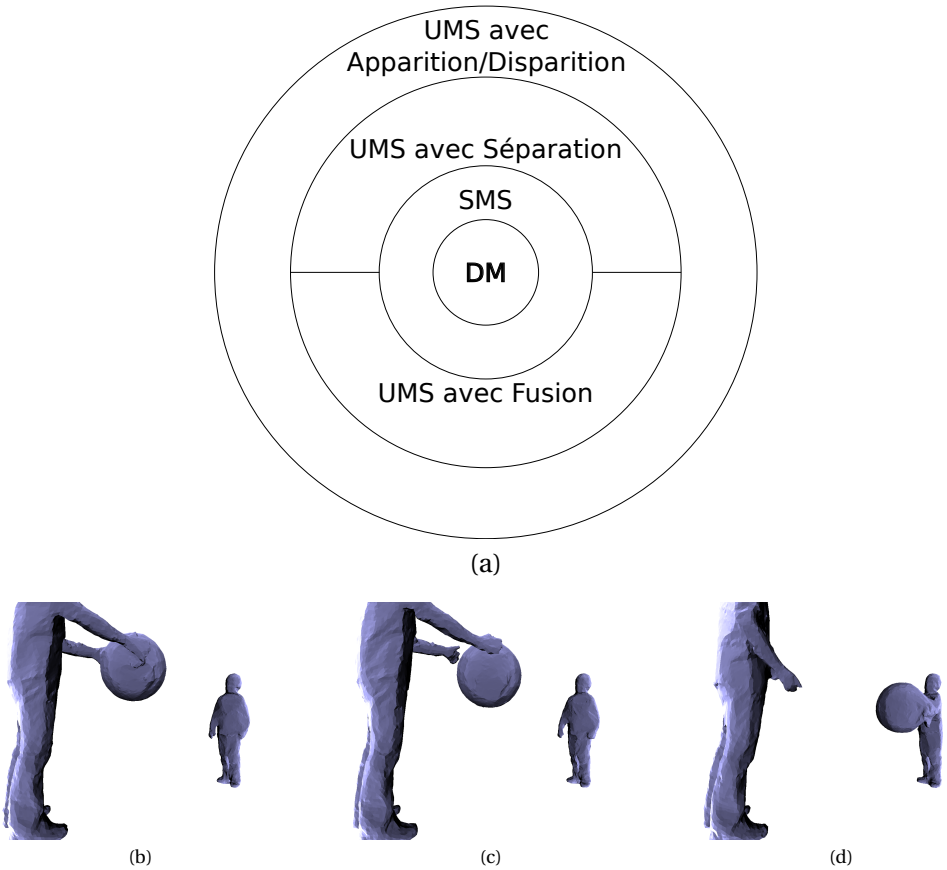


FIGURE 2.7 – Imbrication et hiérarchie des classes. (a) Nouvelle hiérarchie. (b-d) Exemple de séquence appartenant à deux classes de même niveau : à l'image (c), la séquence appartient à la classe UMS avec Séparation (séparation du ballon) tandis qu'à l'image (d), la séquence appartient à la classe UMS avec Fusion (fusion du ballon). La séquence appartient donc à ces deux classes.

hiérarchie est décrite dans la figure 2.7a. Avec la nouvelle classification, une séquence peut appartenir à deux classes du même niveau comme illustré dans les figures 2.7b–d.

2.3.6.2 Décomposition en sous-séquences.

Comme mentionné précédemment, les classes de séquences sont hiérarchiques. En subdivisant les séquences, il est donc possible dans certains cas de se ramener à un ensemble de séquences de classes inférieures. Ainsi, pour le passage de :

- UMS vers SMS : à chaque changement de topologie, il suffit de créer une nouvelle séquence ;
- UMS avec apparition/disparition vers SMS : à chaque apparition/disparition d'objet, il suffit de créer une nouvelle séquence.

Le passage de SMS vers DM est également possible, il nécessite cependant un travail plus important et est un sujet de recherche actif. De telles méthodes

de conversion sont appelées suivi de maillages.

2.4 Conclusion

Dans ce chapitre, nous avons, dans un premier temps, effectué un rappel de la définition classique d'un maillage ainsi que des notions associées. Nous avons également présenté la notion de séquence de maillages, développé une classification et proposé de nouvelles définitions permettant de désigner aisément les propriétés d'une telle séquence, utilisées pour décrire les types de séquences de maillages traitées par les différents algorithmes proposés par la suite. De plus, ce chapitre nous a permis de poser les notations utilisées dans le reste de cette thèse. Un rappel de ces notations est disponible dans le chapitre Notations.

Dans le prochain chapitre, nous allons étudier une classification des segmentations, puis dans les chapitres suivant les différentes méthodes de segmentation proposées au cours de cette thèse qui utilisent les notions présentées au cours de ce chapitre.

Chapitre 3

Classification des méthodes de segmentation de séquences

Sommaire

3.1 Introduction **21**
3.2 Classification des méthodes de segmentation **23**
3.2.1 Etat de l'art 23
3.2.2 Formalisation et raffinement de la classification 27
3.3 Etat de l'art des méthodes de segmentation **32**
3.3.1 Segmentation de maillages dynamiques 33
3.3.2 Segmentation de séquences stables et non contraintes de maillages 34

3.1 Introduction

La segmentation est un pré-traitement couramment utilisé dans le domaine de l'image. En 2D, de très nombreuses méthodes de segmentation pour les images ont été proposées. De manière informelle, la segmentation d'une image consiste à partitionner une image en différentes régions suivant un ou plusieurs critères. Les applications de la segmentation d'images sont nombreuses : parmi les utilisations les plus usuelles, on trouve le domaine médical [55], la détection de visage [74]. La figure 3.1 montre plusieurs exemples d'images segmentées.

Ils existent différentes grandes familles de méthodes de segmentation, telles que les méthodes fondées sur la détection d'arêtes, la segmentation par ligne

1. http://vision.ece.ucsb.edu/segmentation/edgeflow/examples.html

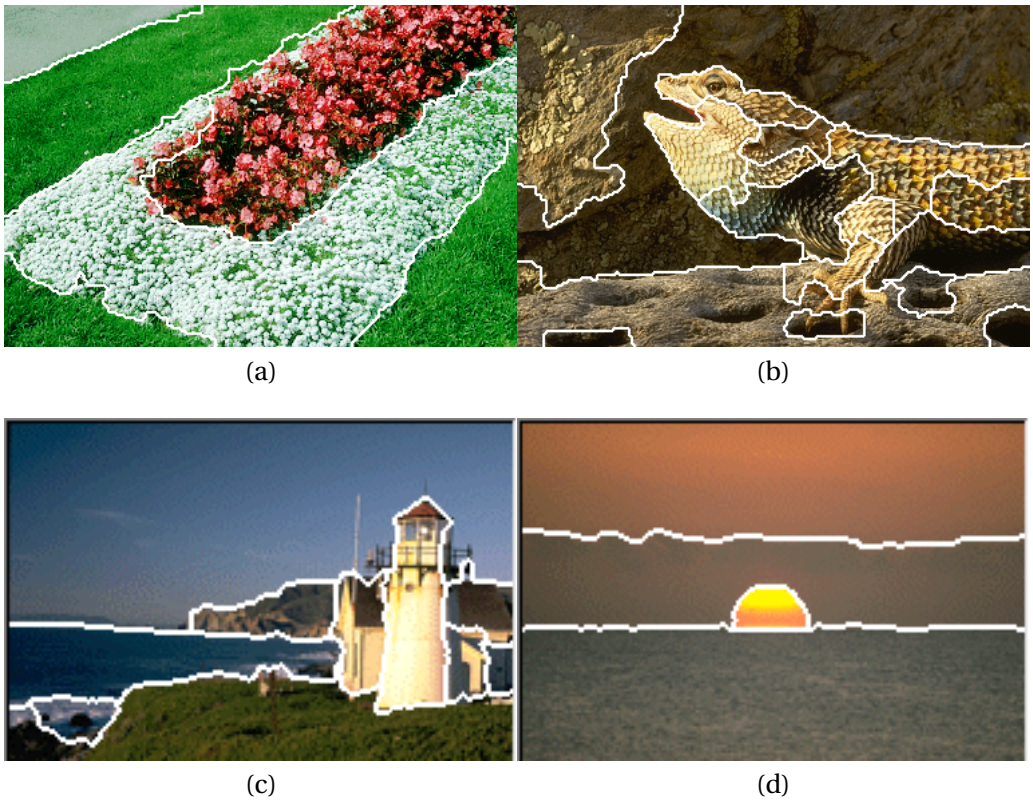


FIGURE 3.1 – Exemple d'images segmentées [1].

de partage des eaux [17], les méthodes fondées sur les graphes [62] ainsi que d'autres [52].

La segmentation de maillages statiques est un sujet de recherche plus récent, mais en raison des nombreuses applications possibles, un nombre important de méthodes ont également été présentées. Ainsi, la segmentation de maillage statique est utilisée par exemple dans les applications suivantes : le placage de texture [48], la compression [39], la simplification [28] ou l'indexation [75]. Ces traitements de nature différente impliquent donc une segmentation différente : le résultat et la méthode de segmentation sont donc dépendants du type de traitement voulu [58].

L'utilisation de séquence de maillage est plus récente, et donc la segmentation comme prétraitement également. La segmentation de séquence de maillages est utilisée par exemple pour la compression ([6, 47]), l'extraction de squelettes [25, 65], la segmentation en sous-séquences [70], ainsi que le transfert de pose [46], et l'indexation et la recherche dans une base de données [70]. Une application prometteuse est l'analyse de mouvement (similaire à la technologie *Kinect* de Microsoft qui est un système de capture travaillant en 2D + profondeur).

Cependant, ces méthodes de segmentation ont été développées sans cadre formel. Les segmentations nécessaires pour la compression et la segmentation en sous-séquences sont deux segmentations de nature différente par exemple. Bien que les entrées soient les mêmes, le résultat attendu est différent. Notre

but dans ce chapitre est donc dans un premier temps, de poser un cadre formel et de proposer une nomenclature pour les différents types de segmentation. Cette nomenclature doit permettre de désigner aisément quel type de segmentation est produit par un algorithme, ainsi que le type de segmentation nécessaire pour un traitement de plus haut niveau. Dans un second temps, nous nous intéresserons à l'état de l'art : quelles sont les méthodes existantes ainsi que leur place dans la nomenclature proposée.

Le plan du chapitre est donc le suivant : en section 3.2 une proposition de classification et nomenclature des méthodes de segmentation de séquences de maillages. La section 3.3 présente l'état de l'art des méthodes de segmentation, et leur placement vis-à-vis de notre classification.

3.2 Classification des méthodes de segmentation

La segmentation consiste à séparer un ensemble en plusieurs sous-ensembles disjoints. Plus formellement :

Définition 3.1. *[58] La segmentation d'un ensemble P consiste en la séparation en n sous-ensembles $P = \{P_0, \ldots, P_{n-1}\}$ tels que :*

$$\begin{cases} \bigcup_{p=0}^{n-1} P_p = P \\ P_q \bigcap P_p = \varnothing \quad \forall q \neq p \end{cases}$$

De plus, si l'ensemble P contient une notion de voisinage, alors chaque sous-ensemble P_i est un ensemble connexe.

Chaque sous-ensemble est appelé segment, cluster ou région. La figure 3.2 montre un exemple de segmentation sur un ensemble de points avec voisinage. Dans cette thèse, nous emploierons le terme cluster.

Il existe différentes propriétés pour choisir une méthode de segmentation. Parmi celles-ci se détache la notion de segmentation hiérarchique. Une méthode de segmentation est dite hiérarchique si elle peut être raffinée, i.e chaque cluster peut-être à nouveau segmenté.

Définition 3.2. *Une segmentation $(P^{(0)}, ..., P^{(m)})$ est dite hiérarchique si :*

$$\begin{cases} P^{(0)} = P \\ \forall p, P^{(p)} \textit{ est une segmentation : } P^{(p)} = \{P_0^{(p)}, ..., P_{n-1}^{(p)}\} \\ \forall q \in [0, n[,\ P_q^{(p+1)} \textit{ est une segmentation de } P_q^{(p)} \end{cases}$$

La figure 3.3 illustre la propriété de segmentation hiérarchique.

3.2.1 Etat de l'art

La classification actuelle prévoit deux types de segmentation : l'une pour les maillages statiques, l'autre pour les séquences de maillages.

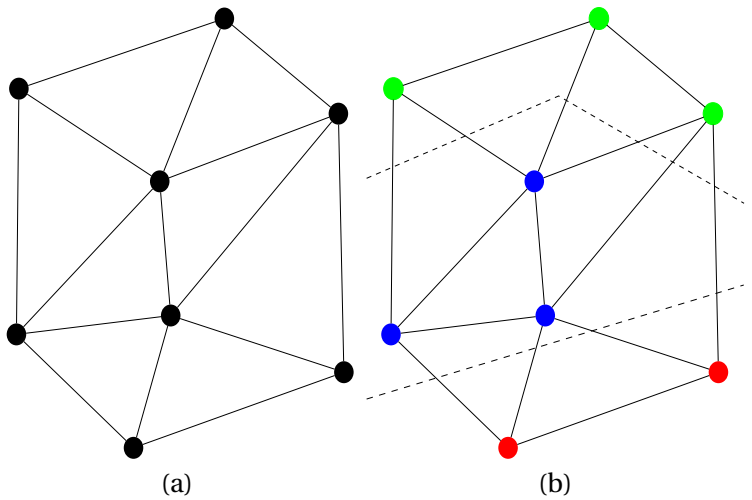


FIGURE 3.2 – Exemple de segmentation. (a) Ensemble de base, la connectivité de l'ensemble est représentée par les arêtes. (b) Segmentation possible sur cet ensemble. Les sommets de même couleur appartiennent à un même cluster, les lignes en pointillés représentent les frontières des clusters.

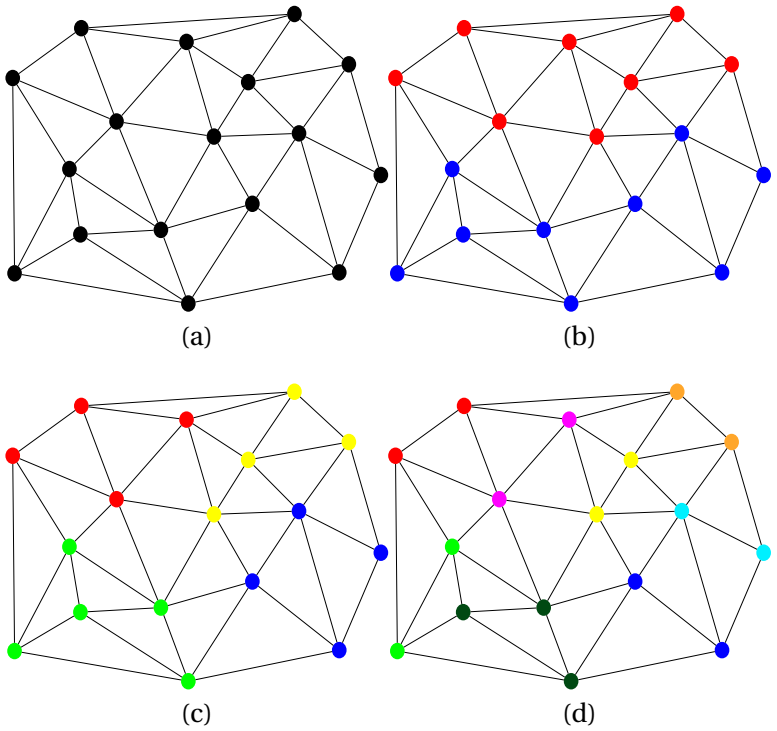


FIGURE 3.3 – Exemple de segmentation Hiérarchique. (a) Ensemble de base. (b) Segmentation de niveau 0. (c) Segmentation de niveau 1 : le cluster rouge au niveau 0 est subdivisé en deux : rouge et jaune, le cluster bleu au niveau 0 est subdivisé en deux : bleu et vert. (d) Segmentation de niveau 2.

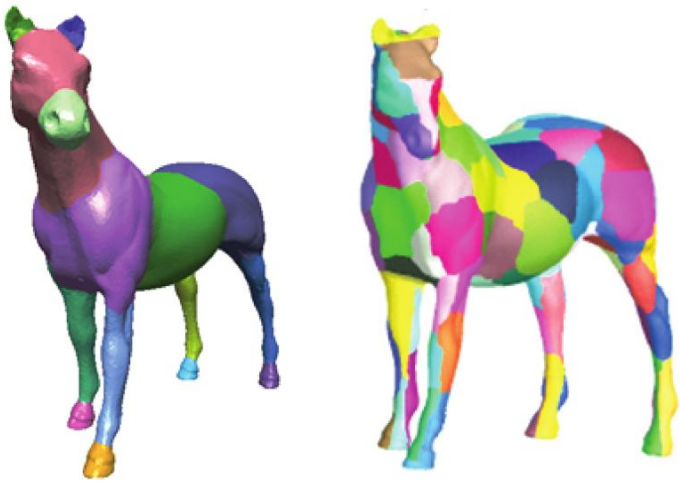


FIGURE 3.4 – Exemple de segmentation de maillage statique. A gauche, une segmentation sémantique. A droite, une segmentation géométrique [58].

3.2.1.1 Segmentation statique

Définition Dans le cas de la segmentation de maillage statique, la définition 3.1 s'applique d'elle-même. L'ensemble de départ contient les sommets du maillage et chaque sous-ensemble est un cluster de sommets ou de faces. La relation de voisinage des clusters est fondée sur le voisinage des sommets ou des faces du maillage.

Il existe de nombreuses méthodes de segmentation statique. Elles peuvent être séparées en deux grandes familles, les méthodes reposant sur :

- des critères géométriques : courbure [43], approximation par des primitives [8]....
- des critères sémantiques : le but est de segmenter un objet en entités correspondant à des parties de l'objet (les membres d'un être humain tels que les bras, les jambes...). Ce type de segmentation repose sur la perception humaine.

La figure 3.4 montre un exemple de segmentation sémantique et un exemple de segmentation géométrique sur un même modèle.

Le type de segmentation, i.e. sémantique ou géométrique, n'est pas le seul critère à prendre en compte pour une méthode de segmentation. En effet d'autre propriétés ont leur importance, par exemple l'indépendance de pose (un maillage d'un même objet dans différentes positions est segmenté de manière similaire) ou encore le nombre de paramètres nécessaires sont des critères à prendre en compte pour choisir une méthode de segmentation [9, 58].

La figure 3.5 montre un exemple de segmentation hiérarchique.

Etat de l'art des méthodes de segmentation statique Comme nous l'avons dit dans la section précédente, A. Shamir a présenté un état de l'art des méthodes de segmentation statique, dans lequel il présente les familles de méthodes (géométriques contre sémantiques, outils utilisés). Nous invitons le lecteur à s'y ré-

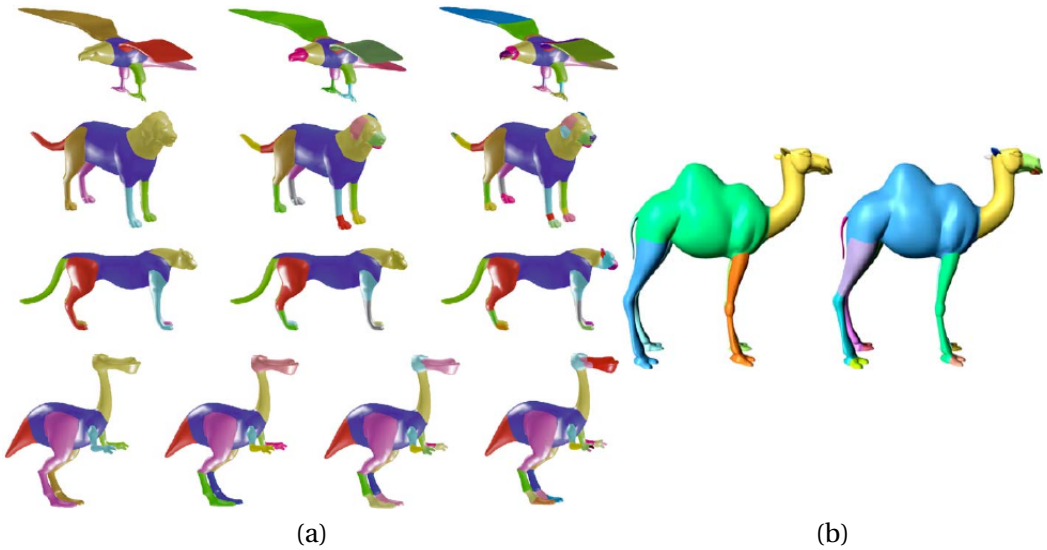


FIGURE 3.5 – Exemples de segmentation hiérarchique de maillages statiques. (a) [40]. (b) [59].

férer pour plus de détails. Nous notons simplement que depuis sa parution, l'activité de recherche s'est focalisée sur :

- la segmentation sémantique [4, 59] ;
- l'évaluation des méthodes de segmentation[14, 15, 21]. Le but de ces évaluations est de déterminer quelle méthode de segmentation il convient d'utiliser pour segmenter un objet en fonction de sa classe (objet CAD, humain, table...). L'autre objectif est de permettre l'évaluation des performances d'une nouvelle méthode face à l'état de l'art.

3.2.1.2 Segmentation de séquence de maillages

Concernant la segmentation de séquences de maillages, il n'y a pas de formalisme contrairement à la segmentation de maillages statiques : il s'agit de travaux récents, réalisés indépendamment, et personne n'a, à notre connaissance, regroupé ces travaux. On distingue deux types de segmentation, que nous formalisons (en même temps que nous proposons d'autres types de segmentations) dans la la section 3.2.2. Les deux types de segmentation sont :

- une segmentation pour l'ensemble de la séquence : les clusters sont définis à partir de l'ensemble de la séquence, de plus tous les clusters sont présents tout au long de la séquence ;
- une segmentation sur les images de la séquence : une séquence est subdivisée en sous-séquences, soit sur des critères topologiques, soit sur des critères sémantiques.

Ces deux types de segmentation sont bien évidemment de nature différente : l'une se fait sur les faces/sommets des maillages, tandis que l'autre se fait sur les images.

3.2.2 Formalisation et raffinement de la classification

Dans cette partie, notre but est de présenter une formalisation des concepts liés à la segmentation de séquences de maillages. Nous présentons donc de manière rigoureuse les deux types de segmentation précédents et en introduisons de nouveaux.

3.2.2.1 Segmentation globale

Nous définissons d'abord la segmentation globale. Ce type de segmentation utilise directement la définition 3.2.2 pour les séquences de maillages. Ce type de segmentation peut être appliqué soit sur les sommets, soit sur les faces des maillages.

Définition 3.3. *La segmentation globale d'une séquence de maillages SM consiste en la séparation en n sous-ensembles $P = \{P_0, \ldots, P_{n-1}\}$ tels que :*

$$\begin{cases} \bigcup_{p=0}^{n-1} P_p = F_0 \\ P_q \bigcap P_p = \varnothing \quad \forall q \neq p \end{cases}$$

Dans le cas d'un maillage dynamique, les clusters sont les mêmes pour chaque maillage : un cluster contient les mêmes sommets sur toute les images de la séquence. Pour les autres types de séquences de maillages, les clusters sont composés des sommets correspondants à F_0.

La relation de voisinage est définie par la connectivité des maillages.

Le paramètre temps est seulement utilisé pour calculer les clusters de la segmentation. Cela revient à effectuer une segmentation sur la trajectoire des sommets ou des faces. Comme nous le verrons par la suite, les méthodes actuelles de segmentation de maillages utilisent pour la plupart cette définition. La figure 3.6 illustre ce type de segmentation tandis que la figure 3.7 montre plusieurs exemples de segmentation globale sur des maillages dynamiques.

Un cluster présent à l'image F_k est présent sur la segmentation de toute image $F_{k'}$ de la séquence, $0 \leqslant k' < F$.

3.2.2.2 Segmentation par instances

Une segmentation par instances consiste à avoir une segmentation de maillage pour chaque instant : cela revient à effectuer N segmentations statiques, où N est le nombre d'images de la séquence. Nous posons cependant la restriction suivante : chaque segmentation doit posséder le même nombre de clusters et il doit y avoir une correspondance une à une des clusters à chaque instant.

Définition 3.4. *Soit r le nombre de régions. Une segmentation par instances P d'une séquence de maillages F est définie par :*

$$P = \bigcup_{1 \leq k \leq N} P^k$$

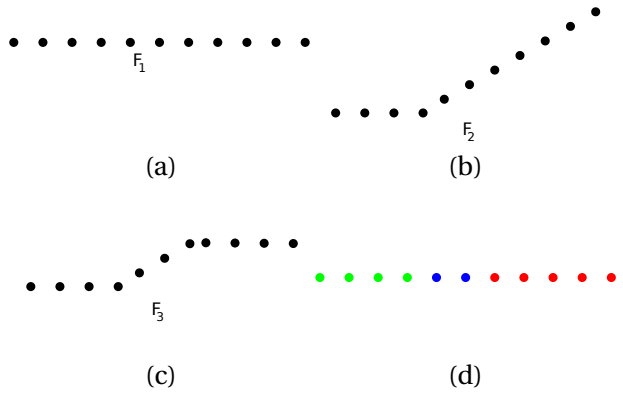


FIGURE 3.6 – Illustration de la segmentation globale. (a), (b) et (c) Position des sommets sur les images 1, 2 et 3. (d) Segmentation finale.

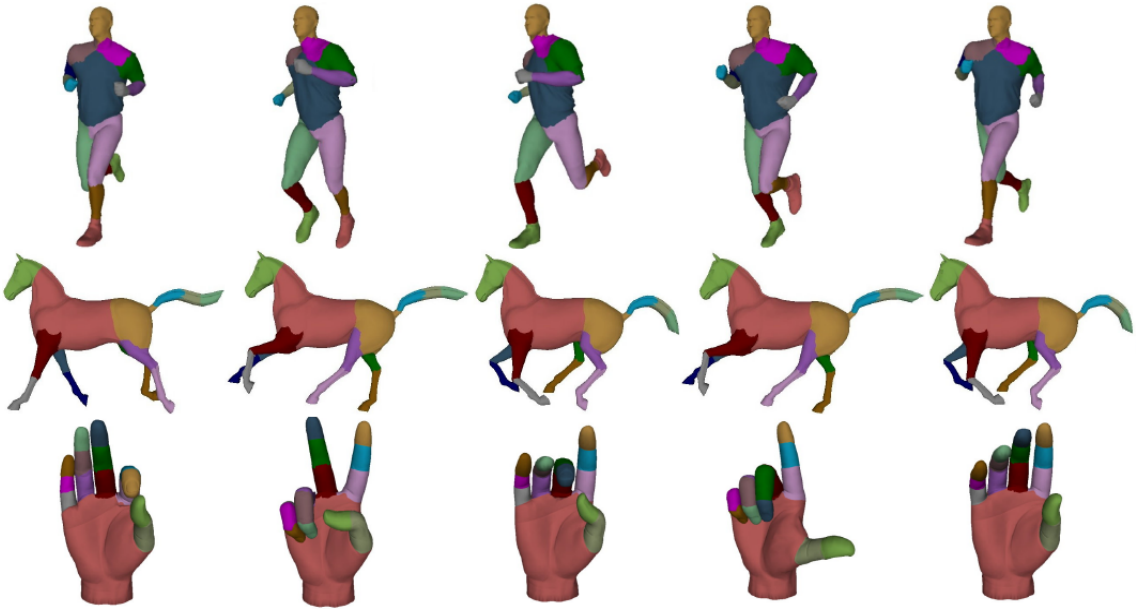


FIGURE 3.7 – Exemples [46] de segmentations globales sur différents maillages dynamiques.

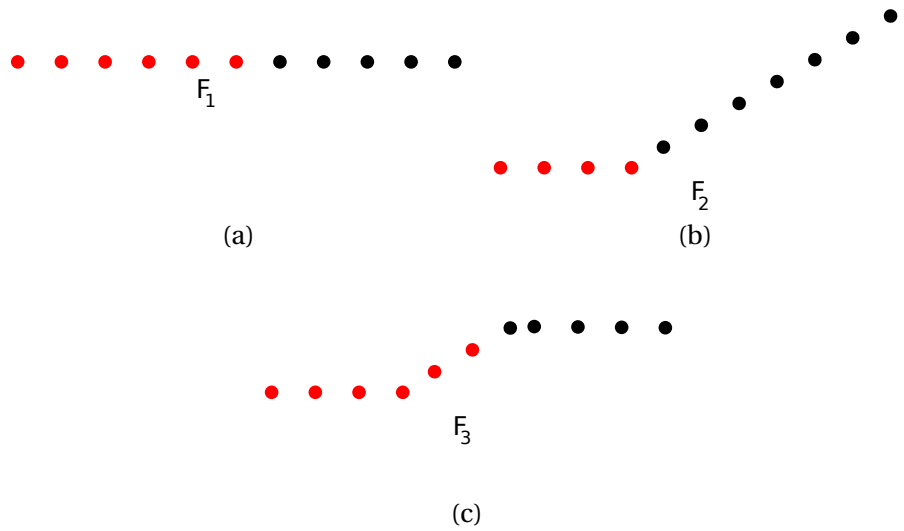


FIGURE 3.8 – Illustration d'une segmentation par instances où le nombre de clusters par segmentation est fixé à 2.

$$
\textit{avec}\begin{cases}
P^0 & = \{P_0^0, P_1^0, \ldots, P_{r-1}^0\} \quad \textit{et } F_0 = \bigcup_{q=0}^{r-1} P_q^0 \\
P^1 & = \{P_0^1, P_1^1, \ldots, P_{r-1}^1\} \quad \textit{et } F_1 = \bigcup_{q=0}^{r-1} P_q^1 \\
P^2 & = \{P_0^2, P_1^2, \ldots, P_{r-1}^2\} \quad \textit{et } F_2 = \bigcup_{q=0}^{r-1} P_q^2 \\
\ldots & \\
P^{N-1} & = \{P_0^{N-1}, P_1^{N-1}, \ldots, P_{r-1}^{N-1}\} \quad \textit{et } F_{N-1} = \bigcup_{q=0}^{r-1} P_q^{N-1} \\
F_k & \textit{ est l'image } k \textit{ de la séquence, } 0 \leq k < N
\end{cases}
$$

Une autre contrainte, qui n'est pas obligatoire et qui est levée pour les séquences de maillages non contraintes, est d'avoir une connectivité constante entre les régions : si la région B est adjacente à la région A à un instant de la séquence, elle doit être adjacente à A à tous les autres instants de la séquence. La figure 3.8 illustre une segmentation par instances sur un cas simple.

La contrainte de cohérence temporelle entre deux segmentations successives n'est pas imposée.

Une segmentation par instances est composée d'autant de segmentations que d'images dans la séquence, tandis qu'une segmentation globale est une seule segmentation pour l'ensemble de la séquence.

3.2.2.3 Segmentation temporellement variable

Le dernier type de segmentation sur les sommets/faces proposé est la segmentation temporellement variable. Comme la segmentation par instances, elle consiste en une succession de segmentations de maillages, une par image, mais elle peut cependant avoir un nombre différent de clusters à chaque image.

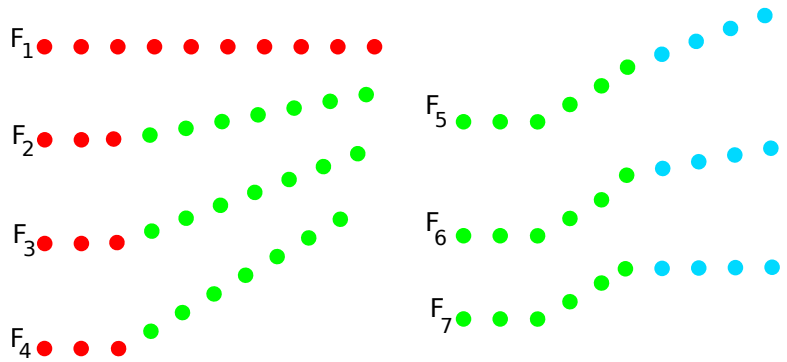


FIGURE 3.9 – Illustration d'une segmentation temporelle variable.

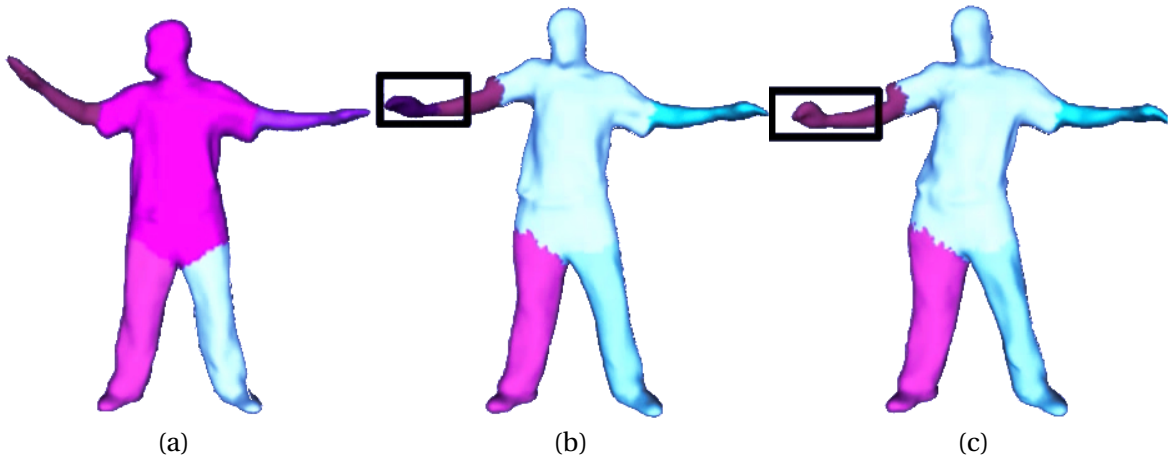


FIGURE 3.10 – Exemple de segmentation temporellement variable. (a) Segmentation sur la première image. (b) Le mouvement de la main gauche est différent du reste du bras, un cluster est donc créé. (c) La main gauche suit de nouveau le mouvement du bras : le cluster est fusionné au reste du bras.

L'objectif de ce type de segmentation est de représenter le mouvement en cours : la segmentation à l'image k permet de décrire les parties en cours de mouvement. Elle dépend donc d'un paramètre supplémentaire qui est la fenêtre temporelle sur laquelle on observe une action. La figure 3.9 illustre une telle segmentation tandis que la figure 3.10 montre un exemple de segmentation temporellement variable sur une séquence réelle.

Définition 3.5. *Une segmentation temporellement variable P d'une séquence F consiste en une segmentation de chaque image de la séquence sur une fenêtre temporelle de taille T, où la segmentation de chaque image reflète le mouvement en cours sur la fenêtre temporelle.*

3.2.2.4 Segmentation temporelle

Une segmentation temporelle consiste à segmenter la séquence en sous-séquences, en utilisant diverses caractéristiques (changement de topologie ou

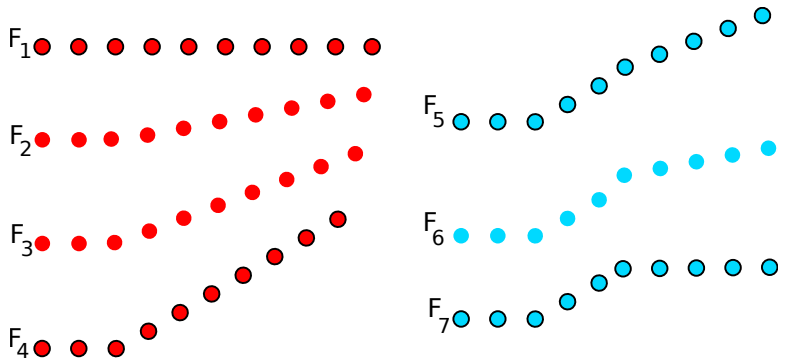


FIGURE 3.11 – Illustration d'une segmentation temporelle. La segmentation se fait sur les images : les images 1 à 4 sont regroupées d'un côté tandis que les images 5 à 7 sont regroupées d'un autre côté. Les images marquant le début et la fin d'un cluster sont celles avec un bordereau noir.

déformations importantes par exemple).

Définition 3.6. *Une segmentation temporelle P d'une séquence F est :*

$$P = \bigcup P_q$$

$$\textit{avec} \begin{cases} P_q = (F_k, ..., F_l), & 0 \leqslant k < l < \\ \bigcup P_q = F_{0...(N-1)} & \\ P_q \bigcap P_p = \varnothing, & \forall q \neq p \\ N & \textit{le nombre d'images de la séquence.} \end{cases}$$

En plus de ces critères simples, une segmentation temporelle peut également être sémantique. Dans ce cas-là, le but est de segmenter la séquence en sous-séquences telle que chaque sous-séquence ait un sens, correspondant à une action (i.e. dans une séquence représentant un homme qui court, puis saute, on aura deux sous-séquences, l'une avec l'homme qui court et l'autre où il saute). La figure 3.11 illustre le résultat d'une segmentation temporelle sur un cas simple : dans le premier cluster sont regroupées les images, F_1 à F_4, d'une sous-séquence représentant un mouvement vers le haut, tandis que le deuxième cluster contient les images, F_5 à F_7, décrivant un nouveau mouvement où certains sommets décrivent un mouvement vers le bas.

3.2.2.5 Combinaison de segmentations

Il est bien sûr possible de combiner ces différentes segmentations : une segmentation globale peut être appliquée à chaque sous-séquence d'une segmentation temporelle, différentes segmentations par instances à des instants différents peuvent être fusionnées en une seule pour obtenir une segmentation globale.

3.2.2.6 Critère de segmentation

Comme pour les segmentations de maillages statiques, il existe différents critères pour calculer une segmentation. Le plus utilisé est le critère de rigi-

dité [25, 27, 46, 66] : un cluster représente un groupe de points dont le déplacement des sommets peut être décrit par une transformation rigide, c'est à dire une combinaison d'une translation et d'une rotation :

$$P_t = R * P + T \tag{3.1}$$

avec :

- R : la matrice de rotation de taille 3×3 ;
- T : le vecteur de translation de taille 3 ;
- P : le point d'origine ;
- P_T : le point résultant de l'application de la translation et de la rotation sur le point P.

De tels clusters sont appelés *clusters rigides*. Il existe d'autres formulations de ce critère de rigidité qui sont moins fortes :

- La conservation des distances entre les arêtes. Cette formulation est problématique lors de l'utilisation avec des séquences de maillages autres que des maillages dynamiques, car les connectivités des maillages dans la séquence sont différentes ;
- La convexité d'une région, qui identifie un cluster rigide.

Dans la suite de cette thèse, nous introduisons un nouveau critère de segmentation qui est celui d'étirement uniforme. Les sommets dans un cluster représentent alors une zone qui est uniformément étirée, ces clusters sont appelés *clusters étirés*. Nous introduisons également les *clusters statiques* : les sommets de ces clusters ne sont pas déplacés.

3.3 Etat de l'art des méthodes de segmentation

Dans la partie précédente, nous avons présenté une classification pour les segmentations de séquences de maillages. Dans cette section, nous allons étudier les méthodes de segmentation proposées antérieurement, ainsi que leur positionnement par rapport à cette classification. La segmentation est un prétraitement utilisé dans plusieurs applications, comme mentionné auparavant : la compression, le transfert de pose... Ces applications utilisent principalement des segmentations de la géométrie, fondée sur le mouvement. Les segmentations globales sont donc les méthodes les plus développées. Historiquement, la recherche s'est focalisée sur les maillages dynamiques, cas le plus simple car la cohérence temporelle est implicitement définie. Cependant des méthodes pour des séquences stables et non contraintes ont été créées par la suite, en explicitant la cohérence temporelle. Ces méthodes sont donc aussi des méthodes de segmentation globale.

Comme nous le verrons par la suite, quelques méthodes de segmentation temporelle ont également été proposées, mais il n'existe pas à notre connaissance de segmentation par instance et temporellement variable.

FIGURE 3.12 – Segmentations obtenues par [37]. Les faces noires n'appartiennent à aucun cluster, de plus les clusters ne sont pas connexes.

FIGURE 3.13 – Segmentations obtenues par [29].

3.3.1 Segmentation de maillages dynamiques

Lengyel [47] utilise la segmentation comme pré-traitement pour la compression. Il propose une méthode de segmentation simple, mais qui souffre d'un problème important dans la mesure où un sommet peut appartenir à plusieurs clusters. James et Twigg [37] utilisent une méthode fondée sur le partitionnement par déplacement de la moyenne pour identifier les composantes rigides d'un maillage dynamique dans le but d'extraire un squelette d'animation. Comme Lengyel, il ne cherche pas à obtenir une segmentation stricte et les faces dans les régions non rigides ne sont dans aucun cluster, de plus certains clusters ne sont pas connectés, voir la figure 3.12. Collins et al. [23] améliorent la méthode de segmentation proposée par Lengyel, en formalisant l'approche.

Dans [57] and [45], les deux articles proposent une méthode de segmentation, mais dans ces deux articles, les frontières des clusters peuvent fortement dévier et donc les clusters peuvent ne plus représenter des composantes rigides.

Dans [46], Lee et al. propose une méthode produisant des résultats visuellement probants, et proposent le transfert de poses comme application de leur méthode. Cependant leur méthode souffre de plusieurs défauts : elle nécessite une empreinte mémoire importante, et en comparaison avec les autres méthodes de segmentation de maillages dynamiques, elle nécessite un temps de calcul important, voir la figure 3.7 pour visualiser les résultats. Dans [29], Guenther et al. proposent également une méthode qui requiert également un temps de calcul important, une solution proposée consiste à ne pas évaluer la segmentation sur toutes les images, voir la figure 3.13, mais ne propose pas de solution sur le choix des images où la segmentation doit être effectuée.

Mamou [49] et al. proposent une méthode différente, utilisant l'algorithme de k-means pour calculer la segmentation. Kalafaltar et Yemez [38] utilisent une méthode spectrale, reposant sur l'invariance de distance entre les sommets lors d'une trasnformation rigide, combinée avec les distances géodésiques et angulaires. Amjoun et Straßer [6] proposent une méthode où la segmentation se fait

en fonction de similarités locales. Toutes ces méthodes ont le problème de nécessiter en entrée le nombre de clusters.

Boulfani et al. [18] proposent une méthode utilisée pour la compression, où la gestion du nombre de clusters est problématique. Dans le cas de la compression, ils corrigent ce problème en segmentant uniquement les deux premières images de la séquence.

Dans [25], de Aguiar et al. proposent une méthode performante pour segmenter un maillage dynamique en composante rigide dans le but d'extraire un squelette d'animation. Ils utilisent l'invariance de distance entre les sommets lors d'une transformation rigide, et se fonde sur une méthode spectrale pour être plus robuste aux valeurs aberrantes. De plus, ils sont capables de déterminer le nombre de clusters automatiquement.

Récemment, Rosman et al. [56] ont proposé une nouvelle méthode. Le principal attrait de cet article provient de la formalisation de la segmentation en composantes rigides dans le cas des maillages dynamiques.

Toutes ces méthodes de segmentation globale sont optimisées pour les maillages dynamiques, et par conséquent ne sont pas capables de gérer les autres types de séquences.

La figure 3.14 montre les résultats de ces méthodes de segmentation sur des séquences classiques.

3.3.2 Segmentation de séquences stables et non contraintes de maillages

Dans cette section, nous présentons l'état de l'art des méthodes de segmentation globale de séquences stables/non contraintes de maillages. Nous faisons dans un premier temps un rappel sur les difficultés liées à ces séquences, puis étudions les méthodes proposées à proprement parler.

3.3.2.1 Difficultés

Les séquences stables/non contraintes de maillages sont des séquences où chaque maillage de la séquence possède sa propre connectivité (voir le chapitre 2). Cette connectivité propre à chaque maillage fait qu'il n'y pas de correspondance directe entre les sommets de deux maillages successifs, contrairement aux maillages dynamiques. Pour résoudre ce problème, il y a plusieurs possibilités comme nous le verrons dans la section suivante :

- calculer explicitement cette correspondance : méthode utilisée dans cette thèse, cela permet de calculer la trajectoire des sommets mais pose des problèmes pour le transfert des segmentations comme nous le verrons par la suite ;
- utiliser un squelette pour guider la segmentation ;
- calculer une segmentation sur chaque image et recaler ces segmentations.

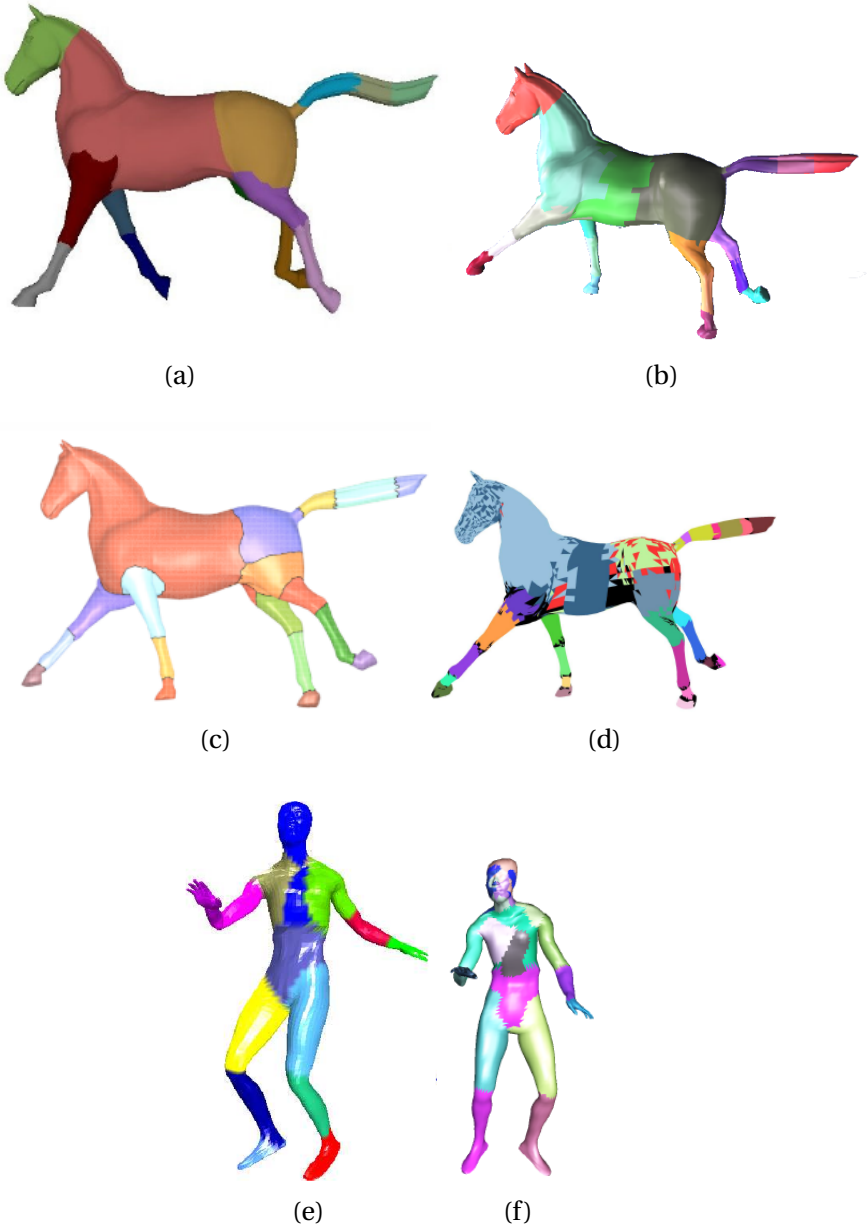


FIGURE 3.14 – Résultats des segmentations globales dans l'état de l'art. (a) [46]. (b) [25]. (c) [56]. (d) [37]. (e) [6]. (f) [49].

3.3.2.2 Etat de l'art des méthodes de segmentation

Sundaresan et Chellappa [64] proposent une méthode pour segmenter des séquences stables de maillages. Cependant leur méthode est destinée à la segmentation de séquence représentant des humains, ce qui pose des contraintes fortes. De plus, pour effectuer la segmentation, ils utilisent une voxelisation des maillages, ce qui soit augmente les temps de calcul si la voxelisation est fine soit pose des problèmes de précision si la voxeliation est grossière. Lee et al. [44] ont également proposé une méthode de segmentation adaptée à la segmentation d'humains, mais tout de même plus généraliste. Elle nécessite cependant l'utilisation d'un squelette en entrée, ce qui reste une contrainte forte. La segmentation consiste à affecter chaque sommet à l'os le plus proche sur chaque image. Cela leur permet de contourner le problème de transfert de segmentation. Cuzzolin et al. [24] proposent une méthode fondée sur les nuages de points et permettent de segmenter les extrémités des maillages dans les séquences. Cependant cette méthode ne permet pas de trouver les composantes rigides.

Toutes les méthodes présentées jusqu'ici ont une limitation commune, elles ne permettent pas de segmenter des séquences contenant plusieurs objets, tels que deux personnes en interaction. A notre connaissance, seules deux méthodes en sont capables, de plus ces deux méthodes ne nécessitent pas de modèle ou de squelette en entrée. La première, proposée par Varanasi et Boyer [66], se fonde sur la constatation suivante : les composantes rigides correspondent généralement à des parties convexes. Ils proposent donc une segmentation en parties convexes de plusieurs images de la séquence. Ensuite un recalage sur les régions de chaque image avec propagation est effectué pour obtenir la segmentation de la séquence. Cette méthode fournie de bons résultats, mais est fondée sur une hypothèse de convexité qui n'est pas toujours vérifiée. Pour corriger cela, lors de l'étape de recalage, les transformations rigides sur les clusters sont calculées. La seconde méthode est proposée par Franco et Boyer [27]. Elle résout simultanément les étapes de segmentation et de mise en correspondance. Le principal défaut de cette méthode est de nécessiter en entrée le nombre de clusters.

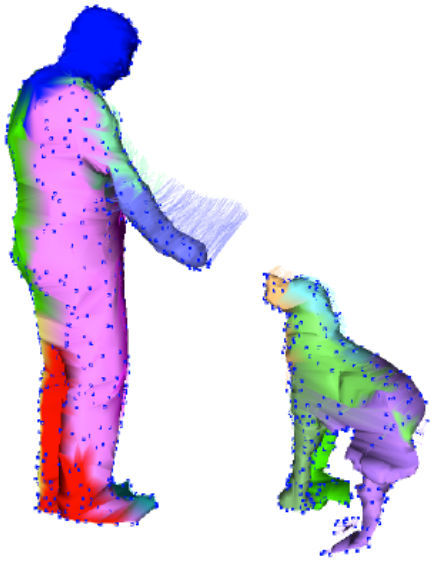
Ces deux dernières méthodes nécessitent également d'avoir un bon échantillonnage temporel sur les séquences, car elles nécessitent de faibles déplacements entre deux images successives.

La figure 3.15 montre les résultats obtenues par plusieurs de ces méthodes.

3.3.2.3 Segmentation temporelle

Concernant la segmentation temporelle de séquences de maillages, il n'existe, à notre connaissance, que les travaux de Yamasaki et al [68, 69, 70, 71, 72]. Dans [68], un histogramme des distances des sommets du maillage par rapport à trois points fixes est généré pour chaque maillage, et la segmentation se fait en fonction de la distance entre les histogrammes de deux images successives en fonctions de plusieurs seuils. Dans [70], une caractéristique appelée *Modified Shaped Distribution* est utilisée, à partir de laquelle les auteurs génèrent un histogramme représentant les degrés de libertés. Les coupures correspondent ainsi

(a) (b)



(c)

FIGURE 3.15 – Résultats des segmentations globales de l'état de l'art sur des séquences de maillages. (a) [66]. (b) [44]. (c) [27].

aux minima locaux dans cet histogramme. Cette méthode génère cependant une sur-segmentation. Dans [71], les auteurs utilisent un recalage par ICP [16] sur des représentations simplifiées des maillages entre chaque paire d'images, et obtiennent une erreur de recalage E_k pour F_k vers F_{k+1} et une erreur E_{k+1} pour F_{k+1} vers F_k. Ils cherchent ensuite les minima locaux de la fonction $DM(k) = E_k + E_{k+1}$. Dans [69], Yamasaki et al. proposent une amélioration des méthodes précédentes. Dans [72], ils reprennent la méthode proposée dans [70], mais combinent la distance "Modified Shaped Distribution" et la distance *Geodesic Shape Distribution* [30].

Dans le chapitre suivant, nous présentons une première méthode de segmentation globale fondée sur la fusion de régions, prenant en entrée une séquence stable de maillages.

Chapitre 4

Segmentation globale fondée sur la fusion de régions

Sommaire

4.1 Nature des clusters **40**
4.2 Chaîne d'exécution **40**
4.3 Mise en correspondance **42**
4.3.1 Maillages dynamiques 42
4.3.2 Recalage fondé sur CPD 42
4.3.3 Projection 43
4.3.4 Mise en correspondance et vecteurs de déplacement . . 44
4.3.5 Mise en correspondance d'un maillage avec le reste de la séquence 45
4.4 Segmentation **46**
4.4.1 Vue d'ensemble 46
4.4.2 Erreurs 47
4.4.3 Segmentation initiale 47
4.4.4 Raffinement 51
4.4.5 Transfert de la segmentation sur la séquence 52
4.5 Discussion **53**
4.5.1 Mise en correspondance 53
4.5.2 Segmentation 55
4.5.3 Limitations 62

La première méthode que nous proposons est une méthode fondée sur la fusion de clusters. Cette méthode prend en entrée une séquence stable de maillages, i.e. avec nombre de sommets et connectivité variables, mais sans chan-

gement global de topologie. La segmentation résultante est une segmentation globale sur les sommets, i.e. une segmentation unique pour toute la séquence.

La chaîne d'exécution est présentée dans la section 4.2. Une mise en correspondance est appliquée sur chaque paire de maillages successifs de la séquence (Section 4.3). Cette mise en correspondance nous fournit :

- une correspondance entre les sommets de l'image F_k et ceux de l'image F_{k+1} ;
- une approximation des vecteurs de déplacement des sommets de F_k.

En utilisant ces vecteurs de déplacement, nous calculons une segmentation à la volée (Section 4.4).
Des résultats ainsi qu'une analyse de ces résultats sont présentés dans la section 4.5.

Cette méthode permet :

- de traiter de longues séquences avec des maillages volumineux ;
- de contrôler l'apparition des clusters et le type de transformation associée ;
- d'intégrer différents types de transformations pour la segmentation. Par exemple, la segmentation actuelle permet de différencier les zones étirées uniformément et les zones rigides.

En outre, elle ne nécessite pas de squelette.

4.1 Nature des clusters

Les clusters générés par la segmentation sont des ensembles de sommets connexes. Chaque cluster (voir la section 3.2.2.6) représente une région des maillages :

- soit subissant une transformation rigide ;
- soit étirée uniformément ;
- soit sans déplacement.

4.2 Chaîne d'exécution

Dans cette partie, nous décrivons la chaîne d'exécution complète du processus de segmentation (voir la Figure 4.1). Ce processus peut être approximativement séparé en deux étapes :

- une étape de mise en correspondance (section 4.3) ;
- une étape de création ou de raffinement de la segmentation (section 4.4).

L'algorithme prend en entrée une séquence stable de maillages et produit une segmentation globale. Le maillage de l'image F_0 de la séquence est mis en correspondance avec le deuxième F_1. En utilisant la mise en correspondance, les vecteurs de déplacement (des sommets du premier maillage) entre les deux images sont estimés.

Sur la base de ces vecteurs de déplacement, une segmentation initiale sur le premier maillage de la séquence est calculée. Cette segmentation contient trois

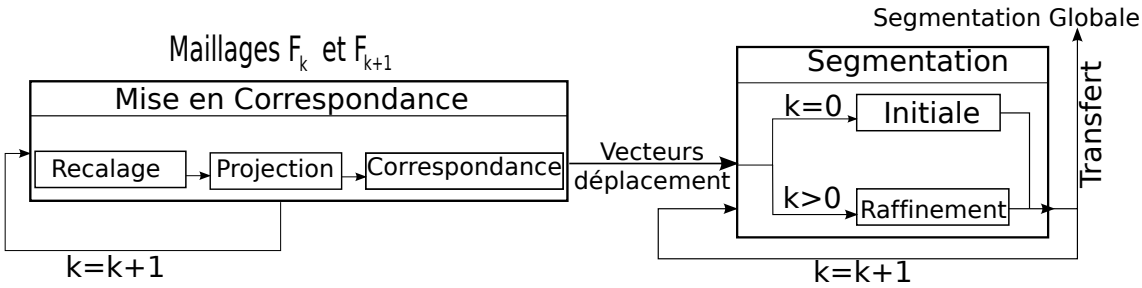


FIGURE 4.1 – Chaîne d'exécution du processus de segmentation.

types de clusters :

1. les zones sans déplacement : les clusters statiques.
2. les régions uniformément étirées : les clusters étirés.
3. les zones subissant une transformation rigide : les clusters rigides.

Ensuite pour chaque paire d'images successives (F_k,F_{k+1}) de la séquence, nous appliquons la même méthode de mise en correspondance et d'estimation de vecteurs de déplacement. La segmentation, initiale ou de l'image précédente, est raffinée en analysant le mouvement des sommets dans chaque cluster. Cela signifie qu'à chaque pas de temps, les clusters calculés précédemment sont subdivisés.

La segmentation calculée pour la dernière image F_{N-1} est donc la segmentation contenant l'ensemble des mouvements intervenant dans la séquence. Afin d'obtenir une segmentation globale visualisable sur l'ensemble de la séquence, cette dernière segmentation est transférée sur les images de toute la séquence.

L'algorithme 1 présente le corps de la méthode.

Algorithme 1: Corps de la méthode de segmentation.

Entrée : Séquence stable de maillages : SM.
Sortie : Segmentation Globale : S.
F_0=Maillage(0,SM)
F_1=Maillage(1,SM)
($VecDeplacement$,$Correspondance$)=MiseEnCorrespondance(F_0,F_1)
$Cluster$=Segmentation(M_0,$VecteurDeplacement$)
pour toutes les *paires de maillages* (F_k,F_{k+1}) *de* $S \setminus F_0$ **faire**
  $NouveauCluster$=$\varnothing$
  ($VecDeplacement$,$Correspondance$)=MiseEnCorrespondance(F_k,F_{k+1})
  pour chaque *cluster* $C \in Cluster$ **faire**
    $SubClusters$=Segmentation(C,$VecteurDeplacement$)
    $NouveauCluster$=$NouveauCluster + SubClusters$
  $Cluster$=$NouveauCluster$
// $Cluster_i$ est la segmentation sur la dernière image
S=TransfertSegmentation(SM,$Cluster$)

4.3 Mise en correspondance

La mise en correspondance se fait en trois étapes successives : le recalage, la projection et l'établissement des correspondances. Elle prend en entrée deux maillages pouvant avoir un nombre de sommets différents, voir l'algorithme 2. Ces différentes étapes sont décrites dans les parties suivantes. A la fin de la mise en correspondance, nous obtenons :

- les vecteurs de déplacement entre deux images successives F_k et F_{k+1} ;
- un ensemble de correspondances unidirectionnel entre les sommets des maillages F_k et F_{k+1}.

De plus, en traversant les ensembles de correspondances sur l'ensemble de la séquence, nous obtenons un ensemble de correspondances entre F_0 et F_k, $0 \leqslant k < N$.

Algorithme 2: Mise En Correspondance

Entrée : F_k,F_{k+1} : Maillages.
Sorties : $Corr$: Sommet[Sommet].
$VecDeplacement$: Sommet[Vecteur].
// Etape de Recalage utilisant Coherent Point Drift [51]
F'_k=CPD(F_k,F_{k+1})
// Projection
P=projection(F'_k,F_{k+1})
// Estimation vecteur et Correspondance
($Corr$,Vec)=correspondance(F_k,F_{k+1},F'_k,P)

4.3.1 Maillages dynamiques

Du point de vue de notre méthode, les maillages dynamiques sont des cas particuliers : il n'est pas nécessaire de calculer une mise en correspondance et cette étape est donc supprimée. En effet, la correspondance entre deux maillages est implicitement fournie, le sommet $v_i^{(k+1)}$ correspond au sommet $v_i^{(k)}$ et les vecteurs de déplacement sont calculés ainsi :

$$\overrightarrow{DV_i^{(k)}} = v_i^{(k+1)} - v_i^{(k)} \tag{4.1}$$

4.3.2 Recalage fondé sur CPD

L'étape de recalage (voir la figure 4.2) est fondée sur l'algorithme "Coherent Point Drift" (CPD) de Myronenko et al. [51]. Le recalage est effectué sur les paires de maillages successifs (F_k, F_{k+1}) de la séquence et génère un maillage F'_k. Le maillage F'_k est le maillage F_k déformé pour qu'il ait la même forme que F_{k+1}. Le nombre de sommets de F'_k est donc le même que celui de F_k .

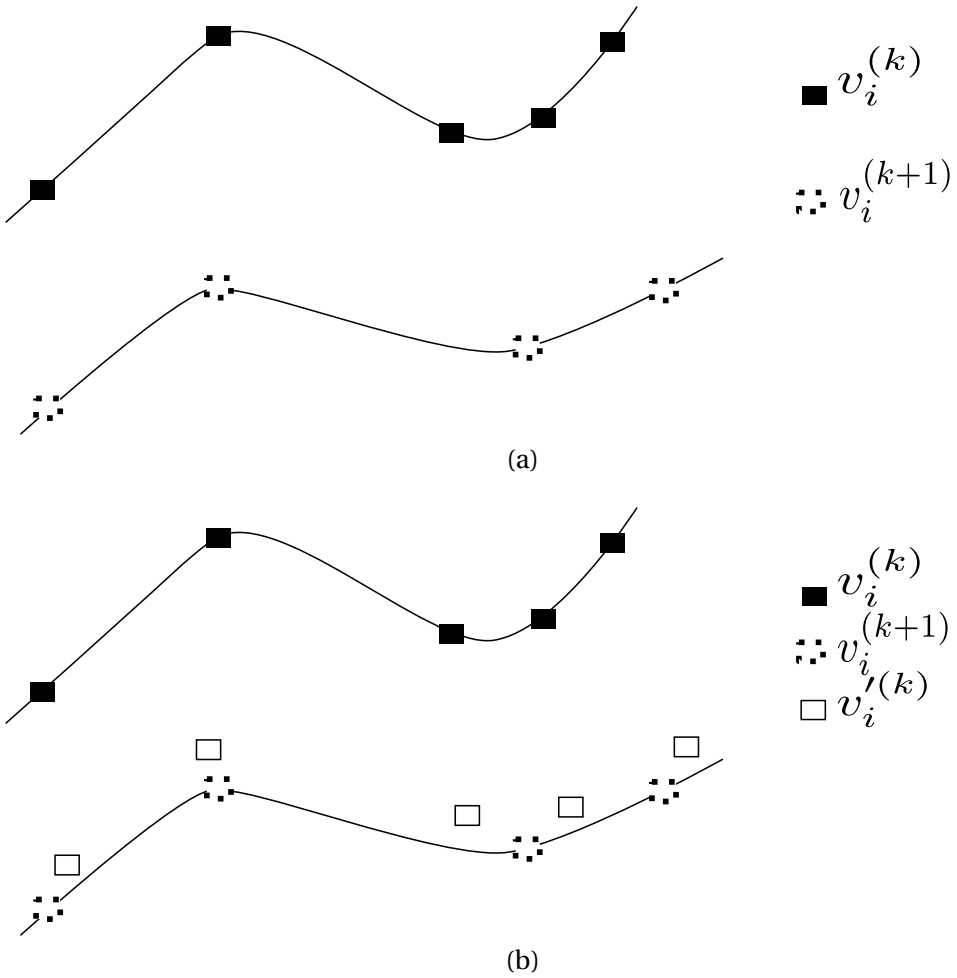


FIGURE 4.2 – (a) Deux maillages (F_k, F_{k+1}). (b) Deux maillages F_k,F_{k+1} et le maillage recalé F'_k.

4.3.3 Projection

Afin d'améliorer la mise en correspondance entre les images F_k et F_{k+1} ainsi que l'estimation des vecteurs de déplacements, les sommets du maillage F'_k sont projetés sur F_{k+1}. Cela est fait de la manière suivante :

1. La normale de chaque sommet de F_{k+1} est évaluée.
2. Pour chaque sommet $v_i'^{(k)}$, sommet d'indice i du maillage F'_k :
   (a) nous cherchons le sommet $v_j^{(k+1)}$ le plus proche sur le maillage F_{k+1}, noté v. La méthode utilisée consiste à itérer sur l'ensemble des sommets de F_{k+1} et calculer la distance avec v. Pour le calcul de la distance, nous utilisons la distance euclidienne. La méthode de calcul choisie, bien que n'étant pas la plus performante, est suffisante pour nos besoins.
   (b) $v_i'^{(k)}$ est projeté sur F_{k+1} en suivant la normale de v. Le point projeté est noté $vp_i^{(k)}$.

En utilisant cette méthode, chaque sommet $v_i^{(k)}$ a un sommet projeté correspondant $vp_i^{(k)}$, voir l'algorithme 3 et la figure 4.3. La prochaine et dernière étape consiste à établir la correspondance entre les sommets de F_k et F_{k+1}.

Algorithme 3: Projection

Entrée : Maillage : F'_k, F_{k+1}
Sortie : Projection : Sommet[Sommet]
pour chaque *sommet* $v \in M_2$ **faire**
  EvalueNormale(v)
pour chaque *sommet* $v \in M_1$ **faire**
  vp=sommetPlusPoche(v,M_2)
  Proj[v]=Projete(v,M_2,normal(vp))

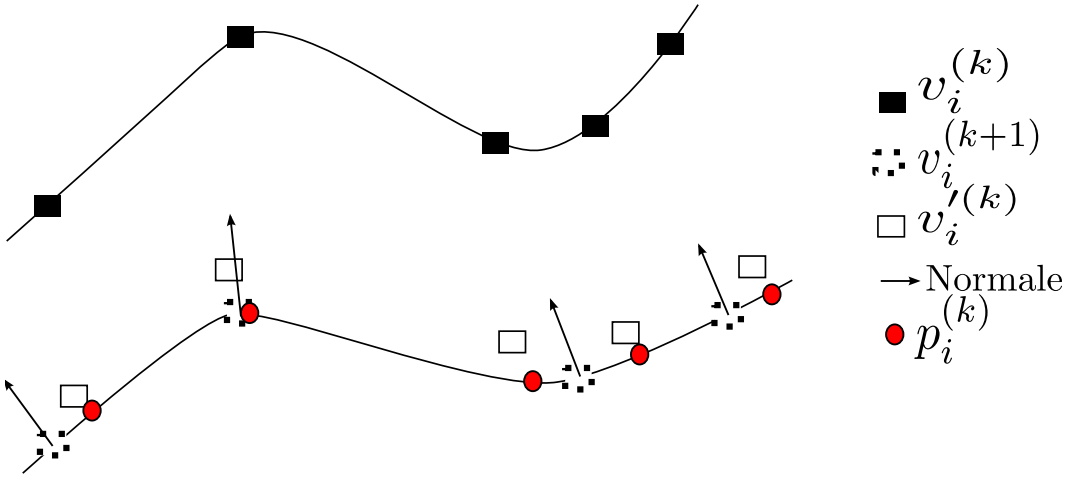


FIGURE 4.3 – Projection de maillage.

4.3.4 Mise en correspondance et vecteurs de déplacement

Nous présentons dans un premier temps comment nous établissons la mise en correspondance puis le calcul des vecteurs de déplacement, voir la figure 4.4 et l'algorithme 4.

Pour trouver le sommet correspondant v_j^{k+1} au sommet v_i^k, nous utilisons le sommet $v_i'^k$ calculé précédemment. $v_i'^k$ est le sommet correspondant à v_i^k dans le maillage $F_i'^k$, maillage recalé de F_k sur F_{k+1}. Le sommet le plus proche de $v_i'^{k+1}$ dans F_{k+1} est le sommet correspondant à v_i^k. De cette manière, nous obtenons une mise en correspondance unidirectionnelle. Il est important de noter que la mise en correspondance de chaque point est effectuée indépendamment des autres. La pertinence de ces choix est débattue dans la section 4.5.

Lors de cette mise en correspondance, deux scénarios sont possibles :

- $nv(F_k) > nv(F_{k+1})$: dans ce cas-là, plusieurs sommets de F_k sont mis en correspondance avec le même sommet de F_{k+1} ;
- $nv(F_k) < nv(F_{k+1})$: dans ce cas-là, des sommets de F_{k+1} n'ont pas de sommets correspondants.

Les vecteurs de déplacement, noté $\overrightarrow{DV_i^{(k)}}$ pour le vecteur de déplacement associé au sommet v_i à l'image k, sont calculés à partir du sommet v_i et vp_i : $\overrightarrow{DV_i^{(k)}} = vp_i - v_i$, où vp_i est le sommet v_i projeté sur F_{k+1}.

Algorithme 4: Correspondance.

Entrées : F_k, F_{k+1}, F'_k : Maillages
P : Sommet[Sommet]
Sortie : MC : Sommet[Sommet]
DV : Sommet[Vecteur]
pour i *allant de* 0 *à* $nv(F_k)$ **faire**
  $MC[v_i^k]$=sommetPlusProche($P[v_i^k]$,F_{k+1})
  $DV[v_i^k]$=$P[v_i^k]$-v_i^k

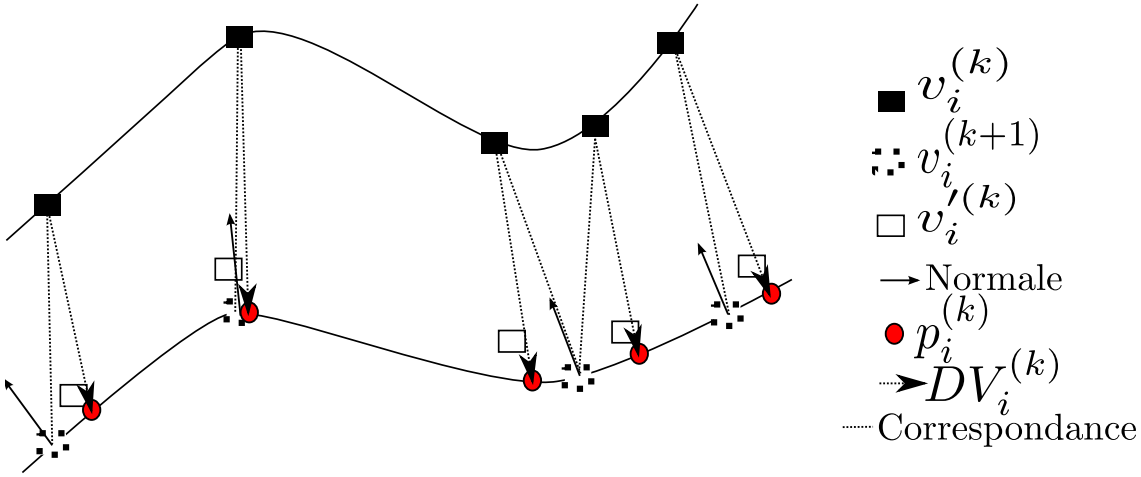


FIGURE 4.4 – Correspondance et vecteur de déplacement.

4.3.5 Mise en correspondance d'un maillage avec le reste de la séquence

La phase de mise en correspondance peut également être utilisée pour calculer rapidement une approximation de la mise en correspondance du maillage F_0 avec l'ensemble des maillages de la séquence, i.e la mise en correspondance de F_0 avec $F_k, 0 \leqslant k < N$. Cette mise en correspondance sur l'ensemble de la séquence est appelée carte de correspondances. Cette étape peut être calculée récursivement ainsi, avec k et l désignant les indices des images et i l'indice du sommet v_i^k à l'image k :

$$v_i^{k\to l} = v_i^{l-1\to l}(v_i^{l-1\to k}) \text{ où } k \geq l \tag{4.2}$$

Cette mise en correspondance de la première image avec l'ensemble de la séquence est utilisée dans l'étape de transfert de segmentation sur l'ensemble de la séquence (section 4.4.5).

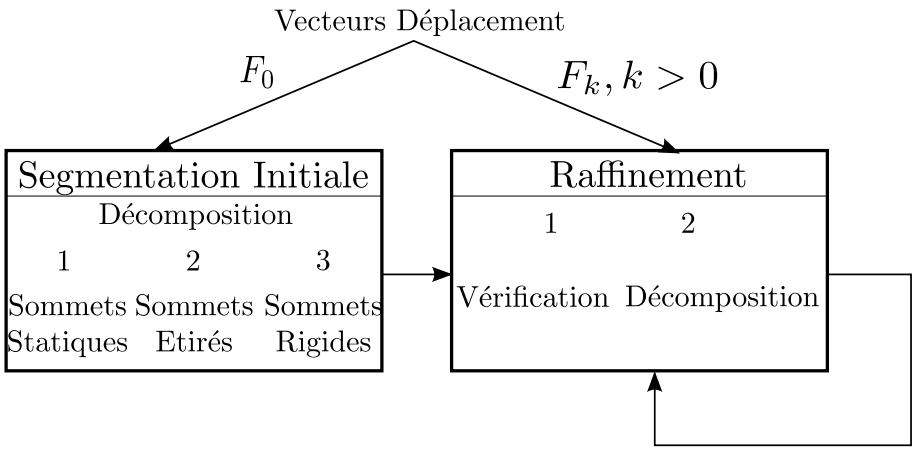


FIGURE 4.5 – Chaîne d'exécution du processus de segmentation.

4.4 Segmentation

Dans cette section, nous présentons l'étape de segmentation fondée sur la fusion de clusters. Cette étape nécessite une mise en correspondance de bonne qualité (i.e. vecteurs de déplacement et mise en correspondance de sommets). Cependant, n'importe quelle méthode de mise en correspondance peut être utilisée tant que ces informations sont fournies. Nous avons par exemple testé notre segmentation en utilisant [41], où la mise en correspondance est trouvée en utilisant le plongement laplacien des maillages (ce qui les "déplie").

Les résultats de la segmentation sont présentés et examinés dans la section 4.5.

4.4.1 Vue d'ensemble

Notre méthode de segmentation produit une segmentation globale, où les clusters représentent non seulement des zones suivant une transformation rigide, mais également contrairement aux méthodes précédentes, des zones uniformément étirées.

Nous utilisons une méthode par raffinement :

- à l'aide des deux premiers maillages F_0 et F_1 de la séquence et des vecteurs de déplacement associés, nous calculons la segmentation initiale (section 4.4.3) ;
- pour les autres images de la séquence, nous raffinons la segmentation obtenue sur l'image précédente en analysant le mouvement des clusters en cours et en les divisant si nécessaire grâce aux vecteurs de déplacement et à la mise en correspondance (section 4.4.4).

Cette approche par raffinement nous permet :

- de traiter de longues séquences contenant des maillages volumineux, car l'algorithme ne nécessite que deux maillages en mémoire à la fois ;
- de suivre les transformations. Nous savons à quelle image un cluster est créé ainsi que le type de transformation associée.

La segmentation de la séquence est la segmentation obtenue sur la dernière image. Cette segmentation est donc ensuite transférée sur l'ensemble de la séquence en utilisant la carte de correspondance (section 4.4.5).

4.4.2 Erreurs

Durant le processus de segmentation, nous utilisons une métrique notée E_T pour trier les clusters par erreur. Nous utilisons la méthode proposée par Horn [34] pour calculer la meilleure transformation rigide possible entre deux nuages de points P_i et P_j avec $card(P_i) = card(P_j)$ et où les points de P_i ont une correspondance directe avec ceux de P_j. La méthode de Horn renvoie un vecteur de translation C_T et une matrice de rotation C_r tels que $P_j = C_R * P_i + C_T$.

Pour deux nuages de points (P_i, P_j), E_T peut être définie comme :

– l'erreur moyenne :

$$AE(P_i, P_j) = \frac{1}{card(P_i)} \sum_{p_i \in P_i} \|C_R p_i + C_T - p_j\|. \tag{4.3}$$

– l'erreur maximum :

$$ME(P_i, P_j) = \max_{p_i \in P_i} \|C_R p_i + C_T - p_j\| \tag{4.4}$$

– la difference d'erreur :

$$DE(P_i, P_j) = AE(P_i, P_j) - ME(P_i, P_j) \tag{4.5}$$

4.4.3 Segmentation initiale

La segmentation initiale est calculée en utilisant les deux premiers maillages de la séquence, F_0 et F_1, en utilisant les vecteurs de déplacement obtenus lors de la mise en correspondance entre F_0 et F_1. Nous regroupons les clusters suivant trois catégories, et la segmentation suivant la catégorie est effectuée successivement :

1. *sommets statiques* : ces sommets ne se déplacent pas (ils sont notés $\mathcal{C}_{static}$).
2. *sommets étirés* : ces sommets sont dans une zone étirée uniformément (notés $\mathcal{C}_{stretched}$).
3. *sommets rigides* : le déplacement de ces sommets peut être décrit par une transformation rigide. (noté $\mathcal{C}_{rigid}$)

Les clusters sont calculés de la manière suivante : les sommets statiques sont définis par un vecteur de déplacement dont la norme est nulle ou inférieure au seuil $T_{statique}$, les sommets dont la norme est supérieure à ce seuil et dont les vecteurs de déplacement sont colinéaires et de même direction sont dans des clusters étirés. Ils sont regroupés dans des ensembles connexes. Les sommets rigides sont plus compliqués à calculer. Nous utilisons une version modifiée de la méthode décrite dans [35].

4.4.3.1 Non-Rigid Registration Under Isometric Deformation [35]

Dans cette article, les auteurs introduisent la notion de *c*luster n-étendu.

Définition 4.1. *Le cluster n-étendu $\widetilde{C}$ d'un cluster C est l'union du cluster C et des n-voisins des sommets du cluster C.*

Huang et al. [35] ont proposé une méthode de recalage de deux maillages subissant des transformations isométriques. Leur approche utilise une méthode par optimisation alternant la mise en correspondance et la déformation du premier maillage vers le second, et des versions simplifiées des deux maillages telles que le nombre de sommets des versions simplifiés est le même. Pour la phase de déformation, une segmentation en composante rigides entre deux maillages est calculée, segmentation sur laquelle nous nous basons par la suite (voir la section 4. Dans cette segmentation, chaque sommet est initialement un cluster. Pour chaque cluster, la méthode calcule la transformation qui transforme le sommet et son 1-voisinage dans le premier maillage vers le second maillage, ainsi que l'erreur associée pour chaque transformation. Elle trie ensuite les clusters en fonction de l'erreur de la transformation. En notant R la rotation et T la translation du cluster C minimisant l'erreur, elle applique la transformation aux clusters voisins, et calcule l'erreur $E_{kl}(C')$ pour chacun de ses clusters C'. Si cette erreur est inférieure à un seuil E_{max}, alors le cluster voisin est fusionné au cluster C. La transformation et l'erreur résultante pour le nouveau cluster sont calculées, les clusters de nouveau triés par erreur et le processus recommence.

$$E_{kl}(C') = max_{s_i \in \tilde{C'}} ||R * s_i + T - s'_i|| \tag{4.6}$$

avec :
- $\tilde{C}'$: le cluster étendu C' ;
- s'_i : le sommet correspondant à s_i.

Le problème de cette méthode est que la segmentation s'arrête trop tôt : des clusters peuvent encore être regroupés. Dans le cas étudié par les auteurs, cette étape est alternée avec la déformation du maillage, ce problème est donc mineur, car les maillages sont de plus en plus proches. La figure 4.6 montre un résultat de leur méthode de segmentation.

4.4.3.2 Version modifiée

Afin de corriger le problème lié à la terminaison précoce, nous le modifions de la manière suivante. L'algorithme que nous appliquons (voir l'algorithme 5 et la figure 4.7) est le suivant :

1. Initialisation : chaque sommet est un cluster.
2. Pour chaque cluster C, calculer la transformation rigide optimale [34] du cluster n-étendu $\widetilde{C}$ entre la première image et la deuxième image. Les sommets du cluster $\widetilde{C}$ dans la seconde image sont reconstruits à partir des points de $\widetilde{C}$ de la première image et des vecteurs de déplacement associés calculés dans l'étape de mise en correspondance.

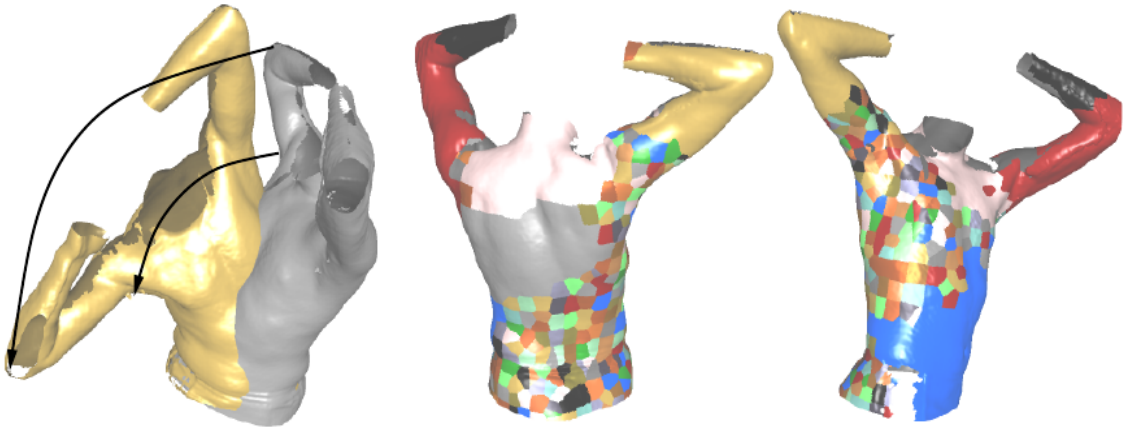


FIGURE 4.6 – Résultat d'une segmentation entre deux maillages avec [35], où de nombreux clusters ne sont pas fusionnés.

3. Prendre le cluster $\widetilde{C}_m$ ayant la plus petite erreur de transformation E_t (voir la section 4.4.2).

4. Pour chaque cluster C_n intersectant le cluster $\widetilde{C}_m$, appliquer la transformation trouvée pour $\widetilde{C}_m$ à chaque point de C_n. Si l'erreur E_f est inférieure au seuil T_{transf}, fusionner les clusters C_m et C_n. Si aucun des clusters n'est fusionné, prendre le cluster minimisant l'erreur E_t hors $\widetilde{C}_m$ et recommencer cette étape. L'erreur E_f est définie comme la distance maximale entre les sommets transformés de C_n et les sommets reconstruits.

5. Répéter les étapes 2 à 4 tant qu'une fusion est possible.

Dans nos expériences, les clusters 3-étendus sont utilisés et suffisants.

Comme mentionné précédemment, l'ordre dans lequel la segmentation est appliquée est important. L'algorithme calcule donc dans un premier temps $\mathcal{C}_{static}$ et $\mathcal{C}_{stretched}$. Ensuite le sous-maillage $SF = F_0 \setminus (\mathcal{C}_{static} \cup \mathcal{C}_{stretched})$ est construit. Les clusters rigides sont calculés en dernier sur SF, afin d'éviter que les clusters rigides ne chevauchent les clusters de types différents, voir la figure 4.8.

Afin d'éviter une sur-segmentation, les clusters dont la taille est inférieure à 3% du nombre de sommets du maillage sont fusionnés. Ces clusters sont fusionnés avec leur cluster voisin minimisant l'erreur E_f comme dans l'étape 4 de la segmentation rigide. Ces petits clusters fusionnés correspondent généralement :

- soit à du bruit dû au système de reconstruction ;
- soit à zones non rigides autour des articulations du maillage.

A l'issue du processus de la segmentation initiale, le maillage de la première image F_0 est décomposé en cluster de différent type, en fonction des vecteurs de déplacement entre les deux premières images. Cette segmentation est ensuite raffinée en utilisant les images suivantes de la séquence après chaque étape de mise en correspondance.

Algorithme 5: Segmentation Rigide.

Entrées : FS : Maillage
DV : Vecteurs déplacement.
Sortie : C : segmentation rigide.
C= F_r
$fusion$=vrai
tant que *fusion* **faire**
  pour chaque *cluster* $c \in C$ **faire**
    $List$= $List$ + TransformationEtErreurF_F+1(c+voisins(c))
  i=0
  $List$=trieParErreur($List$)
  tant que $i < Length(List)$ **faire**
    c=N^{eme}Element(List,i)
    $suivantParErreur$=vrai
    pour chaque $c_c \in$ *voisin*(c) **faire**
      si $(C_c^{k+1} - Transformation(c) \times c_c) < T_{transf}$ **alors**
        Fusion(c_c,c)
        suivantParErreur=faux
    si $suivantParErreur$ **alors**
      i=i+1
    sinon
      arret
Fusion des petits clusters.

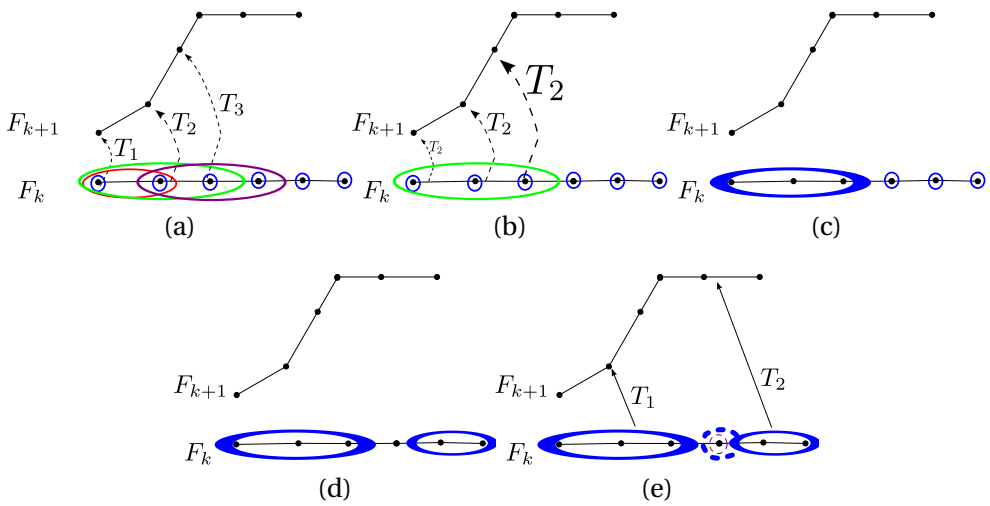


FIGURE 4.7 – Segmentation Rigide : processus. (a) $\widetilde{C}_i$ et transformation. (b) Sélection du cluster et test de la transformation. (c) Fusion. (d) Fusion suivant le même processus. (e) le sommet restant est fusionné avec le cluster voisin ayant l'erreur la plus petite.

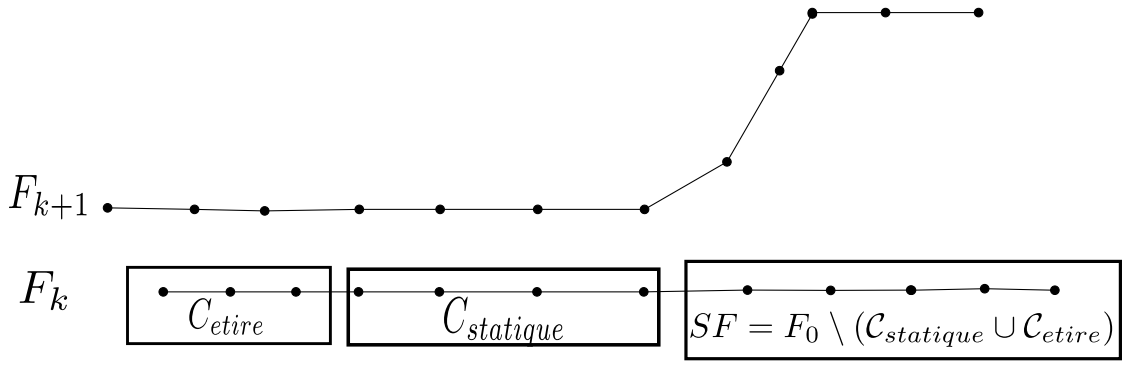


FIGURE 4.8 – Clusters statiques, étirés, rigides et sous-maillage.

4.4.4 Raffinement

Le but de l'étape de raffinement est de trouver les nouveaux clusters apparaissant sur l'image suivante. Chaque cluster de l'image précédente est subdivisé, si nécessaire, en *sous-clusters*. Le processus de raffinement se fonde sur la segmentation calculée lors de l'étape précédente, i.e. la segmentation initiale pour la deuxième image, ou le raffinement précédent dans les autres cas. Les vecteurs de déplacement calculés lors de la mise en correspondance sont également utilisés pour déformer le maillage reconstruit au raffinement précédent. Nous utilisons donc le maillage de la première image comme modèle pour la segmentation, puisqu'il est déformé au fur et à mesure des raffinements pour suivre les autres maillages de la séquence.

Le raffinement se fait en deux étapes :
- vérification des clusters précédents ;
- recherche des nouveaux clusters.

Une étape de vérification est utilisée afin d'éviter de recréer les mêmes clusters, la vérification est plus rapide à calculer que la segmentation. Nous vérifions que :
- pour les clusters statiques : les vecteurs de déplacement sont toujours sous le seuil T_{static} ;
- pour les clusters étirés : les vecteurs de déplacement sont toujours colinéaires et de même direction ;
- pour les clusters rigides, la meilleure transformation entre les points du cluster C_m à l'image F_k et F_{k+1} est valide : si l'erreur la plus importante est au-dessus du seuil T_{transf}, la transformation est considérée comme invalide. La transformation est toujours calculée avec la méthode de Horn et les points pour l'image F_{k+1} sont reconstruits à partir des points de l'image F_k et les vecteurs de déplacement.

Tous les clusters qui sont valides sont conservés, les autres clusters correspondent à des régions où une nouvelle transformation commence. Ces clusters sont donc divisés en sous-clusters :
- si la nouvelle transformation est applicable sur l'ensemble des points du cluster, ce cluster est conservé sans être décomposé ;
- sinon le cluster est décomposé.

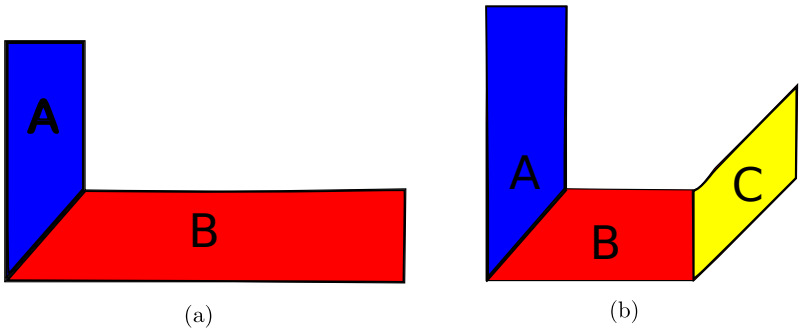


FIGURE 4.9 – Exemple de raffinement pour la méthode par fusion de clusters. (a) Segmentation à l'image F_k. (b) Segmentation à l'image F_{k+1}. Le cluster B est divisé en sous-clusters B et C à cause de la rotation.

Cette décomposition est similaire à celle présentée dans la section 4.4.3, sauf qu'elle se fait sur chaque cluster. Pour chaque cluster C_c parmi les clusters à décomposer, l'algorithme segmente d'abord les sommets statiques, puis étirés et enfin rigides. Comme avant, les sous-clusters statiques et étirés sont calculés et ajoutés respectivement aux ensembles $\mathcal{C}_{statique}$ et $\mathcal{C}_{etire}$ et $FS = F_k \setminus (\mathcal{C}_{statique} \cup \mathcal{C}_{etire})$ est construit. Contrairement à la segmentation initiale, la segmentation des clusters rigides ne se fait pas sur FS, mais sur $FS \cap C_c$, afin d'assurer que la transformation se fait sur C_c (voir la figure 4.9).

4.4.5 Transfert de la segmentation sur la séquence

Le transfert consiste de segmentation consiste à faire passer une segmentation calculer sur une image F_k sur l'image F_{k-1} ou F_{k+1}. Nous présentons différentes méthodes aux cours de cette thèse. Lorsque ce processus est appliqué sur toute les images de la séquence, de F_N à F_{N-1} puis de F_{N-1} à F_{N-2}..., nous appelons cela "transfert de la segmentation sur la séquence". Une fois que la segmentation a été calculée sur toutes les images de la séquence, la segmentation de l'ensemble de la séquence est sur la dernière image. Cette segmentation doit donc être transférée sur la séquence en utilisant les informations de mise en correspondance. Comme certains points peuvent ne pas avoir de correspondance, lors du report de la segmentation de F_k à F_{k+1}, des trous peuvent apparaître dans F_{k+1} : certains points ne sont dans aucun cluster. Nous appliquons donc une méthode de croissance de clusters. Pour chaque sommet P qui n'est pas dans dans un cluster, nous cherchons le cluster C qui contient le plus grand nombre de sommets du 1-voisinage de P : P est mis dans ce cluster C, , voir la figure 4.10.

Pour le transfert de la dernière segmentation, nous utilisons la carte de correspondance : elle nous permet d'obtenir la correspondance des sommets F_0 avec ceux de F_{N-1}, où F_{N-1} est la dernière image de la séquence. Nous renversons la correspondance, i.e si $v_i^{(0)}$ a pour sommet correspondant $v_j^{(N-1)}$, alors $v_j^{(N-1)}$ a pour sommet correspondant $v_i^{(0)}$. En utilisant ce renversement de correspondance, nous transférons la segmentation sur $F_{(N-1)}$ sur F_0 en utilisant

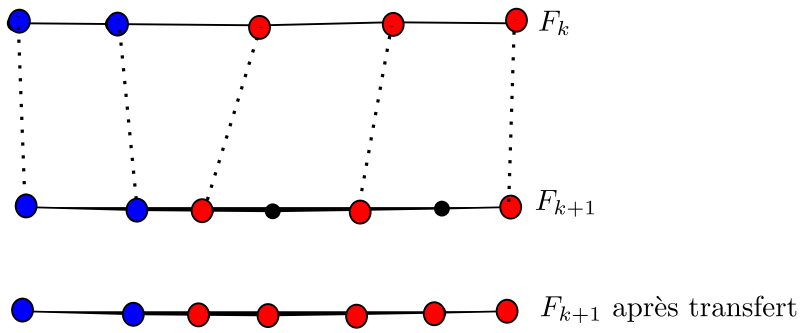


FIGURE 4.10 – Transfert de la séquence.

une croissance de clusters si nécessaire. Ensuite nous transférons la segmentation de F_k vers F_{k+1} en utilisant la mise en correspondance et une croissance de clusters pour chaque instant.

4.5 Discussion

Dans cette partie, nous présentons les résultats obtenus avec cette méthode et nous les analysons. Notre méthode requiert seulement deux images en mémoire à la fois, ce qui signifie que :

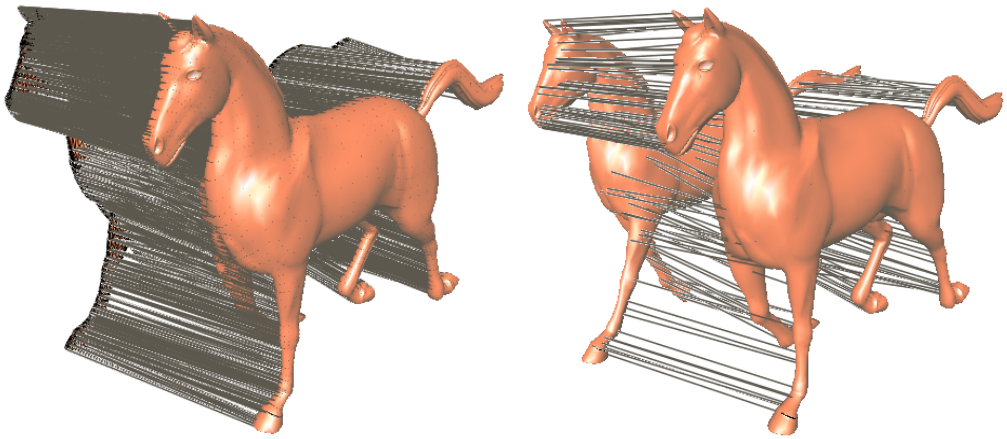
- nous pouvons traiter des séquences composées d'un grand nombre d'images et de maillages volumineux ;
- l'apparition ainsi que le type de transformation associé à chaque cluster sont connus.

4.5.1 Mise en correspondance

Nous présentons des résultats de mise en correspondance sur la figure 4.11, dans laquelle les correspondances entre deux images de la séquence "horse" sont visualisées par des segments. Les vecteurs de déplacement et les correspondances sont par la suite utilisés durant le processus de segmentation. Sur un ordinateur personnel standard (core2 duo, 2.33GHz), la mise en correspondance d'un maillage de 8199 sommets avec un autre de 7254 requiert 13 minutes. L'implantation est en Matlab en utilisant une implantation rapide de CPD fondé sur les transformations gaussiennes rapides. Dans le processus de mise en correspondance, le goulot d'étranglement se situe dans l'étape de recalage, les temps de calcul des étapes de projection et d'établissement des correspondance sont minoritaires.

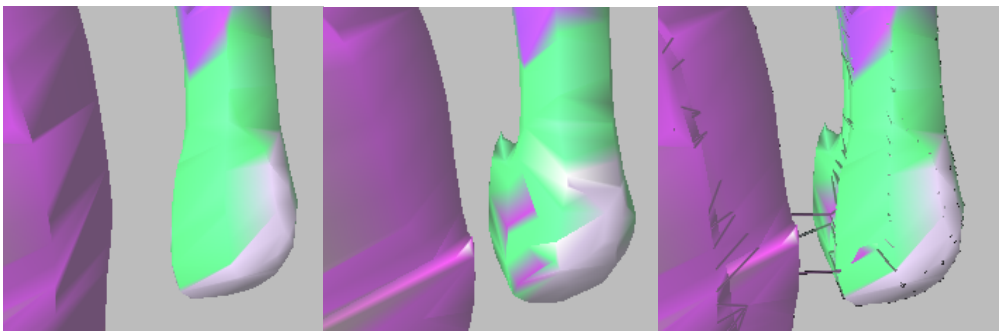
La mise en correspondance est fonctionnelle, elle souffre cependant de deux problèmes :

- elle est unidirectionnelle : dans le cas où $nv(F_k) < nv(F_{k+1})$, certains sommets de F_{k+1} n'ont pas de correspondants dans F_k, ce qui pose problème dans le transfert de segmentation comme nous le verrons dans la section suivante ;



(a) (b)

FIGURE 4.11 – Résultat de la mise en correspondance entre 2 images successives : (a) affichage complet. (b) affichage partiel.



(a) (b) (c)

FIGURE 4.12 – Erreur de mise en correspondance : (a) F_k. (b) F_{k+1}. (c) la mise en correspondance est erronée, certains sommets de la main sont mis en correspondance avec les hanches.

– indépendance de la mise en correspondance : lors de la mise en correspondance, chaque sommet est évalué sans tenir compte des autres. Cela peut poser des erreurs de mise en correspondance, comme le montre la figure 4.12. Une passe de régularisation pourrait être utilisée afin d'améliorer les résultats. En effet si tout un groupe de sommets sauf un ou deux sont mis en correspondance avec un autre groupe de sommets, les quelques sommets mis en correspondance avec des sommets plus distants (en nombre d'arêtes) sont des valeurs aberrantes de la mise en correspondance. Ces correspondances peuvent être corrigées en les rattachant avec un des sommets voisins.

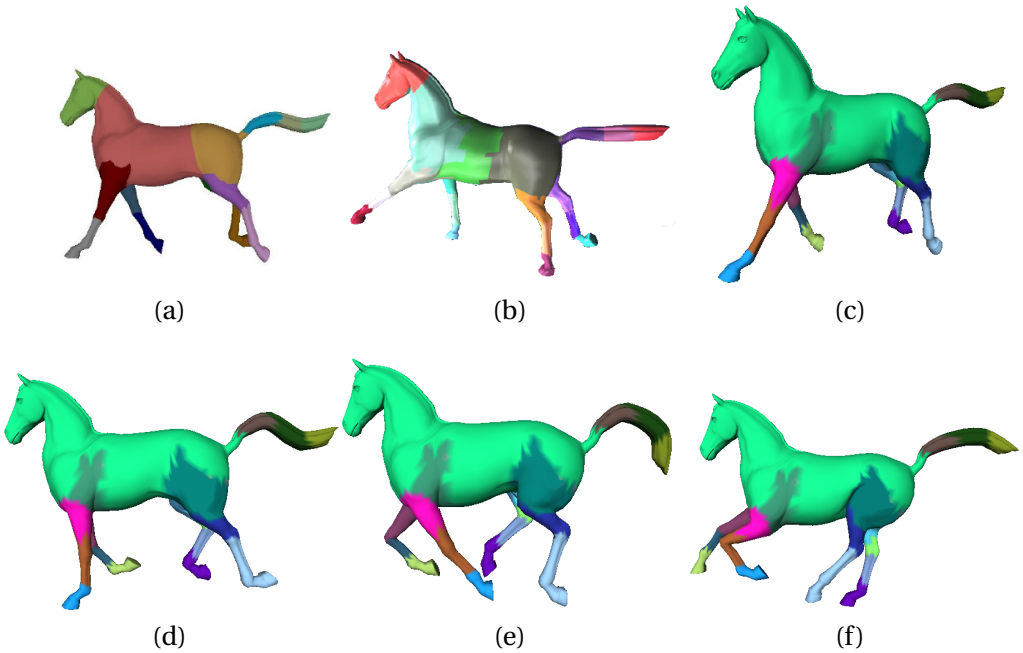


FIGURE 4.13 – Segmentation globale du maillage "horse" : (a) [46].(b) [25], (c) notre méthode. (d,e,f) Autres images de la séquence.

4.5.2 Segmentation

4.5.2.1 Résultats

La figure 4.13 montre un résultat obtenu avec notre méthode sur le maillage dynamique "horse", ainsi que ceux de [46] et [25]. Comme on peut le voir, notre méthode produit des résultats similaires aux autres méthodes. La principale différence avec [46] se situe sur les frontières des clusters, qui ne sont pas lisses et ne passe pas par les goulots d'étranglement. Nous n'appliquons pas de post-traitement pour déplacer les frontières : bien que cela puisse améliorer les résultats sur les maillages dynamiques, les frontières ne sont pas aussi nettement définies dans les autres types de séquences (voir la section 4.5.2.3).

Contrairement aux autres méthodes, notre canevas permet de gérer les étirements uniformes (voir la section 4.1 et 4.5.3.6 pour une discussion sur la nature des mouvements supportés). Nous montrons quelques résultats d'une telle segmentation dans la figure 4.14, qui montre une séquence dans laquelle se trouvent des transformations rigides et étirements uniformes.

Notre méthode est également capable de segmenter des séquences stables de maillages comme le montre la figure 4.15. La séquence stable est obtenue à partir d'une décimation aléatoire sur chaque image de la séquence. Le nombre de sommets varie entre 2624 et 4969. Le résultat obtenu est proche de celui obtenu sur le maillage dynamique "équivalent".

Notre méthode de segmentation peut, comme le montre la figure 4.16, produire une segmentation plus ou moins raffinée en fonction des paramètres : ainsi la transformation d'un cluster est plus ou moins proche des transforma-

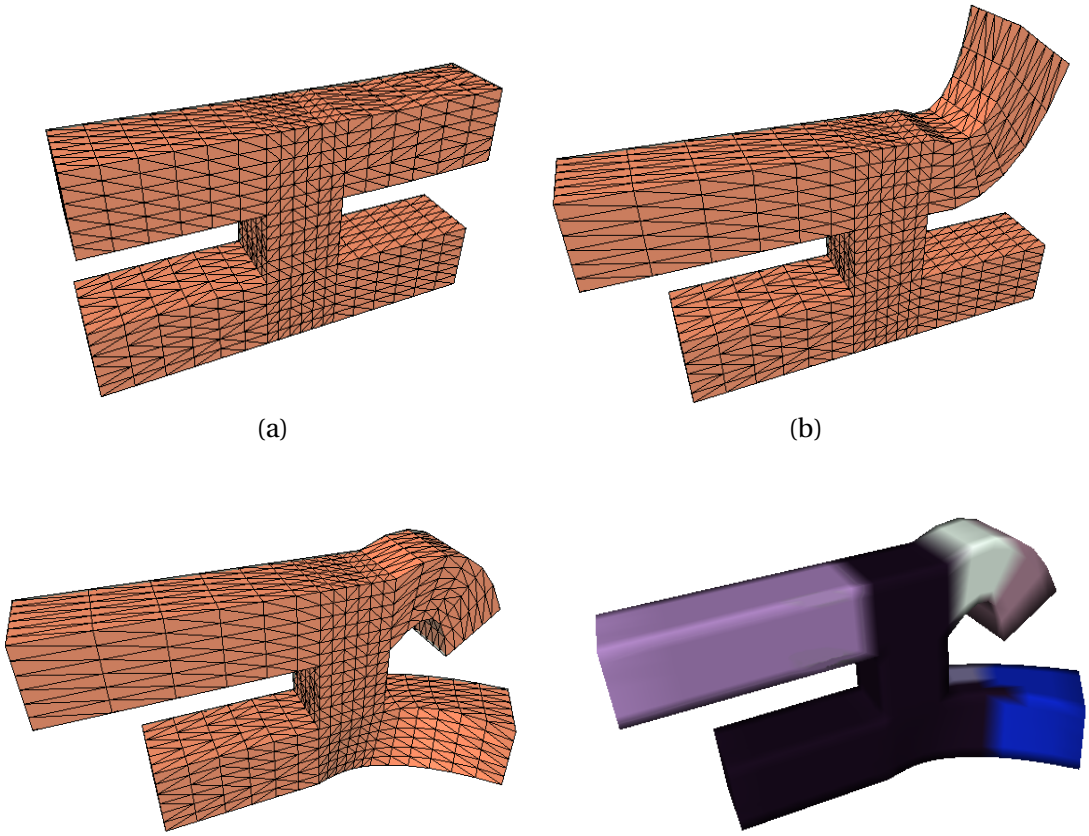


FIGURE 4.14 – Gestion des étirements : (a), (b), (c) images de la séquence. (d) segmentation obtenue.

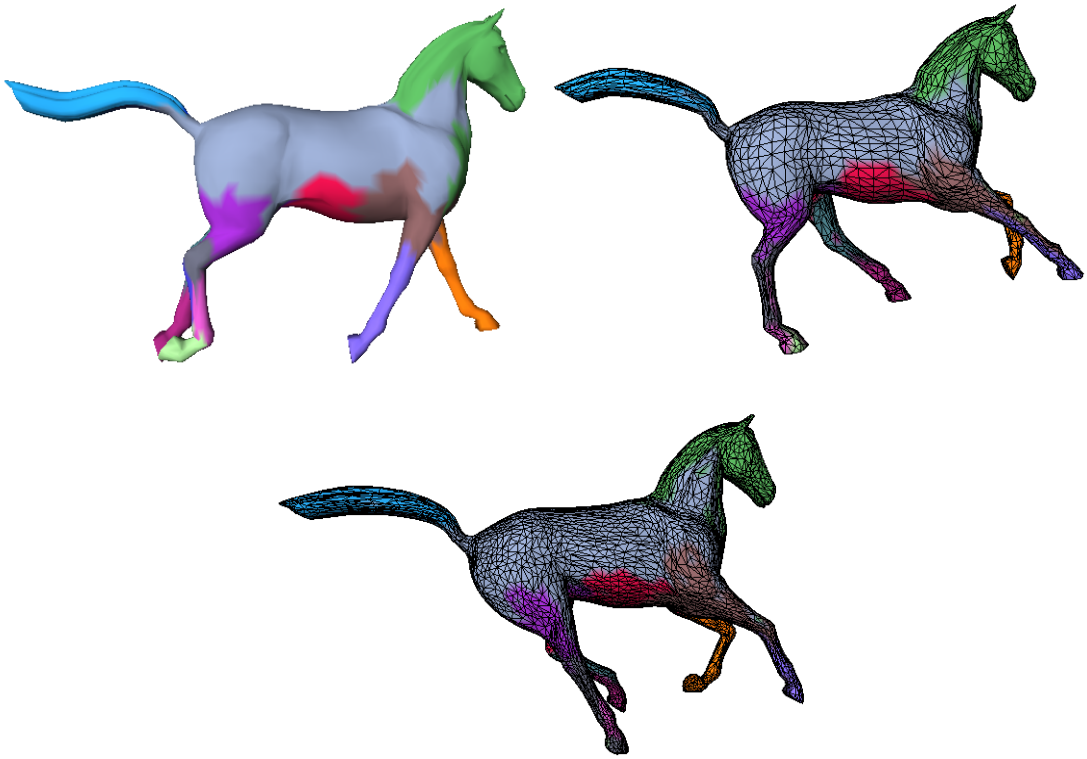
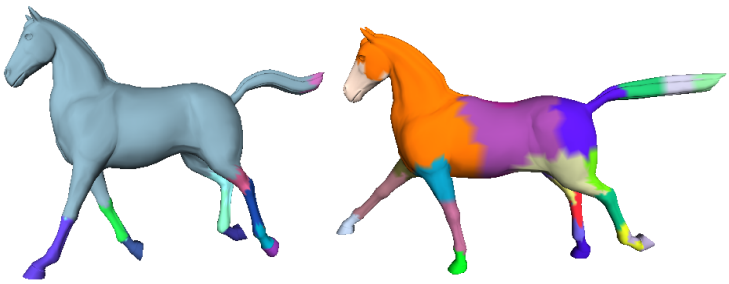


FIGURE 4.15 – Segmentation sur une séquence stable : (a) segmentation obtenue. (b,c) images contenant le nombre de sommets minimum (b) et maximum (c).



(a) (b)

FIGURE 4.16 – Paramètres et raffinement de la segmentation : (a) grossière ($T_{transf} = 0.0015$). (b) fine ($T_{transf} = 0.0007$). Pour comparaison, voir la figure 4.13c ($T_{transf} = 0.0009$)

tions individuelles des sommets dans le cluster, i.e. avec un raffinement élevé, les clusters serons petits mais l'erreur sur les transformations des clusters seront minimes vis à vis des sommets, inversement une segmentation grossière ne produira que les principaux mouvements mais l'erreur des transformations vis à vis des sommets seront importantes. Elle n'est cependant pas hiérarchique car les clusters d'un niveau $n + 1$ ne sont pas strictement des sous-clusters du niveau n.

4.5.2.2 Entrée utilisateur

Pour gérer le bruit présent dans les séquences et l'erreur due aux estimations des vecteurs de déplacement, nous utilisons le seuil T_{static}. Pour la séquence stable "horse" (voir la figure 4.15), le seuil est fixé à 1×10^{-4}, la boite englobante est de 12.3177. Le seuil T_{transf} correspond à l'amplitude du mouvement tolérée pour un cluster entre deux images. Pour les séquences stable de maillages, ce seuil doit prendre en compte l'erreur due aux projections, contrairement aux maillages dynamiques. La segmentation obtenue peut être plus ou moins raffinée en fonction de ce seuil comme on peut le voir dans la figure 4.16. Ce seuil ne dépend pas de la densité des maillages, mais de l'amplitude du mouvement entre deux images. Il dépend également de la séquence, des séquences avec un faible échantillonnage temporel nécessitent un seuil plus important.

Dans la section 4.4.2, nous proposons différents critères d'erreurs pour déterminer l'ordre des fusions. D'après nos expériences, la différence d'erreur est préférable. En effet, cette erreur évite de fusionner les clusters avec une large erreur. Les clusters avec une grande erreur dans les maillages dynamiques sont dus à des changements dans les transformations. En présence de bruit et de valeurs aberrantes, la différence d'erreur est trop sensible et l'erreur moyenne devient la plus efficace : les grandes erreurs peuvent provenir du bruit ou des valeurs aberrantes. Les clusters contenant de telles aberrations ne doivent pas être ignorés. La figure 4.17 montre les résultats en fonction des erreurs et de la présence de bruit (le bruit est appliqué indépendamment sur chaque image).

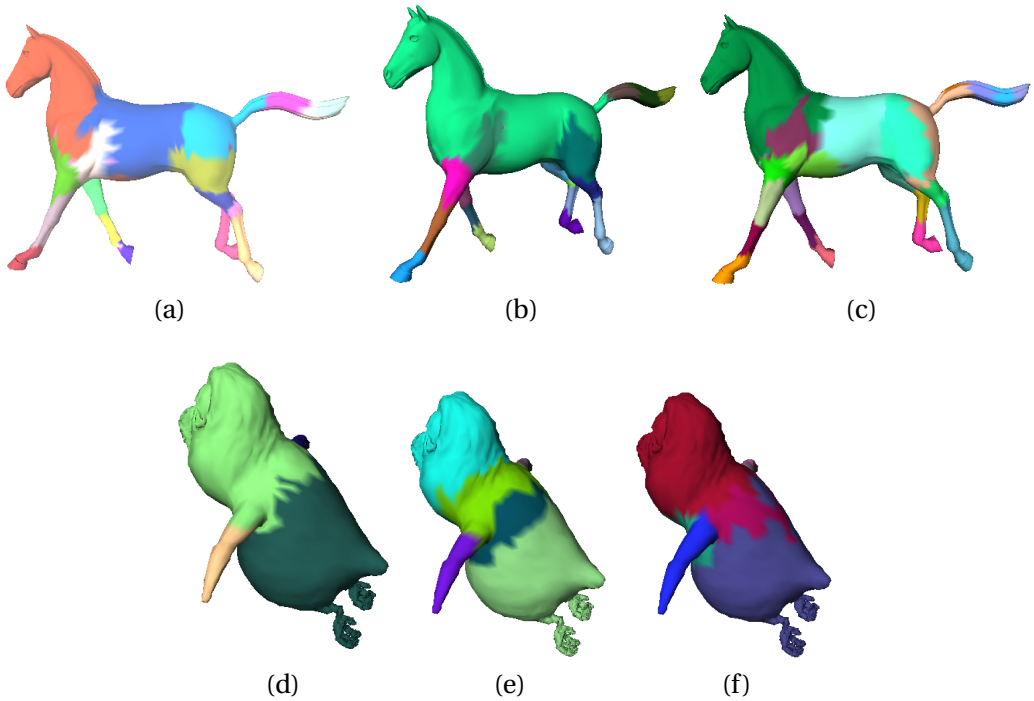


FIGURE 4.17 – Comparaison de segmentation, bruit et erreur. (a) Erreur moyenne. (b) Différence d'erreur. (c) Erreur maximum sur un maillage dynamique. (d) Erreur moyenne. (e) Différence d'erreur. (f) Erreur maximum sur un maillage dynamique bruité.

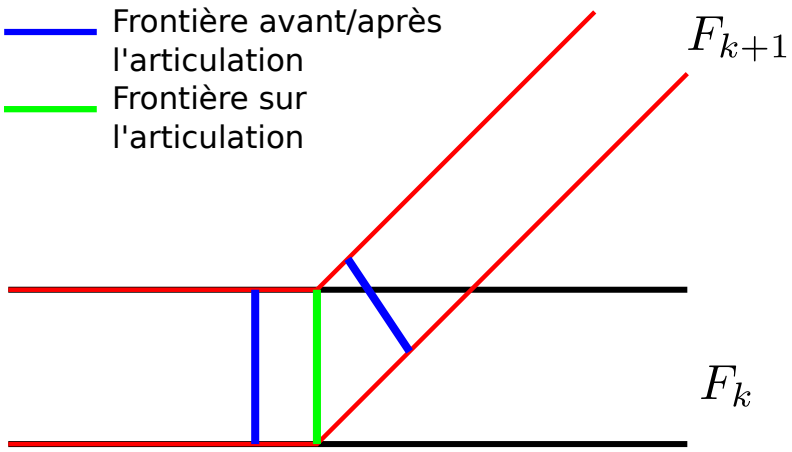


FIGURE 4.18 – Positionnement des frontières d'un cluster.

4.5.2.3 Frontière et résistance au bruit

Il y a deux problèmes, liés, concernant les frontières des clusters : elles ne sont ni lisses, ni situées nécessairement sur l'articulation comme cela est le cas avec certaines méthodes qui appliquent une étape de post-traitement afin de déplacer les frontières calculées sur les goulots d'étranglements qui correspondent généralement aux frontières optimales d'un cluster.

Les frontières optimales peuvent se situer aux alentours des frontières calculées par notre méthode, un peu avant ou après comme le montre la figure 4.18. Les clusters sont fusionnés tant que les transformations trouvées sont similaires ; or il arrive qu'elles ne soient plus similaires juste avant ou après la frontière optimale. Nous travaillons avec des séquences stables de maillages, donc même si nous plaçons de manière optimale les frontières sur une image, elles ne le seront pas sur les autres images. Comme nous voulons travailler avec les séquences d'origine, sans modification, cela est acceptable. Dans le cas contraire, une étape de post-traitement pour ajouter des sommets pour positionner correctement les frontières sur chaque image peut être possible. Le problème du lissage des frontières est similaire et peut être résolu de la même manière.

La présence de valeurs aberrantes ou de mauvaises correspondances est similaire au bruit. L'utilisation de l'erreur moyenne fusionne ces sommets aux clusters voisins. Les valeurs aberrantes restantes ont leurs propres clusters, qui sont de tailles limitées. Ils sont donc fusionnés par la suite avec les autres clusters. Cependant, en présence de valeurs aberrantes, ou de mauvaises mises en correspondance groupées, notre méthode produit un résultat erroné : ces sommets sont regroupés en un seul cluster qui est trop grand pour être fusionné. Cependant, la segmentation est toujours cohérente avec les informations fournies par la mise en correspondance (voir la figure 4.19). De mauvais résultats peuvent être également obtenus lorsque de nombreuses valeurs aberrantes sont présentes de manière éparse sur les maillages : cela oblige à utiliser un seuil

(a) (b) (c)

FIGURE 4.19 – Erreurs de mise en correspondance groupées. (a) F_k. (b) F_{k+1} : les clusters des pattes gauche et droite sont inversés. (c) Résultat de la mise en correspondance.

important et seule une segmentation grossière peut être générée. De plus, la segmentation sur la première image est correcte, mais elle est progressivement détruite lors de l'étape de transfert.

Enfin, les erreurs de mise en correspondance ainsi que les valeurs aberrantes provoquant la création d'un cluster à l'image F_k sont propagées sur toutes les images suivantes. Ce problème est dû à la nature de notre algorithme.

4.5.2.4 Mise en correspondance et segmentation

Nous avons choisi d'utiliser les points projetés pour l'estimation des vecteurs de déplacement. La conséquence est que nous obtenons une bonne estimation des vecteurs de déplacement, avec un coût en calcul important. En utilisant un algorithme tel que [41] qui ne fournit ni les points projetés, ni un maillage recalé, nous devons nous fonder sur les correspondances pour estimer les vecteurs de déplacement. Cela implique une erreur plus importante sur ces estimations. Sur des maillages denses, cela n'est pas très grave, car les correspondances sont proches de la position des sommets qui seraient projetés, cependant sur des maillages peu denses, l'erreur est suffisamment important pour :

- produire des résultats incorrects ;
- forcer l'utilisation d'une grande valeur pour T_{transf} (et obtenir une segmentation grossière).

4.5.2.5 Pile de mouvement

Le canevas proposé est capable de créer des clusters en fonction de différents types de mouvement : déplacement nul, étirement uniforme, et transformation rigide. Il est possible d'injecter de nouveaux types de mouvements dans

le canevas en définissant une nouvelle erreur et une description de ce mouvement en fonction des vecteurs de déplacement.

4.5.3 Limitations

Dans cette section, nous présentons les limitations connues de notre méthode, ainsi que des pistes pour les lever lorsque cela est possible.

4.5.3.1 Dépendance aux résultats de la mise en correspondance

Des erreurs de mise en correspondance peuvent provoquer des erreurs dans la segmentation. Notre méthode possède pour prérequis une mise en correspondance de bonne qualité, cela implique de bons résultats aussi bien pour les vecteurs de déplacement qui sont le cœur de notre méthode de segmentation que pour la correspondance qui nous permet de transférer notre segmentation.

4.5.3.2 Modèle et changement de topologie

L'une des limitations les plus importantes de notre méthode est l'utilisation du premier maillage de la séquence comme modèle pour la segmentation : nous utilisons donc en permanence la connectivité du premier maillage de la séquence, sur lequel on applique les vecteurs de déplacement estimés au cours de la séquence. Cela est problématique si ce maillage est moins dense que les autres et pour gérer les changements de topologie des objets sous-jacents. De plus nous accumulons les erreurs dans les reconstructions des mouvements dues à nos estimations.

4.5.3.3 Goulot d'étranglement

Si une image de la séquence contient significativement moins de sommets que les autres images de la séquence, cette image devient un goulot d'étranglement pour le reste de la séquence. Lors du passage sur cette image, de nombreux sommets sont fusionnés, ce qui pose problème à la fois pour la déformation du maillage et pour le transfert de la segmentation.

4.5.3.4 Correspondance unidirectionnelle et croissance de clusters

La correspondance unidirectionnelle provoque des trous dans la segmentation lors du transfert de la segmentation. Nous remplissons ces trous avec une simple méthode de croissance de clusters, qui ne tient pas compte des mouvements des sommets. Cette croissance de clusters peut provoquer un glissement des clusters sur les maillages au cours de la séquence. Une solution est proposée par la suite en utilisant une mise en correspondance bidirectionnelle.

4.5.3.5 Mouvement lent et subdivision erronée

L'utilisation de deux maillages simultanés est un avantage de notre méthode, mais elle pose cependant deux problèmes :

- il est possible de perdre les mouvements lents présents dans la séquence : lorsque la transformation entre deux images est trop petite, la vérification dans le processus de raffinement est toujours valide et le cluster est donc conservé en l'état. Le seuil T_{transf} limite l'importance de ce problème, mais n'est pas suffisant ;
- des valeurs aberrantes entre deux images peuvent provoquer la création d'un cluster. Regarder uniquement l'image suivante ne permet pas de déterminer si un mouvement est réellement en cours ou s'il s'agit d'une erreur.

Pour ces deux problèmes, l'utilisation d'une fenêtre temporelle de taille réduite, afin que les maillages de la fenêtre tiennent en mémoire, peut permettre de corriger ce problème en permettant d'analyser le mouvement sur plusieurs images.

4.5.3.6 Pile de mouvement

Comme mentionné ci-dessus, le canevas proposé est extensible et il est possible d'intégrer de nouveaux types de mouvements dans le canevas. Cependant cette intégration de nouveaux mouvements n'est pas triviale : il faut insérer la détection du mouvement en fonction des autres mouvements. Ainsi il faut tenir compte des types de mouvements déjà intégrés et placer la détection du nouveau type de mouvement de manière à ce que les autres types de mouvements ne soient pénalisés.

4.5.3.7 Dérive de la segmentation

Comme nous n'utilisons que deux images à la fois pour la mise en correspondance, la segmentation peut se déplacer petit à petit sur le maillage au cours du temps. Ce problème est renforcé par la croissance des clusters.

Nous avons présenté dans ce chapitre une méthode de segmentation globale fondée sur la fusion de régions, dans le prochain chapitre, nous présentons deux méthodes de segmentation utilisant la classification spectrale : la première produit une segmentation globale sur des séquences non-contraintes de maillages, tandis que la seconde génère soit une segmentation globale, soit une segmentation temporellement variable.

Chapitre 5

Méthodes de segmentation fondée sur la classification spectrale

Sommaire

5.1 Segmentation globale. 65
5.1.1 Amélioration visée par rapport à la méthode par fusion de clusters. 66
5.1.2 Mise en correspondance 66
5.1.3 Segmentation 68
5.1.4 Résultats et discussion 73
5.2 Segmentation temporellement variable 78
5.2.1 Introduction 78
5.2.2 Mise en correspondance 79
5.2.3 Segmentation 80
5.2.4 Résultats et discussion 81

Dans le chapitre précédent, nous avons présenté une première méthode de segmentation globale de séquences de maillages, capable de détecter des mouvements de nature différente, fondée sur la fusion de régions. Dans ce chapitre, nous présentons deux nouvelles méthodes fondées sur la segmentation spectrale, qui segmentent des séquences non contraintes de maillages en composantes rigides.

5.1 Segmentation globale.

Dans cette section, nous présentons une méthode de segmentation de séquences non contraintes de maillages soit en segmentation stable soit en un

ensemble de segmentations stables.

5.1.1 Amélioration visée par rapport à la méthode par fusion de clusters.

Cette méthode de segmentation est proche de la chaîne d'exécution de la méthode présentée dans le chapitre 4, mais remplace les deux principaux éléments de cette chaîne, la mise en correspondance et le processus de segmentation, par des éléments plus performants. De plus, une fenêtre temporelle de courte durée est introduite ainsi qu'une gestion des changements de topologie en post-traitement durant l'étape de transfert de la segmentation finale sur l'ensemble de la séquence. La chaîne d'exécution est présentée dans la figure 5.1 et le corps de l'algorithme est présenté dans l'algorithme 1. Elle garde donc les avantages de la précédente méthode, qui sont :

- la possibilité de traiter de longues séquences avec des maillages volumineux ;
- le contrôle de la création des clusters et le type de transformation associé ;
- la non nécessité de squelette en entrée.

Cette méthode vise à améliorer la méthode précédente de la manière suivante :

- mise en correspondance plus performante ;
- segmentation de meilleure qualité, et plus performante ;
- gestion des changements de topologie ;
- gestion des mouvements lents ;
- meilleure résistance au bruit.

Elle prend en entrée une séquence non contrainte, et produit soit :

- une segmentation globale s'il n'y a pas de changement de topologie ;
- un ensemble de segmentations globales si des changements de topologie sont présents dans la séquence.

La gestion de différents types de mouvement n'est pas utilisée dans cette méthode, ainsi les clusters créés dans cette segmentation sont des clusters rigides.

Dans la suite, nous présentons dans un premier temps le processus de mise en correspondance, puis le processus de segmentation, notamment le principe de la classification spectrale. Nous présentons ensuite le processus de segmentation et enfin les résultats obtenus avec notre méthode que nous analyserons.

5.1.2 Mise en correspondance

La mise en correspondance se fait en deux étapes : i) le recalage et ii) l'établissement de la mise en correspondance et l'estimation des vecteurs de déplacements. Nous n'utilisons plus d'étape de projection. A la fin de la mise en correspondance, nous obtenons comme dans la méthode précédente :

- les vecteurs de déplacement entre deux images successives F_k et F_{k+1}, $0 \leqslant k < N-1$;
- un ensemble de correspondance unidirectionnel entre les sommets des maillages F_k et F_{k+1}, $0 \leqslant k < N-1$.

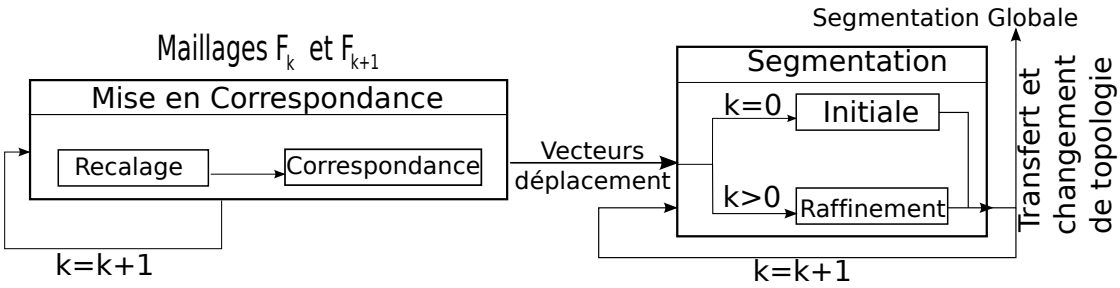


FIGURE 5.1 – Chaîne d'exécution.

De plus, en traversant les ensembles de correspondances sur l'ensemble de la séquence, nous obtenons un ensemble de correspondance entre F_0 et F_k, $0 \leqslant k < N$, voir la section 4.3.5. L'algorithme 6 et la figure 5.2 présente la chaîne d'exécution de la mise en correspondance.

Algorithme 6: Mise en correspondance.

Entrées : F_k, F_{k+1}, F'_k : Maillages
Sorties : MC : Sommet[Sommet]
DV : Sommet[Vecteur]
F'_k=Recalage(F_k,F_{k+1})
(MC,DV)=Correspondance(F_k, F'_k,F_{k+1})

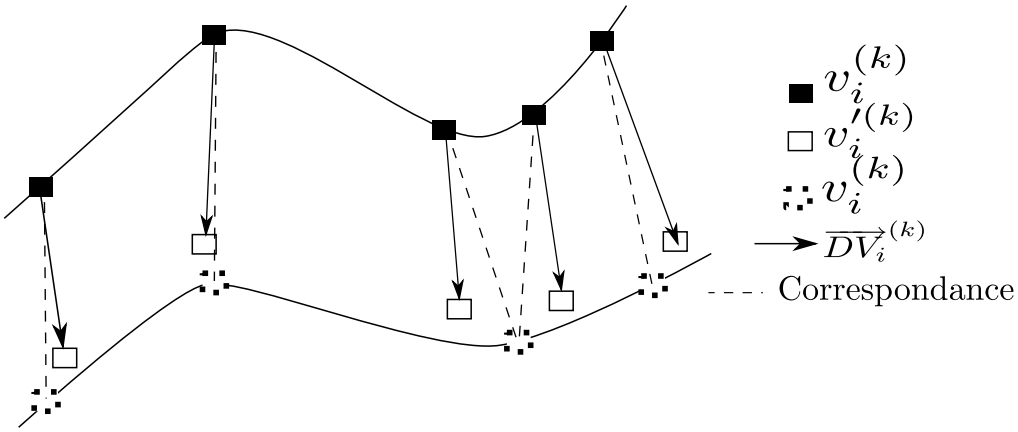


FIGURE 5.2 – Mise en correspondance.

5.1.2.1 Recalage

Pour cette étape, nous avons choisi d'implanter la méthode de Cagniart et al. [20]. En effet, cette méthode se fonde uniquement sur les sommets et les normales pour effectuer le recalage, ce qui correspond à nos critères.

Elle déforme itérativement le maillage F_k pour approcher le maillage F_{k+1} en dissociant la paramétrisation de la déformation de la complexité de la géométrie du maillage en entrée en divisant arbitrairement la surface en patchs. A chacun de ces patchs est associée une transformation rigide qui décrit la déforma-

tion locale par rapport à la pose initiale F_k. La procédure d'optimisation est inspirée de ICP [16] en estimant itérativement i) la correspondance des sommets entre le maillage déformé et le maillage F_{k+1}, puis ii) en minimisant la distance entre les deux nuages de points en pénalisant les transformations non rigides d'un patch par rapport à ses voisins. En utilisant cette méthode du grossier vers le fin, i.e. en faisant varier le diamètre des patchs, elle converge robustement dans nos expérimentations et est plus rapide que les méthodes n'utilisant pas de subdivisions.

5.1.2.2 Correspondance et vecteurs de déplacement

L'étape précédente nous fournit le maillage recalé F'_k de F_k avec F_{k+1}. Pour établir les correspondances entre les sommets des deux maillages, nous cherchons pour chaque sommet $v_i'^{(k)}$ de F'_k le sommet $v_j^{(k+1)}$ le plus proche dans F_{k+1} en utilisant la distance euclidienne : ce sommet $v_j^{(k+1)}$ est le sommet correspondant à $v_i^{(k)}$. Le vecteur de déplacement du sommet $v_i^{(k)}$ est simplement évalué : pour chaque sommet $v_i^{(k)}$ dans F_k et $v_i'^{(k)}$ avec $0 \leq i < nv(F_k)$, le vecteur de déplacement est $\overrightarrow{DV_i^{(k)}} = v_i'^{(k)} - v_i^{(k)}$. L'algorithme 7 illustre cette méthode.

Algorithme 7: Correspondance

Entrées : F_k,F'_k,F_{k+1}
Sorties : MC : Sommet[Sommet]
DV : Sommet[Vecteur]
pour *tout* $i \mid 0 \leq i < nv(F_k)$ **faire**
  $MC[v_i^{(k)}]$=plusProcheSommet($v_i'^{(k)}$,F_{k+1})
  $DV[v_i^{(k)}]$=$v_i'^{(k)} - v_i^{(k)}$

Nous obtenons ainsi les correspondances et les vecteurs de déplacement pour chaque sommet de F_k, cette correspondance est toujours unidirectionnelle.

5.1.3 Segmentation

5.1.3.1 Vue d'ensemble

Notre méthode produit soit une segmentation globale, ou par le transfert de segmentation un ensemble de segmentations globales, en créant des clusters rigides. Cette méthode utilise également le principe de raffinement, et la chaîne d'exécution est la même que pour l'étape de segmentation par fusion de clusters (voir la section 4.4.1), avec dans un premier temps un calcul de de la segmentation initiale (section 5.1.3.3) avec les deux premières images de la séquence, puis pour chaque paire d'images successives, un raffinement de la segmentation précédente (section 5.1.3.4). La segmentation finale de la séquence est donc sur la dernière image, et nous la transférons sur la séquence (section 5.1.3.5). Nous profitons de cette étape pour gérer les changements de topologie (section 5.1.3.5).

Nous conservons donc les avantages de la méthode par fusion de clusters : le traitement de longues séquences avec des maillages volumineux est possible ainsi que le traçage de la création et des transformations associées des clusters.

La segmentation est calculée en utilisant une classification spectrale entre deux images successives et en conservant la connectivité de la première image. Nous utilisons donc le premier maillage de la séquence comme modèle (voir la section 5.1.4.2 pour une analyse de ce choix). La matrice d'adjacence pondérée est différente de celle utilisée par de Aguiar [25] qui repose sur l'invariance de distance entre les sommets comme critère de rigidité. Dans notre cas, nous regroupons les sommets qui ont une matrice de transformation similaire, ce qui est une contrainte plus forte.

La classification spectrale est le cœur de notre méthode de segmentation. Nous présentons donc d'abord son principe, puis son application pour notre méthode.

5.1.3.2 Principe de la classification spectrale

Segmentation spectrale Dans cette section, nous présentons un algorithme de classification spectrale très populaire, celui de Shi and Malik [62].

Dans la suite, les valeurs propres sont toujours triées par ordre croissant, et lorsque nous mentionnons les k premiers vecteurs propres, nous faisons référence aux vecteurs propres correspondants aux k plus petites valeurs propres.

La classification spectrale repose sur le graphe de similitude, noté $G = (V, E)$, où $V = v_1, ..., v_n$ est l'ensemble des sommets, et E est l'ensemble des arêtes. Chaque arête reliant un sommet v_i et v_j, notée e_{ij}, est pondérée par un poids w_{ij} représentant la similarité entre les sommets v_i et v_j, tel que $w_{ij} \geq 0$ et

$$\begin{cases} w_{ij} \to 0 & : \text{les sommets } v_i \text{ et } v_j \text{ ne sont pas similaires} \\ w_{ij} \to \infty & : \text{les sommets } v_i \text{ et } v_j \text{ sont similaires} \end{cases} \tag{5.1}$$

Le but de la classification spectrale est donc de trouver une segmentation telle que les arêtes entre les sommets d'un même cluster aient un poids important, tandis que les arêtes entre les sommets de deux clusters différents aient un poids faible.

Le graphe G peut être représenté par une *matrice d'adjacence pondérée*, notée W, contenant le poids des arêtes, avec $w_{ij} = w_{ji}, \forall i \neq j$ et $w_{ii} = 0$. La matrice des degrés D, matrice diagonale, est également nécessaire :

$$\begin{cases} D_{ii} = \sum\limits_{j \in \mathrm{Ng}(v_i^{(k)})} w_{ij}. \\ D_{ij} = 0, \forall i \neq j. \end{cases} \tag{5.2}$$

La matrice laplacienne non normalisée est alors définie par :

$$L = D - W \tag{5.3}$$

La classification spectrale est alors calculée comme indiqué dans l'algorithme 8.

Algorithme 8: Segmentation spectrale normalisée selon Shi et Malik [62]

Entrées : matrice d'adjacence pondérée : W
matrice des degrés : D;
nombre de clusters : k;
Sortie : Clusters : C_1,...,C_k
$L = D - W$;
Calculer les k premiers vecteurs propres généralisés, u_1,...,u_k de $Lu = \lambda Du$;
U=matrice des u_1,...,u_k en colonnes;
Pour $i \in [1, n]$, soit y_i le vecteur correspondant à la i^{eme} ligne de U;
Cl_1,...,Cl_n=Appliquer l'algorithme k-means sur les points $(y_i)_{i=1,..,n}$;
C_1,...,C_k=$\{j|y_j \in Cl_1\}$,...,$\{j|y_j \in Cl_n\}$;

Cet algorithme utilise les vecteurs propres généralisés, qui correspondent aux vecteurs propres de la matrice L_{rw} :

$$L_{rw} = D^{-1}L = I - D^{-1}W \tag{5.4}$$

Pour plus de détails, nous invitons le lecteur à lire le tutoriel de U. von Luxburg [67].

Détermination du nombre de clusters Déterminer le nombre de clusters est un des enjeux importants pour obtenir une segmentation de bonne qualité. Il existe différentes méthodes pour déterminer ce nombre de clusters, en utilisant une approche bayésienne comme [26] par exemple.

Parmi ces méthodes, l'heuristique fondée sur la distance entre les valeurs propres est couramment utilisée. Nous utiliserons exceptionnellement le terme anglais "eigen-gap". Pour déterminer le nombre de clusters k, le but de cette heuristique est de choisir k tel que les valeurs propres, triées par ordre croissant, λ_1, ..., λ_k soient petites et λ_{k+1} soit grande. Autrement dit, on choisit k tel que $\lambda_{k+1} - \lambda_k$ est grand par rapport aux différences $\lambda_{i+1} - \lambda_i, i < k$.

5.1.3.3 Application à la segmentation de séquences de maillages

Nous utilisons les mêmes notations que celles de [67]. En utilisant la mise en correspondance et les vecteurs de déplacement entre deux images successives, nous construisons un graphe pondéré dont le poids des arêtes représente la similarité entre les transformations de deux sommets voisins. Nous calculons ensuite la matrice laplacienne normalisée associée au graphe et appliquons une classification spectrale sur cette matrice.

Construction du graphe d'adjacence Afin d'appliquer la classification spectrale, nous construisons la matrice d'adjacence pondérée du graphe associé aux transformations entre deux images. Les nœuds du graphe sont les sommets du maillage de la première image de la séquence. Pour construire ce graphe, nous calculons dans un premier temps la transformation qui transfère chaque sommet $v_i^{(k)}$ de l'image F_k et son 1-voisinage Ng($v_i^{(k)}$) vers le maillage reconstruit

précédemment plus les vecteurs de déplacement. Comme dans la méthode précédente, le maillage dans la première image est déformé au fur et à mesure en appliquant à chaque étape les vecteurs de déplacement. Cette transformation est calculée avec la méthode de Horn [34], qui construit une matrice de rotation 3×3 et un vecteur de translation représentant la meilleure transformation entre deux nuages de points. Nous combinons la matrice de rotation et le vecteur de translation en une matrice de transformation 4×4, et nous avons donc pour chaque sommet $v_i^{(k)}$ une transformation $T_i^{(k)}$. Nous utilisons la norme de Frobenius sur la matrice $T_i^{(k)} - T_j^{(k)}$ pour construire la matrice d'adjacence pondérée $W^{(k)}$:

$$\begin{cases} w_{ij}^{(k)} = \frac{1}{\|T_i^{(k)} - T_j^{(k)}\|_F}. \\ w_{ii}^{(k)} = 0. \end{cases} \tag{5.5}$$

Nous avons également testé notre méthode avec la matrice W_k suivante.

$$w_{ij}^{(k)} = \begin{cases} \frac{1}{\|\log(T_i^{(k)-1} T_j^{(k)})\|^2} & \text{if } i \neq j, \\ 0 & \text{if } i = j. \end{cases} \tag{5.6}$$

Comme cela est montrée dans [50], cette distance est la plus significative pour les transformations car elle correspond à la distance sur la variété des transformations rigides $SE(3)$. Le choix de cette distance est débattu dans la section 5.1.4.2.

Matrice laplacienne et segmentation En utilisant la matrice d'adjacence pondérée $W^{(k)}$, nous calculons la matrice laplacienne normalisée $L_{rw}^{(k)}$ et utilisons l'algorithme de classification spectrale normalisée de Shi-Malik [62] pour segmenter le graphe. Cette méthode requiert en entrée le nombre de clusters k. Nous évaluons k automatiquement en utilisant la méthode de l'"eigen gap", (voir la section 5.1.4.2).

$$D_{ii}^{(k)} = \sum_{j \in \mathrm{Ng}(v_i^{(k)})} w_{ij^{(k)}}. \tag{5.7}$$

$$L^{(k)} = D^{(k)} - W^{(k)}. \tag{5.8}$$

$$L_{rw}^{(k)} = D^{(k)-1} L^{(k)} = I^{(k)} - D^{(k)-1} W^{(k)}. \tag{5.9}$$

5.1.3.4 Etape de raffinement

A chaque instant k, nous raffinons la segmentation obtenue à l'instant $k-1$. Pour chaque cluster $C^{(k)}$, nous calculons la matrice d'adjacence pondérée des sommets du cluster, en utilisant les vecteurs de déplacement $\overrightarrow{DV^{(k)}}$ obtenus lors de l'étape de mise en correspondance. Nous utilisons toujours la connectivité de la première image : le maillage à l'instant k est donc construit en fonction des précédents vecteurs de déplacement. La matrice d'adjacence pondérée est

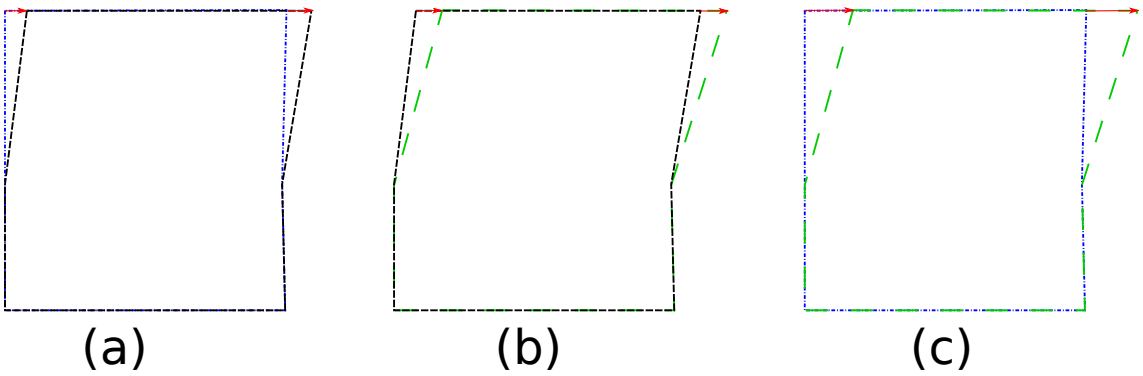


FIGURE 5.3 – Utilisation de la fenêtre temporelle par cluster. (a) Vecteur de déplacement entre (F_k,F_{k+1}) et (b) entre (F_{k+1},F_{k+2}) sont petits, (c) les vecteurs de déplacement entre (F_k,F_{k+2}) sont significatifs.

construite de la même manière que pour la segmentation initiale. La classification spectrale est appliquée sur chaque cluster $C^{(k)}$ et les sommets de chaque sous-cluster sont regroupés en composantes rigides.

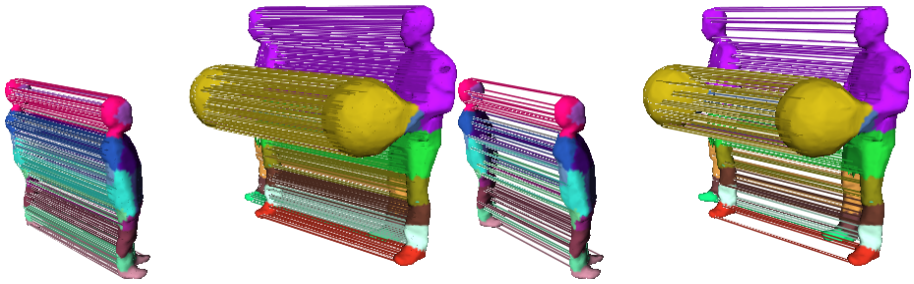
De plus pour améliorer la segmentation avec les mouvements lents et corriger une partie des erreurs de segmentation dues aux valeurs aberrantes et de mise en correspondance, faiblesse de la méthode proposée précédemment, nous introduisons une fenêtre temporelle par cluster pour guider la segmentation. Cette fenêtre est de taille suffisamment réduite pour garder la faible empreinte mémoire de notre algorithme. Lors de l'utilisation de la fenêtre temporelle par cluster, nous n'utilisons pas les vecteurs de déplacement entre les images successives. A la place, nous calculons les vecteurs de déplacement en utilisant :

- les sommets du cluster à l'image F_l avec F_l l'image où le cluster a été créé ;
- les sommets du cluster à l'image F_{k+1}.

Les clusters avec un mouvement lent ont des vecteurs de déplacement trop petits entre deux images pour pouvoir être détectés. Les vecteurs de déplacement calculés sur la fenêtre temporelle par cluster cumulent les mouvements sur plusieurs images, ainsi les vecteurs de déplacements sont significatifs, voir la figure 5.3. De plus, l'utilisation de cette fenêtre permet d'éviter une subdivisions erronée due aux erreurs de mise en correspondance et aux valeurs aberrantes, en analysant le mouvement sur une plage de temps plus grande.

5.1.3.5 Transfert de segmentation et changement de topologie

La segmentation de la séquence est obtenue sur la dernière image de la séquence et nous devons donc la transférer sur l'ensemble de la séquence. Cette étape est effectuée de la même manière que celle présentée en section 4.4.5. Nous profitons de cette étape pour gérer les changements de topologie : en effet, nous travaillons avec la connectivité du premier maillage de la séquence et les changements de topologie sont donc ignorés durant le processus de segmentation. Pour corriger ce problème, quand nous transférons un cluster sur l'image suivante, nous vérifions si le cluster est toujours connexe. Si ce n'est pas le cas,



(a) (b)

FIGURE 5.4 – Résultat de la mise en correspondance. (a) Complet. (b) Partiel.

le cluster est divisé en sous-clusters. Comme cela est fait en post-traitement, ces clusters ne sont pas présents sur l'ensemble de la séquence, mais uniquement à partir de l'image où le cluster apparaît : notre algorithme produit un ensemble de segmentations globales sur des séquences non contraintes de maillages.

5.1.4 Résultats et discussion

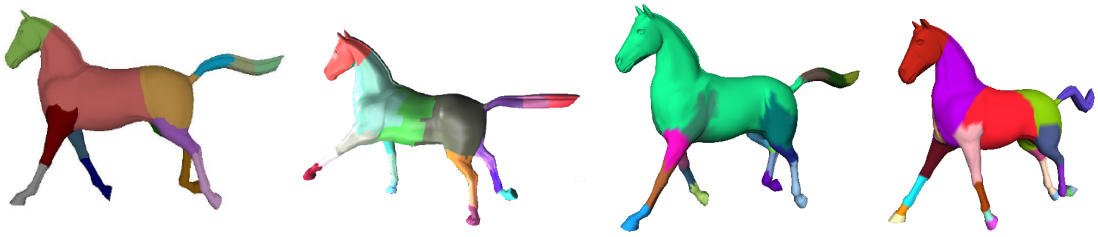
Dans cette partie, nous présentons dans un premier temps les résultats de notre méthode puis dans un second temps, une analyse ainsi qu'une comparaison avec la méthode précédente.

5.1.4.1 Résultats

Mise en correspondance La figure 5.4 montre un résultat de l'étape de mise en correspondance sur une séquence complexe : la séquence "Balloon" [36]. Le temps de calcul est d'une trentaine de secondes pour deux maillages contenant environ 145000 sommets chacun. Cette méthode est plus performante en temps de cacul que celle proposée dans la section 4.3, qui nécessite 13 minutes pour générer le même résultat.

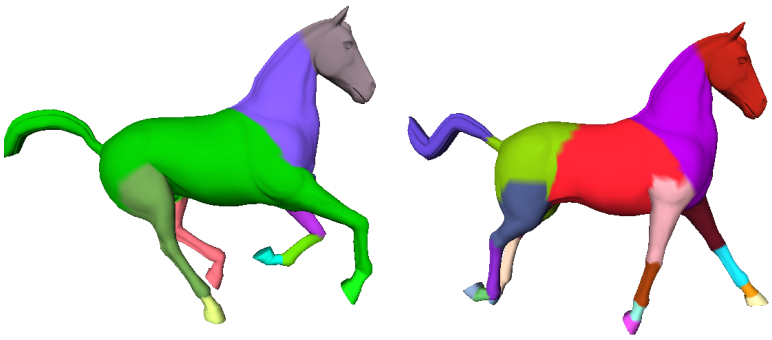
Segmentation La figure 5.5 compare les résultats obtenus sur le maillage dynamique "horse". Le résultat obtenu est proche ce celui proposé par les autres méthodes, et les frontières sont lisses sans post-traitement. Le principal défaut se situe sur la queue du cheval (voir la section 5.1.4.2) où nous ne retrouvons pas certains clusters, en contre-partie certaines parties de la segmentation sont plus détaillées, comme les pattes du cheval. Sur les maillages dynamiques, nous n'utilisons pas l'étape de mise en correspondance. La segmentation 5.5d utilisant la classification spectrale est plus proche des résultats obtenus par 5.5a et 5.5b que 5.5c, comme nous n'utilisons pas la mise en correspondance, cette amélioration est uniquement due à l'étape de segmentation.

La figure 5.6 montre les résultats obtenus avec différents seuils pour l'eigen gap, ce qui nous permet d'obtenir des segmentations plus ou moins fines. Ici



(a) (b) (c) (d)

FIGURE 5.5 – Segmentation sur un maillage dynamique : (a) [46]. (b) [25]. (c) 4. (d) notre méthode.



(a) (b)

FIGURE 5.6 – Raffinement de la segmentation sur la séquence dynamique "Horse". (a) la value de l'eigen-gap est de 0.7. (b) la valeur de l'eigen-gap est de 0.5

aussi, le raffinement est de meilleure qualité que celui proposé dans la méthode précédente, voir la figure 4.16.

La figure 5.7 montre le résultat obtenu sur la séquence stable générée à partir de la décimation aléatoire du maillage dynamique "Horse". Le nombre de sommets varie entre 2624 et 4696, ce qui est une importante variation. La méthode fondée sur la classification spectrale produit une segmentation plus proche de la segmentation sur le maillage dynamique que celle fondée sur la fusion de clusters. Nous perdons toujours des clusters, mais cela est en partie dû à la forte décimation qui supprime beaucoup de sommets.

Cependant l'exemple de la séquence "Horse", tant en maillage dynamique qu'en séquence stable, ne représente pas les séquences de maillage générées par reconstruction multi-caméras. Les résultats fournis sont présentés uniquement pour la comparaison avec l'état de l'art. La figure 5.8 montre une segmentation sur une telle séquence, la séquence "Balloon". Dans cette séquence, de nombreux changements de topologie interviennent. Les maillages sont de bonne qualités : quasi-réguliers et les triangles sont quasi-équilatéraux. Le nombre de sommets varie entre 14777 et 16546.

(a) (b)

FIGURE 5.7 – Résultat sur la séquence stable de maillages. (a) Section 4. (b) Notre nouvelle méthode.

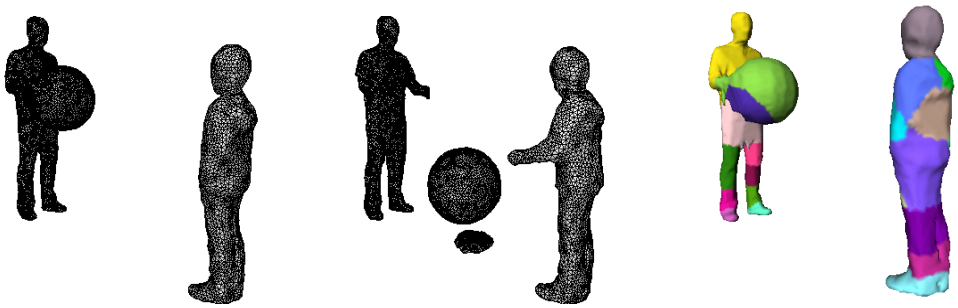
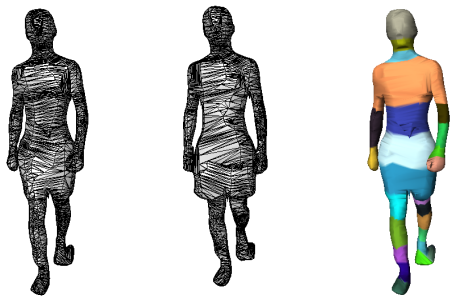


FIGURE 5.8 – Segmentation d'une séquence reconstruite par multi-caméras. (a, b) Images de la séquence "Balloon". (c) Segmentation.



(a) (b) (c)

FIGURE 5.9 – Segmentation sur la séquence "Walk". (a, b) Images de la séquence. (c) Segmentation utilisant la mise en correspondance de la section 4.3 et la classification spectrale de cette section.

5.1.4.2 Discussion

Différences avec 4 Les deux méthodes proposent un canevas similaire, mais les deux briques de base, la mise en correspondance et la segmentation ont été remplacées. Dans cette partie, nous analysons donc les avantages et inconvénients de la méthode fondée sur la classification spectrale par rapport à la méthode fondée sur la fusion de clusters.

La méthode de mise en correspondance est plus efficace que celle proposée précédemment, cependant elle possède une contrainte : les maillages doivent être de relativement bonne qualité : nous avons testé cette méthode sur la séquence "Walk" [1] : elle représente une personne en train de marcher. Elle utilise une méthode de reconstruction multi-caméras très simple et produit par conséquent des maillages irréguliers avec des triangles allongés. La mise en correspondance n'a pas pu gérer cette séquence. Celle proposée précédemment (section 4.4.5) réussit à établir une mise en correspondance, avec beaucoup d'erreurs, mais qui permet une segmentation comme le montre la figure 5.9.

Concernant le processus de segmentation, les résultats de la section précédente montrent la meilleure robustesse de la méthode par classification spectrale. Cette dernière est également plus performante, car elle met seulement 1 minute 30 pour deux images sur la séquence “Balloon” tandis que celle fondée sur la fusion de clusters nécessite 10 minutes. De plus, la segmentation produite par la méthode présentée en section 4 pour la séquence “Balloon” n'est pas cohérente.

L'utilisation d'une fenêtre temporelle permet également d'améliorer les résultats en permettant de détecter les mouvements lents, mais aussi de prévenir des subdivisions erronées dues à des valeurs aberrantes.

Entrée utilisateur Nous utilisons trois paramètres pour guider la segmentation : l'eigen gap, voir section 5.1.3.2, la taille minimale d'un cluster et la subdivision maximale en sous-clusters. En effet, pour déterminer le nombre de sous-

1. Courtoisie Lionel Réveret

clusters, nous utilisons l'eigen gap, qui est la source principale, mais cette méthode n'est pas toujours fiable. Pour compenser ce manque de fiabilité et empêcher une explosion du nombre de clusters dans les zones non rigides, nous utilisons les deux autres seuils. Ces seuils sont utilisés par exemples sur la séquence "Balloon" pour s'assurer que le ballon n'est pas sursegmenté, car il ne bouge pas de manière strictement rigide. La taille minimale de cluster est utilisée principalement pour améliorer les temps de calcul, mais également pour éviter d'avoir des petits clusters dans les zones d'articulations. Ainsi les clusters qui sont en dessous de ce seuil ne sont plus segmentés. Ce seuil est efficace sur les maillages reconstruits par multi-caméras qui sont réguliers. Il est fixé à 4% de la taille des maillages. La subdivision maximale en sous-clusters est utilisée (fixée à 8) sur la séquence du ballon pour assister la taille minimale de cluster : si ce seuil n'est pas utilisé, la balle est subdivisée en de nombreux petits clusters et un gros cluster à chaque pas de temps.

Changement de topologie Comme nous travaillons avec la connectivité du premier maillage durant le raffinement, l'algorithme ne peut pas réagir aux changements de topologie. Nous profitons de l'étape de transfert de segmentation pour gérer les changements de topologie. Comme cette étape est en post-traitement, les clusters n'apparaissent qu'à partir de l'image où le changement de topologie a lieu, et non pas depuis le début de la séquence.

La fusion de deux objets est possible (comme la fusion de la balle et de la main dans l'enfant dans la séquence "Balloon"). De tels changements peuvent provoquer des erreurs dans l'étape de transfert.

Distance Nous avons testé notre méthode avec 3 fonctions de distances entre transformations rigides différentes :

- l'erreur résiduelle entre les clusters, voir la section 4.4 ;
- la norme de Frobenius, voir la section 5.2.3 ;
- le logarithme de matrices, voir la section 5.2.3.

Nos expériences montrent que l'erreur résiduelle entre les clusters est difficile à utiliser correctement. Le logarithme fournit de meilleurs résultats, mais si le temps de calcul est une contrainte importante, alors la norme de Frobenius fournit tout de même de bons résultats. Sur les maillages dynamiques, ces deux distances fournissent les mêmes résultats, cependant sur des séquences non contraintes, le logarithme génère des clusters avec des frontières plus lisses.

Limitations Certaines limitations relevées ici sont déjà présentées dans la section 4.5.3, elles ne sont pas détaillées à nouveau. Par rapport à 4, les limitations suivantes sont toujours présentes : l'utilisation du premier maillage comme modèle, le goulot d'étranglement si un maillage de la séquence contient significativement moins de sommets que les autres images, la dérive de la segmentation, et la correspondance unidirectionnelle nécessitant une étape de propagation lors du transfert de segmentation.

Les limitations suivantes sont corrigées :

- gestion des changements de topologie : la nouvelle méthode permet maintenant de générer un ensemble de segmentations globales en fonction des changements de topologie, cependant cela intervient uniquement en post-traitement ;
- performances : les différentes briques du canevas sont plus performantes, en temps de calcul et en résultats fournis ;
- gestion de séquences complexes ;
- gestion des mouvements lents, et meilleure robustesse aux valeurs aberrantes.

Cependant, certaines limitations apparaissent :

- la classification spectrale est sensible à la régularité et à la longueur des arêtes. Pour corriger cela, il faut envisager il faut envisager l'utilisation d'une matrice spectrale telle que celle dite de "diffusion geometry" ;
- la segmentation ne gère pas correctement les cas d'apparitions et disparitions de géométrie : il faut en effet un support de la mise en correspondance pour gérer correctement ce cas la.

Dans cette section, nous avons proposé une première méthode de segmentation fondée sur la classification spectrale, qui génère soit une segmentation stable, soit un ensemble de segmentation stable en fonction du type de séquence en entrée. Dans la section suivante, nous proposons une méthode de segmentation produisant soit une segmentation temporellement variable, soit une segmentation globale. Elle permet en plus de s'abstraire de l'utilisation d'un modèle.

5.2 Segmentation temporellement variable

5.2.1 Introduction

Dans cette section, nous présentons une méthode de segmentation fondée sur la classification spectrale. Contrairement à la méthode présentée dans la section 5.1, cette méthode est capable de produire deux segmentations de nature différentes : une segmentation temporellement variable et une segmentation globale. Elle présente des similitudes dans son fonctionnement avec [19], une méthode de segmentation de vidéo.

5.2.1.1 But de la méthode

Cette méthode utilise le même principe de raffinement que la méthode de la section 5.1 ainsi que les mêmes blocs de base pour la mise en correspondance et la segmentation. Cette méthode diffère sur les points importants suivants :

- génération de deux types de segmentation ;
- elle n'est plus fondée sur un modèle : nous utilisons la connectivité du maillage de l'image en cours d'analyse. Nous pouvons gérer les changements de topologie au fur et à mesure et non plus en post-traitement ;

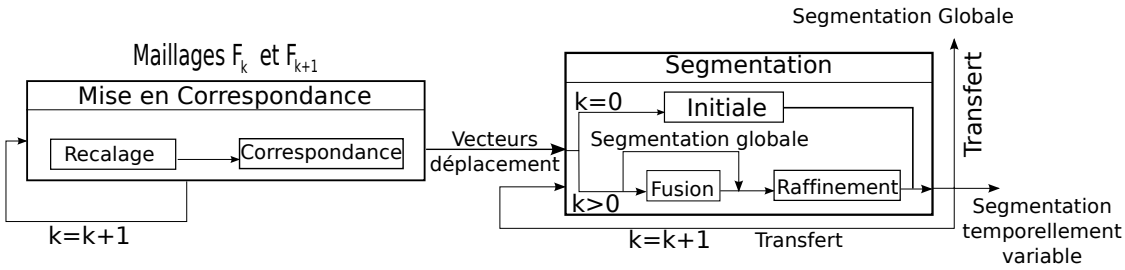


FIGURE 5.10 – Chaîne d'exécution

– mise en correspondance bidirectionnelle, cela nous permet de ne plus avoir de trou lors du transfert de segmentation et supprime l'étape de croissance de clusters.

Elle prend en entrée une séquence non contrainte, et produit :

– soit une segmentation globale ;
– soit une segmentation temporellement variable.

Le choix de la segmentation produite dépend uniquement d'un paramètre booléen. Les clusters créés dans cette segmentation sont des clusters rigides.

5.2.1.2 Chaîne d'exécution

Cette méthode prend en entrée une séquence non contrainte de maillages . Elle est fondée sur une méthode par raffinement et utilise une fenêtre temporelle de taille fixe pour guider la segmentation. Cette fenêtre est de taille réduite (fixée à 5 images dans nos exemples), permet de traiter de longues séquences avec des maillages volumineux.

Nous alternons toujours les étapes de mise en correspondance et de segmentation :

– mise en correspondance entre deux images consécutives et estimation des vecteurs de déplacement dans la fenêtre temporelle ;
– segmentation fondée sur les vecteurs de déplacement et transfert sur l'image suivante.

La chaîne d'exécution est présentée sur la figure 5.10.

5.2.2 Mise en correspondance

La mise en correspondance est similaire à celle effectuée dans 5.1 et se compose en deux étapes : i) recalage et ii) estimation des vecteurs de déplacements et établissement des correspondances. La mise en correspondance est illustrée sur la figure 5.11.

5.2.2.1 Recalage

Cette étape est la même que celle présentée en section 5.1.2.1.

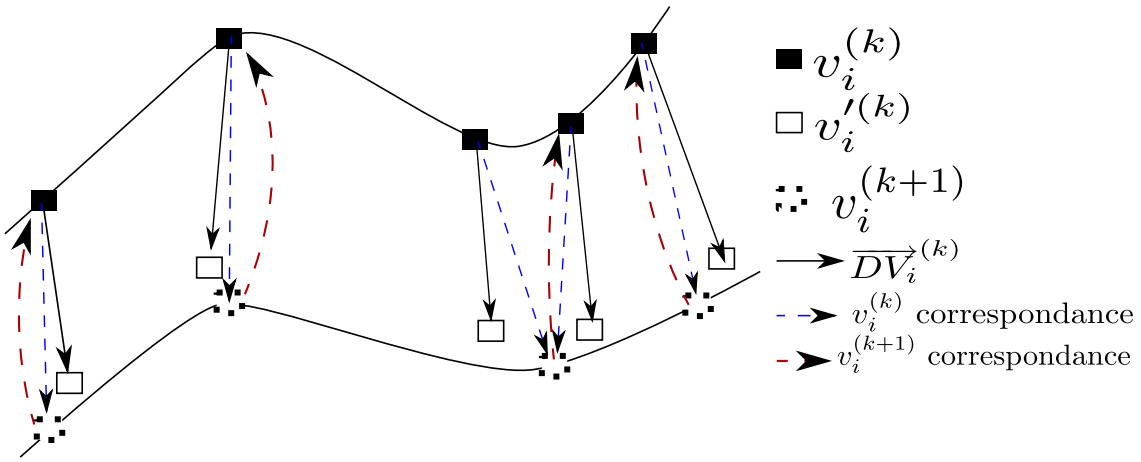


FIGURE 5.11 – Processus de mise en correspondance.

5.2.2.2 Mise en correspondance

Cette étape est la même que celle présentée en section 5.1.2.2. Nous calculons en plus des correspondances de F_k vers F_{k+1}, la mise en correspondance des sommets de F_{k+1} vers F_k de manière simplifiée. Nous ne faisons pas tout le processus de mise en correspondance (i.e pas de recalage de F_{k+1} vers F_k), mais uniquement la recherche du point le plus proche : pour chaque sommet de F_{k+1}, nous cherchons le sommet le plus proche de F'_k. Nous obtenons ainsi une correspondance bidirectionnelle.

5.2.3 Segmentation

Cette méthode est toujours fondée sur le principe de raffinement : pour chaque paire d'images consécutives, nous raffinons la segmentation précédente. La chaîne d'exécution (section 5.2.3.1) pour chaque pas de temps k est la suivante :

1. fusion des clusters voisins ayant un mouvement similaire. Cette étape n'est pas effectuée dans le cas d'une segmentation globale ;
2. segmentation sur les clusters en utilisant les clusters sur la fenêtre $[k,k+4]$;
3. transfert de la segmentation sur l'image suivante.

Dans le cas d'une segmentation globale, où il n'y pas de fusion de clusters, la segmentation obtenue sur la dernière image représente la segmentation de la séquence. Elle doit donc être transférée sur l'ensemble de la séquence.

Nous utilisons une fenêtre temporelle de taille 5 images, de k à $k+4$, pour calculer les déplacements des sommets de chaque image k.

5.2.3.1 Segmentation spectrale

Construction du graphe La construction est la segmentation du graphe est similaire à la méthode présentée en section 5.1.3.3

Fusion Lors de la génération d'une segmentation temporellement variable, nous fusionnons les clusters avec un mouvement similaire à chaque pas de temps avant la classification spectrale. La distance entre les transformations de deux clusters voisins est seuillée par T_{merge}, où les transformations sont estimées avec la méthode de Horn.

La fusion commence avec la plus petite erreur résiduelle, et finit quand on ne peut plus fusionner de cluster. L'erreur résiduelle d'un cluster correspond à la distance moyenne, pour chaque point $v_i^{(k)}$ du cluster, entre le point $v_i^{(k+1)}$ et $v_i^{(k)}$ avec la transformation rigide du cluster appliquée :

$$ErreurResiduelle(C) = \frac{\sum_{v_i(k) \in C} ||v_i^{(k+1)} - (T * v_i^{(k)})||}{card(C)}$$

avec T la transformation associée au cluster.

Un problème avec cette méthode provient du fait que nous avons deux seuils différents pour la fusion et la séparation. Ces deux seuils peuvent se contredire, on peut ainsi fusionner deux clusters qui seront aussitôt séparés par le processus de segmentation. Afin d'éviter ce problème, si deux clusters sont fusionnables, nous appliquons d'abord une classification spectrale sur l'ensemble des deux clusters. Si lors de cette segmentation a lieu une séparation, nous ne les fusionnons pas.

Transfert de segmentation sur l'image suivante La segmentation est calculée à chaque pas de temps sur l'image courante F_k. Les étiquettes sont ensuite transférées sur l'image suivante F_{k+1} en utilisant la mise en correspondance bidirectionnelle. Les clusters sont d'abord transférés en utilisant la correspondance de F_k vers F_{k+1}. Enfin pour chaque sommet sans cluster dans F_{k+1}, nous utilisons la correspondance de F_{k+1} vers F_k. Les clusters qui sont transférés sur deux maillages différents sont séparés.

5.2.3.2 Transfert de segmentation sur la séquence

Pour obtenir une segmentation globale visualisable sur l'ensemble de la séquence, la segmentation sur la dernière image doit être transférée sur l'ensemble de la séquence. Pour cela, nous utilisons le transfert de segmentation décrit dans la section précédente dans l'ordre inverse, de la dernière image vers la première. Pour chaque paire d'images (F_k, F_{k+1}), nous transférons d'abord en utilisant la correspondance de F_{k+1} vers F_k, puis de F_k vers F_{k+1} pour les sommets restants.

5.2.4 Résultats et discussion

5.2.4.1 Résultats

La figure 5.12 montre les segmentations temporellement variables obtenues sur les séquences "Dancer" [63] et "Balloon". La figure 5.13 montre un résultat

de segmentation globale. Par construction, les segmentations globales contiennent plus de clusters que les segmentations temporellement variables, car il n'y a pas de fusion. Les seuils pour les deux types de segmentations ont les mêmes valeurs, à part l'eigen gap qui est un peu plus bas pour les segmentations variables, la valeur de l'eigen gap est de 0.4 pour une segmentation temporellement variable contre 0.48 pour une segmentation globale. De manière générale, les seuils optimaux peuvent être facilement trouvés avec quelques essais. Le temps de calcul entre 2 images est approximativement de 3 minutes avec une implantation Matlab non optimisée.

5.2.4.2 Segmentation entre images distantes

Certaines méthodes de segmentation ne sont pas prévues pour segmenter des séquences de maillages, mais pour segmenter des maillages du même objet dans différentes positions [7]. Ces segmentations ont des contraintes différentes des segmentations de séquences de maillages, notamment sur le nombre de sommets possible dans les maillages. Bien que non prévu initialement par notre méthode, nous avons décider de la tester sur cet exercice. La figure 5.14 montre un résultat sur le maillage dynamique "Horse".

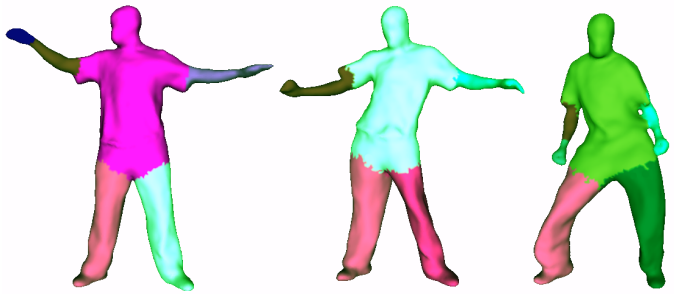
Les résultats sont de bonne qualité. La différence par rapport à la segmentation de séquences avec cette méthode réside dans la gestion des seuils. Tous les seuils ont la même valeur, hormis l'eigen gap qui est beaucoup plus élevé. Cette valeur plus élevée provient du fait que ce seuil “décrit” la tolérance pour la division des clusters. Nous n'avons pas testé sur d'autres séquences, mais nous n'anticipons pas de difficulté majeure sur des maillages avec des connectivités variables.

5.2.4.3 Discussion

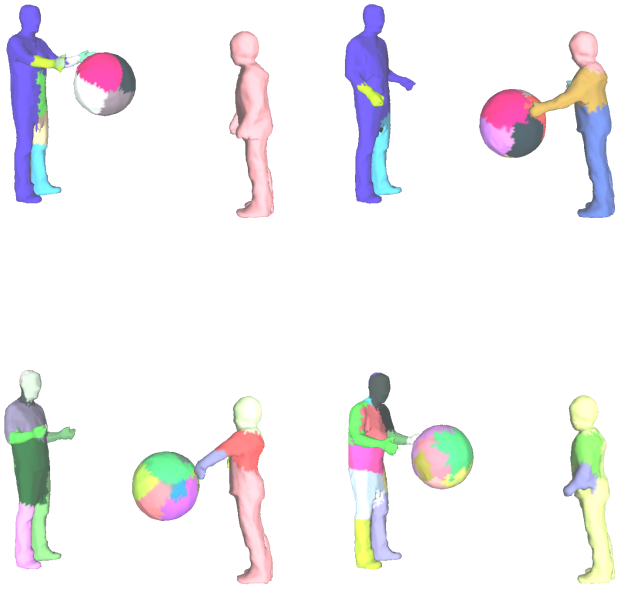
Entrée utilisateur Les seuils utilisés sont les mêmes que la méthode précédente, nous ne les présentons pas à nouveau (voir la section 5.1.4.2). Un seuil est ajouté, T_{merge}, il est utilisé pour déterminer si deux clusters représentent le même mouvement (voir la section 5.2.3.1). Il est lié au seuil de l'eigen gap, et les deux seuils doivent être déterminés ensemble : si ces deux seuils ne sont pas corrects, l'algorithme fusionnera en permanence les clusters divisés au temps précédent (voir la section 5.2.3.1 pour une solution facilement implantable).

Limitations Pour cette méthode, les limitations suivantes sont connues :

- les résultats dépendent clairement de la qualité de la mise en correspondance. Plus il y a d'erreurs dans la mise en correspondance, plus la possibilité d'une segmentation erronée est possible ;
- les clusters qui sont créés suite à des erreurs (de mise en correspondance par exemple) sont transférés sur les images suivantes. Ces erreurs de création de clusters sont de moindre importance dans une segmentation temporellement variable, car ils sont fusionnés par la suite, mais pour une segmentation globale, cela peut amener à un résultat erroné ;



(a)

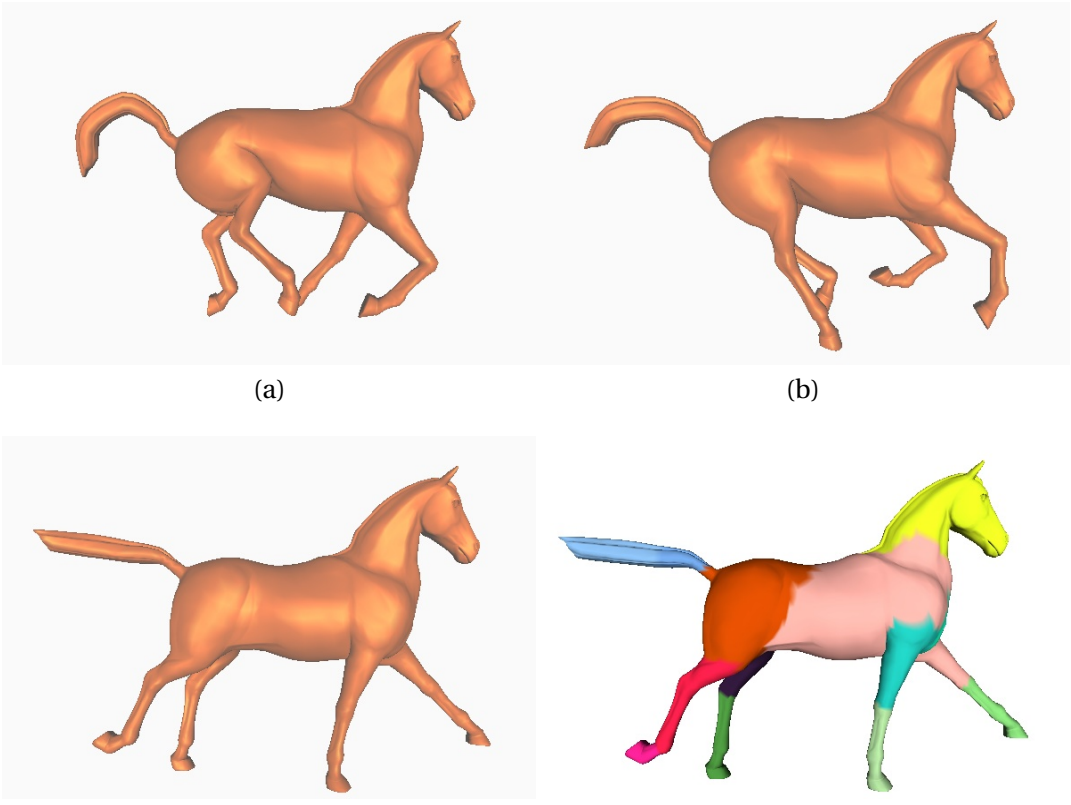


(b)

FIGURE 5.12 – Segmentation temporellement variable : (a) sur la séquence "Dancer". La première image est décomposée en 6 clusters, puis le bras et la main droite sont fusionnés, car elle bouge de manière similaire. Enfin, ce cluster est divisé à nouveau. A noter que les changements de topologie sont gérés (le bras est connecté au corps sur la dernière image). (b) sur la séquence "Balloon".

(a) (b)

FIGURE 5.13 – Segmentation globale sur la séquence "Balloon". (a) Segmentation raffinée. (b) Segmentation grossière.



(a) (b)

(c) (d)

FIGURE 5.14 – Segmentation entre images distantes.

- la segmentation peut toujours dériver, cependant ce problème est moins présent qu'auparavant, car la phase de propagation de cluster est remplacée par la mise en correspondance bidirectionnelle ;
- les changements de topologie sont gérés, cependant les disparitions et apparitions de géométrie restent problématiques. En effet, lors de la segmentation, celle-ci essaye de rattacher les clusters existants sur la nouvelle géométrie dans les cas d'apparitions. Ces clusters sont séparés car ils ne sont pas connexes, mais ce n'est pas une solution optimale. Pour pouvoir gérer de manière correcte ces cas d'apparitions/disparitions, un support dans la mise en correspondance est nécessaire afin de "prévenir" la segmentation de la particularité de ces cas.

Dans ce chapitre, nous avons proposé deux méthodes de segmentation fondées sur la classification spectrale. Ces deux méthodes fournissent des résultats qui sont visuellement probants. Dans le chapitre suivant, nous proposons une méthode d'évaluation quantitative afin de mesurer la pertinence d'une méthode de segmentation, sur laquelle nous testons les méthodes de segmentation présentées au cours de cette thèse.

Chapitre 6

Evaluation quantitative des méthodes de segmentation

Sommaire

6.1 Segmentation optimale : problème et définition 87
6.2 Critères d'erreurs 88
6.3 Résultats sur les méthodes de segmentation présentées 88
6.3.1 Segmentation globale 88
6.3.2 Segmentation temporellement variable 90
6.4 Conclusion 91

Diverses méthodes de segmentation sur les séquences de maillages ont été proposées dans ce mémoire et dans la littérature. Il n'existe pas cependant à notre connaissance de système d'évaluation objective de ces méthodes, et les comparaisons de segmentations se font visuellement. Nous proposons donc dans cette section un système d'évaluation quantitative.

6.1 Segmentation optimale : problème et définition

La segmentation *optimale* d'une séquence de maillages en composantes rigides peut être retrouvée lorsque les mouvements sont connus. C'est le cas par exemple pour les animations générées par squelette telles que celles créées avec un modeleur. Dans ce cas, chaque sommet est attaché à au moins un os du squelette avec un poids donné (voir la section 2.3.2.2). Ces os sont organisés hiérarchiquement. Pour notre évaluation, nous attachons chaque sommet à un seul os, le plus profond dans la hiérarchie. Si cet os n'est pas unique, nous gardons celui avec le poids d'attachement le plus important. Chaque os possède

son déplacement, mais plusieurs os peuvent bouger ensemble de manière rigide. Pour une image donnée, nous pouvons donc regrouper les os en ensembles d'os, chacun représentant un mouvement différent. Nous pouvons donc définir un *c*luster optimal comme l'ensemble des sommets liés aux os d'un ensemble d'os.

6.2 Critères d'erreurs

Nous proposons les critères d'erreurs suivants pour évaluer une segmentation donnée, fondée sur la définition de segmentation optimale précédente :

- Erreur d'affectation d'image (FAE, pour Frame Assignement Error) : pour une image donnée, le ratio de sommets qui ne sont pas affectés au bon cluster. Ceci inclut le cas des clusters qui ne sont pas créés, ou qui sont créés à tort ;
- Erreur d'affection de sommet (VAE, pour Vertex Assignement Error) : pour un sommet donné, le ratio d'images dans lesquelles le sommet n'est pas affecté au bon cluster. Cette erreur est définie uniquement sur les maillages dynamiques ou pour toute séquence de maillages pour laquelle on peut définir une correspondance unique entre les sommets de chaque image.

Intuitivement, la FAE permet de détecter quelles sont les images posant problème dans la segmentation, car un mouvement non rigide intervient par exemple. De manière similaire, la VAE permet de détecter quelles sont les zones posant problème, comme les frontières des clusters. Ces informations permettent donc d'identifier les erreurs dans les méthodes de segmentation et les cas limites.

6.3 Résultats sur les méthodes de segmentation présentées

Dans cette section, nous présentons les résultats des différentes méthodes proposées au cours de cette thèse en fonction de notre protocole de tests. La figure 6.1 montre les résultats de la FAE sur la séquence "Cat" [10], séquence générée avec un squelette dans un modeleur. Cette figure présente trois images extraites de la séquence, ainsi que les résultats.

6.3.1 Segmentation globale

Dans le cas d'une segmentation globale obtenue avec la méthode par fusion de régions, voir le chapitre 4, nous obtenons une FAE avec 42%, tandis que la VAE varie entre 0% et 100%.

Avec la segmentation globale obtenue avec la méthode présentée dans la section 5.1, sans prendre en compte les erreurs de mise en correspondance, car il s'agit d'un maillage dynamique, la FAE est la même pour toutes les images. Pour cette séquence, elle est de 17%. La VAE varie entre 0% et 100%, elle est

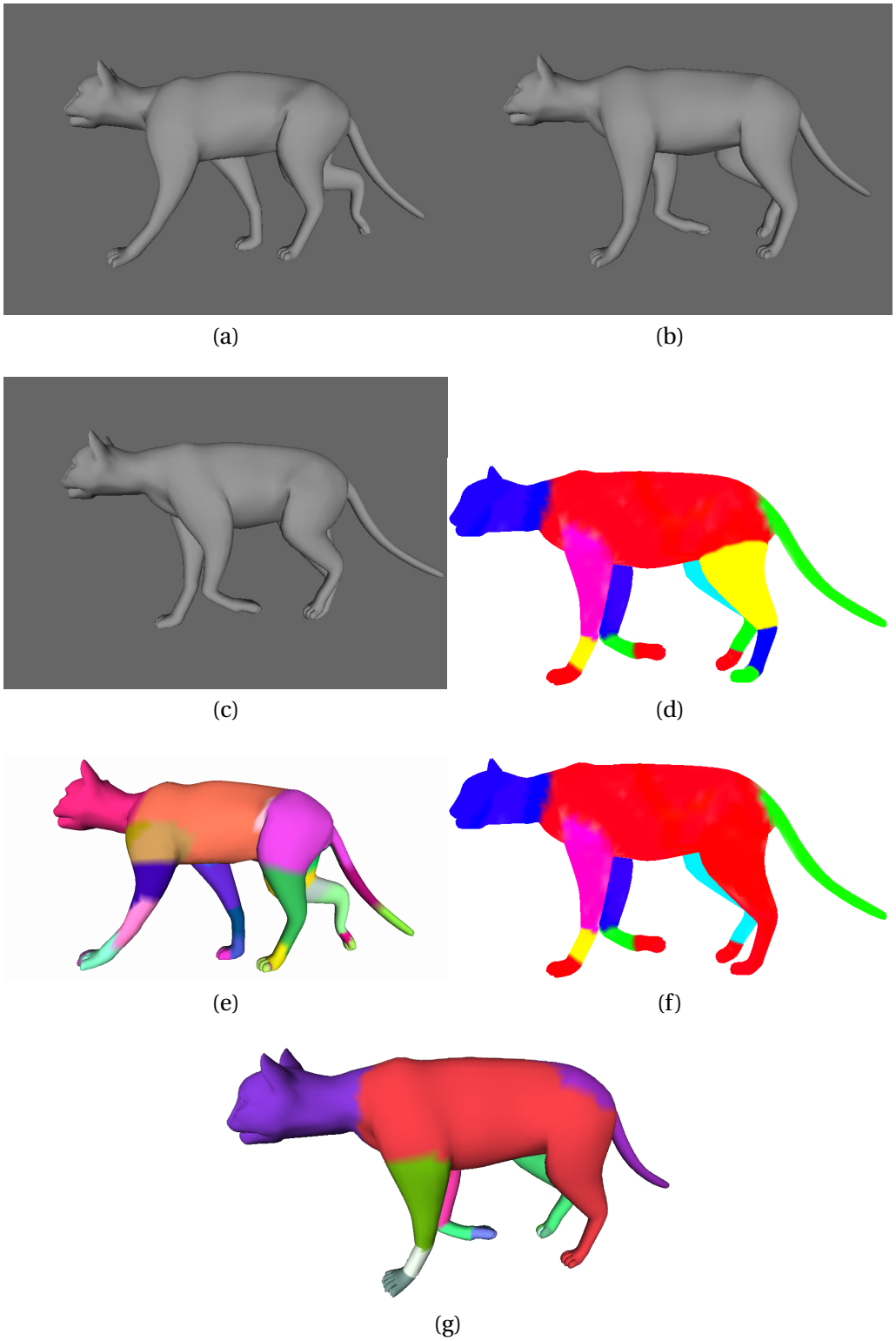


FIGURE 6.1 – Evaluation sur la séquence "Cat". (a,b,c) Images de la séquence. (d) Segmentation globale optimale. (e) Segmentation globale avec la méthode par fusion de régions (section 4). (f) Segmentation temporellement variable optimale sur une image de la séquence. (g) Segmentation temporellement variable obtenue sur la même image avec la méthode spectrale (section 5.2).

Méthode	FAE	Temps	Mémoire
Segmentation Globale			
Fusion de Régions	42%	20 minutes	++
Segmentation Spectrale 1 norme de Frobenius	19%	4 minutes	+
Segmentation Spectrale 2 norme de Frobenius	19%	4 minutes	+
Segmentation Spectrale 1 logarithme de matrice	17%	6 minutes	+
Segmentation Spectrale 2 logarithme de matrice	17%	6 minutes	+
Segmentation Temporellement Variable			
Segmentation Spectrale 2 norme de Frobenius	20%	5 minutes	+
Segmentation Spectrale 2 logarithme de matrice	17%	7 minutes	+

TABLE 6.1 – Résultat de l'évaluation qualitative sur la séquence "Cat" avec 5165 sommets. La méthode par fusion de régions (4) fournit globalement des résultats moins bons, tandis que les deux méthodes spectrales (1 : 5.1 et 2 : 5.2) fournissent les mêmes résultats sur des maillages dynamiques. La méthode par fusion de régions nécessite à la fois plus de temps et plus de mémoire. L'utilisation du logarithme améliore les résultats mais entraîne une hausse du temps de calcul.

pertinente uniquement en tant que critère relatif pour comparer les sommets dans les zones d'articulations. Sur cette séquence, les sommets dans les régions hautement rigides (les pattes, la queue, le corps) sont toujours correctement affectés, tandis que certains sommets sur les zones d'articulations sont toujours assignés à un cluster voisin.

La méthode présentée dans la section 5.2 produit les mêmes résultats car il s'agit d'un maillage dynamique, et cette dernière méthode n'améliore pas les résultats dans ce cas là.

6.3.2 Segmentation temporellement variable

La méthode présentée dans la section 5.2 est capable de produire des segmentations temporellement variables. Nous obtenons avec cette méthode une segmentation temporellement variable avec une FAE maximale de 17% (voir figure 6.1). Les sommets qui sont assignés de manière incorrecte correspondent aux zones d'articulations et à une erreur de création de cluster sur la patte du chat.

Le tableau 6.1 présente la synthèse des résultats pour la séquence "Cat".

6.4 Conclusion

Nous avons présenté une méthode d'évaluation qualitative permettant de juger de la pertinence d'une méthode de segmentation. Il est important de noter que cette méthode d'évaluation se fonde uniquement sur l'animation effective du maillage et ne prend pas en compte le mouvement perçu par un être humain.

Il serait également intéressant de tester cette méthode sur d'autres séquences pour évaluer sa robustesse, ainsi que d'évaluer les autres méthodes de segmentation existantes [27, 38, 46, 47, 56, 66].

Chapitre 7

Conclusion

Sommaire

7.1 Bilan **93**
7.2 Perspectives **94**
7.2.1 Segmentation 94
7.2.2 Protocole d'évaluation quantitative 96

7.1 Bilan

Cette thèse avait pour but de proposer des méthodes de segmentation de séquences de maillages. Nous avons proposé dans un premier temps une classification de séquences de maillages, permettant de désigner aisément les propriétés d'une séquence de maillages. Dans un second temps, nous avons présenté une classification des méthodes de segmentation de séquences de maillages, ce qui permet de définir précisément le type de segmentation généré par une méthode mais aussi quel type de segmentation est nécessaire pour un traitement de plus haut niveau.

Par la suite, nous avons présenté trois méthodes de segmentation. La première est de type segmentation globale et prend en entrée une séquence stable de maillages. Elle est fondée sur la fusion de régions segmentant les séquences en composantes rigides et uniformément étirées, et permettant de segmenter d'autres types de mouvement en formalisant le mouvement en termes de vecteurs de déplacement et en l'insérant dans le canevas. Les deux autres méthodes utilisent la classification spectrale. Elle prennent toutes les deux en entrée une séquence non contrainte de maillages. La première génère un ensemble de segmentations stables tandis que la deuxième produit soit une segmentation stable, soit une segmentation temporellement variable.

Nous avons ensuite présenté un protocole d'évaluation de segmentation, permettant d'évaluer les segmentations produites en fonction de deux critères,

la FAE pour Frame Assignement Error et la VAE pour Vertex Assignement Error. Le premier est le ratio de sommets qui ne sont pas affectés au bon cluster dans une image, tandis que le second est le ratio d'images dans lesquelles le sommet n'est pas affecté au bon cluster.

7.2 Perspectives

Nous présentons dans cette section des perspectives générales sur les méthodes de segmentation de séquence.

7.2.1 Segmentation

Nous avons présenté dans les chapitres précédents les différentes méthodes proposées au cours de cette thèse pour segmenter des séquences de maillages. Dans cette partie, nous présentons les différentes pistes pour améliorer les méthodes de segmentation proposées ainsi que les pistes qui nous paraissent intéressantes à étudier pour la segmentation de séquences en général dans le futur.

7.2.1.1 Améliorations des méthodes présentées

Les méthodes proposées bien que fonctionnelles peuvent être améliorées.

Mise en correspondance Dans toutes les méthodes que nous avons présentées, nous avons proposé une méthode de mise en correspondance et d'estimation de vecteurs de déplacement. Ces méthodes fournissent de bons résultats, cependant il s'agit d'un bloc autonome et si une méthode plus performante apparaît, il est tout à fait possible de l'utiliser. En fait, l'estimation des vecteurs de déplacement peut même utiliser une méthode différente fondée sur l'analyse de flux de scène 3D [60] par exemple, couplée avec une méthode de mise en correspondance telle que celle proposée par Knossow et al. [41].

Intégration de la fenêtre variable Dans la segmentation temporellement variable, nous utilisons une fenêtre temporelle de taille fixe, il serait intéressant d'utiliser une taille de fenêtre variable par clusters comme proposé dans la section 5.1, ainsi la durée du mouvement ne serait pas restreinte sur la fenêtre.

Gestion des frontières Comme le montrent les résultats de l'évaluation, les frontières des clusters dans la segmentation sont souvent légèrement décalées. Une des perspectives consiste donc à améliorer le positionnement des frontières.

Apparition et disparition de géométrie Les apparitions et disparitions de géométrie restent difficiles à gérer : en effet, nous disposons uniquement des informations de mise en correspondance ce qui n'est pas suffisant pour différencier des cas de séparation/fusion d'une apparition/disparition, et par conséquent de les gérer correctement. Il est nécessaire que le processus de mise en correspondance fournisse ces informations pour pouvoir les gérer dans la segmentation.

Dépendance à la régularité du maillage Nous utilisons une matrice laplacienne pour la classification spectrale, ce qui nous rend dépendants de la régularité du maillage. L'utilisation d'autres matrices spectrales pourrait être intéressante (carte de diffusion [22]).

Transfert des segmentations Le transfert de segmentation d'une image à la suivante , même avec la méthode bidirectionnelle, reste une étape critique. Une piste pour améliorer ce transfert peut être la méthode proposée par Sharma et al. [61].

7.2.1.2 Perspectives plus générales

Nous avons présenté les améliorations possibles sur les méthodes présentées au cours de cette thèse. Dans cette partie, nous nous intéressons plus généralement aux perspectives sur la segmentation de séquences de maillages, sous toutes ses formes.

Segmentation temporelle Nous nous sommes intéressés uniquement aux segmentations sur les sommets, et non aux segmentations en sous-séquences. Certains travaux existent [71]. Il serait intéressant de proposer des méthodes permettant d'indexer des séquences de maillages. Une méthode de segmentation temporelle pourrait se fonder sur la méthode proposée dans la section 5.2 pour analyser les mouvements en cours.

Segmentation fondée perception Les méthodes de segmentation actuelles se fondent uniquement sur le mouvement pour guider la segmentation, i.e. elles segmentent la séquence en fonction des mouvements apparaissant dans la séquence, sans savoir s'ils sont perçus ou non par un être humain. Il serait intéressant de tenir compte de la perception humaine pour guider la segmentation et ainsi ne tenir compte que des mouvements perçus. Cela serait extrêmement profitable pour la compression.

Résolution conjointe de la mise en correspondance et de la segmentation Dans nos méthodes, les étapes de mise en correspondance et de segmentation sont disjointes, or résoudre les deux problèmes simultanément est une piste prometteuse. Deux voies sont possibles :

– soit résoudre la mise en correspondance et la segmentation simultanément, comme le proposent Franco et al. [27] ;
– soit résoudre ces deux étapes itérativement : alterner les étapes de mise en correspondances et de segmentation en utilisant le résultat de la segmentation en entrée de la mise en correspondance afin de guider celle-ci.

Dans le canevas proposé dans les sections 5.1 et 5.2, la deuxième solution paraît implantable, notamment pour l'initialisation de la position et le nombre de patchs nécessaire pour le recalage.

7.2.2 Protocole d'évaluation quantitative

Dans cette thèse, nous avons proposé une méthode d'évaluation quantitative des méthodes de segmentation. Ce protocole est fondé uniquement sur l'animation de la séquence. Une des pistes suggérée dans la section précédente concerne la segmentation fondée perception de séquences de maillages. Une telle segmentation serait automatiquement mal notée par le système d'évaluation que nous avons proposé car il ne tient pas compte de la perception humaine pour évaluer la segmentation. Une piste qui nous semble intéressante est de développer un tel protocole comme il existe dans le cas des maillages statiques [14]. Cela nécessite cependant d'important travaux de recherche dans la perception du mouvement.

Bibliographie

[1] 3dtv project, 2011. URL http://www.3dtv-research.org/publications.php. 4

[2] Anr madras - anr-07-mdco-015, 2011. URL http://www-rech.telecom-lille1.eu/madras/. 4

[3] Anr morpho, 2011. URL http://morpho.inrialpes.fr/. 4

[4] A. Agathos, I. Pratikakis, S. J. Perantonis, and N. S. Sapidis. Protrusion-oriented 3d mesh segmentation. *The Visual Computer*, 26(1) :63–81, 2010. 26

[5] J. Allard, J.-S. Franco, C. Ménier, E. Boyer, and B. Raffin. The GrImage Platform : A Mixed Reality Environment for Interactions. In *4th International Conference on Computer Vision Systems, ICVS'06, January, 2006*, pages 46–46. IEEE, 2006. URL http://hal.inria.fr/inria-00349084/en/. 16

[6] R. Amjoun and W. Straßer. Efficient compression of 3d dynamic mesh sequences. In *Journal of the WSCG*, 2007. 2, 22, 33, 35

[7] D. Anguelov, D. Koller, H.-C. Pang, P. Srinivasan, and S. Thrun. Recovering articulated object models from 3d range data. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, UAI '04, pages 18–26. AUAI Press, 2004. ISBN 0-9749039-0-6. 82

[8] M. Attene, B. Falcidieno, and M. Spagnuolo. M. : Hierarchical mesh segmentation based on fitting primitives. *The Visual Computer*, 22 :181–193, 2006. 25

[9] M. Attene, S. Katz, M. Mortara, G. Patane, M. Spagnuolo, and A. Tal. Mesh segmentation - a comparative study. In *Proceedings of the IEEE International Conference on Shape Modeling and Applications 2006*, pages 7–. IEEE Computer Society, 2006. ISBN 0-7695-2591-1. 25

[10] G. Aujay, F. Hétroy, F. Lazarus, and C. Depraz. Harmonic skeleton for realistic character animation. In *Symposium on Computer Animation, SCA 07, August, 2007*, pages 151–160. Eurographics, aou 2007. 88

[11] I. Baran, D. Vlasic, E. Grinspun, and J. Popović. Semantic deformation transfer. *ACM Transactions on Graphics*, 28(3) :36, 2009. 2, 3

[12] B. G. Baumgart. *Geometric modeling for computer vision*. PhD thesis, Stanford, CA, USA, 1974. AAI7506806. 15

[13] D. Bechmann and B. Peroche. *Informatique Graphique, modélisation géométrique et animation*. Traité IC2 - Information - Commande - Communication. Hermès, Feb 2007. ISBN 978-2-7462-1514-6. URL http://lsiit-cnrs.unistra.fr/Publications/2007/11-BP07. 9

[14] H. Benhabiles, J.-P. Vandeborre, G. Lavoué, and M. Daoudi. A framework for the objective evaluation of segmentation algorithms using a ground-truth of human segmented 3D-models. In *IEEE International Conference on Shape Modeling and Applications (Shape Modeling International 2009)*, June 26-28 2009. 26, 96

[15] H. Benhabiles, J.-P. Vandeborre, G. Lavoué, and M. Daoudi. A comparative study of existing metrics for 3d-mesh segmentation evaluation. *The Visual Computer*, 26 :1451–1466, 2010. ISSN 0178-2789. 26

[16] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14 :239–256, February 1992. ISSN 0162-8828. 38, 68

[17] S. Beucher and C. Lantuejoul. Use of watersheds in contour detection. In *International Conference on Image Processing*, 1979. 22

[18] Y. Boulfani, M. Antonini, and F. Payan. Motion-based mesh clustering for mcdwt compression of 3d animated meshes. In *Proceedings of EUSIPCO'2007*, September 2007. 2, 34

[19] T. Brox and J. Malik. Object segmentation by long term analysis of point trajectories. In *ECCV*, 2010. 78

[20] C. Cagniart, E. Boyer, and S. Ilic. Iterative mesh deformation for dense surface tracking. In *ICCV Workshop on 3-D Digital Imaging and Modeling*, 2009. 67

[21] X. Chen, A. Golovinskiy, , and T. Funkhouser. A benchmark for 3D mesh segmentation. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 28(3), Aug. 2009. 26

[22] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, F. Warner, and S. Zucker. Geometric diffusions as a tool for harmonic analysis and structure definition of data : Diffusion maps. In *Proceedings of the National Academy of Sciences*, pages 7426–7431, 2005. 95

[23] G. Collins and A. Hilton. A rigid transform basis for animation compression and level of detail. In *Proceedings of the IMA Conference on Vision, Video and Graphics*, 2005. 2, 33

[24] F. Cuzzolin, D. Mateus, D. Knossow, E. Boyer, and R. Horaud. Coherent Laplacian 3-D Protrusion Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR '08)*, pages 1–8. IEEE Computer Society, 2008. 36

[25] E. de Aguiar, C. Theobalt, S. Thrun, and H.-P. Seidel. Automatic conversion of mesh animations into skeleton-based animations. *Computer Graphics Forum (Proc. Eurographics EG'08)*, 27, 4 2008. 2, 3, 22, 32, 34, 35, 55, 69, 74

[26] C. Fraley and A. E. Raftery. Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97 (458) :611–631, 2002. 70

[27] J.-S. Franco and E. Boyer. Learning Temporally Consistent Rigidities. In *IEEE Computer Vision and Pattern Recognition*, pages 1241–1248, 2011. 32, 36, 37, 91, 96

[28] M. Garland, A. Willmott, and P. Heckbert. Hierarchical face clustering on polygonal surfaces. In *I3D '01 : Proceedings of the 2001 symposium on Interactive 3D graphics*, pages 49–58, 2001. 22

[29] J. Günther, H. Friedrich, I. Wald, H.-P. Seidel, and P. Slusallek. Ray tracing animated scenes using motion decomposition. *Computer Graphics Forum (Proceedings of Eurographics)*, 25(3) :517–525, Sept. 2006. xi, 33

[30] A. B. Hamza and H. Krim. Geodesic object representation and recognition. In *DGCI*, pages 378–387, 2003. 38

[31] S.-R. Han, T. Yamasaki, and K. Aizawa. Time-varying mesh compression using an extended block matching algorithm. *IEEE Trans. Circuits Syst. Video Techn.*, 17(11) :1506–1518, 2007. 2

[32] S.-R. Han, T. Yamasaki, and K. Aizawa. Automatic preview video generation for mesh sequences. In *ICIP*, pages 2945–2948, 2010. 4

[33] F. Hétroy, C. Gérot, L. Lu, and B. Thibert. Simple flexible skinning based on manifold modeling. In *International Conference on Computer Graphics Theory and Applications (GRAPP)*, Lisbon, Portugal, 2009. URL `http://hal.inria.fr/inria-00339413/en/`. Projet IMAG/ELESA/INRIA MEGA. 13

[34] B. Horn. Closed-form solution of absolute orientation using unit quaternions. *J. Opt. Soc. Am. A*, 4(4) :629–642, 1987. 47, 48, 71

[35] Q. Huang, B. Adams, M. Wicke, and L. Guibas. Non-rigid registration under isometric deformations. *Computer Graphics Forum*, 27(5) :1449–1457, 2008. viii, xii, 47, 48, 49

[36] Inria. 4drepository, 2011. URL http://4drepository.inrialpes.fr/pages/home. 18, 73

[37] D. L. James and C. D. Twigg. Skinning mesh animations. *ACM Transactions on Graphics (SIGGRAPH 2005)*, 24(3), aug 2005. xi, 33, 35

[38] E. Kalafatlar and Y. Yemez. 3d articulated shape segmentation using motion information. In *Proceedings of the 2010 20th International Conference on Pattern Recognition*, ICPR '10, pages 3595–3598. IEEE Computer Society, 2010. ISBN 978-0-7695-4109-9. 33, 91

[39] Z. Karni and C. Gotsman. Spectral compression of mesh geometry. In *Siggraph 2000, Computer Graphics Proceedings*, pages 279–286, 2000. 22

[40] S. Katz, G. Leifman, and A. Tal. Mesh segmentation using feature point and core extraction. *The Visual Computer*, 21(8-10) :649–658, 2005. 26

[41] D. Knossow, A. Sharma, D. Mateus, and R. Horaud. Inexact matching of large and sparse graphs using laplacian eigenvectors. In *GbRPR '09 : Proceedings of the 7th IAPR-TC-15 International Workshop on Graph-Based Representations in Pattern Recognition*, pages 144–153. Springer-Verlag, 2009. 46, 61, 94

[42] A. Laurentini. The visual hull : A new tool for contour-based image understanding. In *7th Scandinavian Conf. Image Analysis*, pages 993–1002, 1991. 15

[43] G. Lavoué, F. Dupont, and A. Baskurt. A new CAD mesh segmentation method, based on curvature tensor analysis. *Computer Aided Design*, 37(10) : 975–987, 2005. 25

[44] N. S. Lee, T. Yamasaki, and K. Aizawa. Hierarchical mesh decomposition and motion tracking for time-varying-meshes. In *ICME*, pages 1565–1568, 2008. 36, 37

[45] T.-Y. Lee, P.-H. Lin, S.-U. Yan, and C.-H. Lin. Mesh decomposition using motion information from animation sequences : Animating geometrical models. *Comput. Animat. Virtual Worlds*, 16 :519–529, July 2005. 33

[46] T.-Y. Lee, Y.-S. Wang, and T.-G. Chen. Segmenting a deforming mesh into near-rigid components. *Visual Computer*, 22(9) :729–739, 2006. ISSN 0178-2789. xi, 2, 3, 22, 28, 32, 33, 35, 55, 74, 91

[47] J. E. Lengyel. Compression of time-dependent geometry. In *I3D '99 : Proceedings of the 1999 symposium on Interactive 3D graphics*, pages 89–95. ACM, 1999. ISBN 1-58113-082-1. 2, 22, 33, 91

[48] B. Lévy, S. Petitjean, N. Ray, and J. Maillot. Least squares conformal maps for automatic texture atlas generation. In *ACM SIGGRAPH conference proceedings*, pages 362–371, 2002. 22

[49] K. Mamou, T. Zaharia, and F. Preteux. Multi-chart geometry video : A compact representation for 3d animations. *3D Data Processing Visualization and Transmission, International Symposium on*, 0 :711–718, 2006. 33, 35

[50] R. M. Murray, S. S. Sastry, and L. Zexiang. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Inc., 1994. 71

[51] A. Myronenko, X. Song, and M. Carreira-Perpinan. Non-rigid point set registration : Coherent point drift. In *Advances in Neural Information Processing Systems (NIPS) 19*, pages 1009–1016. 2007. 42

[52] N. R. Pal and S. K. Pal. A review on image segmentation techniques. *Pattern Recognition*, 26(9) :1277–1294, sep 1993. ISSN 00313203. 22

[53] F. Payan, S. Hahmann, and G.-P. Bonneau. Deforming surface simplification based on dynamic geometry sampling. In *Proceedings of IEEE Shape Modeling International (SMI) Conference*, Lyon, France, june 2007. 2

[54] B. Peroche and D. Bechmann. *Informatique Graphique et rendu*. Traité IC2 - Information - Commande - Communication. Hermès, Feb 2007. ISBN 978-2-7462-1515-3. URL http://lsiit-cnrs.unistra.fr/Publications/2007/11-PB07. 9

[55] D. L. Pham, C. Xu, and J. L. Prince. A survey of current methods in medical image segmentation. In *Annual Review of Biomedical Engineering*, volume 2, pages 315–338. 2000. 21

[56] G. Rosman, M. M. Bronstein, A. M. Bronstein, A. Wolf, and R. Kimmel. Group-valued regularization framework for motion segmentation of dynamic non-rigid shapes. In *Proc. Scale Space and Variational Methods (SSVM)*, 2011. 34, 35, 91

[57] M. Sattler, R. Sarlette, and R. Klein. Simple and efficient compression of animation sequences. In K. Anjyo and P. Faloutsos, editors, *Eurographics/ACM SIGGRAPH Symposium on Computer Animation (2005)*. The Eurographics Association, July 2005. 2, 33

[58] A. Shamir. A survey on mesh segmentation techniques. *Computer Graphics Forum*, 27(6) :1539–1556, 2008. 8, 22, 23, 25

[59] L. Shapira, A. Shamir, and D. Cohen-Or. Consistent mesh partitioning and skeletonisation using the shape diameter function. *The Visual Computer*, 24 :249–259, 2008. ISSN 0178-2789. 26

[60] A. Sharf, D. Alcantara, T. Lewiner, C. Greif, A. Sheffer, N. Amenta, and D. Cohen-Or. Space-time surface reconstruction using incompressible flow. *ACM Transactions on Graphics (SIGGRAPH Asia Proceedings)*, 27(5), 2008. 94

[61] A. Sharma, E. von Lavante, and R. P. Horaud. Learning shape segmentation using constrained spectral clustering and probabilistic label transfer. In *Proceedings of the Eleventh European Conference on Computer Vision*, LNCS, pages 743–756. Springer, September 2010. 95

[62] J. Shi and J. Malik. Normalized cuts and image segmentation. *PAMI*, 22(8) : 888–905, 2000. xv, 22, 69, 70, 71

[63] J. Starck and A. Hilton. Surface capture for performance based animation. *IEEE Computer Graphics and Applications*, 2007. 18, 81

[64] A. Sundaresan and R. Chellappa. Model driven segmentation of articulating humans in laplacian eigenspace. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30 :1771–1785, 2008. 36

[65] J. Tierny, J.-P. Vandeborre, and M. Daoudi. Fast and precise kinematic skeleton extraction of 3D dynamic meshes. In *IEEE International Conference on Pattern Recognition (ICPR 2008)*, 2008. 22

[66] K. Varanasi and E. Boyer. Temporally coherent segmentation of 3d reconstructions. In *In Proceedings of the Fifth International Symposium on 3D Data Processing, Visualization and Transmission*, May 2010. 32, 36, 37, 91

[67] U. Von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4) :395–416, 2007. ISSN 0960-3174. 70

[68] J. Xu, T. Yamasaki, and K. Aizawa. 3d video segmentation using point distance histograms. In *International Conference on Image Processing*, pages 701–704, 2005. 36

[69] J. Xu, T. Yamasaki, and K. Aizawa. Temporal segmentation of 3-d video by histogram-based feature vectors. *IEEE Trans. Cir. and Sys. for Video Technol.*, 19 :870–881, June 2009. 36, 38

[70] T. Yamasaki and K. Aizawa. Motion segmentation and retrieval for 3d video based on modified shape distribution. *EURASIP J. Appl. Signal Process.*, 2007 :211–211, January 2007. ISSN 1110-8657. 4, 22, 36, 38

[71] T. Yamasaki and K. Aizawa. Motion segmentation for time-varying mesh sequences based on spherical registration. *EURASIP J. Adv. Sig. Proc.*, 2009, 2009. 2, 36, 38, 95

[72] T. Yamasaki and K. Aizawa. A euclidean-geodesic shape distribution for retrieval of time-varying mesh sequences. In *Proceedings of the 2009 IEEE international conference on Multimedia and Expo*, ICME'09, pages 846–849. IEEE Press, 2009. 36, 38

[73] T. Yamasaki and K. Aizawa. Patch-based compression for time-varying meshes. In *ICIP*, pages 3433–3436, 2010. 2

[74] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition : A literature survey. *ACM Comput. Surv.*, 35 :399–458, December 2003. ISSN 0360-0300. 21

[75] E. Zuckerberger. Polyhedral surface decomposition with applications. *Computers and Graphics*, 26(5) :733–743, 2002. 22

Lexique

n-voisinage	n-neighborhood. 10
2-variété à bord	2-manifold with boundary. 9
adaptatif	adaptative. 10
arête	edge. 8
Attachement du squelette à la surface	skinning. 13
complexe	non-manifold. 9
connectivité	connectivity. 8
dense	dense. 10
distance entre les valeurs propres	eigen-gap. 70
Erreur d'affectation d'image	Frame Assignement Error. 88
Erreur d'affection de sommet	Vertex Assignement Error. 88
face	face. 8
géométrie	geometry. 9
images	frame. v, 11
irrégulier	irregular. 10
maillage	mesh. 8
maillage dynamique	dynamic mesh. 11
maillage statique	static mesh. 10
modèle	template. 62
orientabilité	orientability. 9
os	bone. 13

partitionnement par déplacement de la moyenne	mean shift clustering. 33
régulier	regular. 10
segment, cluster ou région	cluster. 23
segmentation globale	global segmentation. 27
segmentation par instances	instance segmentation. 27
segmentation par ligne de partage des eaux	watershed. 21
segmentation temporelle	temporal segmentation ou motion segmentation. 30
segmentation temporellement variable	Temporally Varying Segmentation. 29
sommets	vertex. 8
suivi de maillages	mesh tracking. 20
séquence non contrainte de maillages	unconstrained mesh sequences. 15
séquence stable de maillages	stable mesh sequence. 13
uniforme	uniform. 10
valence	valence. 9

TITLE

Séquences de maillages :
classification et méthodes de segmentation

ABSTRACT

Mesh sequences are becoming a common tool in computer graphics as can be seen in video games or medical imaging. As a consequence, mesh sequences generation's methods have been developed. Sequences can be created using a modeler or by multi-cameras acquisition. These methods can generate sequences which have different properties. Moreover, different applications using mesh sequences have been proposed such as compression or pose transfer. Most of these applications require to compute a segmentation as a pre-processing step.

This thesis deals with mesh sequence segmentation into rigid components. First, we formalize the notion of mesh sequence, and propose a nomenclature, allowing us to design the properties attached to each kind of mesh sequence, and to describe precisely what kind of sequence an algorithm requires as input.

Next, we propose a formalization of mesh sequence segmentation by providing different definitions of segmentation. We then present mesh sequences segmentation state of the art.

We then propose a first segmentation method, producing a global segmentation and taking as input a stable mesh sequence. This method is based on a region-growing process.

Next, we present two segmentation methods, both based on spectral clustering and taking as input an unconstrained mesh sequence. The former method produces a set of global segmentations, while the latter generate either a global segmentation or a time-varying segmentation.

We also propose a quantitative segmentation evaluation system.

Finally, we provide future work leads on mesh sequence segmentation.

KEYWORDS

Mesh Sequences, Classification, Segmentation

ADRR : Laboratoire d'InfoRmatique en Image et Systèmes d'information
LIRIS - UMR 5205
Université Claude Bernard Lyon 1
Bâtiment Nautibus
43, bd du 11 novembre 1918
69622 Villeurbanne cedex
ISBN : ␣␣␣␣␣␣␣␣␣␣␣␣␣␣