Modélisation 4D à partir de plusieurs caméras

Antoine Letouzey

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Informatique**

Arrêté ministérial : 7 aout 2006

Présentée par

Antoine Letouzey

Thèse dirigée par **Edmond Boyer**

préparée au sein du **Laboratoire Jean Kuntzmann**
et de l'école doctorale **Mathématiques, Sciences et Technologies de l'Information, Informatique**

Modélisation 4D à partir de plusieurs caméras

Thèse soutenue publiquement le **30 juillet 2012**,
devant le jury composé de :

M. James Crowley
Professeur Institut national polytechnique de Grenoble, Président
Mme. Marie-Odile Berger
Chargée de recherche INRIA Nancy Grand Est, Rapporteur
M. Sylvain Paris
Chercheur Adobe, Rapporteur
M. Vincent Charvillat
Professeur Institut national polytechnique de Toulouse, ENSEEIHT, Examinateur
M. Edmond Boyer
Directeur de recherche INRIA Grenoble Rhône-Alpes, Directeur de thèse

Remerciements

Je tiens tout d'abord à remercier Marie-Odile Berger, Sylvain Paris, Vincent Charvillat et James Crowley d'avoir accepté d'être dans mon jury de thèse et d'avoir fait le déplacement jusqu'à Grenoble au milieu de l'été. Merci pour l'effort réalisé ainsi que pour vos commentaires sur mon travail.

Je tiens ensuite à remercier chaleureusement Edmond Boyer pour avoir su m'encadrer pendant ces quelques années à l'INRIA. Avoir su me guider tout en me laissant un sentiment de liberté n'est pas une chose facile. Merci de m'avoir soutenu scientifiquement et moralement dans les périodes un peu rudes. Merci également de m'avoir fait confiance et donné l'opportunité de découvrir le monde de l'enseignement.

J'arrive maintenant à la partie la plus compliquée de ces remerciements : les amis. Comment être sur de n'oublier aucun d'eux tant j'ai rencontré de personnes merveilleuses durant ces années à l'INRIA ?

Commençons par mes deux années dans l'équipe PERCEPTION. I would like to thank the many very close friends I made in this team from the first day I arrived to the very end. Ramya, you are the sweetest and funniest Indian girl I had the chance to meet. Simone, your amazing tiramisu and BBQ will always been remembered. Miles, you are definitely the most British guy I ever met, keep it this way, I like your calibration trousers. Avinash and Visesh, we arrived the same day at INRIA and I'm glad to be the first out, thanks for the mutual support. Kiran, merci pour toutes les soirées, les sorties et le soutien au labo. Benjamin, j'ai particulièrement aimé partager le bureau avec toi pendant toutes ces années, un grand merci aussi pour le premier chapitre de cette thèse. Je voudrais aussi remercier la foule des amis qui n'étaient que de passage pour une courte période mais dont je me souviendrai toujours, Jane, Jamil, Cedric et Mike en particulier. Parmi les petits nouveaux de l'équipe PERCEPTION je voudrais remercier Antoine, Jan, Jordi et Kausthub, courage pour la suite ! Une pensée également pour les nouveaux amis de l'équipe MORPHEO, particulièrement Aziz, Estelle et Simon. Courage à vous aussi.

La vision par ordinateur c'est bien sympa mais il faut s'ouvrir un peu, je voudrais maintenant remercier les amis des autres équipes de l'INRIA, en particulier NANO-D et MISTIS. Merci Pierre d'avoir toujours un mot pour rire et de partager ta capacité à tout tourner en dérision quand rien ne va. Merci à Vasil, Senan et Daren d'avoir contribué à rendre ces années mémorables. Un grand merci au trio infernal Svetlana, Mael et Sergei, surtout ne changez rien vous êtes parfaitement bien tous les trois ensembles. Je tiens aussi à remercier Laurentiu et Jelmer, d'avoir rendu mes dernières années à l'INRIA aussi agréables. Afin d'oublier le moins de personnes possible je voudrais aussi remercier Diana et Valentin de l'équipe IBIS.

Je voudrais maintenant remercier les quelques personnes rencontrées à l'INRIA et qui son devenues des amis sur les sommets en neige, glace ou roche autours de Grenoble. Un énorme merci à Gaëtan, Amaël, Régis et Michel. Merci d'avoir été des compagnons de cordées, d'avoir été avec moi pour me faire découvrir la montagne en randonnée à pied, à ski, en escalade ou bien même en parapente. Merci à Amaël de m'avoir supporté en coloc ; même si tu es parti juste après t'exiler au Japon, je veux croire que les deux événements sont sans relation directe.

Je souhaite aussi remercier tout particulièrement Lamiae et Xavi pour leur soutien sans faille lors de ces dernières années. Merci d'avoir toujours été là pour moi pendant les périodes de haut et bas.

Merci également à tout ceux qui ont contribué à rendre mon passage à Grenoble aussi agréable, je pense entre autres à Amandine, Carole, Caroline, Sandra, Ahmad, Raphy, Matthieu, Josselin et Alex.

Ces remerciements ne seraient pas complets sans un mot à ma famille dont le soutien aveugle remonte toujours le moral. Merci à mes parents. Merci à Laine et Romain, maintenant que vous êtes dans la même galère je vous souhaite bien du courage. Merci à Léo, je sais que tout ça est un peu compliqué pour toi mais c'est gentil d'avoir fait semblant de comprendre ce que je racontai pendant ces 3 années.

Merci aussi à tous ceux que j'oublie.

Table des matières

Table des figures

Introduction et état de l'art

Chapitre 1

Contexte

Sommaire

Dans ce chapitre nous introduisons le contexte et les problématiques liés à cette thèse. De manière large, ce travail s'intègre dans le cadre de la capture et de l'analyse, par un système numérique, d'une scène dynamique. Ces deux composantes de nos travaux sont des problèmes assez récents. Ils couvrent un large panel de domaines et de compétences allant de l'architecture logicielle au traitement d'image en passant par la géométrie projective ou la modélisation 3D. Ici seuls les éléments ayant un intérêt direct pour les travaux de cette thèse seront présentés. Cette introduction a deux buts. Le premier est de présenter et définir les concepts généraux qui seront utilisés et développés tout au long de ce manuscrit, le second de situer précisément le contexte des travaux présentés.

1.1 Modélisation de scènes dynamiques

1.1.1 Définitions

Afin d'éviter tous malentendus ou imprécisions lors de la lecture de ce manuscrit, nous introduisons ici précisément les différents concepts récurrents de cette thèse.

1.1.1.1 Scène dynamique

Une scène est une zone délimitée de l'espace contenant des objets solides. Ces objets peuvent être classés en deux familles. Premièrement, les objets d'intérêt ou d'avant plan sont ceux qui forment le contenu de la scène. Deuxièmement, les objets de fond ou d'arrière plan sont ceux qui composent l'environnement de la scène. Faire la distinction entre ces deux groupes n'est pas toujours une tâche aisée mais c'est une des étapes fondamentale qui conduit à la compréhension de la scène, nous y reviendrons plus loin. Les différents éléments composant une scène peuvent être décrits par leur forme, leur structure en trois dimension et leur apparence. L'apparence se définit comme la couleur perçue par un oeil ou une caméra en chaque point de la surface d'un objet. Elle dépend à la fois de l'éclairage et des propriétés de reflectance propres à chaque objet.
Une manière simpliste de définir une scène dynamique est de la voir comme une succession de scènes statiques. Le principal défaut de cette vision est qu'elle fait totalement abstraction des relations très fortes qui lient deux observations successives, proches dans le temps, d'une même scène. À moins d'une grande disparité entre la fréquence d'acquisition des observations et l'amplitude des déplacements des objets, une scène dynamique reste très similaire en terme d'apparence et de forme sur une échelle de temps suffisamment courte. L'aspect dynamique d'une scène vient donc des mouvements propres de chacun des objets qui la composent. Ces mouvements peuvent être des déplacements rigides ou même des déformations des objets. Nous définissons donc une scène dynamique par les trois composantes suivantes :

- Le **sujet**, c'est-à-dire les objets d'intérêt de la scène. Il s'agit souvent d'un ou de plusieurs objets ou acteurs, en mouvement ou non.
- L'**environnement**, composé d'objets statiques qui n'entrent pas en compte dans les étapes d'analyse de la scène.
- Les **mouvements** propres au sujet, qu'il s'agisse de déplacements, de déformations, ou encore de phénomènes d'apparitions / disparitions.

Nous verrons dans la suite de ce manuscrit que suivant le type d'applications recherchées, il est possible de formuler des hypothèses sur chacune de ces trois composantes dans le but de simplifier le traitement et l'analyse d'une scène dynamique. Par exemple, il est possible de supposer que le **sujet** est un homme

seul évoluant dans un **environnement** vide et de couleur uniforme. Si en plus nous disposons d'un modèle pré-existant pour l'homme présent dans la scène, de telles hypothèses permettent de se focaliser plus facilement sur la dernière composante restante : le **mouvement**.
Au contraire, dans ces travaux, nous considérons des scènes les plus génériques possibles. Ainsi nous ne faisons pas d'hypothèses sur le nombre, la forme ou le comportement des objets d'intérêt de la scène.

1.1.1.2 Modélisation spatio-temporelle : données 4D

Nos travaux se placent dans le cadre de l'analyse automatisée de scènes dynamiques et nécessitent donc une étape de modélisation. Nous entendons par modélisation la représentation d'une scène dynamique appartenant à un domaine spatio-temporel en quatre dimension (3D + temps), en une structure numérique compréhensible et utilisable par un ordinateur. Dans le cadre de cette thèse nous nous intéressons à un système multi-caméra et définissons une structure contenant les trois informations suivantes :

- La **forme** de la scène, c'est-à-dire la structure géométrique, comprenant la surface et le volume de chacun des objets et acteurs qui prennent part à la scène.
- L'**apparence** de chacun des éléments de la scène, c'est-à-dire l'information de couleur définie comme une fonction continue à la surface des objets.
- Les **mouvements** des différents éléments de la scène. Il s'agit encore une fois d'une fonction continue à la surface des objets.

La combinaison de ces trois données doit permettre une modélisation cohérente, à la fois spatialement et temporellement, avec les observations faites par les différents capteurs.

Il s'agit évidemment d'une modélisation avec perte d'information puisqu'elle implique plusieurs niveaux de discrétisation. Une première discrétisation est faite au niveau spatial puisque nous ne disposons que d'un nombre fini de points de vue. Les observations faites sont, elles aussi, une représentation discrétisée de la scène, puisque cette dernière est limitée par la résolution des capteurs. Le deuxième niveau de discrétisation se situe sur l'axe temporel. Du fait que les caméras acquièrent un nombre fini d'images à chaque seconde, il est impossible de modéliser parfaitement des phénomènes continus. Il s'agit ici d'un modèle idéal difficilement réalisable en pratique.
Nous verrons, au chapitre 2, que dans la pratique les scènes dynamiques sont souvent représentées comme une suite de scènes statiques indépendantes. La **forme** vient de reconstructions indépendantes effectuées à chaque trame. Il en va de même pour l'**apparence** qui est recalculée indépendamment à chaque trame (ensemble des images acquises à un même instant) à partir des images

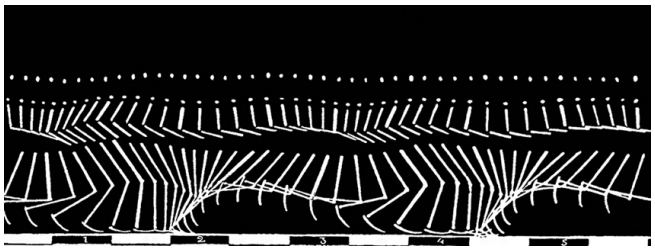
couleur. Le **mouvement** est la seule information qu'il est impossible d'obtenir à partir d'une scène statique ; cette dernière est donc tout simplement absente dans la plupart des applications de modélisation de scène dynamique. Lorsqu'elle est vraiment nécessaire, cette information est souvent obtenue par un traitement spécifique à l'application visée.

1.1.2 Diversité des modes d'acquisition

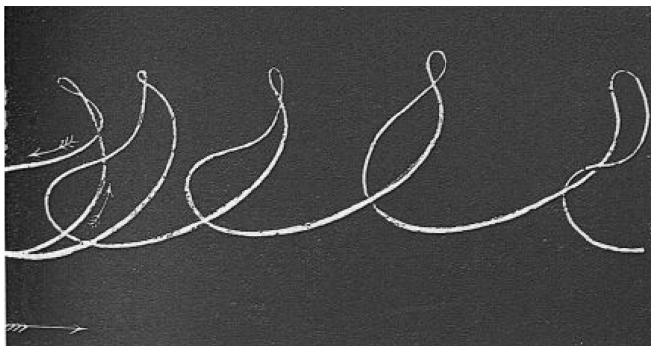
Aujourd'hui il existe un grand nombre de familles de capteurs et de modes d'acquisition permettant de modéliser tout ou partie d'une scène dynamique. Nous présentons dans cette section quelques systèmes historiques ou couramment utilisés par ordre décroissant d'intrusivité pour les acteurs.

1.1.2.1 Systèmes basés marqueurs

(a) Homme vêtu de noir



(b) Chronophotographie



(c) Trajectoire d'une aile de corneille

Figure 1.1 – La Chronophotographie. (a) Homme vêtu de noir portant des lignes et des points blancs pour l'étude chronophotographique du mouvement des points remarquable du corps. (b) Images d'un coureur. (c) Trajectoire d'une aille de corneille en vol.

Les systèmes d'acquisition basés sur l'utilisation de marqueurs visuels sont parmi les plus intrusifs puisqu'ils nécessitent de fixer des éléments sur les acteurs ou les objets d'intérêt. Il s'agit d'éléments permettant de retrouver avec

précision la position d'un nombre fixe de points clés sur les acteurs de la scène.

Historiquement, la première méthode de capture de mouvements humains par un appareil photographique a été mis au point par Etienne-Jules Marey et présenté dans son ouvrage *Le Mouvement* [Marey 84] en 1884. Ses premières études se portent sur l'observation du mouvement des ailes d'oiseaux. L'auteur est capable d'isoler le mouvement d'une paillette brillante placée sur la seconde rémige d'une aile de corneille en faisant voler l'oiseau devant un fond obscur. L'auteur ne disposait pas, à l'époque, d'appareil capable d'enregistrer une vidéo. Ainsi la trajectoire complète, visible sur la figure 1.1c, est capturée sur une seule photographie en laissant le diaphragme ouvert durant toute la séquence.

Dans la suite de ses travaux, Marey se penche sur l'étude des mouvements humains. Il confectionne un vêtement noir sur lequel sont dessinés des lignes et des points blancs (voir figure 1.1a). Comme précédemment, le sujet se déplace devant un fond noir. De cette manière seuls les éléments peints s'impriment sur la photographie. Le capteur est obstrué à intervalles réguliers afin d'obtenir une observation discrète du mouvement (voir figure 1.1b). Ceci permet d'éviter un effet de superposition des poses qui gênerait leurs interprétations.

(a) Motion Capture (b) Performance Capture

Figure 1.2 – (a) Exemple d'utilisation du système Vicon dans le cas de la modélisation de joueurs de football et (b) Exemple de capture de l'expression du visage à l'aide de marqueurs peints sur le visage d'acteurs lors du tournage du film Tintin.

Aujourd'hui, le système existant le plus connu dans cette catégorie est le système proposé par Vicon®[1]. Son fonctionnement n'est pas si éloigné de celui du système de Marey. Il se base sur l'utilisation de marqueurs réfléchissant la

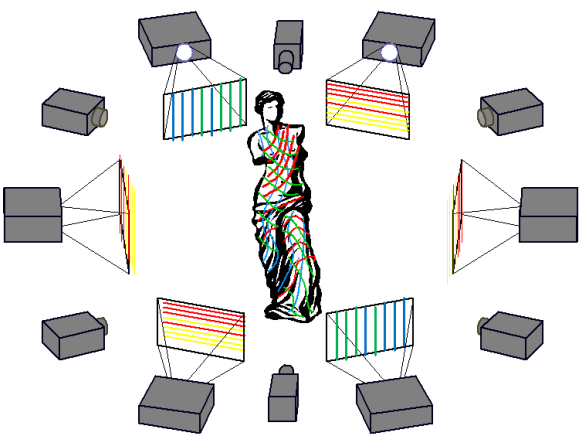
[1] http://www.vicon.com

lumière infra-rouge couplés avec un système multi-caméra. Dès qu'un marqueur est visible dans plusieurs caméras simultanément sa position peut être obtenue avec précision par triangulation. La figure 1.2a montre un exemple d'utilisation du système Vicon dans le cadre de la modélisation de deux joueurs de football. L'information obtenue consiste en un nombre de trajectoires de points choisis. Typiquement des points de jointure entre des parties rigides d'un objet ou d'un acteur (coudes, genoux, mains par exemple). Ce type d'information très discrétisée mais aussi très ciblée est idéale pour l'animation de squelettes. C'est pourquoi ces systèmes sont principalement utilisés par l'industrie du cinéma ou du jeu vidéo.

Un dérivé de cette technique est de plus en plus utilisée non pas pour capturer des mouvements (*mocap* pour *motion capture*) mais plutôt la performance des acteurs (*perfcap* pour *performance capture*). Le principe est sensiblement identique à la différence près que les marqueurs sont peints sur le visage des acteurs. Ces derniers sont ensuite filmés par une caméra attachée à un casque. Cette technique, illustrée par la figure 1.2b, a par exemple été utilisée dans les films *Avatar* ou plus récemment *Les Aventures de Tintin : Le Secret de La Licorne*. Elle permet de transposer de manière très fidèle les expressions d'un acteur sur un modèle 3D animé.

Dans la famille des systèmes utilisant des marqueurs nous pouvons aussi noter l'existence de système utilisant des marqueurs actifs. Contrairement aux méthodes décrites précédemment, la position des marqueurs n'est plus calculée par rapport à leur visibilité dans des images, mais par rapport à des enregistrements faits par les marqueurs eux-mêmes. Par exemple, le système Xsens®[2] combine des centrales inertielles et des gyroscopes pour estimer la trajectoire en trois dimensions d'un ensemble de points.

Les systèmes présentés dans cette section ont comme avantage de fournir une information de grande précision. Celle-ci concerne à la fois la localisation ponctuelle et la trajectoire en trois dimensions de chaque point suivi. Néanmoins, il s'agit d'une information très discrétisée puisqu'elle n'est disponible que pour un faible nombre de points de la scène, une vingtaine par exemple pour un acteur humain. De plus, il est nécessaire de faire porter aux acteurs une combinaison spécifique, parfois assez contraignante.



(a) Lumière structurée multi-capteur

(b) Kinect

Figure 1.3 – (a) Installation avec plusieurs projecteurs diffusant une lumière structurée et caméras couleur (image de Ryo Furukawa) et (b) Le motif de la caméra Kinect sur une scène simple (image infra-rouge).

1.1.2.2 Systèmes utilisant des capteurs actifs

Plusieurs méthodes existent ne nécessitant pas de contraindre l'habillement des acteurs, les laissant ainsi libres de leurs mouvements. Parmi celles-ci nous nous concentrons dans cette section aux méthodes utilisant des capteurs actifs. Un capteur actif est un capteur (souvent une caméra) qui ne se contente pas de recevoir de l'information. Il émet un signal et l'interprétation de son retour permet d'obtenir des informations sur la forme de la scène.

Les systèmes basés sur la méthode appelée "lumière structurée" nécessitent l'utilisation conjointe de rétro-projecteurs et de caméras couleur. Cette méthode consiste à projeter un motif connu sur la scène et d'en analyser les déformations dans les images pour en déduire une information 3D sur la scène. La figure 1.3a issue des travaux de [Furukawa 10a] montre une installation mélangeant six couples rétro-projecteur–caméra. En combinant les informations issues de chaque capteur, la structure complète de la scène est calculée.

L'utilisation d'une lumière dans le spectre visible a deux principaux désavantages. Premièrement, cela empêche de retrouver une information d'apparence fiable pour les objets de la scène. Deuxièmement, cela peut entraîner une gêne pour les acteurs. De récents capteurs utilisent le même principe mais dans un spectre infra-rouge. Le plus connu d'entre eux est le capteur Kinect de Microsoft [Microsoft 10]. Il combine un projecteur infra-rouge à deux caméras, l'une couleur, l'autre infra-rouge. Le motif projeté, dont un exemple est montré

[2] http://www.xsens.com

dans la figure 1.3b, est analysé pour fournir une carte de profondeur de la scène.

Les caméras type temps-de-vol (*Time-of-Flight*) agissent sur le même principe que le sonar à la différence près que l'onde envoyée est lumineuse. La profondeur de la scène par rapport au capteur est ensuite calculée en fonction du temps que met ce rayon pour revenir vers la caméra.

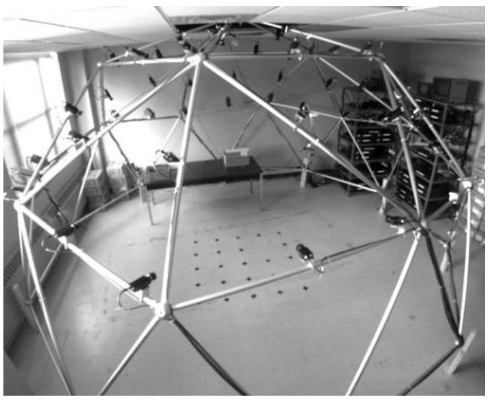
Toutes ces méthodes ont l'avantage de fournir une information de bonne qualité, et souvent à faible coût, sur la structure de la scène. Certains désavantages sont tout de même à noter. Tout d'abord, les informations sur la structure de la scène sont obtenues sous la forme de nuage de cartes de profondeurs. Ces dernières étant propres à chaque capteur de l'installation, il est nécessaire de faire appel à une étape de fusion des données. Ensuite il est nécessaire que l'éclairage de la scène soit bien contrôlé. En particulier l'utilisation de capteur infra-rouge interdit toutes applications en environnement extérieur. Il est aussi important de noter que ces méthodes sont inapplicables si la scène contient des éléments fortement spéculaires.

1.1.2.3 Systèmes utilisant des capteurs passifs

Les systèmes qui font usage de capteurs passifs sont les plus adaptatifs. Ils n'imposent pas de contraintes fortes sur l'environnement d'acquisition. Typiquement il s'agit de systèmes composés de plusieurs caméras couleurs disposées autour de la scène à capturer. Suivant le nombre de capteurs disponibles, plusieurs terminologies sont utilisées.

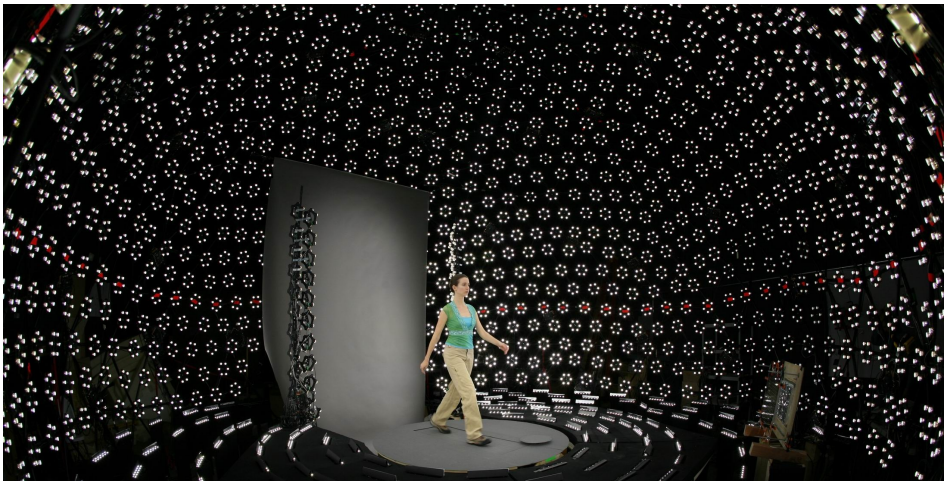
Un système stéréoscopique comprend deux caméras disposées côte-à-côte. C'est un modèle dérivé de la vision humaine. La mise en correspondance des objets entre deux images permet d'en estimer la profondeur. Ce problème n'est pas trivial et n'est souvent pas applicable en temps-réel dans le cas de scène en mouvement.

Un système multi-vue stéréo est la généralisation du cas précédent à un nombre N de caméras. Si le but est d'obtenir un modèle statique précis, il est intéressant d'utiliser des algorithmes de reconstructions 3D basés sur l'information photométrique. Dans le cas de modélisation de scènes dynamiques il est plus pertinent de faire appel à des méthodes, certes moins précises, mais plus rapides. Typiquement les méthodes utilisant les enveloppes visuelles permettent d'obtenir en temps réel une information sur la structure de la scène. Plusieurs systèmes multi-caméra ont été développés durant les quinze dernières années. Le système *Virtualized Reality* illustré dans la figure 1.4a et proposé par Kanade *et al.* dans [Kanade 97] est le premier d'entre eux.



(a) Virtualized Reality

(b) Capture d'expressions faciales



(c) Light Stage 6

Figure 1.4 – (a) Premier système multi-caméra présenté par Kande *et al.* [Kanade 97]. (b) Système multi-caméra utilisé pour la réalisation du jeu vidéo *L.A. Noir*. (c) Light Stage 6

Plus récemment, le système *Light Stage 6* [Vlasic 09] est composé de huit caméras hautes fréquences et d'un ensemble de 1200 sources lumineuses. Cette installation permet de capturer la scène sous différentes conditions d'illumination et d'en déduire la structure. La figure 1.4c montre une vue intérieure de ce système.

Aujourd'hui de plus en plus d'applications font usage de systèmes composés de plusieurs caméras couleur. Nous pouvons citer par exemple le jeu vidéo *L.A. Noir* commercialisé en mai 2011. Toutes les expressions des personnages du jeu ont été finement modélisées en utilisant le système présenté dans la figure 1.4b. Ce travail a permis d'atteindre un degré de réalisme difficilement réalisable par des techniques conventionnelles, typiquement des expression de

synthèses définies par des artistes. Les mouvements et interactions des acteurs ont, quant à eux, été modélisés par un système basé marqueurs tel que celui de la figure 1.2a. Il est intéressant de noter que ces deux approches sont plus complémentaires que concurrentes et interviennent à différents niveaux lors de la modélisation d'une scène dynamique. L'industrie du cinéma commence à se tourner vers des systèmes d'acquisition 3D qui n'imposent pas le port d'une combinaison recouverte de marqueurs et permettent aux acteurs de revêtir leurs vrais habits ou costumes. Le film *Le destin de Rome* en est un exemple récent. Entièrement tourné en images de synthèse, toutes les performances d'acteurs ont été acquises à l'aide d'un système multi-caméra qui permet de modéliser directement les acteurs en mouvements et en costumes. L'avantage principal, si le but est de faire du rendu réaliste, est de pouvoir s'affranchir d'une étape intermédiaire qui consiste à transférer les mouvements capturés à un modèle 3D créé par un artiste.

Dans le cadre de cette thèse, nous nous sommes concentré sur des données provenant de ce type de système. Nous détaillerons dans la section 1.3 le système GrImage utilisé dans nos travaux.

1.2 Traitement de données 4D

Nous allons présenter dans cette partie plusieurs cadres d'utilisations possibles des systèmes de modélisation de scènes dynamiques.

1.2.1 Industrie du cinéma et du jeu vidéo

Comme nous avons vu à plusieurs reprises, de nombreuses applications sont directement orientées vers le monde du cinéma ou du jeu vidéo. Dans le souci de proposer toujours plus de réalisme lors de la création de films en images de synthèse, les outils de captures de mouvements ont fait de grands progrès lors des quinze à vingt dernières années. Leur utilisation se limitait, dans un premier temps, à la modélisation de la trajectoire d'un ensemble restreint de points d'intérêt. Aujourd'hui, il est possible de restituer avec une grande fidélité quasiment n'importe quelle expression du visage d'un acteur et de la transférer sur un modèle graphique. Le film *Avatar* fut le premier où tous les plans sont directement générés en 3D en transférant la performance des acteurs sur des modèles de synthèses. Par performance nous entendons deux aspects distincts. Dans un premier temps, les mouvements des acteurs dans la scène ont été capturés à l'aide d'un système basé marqueurs de type Vicon. Ensuite, les expressions de visages ont été modélisées à l'aide d'un système tel que celui présenté dans la figure 1.2b.

De la même manière l'industrie du jeu vidéo fait usage des outils de capture de mouvement afin d'améliorer le réalisme des situations dans lesquelles elle plonge le joueur. De nos jours les moyens mis en œuvre pour la réalisation d'un jeu vidéo sont souvent comparables à ceux d'un film. Dans le jeu *L.A. Noir* par exemple, les mouvements (du corps et du visage) des personnages ont été transposés directement à partir de ceux de vrais acteurs. Cela a nécessité l'utilisation conjointe d'un système basé marqueurs (voir figure 1.2a) et d'un système multi-caméra passif (voir figure 1.4b).

1.2.2 Vidéo en 3D

Depuis deux à trois ans maintenant, beaucoup de films produits sont qualifiés de film 3D. Bien souvent cette affirmation est fausse. Dans le meilleur des cas le spectateur se voit projeter deux images stéréoscopiques. Grâce à l'utilisation d'une paire de lunettes spécifiques, l'image reconstituée par notre cerveau est accompagnée d'une impression de profondeur.

Par vidéo 3D nous entendons que la totalité de la scène est modélisée et donc que le spectateur peu potentiellement choisir n'importe quel point de vue, y compris des points de vue qui ne correspondent pas à ceux des capteurs utilisés lors de l'acquisition [Carranza 03]. Dans le paragraphe précédent nous nous sommes attardés sur la notion de transfert de mouvement d'une scène réelle à sa représentation graphique. Ici, il s'agit de directement reconstruire en trois dimension la scène telle qu'elle est et telle qu'elle évolue dans le temps. Typiquement ce type d'applications permet de reproduire des effets de type *Bullet-time* du film *Matrix* sans avoir recours à un très grand nombre de caméras.

Il s'agit d'une approche assez récente et pour laquelle beaucoup de travaux restent à faire. Par exemple il n'existe pas de standard de stockage pour de telles vidéos. Elles sont donc enregistrées comme une suite de scène 3D statiques. La création de meilleurs modèles spatio-temporels pourrait permettre de limiter la taille de telles données en limitant la redondance des informations géométriques et de texture.

1.2.3 Interface homme-machine

Dans certains cas, les interactions entre un homme et une machine ne peuvent pas passer par les canaux usuels (clavier, souris, écran tactile, ...). Par exemple dans le cadre d'applications robotiques humanoïdes, il est important de rendre les interactions entre le robot et les humains les plus naturelles possibles. Il existe des applications d'interface homme-machine basées sur la compréhension des gestes de l'utilisateur. Ces gestes sont captés par un système

composé d'une ou plusieurs caméras puis interprétés suivant le contexte par le robot [Triesch 98].

À la fin de l'année 2010, Microsoft a commencé à commercialiser le capteur Kinect. Cet outil, présenté dans la section 1.1.2.2, est destiné à remplacer les manette de jeu pour la console XBox 360. Il permet de modéliser en temps réel l'environnement où se trouvent les joueurs et de détecter en temps réel la pose de ces derniers.

1.2.4 Réalité augmentée et environnements virtuels

La connaissance de la structure géométrique de la scène et de la position des différents points de vue permet de proposer des applications de réalité augmentée. Cela implique l'insertion dans des points de vue réels de données de synthèse dont la position et le rendu sont cohérents avec la scène (occultations, éclairage, ...). Typiquement ce type d'applications ajoute des informations contextuelles à une scène. Des travaux [Bichlmeier 09] permettent ainsi de fournir des informations pertinentes et localisées à un chirurgien en cours d'opérations. Il est intéressant de noter la récente apparition d'applications pour téléphones portables qui utilisent des informations venant des caméras intégrées. Par exemple, l'application *Peak A.R.*[3] permet de mettre en surimpression des informations sur les sommets des alpes visibles sur une image.

Proche de la réalité augmentée les applications d'environnements virtuels plongent l'utilisateur dans un monde hybride, mélangeant éléments réels et virtuels, ou bien totalement numérique. Un exemple est présenté dans la section 1.3.1.1.

1.2.5 Reconnaissance d'actions et surveillance

De nombreuses applications de reconnaissance d'actions existent. Elles se basent principalement sur l'étude de données en deux dimensions. Néanmoins, certains travaux, [Weinland 06] par exemple, ont permis de mettre en avant l'utilité de la modélisation spatio-temporelle pour la reconnaissance de gestes.

Si la structure de la scène est connue alors il est possible de détecter des événements imprévus ou interdits. Par exemple, les travaux de Ladikos *et al.* [Ladikos 08] proposent de modéliser une salle d'opération afin de détecter des comportements potentiellement à risque des intervenants.

[3] http://peakar.salzburgresearch.at

1.3 Contexte des travaux : la salle GrImage

1.3.1 Une plateforme de recherche multi-usage

La plateforme GrImage[4] (pour Grid et Image) est un système d'expérimentation multi-caméra développé à l'INRIA de Grenoble. Elle est le fruit du travail de plusieurs équipes de recherche spécialisées dans des domaines de vision par ordinateur, de calcul parallèle et de graphisme. Le système se compose d'un ensemble de quantité variable de caméras synchronisées reliées à une grappe de PC. La principale différence avec les autres systèmes de modélisation de scène dynamique est son aspect temps-réel. En effet, la plateforme GrImage est utilisée dans deux cadres différents :

- Utilisation en ligne. Le but est de permettre à un ou plusieurs utilisateurs d'interagir physiquement avec un monde virtuel.
- Utilisation hors ligne. La plateforme sert de système d'acquisition de séquences de données 4D.

1.3.1.1 Réalité virtuelle et interactions temps réel

Dans le cas d'une utilisation en temps-réel, le système est limité à huit caméras. Il permet de reconstruire la structure et l'apparence de la scène à une fréquence de 30Hz avec une latence inférieure à 60 millisecondes. En utilisant un casque de réalité virtuelle, il est possible d'immerger complètement un utilisateur dans un monde virtuel avec lequel son avatar 3D peut interagir. Le projet VGate [Petit 09] illustré par la figure 1.5 est un exemple d'utilisation du système temps-réel. Aujourd'hui les interactions entre l'utilisateur et le monde virtuel sont gérées par le *framework* SOFA[5]. Dans la mesure où le système produit à chaque instant un nouveau modèle de la scène indépendant des reconstructions précédentes, il n'est pas possible d'obtenir directement une information de vitesse. Les collisions sont donc basées sur une notion d'intersection entre des boîtes englobantes. Cette solution présente des limites au niveau du réalisme des réactions entre objets qui peuvent sembler incohérentes par moment. Pour plus d'informations au sujet de la plateforme temps-réel, nous renvoyons à la lecture de la thèse de Benjamin Petit [Petit 11a].

1.3.1.2 Système d'acquisition de séquences 4D

La plateforme GrImage sert aussi de système d'acquisition et de modélisation de scènes dynamiques. Les données 4D acquises sont directement liées aux travaux de recherche menés par les équipes en charge de la plateforme. Suivant le sujet d'étude, les données peuvent avoir des caractéristiques différentes. Par

[4] http://www.grimage.inrialpes.fr
[5] http://www.sofa-framework.org/

Figure 1.5 – Le projet *Virtualization Gate* permet de plonger un utilisateur et son avatar 3D dans un monde virtuel. Ici le modèle de l'utilisateur est reconstruit et intégré en temps-réel dans le monde virtuel. Les interactions sont ensuite calculées avec les objets ce qui permet ici à l'utilisateur de renverser le vase avec son pied. La vue en bas à droite est une vue déportée du monde virtuel. Le casque muni de lunettes-écran permet à l'utilisateur d'avoir un point de vue cohérent avec sa position dans le monde réel.

exemple des travaux en reconnaissances d'actions auront besoin d'une grande redondance dans les données pour effectuer un apprentissage. Au contraire, des travaux plus orientés sur la reconstruction précise de modèle auront plutôt besoin d'une grande quantité d'informations à chaque trame et donc nécessiteront plus de caméras. Ces différents cas sont discutés plus en détail dans la section 2 de ce manuscrit.

1.3.2 Installation

De par la multitude de ses utilisations, la plateforme GrImage a été conçue dès le départ avec un grand souci de modularité. Ainsi chaque utilisateur peut facilement mettre en place son propre système multi-caméra qui répondra au mieux à ses besoins. Il pourra choisir le nombre de caméras dont il a besoin. Un nombre plus élevé de caméras permet d'obtenir des informations plus denses sur la scène ainsi que des maillages de meilleure qualité (voir section 1.3.3). Il faut néanmoins garder à l'esprit que cela augmente aussi la complexité du traitement des données. L'utilisateur peut aussi décider de la disposition des caméras autour de la scène. Généralement nous répartissons les caméras de manière régulière sur un demi-dôme autour de la scène afin de couvrir chaque zone de manière homogène. Pour certaines applications, il

peut être intéressant de placer plus de caméras sur zone particulière dans le but d'obtenir une meilleur information photométrique, pour texturer un visage humain par exemple.

En pratique la plateforme GrImage est installée dans une salle dont le sol et les murs sont recouverts d'un tissu vert qui facilite l'étape de soustraction de fond (séparation du fond de la forme de la scène). Les caméras sont disposées autour de la scène grâce à des supports posés au sol ou accrochés au plafond. Le volume utile de la scène est défini par l'intersection de toutes les pyramides de vision des caméras. C'est-à-dire que pour être reconstruit, un objet de la scène doit se trouver dans ce volume. La figure 1.5 présente un exemple d'installation et la figure 1.7 illustre le principe de pyramide de vision.

1.3.3 Reconstruction statique de la scène

Cette partie présente de manière succincte le processus qui conduit à l'obtention d'un modèle géométrique de la scène pour chaque trame capturée par le système multi-caméra.

1.3.3.1 Étalonnage du système

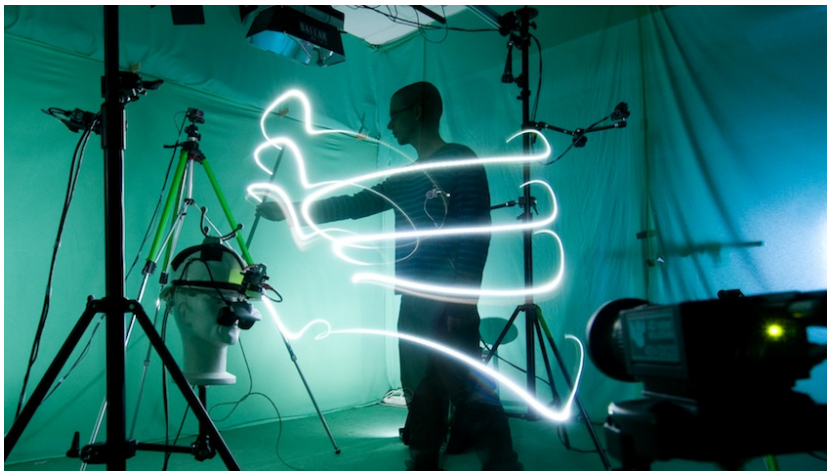


Figure 1.6 – Processus d'étalonnage de la plateforme GrImage.

La première étape consiste à connaître de manière précise les paramètres intrinsèques et extrinsèques de chaque caméra. Les paramètres extrinsèques correspondent à la disposition relative des caméras dans l'espace autour de la scène. Les paramètres intrinsèques sont quand à eux propres à chaque capteur (le point principal et la longueur focale). Cette étape est appelée l'étalonnage. Elle se base principalement sur la mise en correspondance entre points de l'espace 3D et leur projection en 2D dans les différentes images.

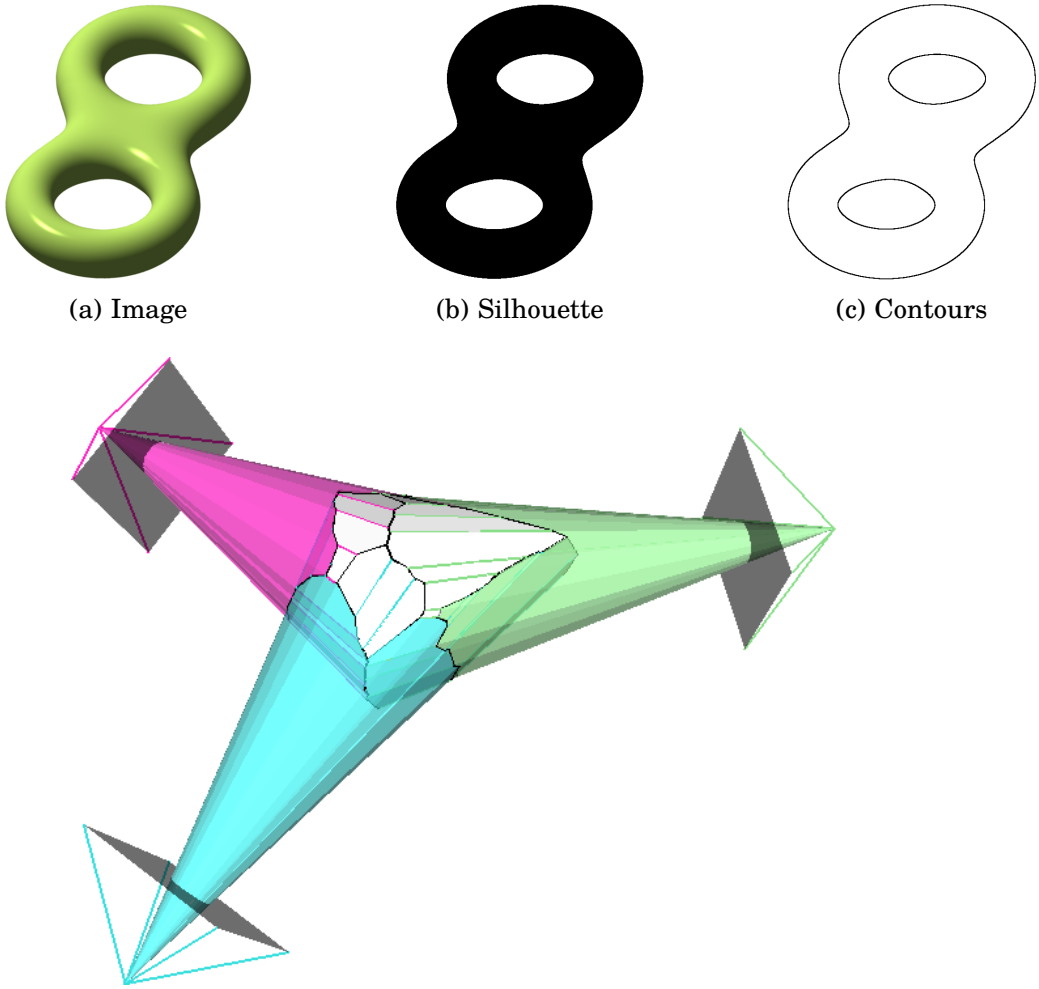
Certaines méthodes existent qui font l'usage des silhouettes [Sinha 04, Boyer 06]. Bien qu'il s'agisse d'une information disponible, dans un souci de grande précision il a été choisi d'avoir recours à une méthode propre à la plateforme. Pour cela, nous utilisons une approche basée sur [Zhang 04] et utilisant une baguette sur laquelle sont disposées, avec des écartements connus, quatre diodes lumineuses (voir figure 1.6). Une succession d'images sont acquises pendant que la baguette est déplacée dans la scène en essayant de couvrir au mieux tout l'espace d'acquisition. Les positions et trajectoires des différents points lumineux de la baguette sont ensuite détectées dans les images. Pour finir un processus itératif d'ajustement de faisceaux permet de retrouver l'étalonnage du système en minimisant l'erreur de reprojection dans les images des positions 3D estimées de chaque point lumineux de la baguette.

1.3.3.2 Extraction de silhouettes

La forme des objets et acteurs de la scène est directement inférée à partir des silhouettes détectées dans les images. Ces dernières sont obtenues par une méthode de soustraction de fond. L'environnement de la scène est supposé statique, nous pouvons ainsi procéder à son apprentissage. Pour cela nous enregistrons quelques trames avec une scène vide pour construire un modèle par pixel du fond. Basé sur le modèle de [Horprasert 99], nous utilisons une distance basée sur un écart chromatique et d'intensité lumineuse qui prend en compte la détection des ombres. Sa mise en œuvre est grandement facilitée par la mise en place d'un éclairage de bonne qualité et l'utilisation d'un fond de couleur unie et différente des couleurs de la scène. Elle n'en reste pas moins assez générique pour s'adapter à un environnement quelconque. Cette étape influe directement sur la qualité du modèle reconstruit. Si une partie de l'image est associée au fond alors qu'elle correspond au premier plan alors un artefact apparaît dans le modèle reconstruit qui s'apparente à un trou (voir figure 1.8).

1.3.3.3 Exact Polyhedral Visual Hulls

L'algorithme utilisé par notre système vise à reconstruire l'enveloppe visuelle de la scène. Cette méthode a été choisie pour sa robustesse et sa rapidité. À chaque nouvelle trame le modèle est reconstruit indépendamment des modèles précédents. Ainsi nous évitons le phénomène d'accumulation d'erreur qui détériore la qualité des reconstructions au fil des observations. La rapidité d'exécution est indispensable dans le cadre d'une utilisation temps-réel. L'algorithme utilisé nous permet de reconstruire la scène avec une fréquence de 25Hz environ. L'enveloppe visuelle [Laurentini 94, Lazebnik 01] est une enveloppe convexe qui correspond au volume maximum occupé par les objets de la scène et dont la projection dans les différentes images coïncide avec l'information de silhouette. À partir des silhouettes, une image binaire est créée séparant le fond de la forme.



(a) Image (b) Silhouette (c) Contours

(d) Intersection

FIGURE 1.7 – (a) Image couleur, (b) silhouette associée et (c) le contour extérieur et les deux contours intérieurs détectés. (d) Intersection des cônes de visibilité provenant de trois caméras.

Les contours extraits sur cette image permettent de transformer une silhouette en un ensemble de contours intérieurs et extérieurs qui vont servir à construire des cônes de visibilité pour chaque caméra. C'est l'intersection de ces cônes dans l'espace 3D qui correspond au modèle de la scène. La figure 1.7, présente ces différentes étapes.

L'algorithme utilisé dans la plateforme GrImage est appelé EPVH [Franco 03] (*Exact Polyhedron Visual Hull*). Il présente l'avantage de fournir une reconstruction exacte dans le sens où la reprojection du modèle dans les images coïncide avec les silhouettes ce qui permet d'utiliser directement les silhouettes comme masque pour trouver l'information de texture dans les images en vue de leur reprojection sur le maillage. Cet algorithme est de plus fortement

parallélisable ce qui a permis son implémentation dans un cadre temps-réel.

1.4 Problématique et contributions

1.4.1 Comment augmenter l'information disponible ?

En se référant à la définition d'un modèle de scène dynamique proposé dans la section 1.1.1, les informations fournies par la plateforme GrImage à chaque trame correspondent à la forme et à l'apparence de la scène. Il manque donc l'information de mouvement pour pouvoir affirmer avoir un modèle complet. Cette information est pourtant indispensable, surtout dans le cadre d'interactions entre acteurs réels et monde virtuel. Si les impressions d'interactions d'un utilisateur sur un monde virtuel ne lui semblent pas réalistes, alors le sentiment d'immersion est immédiatement brisé. En effet sans information pertinente sur la vitesse de déplacement instantanée de chaque éléments de la scène, il est impossible de fournir une expérience interactive de qualité à l'utilisateur. Comment déduire la réaction que doit avoir un objet du monde virtuel après impact si aucune information fiable n'est disponible sur la nature du mouvement qui a créé cette interaction ?

En partant de ce constat nous avons proposé une méthode permettant d'enrichir le modèle de la scène en y incorporant une information dense de mouvement. C'est-à-dire que chaque point de la surface reconstruit est associé à un vecteur de déplacement instantané. Cette information, appelée le flot de scène, n'est pas limitée à une utilisation dans le cadre d'application de réalité augmentée. En effet, de la même manière que le flot optique en 2D, il s'agit d'une brique de base qui permet de construire des applications de plus haut niveau. Cette information s'avère très utile pour des travaux de suivi d'objets, de reconnaissances d'action ou de détection de parties rigides entre autres. Les travaux menés dans ce sens sont présentés dans le chapitre 3.

1.4.2 Comment améliorer la qualité des modèles ?

À chaque nouvelle trame, le processus de reconstruction du modèle de la scène reprend à zéro en faisant abstraction des résultats obtenus aux instants précédents. Or nous avons vu dans la section 1.1.1 que la structure d'une scène dynamique doit être cohérente spatialement et temporellement. Jusqu'à présent, cette notion de cohérence temporelle n'a pas du tout été prise en compte et deux reconstructions successives peuvent avoir des structures très différentes. La figure 1.8 présente quelques artefacts propres aux modèles reconstruits par un algorithme basé sur le principe d'enveloppes visuelles. Certains d'entre eux (1.8a et 1.8b) sont dûs à des erreurs dans l'estimation des silhouettes. Dans

le premier cas, l'ombre projetée sur le sol par le ballon est considérée dans les images comme appartenant à la forme et non plus au fond (voir figure 1.9a). Cette mauvaise appréciation se traduit par l'apparition, sur l'intervalle d'une ou deux trames, d'un blob dans le modèle 3D. C'est l'effet inverse qui se produit dans le second cas, il suffit que dans une des images l'arrière de la tête soit associé par erreur au fond pour que le modèle se trouve percé d'un trou (voir figure 1.9b). L'artefact présenté sur la figure 1.8c provient du fait que le chien sorte physiquement du volume maximum de reconstruction, c'est-à-dire qu'il sort du champ de vue d'au moins une des caméras (voir figure 1.9c). Le champ de vue limité des caméras restreint l'espace utile qui est effectivement reconstruit. Tout objet se situant hors de ce volume se retrouve coupé ou supprimé. Le dernier type d'artefact, mis en évidence par la figure 1.8d, vient de la résolution spatiale et temporelle limitée des capteurs des caméras. Les objets fins se déplaçant rapidement apparaissent flous dans les images et sont donc difficilement détectables lors de la construction des silhouettes (voir figure 1.9d). Toutes ces erreurs de reconstructions sont ponctuelles et n'apparaissent pas sur l'intégralité de la séquence.

Pour répondre à ce soucis d'incohérence temporelle nous avons proposé une méthode qui permet d'accumuler l'information géométrique du modèle sur une séquence complète. Ces travaux permettent d'inférer la structure de la scène la plus complète possible à partir d'observations séquentielles. Basé sur une hypothèse que les objets de la scène ont une topologie fixe (pas de division, pas de liquide), nous augmentons au fil de la séquence la quantité d'information topologique contenue dans le modèle. Cette méthode, présentée au chapitre 4, se base sur des outils de traitement de maillages de la communauté.

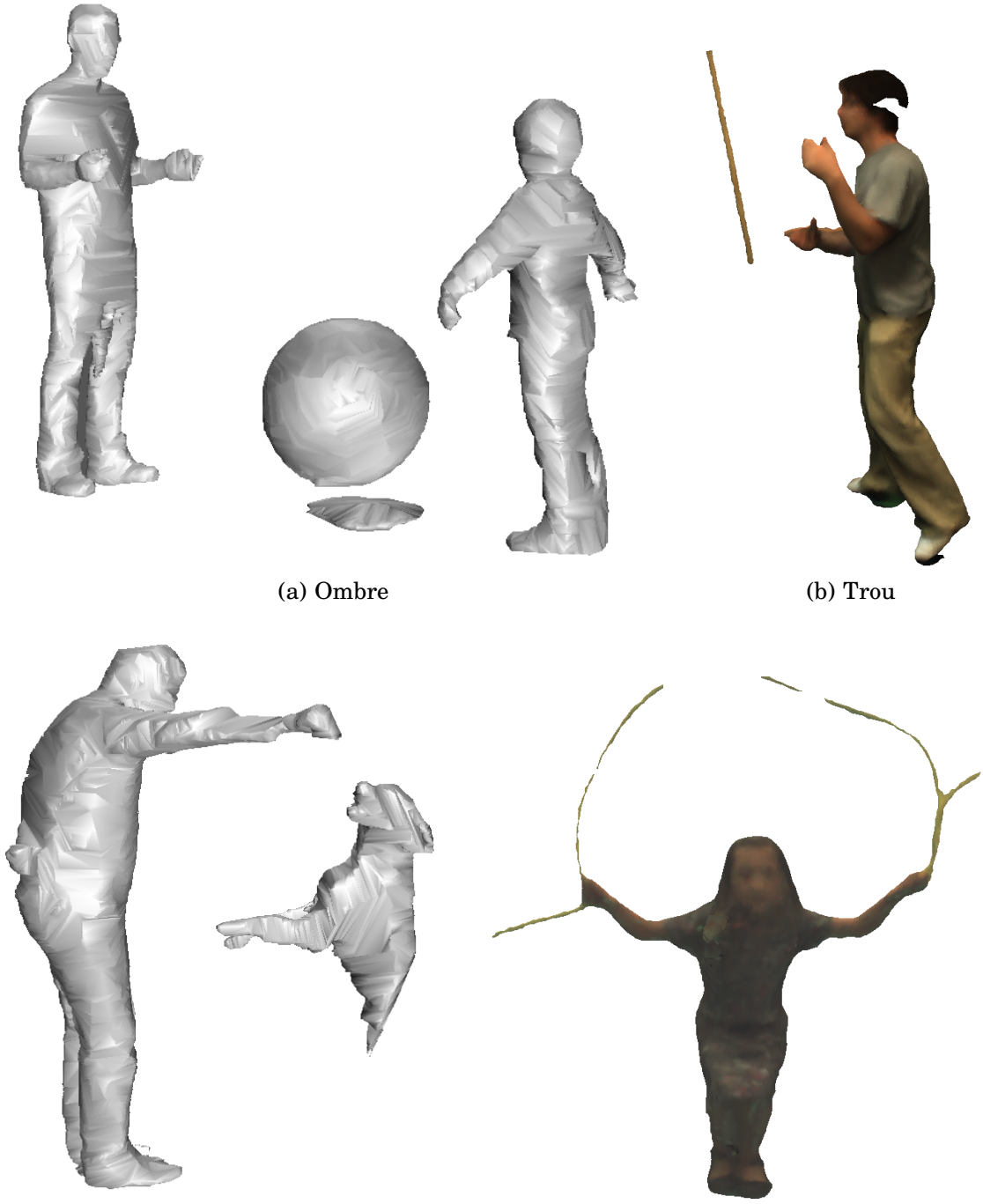
1.5 Plan du manuscrit

La suite de ce manuscrit est organisée comme ceci :

- Le chapitre 2 propose un aperçu des principales bases de données 4D disponibles pour la communauté aujourd'hui. Leurs caractéristiques, avantages et inconvénients y sont discutés.
- Le chapitre 3 présente une méthode nouvelle d'estimation dense du flot de scène. Le résultat obtenu permet d'enrichir les données 4D en y incorporant une information instantanée de mouvement. La méthode proposée se montre très souple et nous proposons et évaluons son utilisation dans le cadre d'un système multi-caméra traditionnel ainsi que dans le cas d'un système hybride incluant des capteurs de profondeur.
- Le chapitre 4 présente nos travaux menés dans le cadre de la construction d'un modèle de surface cohérent temporellement à partir d'une séquence de

maillages distincts. La méthode proposée permet de retrouver la topologie correcte d'une scène même si celle ci n'est jamais visible entièrement dans les observations.

- Enfin, le chapitre 5 propose de conclure ce manuscrit en prenant un peu de recul sur nos travaux en en proposant plusieurs pistes pour des perspectives futures.



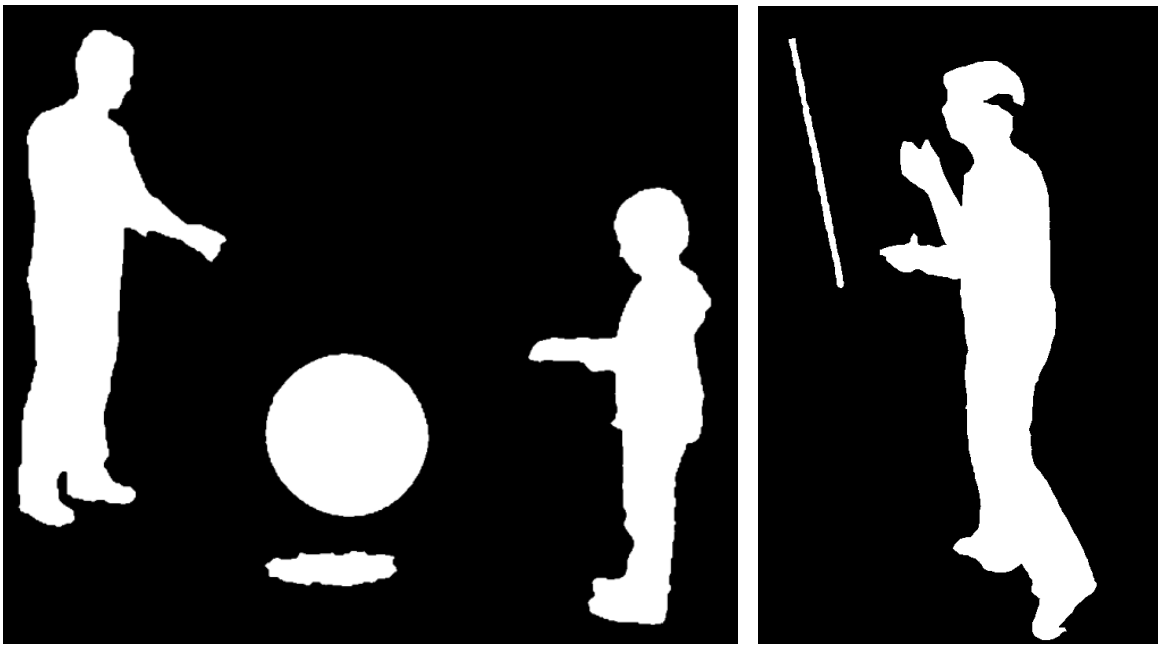
(a) Ombre

(b) Trou

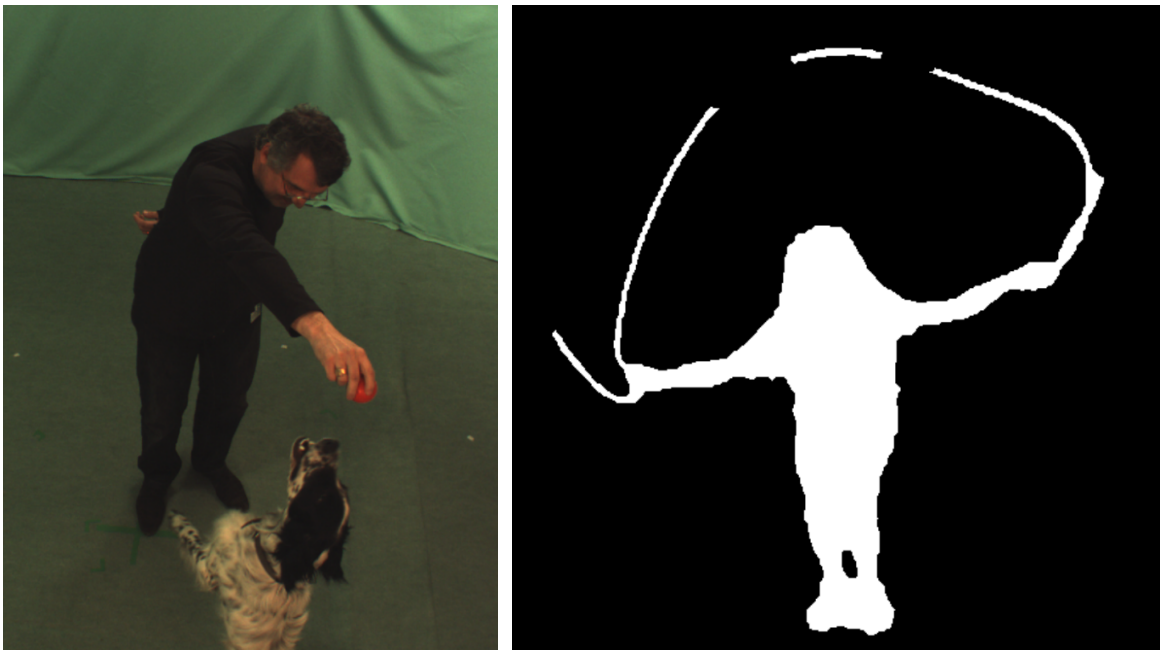
(c) Limite du volume de reconstruction

(d) Mouvement rapide

Figure 1.8 – Artefacts de reconstructions : effets. (a) L'ombre projetée au sol du ballon apparaît sur les silhouettes et forme un blob incohérent dans le modèle. (b) L'arrière de la tête est associée au fond sur une des images au moins, ce qui conduit à la création d'un trou dans le modèle. (c) Le chien sort des limites du volume de reconstruction et disparaît du modèle. (d) La finesse et le mouvement rapide de la corde à sauter la rend difficilement détectable dans les silhouettes, le modèle s'en retrouve morcelé.



(a) Ombre (b) Trou



(c) Limite du volume de reconstruction (d) Mouvement rapide

Figure 1.9 – Artefacts de reconstructions : causes. (a) L'ombre projetée au sol du ballon. (b) L'arrière de la tête est associé au fond. (c) Le chien sort du champ de vue. (d) La finesse et le mouvement rapide de la corde à sauter la rend difficilement détectable dans les silhouettes.

Chapitre 2

Bases de données 4D

Sommaire

Les caractéristiques des données obtenues lors d'un processus d'acquisition d'une séquence 4D sont définies par de nombreux facteurs. Ceux-ci vont du contexte de recherche qui a conduit à ces acquisitions à la qualité des équipements par exemple. Il est important d'être conscient de ces facteurs et d'en comprendre les enjeux. Dans ce chapitre nous présentons les caractéristiques principales de quelques une des bases de données 4D les plus connues ainsi que des éléments de comparaisons objectifs que chacun doit avoir à l'esprit lorsqu'il s'agit d'évaluer des travaux dans les meilleures conditions. C'est-à-dire de manière non pas à obtenir forcement le meilleur résultat, mais plutôt à tester les limites d'une méthode. Cette section, bien qu'elle présente des données acquises lors de cette thèse, se veut la plus objective possible et n'est pas orientée vers un traitement particulier des données 4D. Les différentes bases de données sont présentées dans leur contexte respectif qui permet de comprendre leur nature. Leur évaluation porte sur la qualité intrinsèque des données proposées suivant les critères définis dans la première partie de ce chapitre.

2.1 Contexte

Les systèmes d'acquisition 4D peuvent être assez coûteux et complexes à mettre en place. De plus même dans le cas où une telle installation est disponible, procéder à l'acquisition de données n'est jamais une opération anodine ou triviale. Il est donc important pour la communauté de partager les données acquises dans le cadre de nos travaux. En plus d'augmenter le choix disponible pour éprouver les méthodes proposées par les différents chercheurs de la communauté, cela permet de comparer le plus justement possible les résultats obtenus dans le cas de travaux concurrents. Chaque groupe proposant ses données dispose d'un système d'acquisition unique, ainsi il peut exister une grande disparité entre les différentes installations. Que cela soit au niveau matériel ou au niveau des méthodes d'acquisition utilisées.

D'un point de vue matériel, nous pouvons citer les quelques critères suivants :

- **Le nombre de caméras :** de deux à plus d'une trentaine. Il influe directement sur la qualité des données puisque qu'un nombre de vues plus élevé conduit à la création de surfaces plus fines. En contrepartie, l'ajout de caméras rend l'étape d'étalonnage plus délicate et sujette à erreur.
- **Le type de ces caméras :** couleur, infrarouge ou encore caméras de profondeur. Des capteurs de types distincts donnent une information de nature différente sur une même scène. Bien qu'une telle combinaison de capteurs puisse être intéressante pour certains traitements, cela ajoute une difficulté non négligeable lorsque les informations doivent être fusionnées et analysées.
- **La qualité propre des équipements :** qualité des optiques, éclairage, résolution et synchronisation des capteurs par exemple. Ce dernier critère influe principalement sur le niveau de bruit dans les données.

Une autre source de disparité entre les différentes données disponibles est directement liée aux travaux propres aux équipes de recherche à l'origine des acquisitions. Chacun travaillant sur des projets différents tant au niveau des contraintes financières, de temps, qu'au niveau des objectifs ; nous sommes amenés à acquérir des données qui collent au plus près avec ces contraintes. Ainsi le contenu des scènes capturées diffèrent souvent énormément d'une base de données à l'autre. Certains travaux sont orientés vers l'analyse du mouvement humain dans le cadre mono-utilisateur [Vlasic 08], d'autres ont pour but de générer des transitions entre différentes séquences de danse [Starck 07b] par exemple. Mais ces acquisitions ne se limitent pas aux cas de traitement des mouvements humains. Dans le passé, la salle grimage a déjà servi à capturer des scènes avec des animaux, ou contenant plusieurs enfants jouant ensemble.

Toutes ces disparités sont à encourager pour assurer la robustesse des algorithmes testés. En effet le cadre de travail dans le cas du traitement de séquences 4D est très souple et une bonne méthode doit être capable de faire abstraction du plus grand nombre possible des critères cités ci-dessus.

Ce chapitre est organisée comme suit : dans un premier temps, nous définissons dans la section 2.2 les différents types de données mis à disposition dans les différentes bases publiques. Dans un second temps, dans la section 2.3, nous exposons quelques-unes des bases de données 4D disponibles parmi les plus utilisées et reconnues aujourd'hui. Cette seconde partie comprend la présentation de nouvelles données acquises dans le cadre des travaux de cette thèse, leurs caractéristiques techniques ainsi que leurs principaux intérêts.

2.2 Informations disponibles et définitions

Les informations disponibles dans une base de données 4D peuvent varier en nature, en qualité et en quantité. Certaines d'entre elles sont indispensables et nous y apporterons donc un regard plus appuyé dans la suite de ce chapitre. Nous détaillons dans cette section quelques types de données couramment présentes. Nous pouvons distinguer deux classes distinctes d'informations : les informations **brutes** et les informations **pré-traitées**. Ces dernières pouvant être retrouvées à partir des premières, elles ne sont donc pas considérées comme indispensables.

2.2.1 L'étalonnage

L'étalonnage d'un système multi-caméra permet de connaître avec précision la position, l'orientation et les caractéristiques propres à chaque capteur. Dans le cadre d'une caméra couleur, il s'agit des paramètres intrinsèques et extrinsèques de la caméra. Les paramètres d'étalonnage d'un système multi-vue met en relation les informations contenues dans les images avec la structure géométrique de la scène. Il est très important que cette étape soit effectuée le plus précisément possible puisque la très grande majorité des applications faisant usage de l'information 3D ont besoin de cette information. Il existe plusieurs méthodes pour procéder à l'étalonnage d'un système multi-caméra, par exemple [Svoboda 05] pour les systèmes contenant uniquement des caméras couleurs, ou [Hansard 11] dans le cas de systèmes hybrides incluant des caméras de profondeur. Il ne s'agit à proprement parler pas d'une information brute, mais dans la mesure où il est impossible pour chaque utilisateur des données de venir procéder à l'étalonnage du système lui même, nous considérons cette information comme indispensable.

2.2.2 Les images

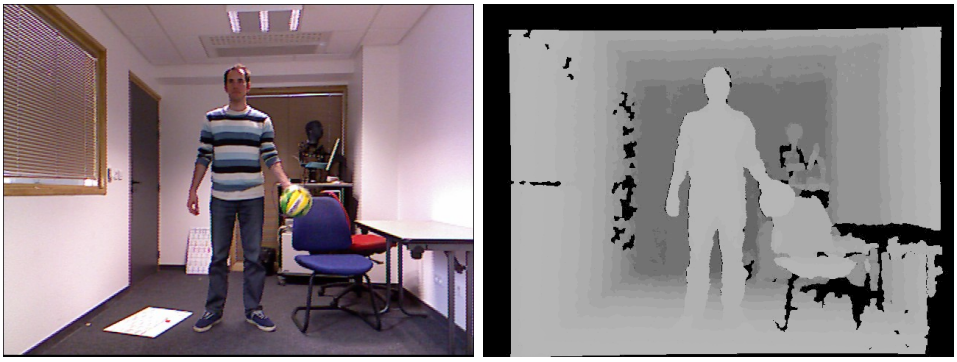


Figure 2.1 – La même scène vue par une caméra couleur et une caméra de profondeur, deux types de capteurs communs.

Les images sont la première source d'information à partir de laquelle tous les traitements sont effectués. Qu'elles soient couleur, infrarouge, à rayon X ou encore de profondeur (voir figure 2.1), les images font l'interface entre le monde observé et son traitement numérique. Leur qualité est donc un élément crucial qui déterminera celle de toutes les applications suivantes. Par qualité nous comprenons plusieurs critères. Premièrement la résolution en pixel influe directement sur la qualité et la quantité de détails visibles dans les images. Typiquement, une application qui se base fortement sur des points d'intérêt aura besoin d'images de haute résolution pour augmenter le nombre et la précision de localisation des points d'intérêt détectés. La résolution dépend aussi du type de capteur, aujourd'hui la résolution classique pour une caméra couleur est d'environs 2M pixels alors qu'elle sera bien plus faible dans le cas d'un capteur de profondeur de type *time of flight*. Un autre critère de qualité est la fréquence d'acquisition. Dans le cas d'une scène dynamique avec des mouvements rapides, il est important d'avoir une fréquence d'acquisition assez élevée pour ne pas trop morceler les actions et les rendre ininterprétables. Aujourd'hui la plupart des systèmes d'acquisition permettent d'obtenir entre 20 et 30 images par seconde. Un élément lié à la fréquence d'acquisition est la synchronisation du système. Il s'agit là d'un point fondamental. Plus le nombre de capteurs augmente plus il est important d'avoir une information synchronisée. En effet il est incohérent d'essayer de fusionner des informations venant de plusieurs images si ces dernières ne voient pas exactement la scène au même instant. Même à une fréquence d'acquisition élevée, le moindre décalage temporel dans la capture d'une scène dynamique peut entraîner de grandes erreurs. Des caractéristiques propres à la scène influencent directement la qualité et l'utilisabilité des images. Dans le cas d'images couleur, nous pouvons citer l'éclairage de la scène. Si ce dernier est trop faible alors les images seront sombres ou sujettes à un fort grain (bruit dans les images). Si au contraire l'éclairage est trop fort ou mal orienté alors

des zones de saturation peuvent apparaître dans les images. Les deux situations sont à éviter puisqu'elles dégradent l'information disponible dans les images.

2.2.3 Le fond

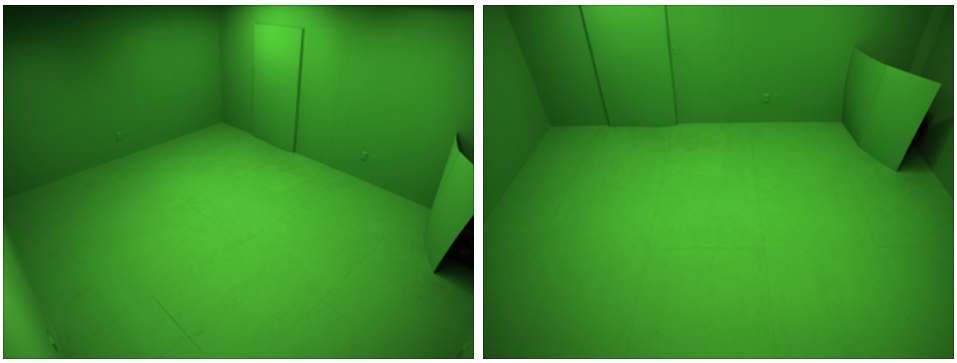


FIGURE 2.2 – Deux images de fond correspondant à deux caméras différentes de l'installation présentée dans [Vlasic 08].

Souvent, la première étape du traitement d'une séquence multi-vue consiste à extraire le ou les objets d'intérêt de l'espace d'acquisition. Pour cela, il est courant de procéder à une soustraction de fond dans les images. C'est-à-dire qu'une distinction est faite dans l'espace image entre les pixels appartenant au contenu de la scène et ceux appartenant au fond. Pour faire cette distinction, de nombreuses techniques ont été mises au point [Piccardi 04]. Les données géométriques présentées dans la suite de ce chapitre sont issues d'une extraction de la forme basée sur une comparaison directe entre les images observées et un modèle du fond appris au préalable. Bien qu'il existe des approches plus complexes ne nécessitant pas de connaissance a priori sur le fond [Lee 10], le fait de fournir des enregistrements pris lorsque la scène est vide pour chaque caméra permet à chaque utilisateur des données d'appliquer l'algorithme qui lui convient le mieux pour l'extraction des zones d'intérêt dans les images. Il s'agit, comme pour les images, d'une information brute.

2.2.4 Les silhouettes

Dans chacune des bases de données présentées dans ce chapitre, les données géométriques fournies sont générées avec des algorithmes qui se basent sur l'information de silhouettes extraite des images. Ces silhouettes sont donc importantes pour les utilisateurs qui souhaitent reconstruire la géométrie de la scène par leurs propres moyens. La qualité de ces silhouettes dépend directement de celle des images, du modèle du fond et de la méthode utilisée pour les calculer.

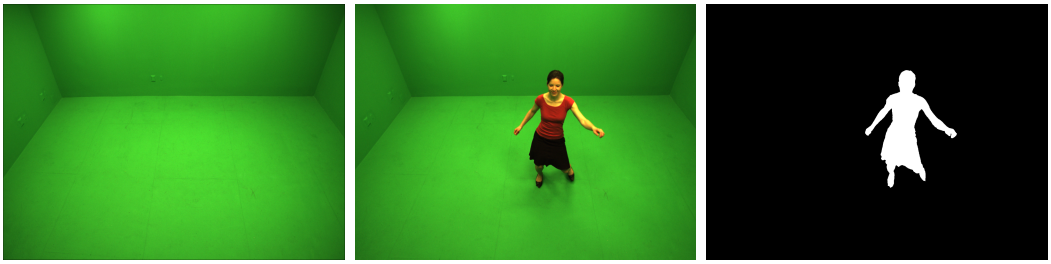


FIGURE 2.3 – Image du fond et de la scène et la silhouette extraite associée.

2.2.5 La géométrie de la scène

Avec les images il s'agit certainement de l'information la plus utilisée par les applications se basant sur des données issues de systèmes multi-caméra. Comme il s'agit souvent du produit final d'une acquisition multi-vue, il existe autant d'approche différente pour obtenir cette information géométrique qu'il existe de bases de données pour des séquence 4D multi-vue. Nous détaillerons dans les sections suivantes, lors de la présentation de chaque séquence, les algorithmes utilisés ainsi que leurs avantages et inconvénients respectifs. Il est toutefois intéressant de noter que cette information géométrique peut prendre plusieurs formes comme par exemple des surfaces sous forme de maillages triangulaires ou des grilles d'occupation voxeliques.

2.2.6 Autres données

Nous avons présenté jusqu'ici l'ensemble des données que nous jugeons indispensables lorsqu'une base de données 4D est rendue publique. À cela s'ajoute d'autres informations dont l'importance varie suivant le type d'application voulue. Parmi ces informations nous pouvons donner l'exemple d'un maillage de référence. Dans certains cas, [Vlasic 08] par exemple, l'information géométrique est obtenue par déformation d'un maillage de référence acquis hors ligne à l'aide d'un scanner laser. Il est alors important de fournir ce modèle de référence pour permettre aux personnes utilisant ces données de travailler dans les mêmes conditions que les auteurs originaux. De la même manière, Vlasic *et al*. fournissent en addition à ces données une description du squelette utilisé et de ses déformations au fil de la séquence. La section 2.3.2 fournit plus de détails à ce sujet.

2.3 Bases de données disponibles publiquement

Nous présentons ici quelques-unes des bases de données 4D disponibles publiquement pour un usage académique. Cette liste n'a pas pour but d'être exhaustive mais plutôt de présenter les principales caractéristiques des séquences

les plus utilisées aujourd'hui.

2.3.1 Surfcap – Université de Surrey

Figure 2.4 – Images couleur et maillages de la séquence *flashkick* de la base de données SurfCap.

En 2007, Starck et Hilton [Starck 07b] présentent une base de données[1] de séquences 4D mettant en scène un danseur de hip-hop professionnel. Les données mises à disposition comprennent huit séquences de durée variable, entre 10 et 20 secondes. Pour chacune de ces séquences, les informations disponibles sont : les flux d'images couleur synchronisés, l'étalonnage du système, les silhouettes et les maillages reconstruits. Une des caractéristiques propres à ces données est que les maillages proposés sont obtenus après un traitement combinant des techniques de cohérence photométrique, d'enveloppes visuelles et d'extraction de points d'intérêt. Ceci permet d'obtenir des maillages assez précis, contenant notamment les plis bien marqués des vêtements. Bien qu'il s'agisse de données pré-traitées, cela ne présente pas un désavantage dans la mesure où le triplet [images–étalonnage–silhouettes] permet de retrouver une information brute de surface.

Le système multi-caméra utilisé se compose de huit caméras HD de résolution 1920×1080 pixels, synchronisées. Ces dernières sont réparties à intervalle régulier sur un cercle de 8m de diamètre à 2m du sol (voir figure 2.9a). Les séquences sont capturées à une fréquence de 25Hz dans une salle dont les murs et le sol sont recouverts d'une peinture bleue afin de faciliter et d'améliorer la qualité de l'étape de soustraction de fond.
La figure 2.4 présente deux exemples d'images et de surfaces maillées d'une des séquences de cette base de données.

D'un point de vue pratique, le traitement de ces données présentes plusieurs challenges. Premièrement, les mouvements présents, dans une des

[1] http://www.ee.surrey.ac.uk/cvssp/visualmedia/visualcontentproduction/projects/surfcap

scènes en particulier (*flashkick*), sont extrêmement rapides et rendent la tâche compliquée dans le cas de suivi de surface ou d'estimation des mouvements. Ensuite, les vêtements unis portés par le danseur font que l'information de texture disponible dans les images couleur est assez pauvre, rendant difficile l'utilisation de méthodes basées sur des points d'intérêts. Cette assertion est d'autant plus vraie dans le cas de mise en correspondance de points d'intérêt entre des images issues de caméras différentes. Dans la mesure où l'écartement entre deux caméras voisines est de 45°, cela influe directement sur l'apparence des objets de la scène dans les différents flux d'images et donc sur la fiabilité des comparaisons de la plupart des descripteurs actuels dont la robustesse décroît souvent rapidement au-delà de 30° [Ancuti 10].

Ces séquences présentent tout de même quelques limitations. Le but initial des travaux de Starck et Hilton, donnant lieu à la création de cette base de données, est de générer des transitions douces entre deux séquences. Les différentes séquences sont ainsi assez proches les unes des autres à la fois au niveau visuel et au niveau sémantique. Les différentes séquences proposées sont assez identiques. La personne mise en scène est toujours la même : mêmes vêtements, même situation de danse et souvent même position initiale. Ce manque de variété pose problème lorsque l'on veut évaluer la robustesse d'algorithmes qui se placent à un niveau plus bas, comme c'est le cas pour l'estimation du flot de scène présentée au chapitre 3 et qui sont sensés fonctionner correctement sur une grande gamme de situations. Une autre source de limitations est la relative pauvreté, mentionnée précédemment, de l'information de texture disponible dans les images couleur. Bien que cela puisse être intéressant de confronter une méthode à des données complexes et non triviales, le fait que la base entière soit du même niveau peut être un frein à son utilisation pour des applications qui reposeraient principalement sur une information fiable de texture. Pour une raison similaire, le faible nombre de caméras peut aussi être vu comme une raison limitante pour des applications basées essentiellement sur des indices photométriques.

2.3.2 Articulated Mesh Animation from Multi-view Silhouettes – MIT

Vlasic *et al*. ont proposé en 2008 une méthode de suivi précis de surface basée sur l'alignement d'un squelette et d'un modèle de référence, obtenu par scanner laser, sur des observations multi-vue [Vlasic 08]. Ces travaux ont mené à la création d'une base de données[2] de séquences 4D aujourd'hui disponible publiquement. Il s'agit d'une série de 10 séquences dont la durée

[2] http://people.csail.mit.edu/drdaniel/mesh_animation/index.html

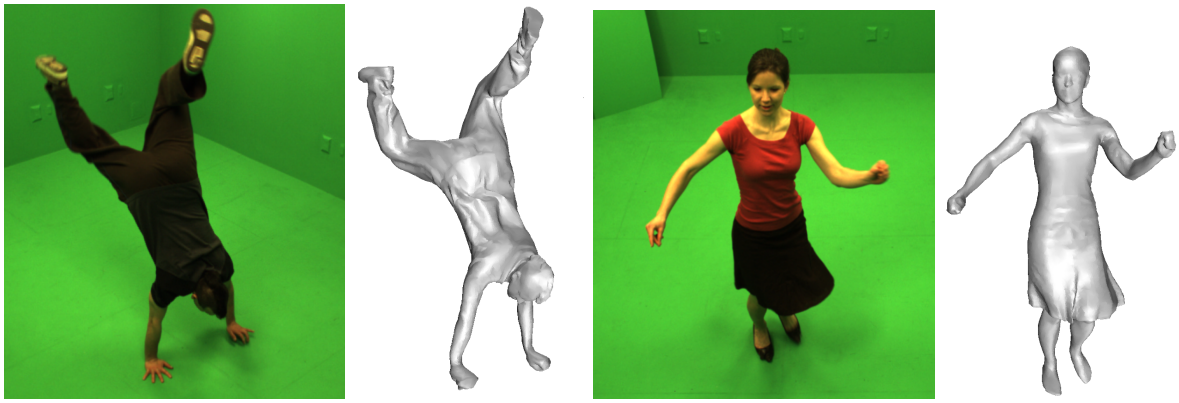


Figure 2.5 – Images couleur et maillages issus des séquences *handstand* et *samba*.

est d'environ sept secondes chacune. Les données distribuées contiennent, pour chaque séquence, les flux d'images couleur, les silhouettes ainsi qu'un flux de maillages triangulaires et les positions successives du squelette, utilisé lors de l'étape de reconstruction 3D, alignées avec les observations. En addition, un certain nombre de données sont communes à toutes les séquences, l'étalonnage du système multi-caméra, des enregistrements du fond pour chaque caméra et les modèles de référence, obtenus à l'aide d'un scanner laser, de chaque acteur présent dans les scènes proposées. La principale particularité de cette base de données est que les maillages fournis sont cohérents temporellement. C'est-à-dire qu'il s'agit en fait pour chaque séquence d'un unique maillage dont les sommets sont déplacés à chaque trame pour obtenir une déformation globale de la surface qui correspond aux observations. Concrètement, dans [Vlasic 08], les auteurs combinent une méthode semi-supervisée qui aligne un squelette dans l'enveloppe visuelle reconstruite à chaque trame avec un modèle de déformation fin de surface qui permet au résultat final d'expliquer au mieux les observations. En d'autres termes, le maillage de référence subit une première déformation dirigée par l'évolution du squelette associé, puis dans un second temps la surface est déformée de manière plus locale pour maximiser son alignement avec les observations, en particulier les silhouettes. Ce type de résultat est très intéressant dans la mesure où en plus d'avoir une représentation 3D de la scène, les informations de mouvement et de vitesse sont directement accessibles grâce à la correspondance point à point des différents maillages. Il faut bien sûr garder à l'esprit qu'il s'agit du résultat de leur travaux et ne pas l'utiliser comme vérité terrain dans le cadre de comparaisons. De manière semblable aux données de la base *SurfCap* présentées précédemment, les données essentielles, à savoir les images, l'étalonnage et les silhouettes, permettent à chacun d'utiliser sa propre méthode de reconstruction 3D afin d'avoir des maillages qui conviennent à chaque application. Il y a tout

de même une différence qu'il est intéressant de noter entre ces deux bases de données. Celle mise à disposition par Vlasic *et al*. propose des enregistrements du fond pour chaque caméra. Ces images sont indispensables pour quelqu'un qui voudrait reprendre tout le processus de reconstruction des maillages 3D, en commençant par faire sa propre soustraction de fond.

Le système multi-caméra mis en place pour ces acquisitions est sensiblement le même que celui présenté précédemment. Il est constitué de huit caméras HD synchronisées, de résolution 1600×1200 pixels. Les caméras sont installées sur les coins et au centre des côtés d'un carré situé à environ 2,5m du sol sont toutes orientées vers le centre de la scène (voir figure 2.9b). L'espace d'acquisition est une salle dont les murs et le sol sont d'une couleur uniforme verte, dans le même souci de simplification et d'amélioration de l'étape de soustraction de fond. La figure 2.5 présente deux couples, images couleur et maillages, issus des séquences *handstand* et *samba*.

Les séquences disponibles dans cette base de données présentent une diversité un peu plus prononcée que pour *SurfCap*. Premièrement elles mettent en scène trois acteurs différents, une femme et deux hommes. Deuxièmement, les actions qui sont effectuées par les acteurs sont plus variées tout en restant simples (marche, saut, pas de danse par exemple). Certaines séquences, en particulier *bouncing*, présentent des déplacements rapides, intéressants pour tester la robustesse d'algorithmes d'estimation de mouvements. D'autres, comme les couples (*march_1 – march_2*) et (*squat_1 – squat_2*), sont particulièrement intéressantes puisqu'elles présentent la même action effectuée par deux personnes différentes. Ce type de données est indispensable pour des applications de reconnaissance d'actions, par exemple, qui ont besoin de données d'apprentissages plus vastes qu'une simple observation, comme expliqué plus en détails dans la section 2.3.3.

Les séquences mises à disposition dans cette base de données souffrent des mêmes limitations que celles présentées précédemment. Le système d'acquisition étant comparable, nous retrouvons les problèmes liés au fort espacement des caméras déjà décrits. De la même manière, les acteurs présents dans ces séquences portent des vêtements relativement sombres et de couleurs unies. Mêmes si les actions présentes dans les différentes séquences sont un peu plus variées que dans le cas précédent, il ne s'agit, et ce dans chaque séquence, que d'une seule personne effectuant un mouvement simple. Jamais ne sont mises en scène des interactions, des éléments de décors ou même autre chose que des humains seuls.

2.3.3 IXmas – Inria Xmas Motion Acquisition Sequences

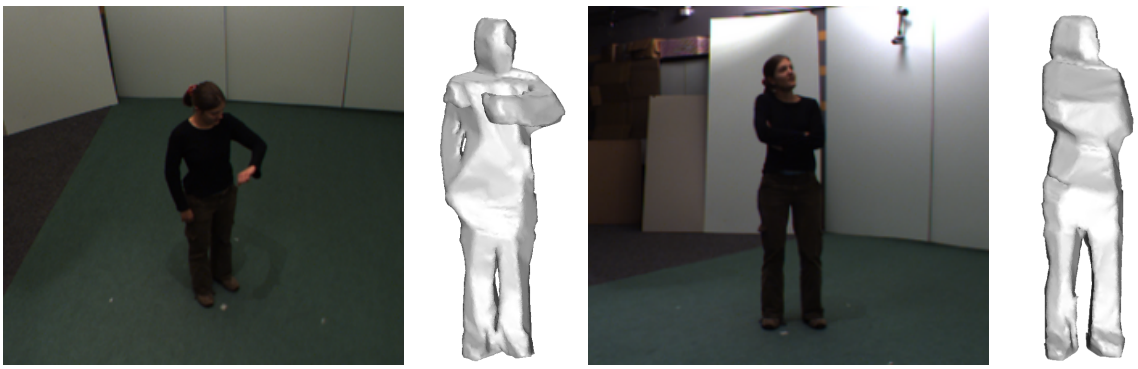


Figure 2.6 – Images couleur et maillages (reconstruits à partir des silhouettes) de deux trames issues de la séquence *Alba1*.

En 2006, Weinland *et al*. [Weinland 06] ont présenté une nouvelle base de données[3] 4D pour la reconnaissance d'actions. Le but initial de ces travaux est de pouvoir identifier, à partir de données séquentielles volumiques, quelles actions sont effectuées par une personne observée. Cette base propose volontairement beaucoup de redondance dans les séquences, afin de fournir assez de données pour servir à la fois de base d'entraînement, ou d'apprentissage, et de base d'évaluation à des méthodes de reconnaissance. 36 séquences sont donc disponibles. Elles correspondent à 12 acteurs distincts effectuant trois fois, le même enchaînement de 15 actions basiques (regarder sa montre, se gratter la tête, s'asseoir ou croiser les bras par exemple). La figure 2.6 montre deux trames caractéristiques des actions "regarder sa montre" et "croiser les bras". La durée des séquences est en moyenne de 50 secondes. Les données disponibles comprennent les flux synchronisés et étalonnés d'images couleur, les silhouettes associées ainsi que des données volumiques sous forme de grilles d'occupation 3D (voxels). Comme pour les acquisitions de Vlasic *et al*., des enregistrements du fond sont disponibles. Les données incluent aussi d'autres informations propres aux applications de reconnaissances d'actions qui ne seront pas détaillées ici. Le fait que les informations 3D soient encodées sous forme de grilles de voxels contrairement aux autres bases de données n'est pas un point très important. il est en effet toujours possible de reconstruire une surface maillée à partir des observations (images et silhouettes).

Le système multi-vue utilisé lors de l'acquisition de cette base de données est assez différent des deux présentés précédemment. Il ne se compose que de cinq caméras de faible résolution, 390×291 pixels, cadencées à 25 images par secondes. Les caméras sont réparties sans motif particulier autour de la

[3] http://4drepository.inrialpes.fr/public/viewgroup/6

scène de manière à permettre une reconstruction de type enveloppe visuelle, leur disposition est illustrée dans la figure 2.9c. La pièce utilisée pour les acquisitions n'a pas été spécialement préparée à cet effet, ce qui influe sur la méthode qu'il convient d'utiliser pour la soustraction de fond. Typiquement, une méthode de *chromakeying* donnerait de très bons résultats dans les deux installations précédemment présentées, mais pour le cas présent il serait nécessaire de faire appel à une méthode différente, basée sur une comparaison pixel à pixel d'un modèle et des observations par exemple.

Ces séquences présentent un intérêt certain dans le cadre de la reconnaissance d'actions. La diversité des acteurs et des actions ainsi que la présence forte de redondance dans les données sont des propriétés très recherchées pour ces applications. Néanmoins, ces séquences présentent deux principaux inconvénients pour une utilisation dans le cadre d'applications nécessitant des données 3D d'assez bonne précision. Premièrement, le faible nombre de points de vue ne permet pas d'obtenir des modèles 3D précis via l'utilisation de méthodes simples et rapides, telles que l'enveloppe visuelle par exemple. Il serait théoriquement possible de reconstruire la géométrie à chaque instant de temps avec précision en utilisant une méthode de type stéréo-vision multi-vue, mais ces approches sont le plus souvent coûteuses, complexes à mettre en place et requièrent pour certaines un paramétrage manuel. De plus ces méthodes ont besoin d'un ensemble d'images de bonne qualité. Ce dernier point permet d'introduire le second défaut notable de ces séquences : la très faible résolution des images disponibles. Avec une résolution presque 20 fois inférieure à celles des bases de données présentées précédemment, il devient assez compliqué d'extraire une information pertinente et de qualité des images couleur.

2.3.4 4D repository

Le portail web 4D Repository[4] met librement à la disposition de la communauté une grande partie des données 4D acquises au fil des années par les différents utilisateurs de la salle GrImage. À ce jour, plus d'une vingtaine de séquences, sans compter la base IXmas, sont proposées au téléchargement. Durant le déroulement de cette thèse, nous avons procédé à l'acquisition de plusieurs séquences. Elles seront détaillées dans la seconde partie de cette section(2.3.4.2), à la suite de la présentation des séquences déjà existantes (2.3.4.1).

2.3.4.1 Données déjà existantes

Nous ne parlerons dans cette section que de la dernière vague de mise à jour du dépôt, datant du printemps 2009. Les données ajoutées à cette période

[4] http://4drepository.inrialpes.fr

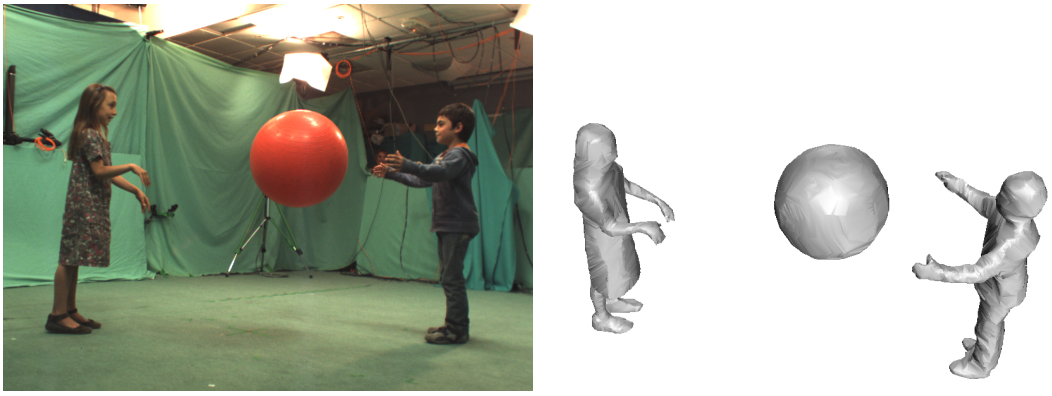


Figure 2.7 – Image couleur et maillage (enveloppe visuelle) de deux trames issues de la séquence *two children ball*.

correspondent à des séquences acquises par Kiran Varanasi, principalement, dans le cadre de ses travaux de thèse. Cela concerne une petite vingtaine de séquences de durée variable, comprise entre cinq et 20 secondes. Toutes ces séquences ont été acquises dans les mêmes conditions et proposent les mêmes ensembles de données. Cela comprend, l'étalonnage du système, les images couleurs, des enregistrements du fond, les silhouettes et pour finir, les surfaces maillées reconstruites par un algorithme d'enveloppes visuelles. Le système comprend un ensemble de 16 caméras couleurs de résolution 1624×1224, synchronisées à 25Hz. Les caméras sont réparties en dôme autour de la scène afin d'avoir une bonne couverture mais sans suivre de motif particulier. cette répartition est illustré dans la figure 2.9d.

Le but initial de ses séquences est de proposer une grande variété de situations, d'actions et d'acteurs. Les séquences sont ainsi classées en trois groupes principaux. Les données du groupe *Children* (voir figure 2.7) présentent des enfants jouant, seul, à deux avec ou sans balle. Le groupe *Martial* met en avant des séquences de combat type Jiu-jitsu. Le dernier groupe, nommé *Dog*, rassemble des scènes présentant un chien à qui son maître fait faire des tours. Le principal attrait de ces données, par rapport aux séquences présentées jusqu'ici, vient de leur diversité et du fait qu'elles présentent des scènes ne se limitant pas à un unique acteur. Au contraire, l'accent est mis sur les interactions entre les acteurs et l'environnement, qu'il s'agisse d'autres acteurs, d'objets où même d'animaux.

Le traitement de ces données nécessite de faire appel à des méthodes les plus générales possibles, n'ayant que très peu, voire aucun, a priori sur la scène. En effet, il semble difficile d'imaginer disposer d'un modèle obtenu par scanner laser d'un chien ou d'une scène aussi étendue que celle des deux enfants jouant au ballon, présentée dans la figure ci dessus. Ce dernier point est aussi un frein

à l'utilisation d'une représentation des acteurs par une grille d'occupation. Avec une scène aussi étendue, il est délicat de faire un compromis efficace entre utilisation mémoire et niveau de détail de la surface. Les séquences du groupe *martial* sont quant à elles complexes à modéliser et traiter de par la nature des interactions entre les deux adversaires. Il est difficile pour un regard non humain de faire la distinction entre les deux personnes lorsque ces dernières s'empoignent comme c'est le cas dans ces séquences de combats à mains nues. En plus de la nature des interactions dans les séquences de ce groupe, les deux personnes qui combattent son habillées d'un kimono blanc uni ; ce qui limite considérablement l'utilisation de méthodes basées sur des points d'intérêt dans les images. L'utilisation d'applications fondées sur des modèles à base de squelettes sera quant à elle limitée dans le cas des séquences du groupe *dog* dans la mesure où elles sont généralement plus orientées vers le traitement des mouvements humains. Il en va de même pour les séquences du groupe *children* dans lesquelles les enfants jouent avec un ballon, scène qui se prête difficilement à une représentation par un squelette.

2.3.4.2 Données acquises durant cette thèse

Lors de cette thèse, l'occasion s'est présentée de procéder à de nouvelles acquisitions dans la salle GrImage. La principale différence avec les données présentées jusque là, et notamment avec les données de la section précédente également acquises dans la salle GrImage, est le nombre de caméras utilisé. En l'occurrence nous avons mis en place un système multi-vue disposant de 32 caméras. Le nombre de capteurs est donc deux fois supérieur à celui de la précédente campagne d'acquisition et quatre fois supérieur aux installations présentés dans les sections 2.3.1 et 2.3.2.

Comme dans le cas précédent les données rendues disponibles à la communauté comprennent l'étalonnage du système, les flux d'images couleur, des enregistrements du fond, les silhouettes ainsi que des surfaces maillées obtenues par reconstruction de la scène à chaque instant par un algorithme d'enveloppes visuelles. Les caméras sont placées en dôme autour de la scène de manière à correctement couvrir l'espace d'acquisition, leur répartition dans l'espace est représentée dans la figure 2.9e. Elles sont synchronisées à une fréquence de 25 images par secondes et fournissent des images de résolution 1624×1224 pixels.

Le principal intérêt de ces nouvelles données réside clairement dans le nombre de capteurs utilisés. Avoir plus d'information n'est jamais un défaut dans la mesure où il est possible de ne pas tout prendre en compte. Une telle densité de capteurs est un point important dans le cas d'applications utilisant des mises en correspondance de points d'intérêt entre les différentes images.

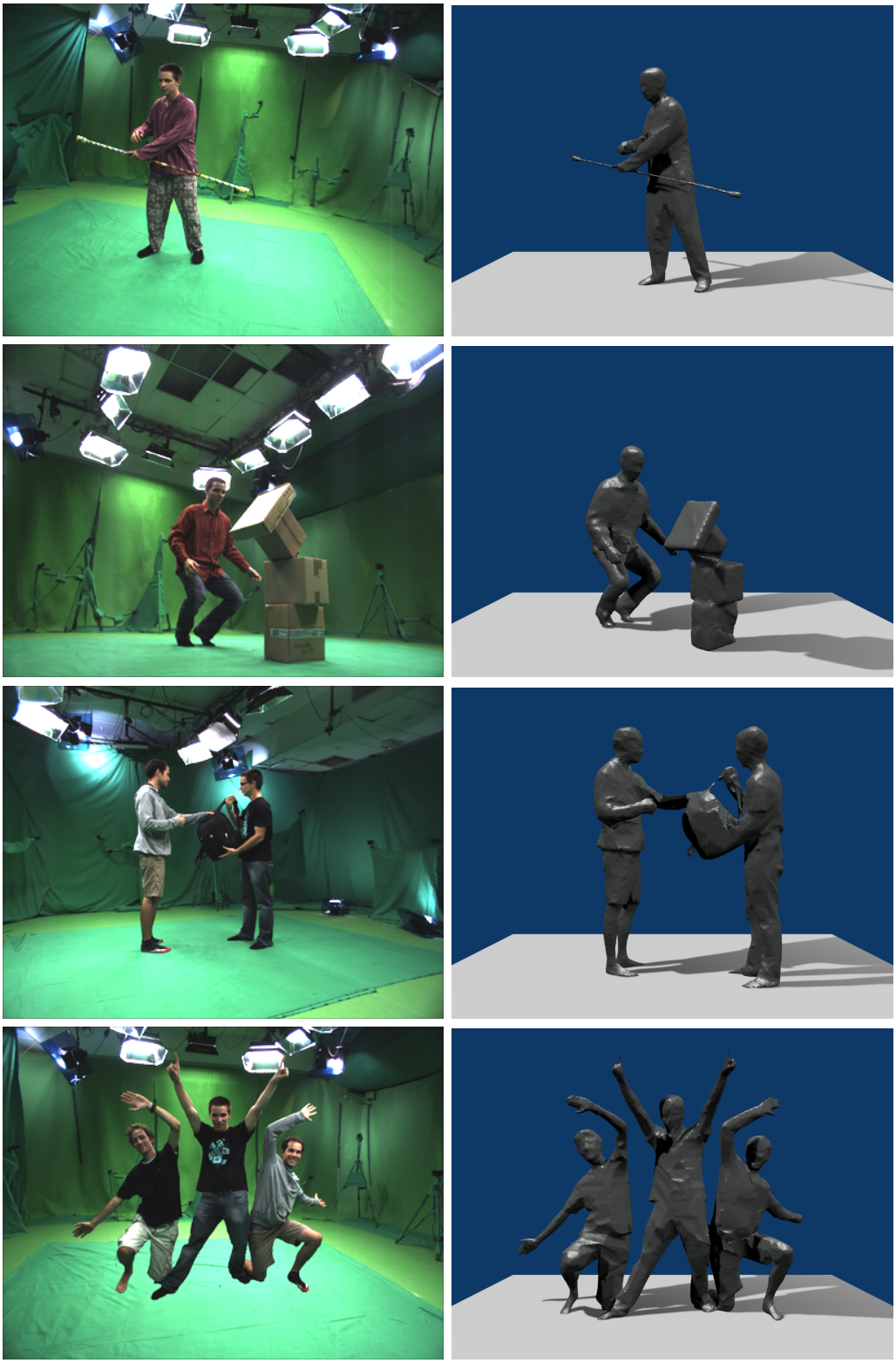


FIGURE 2.8 – Images couleurs et maillages (reconstruits à partir des silhouettes) issus de quatre séquences acquises avec le système à 32 caméras.

En effet cette étape est d'autant moins sujette à erreur que les points de vue sont proches (faible *baseline*). Lors de ces acquisitions, nous avons tenu à proposer des scènes au contenu varié. Contrairement aux données présentées dans la section précédente, pour lesquelles toutes les séquences présentent un challenge non négligeable ; nous avons acquis plusieurs scènes représentant un unique acteur faisant des mouvements simples, tel que marcher ou lever les bras. Ces séquences en particulier ont été utilisées lors des travaux présentés dans cette thèse et seront donc détaillées plus loin dans ce manuscrit (sections 3.6.3 et 4.4.2). Néanmoins nous avons aussi tenu à proposer des scènes plus complexes, mettant en situation plusieurs acteurs interagissant avec leur environnement. La figure 2.8 expose quelques unes de ces séquences ; de haut en bas un homme jonglant avec un bâton, un homme empilant et faisant chuter un ensemble de boites en carton, deux personnes échangeant un sac à dos et pour finir une scène contenant trois personnes. Cette liste est bien sûr non exhaustive.

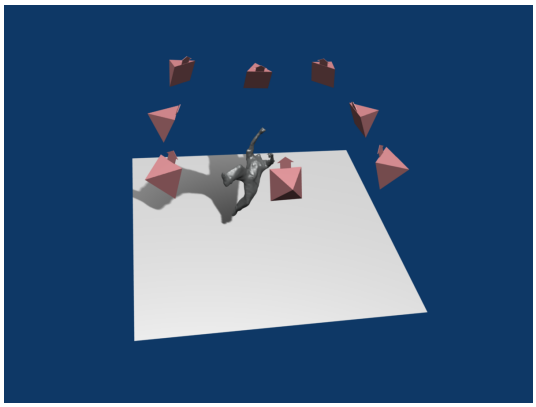
Les séquences acquises sont relativement longues, de l'ordre de 30 à 40 secondes pour certaines. Il s'agit d'une caractéristique importante pour évaluer la robustesse ou la dérive d'algorithmes de suivi d'objets par exemple. En effet ce type d'applications accumulent souvent de l'erreur d'une trame à la suivante, si bien que la qualité des résultats obtenus se détériore au fil du temps.

2.4 Conclusion

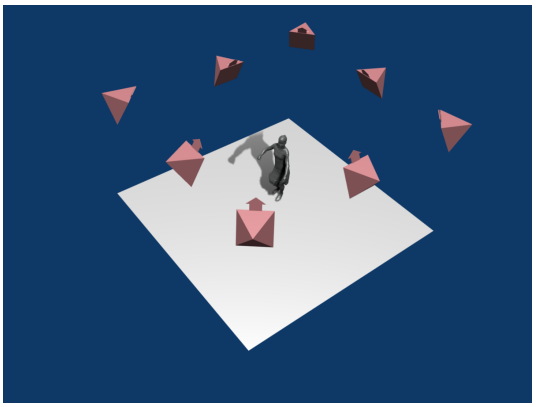
Le tableau 2.1 présente une synthèse rapide des différents éléments de comparaison entre les bases de données présentées dans ce chapitre. La figure 2.9, regroupe toutes les dispositions de caméras des différentes installations citées précédemment. Avant de clore ce chapitre, il est important de noter que chaque base de données 4D dispose de ses propres avantages et inconvénients et qu'il est important d'en être conscient avant de choisir une séquence en particulier pour évaluer une méthode ou un algorithme. En choisissant des données pertinentes lors d'évaluations ou de comparaisons, nous pouvons faciliter l'appréciation de nos travaux par les membres de la communauté. Une séquence trop simple présente aussi peu d'intérêt qu'une séquences trop complexe même si les résultats obtenus sont parfaits. L'important est d'être conscient des limites des travaux proposés et de pouvoir choisir en conséquence un ou des jeux de données, mettant en évidence les différents avantages et limitations de la méthode testée.

Base	# de séquences	# de caméras	Avantages	Données
Surfcap 2.3.1	8	8	information géométrique de bonne qualité. Les mouvements rapides du danseur présentent un challenge intéressant pour nos applications d'estimation du flot de scène	étalonnage, images, fond, silhouettes, surfaces maillées
MIT 2.3.2	10	8	maillages cohérents temporellement	étalonnage, images, fond, silhouettes, surface maillées, maillage de référence, poses estimées du squelette pour chaque trame
IXmas 2.3.3	36	5	grande répétitivité des actions, idéale pour des applications de reconnaissance d'actions ou d'apprentissage	étalonnage, images, silhouettes, grille d'occupation voxélique, annotations des séquences
GrImage 16 2.3.4.1	20+	16	grande variété d'actions et d'acteurs dont des animaux, interactions avec des objets	étalonnage, images, fond, silhouettes, surfaces maillées
GrImage 32 2.3.4.2	15	32	grand nombre de points de vue, qualité des enveloppes visuelles, diversité des actions et des environnements	étalonnage, images, fond, silhouettes, surfaces maillées

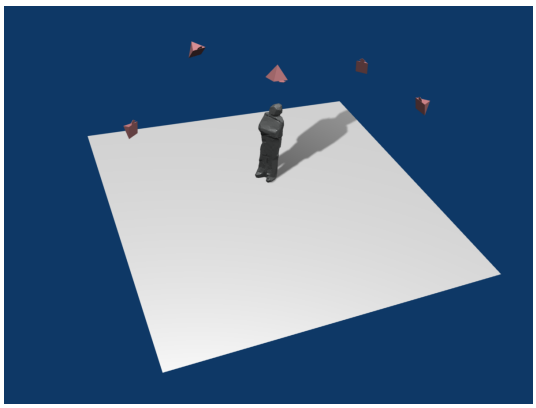
Table 2.1 – Récapitulatif des caractéristiques des principales bases de données 4D.



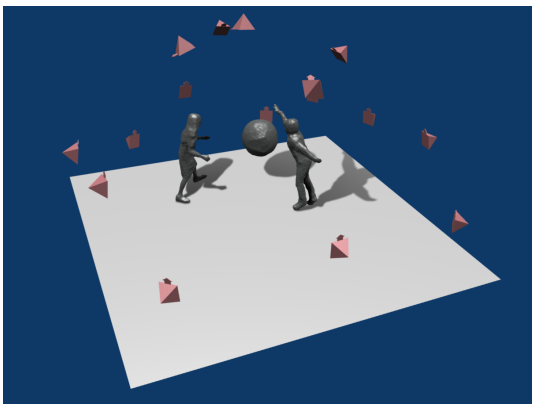
(a) SurfCap 2.3.1



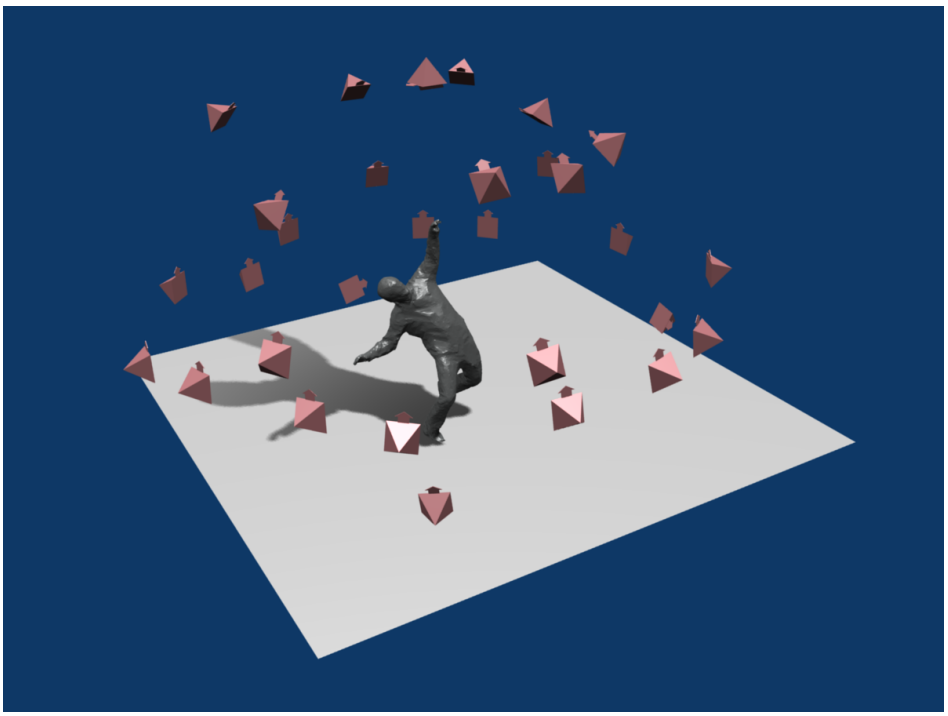
(b) MIT 2.3.2



(c) Xmas 2.3.3



(d) GrImage 16 2.3.4.1



(e) GrImage 32 2.3.4.2

Figure 2.9 – Comparaison des positions des caméras dans les installations présentées, *Surfcap*, Vlasic *et al.* et GrImage.

Première partie

Flot de scène

Chapitre 3

Flot de scène

Sommaire

Résumé

Dans ce chapitre nous nous intéressons à l'estimation des champs de déplacement 3D denses d'une scène non rigide, en mouvement, capturée par un système multi-caméra. La motivation vient des applications multi-caméra qui nécessitent une information de mouvement pour accomplir des tâches telles que le suivi de surface ou la segmentation. Dans cette optique nous présentons une approche nouvelle qui permet de calculer efficacement un champ de déplacement 3D, en utilisant des informations visuelles de bas niveau et des contraintes géométriques. La contribution principale est la proposition d'un cadre unifié qui

combine des contraintes de flot pour de petits déplacements et des correspondances temporelles éparses pour les déplacements importants. Ces deux types d'informations sont fusionnés sur une représentation surfacique de la scène en utilisant une contrainte de rigidité locale. Le problème se formule comme une optimisation linéaire permettant une implémentation rapide grâce à une approche variationnelle. La méthode proposée s'adapte de manière quasiment identique que les informations de surface proviennent d'une reconstruction 3D complète, par exemple en utilisant l'enveloppe visuelle, ou d'une simple carte de profondeur. Les expérimentations menées sur des données synthétiques et réelles démontrent les intérêts respectifs du flot et des informations éparses, ainsi que leur efficacité conjointe pour calculer les déplacements d'une scène dynamique de manière robuste.

3.1 Contexte et motivations

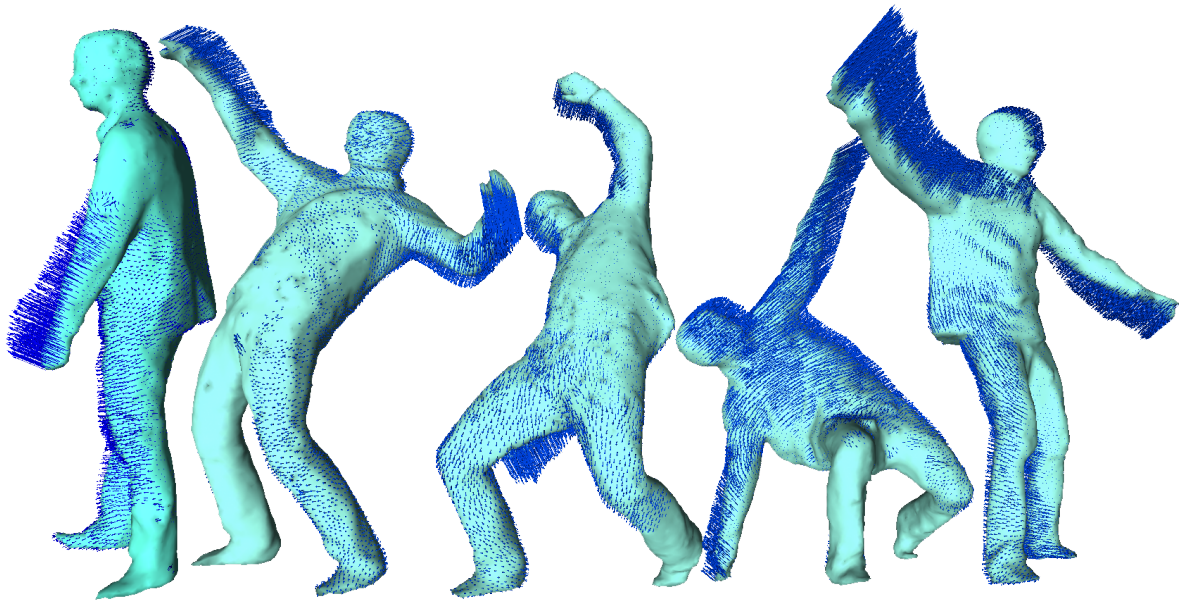


Figure 3.1 – Exemple de flot de scène dense (en bleu) calculé à partir de correspondances de points d'intérêts 2D et 3D et de flot de normal dense.

Le déplacement est une source d'information importante lors de l'analyse et de l'interprétation de scènes dynamiques. Il fournit une information riche et discriminante sur les objets qui composent la scène et est utilisé, par exemple, dans les systèmes de vision humaine et artificielle pour suivre et délimiter ces objets. L'intérêt apparaît surtout dans le cas d'applications interactives, telles que les jeux vidéos ou les environnements intelligents, pour lesquels le mouvement est une source d'information primordiale dans la

boucle perception-action. Pour cela, l'observation des pixels, issus des images, fournit des informations utiles sur le mouvement, à travers les variations temporelles de la fonction d'intensité. Dans une configuration mono-caméra, ces variations permettent d'estimer des champs de vitesse 2D denses dans l'image : le *flot optique*. L'estimation du flot optique a été un sujet d'intérêt dans la communauté de la vision par ordinateur ces dernières dizaines d'années et de multiples méthodes ont été proposées [Barron 94, Horn 81, Lucas 81].

Dans le cas d'un système multi-caméra, l'intégration depuis les différents points de vue permet de considérer le mouvement des points 3D de la surface observée et d'estimer le champ de vecteur de déplacement 3D : le *flot de scène* [Vedula 05, Neumann 02]. Autant en 2D qu'en 3D, l'information de mouvement ne peut pas être déterminée indépendamment pour chaque point avec pour seule information la variation de la fonction d'intensité ; une contrainte additionnelle doit-être introduite, par exemple, une hypothèse de continuité du champ de mouvement. De plus, du fait de l'approximation des dérivées par la méthode des différences finies, l'estimation du flot est connue pour être limitée à de petits déplacements. Bien que plusieurs approches en 2D aient été proposées pour faire face à ces limitations [Xu 10], moins d'efforts ont été consacrés au cas de la 3D. Il est bien sur possible d'utiliser des capteurs actifs ou des systèmes de vision basés marqueurs. Ces derniers peuvent fournir directement un ensemble épars d'informations de déplacement sur des scènes en mouvement. Mais ces systèmes sortent du cadre de nos travaux, en effet nous nous contraignons à utiliser un système le moins intrusif possible, c'est-à-dire sans marqueur et sans hypothèse sur l'éclairage, voir même sous éclairage naturel.

Dans ce travail, nous avons étudié la façon d'intégrer, de manière efficace, diverses contraintes pour estimer des informations de mouvements denses instantanés sur des surfaces 3D, à partir des variations temporelles de la fonction d'intensité issue de plusieurs images. Notre motivation première a été de fournir des indices de mouvement robuste qui peuvent être directement utilisés par une application interactive, ou qui peuvent être introduits dans des applications plus avancées comme le suivi de surface ou la segmentation. Bien que notre but ait été d'intégrer le calcul des champs de vitesse avec notre application de reconstruction 3D, l'approche n'est pas limitée à un scénario spécifique et fonctionne pour toute application qui peut bénéficier d'une information de mouvement de bas niveau.

La plupart des approches existantes qui estiment le flot de scène font l'hypothèse des petits déplacements entre les instants de temps pour lesquels les approximations aux différences finies des dérivées temporelles sont valides.

Cependant, cette hypothèse est souvent incorrecte avec les systèmes d'acquisition actuels et des objets réels en mouvement. En effet, l'amplitude des mouvements observés et la fréquence d'acquisition utilisée ne permettent pas d'effectuer cette hypothèse dans tous les cas.

Dans ce chapitre, nous présentons une méthode unifiée permettant de lier de manière cohérente les contraintes visuelles, issues des images consécutives temporellement, avec des contraintes de déformation de surface. Pour traiter les grands déplacements, nous utilisons des mises en correspondances temporelles entre les images issues d'une même caméra. Ces contraintes agissent comme des points d'ancrage pour les régions de la surface où les déplacements sont plus importants et où les informations de variation d'intensité ne sont pas utiles. Ces contraintes visuelles sont diffusées sur la surface grâce à un schéma laplacien qui régularise les vecteurs de déplacements estimés entre les points voisins de la surface. Un élément clé de cette méthode est qu'elle conduit à des optimisations linéaires ce qui permettrait, à terme, une implémentation temps-réel.

Notion de Surface. La méthode que nous proposons ici nécessite en entrée un ensemble de flux d'images couleur venant d'un système multi-caméra pré-étalonné et nous supposons qu'une information géométrique sur la scène est disponible. Nous avons adapté notre méthode à deux cas de figures distincts. Le premier est le cas où l'information de surface provient d'une reconstruction 3D indépendante de chaque trame de la séquence traitée, en utilisant l'enveloppe visuelle par exemple. Le second cas de figure est un système où la géométrie partielle de la scène est donnée par une caméra de profondeur, par exemple des caméras à temps de vol ou à lumière structurée, qui fournissent directement une information 3D, sans recourir à un traitement multi-vue additionnel. La carte de profondeur représente un nuage de points 3D à partir duquel une surface maillée peut être construite en utilisant la connectivité dans l'image de profondeur. C'est-à-dire que chaque pixel devient un sommet du maillage en 3D, connecté à ses voisins dans l'image. Dans la suite de ce chapitre, sauf mention contraire, nous appellerons *surface* cette information géométrique indistinctement de sa provenance.

Ce chapitre est organisé de la manière suivante : dans un premier temps, nous présentons un état de l'art dans la section 3.2, ensuite nous entrons dans les détails de la méthode proposée dans la section 3.3. Dans la section 3.5, nous expliquons les différents choix d'implémentations que nous avons fait suivant le type de données traitées. Les résultats obtenus dans le cas des enveloppes visuelles et celui des cartes de profondeurs sont présentés respectivement dans les sections 3.6 et 3.7. Nous concluons ce chapitre dans la section 3.8.

3.2 Etat de l'art

Un grand nombre de travaux ont été menés dans le but d'estimer des champs de déplacement en utilisant des informations photométriques. Les premiers travaux dans ce domaine se concentraient sur le champ de déplacement entre deux images consécutives. L'estimation du flot optique par [Horn 81, Barron 94] fait appel aux contraintes de flot normal dérivées des variations d'intensité dans les images. Lorsque l'information vient d'images stéréo, le champ de déplacement 3D, le flot de scène, peut être calculé.

Dans un article fondateur sur le flot de scène, Vedula *et al*. [Vedula 05] explicitent la contrainte de flot normal qui lie les dérivées de la fonction d'intensité dans les images au flot de scène des points 3D de la surface. Comme mentionné précédemment, ces contraintes ne permettent pas d'estimer le flot de scène de façon indépendante à un point de la surface, des contraintes supplémentaires doivent être introduites. Au lieu d'utiliser la contrainte de flot normal, un algorithme est proposé qui estime de façon linéaire le flot de scène à partir de la géométrie 3D de la surface et du flot optique 2D. Le flot optique permet de mieux contraindre le flot de scène que le flot normal, mais son estimation est fondée sur des hypothèses de lissage qui tiennent rarement dans le plan image mais sont souvent vérifiées dans le cas de surfaces.

Dans [Neumann 02], Neumann et Aloimonos introduisent un modèle de subdivision de surface qui permet d'intégrer sur la surface, les contraintes de flot normal avec des contraintes de régularisation. Néanmoins, cette solution globale suppose encore de n'être en présence que de petits mouvements et peut difficilement faire face à des cas comme ceux présentés dans nos expérimentations.

Une autre stratégie est d'estimer conjointement la structure et le mouvement. Cette voie est explorée par [Pons 05, Basha 10]. Dans [Pons 05] Pons *et al*., présentent une approche variationnelle qui optimise un critère de cohérence photométrique au lieu des contraintes de flot normal. L'intérêt est que la cohérence spatiale comme la cohérence temporelle peuvent être appliquées, mais au prix d'une optimisation coûteuse en calcul. Au contraire, notre objectif n'est pas d'optimiser la forme observée, mais de fournir une information de mouvement dense de façon efficace et rapide.

Plusieurs travaux [Zhang 01, Isard 06, Wedel 08, Huguet 07, Li 08] considèrent le cas où la structure de la scène est décrite par une carte de disparité issue d'un système stéréoscopique. Ils proposent l'estimation combinée de la disparité spatiale et temporelle du mouvement 3D. Des travaux récents [Rabe 10] ajoutent à ceci une contrainte de cohérence temporelle. Nous considérons une situation différente dans laquelle la surface de la forme observée est connue,

par exemple, un maillage obtenu en utilisant une approche multi-vues. Ceci permet une régularisation du champ de déplacement sur un domaine où les hypothèses de régularité sont vérifiées.

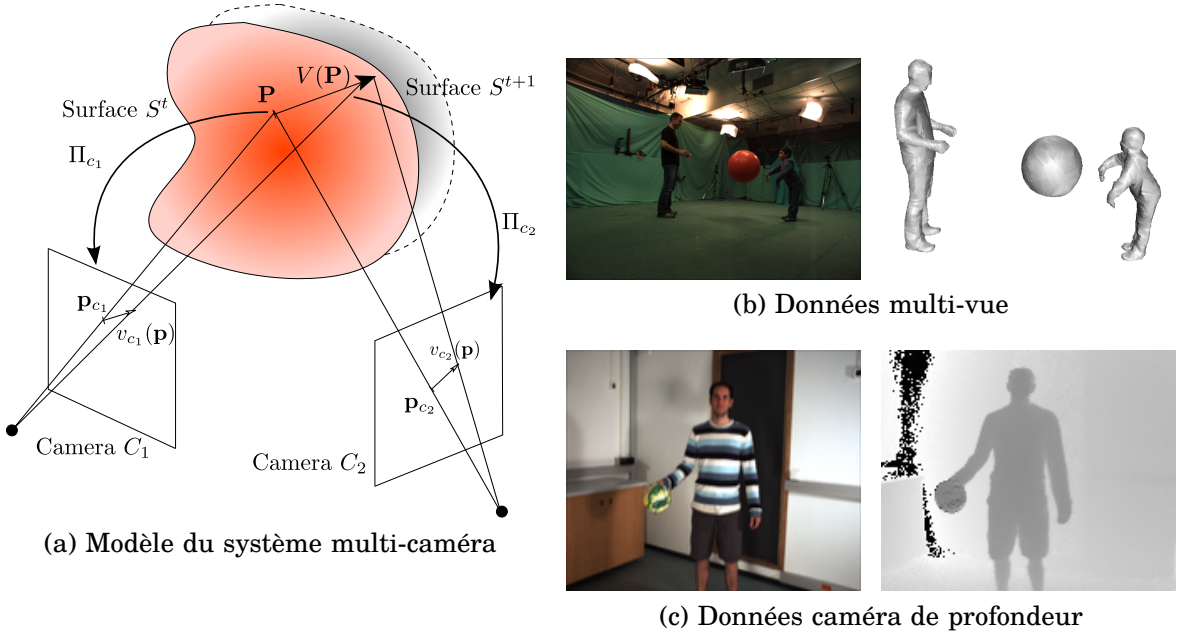
Il convient de mentionner également les approches récentes sur le suivi temporel de surface [Starck 07b, Varanasi 08, Naveed 08, Cagniart 10] qui peuvent également fournir des champs de vitesse. C'est en effet une conséquence de la mise en correspondance de surfaces dans le temps. Notre but est ici différent, notre méthode ne fait aucune hypothèse sur la forme observée, seulement quelques hypothèses sur le modèle de déformation locale de la surface. Notre méthode fournit des informations bas niveau, le mouvement instantané, qui peuvent à leur tour être utilisées comme données d'entrée d'une méthode d'appariement ou de suivi de surface.

Nos contributions à l'égard des approches mentionnées sont de trois ordres :
- En suivant les travaux sur l'estimation robuste du flot optique 2D [Liu 08, Xu 10], nous utilisons avantageusement les valeurs de déplacement robuste fournies par le suivi de points d'intérêts dans des images consécutives dans le temps. Ces points intérêts permettent de contraindre les grands déplacements alors que les contraintes de flot de normal permettent de modéliser précisément les déplacements les plus petits.
- Une résolution linéaire combine ces différentes contraintes visuelles avec un modèle de déformation de surface et permet une résolution itérative ainsi qu'un raffinement de type multi-échelle.
- Un cadre de résolution qui prend en compte les données venant de systèmes multi-caméra de natures différentes, contenant un nombre quelconque de caméras couleur et pouvant intégrer un capteur de profondeur.

3.3 Estimation du flot de scène

L'approche proposée estime directement un champ de mouvement 3D sur la surface en utilisant des contraintes photométriques 2D. Pour cela, elle prend en entrée des flux d'images couleur et de surfaces venant d'un système multi-caméra pré-étalonnées et synchronisées. La configuration prise en compte se compose d'une ou plusieurs caméras couleur et d'un flux de surface. Dans la suite, pour des raisons de simplicité, nous ne détaillons que le cas disposant d'une caméra couleur. Néanmoins l'extension à plusieurs caméras couleur est directe et sera expliquée plus loin dans ce chapitre. Un des avantages de la méthode proposée est qu'elle s'adapte indifféremment à une grande varité de systèmes d'acquisition. Du simple couple comprenant une caméra couleur et une caméra de profondeur, telle que la caméra Kinect, jusqu'à la salle d'acquisition complète contenant 32 caméras voir plus.

3.3.1 Notations



(a) Modèle du système multi-caméra

(b) Données multi-vue

(c) Données caméra de profondeur

Figure 3.2 – (a) Modèle multi-caméra considéré par notre approche, (b) type de données en entrée dans le cas où la surface est reconstruite par une méthode type enveloppe visuelle et (c) type de données dans le cas où l'on dispose d'une caméra de profondeur.

La surface au temps t est dénotée $\mathcal{S}^t \subset \mathbb{R}^3$ et associée à un ensemble d'images couleur, acquises au même instant de temps, noté $\mathcal{I}^t = \{\mathbf{I}_c^t \mid c \in [1..N]\}$. Un point 3D P sur la surface est décrit par le vecteur $(x, y, z)^T \in \mathbb{R}^3$. Sa projection dans l'image $\mathbf{I}^t$ est le point 2D p qui a comme coordonnées $(u, v)^T \in \mathbb{R}^2$, calculées en utilisant la matrice de projection 3x4 $\mathbf{\Pi} : \mathbb{R}^3 \mapsto \mathbb{R}^2$ de la caméra (voir figure 3.2). La région 3D de l'image correspondant à la visibilité de $\mathcal{S}^t$ dans $\mathbf{I}^t$ est notée $\Omega^t = \mathbf{\Pi}\mathcal{S}^t$.

Notre méthode recherche le meilleur champ de déplacement 3D de la surface entre le temps t et $t+1$, noté $V^t : \mathcal{S}^t \mapsto \mathbb{R}^3$ avec $V^t(\mathbf{P}) = \frac{\mathrm{d}\mathbf{P}}{\mathrm{d}t}$ $\forall$ $\mathbf{P} \in \mathcal{S}^t$. Ce champ de déplacement est contraint par :

- les données d'entrée comme le jeu d'images calibrées $\mathbf{I}^t$ et $\mathbf{I}^{t+1}$, et les surfaces $\mathcal{S}^t$ et $\mathcal{S}^{t+1}$,
- un modèle de déformation.

Ainsi le flot optique v^t est la projection du champ du flot de scène V^t sur l'image couleur $\mathbf{I}^t$. La relation entre un petit déplacement à la surface de $\mathcal{S}^t$ et son image prise par la caméra couleur est décrite par la matrice jacobienne 2x3 $J_{\mathbf{\Pi}}(\mathbf{p}) = \frac{\partial \mathbf{p}}{\partial \mathbf{P}}$, telle que $v^t = J_{\mathbf{\Pi}}(\mathbf{p})V^t$.

pour estimer le flot 3D $V^t(\mathbf{P})$, le problème est formulé sous la forme d'une optimisation où un terme d'attache aux données renforçant les contraintes photométriques est associé à un terme de lissage favorisant un champ de déplacement régulier :

$$\mathbf{E} = \mathbf{E}_{data} + \mathbf{E}_{smooth}. \tag{3.1}$$

Le terme d'attache aux données contrôle à la fois les grands et petits déplacements tandis que le terme de lissage impose un modèle de déformation avec des contraintes de rigidité locale.

Dans les sections suivantes, nous expliciterons les contraintes visuelles et géométriques venant des données en entrée et le modèle de déformation utilisé pour propager le mouvement sur la surface.

3.3.2 Contraintes visuelles

Notre méthode peut utiliser trois types de contraintes visuelles pour estimer le déplacement 3D :

1. des contraintes denses de flot normal dans les images,
2. des correspondances éparses de points d'intérêts 3D,
3. des correspondances éparses de points d'intérêts 2D.

Chacune de ces contraintes mènera à un terme dans notre fonctionnelle d'erreur telle qu'elle sera réécrite dans la section 3.4 et qui décrit comment le champ de déplacement estimé se rapporte aux observations. Ces contraintes n'incluent pas de cohérence photométrique spatiale ou temporelle car ces dernières impliquent des termes non linéaires dans la fonctionnelle d'erreur. Elles sont plus adaptées aux problèmes liés à l'optimisation de la forme de la surface qu'à l'estimation plus bas niveau du mouvement.

3.3.2.1 Flot normal dense

Des informations denses sur V^t peuvent être obtenues en utilisant le flot optique accessible dans les images. En effet, en prenant comme hypothèse que l'illumination reste constante entre $\mathbf{p}^{t+1}$ et $\mathbf{p}^t$, la projection du même point de la surface entre deux trames consécutives, on peut définir l'équation du **flot normal** [Barron 94] comme étant :

$$\nabla I^t . v^t + \frac{\mathrm{d}I^t}{\mathrm{d}t} = 0\,,$$

ou équivalent en 3D à [Vedula 05] :

$$\nabla I^t . \left[J_{\mathbf{\Pi}} V^t\right] + \frac{\mathrm{d}I^t}{\mathrm{d}t} = 0\,.$$

La fonction d'erreur suivante décrit comment la projection v^t dans l'image du champ de déplacement 3D calculé vérifie la contrainte de flot normal :

$$\mathbf{E}_{flow} = \int_{\Omega^t} \|\nabla I^t . \left[J_{\mathbf{\Pi}} V^t\right] + \frac{\mathrm{d}I^t}{\mathrm{d}t}\|^2 \, \mathrm{d}\mathbf{p} \,. \tag{3.2}$$

Cependant, cette fonctionelle ne permet de contraindre le mouvement 2D que dans la direction tangente au gradient d'intensité dans l'image ∇I^t. C'est-à-dire que seule la projection du vecteur de flot optique sur l'axe du gradient d'intensité dans l'image est connue. Cette limitation est connue sous le nom de problème de l'ouverture (*aperture problem*) dans le cas du l'estimation du flot optique. Il s'avère que ce problème s'étend en 3D pour l'estimation du flot de scène. En reprenant la démonstration faite par Vedula *et al.* dans [Vedula 05], nous pouvons noter que les contraintes de flot normal ne sont pas dépendantes du point de vue et qu'ainsi, quelque soit le nombre de points de vue considérés, l'information accumulée sera toujours ambigüe. En effet la seule information complète qu'il est possible d'obtenir est la projection du flot de scène associé à un point P de la surface sur le plan tangent à la surface en ce même point P. Ce concept est illustré par la figure 3.3.

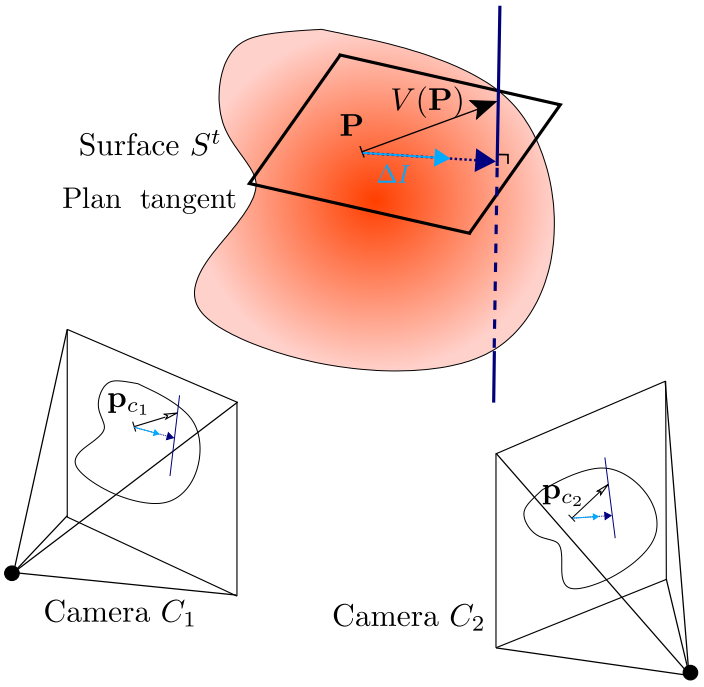


Figure 3.3 – Illustration de l'extension du problème de l'ouverture dans le cas 3D.

3.3.2.2 Correspondances 3D éparses

Dans certaines situations, mouvements de faible intensité ou haute fréquence d'acquisition par exemple, le champ de déplacement peut être estimé uniquement à l'aide des contraintes denses de flot normal (accompagnées d'une régularisation). Néanmoins dans un contexte plus général, nous devons considérer d'autres sources d'information. La mise en correspondance de points d'intérêts 3D permet de recueillir de l'information pour un jeu de points 3D à la surface de $\mathcal{S}^t$. Ces points 3D et leurs déplacements associés sont obtenus par la détection des points d'intérêts 3D sur $\mathcal{S}^t$ et $\mathcal{S}^{t+1}$, en leur créant un descripteur et en les appariant grâce à la comparaison de ces descripteurs.

Il existe différentes voies pour obtenir des correspondances 3D entre deux formes. Dans notre approche, nous utilisons MeshDOG pour détecter des points d'intérêts 3D et MeshHOG pour les décrire [Zaharescu 09]. Cette méthode définit et met en correspondance les extremas locaux de n'importe quelle fonction scalaire définie sur la surface. Dans le cas où l'information géométrique provient d'une image de profondeur, des méthodes de détection et d'appariement de point d'intérêts 2D, tels que ceux décrits dans la section suivante, peuvent être utilisés directement sur celle ci. D'autres techniques de détection et mise en correspondance peuvent être utilisées (telle que [Starck 07a]), tant qu'elles récupèrent un ensemble de correspondances robustes entre les deux surfaces.

L'avantage de l'utilisation de points d'intérêts 3D est que, contrairement au flot optique (décrit à la section 3.3.2.1), ils permettent de contraindre le mouvement même lors de grands déplacements dans l'espace 3D.

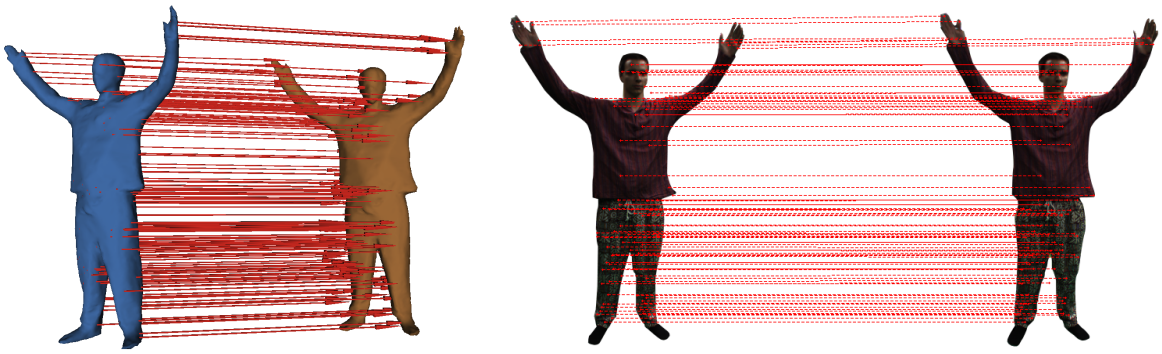
On obtient un ensemble épars de déplacements 3D V_m^t pour des points 3D $\mathbf{P}_m \in \mathcal{S}^t$ (voir figure 3.4a). Ces points forment un sous-ensemble discret de $\mathcal{S}^t$ appelé $\mathcal{S}_m^t$. La fonction d'erreur suivante décrit la proximité du champ de déplacement calculé V^t au champ de déplacement épars V_m^t :

$$\mathbf{E}_{3D} = \sum_{\mathcal{S}_m^t} \|V^t - V_m^t\|^2 \ . \tag{3.3}$$

3.3.2.3 Correspondances 2D éparses

Dans notre approche, nous considérons des correspondances 2D éparses entre les images $\mathbf{I}^t$ et $\mathbf{I}^{t+1}$. Comme dans le cas de la 3D, il y a différentes techniques existantes pour calculer des correspondances 2D entre une paire d'images, par exemple, SIFT [Lowe 04], SURF [Bay 06] ou Harris [Harris 88]. Sans pour autant perdre en généralité, nous nous appuyons sur le détecteur et descripteur SIFT. Il s'est avéré robuste et bien adapté dans notre cas, car invariant aux rotations et aux changements d'échelle.

Nous calculons des points d'intérêts sur les images $\mathbf{I}^t$ et $\mathbf{I}^{t+1}$. Nous mettons ensuite en correspondance les points d'intérêts ainsi obtenus. Cela nous donne



(a) Correspondances 3D–3D (b) Correspondances 2D–2D

FIGURE 3.4 – (a) Correspondances 3D éparses entre deux surfaces et (b) correspondances 2D éparses entre deux images.

un jeu de déplacements 2D épars v_s^t pour quelques points 2D $\mathbf{p}_s \in \Omega^t$ (voir figure 3.4b). Ces points forment un sous-ensemble de Ω^t appelé Ω_s^t. La fonction d'erreur suivante décrit la proximité du champ de déplacement 2D calculé v^t au champ de déplacement 2D épars v_s^t :

$$\mathbf{E}_{2D} = \sum_{\Omega_s^t} \|v^t - v_s^t\|^2 ,$$

ce qui est équivalent à :

$$\mathbf{E}_{2D} = \sum_{\Omega_s^t} \|J_{\mathbf{\Pi}} V^t - v_s^t\|^2 . \tag{3.4}$$

Il est important de noter que des correspondances 3D peuvent être obtenues à partir des points d'intérêts 2D en re-projetant les points détectés depuis $\mathcal{I}^t$ sur la surface $\mathcal{S}^t$ et ceux de $\mathcal{I}^{t+1}$ sur la surface $\mathcal{S}^{t+1}$. Cela fournit une liste de points d'intérêts 3D qui peuvent être mis en correspondance grâce à leurs descripteurs SIFT. Au lieu de réaliser cette mise en correspondance seulement dans l'espace d'une seule image, cela permet de prendre en compte les descripteurs issus de plusieurs images. Dans ce cas, la fonctionnelle d'erreur est similaire à $\mathbf{E}_{3D}$ décrite dans l'équation (3.3).

Bien que les points d'intérêts 3D soient plus robustes, en particulier aux occultations, et fournissent une meilleure information sur de longues séquences, ils présentent des désavantages, par rapport aux points d'intérêts 2D, pour l'estimation du flot de scène. Ils ne sont pas robustes aux changements de topologie et sont plus demandeurs en puissance de calcul, ce qui peut être crucial dans certaines applications. De plus, la surface S^{t+1} est forcément requise ce qui peut être problématique pour des applications interactives.

3.3.3 Contrainte de régularité

Les correspondances éparses 2D et 3D contraignent seulement le déplacement de la surface pour des points 3D spécifiques et pour leur re-projection dans les images. Pour trouver un champ de mouvement dense sur la surface, nous avons besoin de propager ces contraintes en utilisant un terme de régularisation.

En outre, comme mentionné précédemment, les contraintes denses de flot normal ne fournissent pas assez de contraintes pour estimer les déplacements 3D. En effet, il peut être démontré que les équations du flot normal pour des projections dans différentes images d'un même point 3D P contraignent de façon indépendante V^t à P, et ne résolvent donc que 2 degrés de liberté sur 3. Vedula *et al*. [Vedula 05] mentionnent deux stratégies de régularisation pour faire face à cette limitation. La régularisation peut être effectuée dans les plans images en estimant les flux optiques qui fournissent des contraintes plus complètes sur le flot de scène, ou elle peut être effectuée sur la surface 3D.

Puisque nous avons connaissance de la surface 3D et que les contraintes éparses 2D et 3D doivent être également intégrées, un choix naturel dans notre contexte est de régulariser en 3D. En plus, la régularisation dans l'espace image souffre d'artefacts et d'incohérences résultant des discontinuités de profondeur et des occultations qui contredisent l'hypothèse de lissage, alors qu'une telle hypothèse se justifie sur la surface 3D.

3.3.3.1 Modèle de déformation

Les hypothèses de régularité sur les champs de déplacement 3D de la surface limitent les déformations de cette surface à un niveau local. Elles définissent ainsi un modèle de déformation de la surface, par exemple, une rigidité locale. En 2D, de nombreuses méthodes de régularisation ont été proposées pour l'estimation du flot optique, elles se répartissent en 2 grandes catégories : les régularisations locales ou globales. Elles peuvent être étendues à la 3D.

Par exemple, la méthode 2D de Lucas et Kanade, qui utilise un voisinage local, a été appliquée en 3D par Devernay *et al*. [Devernay 06]. Toutefois, le modèle de déformation associé à la surface n'a pas de signification réelle, car les contraintes de déformation ne se propagent que localement, ce qui amène à des incohérences entre les voisins. D'autre part, la stratégie globale introduite par Horn et Schunck [Horn 81] est bien mieux adaptée à notre contexte. Bien que moins robuste au bruit que les méthodes locales telles que Lucas-Kanade, elle permet la propagation de contraintes éparses sur toute la surface. En outre, le modèle de déformation associé a prouvé son efficacité dans le domaine du graphisme [Sorkine 07].

L'extension du modèle de déformation d'Horn et Schunck à des points 3D est décrit par la fonction d'erreur suivante qui assure une rigidité locale du champ

de mouvement :

$$\mathbf{E}_{smooth} = \int_S \|\nabla V\|^2 \mathrm{d}\mathbf{P}\,. \tag{3.5}$$

3.4 Formulation et résolution

En regroupant tous les termes précédemment définis, notre fonctionelle d'énergie présentée dans l'équation (3.1) se réécrit comme ceci :

$$\mathbf{E} = \left[\lambda^2_{flow}\mathbf{E}_{flow} + \lambda^2_{3D}\mathbf{E}_{3D} + \lambda^2_{2D}\mathbf{E}_{2D} + \lambda^2_{smooth}\mathbf{E}_{smooth}\right]\,, \tag{3.6}$$

où les paramètres lambdas sont des valeurs scalaires servant à pondérer l'influence des différents termes. Minimiser cette équation peut se formuler de la manière suivante :

$$\begin{aligned} \arg\min_{V^t} \quad & \lambda^2_{flow}\delta_{\overline{F^t}}\|\nabla I^t.\left[J_\Pi V^t\right] + \frac{\mathrm{d}I^t}{\mathrm{d}t}\|^2 \\ + \quad & \lambda^2_{3D}\delta_{S^t_m}\|V^t - V^t_f\|^2 \\ + \quad & \lambda^2_{2D}\delta_{\Omega^t_s}J_{\mathbf{\Pi}}\left[V^t - V^t_s\right] \\ + \quad & \lambda^2_{smooth}\|\nabla V^t\|^2\,, \end{aligned} \tag{3.7}$$

où δ est le symbole de Kronecker indiquant que ce terme ne s'applique qu'à un sous ensemble de points et $\overline{F^t}$ indique les points de la surface pour lesquels aucune information venant des points d'intérêt, 2D ou 3D, n'est disponible.

En dérivant l'équation (3.7), nous obtenons pour chaque point $\mathbf{P}$ de la surface, l'équation d'Euler-Lagrange discrète, de la forme :

$$\mathbf{A_P}V_\mathbf{P} + \mathbf{b_P} - \Delta V_\mathbf{P} = 0\,, \tag{3.8}$$

où Δ est l'opérateur de Laplace-Beltrami normalisé sur la surface.

3.4.1 Système linéaire

Étant donné que l'équation (3.8) met en jeu un ensemble de contraintes linéaires pour chaque point 3D de la surface, une solution est donnée par la résolution du système linéaire suivant :

$$\begin{bmatrix}\mathbf{L}\\ \mathbf{A}\end{bmatrix} V^t + \begin{bmatrix}\mathbf{0}\\ \mathbf{b}\end{bmatrix} = \mathbf{0}\,, \tag{3.9}$$

où $\mathbf{L}$ est la matrice laplacienne du maillage de la surface construite de telle manière que $\mathbf{L}(i,j)$ pondère la relation entre les points i et j (les poids du Laplacien sont discutés dans la section 3.5.1). $\mathbf{A}$ et $\mathbf{b}$ stockent toutes les contraintes visuelles sur le déplacement venant des termes d'attache aux

données. Ce système linéaire est creux et peut être résolu en utilisant un solveur adapté tel que *Taucs*.

Il est intéressant de remarquer que cette formulation revisite le principe de déformation laplacienne de maillages de manière aussi rigide que possible (*as rigide as possible*) présenté dans la communauté du graphisme [Sorkine 07]. Bien que le schéma de déformation soit similaire, la différence se trouve dans les contraintes utilisées : des points ancre dans [Sorkine 07] et des contraintes visuelles dans notre cas. Dans les deux cas, il est clairement identifié que le modèle de déformation ne prend pas en compte les rotations de manière explicite. Bien que cela présente un désavantage certain dans le cas où l'on dispose d'un faible nombre de contraintes, comme c'est souvent le cas dans les applications du graphisme, la densité des contraintes que nous utilisons permet de retrouver ces rotations sans recourir à une résolution non linéaire.

L'équation (3.7) peut aussi être résolue de manière itérative en appliquant la méthode de Jacobi. De cette manière on résout le système indépendamment en chaque point en utilisant la solution courante du voisinage comme présenté dans la section suivante.

3.4.2 Résolution itérative

Inspiré par les travaux de Horn et Schunck [Horn 81], nous dérivons de l'équation (3.8) la résolution itérative suivante en chaque point de la scène :

$$\begin{aligned} v_x^{k+1} &= \overline{v}_x^k + A_x^x v_x^k + A_y^x v_y^k + A_z^x v_z^k - b_x \\ v_y^{k+1} &= \overline{v}_y^k + A_x^y v_x^k + A_y^y v_y^k + A_z^y v_z^k - b_y \\ v_z^{k+1} &= \overline{v}_z^k + A_x^z v_x^k + A_y^z v_y^k + A_z^z v_z^k - b_z \end{aligned} \tag{3.10}$$

où (v_x, v_y, v_z) et $(\overline{v}_x, \overline{v}_y, \overline{v}_z)$ représentent respectivement le déplacement propre et le déplacement moyen localement d'un point, l'indice k représente l'itération courante et les A_i^j et b_i sont les éléments de la matrice A et du vecteur b de l'équation (3.8).

Nous pouvons remarquer que les équations (3.10) sont indépendantes, à une itération donnée, pour chaque point de la surface. Ainsi l'implémentation de la résolution peut être massivement parallélisée. Dans ces équations, le déplacement local moyen pour un point 3D est donné par le voisinage de ce point en respectant la connectivité de la surface discrétisée et est pondéré exponentiellement en utilisant la taille des arêtes. Ainsi nous renforçons la relation qui lie deux points proches tout en empêchant les points aux frontières des objets d'être perturbés par des points lointains.

Cette formulation permet une approximation rapide du champ de déplacement. La rapidité et la précision de la résolution dépend fortement d'une bonne

initialisation. Comme mentionné dans [Horn 81], une bonne solution initiale peut être donnée par l'estimation obtenue à la trame précédente.

3.5 Détails d'implémentation

Cette section présente en détail les choix importants faits au moment de l'implémentation. Premièrement, nous discutons les poids qui déterminent, lors de la régularisation, l'influence du voisinage de la surface. Ensuite, nous présentons comment les grands et petits déplacements sont gérés séparément par le biais d'un algorithme en deux passes.

3.5.1 Poids laplaciens

Dans le terme de lissage de l'équation (3.7), l'opérateur de Laplace-Beltrami ∇^2 défini sur la surface de manière continue est approché par la matrice Laplacienne du graphe du maillage $\mathbf{L}$, c'est-à-dire $\nabla^2 V^t = \mathbf{L}V^t$, où :

$$\mathbf{L}(i,j) = \begin{cases} \deg(P_i) & \text{si } \ i = j, \\ -w_{ij} & \text{si } \ i \neq j \text{ et } P_i \text{ est adjacent à } P_j, \\ 0 & \text{sinon,} \end{cases}$$

où les w_{ij} correspondent aux poids des arêtes et $\deg(P_i) = \sum_{j \neq i} w_{ij}$. La matrice $\mathbf{L}$ peut être purement combinatoire, c'est-à-dire $w_{ij} \in \{0, 1\}$, ou contenir des poids $w_{ij} \geq 0$.

3.5.1.1 Dans le cas de maillages watertight

Dans le cas où nous disposons d'une surface complète, c'est-à-dire, close et sans trou. Nous pouvons pré-traiter ces données pour obtenir des maillages réguliers. Typiquement, les maillages issus d'algorithme de reconstruction type enveloppe visuelle présentent souvent de très grandes disparités dans la taille de leurs arêtes et une faible homogénéïté dans la répartition spatiale de leur sommets. Une simple étape de ré-échantillonnage permet d'obtenir un maillage bien plus propre pour les traitements suivants sans pour autant altérer les propriétés de forme du maillage initial. Lorsque l'échantillonnage du maillage est uniforme, les poids cotangents, souvent utilisés en graphisme [Wardetzky 07], permettent de garantir que la déformation appliquée à la surface maillée conservera au mieux les rigidités locales de la surface [Sorkine 07].

3.5.1.2 Dans le cas des nuages de points

Dans le cas des nuages de points, la connectivité du maillage vient de celle de l'image de profondeur, c'est-à-dire que les points voisins dans l'image sont

reliés sur la surface maillée par une arête. Cela aboutit à la construction d'un maillage cohérent, c'est-à-dire sans auto-intersection, mais avec potentiellement des arêtes de très grande taille correspondant aux discontinuités de la carte de profondeur. Pour gérer correctement ces discontinuités lors de la régularisation, nous proposons l'utilisation des poids suivants :

$$w_{ij} = -G(|P_i - P_j|, \sigma),$$

où G est un noyau gaussien, $|\cdot|$ est la distance euclidienne et σ l'écart type. En plus de fortement limiter la diffusion le long des grandes arêtes, les noyaux gaussiens sont aussi préconisés par Belkin *et al.* [Belkin 08] pour leur propriété de convergence vers le cas continu de l'opérateur de Laplace-Beltrami lorsque la résolution du maillage augmente.

3.5.2 Algorithme en deux passes

Dans l'équation (3.7), les paramètres λ_{2D}, λ_{3D}, λ_{flow} et λ_d indiquent le poids, respectivement, des points d'intérêts 2D et 3D, du flot normal 2D et du laplacien. Une forte valeur indique une influence plus importante pour le terme associé.

Dans notre contexte, de manière similaire à [Xu 10] en 2D, nous faisons confiance à nos points d'intérêts pour être robustes même lors de grands déplacements et nous sommes conscients que les contraintes de flot ne sont pas fiables quand la re-projection du déplacement est plus grande que quelques pixels dans les images. En conséquence, nous proposons une méthode itérative qui effectue deux minimisations successives de la fonctionnelle d'énergie avec deux jeux de paramètres différents. Les étapes de notre algorithme, illustré dans la figure 3.5, sont les suivantes :

1. Nous commençons par calculer les correspondances éparses 2D et 3D entre $\mathcal{S}^t$ et $\mathcal{S}^{t+1}$ et entre $\mathbf{I}^t$ et $\mathbf{I}^{t+1}$. Nous calculons également la matrice laplacienne $\mathbf{L}$ de notre surface discrétisée.
2. Nous résolvons l'équation (3.9), avec $\lambda_{flow} = 0$ et des valeurs plus importantes pour λ_{3D} et λ_{2D} que pour λ_{smooth}. Nous obtenons alors une première estimation de V^t dénotée V'^t qui récupère les grands déplacements de la surface.
3. Nous créons une surface déformée $\mathcal{S}'^t = \mathcal{S}^t + V'^t$ que nous projetons dans toutes les caméras en utilisant l'information de texture d'origine, venant de la projection de $\mathcal{I}^t$ sur $\mathcal{S}^t$. Nous obtenons alors un nouveau jeu d'images $\mathcal{I}'^t$.
4. Nous calculons alors la visibilité de la surface S'^t sur chaque caméra ainsi que les contraintes denses de flot normal entre $\mathcal{I}'^t$ et $\mathcal{I}^{t+1}$ pour chaque point visible de la surface. Nous obtenons donc plusieurs contraintes par points échantillonnés sur la surface.

5. Tout comme dans l'étape 2, nous résolvons l'équation (3.9) en utilisant le flot calculé dans l'étape 4 et les points d'intérêts 2D et 3D calculés précédemment dans l'étape 1. Ces derniers sont utilisés comme des points d'ancrage ayant une contrainte de déplacement nul. Pour cette étape, nous utilisons des valeurs fortes de λ_{3D} et λ_{2D} et des valeurs plus faibles pour λ_{flow} et λ_{smooth}. Nous obtenons alors le déplacement entre $\mathcal{S}'^t$ et $\mathcal{S}^{t+1}$ dénoté V''^t et donc également une version raffinée de $V^t = V'^t + V''^t$. Cette seconde minimisation permet de récupérer de plus petits déplacements, mieux contraints par les contraintes de flot.

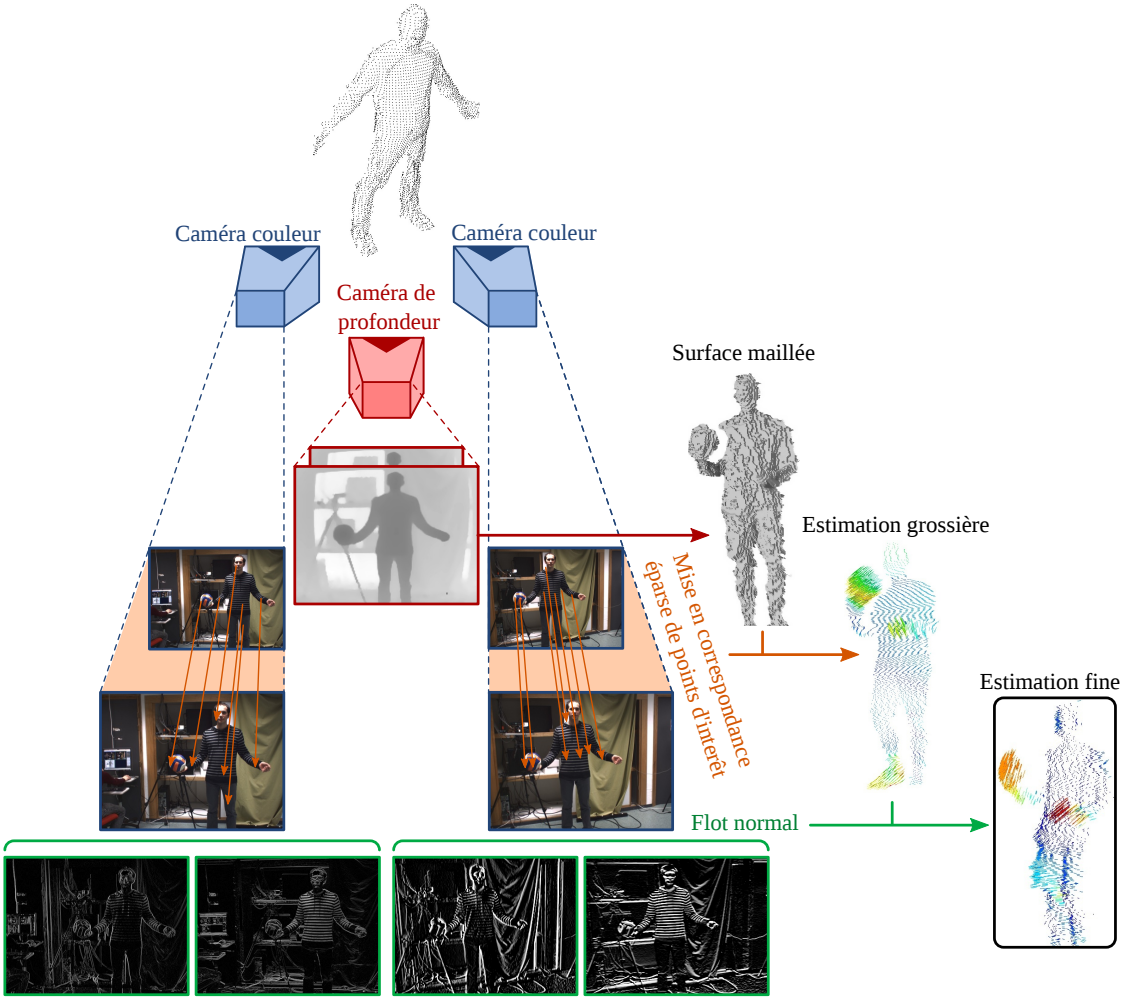


Figure 3.5 – Illustration des deux passes de notre algorithme dans le cas de deux caméras couleur et d'une caméra de profondeur.

Nous avons observé par nos résultats que, dans la pratique, notre approche peut gérer aussi bien de grands déplacements que des petits. Ceci grâce aux points d'intérêts qui gèrent bien les grands déplacements et au flot de normal qui récupère mieux les détails précis.

3.6 Evaluations pour les maillages watertight

Pour notre évaluation nous avons utilisé aussi bien des données synthétiques que des données réelles :

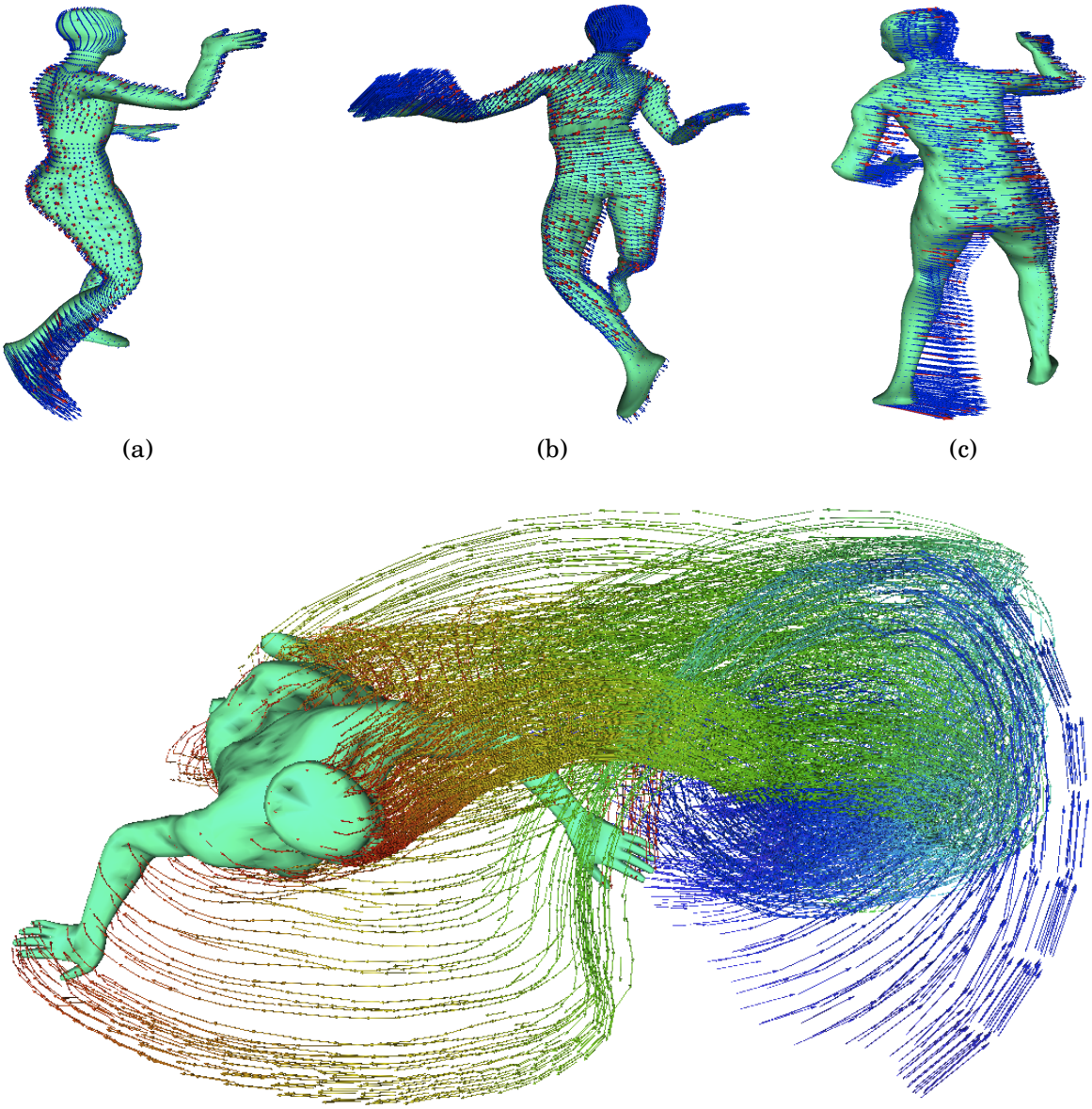
1. Les données synthétiques ont été obtenues grâce à un modèle humain articulé, déformé au cours du temps pour créer une séquence de danse. Nous avons rendu cette séquence dans dix caméras virtuelles de résolution 1 MPixels, réparties sur une sphère autour de la danseuse. Le modèle utilisé est un maillage triangulaire avec $7K$ sommets, déformé pour générer une séquence de 200 trames. La figure 3.6 montre trois images prises au même instant par trois caméras virtuelles différentes.
2. Les données réelles ont été récupérées à partir de banques de données accessibles au public. La première séquence a été prise à partir de 32 caméras 2 MPixels. Les maillages, obtenus avec EPVH, comportent $\sim 10K$ sommets. Nous avons également utilisé la séquence du *flashkick* de la base de données multi-vidéo *SurfCap* [Starck 07b] de l'Université de Surrey. Cette séquence a été enregistrée à partir de huit caméras 2 MPixels, et produit des maillages lisses de $\sim 140K$ sommets.

3.6.1 Évaluation quantitative sur des données synthétiques

Figure 3.6 – Images d'entrée de notre méthode.

Grâce à l'algorithme décrit dans la section 3.5.2, nous avons calculé les champs de mouvement sur la séquence synthétique de la danseuse. Les figures 3.7a, 3.7b et 3.7c montrent le champ de déplacement sur une des trames de la séquence. Les flèches rouges désignent les contraintes issues des points d'intérêts 3D et de la projection des points d'intérêts 2D, alors que les bleues désignent les vecteurs du champ de déplacement dense 3D.

La figure 3.7d montre le champ de déplacement accumulé sur plusieurs trames à partir d'une vue de dessus. Ce résultat peut être en quelque sorte



(a) (b) (c)

(d) Trace du mouvement

Figure 3.7 – Champ de déplacement sur plusieurs trames de notre séquence synthétique de danseuse (a), (b) et (c). Et historique du mouvement, sur plusieurs trames, vu du dessus (d) (les couleurs indiquent l'ancienneté du mouvement).

comparé à celui de Varanasi *et al*. [Varanasi 08], en effet leur méthode, permet de mettre en correspondance deux maillages consécutifs dans le temps, et est capable de fournir un champ de vitesse en conséquence de cet appariement.

Comme les maillages sont cohérents dans le temps, nous avons pu obtenir la réalité terrain et donc évaluer nos résultats quantitativement. La figure 3.8

montre l'erreur sur l'angle et la taille de chaque vecteur de mouvement après chacune des deux étapes de régularisation de notre algorithme. Nous pouvons voir les avantages de l'utilisation des contraintes de flot normal pour affiner le champ de déplacement (voir agrandissement sur la figure 3.9).

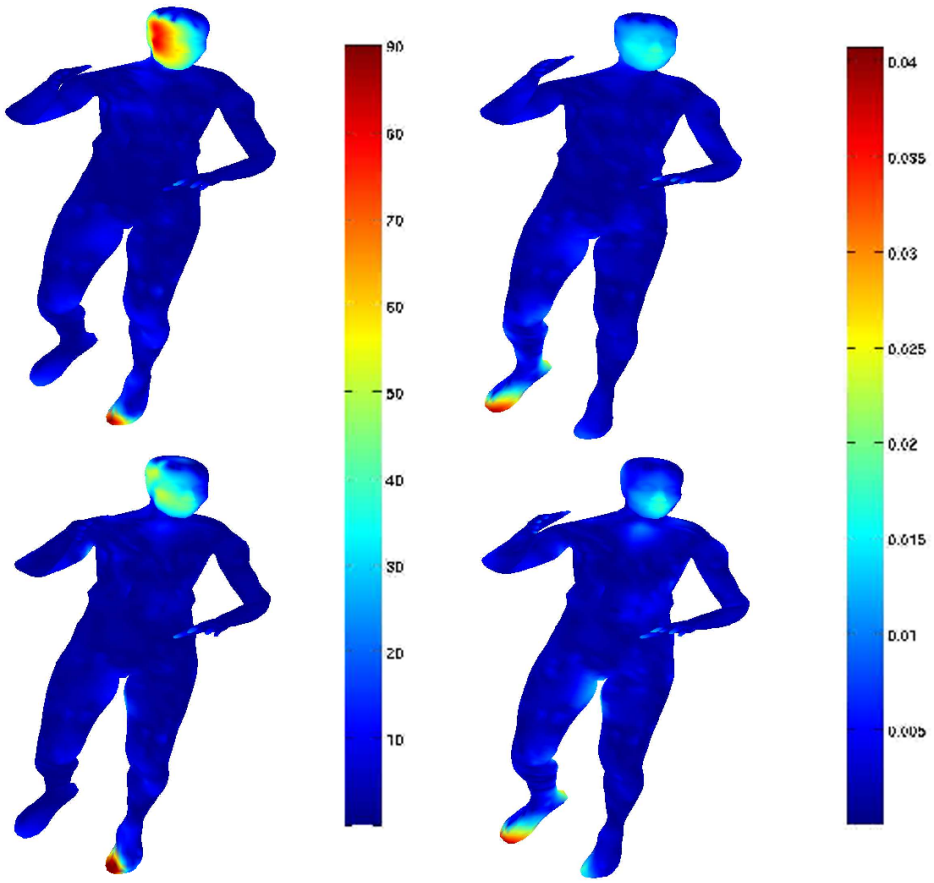
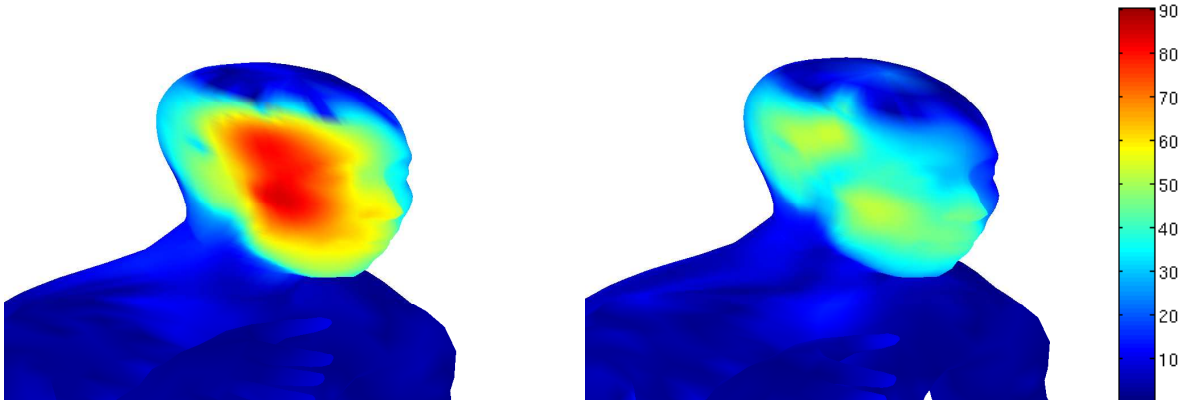


Figure 3.8 – Erreur sur le champ de déplacement : en angle en degré (gauche) et en norme en mètre (droite), après la première (haut) et la deuxième (bas) régularisation.

Les graphes de la figure 3.10 montrent des résultats quantitatifs sur deux séquences de synthèse. Chacune de ces séquences est composée de 34 flux de 15 trames montrant une sphère en mouvement.
Dans la première séquence la sphère subit un mouvement de translation pure et dans la seconde, le mouvement est une rotation par rapport au centre de la sphère. Dans les deux cas, l'intensité du mouvement est grandissante au fur et à mesure de la séquence générée ; avec par exemple jusqu'à 12°de rotation entre deux trames successives. Nous pouvons observer dans les graphes que la seconde passe de régularisation (en vert) permet d'obtenir grossièrement le même niveau d'amélioration des résultats par rapport à la première passe, en



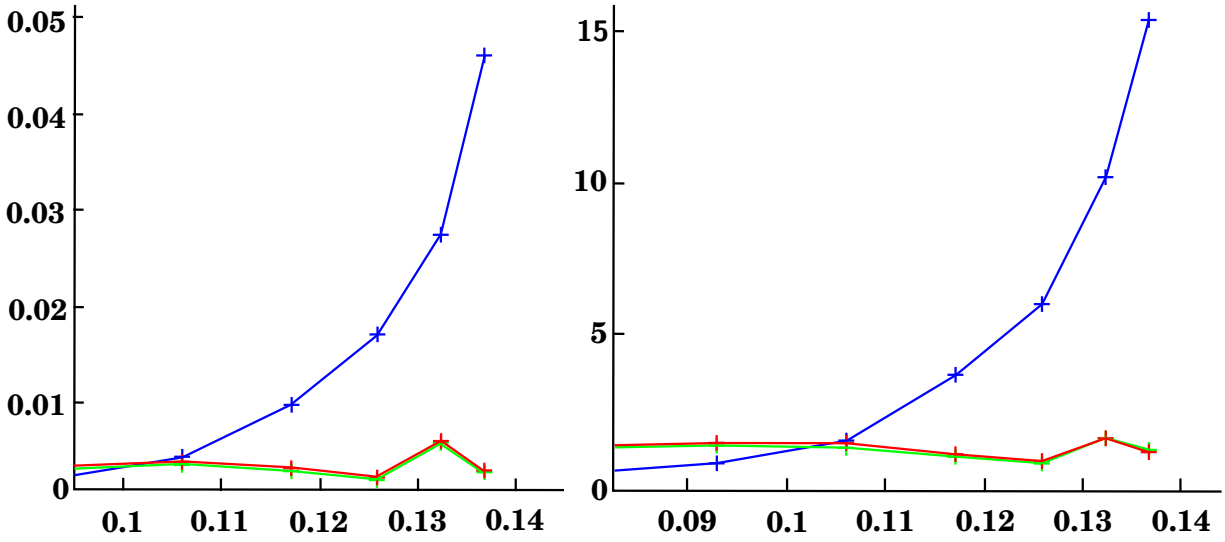
(a) Première régularisation (b) Seconde régularisation

Figure 3.9 – Agrandissement sur l'erreur en angle sur la face de la danseuse. Ces images montrent l'amélioration après la deuxième étape de régularisation qui aide pour récupérer les petits déplacements.

rouge ; et ce quel que soit l'amplitude du mouvement. Ceci est dû au fait que la première passe de notre méthode permet de retrouver les grands mouvements de telle manière que les déplacements résiduels se trouve à un niveau sous pixelique, niveau auquel l'information de flot devient cohérente et utilisable. Les graphes montrent aussi clairement que la qualité de nos résultas n'est pas dépendante de l'amplitude des mouvements, comme c'est le cas pour d'autre méthodes comme nous allons le voir plus loin.
Il est aussi intéressant de noter que l'apparence des courbes est strictement la même quel que soit le type de déplacement considéré : translation ou rotation. Pour cette raison, nous ne présentons que les résultats quantitatifs sur la séquence en translation ici, la figure 3.11 montre un exemple de mouvement estimé dans le cas d'une sphère en rotation pure. Cela signifie que, bien que notre modèle de déformation ne prenne pas directement en compte les rotations comme mentioné dans la section 3.4.1, nous sommes tout de même en mesure d'estimer correctement le mouvement de la scène.

3.6.2 Comparaison

Dans le but de fournir une comparaison de notre méthode avec l'état de l'art, nous avons implémenté l'approche proposée par Vedula *et al*. dans [Vedula 05]. Puisque cet article présente trois différentes approches pour calculer le flot de scène, nous avons choisi d'utiliser celle qui utilise les mêmes données en entrée que la nôtre, à savoir "*Multiple cameras, known Geometry*". Nous avons pour ce faire utilisé la denière implémentation OpenCV du calcul du flot optique à l'aide de l'algorithme de Lukas-Kanade [Lucas 81] avec les paramètres standards. Cette information de flot optique est ensuite intégrée comme décrit dans



(a) Erreur sur la norme (b) Erreur sur l'orientation

Figure 3.10 – (a) Norme (en mètres) et (b) angle (en degrés) d'erreur du déplacement éstimé en fonction de l'amplitude du mouvement réel de la surface (en mètres). En bleu : Vedula *et al.*, en rouge : la méthode proposée après la première régularisation, et en vert : après la seconde régularisation.

l'article pour en déduire le flot de scène. Les graphes de la figure 3.10 présentent les niveaux d'erreur obtenus utilisant la méthode de Vedula *et al.* (courbe bleue) en comparaison avec les résultats de notre méthode. Les séquences utilisées pour faire nos comparaisons sont les deux séquences décrites en fin de section précédente. Comme prévu, notre méthode présente des niveaux d'erreur bien plus convaincants dès que l'amplitude du mouvement dépasse la taille des pixels dans les images. Notre hypothèse pour expliquer ceci est que pour des déplacements sub-pixeliques, le calcul du flot optique peut-être très précis et ainsi il fournit une meilleure information que celle apportée par le flot normal uniquement.
Il est intéressant de noter que nos résultats sont fortement corrélés avec la résolution du modèle géométrique utilisé, c'est-à-dire la densité de sommets du maillage. Tandis que Vedula *et al.* font de la régularisation dans l'espace image, nous la faisons directement sur la surface. Ainsi nous pourrions augmenter légèrement la qualité de nos résultats en augmentant la résolution des maillages utilisés.

3.6.3 Expériences sur des données réelles

Notre première séquence réelle montre un sujet qui réalise des actions simples : il déplace ses deux mains à partir des hanches jusqu'au dessus de sa tête. Le sujet porte des vêtements amples et bien texturés ce qui permet de calculer un nombre élevé et fiable de correspondances 2D et 3D.

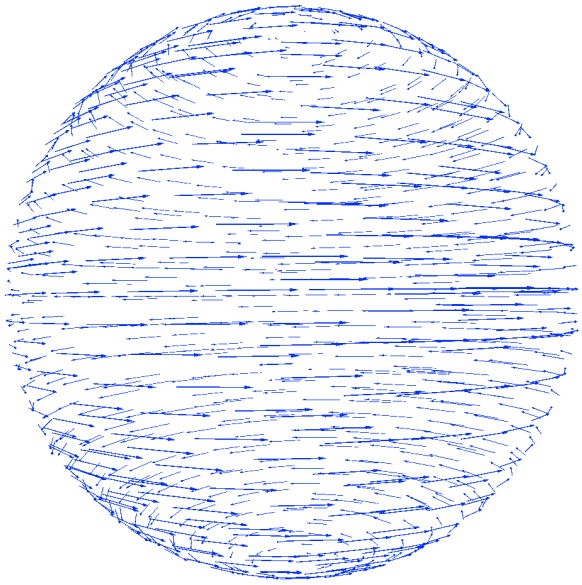


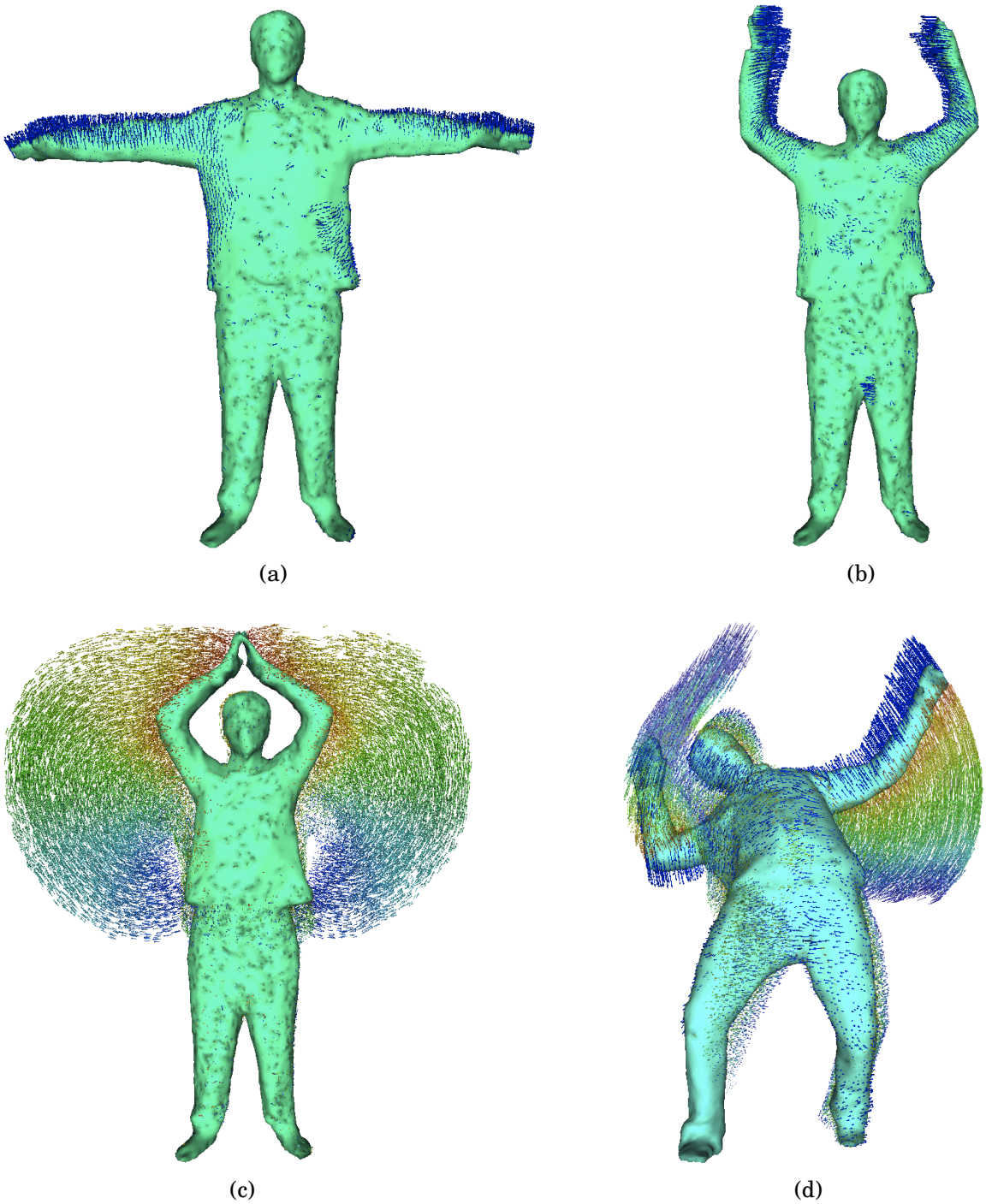
Figure 3.11 – Estimation du mouvement d'une sphère en rotation pure.

Les figures 3.12a, 3.12b et 3.12c montrent le mouvement instantané récupéré en utilisant notre méthode. Notez que nous ne calculons qu'un mouvement dense sur la surface et non une déformation du maillage. Ainsi, nous n'avons pas une connectivité constante dans le temps et nous ne pouvons pas effectuer le suivi des sommets du maillage sur toute la séquence. Par conséquent, l'évaluation quantitative des données n'est pas possible, mais la visualisation des résultats est très satisfaisante. La figure 3.12d montre le champ de déplacement accumulé sur toute la séquence.

Nous avons également calculé le champ de déplacement 3D sur la séquence du *flashkick* qui est très populaire. Dans cette séquence difficile, le sujet porte des vêtements amples avec peu d'information de texture. En outre, l'amplitude du mouvement est très élevée entre deux trames. Nous pouvons donc calculer moins de correspondances 2D et 3D. Elles sont pourtant nécessaires pour récupérer les grands déplacements.

Nous avons cependant réussi à calculer un champ de mouvement cohérent sur la plupart des trames (voir les figures 3.13a et 3.13b). Sur certaines trames, notre algorithme n'a pas trouvé de points d'intérêts sur les jambes ou les pieds du danseur, le champ de mouvement calculé à partir de ces indices montre bien la bonne direction, mais pas la bonne norme des vecteurs. Le manque de contraintes visuelles pour la première estimation du champ de mouvement ne permet pas de calculer certains déplacements complètement, le déplacement restant ne peut pas être récupéré entièrement avec les contraintes de flot normal. La figure 3.13c montre une trame problématique où le mouvement de la jambe droite du danseur n'est pas correctement calculé. Pour visualiser cette

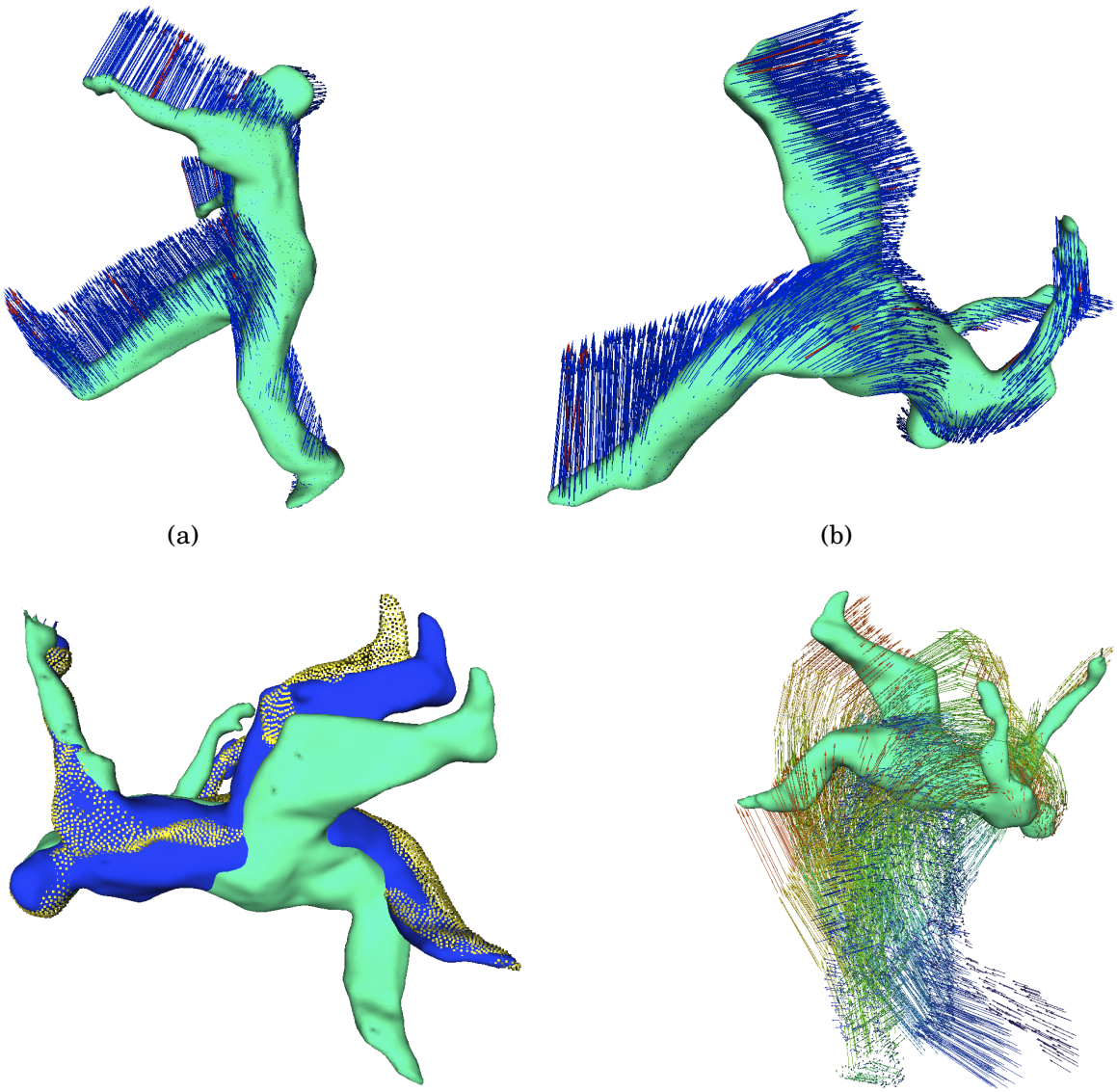
erreur, nous avons affiché les surfaces d'entrée au temps t et $t+1$ (respectivement cyan et bleu foncé), tandis que la surface déplacée avec le champ de mouvement calculé est indiquée par des points jaunes. Enfin, la figure 3.13d montre l'historique du mouvement sur quelques trames.



(a) (b)

(c) (d)

FIGURE 3.12 – (a) et (b) : champs de déplacement sur certaines trames de nos données réelles. (c) et (d) : historiques du mouvement accumulés sur toute la séquence (les couleurs indiquent l'ancienneté du mouvement).



(a) (b)

(c) Estimation incomplète du mouvement (d)

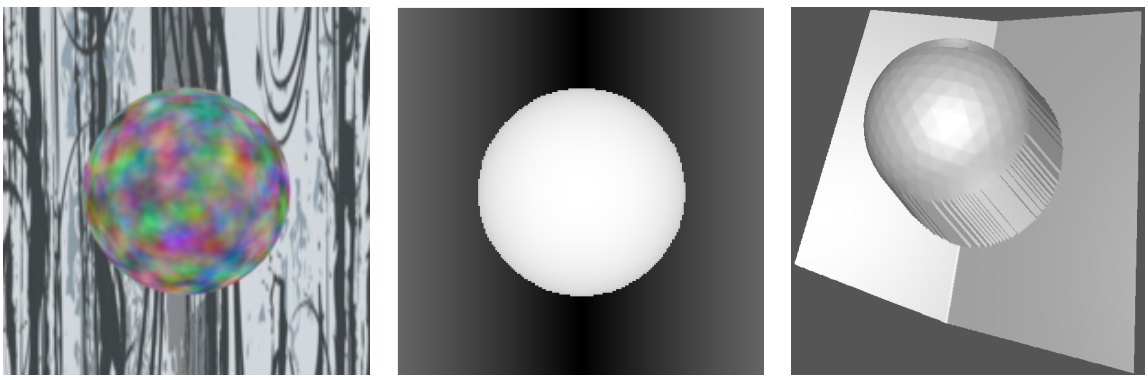
Figure 3.13 – Champs de déplacement sur plusieurs trames de la séquence du flashkick (a) et (b), mouvement partiellement calculé (c), et historique du mouvement sur toute la séquence (d) (les couleurs indiquent l'ancienneté).

3.7 Evaluations pour les cartes de profondeur

Pour procéder à l'évaluation de la méthode proposée, nous avons utilisé plusieurs séquences dans différentes conditions. Pour commencer, nous avons créé des données de synthèse pour permettre une évaluation quantitative. Dans un second temps nous avons procédé à l'acquisition et au traitement de données réelles à l'aide de deux configurations différentes, avec une ou plusieurs caméras couleur. Les différentes configurations ainsi que les résultats obtenus sont détaillés dans cette section.

3.7.1 Données de synthèse

Les données de synthèse représentent une sphère en mouvement devant deux plans également en déplacement. Cette scène est ensuite projetée dans deux caméras de synthèse de 1M pixels. Nous utilisons le *depth buffer* d'une de ces deux caméras virtuelles pour obtenir la carte de profondeur de la scène (voir figure 3.14a et 3.14b). Cette carte de profondeur a été rééchantillonnée à une résolution de 200×200 et utilisée pour créer un maillage (voir figure 3.14c). Dans la séquence générée, la sphère se déplace en s'éloignant de la caméra, tandis que l'un des plans se déplace vers le haut et l'autre vers le bas. Il est important de noter que l'extension de la méthode proposée à $N > 1$ caméras couleur ne change rien à la formulation, cela ne fait qu'empiler plus de contraintes dans l'équation (3.9).

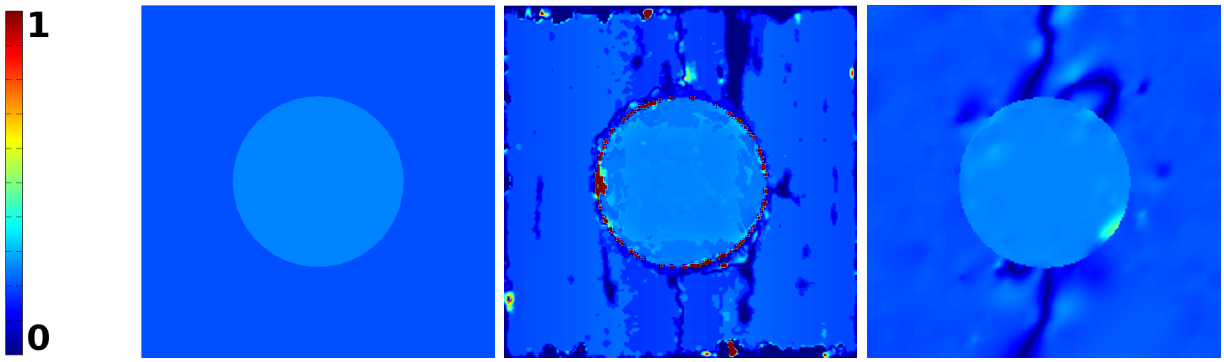


(a) Image couleur (b) Carte de profondeur (c) Maillage associé

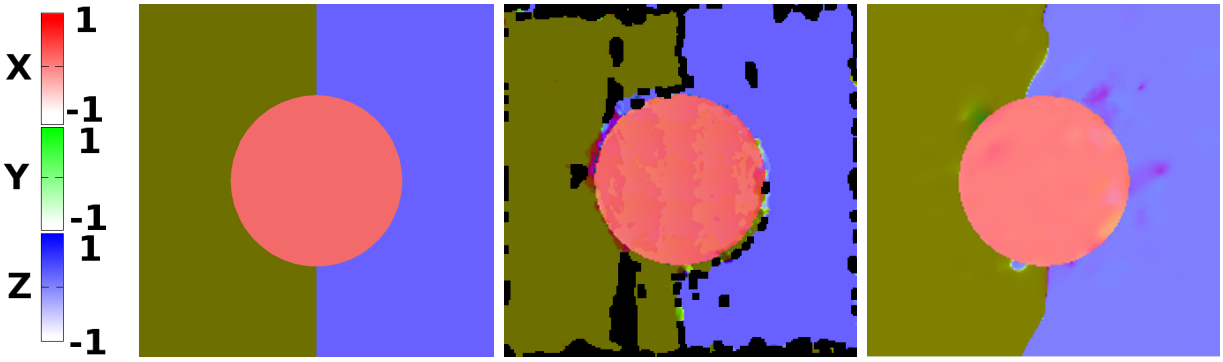
Figure 3.14 – Données de synthèse : image couleur (a), carte de profondeur (b) et le maillage inféré (c).

Nous avons comparé notre approche à la méthode de référence présentée dans [Vedula 05] dans le cas "*Single camera, known scene geometry*". Cette méthode requiert les mêmes données en entrée que la nôtre et est aussi extensible au cas multi-caméra.

Les résultats obtenus sont présentés dans la figure 3.15 où les normes et orientations des déplacements 3D sont représentés depuis le point de vue de la caméra avec un code couleur. La figure 3.16 montre l'erreur en chaque point de la surface maillée, tandis que la table 3.1 présente une comparaison numérique.



(a) Norme du déplacement 3D (code couleur)



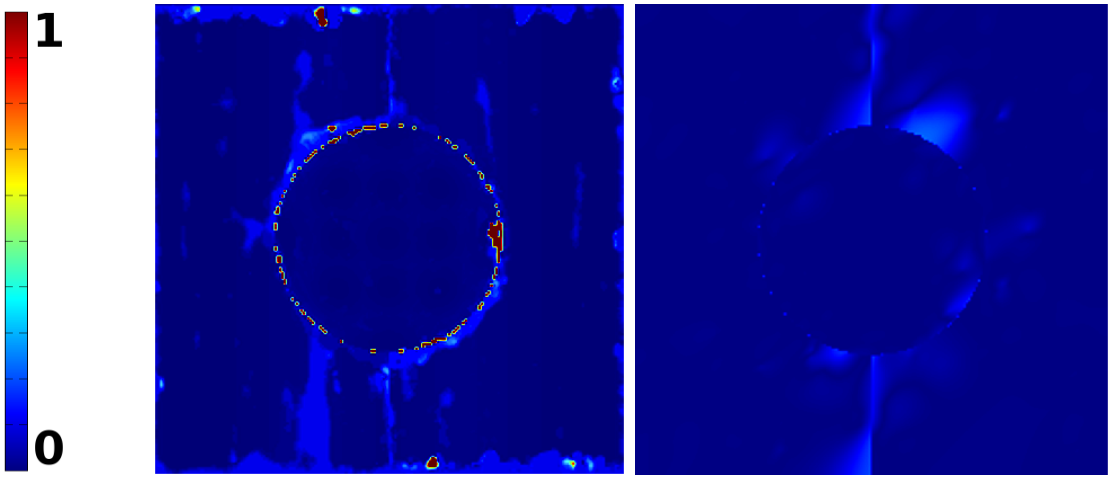
(b) Direction du déplacement 3D (code couleur)

Figure 3.15 – Résultats sur des données de synthèse et comparaison entre vérité terrain (gauche), la méthode de Vedula [Vedula 05] (centre) et notre méthode (droite).

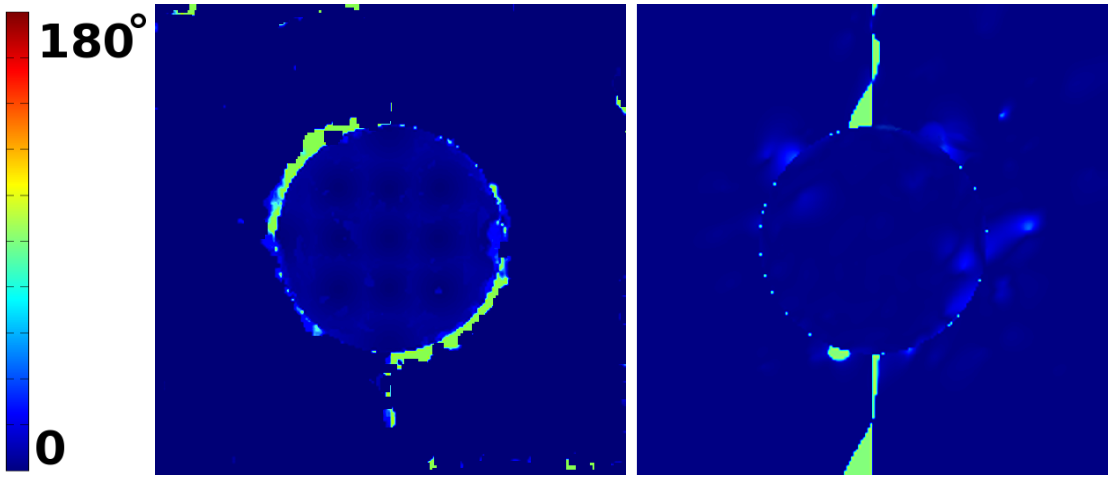
	Vedula [Vedula 05]		Notre méthode	
Erreur	Moyenne	Médiane	Moyenne	Médiane
Norme	33%	7.27%	**8.68%**	**2.33%**
Angle	8.6°	**0.10°**	**2.7°**	0.12°

Table 3.1 – Erreurs numériques sur des données de synthèse avec comparaison entre la méthode de Vedula [Vedula 05] et la méthode proposée.

Les résultats obtenus montrent que la méthode proposée est capable de gérer correctement les discontinuités de la carte de profondeur entre la sphère et les plans. Néanmoins, à l'endroit où les deux plans se croisent, il y a une ambiguïté qui conduit à supposer que les deux plans sont connectés sur la surface maillée.



(a) Erreur sur la norme du déplacement 3D



(b) Erreur sur l'orientation du déplacement 3D

FIGURE 3.16 – Erreur sur des données de synthèse avec comparaison entre la méthode de Vedula [Vedula 05] (gauche) et la méthode proposée (droite).

Ainsi la régularisation a du mal à évaluer correctement les déplacements dans cette zone. Ce comportement était attendu dans la mesure où notre hypothèse de régularisation en 3D est violée à la jonction des deux plans. C'est-à-dire que les deux plans se touchent mais ont des déplacements complétement différents. Cet exemple met ainsi en avant la force et la faiblesse de notre méthode.

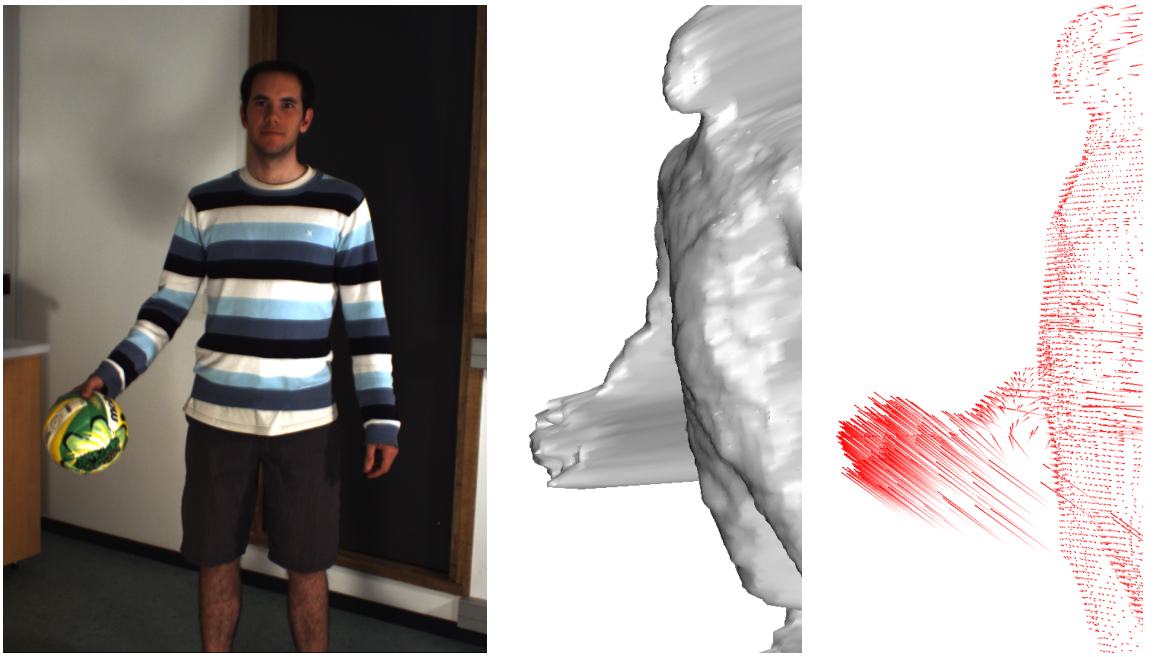
Les expérimentations menées montrent qu'avec de bonnes textures et des données de synthèse, les contraintes du flot normal n'aident pas réellement à améliorer les résultats puisque les déformations sont strictement rigides et beaucoup de points d'intérêt sont détectés et appariés correctement, ce qui suffit à retrouver les mouvements dans le cas de scènes basiques. L'ajout de caméras couleur supplémentaires n'améliore pas significativement l'estimation des déplacements puisque les points d'intérêt détectés tendent à être les mêmes d'une

image à l'autre dans le cas d'un faible parallaxe.

3.7.2 Données réelles

Nous avons aussi procédé à des expérimentations sur des données réelles acquises avec deux systèmes différents : (1) un système multi-caméra composé d'une caméra temps de vol Swiss Ranger SR4000 de résolution 176×144 accompagnée de deux caméras couleur de 2M pixels, et (2) une caméra Kinect de Microsoft capable de fournir un flux d'images couleur, chacune alignée sur une carte de profondeur de résolution 640×480. Le système multi-caméra avec la caméra temps de vol a été calibré en utilisant les travaux présentés dans [Hansard 11].

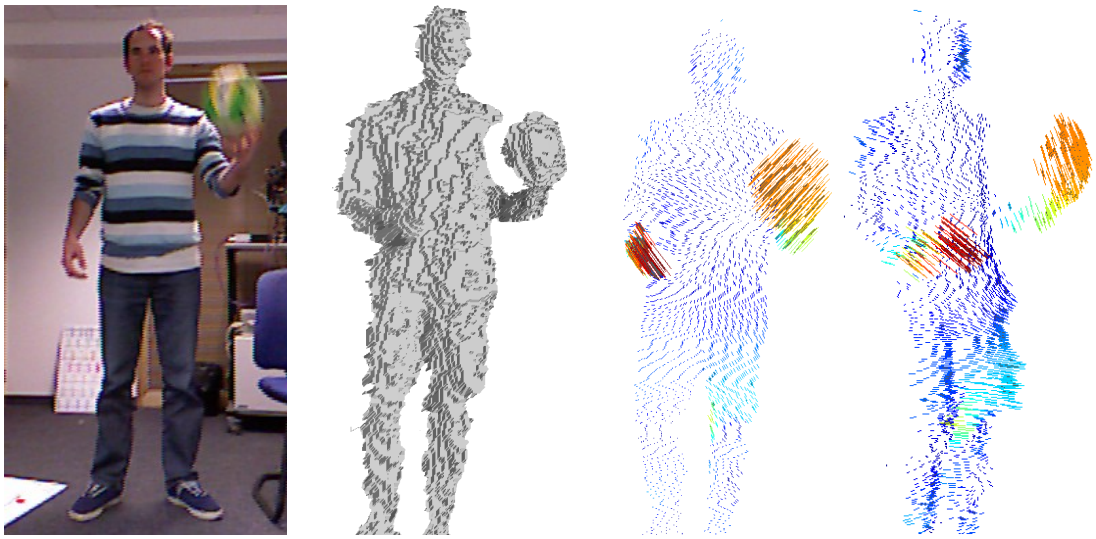
Pour tester efficacement notre méthode nous avons acquis avec les deux systèmes une scène identique dans laquelle un homme se tient debout dans une pièce et joue avec une balle, la faisant sauter d'une main à l'autre. Cette scène présente à la fois de grandes discontinuités dans la carte de profondeur et des déplacements larges et rapides.



(a) Image couleur (b) Maillage (c) Déplacements

Figure 3.17 – Données en entrée : une des deux images couleur (gauche) et la surface calculée (centre). Résultat : le champ de déplacement 3D obtenu sur les données de la caméra temps de vol (droite).

Les résultats présentés sur les figures 3.17 et 3.18 démontrent l'intérêt ainsi que la faisabilité de notre méthode sur des données réelles. Les codes couleurs des figures 3.17c et 3.18c indiquent respectivement l'orientation et l'intensité



(a) Image couleur (b) Maillage (c) Champs de déplacements

Figure 3.18 – Données en entrée : l'image couleur (gauche) et la surface calculée (centre). Résultat : le champ de déplacement 3D obtenu sur les données de la caméra Kinect (droite), la couleur encode la norme des vecteurs.

du déplacement. Le champ de déplacement estimé est cohérent avec les actions exécutées par la personne. L'utilisation de deux caméras couleur dans le cas de la caméra temps de vol permet d'obtenir un résultat satisfaisant bien que les données géométriques soient très bruitées. En ce qui concerne la caméra Kinect, la résolution des données acquises induit un maillage de haute densité qui accentue la complexité du système linéaire. Dans ce cas, une implémentation parallèle peut permettre de balancer la complexité des données.

3.8 Conclusion et discussion

La contribution de ce travail est double : premièrement, nous avons présenté une méthode unifiée qui permet de combiner des informations photométriques pour calculer le mouvement d'une surface, d'autre part, nous avons introduit une méthode itérative qui permet de gérer de grands déplacements tout en récupérant les petits détails.

Comme le montrent les résultats, notre méthode est assez robuste et polyvalente. En effet elle peut s'adapter sans surcoût pour l'utilisateur sur des systèmes multi-caméra de nature différente. Nos expériences vont dans ce sens en démontrant l'adaptabilité de la méthode présentée à des systèmes contenant de une à 32 caméras couleur, avec ou sans capteur de profondeur. Néanmoins, nos expériences ont mis en évidence certaines faiblesses potentielles.

Comme nous le pensions, s'appuyer sur des caractéristiques visuelles impose d'avoir une bonne information de texture dans les images. Notre méthode pourrait être améliorée par l'ajout d'autres contraintes, par exemple, un critère de cohérence photométrique tel que celui utilisé par Pons *et al*. dans [Pons 05]. Il serait interessant d'étudier une meilleure manière d'intégrer l'information de flot de normal. Pour le moment, nous n'utilisons cette information que pour retrouver les détails du champ de déplacements. Par la mise en oeuvre d'une approche multi-échelle, nous pourrions intégrer cette contrainte même dans le cas de déplacements conséquents. Nous avons rejeté les approches multi-échelle qui proposent un lissage dans l'espace image sous prétexte qu'elles souffrent de sur-lissage pour les pixels la frontière des objets. Néanmoins, puisque nous disposons de la géométrie de la scène, il est possible d'imaginer effectuer un lissage dans l'espace image qui tient compte de la géométrie de la scène (voir à ce sujet l'annexe A).

Nous avons aussi mis en avant dans nos expérimentations (section 3.7.1) un cas limite pour notre méthode. Si deux objets distincts et aux déplacements différents sont proches au point d'être assimilés à la même forme dans la reconstruction de la scène ; alors notre hypothèse de régularisation est violée et la précision de nos résultats chute. Il s'agit d'un cas marginal mais qui présente tout de même une limitation en l'état actuel de nos travaux. Pour y remédier, il faudrait avoir accès à une meilleure information de géométrie. Si nous pouvions délimiter physiquement les objets alors ce problème de sur-lissage serait résolu. Les travaux présentés au chapitre suivant proposent un début de réponse.

La méthode que nous proposons donne de toute façon des informations utiles et fiables sur les propriétés intrinsèques d'une séquence 4D. La connaissance du déplacement instantané peut être utilisée comme donnée d'entrée pour de nombreuses tâches en vision par ordinateur, telles que le suivi de surface, le transfert de mouvement ou la segmentation de maillages.

Même si nous n'avons pas mis l'accent sur les performances de calcul pour notre première implémentation, nous sommes certains que la plupart des calculs pourraient s'exécuter en parallèle. En effet, l'extraction des points d'intérêts 2D, ainsi que le calcul des contraintes de flot normal, sont indépendants par caméra. De plus, des implémentations temps-réel de SIFT et des méthodes de flux optique existent déjà. La propriété linéaire de la régularisation permet de s'attendre à une exécution en temps-réel également.

Pour le moment, les systèmes multi-caméra qui permettent de faire de la reconstruction 3D en temps-réel ne calculent pas vraiment le mouvement associé au modèle. Ces informations supplémentaires pourraient améliorer considérablement les applications interactives telles que les interactions basées sur des collisions entre le sujet reconstruit et toutes sortes d'objets virtuels. Il pourrait également être utilisé pour la reconnaissance d'actions.

Deuxième partie

Modèles progressifs de forme

Chapitre 4

Modèles progressifs de forme

Sommaire

Résumé

Dans cette nouvelle partie nous nous concentrons sur un problème récurrent des systèmes d'acquisition 4D : l'apprentissage de la géométrie et de la topologie d'une scène déformable à partir d'une séquence temporelle de maillages. Il s'agit d'une étape fondamentale dans le traitement de scènes naturelles et dynamiques. Tandis que de nombreux travaux ont été menés pour la reconstruction de scènes statiques, assez peu considèrent le cas de scènes dynamiques dont la topologie évolue et sans connaissances *a priori* . Dans cette situation, une simple observation à un unique instant de temps n'est souvent pas suffisante pour retrouver entièrement l'information de topologie propre à la scène observée. Il semble ainsi évident que les indices sur la forme doivent être accumulés intelligemment sur une séquence complète afin d'acquerir une information aussi complète que possible sur la topologie de la scène et permettre l'apprentissage d'un modèle cohérent à la fois spatialement et temporellement. À notre connaissance cela semble un problème nouveau pour lequel aucune solution formelle n'a été proposée. Nous formulons dans cette thèse un principe de solution basé

sur l'hypothèse que les objets composant la scène observée possèdent une topologie fixe. À partir de cette hypothèse de base nous pouvons progressivement apprendre la topologie et en parallèle capturer les déformations d'une scène dynamique. Les travaux présentés dans cette partie visent à retrouver une information de basse fréquence sur la géométrie de la scène. En l'état actuel, la méthode que nous proposons ne peut pas être directement utilisée pour accumuler les informations de bas niveau (détails de la surface) sur une séquence de maillages.

4.1 Contexte et motivations

À moins de disposer *a priori* d'informations fiables et complètes sur la forme de la scène, par exemple un modèle obtenu avec un scanner laser [Anguelov 05, de Aguiar 08], un système d'acquisition 4D produit une séquence temporelle de modèles 3D sous la forme de maillages dont la connectivité, et potentiellement la topologie, diffèrent. Un problème clé dans le processus d'acquisition est donc d'obtenir un modèle qui soit cohérent avec toutes les observations, permettant ainsi le suivi d'objet ou la mise en place d'applications basées sur les mouvements.

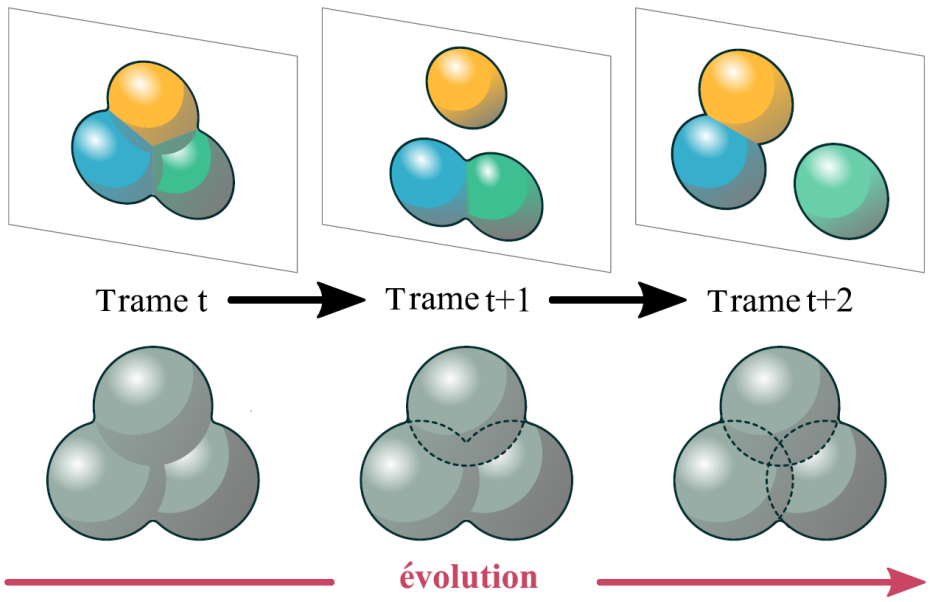


Figure 4.1 – Haut, séquence temporelle d'observations d'une scène contenant trois objets qui interragissent. Bas, évolution associée du modèle appris au fil du temps. Nous remarquons qu'aucune des données en entrée ne contient la totalité de l'information topologique.

Dans le cas général, il est souvent impossible de supposer qu'un tel modèle existe pour des scènes dynamiques dans la mesure où elles peuvent être composées de divers objets distincts qui interragissent. De plus, même un unique objet peut être difficile à modéliser *a priori* à l'aide d'un système d'acquisition statique. Par exemple, il est souvent compliqué, voire impossible, de modéliser un animal à l'aide d'un scanner laser. Par ailleurs, considérer une des trames de la séquence séléctionnée à la main comme modèle de référence, comme par exemple [Cagniart 10], ne résout pas le problème non plus puisqu'il est assez rare qu'une trame seule contienne la totalité de l'information topologique d'une scène ou même d'un objet, voir figure 4.1. Dans ce chapitre nous considérons tous ces points et proposons une méthode à la fois simple et efficace qui permet d'accumuler l'information de topologie contenue dans une séquence de maillages 3D et qui progressivement apprend et suit un modèle de la vraie forme observée.

Récemment, de grands efforts ont été portés sur le problème de la modélisation précise de la géométrie d'un modèle à partir d'un ensemble statique d'images [Seitz 06] ou sur le suivi d'objets à partir de plusieurs flux vidéos et d'un modèle de référence fourni, par exemple [Vlasic 08, de Aguiar 08, Cagniart 10], ou encore la mise en correspondance de formes, par exemple [Bronstein 07], avec ou sans changements de topologie, par exemple [Sharma 11]. Des travaux plus récents permettent d'améliorer des modèles de formes en utilisant des séquences d'observations temporelles, par exemple [Popa 10]. Néanmoins le problème de la construction d'un modèle de forme cohérent pour des observations temporelles où des changements de topologie interviennent est toujours ouvert.

Dans le but d'aprendre et suivre un modèle de référence à partir d'une séquence temporelle de maillages 3D, nous introduisons des *modèles progressifs de forme*. Basé sur une stratégie d'agrandissement, ces modèles évoluent à la fois en terme de topologie mais aussi de géométrie tout au long de la sequence. Dans cette étude, la topologie fait référence aux propriétés telles que le nombre d'objets et les trous présents dans ces objets ; la géométrie quant à elle fait référence à la position dans l'espace des points qui composent ces objets. La méthode d'estimation proposée alterne entre l'évolution du modèle de référence pour la topologie et le suivi de forme pour la géométrie. Par rapport au premier aspect, nous faisons l'hypothèse que la quantité d'information topologique contenue dans le modèle progressif de forme ne peut qu'augmenter au fil de son apparition dans les observations successives de la séquence. Cette augmentation constante est modérée par un facteur de résistance au bruit qui empêche d'inclure dans le modèle des informations certes nouvelles, mais éronnées. Cette hypothèse est basée sur le fait que dans un contexte réel, la plupart des objets ont une topologie fixe qui est préservée lors des déformations.

Ainsi notre modèle d'évolution ne permet pas la fusion de deux objets distincts, ce qui reviendrait à diminuer la quantité d'information topologique contenue dans le modèle. À partir de cette hypothèse nous proposons une solution cohérente au problème de l'apprentissage de la géométrie et de la topologie pour des séquences temporelles de maillages 3D, et ainsi nous permettons la capture et l'analyse de scènes de plus en plus complexes.

Ce chapitre se présente de la manière suivante : Pour commencer, nous faisons un état de l'art dans la section 4.2 puis le coeur de notre approche est détaillée dans la section 4.3. Nous présentons et explicitons ensuite les résultats obtenus lors de nos expérimentations dans la section 4.4 avant de conclure ce chapitre dans la section 4.5.

4.2 État de l'art

Dans le passé, plusieurs travaux ont considéré l'estimation des déformations et les évolutions de formes d'objets à travers l'étude de séquences temporelles de données telles que des images couleur et des cartes de profondeur. Ces approches peuvent être grossièrement classées par rapport à la quantité d'informations préalables dont elles ont besoin. Premièrement, plusieurs méthodes supposent un modèle déjà connu de la scène, sous la forme d'une surface ou d'un volume tel que dans [de Aguiar 08, Furukawa 09, Li 09]. D'autres remplacent ce modèle de forme par un modèle de structure tel qu'un squelette articulé [Vlasic 08]. Bien que ces approches peuvent se contenter des observations partielles uniquement, une telle connaissance du modèle *a priori* est souvent une contrainte impossible à satisfaire, en particulier lorsque la scène traitée est complexe. Une autre méthode qui trouve sa place ici est de considérer une des trames de la séquence sélectionnée par l'utilisateur comme le modèle, comme c'est le cas dans [Cagniart 10]. Bien que cette hypothèse puisse apporter une solution convenable, au prix d'un effort suplémentaire de l'utilisateur, il n'existe aucune garantie quant à la cohérence spatiale et temporelle du modèle choisi. De plus, il est fort probable qu'aucune trame en particulier ne contiennent toute l'information de forme et de topologie de la scène considérée. Dans ce cas, la dernière méthode mentionnée ne peut tout simplement pas être mis en œuvre. Dans ce chapitre, nous proposons une méthode qui répond à toutes ces limitations.

D'autres méthodes ont été proposées qui nécessitent moins de connaissances *a priori* sur la scène. Par exemple, [Mitra 07] mets en correspondances de manière globale une séquence de nuages de points en faisant l'hypothèse de champs de déplacements lisses et d'un échantillonnage temporel et spatial dense. [Zheng 10] met aussi en correspondance une séquences de nuages de

points représentant des objets déformables en faisant l'hypothèse que ces derniers présentent des structures de squelettes qui sont consistantes dans le temps et qui peuvent ainsi être alignées. Ces deux approches répondent au problème de la mise en correspondance temporelle, mais elle font l'hypothèse implicite que la topologie des observations est simple et statique. Il est intéressant de mentionner aussi les travaux présentés dans [Sharf 08] et [Li 11]. Dans ces deux approches, les informations sont accumulées sur une fenêtre temporelle dans le but d'améliorer la reconstruction statique de chaque trame. Bien que ces approches améliorent la reconstruction individuelle de chaque trame, elles ne permettent pas d'apprendre un modèle global de la scène. Une autre différence importante est que notre méthode ne se contente pas des observations disponibles sur une courte fenêtre temporelle, elle considère la séquence dans sa totalité pour construire un modèle aussi complet et précis que possible.

Plus proche des travaux présentés dans ce chapitre, il existe quelques approches qui proposent d'apprendre un modèle de forme en se basant sur des évolutions temporelles. Par exemple, [Wand 07] introduit l'estimation jointe des déformations et de la forme d'une scène. [Popa 10] propose une stratégie hiérarchique interessante qui améliore progressivement un modèle de forme en accumulant des informations géométriques basées sur des différences trame-à-trame en construisant un arbre binaire sur la séquence traitée. Bien qu'elles autorisent des changements de topologie du modèle, ces deux approches sont principalement adaptées pour combler des trous dans le modèle issu des données incomplètes venant de cartes de profondeur. Notre champ d'action est différent puisque nous considérons en entrée, des données de forme complètes, c'est-à-dire des maillages, au lieu de nuages de points. Une différence cruciale entre ces méthodes et celle que nous proposons est que nous ne faisons pas d'estimations des changements de topologie, ces derniers sont appris à partir de leurs observations dans les données en entrée.

La contribution de la méthode présentée ici, par rapport aux approches citées est double. Premièrement, nous introduisons la notion de modèle progressif de forme qui construit de manière incrémentale un modèle d'une scène dynamique dont les objets qui la composent peuvent se déplacer et se déformer. Ce modèle est directement appris à partir d'une séquence de maillages reconstruits indépendamment à chaque trame. La seconde contribution est une méthode d'amélioration de modèle basée sur une formulation théorique cohérente du traitement des changements de topologie **observés** dans les données.

4.3 Apprentissage d'un modèle

4.3.1 Principes généraux et hypothèse

L'objectif est de retrouver à la fois la géométrie et la topologie des objets qui composent une scène dynamique observée durant un intervalle de temps. Notre approche prend comme données en entrée une séquence temporelle de n maillages triangulaires inconsistants et estime un modèle de la scène cohérent à la fois spatialement et temporellement. Ce modèle final est lui aussi sous la forme d'un maillage triangulaire. Le modèle progressif de forme est initialisé à l'aide de la première observation disponible, il est ensuite déformé et augmenté séquentiellement au fur et à mesure que de nouvelles informations sont apportées par les observations suivantes. À chaque nouvelle trame, la méthode que nous proposons fait évoluer le modèle pour qu'il s'aligne spatialement aux observations courantes tout en y incluant d'éventuelles informations topologiques nouvelles. Comme expliqué précédemment, nous faisons l'hypothèse que la topologie des objets de la scène est fixe, c'est-à-dire que le nombre d'objets et de trous n'évolue pas. Il est intéressant de remarquer que la topologie finale de notre modèle progressif est la meilleure possible mais pas forceément la *vraie* ; à moins que toute l'information ne soit contenue dans les observations.

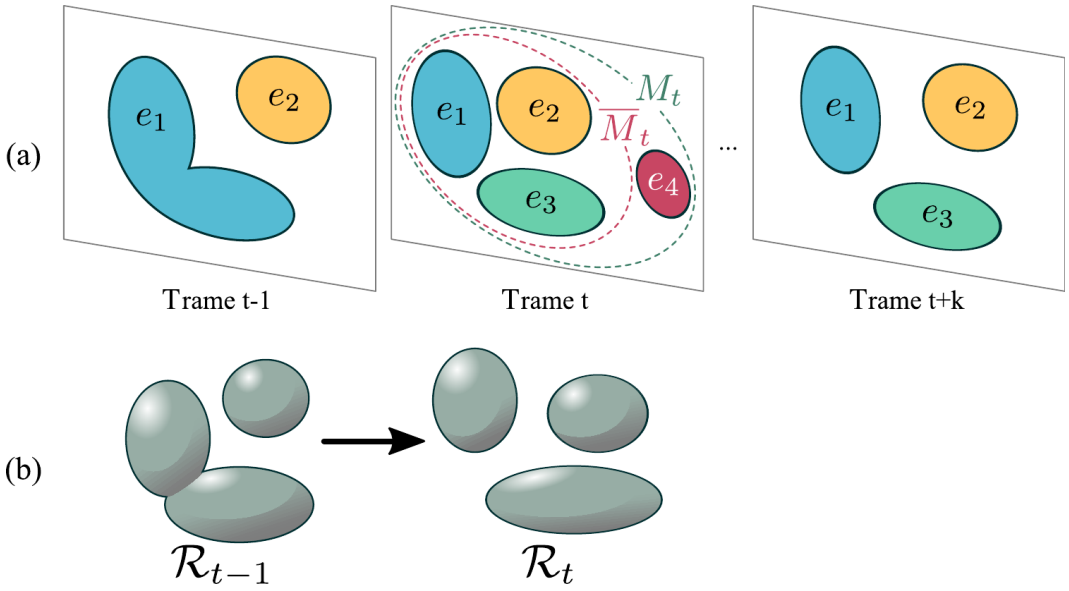


Figure 4.2 – Evolution séquentielle du modèle progressif. (a) Objets distincts observés sur plusieurs trames, qui présentent une donnée erronée à l'instant t et (b) l'évolution correspondante du modèle $\mathcal{R}$.

Ainsi, la méthode fait évoluer le modèle progressif tout au long de la séquence. En utilisant une formulation ensembliste, le problème s'exprime comme suit. Soit $M_t = \{e_j\}$ l'ensemble des éléments distincts observable à l'instant t

(voir figure 4.2). Comme nous l'avons vu au début de ce manuscrit (voir figure 1.8), les données en entrée ne sont pas toujours parfaites. Pour cette raison nous mettons en place un filtrage simple qui consiste à ne prendre en compte les nouvelles informations sur la topologie que si elles sont visibles sur k trames successives. Ainsi,

$$\overline{M}_t = \{e_j \mid e_j \in \bigcap_{i=0}^{k} M_{t+i}\}, \tag{4.1}$$

est l'ensemble des éléments géométriques distincts à l'instant t. Soit $\mathcal{R}_n$ l'ensemble des éléments géométriques dans le modèle de référence à l'instant n. En suivant notre hypothèse que les objets de la scène observée ont une topologie fixe, nous déduisons que le modèle progressif de la scène doit inclure tout les éléments valides qui apparaissent dans les observations successives. Ainsi nous pouvons écrire :

$$R_n = \bigcup_{i=1}^{n} \overline{M}_i = R_{n-1} \bigcup \overline{M}_n \,. \tag{4.2}$$

La figure 4.2 montre comment le modèle progressif $\mathcal{R}_n$ évolue avec un exemple simple montrant l'identification de deux nouveaux éléments e_3 et e_4 ; le premier venant améliorer le modèle et le second étant associé à du bruit dans les observations.

La modélisation progressive d'un modèle se fait en quatre étapes qui sont répétés à chaque nouvelle trame, tels que décrits dans la figure 4.3. Chacune de ces étapes sera ensuite détaillée dans une section spécifique.

1. Les différents composants du maillage observé dans la nouvelle trame sont mis en correspondance avec le modèle progressif courant (4.3.2).
2. Les nouvelles informations topologiques sont détectées comme des faces internes du maillage précédemment déformé (4.3.3).
3. Le maillage de la trame courante déformé et sans face interne est aligné avec précision sur le modèle courant (4.3.4).
4. Les nouvelles informations topologiques, c'est-à-dire les faces internes, sont ajoutées au modèle courant qui est ainsi mis à jour (4.3.5).

Dans la suite, nous déroulons la méthode sur une unique itération et nommons le maillage obtenu à l'observation courante et le modèle progressif courant respectivement $\mathcal{M}$ et $\mathcal{R}$.

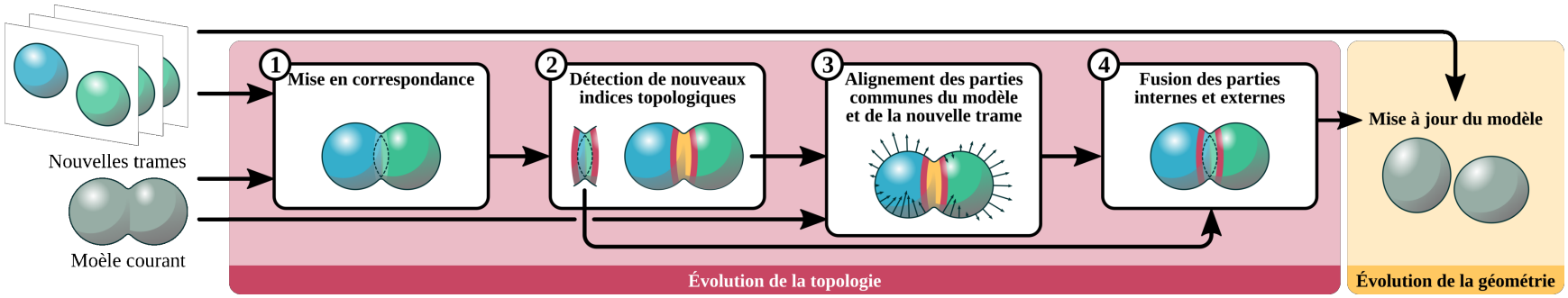


Figure 4.3 – Vue globale du déroulement de la méthode.

4.3.2 Mise en correspondance

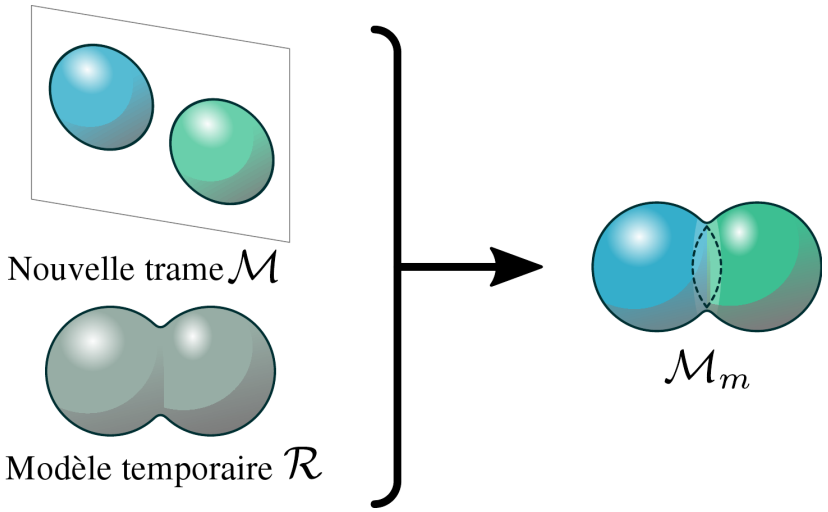


Figure 4.4 – Mise en correspondance : la nouvelle observation $\mathcal{M}$ est mise en correspondance avec le modèle progressif courant $\mathcal{R}$. Le résultat est une version déformée $\mathcal{M}_m$ de $\mathcal{M}$ qui est alignée sur $\mathcal{R}$.

La première étape de notre méthode consiste à mettre en correspondance les nouvelles observations $\mathcal{M}$ avec la version courante du modèle progressif $\mathcal{R}$. Cette étape permet d'identifier les différences de topologie entre le modèle et les observations. La difficulté principale vient du fait que les deux maillages présentent des différences de connectivités et possiblement un nombre de composantes connexes différent. Pour répondre à ce problème, nous formulons la mise en correspondance comme un problème d'optimisation où la distance entre la déformation de $\mathcal{M}$ et le modèle courant est minimisée tout en forçant la fonction de déformation de la surface à être localement lisse.

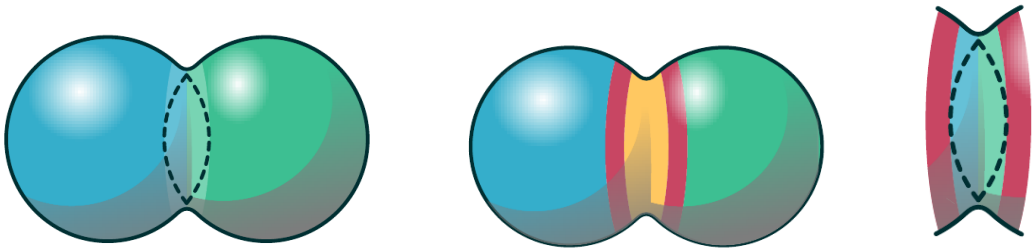
En appelant Θ les paramètres de la déformation, c'est-à-dire les déplacements de chaque sommet du maillage, l'optimisation consiste à maximiser la log-vraisemblance de la distribution de la probabilité jointe des observations $\mathcal{M}$, du modèle $\mathcal{R}$ et des paramètres de déformation Θ :

$$\arg\max_{\Theta} \quad \ln P(\mathcal{M}, \mathcal{R} \mid \Theta) P(\Theta)\,. \tag{4.3}$$

Plusieurs approches peuvent être considérées pour cette étape, utilisant soit des informations photométriques soit des informations géométriques. Par exemple, Popa *et al.* [Popa 10] utilisent une méthode basée sur le flot optique calculé dans les images et re-projeté sur la surface. Néanmoins, le flot optique ne nous semble pas être la meilleure information à utiliser. En effet, en tant que

résultat d'approximations de différences finies, il n'est pas adapté pour gérer de grandes déformations. Nous nous sommes donc orientés vers une approche purement géométrique proposée par Cagniart *et al*. [Cagniart 10]. Cette approche a été initialement proposée pour effectuer le suivi d'une surface maillée sur une séquence complète de maillages reconstruit indépendamment à la topologie identique. Par la mise en oeuvre d'un cadre probabilistique cette méthode peut gérer efficacement plusieurs objets et leurs déformations ainsi que les données manquantes. C'est pourquoi elle s'adapte particulièrement bien au problème qui nous intéresse dans cette section. Nous en utilisons une version simplifiée dans notre approche qui permet de mettre en correspondance les nouvelles observations $\mathcal{M}$ avec la version courante du modèle progressif $\mathcal{R}$, comme explicité par la figure 4.4. Nous obtenons ainsi $\mathcal{M}_m$, la version déformée de $\mathcal{M}$.

4.3.3 Détection des changements de topologie



(a) $\mathcal{M}_m$ (b) $\mathcal{M}_{out}$ (c) $\mathcal{M}_{in}$

Figure 4.5 – Détection des changements de topologie : (a) Maillage en entrée avec auto-intersections dues à une évolution de topologie ; (b) suppression des auto-intersections - en rouge : sommets où les coupure ont lieu, en orange : nouvelle géométrie ajoutée à $\mathcal{M}_m$; (c) La partie *intérieure* $\mathcal{M}_{in}$. Les sommets en rouges de (b) et (c) sont communs à $\mathcal{M}_{out}$ et $\mathcal{M}_{in}$.

Une fois la nouvelle observation mise en correspondance avec le modèle courant, nous disposons d'un maillage $\mathcal{M}_m$ dont la forme est identique au modèle progressif $\mathcal{R}$ mais qui détient potentiellement de nouvelles informations de topologie sur la scène observée. L'étape décrite dans cette section consiste donc à dectecter les changements de topologies contenus dans $\mathcal{M}_m$ qui permettrons d'améliorer $\mathcal{R}$ dans les étapes suivantes. Dans le cas de formes compactes ces changements topologiques peuvent être de quatre types distintcs comme le montre la figure 4.6 : *séparation*, *fusion*, *création d'un trou* et *disparition d'un trou*. Comme mentionné précédemment, nous faisons l'hypothèse que les différents objets de la scène ont une topologie fixée. Ceci implique que nous pouvons observer une création de trou ou une séparation en conséquence d'un ajout d'information au modèle progressif. Mais à l'inverse, la disparition d'un

trou et une fusion ne peuvent pas avoir lieu puisque ces deux cas impliquent un changement dans la forme des objets observés. La figure 4.5 montre un exemple de ce principe.

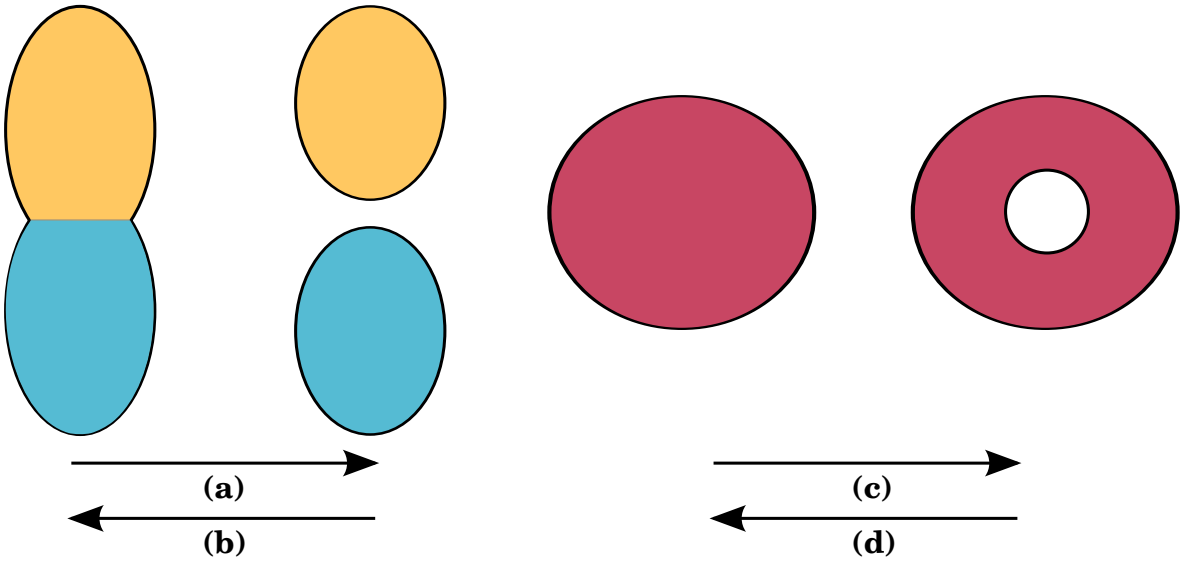


Figure 4.6 – Les types de changements topologiques : (a) séparation, (b) fusion, (c) création d'un trou et (d) disparition d'un trou.

Le cas de nouveaux objets entrant dans la scène est trivialement géré par la détection des composantes connexes des maillages venant des observations. L'idée est de vérifier si une composante connexe de $\mathcal{M}$ est classée comme appartenant au bruit lors de l'étape de mise en correspondance (voir section 4.3.2). Dans ce cas, l'objet n'appartiens pas encore au modèle progressif et y est donc tout simplement inclus.

Les autres changements de topologie non triviaux sont détectés lorsque des auto-intersections sont présentes dans le maillage mis en correspondance $\mathcal{M}_m$. Les facettes marquées comme appartenant à l'intérieur de $\mathcal{M}_m$ sont considérées comme des parties manquantes du modèle progressif $\mathcal{R}$. Dans cette optique, nous nous basons sur le travail fait par Zaharescu *et al.* [Zaharescu 11]. Cette approche identifie la partie *exterieure* d'un maillage contenant des auto-intersections et produit une surface qui représente cette partie du maillage d'entrée (voir figure 4.5b). La principale caractéristique de cette méthode est qu'elle preserve la géométrie du maillage et n'en modifie que sa connectivité ; des triangulations de Delaunay sont effectuées au niveau des facettes qui sont sur la frontière interieur / exterieur. En appliquant cette méthode sur $\mathcal{M}_m$, nous obtenons $\mathcal{M}_{out}$: un maillage fermé (TODO watertight) ayant la même géométrie que le modèle $\mathcal{R}$. Nous calculons aussi la partie intérieur $\mathcal{M}_{in}$ telle

que $\mathcal{M}_m \setminus \mathcal{M}_{out}$. Durant ce processus, nous procédons aussi au marquange des sommets du maillage où les coupures ont lieu (voir figure 4.5). Si aucune auto-intersection n'est apparue dans $\mathcal{M}_m$, le modèle progressif $\mathcal{R}$ reste inchangé et le processus s'arrête ici pour la trame courante.

En sortie de cette étape, nous disposons donc de deux maillages. $\mathcal{M}_{out}$ une version re-maillée de l'exterieur de $\mathcal{M}_m$ et $\mathcal{M}_{in}$, l'ensemble des facettes internes ou partiellement internes de $\mathcal{M}_m$. Dans ces deux maillages, les sommets qui correspondent aux endroits où les coupures ont eu lieu sont marquée pour simplifier leur accés lors des étapes suivantes.

4.3.4 Alignement précis

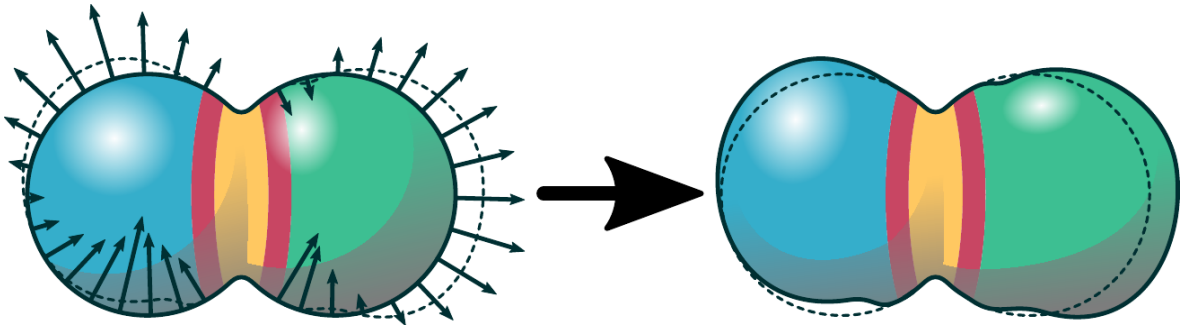


Figure 4.7 – Alignement précis de $\mathcal{M}_{out}$ sur la forme courante du modèle progressif.

La troisième étape de la méthode proposée consiste à aligner très précisement les observations partielles mis en correspondance $\mathcal{M}_{out}$ sur le modèle courant $\mathcal{R}$. Ceci dans le but d'incorporer $\mathcal{M}_{in}$ à $\mathcal{R}$ lors de l'étape suivante. À la suite des opérations précédentes, $\mathcal{M}_{out}$ est équivalent topologiquement à $\mathcal{R}$. Ainsi nous appliquons une méthode de déformation où chacun des sommets de $\mathcal{M}_{out}$ est déplacé vers la surface $\mathcal{R}$. Plus précisement, à chaque itération, les sommets sont déplacés le long de leur normale en utilisant le vecteur de déplacement d suivant, calculé en chaque sommet p de $\mathcal{M}_{out}$:

$$\mathbf{d}_p = \gamma^{\mathcal{R}}(p)\ N(p)\,, \tag{4.4}$$

où $N(p)$ est la normale à $\mathcal{M}_{out}$ au sommet p et $\gamma^{\mathcal{R}}()$ une fonction de distance signée vers $\mathcal{R}$. L'algorithme de déformation utilisé est iteratif. Apres chaque étape, le maillage subit une passe de ré-échantillonage. C'est-à-dire que de nouveaux sommets sont créés dans les zones étirés et qu'à l'inverse d'autres sont supprimés dans les zones comprimées. Pour préserver la topologie durant ces déformations, nous avons de nouveau recours à la méthode présenté dans [Zaharescu 11] sur la version déformée de $\mathcal{M}_{out}$. Cette méthode

Dans le processus de déformation décrit ci-dessus, les sommets marqués lors de l'étape précédente comme formant les frontières entre intérieur et extérieur (en rouge et orange sur la figure 4.7) ne sont pas déplacés. Ceci pour deux raisons. La première est que, par nature, ces points n'appartiennent pas au modèle progressif $\mathcal{R}$, et en tant que tel ne doivent pas participer au processus de déformation. La seconde est qu'ils correspondent à une partie des observations qui ne doit pas être modifiée puisqu'elle est commune à $\mathcal{M}_{in}$ et $\mathcal{M}_{out}$ et qui sera utilisée dans l'étape suivante pour fusionner les parties internes et externes du maillage.

4.3.5 Mise à jour du modèle progressif

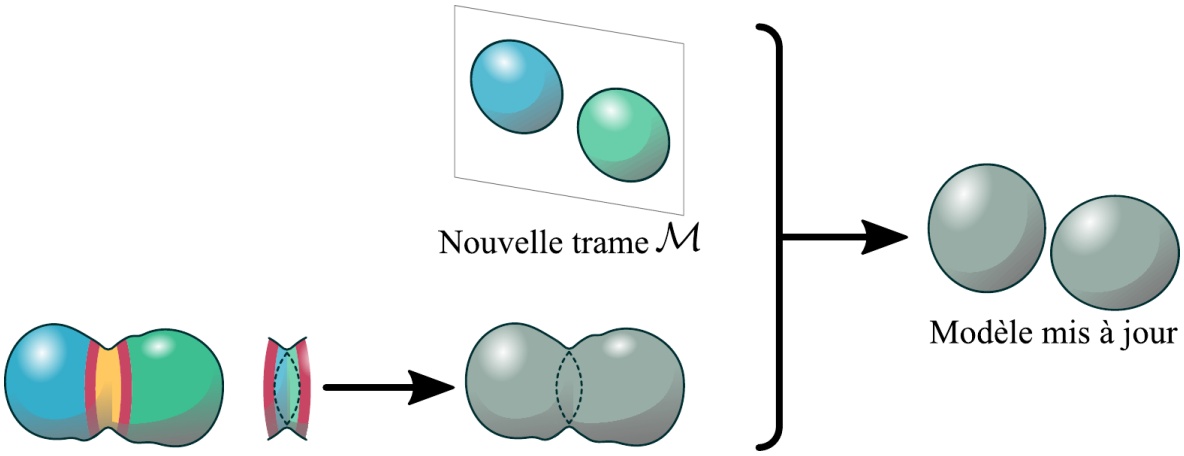


Figure 4.8 – Mise à jour du modèle : fusion de la version déformée de $\mathcal{M}_{out}$ et de $\mathcal{M}_{in}$ suivi d'une mise en correspondance avec les dernières observations.

À ce stade, nous disposons de deux maillages. $\mathcal{M}_{in}$ qui encode les évolutions de topologie apportées par la dernière observation $\mathcal{M}$ et $\mathcal{M}_{out}$ qui est géométriquement équivalent au modèle progressif courant $\mathcal{R}$ mais qui contient en plus une information de localité concernant les endroits où les coupures ont eu lieu lors de l'étape présentée dans la section 4.3.3. L'étape finale de la méthode proposée fusionne simplement ces deux maillages ensemble. Ce processus est grandement facilité par la correspondance un-à-un entre les sommets qui servent de jointure entre $\mathcal{M}_{in}$ et $\mathcal{M}_{out}$. Cette correspondance est garantie par les différents traitements spécifiques que nous avons appliqués à ces sommets durant les étapes précédentes. Lors de cette étape, les facettes créées lors des triangulations de Delaunay (en orange sur la figure 4.8) sont supprimées et les sommets communs à $\mathcal{M}_{in}$ et $\mathcal{M}_{out}$ (en rouge sur la figure 4.8) sont fusionnés.

Le maillage ainsi obtenu contient à la fois les informations accumulées sur toutes les trames précédentes par le modèle progressif $\mathcal{R}$ et les nouvelles

informations apportées par la dernière observation $\mathcal{M}$. Le modèle $\mathcal{R}$ est donc remplacé par ce nouveau maillage. Avant de traiter la trame suivante de la séquence, le nouveau modèle progressif est déformé pour être mis en correspondance avec les observations de la trame traitée. C'est une étape de suivi où la géométrie du modèle évolue pour être le plus proche possible des dernières observations. Pour cela nous faisons usage de la même méthode de mise en correspondance que celle présentée dans la section 4.3.2. Cette fois la source et la cible sont respectivement le modèle et les observations (voir figure 4.8).

En suivant ces étapes, à la fin du déroulement de notre méthode, nous somme en disposition d'un modèle dont la topologie est cohérente avec toutes les observations précédentes et dont la géométrie est alignée sur la dernière trame traitée. Ce modèle peut donc être déformé pour correspondre à toutes les observations en traitant les trames successivement en sens inverse.

4.3.6 Notes d'implémentation

Durant l'étape d'alignement précis entre $\mathcal{M}_{out}$ et le modèle progressif courant $\mathcal{R}$ (voir section 4.3.4), une partie du maillage $\mathcal{M}_{out}$ est préservée des déformations (les parties en rouge et jaune sur la figure 4.7). Cette zone correspond à la géométrie ajoutée au maillage $\mathcal{M}_m$ lors de l'étape de détection des changements de topologie. Afin d'éviter les transitions abruptes entre les zones qui sont sujettes aux déformations et celles qui ne le sont pas, nous étendons la contrainte de non-déformation en définissant une zone de transition douce. Pour cela, le N-voisinage des sommets contraints sont marqués comme appartenant à la zone de transition. Dans cette zone, la connectivité initiale du maillage est préservée en interdisant tout ajout ou suppression de sommets ; néanmoins, ces sommets sont autorisés à se déplacer librement. Une fois le critère de convergence défini dans la section 4.3.4 atteint, leurs positions finales sont calculées comme une fonction de leurs anciennes et nouvelles positions, pondérées par leurs distances à la frontière intérieur/extérieur. Cela revient à dire que les sommets proches de la frontière bougeront peu, tandis que ceux éloignés seront de moins en moins entravés. La taille de ce voisinage est le seul paramètre dépendant des données de la méthode proposée. Il peut facilement être déterminé en examinant les données et plus particulièrement le ratio entre la taille des arêtes et celle des auto-intersections.

4.4 Évaluations

Les évaluations que nous présentons ici ont été effectuées à la fois sur des données de synthèse et des données réelles, ceci pour trois raisons principales.

Premièrement l'utilisation de données de synthèse nous permet pour proposer à la fois des résultats qualitatifs et quantitatifs, en second ces mêmes données permettent de couvrir de manière exhaustive les différents cas de figures qui peuvent se présenter. Finalement, les données réelles permettent de démontrer la faisabilité et la robustesse de notre méthode dans des cas de données non totalement maîtrisées.

Avant de présenter les données traitées ainsi que les résultats obtenus, il est important de noter que dans tous les exemples qui suivent, nous ne mettons en avant, dans les figures, que les quelques trames clés des séquences traitées. Ces trames sont celles qui présentent un intérêt pour le problème traité et qui déclenchent un traitement complet par notre méthode. Les trames intermédiares, n'engendrant pas de changement de topologie, ne présentent pas un interet particulier pour ces travaux.

Nous avons créé trois jeux de données synthétiques. Les trois séquences présentées ci-dessous ont toutes été générées à l'aide du logiciel Blender[1] en animant une scène et en sauvant les maillages correspondant à chaque trame. Ensuite l'intégralité de ces maillages sont traités afin de n'en conserver que la surface *extérieure* ; ainsi les données traitées se rapproche de celles que l'on peut obtenir avec une méthode de reconstruction 3D statique, telle que les enveloppes visuelles par exemple.
Le premier jeu de données est une scène composée de trois sphères qui se déplacent et entrent en contact les unes avec les autres. Les maillages générés et utilisés en entrée de notre méthode sont composés d'environ 2000 sommets. L'intérêt principal de cette séquence est qu'aucune des trames ne présente la vraie topologie de la scène.
La seconde séquence montre un objet en forme de Y qui présente des changements de type fusion et séparation. Les maillages de cette séquence sont composés d'environ 6000 sommets.
La dernière séquence de synthèse proposée montre un objet, composé d'environ 3500 sommets, qui se déforme de manière non rigide et la séquence ainsi générée présente une création et disparition de trou.

Des expérimentations ont aussi été conduites sur des données réelles. Nous avons traité trois différentes séquences disponibles publiquement pour la communauté. La première est un extrait de la séquence *flashkick* de l'université de Surrey [Starck 07b]. Les deux autres sont des séquences qui ont été acquises à l'INRIA dans la salle GrImage, dans le cadre des travaux de cette thèse (voir section 2.3.4.2). Ces trois séquences présentent des séquences temporelles de maillages incohérents dans le temps reconstruits avec une méthode standard

[1] http://www.blender.org/

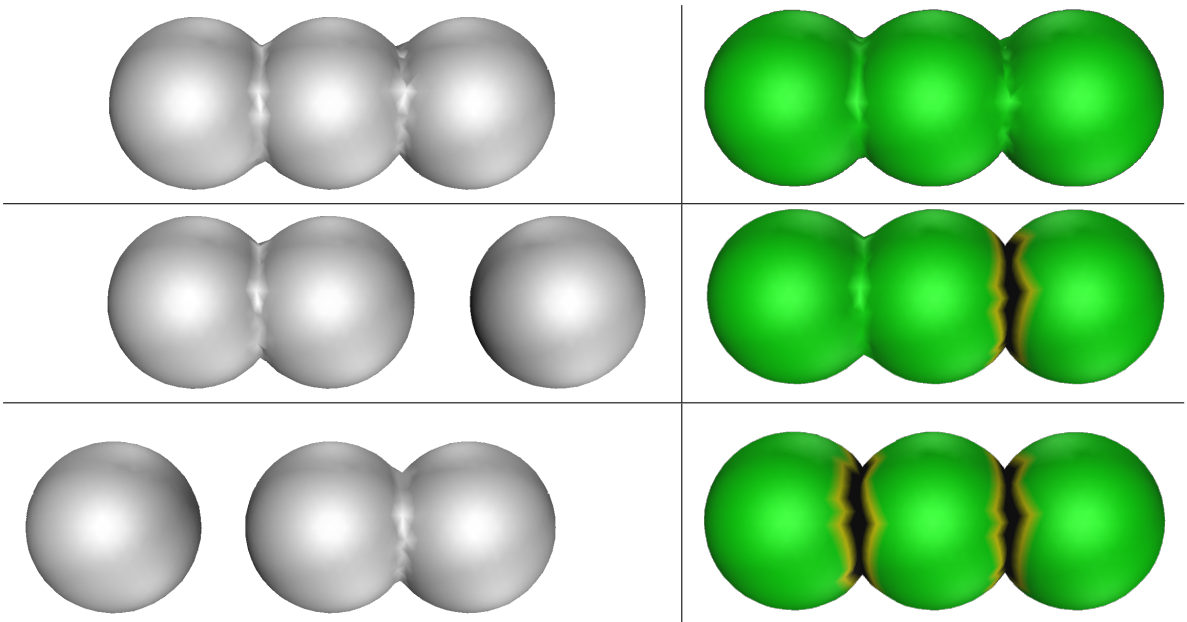
de modélisation 3D, l'enveloppe visuelle[Franco 08]. Les maillages de ces séquences présentent régulièrement des changements de topologie, problème caractéristique de ce type de données auquel ce travail propose une solution.

Les temps de calculs dépendent principalement de la complexité des maillages et sont de l'ordre d'une minute ou deux sur un PC standard (Core 2 Duo, 4Go de RAM) dans le cas le plus complexe traité ici, soit environ 10.000 sommets. L'étape qui demande le plus de calcul est celle durant laquelle les auto-intersections sont détectées et isolées. Les temps de calcul ne sont donc pas directement dépendants de la complexité des maillages traités mais plutôt de la taille des zones qui présentent des auto-intersections et du nombre de facettes concernées.

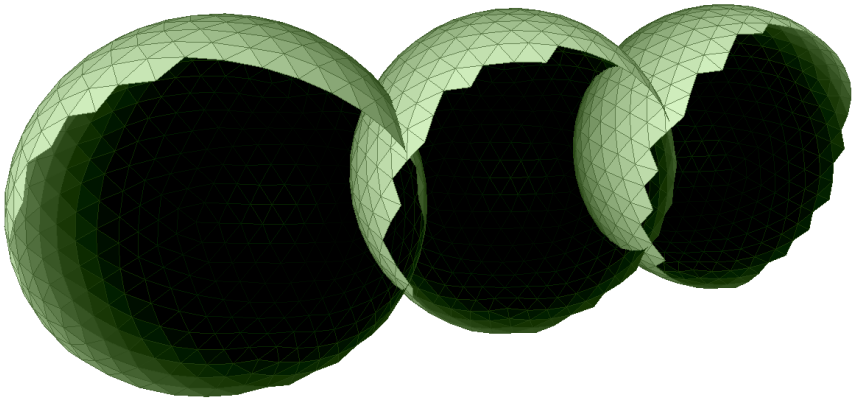
4.4.1 Données de synthèse

4.4.1.1 Sphères

Dans le but de démontrer la capacité de notre méthode à produire un modèle de scène qui est de meilleure qualité que chacune des trames de la séquence prise individuellement, nous avons généré cette séquence de trois trames où trois sphères s'intersectent les unes avec les autres. La figure 4.9a montre les trois maillages générés dans le cadre de cette séquence. Il est intéressant de noter qu'aucune des données en entrée ne contient la totalité de l'information topologique de la scène ; cette dernière est dispersée dans la totalité de la séquence. La figure 4.9b montre l'évolution du modèle progressif au fur et à mesure que les trames sont observées séquentiellement. La figure montre que la méthode proposée fait converger graduellement le modèle progressif vers la vraie forme de la scène. L'information de topologie est correctement accumulée à chaque nouvelle trame traitée puisque le dernier modèle construit par notre méthode contient bien les trois sphères complétes et indépendantes. Les parties vertes de la surface du modèle représente les zones communes à deux versions successives du modèle. Les zones noires correspondent à la géométrie ajoutée au modèle précédent, c'est-à-dire les frontières physiques entre objets détectées lors du traitement de la trame courante. Les anneaux jaunes entourant les zones noires correspondent aux N-voisinages des zones de coupure où les sommets sont protégés lors de l'étape d'alignement précis, comme spécifié dans la section 4.3.6. Ce code couleur sera le même pour tous les résultats présentés dans cette section. Pour ce jeu de données, nous avons utilisé un voisinage de taille 1. La figure 4.9c montre l'intérieur de la dernière évolution du modèle progressif. Nous pouvons remarquer que la scène est bien composée de trois objets distincts. Les intersections entre les sphères est un résultat de l'algorithme de mise en correspondance utilisé.



(a) Données en entrée (b) Évolution du modèle

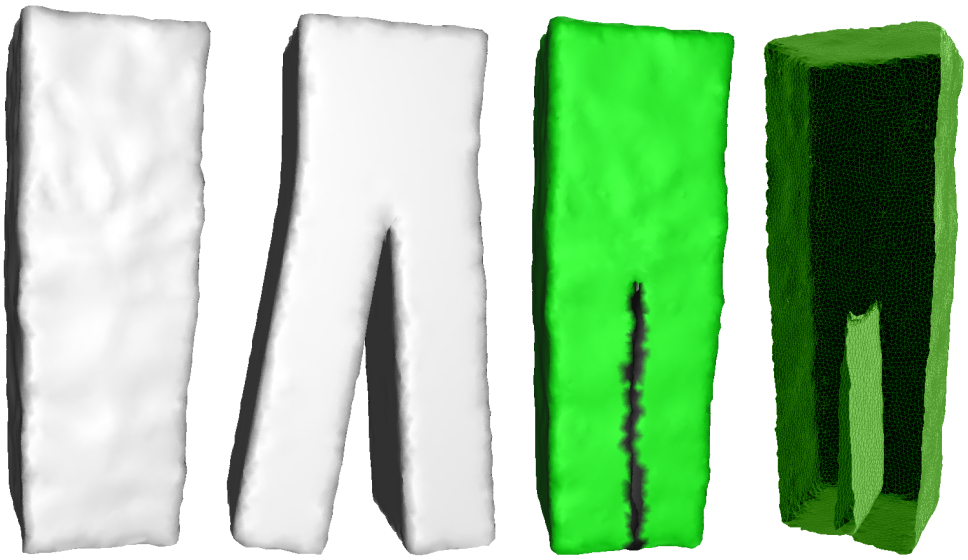


(c) Vue intérieure du dernier modèle

Figure 4.9 – Résultats obtenus sur la première séquence de synthèse. (a) Maillages d'entrée de notre méthode. (b) Évolution associée du modèle progressif, alignée sur la première trame. (c) Vue intérieure de la dernière évolution du modèle progressif.

4.4.1.2 Objet en forme de Y

Les résultats obtenus sur la seconde séquence de synthèse montrent que notre méthode ne retrouve pas uniquement les déformations qui implique un changement des nombres de Betti du maillage, c'est-à-dire les trous et le nombre de composantes connexes, mais aussi les changements moins radicaux. Dans la première trame, les jambes de l'objet ne sont pas distinguables, ceci est dû à la simplification par l'enveloppe convexe. Plus tard dans la séquence, les jambes



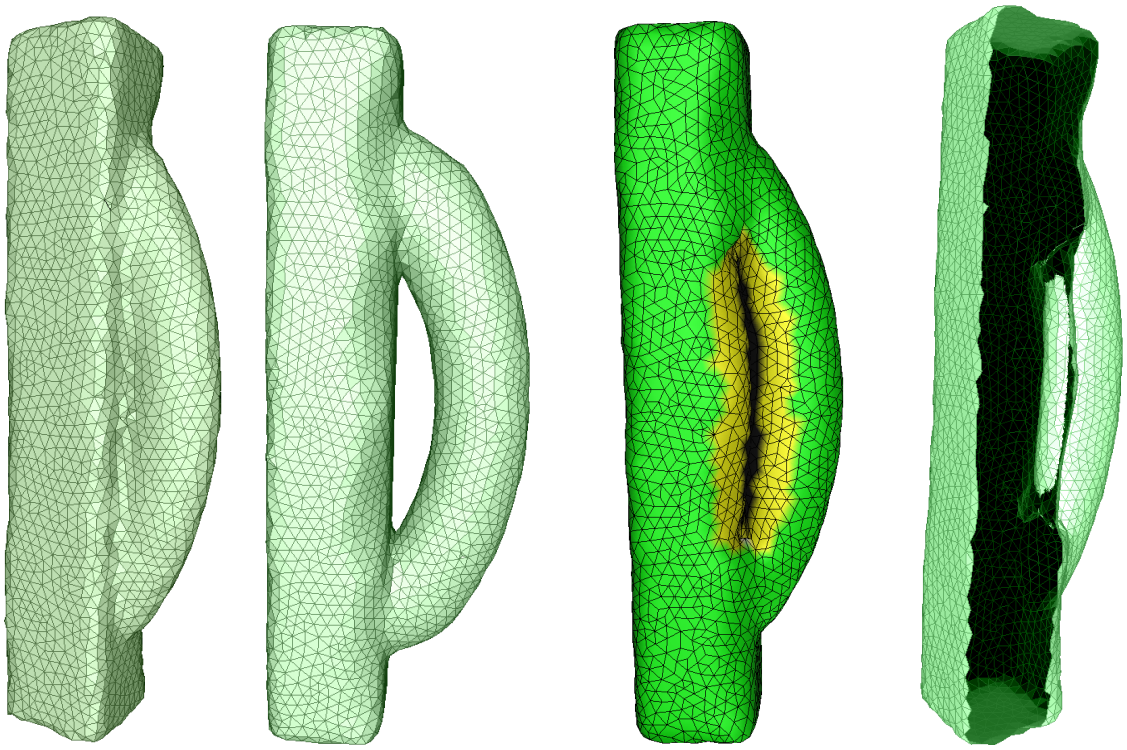
(a) Données en entrée (b) Modèle (c) Intérieur

Figure 4.10 – Résultats obtenus sur l'objet en forme de Y. (a) Deux trames issues de la séquence. (b) Le modèle obtenu après le traitement de la seconde trame, aligné sur la forme de la première. (c) Vue de l'intérieur du modèle déformé pour être aligné sur la première trame de la séquence.

sont clairement identifées et séparées, voir figure 4.10a. Comme on peut le voir dans les figures 4.10b et 4.10c le modèle obtenu en sortie de notre méthode contient bien la séparation des jambes. Dans le cas de cette séquence, nous n'avons pas eu besoin de recourir à la protection du voisinage des zones de coupures (0-voisinage).

4.4.1.3 Apparition d'un trou

La dernière séquence de synthèse que nous présentons ici illustre le second type de changement topologique présenté dans la figure 4.6, à savoir l'apparition d'un trou. L'objet présent dans cette séquence présente une déformation continue qui provoque l'apparition d'un trou dans les maillages de la séquence. Ce trou indique une séparation physique, qui existe dans la vraie forme observée, entre les différentes parties de l'objet. Les deux images de la figure 4.11a montrent la première trame, aussi utilisée comme modèle initial, et une trame ultérieure pour laquelle la séparation est présente dans le maillage associé. La figure 4.11b montre l'évolution associée à cette dernière observation, alignée sur le maillage de la première trame, du modèle progressif. Nous pouvons voir clairement sur la vue intérieure présentée dans la figure 4.11c que le trou a bien été incorporé au modèle. Pour cette séquence, la dimension de la zone de transition des déformations a été fixée à 4.



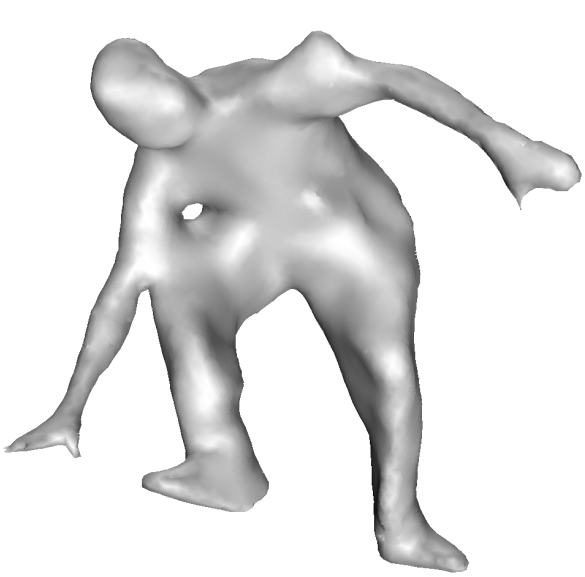
(a) Données en entrée (b) Modèle (c) Intérieur

Figure 4.11 – Résultats obtenus sur l'objet présentant un trou. (a) Deux trames issues de la séquence. (b) Le modèle obtenu après le traitement de la seconde trame, aligné sur la forme de la première. (c) Vue de l'intérieur du modèle déformé pour être aligné sur la première trame de la séquence.

4.4.2 Données réelles

4.4.2.1 Séquence Flashkick

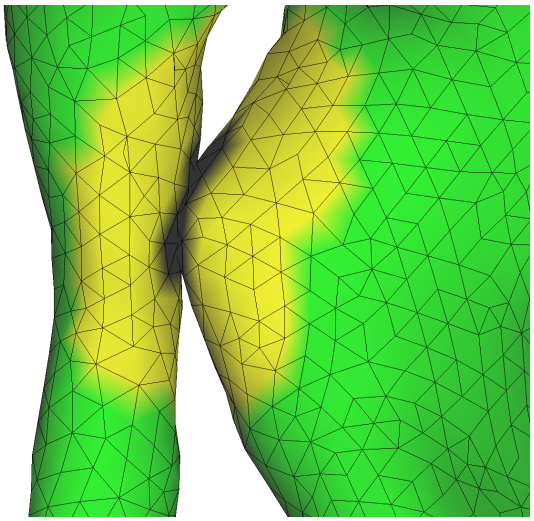
La première séquence de données réelles traitée a été acquise à l'aide d'un système multi-caméras composé de huit capteurs couleur. Le nombre relativement bas de capteurs, combiné à l'utilisation d'un algorithme de reconstruction 3D grossier conduit à l'apparition d'un grand nombre d'artefacts topologiques dans les maillages reconstruits. Les figures 4.12a et 4.12b montrent deux trames successives issues de cette séquence. Le modèle a été initialisé avec la trame (a). La figure 4.12c montre l'évolution du modèle progressif associée au traitement de la trame (b), aligné sur la première trame. Nous pouvons remarquer que le coude et le genou sont correctement disjoints. À cause de la haute résolution des maillages en entrée ainsi que de la taille relativement large des zones de fusion des données en entrée, la taille de la zone de transition de déformation (en jaune) a été fixée à 4 (4-voisinage) pour cette séquence.



(a) Premier modèle

(b) Observation ultérieure

(c) Évolution du modèle



(d) Agrandissement

Figure 4.12 – Résultats obtenus sur la séquence *Flashkick* de l'université de Surrey. (a) La première trame, qui sert aussi d'initialisation du modèle progressif. (b) Une trame suivante présentant un changement de topologie par rapport au modèle. (c) L'évolution associée du modèle, aligné sur la première trame. (d) Agrandissement sur la zone de changement de topologie, intersection coude-genou.

4.4.2.2 Séquence Homme debout

La seconde séquence de données réelles présentée ici représente un homme debout déplaçant ses mains de ses hanches à sa tête. Cette séquence de maillages à été reconstruite à partir d'un système multi-caméra de 32 capteurs, conduisant à des maillages plus précis que ceux de la séquence précédente. Néanmoins la topologie de ces données d'entrée n'est toujours pas convaincante lorsque des contacts ont lieu. La figure 4.13a présente la première trame de la séquence, aussi utilisée comme initialisation du modèle progressif côte-à-côte avec deux trames de la suite de la séquence, respectivement les trames numéro 37 et 69. Ces deux dernières sont mises en avant car elles présentent des changements de topologie. Comme illustré dans les figures 4.13b et 4.13c, notre méthode parvient correctement à faire évoluer séquentiellement le modèle progressif pour construire un maillage qui soit cohérent à la fois spatialement et temporellement avec les observations. Dans le cas de cette séquence, la taille de la zone de transition des déformations autour des coupures a été fixée à 2.

4.4.2.3 Séquence Homme, enfant et balle

La dernière séquence de test présentée montre un homme et un enfant jouant avec une balle. Les maillages de cette séquence sont des enveloppes visuelles issues d'un système disposant de 16 caméras. Il s'agit d'une scène assez complexe contenant 3 objets déformables distincts qui interagissent. Il s'agit typiquement du genre de scène qui est trop complexe pour des approches basées modèles. La figure 4.14a montre deux trames successives de la séquence, la première étant utilisée pour initialiser le modèle progressif. Les figures 4.14b et 4.14c montrent que les deux mains de l'homme ont correctement été détachées de la balle dans la nouvelle version du modèle progressif. Cet exemple démontre la capacité de notre approche à gérer plusieurs changements de topologie en une seule passe. Dans le cas de cette séquence, la taille de la zone de transition des déformations a été fixée à 1.

4.4.3 Evaluation quantitative

Nous avons pratiqué une évaluation quantitative de nos résultats en utilisant la distance de Hausdorff (voir table 4.1). Dans tous les cas nous avons calculé la distance entre la forme finale du modèle progressif et les données en entrée. Quand cela était possible, pour les données de synthèse principalement, nous avons aussi calculé la distance entre le modèle et la vérité terrain. Les nombres données sont à chaque fois un ratio de la diagonale de la boîte englobante des maillages. Les résultats obtenus sont très satisfaisants car proches de zéro, ceci est principalement dû à l'étape d'alignement précis décrit dans la section 4.3.4. Il est aussi important de noter que dans le cas des données de synthèse, la

Séquence	Distance moyenne aux données en entrée	Distance moyenne à la vérité terrain
Spheres	0,003410	0,003194
Y	0,002036	0,001952
Apparition d'un trou	0,002887	0,002635
Flashkick	0,000680	–
INRIA - homme	0,000550	–
INRIA - homme, enfant et balle	0,000537	–

TABLE 4.1 – Distance de Hausdorff moyenne entre les modèles finaux d'une part et les données en entrée et la vérité terrain quand celle ci est disponible d'autre part.

distance entre le modèle calculé et la vérité terrain est plus faible que celle entre le modèle et les observations. Cette observation démontre clairement l'efficacité de notre méthode et sa capacité à accumuler de l'information sur des données incomplètes pour se rapprocher au plus près de la vraie forme de la scène. D'autre part, les valeurs sont plus grandes dans le cas des données de synthèse à cause de la plus faible résolution des maillages utilisés.

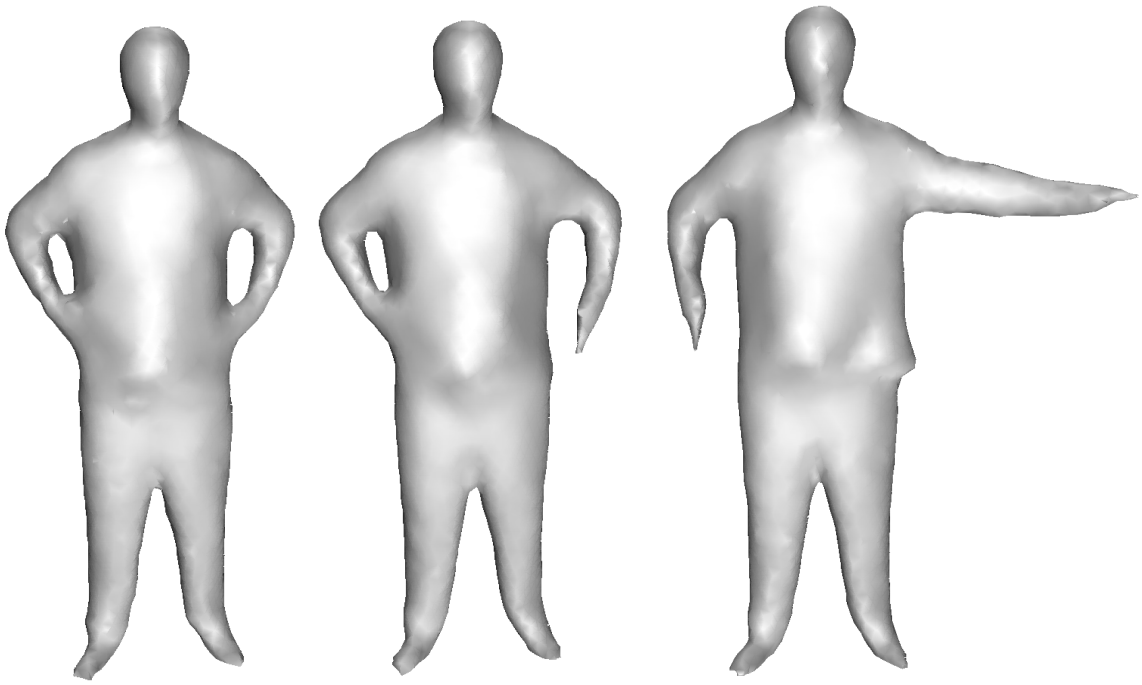
4.5 Conclusion et discussion

Ce chapitre introduit la notion de modèles progressifs de forme. Basés sur l'hypothèse voulant que les différents objets qui composent une scène aient une topologie fixée, ces derniers permettent d'apprendre à la fois la forme en mouvement et la topologie des objets à partir d'une séquence temporelle d'observations. Ces travaux permettent de produire une information très utile pour un traitement ultérieur des maillages de la séquence, tel que du suivi de surface, du calcul de flot de scène, ou encore de la capture de mouvement sans recourir à l'utilisation de marqueurs. Les expérimentations faites sur des données synthètiques et réelles ont permis de mettre en avant la faisabilité et l'exactitude de notre approche.

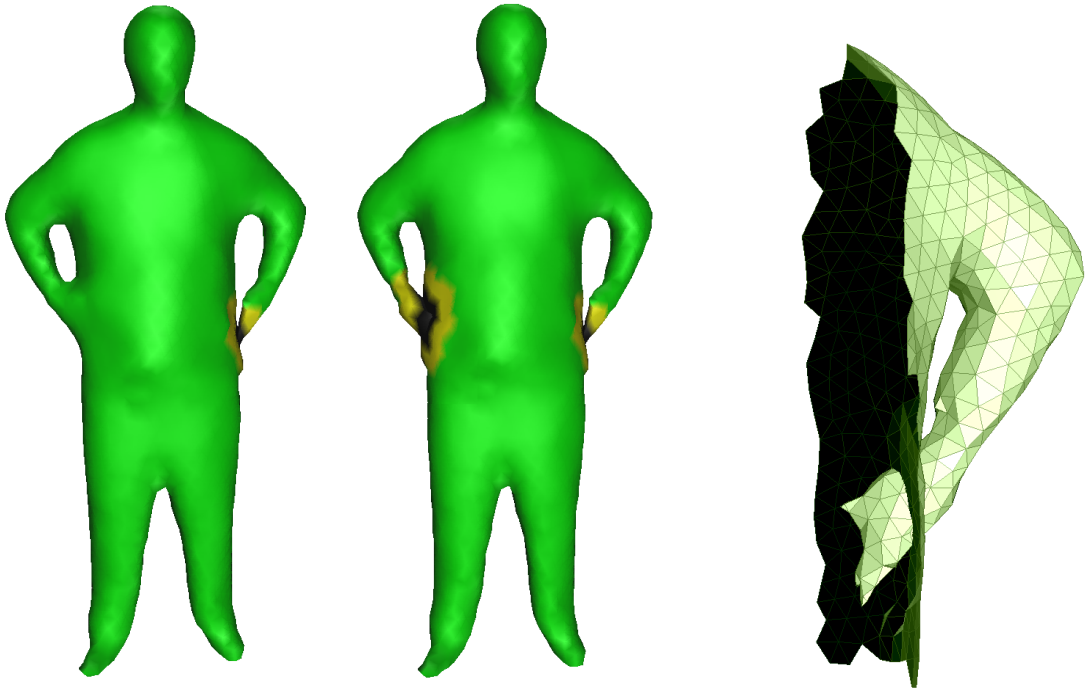
Bien que les résultats soient encourageants, nous sommes concients qu'il s'agit ici d'une première tentative de réponse au problème de l'apprentissage de la topologie d'une scène dynamique. Notre approche est très dépendante d'outils externes, c'est notamment vrai pour l'outil d'alignement de maillage. Bien que cette dernière soit considérée comme fiable, nous ne sommes pas en mesure de détecter une erreur d'alignement qui pourrait conduire à introduire de fausses informations de topologie dans notre modèle.

Notre méthode ne repose que sur des informations géométriques. Pourtant

la plupart des séquences obtenues à l'aide d'un système multi-caméra donnent accès à une information photométrique. Nous pensons que notre approche pourrait bénéficier de l'utilisation de cette autre source d'information, soit lors de l'étape de mise en correspondance 4.3.2, ou alors pendant la phase d'alignement précis 4.3.4. Dans le second cas, l'utilisation d'un algorithme de mutli-vue stéréo pourrait être utilisé pour améliorer la qualité du modèle, plus uniquement à un haut niveau pour la topologie mais aussi à un niveau plus fin en retrouvant les petits détails géométriques de la surface.



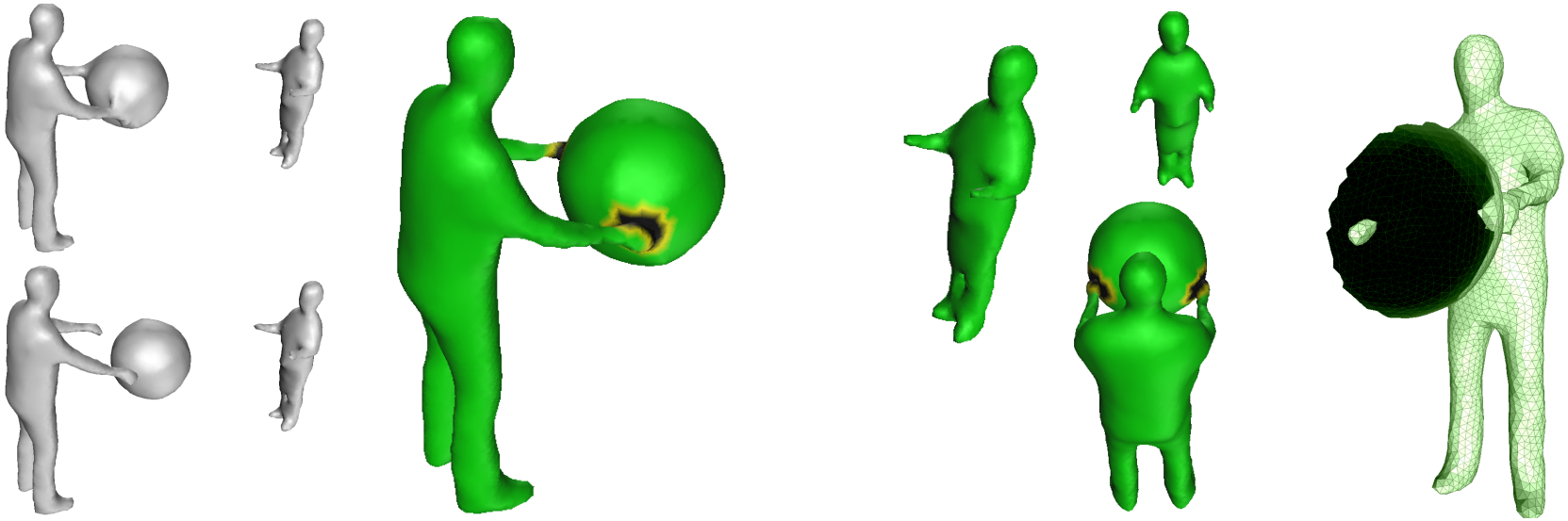
(a) Observations successives



(b) Évolutions du modèle

(c) Vue intérieure

Figure 4.13 – Résultats obtenus sur la séquence de l'homme debout acquise durant cette thèse. (a)-1 Premier maillage de la séquence et aussi initialisation du modèle progressif. (a)-2 et (a)-3 les maillages (#37 et #69) de la séquence traitée présentant des changement de topologie. (b) les évolutions respectives du modèle progressif alignées sur la premiere trame. (c) Agrandissement sur l'intérieur du bras gauche du modèle final, nous remarquons que le bras et la hanche y sont correctement séparés.



(a) Données en entrée (b) Évolution du modèle (c) Vue intérieure

Figure 4.14 – Résultats obtenus sur la séquence avec un homme et un enfant jouant avec une balle. (a) Deux trames successives issues de la séquence, la première sert aussi d'initialisation pour le modèle progressif. (b) Deux vues de l'évolution du modèle associée au traitement de la seconde trame, nous pouvons remarquer que le modèle a subi deux modifications distinctes durant la même passe d'amélioration. (c) Vue interne de la balle, cette vue montre clairement que les deux mains ont été correctement détachées de la balle.

Conclusion

Chapitre 5

Conclusion et perspectives

Ce chapitre final est l'occasion de revenir sur l'ensemble des travaux exposés dans cette thèse. Nous y présentons tout d'abord un rapide résumé des différentes contributions au domaine de la modélisation de scènes dynamiques avant de proposer quelques pistes pour poursuivre ces travaux de recherche.

5.1 Rappel des contributions

Flot de scène. Nous avons défini, dans l'introduction de cette thèse, les différents éléments qui composent une scène dynamique : sa forme, son apparence et ses mouvements. Nos travaux sur le flot de scène, présentés au chapitre 3, permettent justement de retrouver le mouvement d'une scène à partir des informations de forme et d'apparence de cette dernière. Ces travaux sont pertinents dans le contexte actuel. La plupart des systèmes d'acquisition multi-caméras fournissent en sortie, pour chaque instant de temps, une surface 3D de la scène ainsi que de multiples images couleur permettant de texturer de manière cohérente le modèle reconstruit. Le seul élément absent est donc l'information de mouvement, que notre méthode propose d'estimer.
L'approche que nous avons développée présente les caractéristiques suivantes :

- Notre méthode repose sur deux types d'informations visuelles : des points d'intérêts épars et le flot normal dense. Cet aspect la rend capable, via un algorithme en deux passes, de retrouver correctement les grands déplacements tout en conservant une bonne précision sur les détails.
- Comme c'est très souvent le cas pour l'estimation du flot de scène, nous avons recours à une passe de régularisation. Notre méthode se démarque dans le sens où cette régularisation des différentes contraintes photométriques est directement effectuée sur la surface, dans l'espace 3D donc. Ce point est important pour éviter les biais d'estimation dûs au surlissage entre objets distincts si la régularisation est faite dans l'espace image sans tenir compte de la géométrie de la scène.
- La formulation de notre méthode se fait par le biais d'un système linéaire. Sa résolution est donc simplifiée. Nous proposons une résolution qui minimise l'erreur au sens des moindres carrés ainsi qu'une résolution itérative, moins optimale mais potentiellement beaucoup plus rapide à calculer.
- Comme cela a été démontré dans la partie évaluation (3.6 et 3.7), notre méthode est très permissive quant au contexte d'acquisition des données traitées. Elle prend en compte un nombre arbitraire de flux d'images couleur

accompagnés d'une représentation géométrique de la scène. Cette dernière peut être une reconstruction 3D précise (stéréo multi-vue) ou non (enveloppe visuelle) ou bien encore une simple carte de profondeur.

Modèles progressifs de forme. Les données géométriques issues d'un système d'acquisition 4D se présentent souvent sous la forme d'une séquence de maillages sans cohérence temporelle. Cette caractéristique présente l'inconvénient de compliquer grandement les traitements effectués sur ces données. Par exemple, si le modèle géométrique de la scène était cohérent alors l'information de mouvement serait disponible instantanément. C'est pourquoi nous avons proposé une méthode, la première à notre connaissance, qui permet d'**apprendre** la topologie d'une scène à partir d'une série temporelle d'observations incohérentes. L'approche que nous proposons fait progressivement évoluer un modèle initial au fur et à mesure que les observations apportent de nouvelles informations. En se basant sur un principe d'augmentation, notre méthode est capable de retrouver la meilleure information de topologie possible d'une scène. Ceci même si cette dernière n'est jamais visible entièrement dans une des observations.

5.2 Perspectives

Nous avons déjà présenté, à la fin des chapitres 3 et 4, des pistes d'améliorations, à court terme, directements liées aux travaux menés. Les perspectives que nous exposons ici sont plutôt orientées vers le long terme.

Les travaux menés dans le cadre de l'apprentissage de la topologie d'une scène à partir d'observations sans cohérence spatiale représentent un premier pas vers l'apprentissage précis de tous les éléments qui composent une scène dynamique. Il serai interessant de développer cette idée et de l'étendre à l'apprentissage précis de la forme mais aussi de l'apparence d'une scène. Pour cela il sera nécessaire de coupler des approches multi-échelle permettant de fusionner des informations géométriques et photométriques. Concentrons nous d'abbord sur l'aspect géométrique de la scène.

Les travaux présentés dans le chapitre 4 permettent de retrouver la topologie d'une scène dynamique à partir de séquences d'pbservations. Il s'agit d'une information de basse fréquence cohérente temporellement. D'autre part, il existe une multitude de méthode de reconstruction 3D multi-vues qui ont pour but d'estimer le plus précisement possible la forme de la scène en un instant de temps. Il s'agit cette fois ci d'une information de haute fréquence valide uniquement en une trame. Le challenge évident est d'arriver à coupler ces

deux types d'approches pour obtenir une information géométrique très précise et valide sur toute une séquence. Notons au passage que le flot de scène serait alors directement obtenu à partir des déformations du modèle.

En partant du principe qu'une information géométrique fiable et précise (dans le sens où elle est cohérente avec toutes les observations) de la scène est disponible, alors l'estimation de l'apparence serait assez simple et rapide à effectuer. En effet, si nous disposons à la fois d'un maillage complet et de ses déformations à chaque trame, alors nous pouvons fusionner l'information photométrique venant de toutes les caméras sur toute la séquence par simple reprojection. En tenant compte des occultations et des changement d'illumination, il est possible d'obtenir une carte de texture de bonne qualité, unique pour toute la séquence. À ce sujet, nous pouvons nous inspirer des traveaux de Janko *et al*. [Janko 09].

Dans le cadre d'applications temps-réel, tel que c'est le cas pour la plateforme GrImage, la forme et l'apparence du modèle reconstruit et présenté à l'utilisateur est primordial afin de donner un sentiment d'imersion convaincant. Si nous pouvons proposer un modèle cohérent de par sa forme, son aspect et ses interactions avec le monde virtuel, alors il sera de plus en plus difficil pour l'utilisateur de faire la différence entre la part du réel et celle du virtuel dans une application de réalité augmentée. Ce sentiment d'imersion totale ne pourra être atteint que lorsque les trois composantes d'une scène dynamique (forme, aspect et mouvements) seront modélisés de manière satisfaisante.

Annexes

Annexe A

Filtrage bilatéral tenant compte de la profondeur

A.1 Introduction

Le filtre bilatéral [Tomasi 98] est un outil de lissage et de réduction du bruit qui vise à préserver les contours dans les images. La valeur de chaque pixel est remplacée par une valeur moyenne pondérée des valeurs des pixels voisins. Les poids utilisés sont des poids gaussiens. La caractéristique principale vient du fait que contrairement au filtre gaussien standard, le filtre bilatéral prend en compte la distance spatiale dans l'espace image mais aussi la distance de valeurs (typiquement l'intensité) entre les pixels pour le calcul des poids. En d'autres termes, deux pixels côte-à-côte dans l'image peuvent avoir une interaction très faible s'ils ont des valeurs très différentes. Cette technique de filtrage très répandue en traitement d'images est présentée de manière très détaillée dans [Paris 09].

Cette approche donne de bons résultats si l'on souhaite obtenir un lissage qui conserve les contours couleur de l'image. Or, ces contours ne correspondent pas forcément à des contours physiques de la scène capturée. Dans nos travaux nous avons accès à une information géométrique associée aux images couleur. Nous avons donc eu l'idée de prendre en compte les informations de profondeur dans le processus de lissage des images couleur.

A.2 Formulation

En reprenant la formulation présentée dans [Paris 09], le filtre bilatéral est défini de la manière suivante pour un pixel $\mathbf{p}$ d'une image I :

$$BF[I]_{\mathbf{p}} = \frac{1}{W_{\mathbf{p}}} \sum_{\mathbf{q} \in I} G_{\sigma_s}(\|\mathbf{p} - \mathbf{q}\|) G_{\sigma_r}(\|\mathbf{p} - \mathbf{q}\|) I_{\mathbf{q}}, \tag{A.1}$$

où le facteur de normalisation $W_{\mathbf{p}}$ assure que la somme des poids des pixels soit égale à 1,

$$W_{\mathbf{p}} = \sum_{\mathbf{q} \in I} G_{\sigma_s}(\|\mathbf{p} - \mathbf{q}\|) G_{\sigma_r}(\|\mathbf{p} - \mathbf{q}\|). \tag{A.2}$$

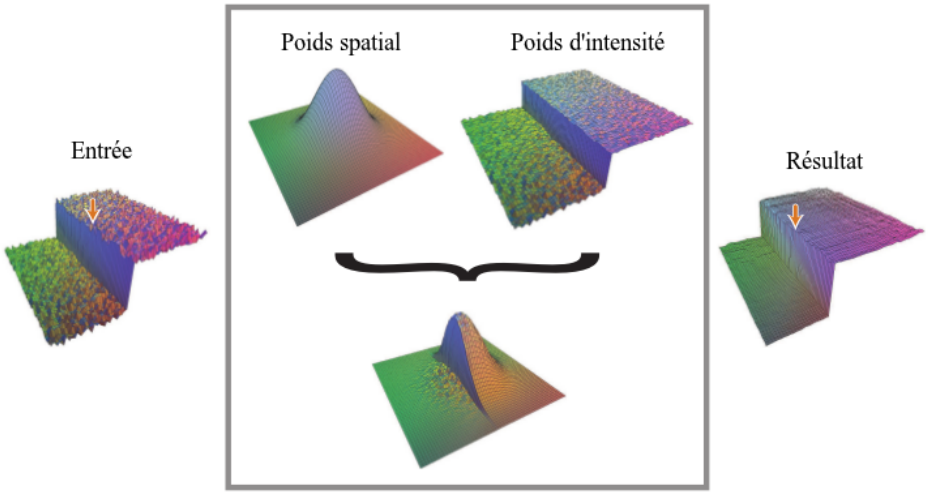


Figure A.1 – Fonctionnement du filtre bilatéral sur un pixel au centre de l'image. Image issue de [Paris 09].

Dans les deux équations précédentes, σ_r et σ_s spécifient respectivement les noyaux gaussiens pour la sensibilité à la distance en intensité et spatiale des pixels voisins.

La figure A.1 présente la combinaison de ces deux poids dans le cas du filtrage d'un pixel au centre de l'image. On voit clairement que les pixels d'intensité éloignés sont très peu pris en compte et que l'image résultat préserve la séparation nette.

A.3 Application aux images RGB-Depth

Le filtre bilatéral tient compte d'une distance en deux dimensions lors du calcul de l'influence d'un pixel sur ses voisins. Nous proposons dans cette annexe de profiter de l'information géométrique relative à une image afin de prendre en compte la *vraie* distance entre un pixel et ses voisins. Si l'image couleur est associée à une carte de profondeur (cas des données Kinect ou Time-of-Flight décrites au chapitre 3) ou bien à une information complète sur la géométrie de la scène, alors il est possible d'utiliser une distance en trois dimmension afin de pondérer l'influance d'un pixel sur ses voisins. Cette approche permet de faire un lissage d'image qui soit géométriquement cohérent et non plus visuellement cohérent. C'est à dire que lors du calcul de la nouvelle intensité d'un pixel, seul les pixels qui correspondent à des points physiques proches dans la scène capturée seront pris en compte.

La figure A.2 présente quelques résultats obtenus en faisant varier σ_r et σ_s. Et les images de la figure A.3 montrent les effets obtenus avec différents filtres, filtre gaussien, filtre bilatéral standard et filtre bilatéral tenant compte de la

profondeur. Comme attendu, nous remarquons que le filtre bilatéral qui prend en compte l'information géométrique permet de lisser ensemble les pixels qui correspondent à un même élément de la scène. Les imagettes d'agrandissements mettent en évidence une bien meilleure séparation entre les pixels du visage et ceux du ballon dans le cas d'un lissage dépendant de la géométrie, figure A.3d.

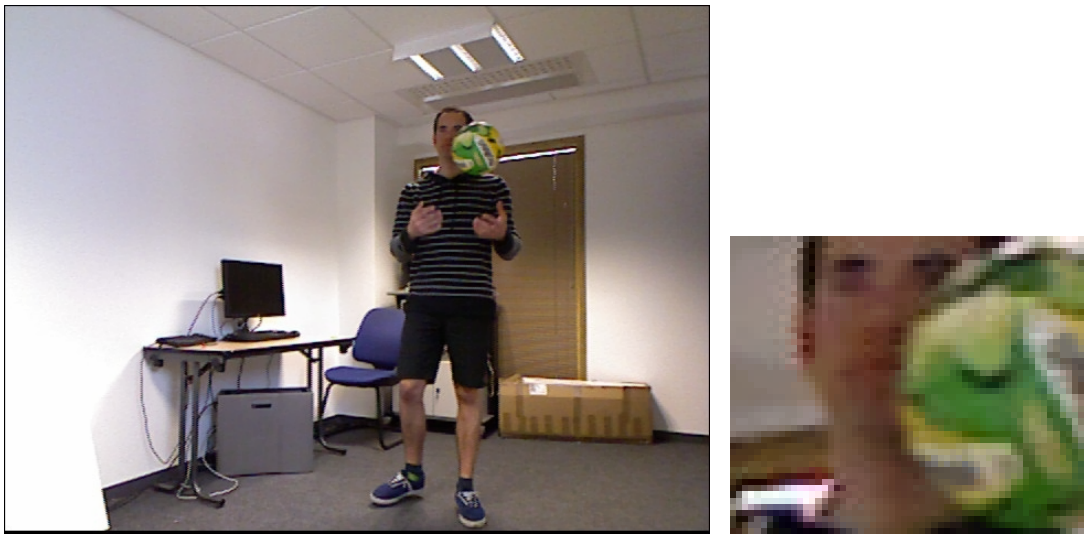
$\sigma_r = 0.004, \sigma_s = 2$ | $\sigma_r = 0.004, \sigma_s = 5$ | $\sigma_r = 0.004, \sigma_s = 15$

$\sigma_r = 0.016, \sigma_s = 2$ | $\sigma_r = 0.016, \sigma_s = 5$ | $\sigma_r = 0.016, \sigma_s = 15$

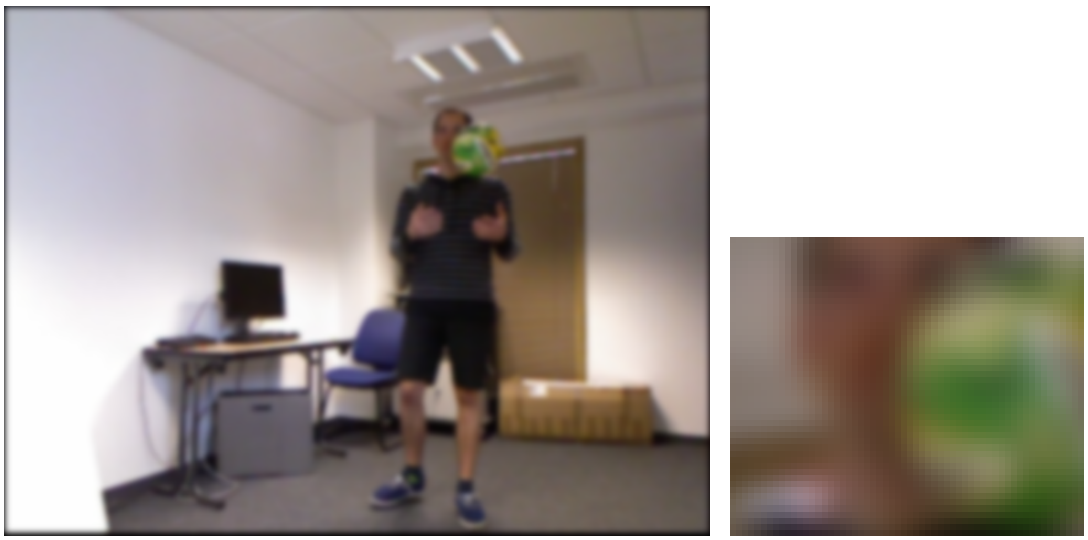
$\sigma_r = 0.031, \sigma_s = 2$ | $\sigma_r = 0.031, \sigma_s = 5$ | $\sigma_r = 0.031, \sigma_s = 15$

$\sigma_r = 0.063, \sigma_s = 2$ | $\sigma_r = 0.063, \sigma_s = 5$ | $\sigma_r = 0.063, \sigma_s = 15$

Figure A.2 – Résultats obtenus en faisant varier σ_r et σ_s.



(a) Image originale



(b) Filtre gaussien $\sigma_s = 7$

(c) Filtre bilatéral $\sigma_r = 0.3, \sigma_s = 7$

(d) Filtre bilatéral utilisant l'information de profondeur $\sigma_r = 0.03, \sigma_s = 15$

FIGURE A.3 – Comparaison qualitative entre image originale, filtrage gaussien, filtrage bilatéral standard et filtrage bilatéral utilisant l'information de profondeur.

Annexe B

Publications associées

Conférences

- **Antoine Letouzey**, Benjamin Petit et Edmond Boyer.
  Scene Flow from Depth and Color Images
  BMVC – British Machine Vision Conference, 2011.

- Benjamin Petit, **Antoine Letouzey**, Edmond Boyer et Jean-Sébastien Franco.
  Surface Flow from Visual Cues
  VMV – Vision, Modeling and Visualization Workshop, 2011.

- Benjamin Petit, **Antoine Letouzey** et Edmond Boyer.
  Flot de surface à partir d'indices visuels
  ORASIS – Congrès des jeunes chercheurs en vision par ordinateur, 2011.

- **Antoine Letouzey**, Benjamin Petit et Edmond Boyer.
  Flot de scène à partir d'images couleur et de cartes de profondeur
  RFIA – Reconnaissance de Formes et Inteligence Artificielle, 2012.

- **Antoine Letouzey** et Edmond Boyer.
  Progressive Shape Models
  CVPR – Computer Vision and Patern Recognition, 2012.

- **Antoine Letouzey** et Edmond Boyer.
  Modèles progressifs de forme
  CORESA – COmpression et REprésentation des Signaux Audiovisuels, 2012

Revues

- **Antoine Letouzey**, Benjamin Petit et Edmond Boyer.
  Flot de Scène
  Traitement du Signal, 2012

Bibliographie

[Ancuti 10] C.O. Ancuti, C. Ancuti & P. Bekaert. *CC-SIFT : Exploiting chromatic contrast forwide-baseline matching*. In Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on, pages 938 –941, 2010. 34

[Anguelov 05] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers & J. Davis. *SCAPE : Shape Completion and Animation of People*. ACM Trans. Graphics (Proc. SIGGRAPH), 2005. 82

[Barron 94] J.-L. Barron, D.-J. Fleet & S.S. Beauchemin. *Performance of Optical Flow Techniques*. International Journal of Computer Vision, 1994. 49, 51, 54

[Basha 10] T. Basha, Y. Moses & N. Kiryati. *Mutli-View Scene Flow Estimation : A View Centered Variational Approach*. In Computer Vision and Pattern Recognition, 2010. 51

[Bay 06] H. Bay, A. Ess, T. Tuytelaars & L. Van Gool. *SURF : Speeded Up Robust Features*. Computer Vision and Image Understanding, 2006. 56

[Belkin 08] M. Belkin, J. Sun & Y. Wang. *Discrete Laplace Operator on Meshed Surfaces*. In Proceedings of the Symposium on Computational Geometry, 2008. 62

[Belkin 09] M. Belkin, J. Sun & Y. Wang. *Constructing Laplace operator from point clouds in Rd*. In ACM -SIAM Symposium on Discrete Algorithms, pages 1031–1040, 2009.

[Bichlmeier 09] C. Bichlmeier, M. Kipot, S. Holdstock, S. M. Heining, E. Euler & N. Navab. *A Practical Approach for Intraoperative Contextual In-Situ Visualization*. In International Workshop on Augmented environments for Medical Imaging including Augmented Reality in Computer-aided Surgery (AMI-ARCS 2009), New York, USA, Sept. 2009. MICCAI Society. 16

[Boyer 06] E. Boyer. *On Using Silhouettes for Camera Calibration*. In Asian Conference on Computer Vision (ACCV '06), pages 1–10, Hyderabad, India, 2006. 20

[Bradley 08] D. Bradley, T. Popa, A. Sheffer, W. Heidrich & T. Boubekeur. *Markerless Garment Capture*. ACM Trans. Graphics (Proc. SIGGRAPH), vol. 27, 2008.

[Bradley 10] D. Bradley, W. Heidrich, T. Popa & A. Sheffer. *High Resolution Passive Facial Performance Capture*. ACM Trans. Graphics (Proc. SIGGRAPH), vol. 29, 2010.

[Bronstein 07] M. M. Bronstein A. M.and Bronstein & R. Kimmel. *Calculus of non-Rigid Surfaces for Geometry and Texture Manipulation*. IEEE TVCG, 2007. 83

[Brox 10] T. Brox & J. Malik. *Large displacement optical flow : descriptor matching in variational motion estimation*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010.

[Cagniart 10] C. Cagniart, E. Boyer & S. Ilic. *Probabilistic Deformable Surface Tracking From Multiple Videos*. European Conference on Computer Vision, 2010. 52, 83, 84, 90

[Carranza 03] J. Carranza, C. Theobalt, M. A. Magnor & H.-P. Seidel. *Free-Viewpoint Video of Human Actors*. In ACM SIGGRAPH, 2003. 15

[Cech 11] J. Cech, J. Sanchez-Riera & R.P. Horaud. *Scene Flow Estimation by Growing Correspondence Seeds*. In Computer Vision and Pattern Recognition, 2011.

[de Aguiar 08] E. de Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H.-P. Seidel & S. Thrun. *Performance Capture from Sparse Multi-view Video*. ACM Trans. Graphics (Proc. SIGGRAPH), vol. 27, 2008. 82, 83, 84

[Devernay 06] F. Devernay, D. Mateus & M. Guilbert. *Multi-Camera Scene Flow by Tracking 3-D Points and Surfels*. In Computer Vision and Pattern Recognition, 2006. 58

[Franco 03] J.-S. Franco & E. Boyer. *Exact Polyhedral Visual Hulls*. In British Machine Vision Conference, volume 1, 2003. 21

[Franco 08] J.-S. Franco & E. Boyer. *Efficient Polyhedral Modeling from Silhouettes*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2008. 96

[Furukawa 06] Y. Furukawa & J. Ponce. *Carved Visual Hulls for Image-Based Modeling*. In European Conference on Computer Vision, 2006.

[Furukawa 09] Y. Furukawa & J. Ponce. *Dense 3D Motion Capture for Human Faces*. IEEE Conference on Computer Vision and Pattern Recognition, 2009. 84

[Furukawa 10a] R. Furukawa, R. Sagawa, H. Kawasaki, K. Sakashita, Y. Yagi & N. Asada. *One-shot entire shape acquisition method using multiple projectors and cameras*. In Pacific-Rim Symposium on Image and Video Technology (PSIVT2010), pages 107–114, 2010. 11

[Furukawa 10b] Y. Furukawa & J. Ponce. *Accurate, Dense, and Robust Multi-View Stereopsis*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 8, 2010.

[Goldlueke 07] B. Goldlueke, I. Ihrke, C. Linz & M. Magnor. *Weighted Minimal Hypersurface Reconstruction*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, pages 1194–1208, 2007.

[Hansard 11] Miles Hansard, Radu Horaud, Michel Amat & Seungkyu Lee. *Projective Alignment of Range and Parallax Data*. In Computer Vision and Pattern Recognition, 2011. 29, 76

[Harris 88] C. Harris & M. Stephens. *A combined corner and edge detector*. In Alvey Vision Conference, 1988. 56

[Horn 81] B.K.P. Horn & B.G. Schunck. *Determining Optical Flow*. Artificial Intelligence, 1981. 49, 51, 58, 60, 61

[Horprasert 99] T. Horprasert, D. Harwood & L. S. Davis. *A Statistical Approach for Real-Time Robust Background Subtraction and Shadow Detection*. In IEEE International Conference on Computer Vision, pages 1–19, 1999. 20

[Huguet 07] F. Huguet & F. Devernay. *A Variational Method for Scene Flow Estimation From Stereo Sequences*. In International Conference on Computer Vision, 2007. 51

[Isard 06] M Isard & J MacCormick. *Dense Motion and Disparity Estimation via Loopy Belief Propagation*. In Asian Conference on Computer Vision, 2006. 51

[Janko 09] Z. Janko & J.-P. Pons. *Spatio-Temporal Image-Based Texture Atlases for Dynamic 3-D Models*. In Proceedings of the 7th International Conference on 3-D Digital Imaging and Modeling (ICCV Workshop), 2009. 111

[Kanade 97] T. Kanade & P.J. Rander P.and Narayanan. *Virtualized Reality : Constructing Virtual Worlds from Real Scenes*. In IEEE Multimedia, Immersive Telepresence, 1997. 12, 13

[Ladikos 08] A. Ladikos, S. Benhimane & N. Navab. *Real-time 3D Reconstruction for Collision Avoidance in Interventional Environments*. In International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), 2008. 16

[Laurentini 94] A. Laurentini. *The Visual Hull Concept for Silhouette-Based Image Understanding*. IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 16, February 1994. 20

[Lazebnik 01] S. Lazebnik, E. Boyer & J. Ponce. *On How to Compute Exact Visual Hulls of Object Bounded by Smooth Surfaces*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Press, Dec 2001. 20

[Lee 10] W Lee, W Woo & E Boyer. *Silhouette Segmentation in Multiple Views*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010. 31

[Li 08] R. Li & S. Sclaroff. *Multi-scale 3D Scene Flow from Binocular Stereo Sequences*. Computer Vision and Image Understanding, 2008. 51

[Li 09] Hao Li, Bart Adams, Leonidas J. Guibas & Mark Pauly. *Robust Single-view Geometry and Motion Reconstruction*. ACM Trans. Graphics (Proc. SIGGRAPH Asia), vol. 28, 2009. 84

[Li 11] H. Li, L. Luo, D. Vlasic, P. Peers, J. Popovič, M. Pauly & S. Rusinkiewcz. *Temporally Coherent Completion of Dynamic Shapes*. ACM Trans. Graphics (Proc. SIGGRAPH), vol. 30, 2011. 85

[Liu 08] C. Liu, J. Yuen, A. Torralba, J. Sivic & W. Freeman. *SIFT Flow : Dense Correspondence across Different Scenes*. In European Conference on Computer Vision, 2008. 52

[Lowe 04] D.G. Lowe. *Distinctive Image Features from Scale-invariant Keypoints*. International Journal of Computer Vision, 2004. 56

[Lucas 81] B.D. Lucas & T. Kanade. *An Iterative Image Registration Technique with an Application to Stereo Vision*. In International Joint Conference on Artificial Intelligence, 1981. 49, 67

[Marey 84] E.-J. Marey. Le Mouvement. G Manson, 1884. 9

[Microsoft 10] Microsoft. http://www.xbox.com/kinect, 2010. 11

[Mitra 07] N. J. Mitra, S. Flöry, M. Ovsjanikov, N. Gelfand, L. Guibas & H. Pottmann. *Dynamic Geometry Registration*. In Proceedings of the Symposium on Geometry Processing, SGP'07, 2007. 84

[Naveed 08] A Naveed, C Theobalt, C Rossl, S Thurn & H.P. Seidel. *Dense Correspondence Finding for Parametrization-free Animation Reconstruction from Video*. In Computer Vision and Pattern Recognition, 2008. 52

[Neumann 02] J. Neumann & Y. Aloimonos. *Spatio-Temporal Stereo Using Multi-Resolution Subdivision Surfaces*. International Journal of Computer Vision, 2002. 49, 51

[Paris 09] S. Paris, P. Kornprobst, J. Tumblin & F Durand. *Bilateral Filtering : Theory and Applications*. Foundations and Trends® in Computer Graphics and Vision, vol. 4, 2009. 115, 116

[Pekelny 08] Y. Pekelny & C. Gotsman. *Articulated Object Reconstruction and Markerless Motion Capture from Depth Video*. Computer Graphics Forum, vol. 27, 2008.

[Petit 09] B. Petit, J.-D. Lesage, E. Boyer & B. Raffin. *Virtualization Gate*. In Sigggraph - Emerging Technologies. ACM Press, August 2009. 17

[Petit 11a] B Petit. *Téléprésence, immersion et interactions pour le reconstruction 3D temps-réel*. These, Université de Grenoble, Feb 2011. 17

[Petit 11b] B. Petit, A. Letouzey, E. Boyer & J.S. Franco. *Surface Flow from Visual Cues*. In Vision, Modeling and Visualization Workshop, 2011.

[Piccardi 04] M. Piccardi. *Background subtraction techniques : a review*. In Systems, Man and Cybernetics, 2004 IEEE International Conference on, volume 4, 2004. 31

[Pons 05] J.-P. Pons, R. Keriven & O. Faugeras. *Modelling Dynamic Scenes by Registering Multi-View Image Sequences*. In Computer Vision and Pattern Recognition, 2005. 51, 78

[Pons 07] J.-P. Pons, R. Keriven & O. Faugeras. *Multi-view Stereo Reconstruction and Scene Flow Estimation with a Global Image-based Matching Score*. International Journal of Computer Vision, 2007.

[Popa 10] T. Popa, I. South-Dickinson, D. Bradley, A. Sheffer & W. Heidrich. *Globally Consistent Space-Time Reconstruction*. Computer Graphics Forum, vol. 29, 2010. 83, 85, 89

[PrimeSense 10] PrimeSense. *PrimeSensor™ Reference Design Datasheet*. http://www.primesense.com/?p=514, 2010.

[Rabe 10] C. Rabe, T. Müller, A. Wedel & U. Franke. *Dense, Robust, and Accurate Motion Field Estimation from Stereo Image Sequences in Real-Time*. In European Conference on Computer Vision, 2010. 51

[Seitz 06] S. Seitz, B. Curless, J. Diebel, D. Scharstein & R. Szeliski. *A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms*. In IEEE CVPR, 2006. 83

[Sharf 08] A. Sharf, D. A. Alcantara, T. Lewiner, C. Greif, A. Sheffer, N. Amenta & D. Cohen-Or. *Space-Time Surface Reconstruction using Incompressible Flow*. ACM Trans. Graphics (Proc. SIGGRAPH Asia), vol. 27, 2008. 85

[Sharma 11] A. Sharma, R. Horaud, J. Cech & E. Boyer. *Topologically-Robust 3D Shape Matching Based on Diffusion Geometry and Seed Growing*. In IEEE CVPR, 2011. 83

[Sinha 04] S.N. Sinha & M. Pollefeys. *Synchronization and Calibration of Camera Networks from Silhouettes*. In Pattern Recognition. ICPR 2004. Proceedings of the 17th International Conference on, pages 116 – 119 Vol.1, 2004. 20

[Sorkine 07] O. Sorkine & M. Alexa. *As-Rigid-As-Possible Surface Modeling*. In Eurographics Symposium on Geometry Processing, 2007. 58, 60, 61

[Starck 07a] J. Starck & A. Hilton. *Correspondence Labeling for Wide-Timeframe Free-Form Surface Matching*. In European Conference on Computer Vision, 2007. 56

[Starck 07b] J Starck & A Hilton. *Surface Capture for Performance-Based Animation*. IEEE Computer Graphics and Applications, 2007. 28, 33, 52, 64, 95

[Svoboda 05] T. Svoboda, D. Martinec & T. Pajdla. *A Convenient Multi-Camera Self-Calibration for Virtual Environments*. PRESENCE : Teleoperators and Virtual Environments, vol. 14, no. 4, 2005. 29

[Tomasi 98] C. Tomasi & R. Manduchi. *Bilateral Filtering for Gray and Color Images*. In International Conference on Computer Vision, pages 839 –846, 1998. 115

[Triesch 98] J. Triesch & C. Von Der Malsburg. *A Gesture Interface for Human-Robot-Interaction*. In International Conference on Automatic Face and Gesture Recognition, 1998. 16

[Valgaerts 10] L. Valgaerts, A. Bruhn, H. Zimmer, J. Weickert, C. Stoll & C. Theobalt. *Joint Estimation of Motion, Structure and Geometry from Stereo Sequences*. In European Conference on Computer Vision, 2010.

[Varanasi 08] Kiran Varanasi, Andrei Zaharescu, Edmond Boyer & Radu P. Horaud. *Temporal Surface Tracking Using Mesh Evolution*. In European Conference on Computer Vision, 2008. 52, 65

[Vedula 05] S. Vedula, S. Baker, P. Rander, R. Collins & T. Kanade. *Three-Dimensional Scene Flow*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005. 49, 51, 54, 55, 58, 67, 73, 74, 75

[Vlasic 08] D. Vlasic, I. Baran, W. Matusik & J. Popović. *Articulated Mesh Animation from Multi-view Silhouettes*. ACM Trans. Graphics (Proc. SIGGRAPH), vol. 27, 2008. 28, 31, 32, 34, 35, 83, 84

[Vlasic 09] D. Vlasic, P. Peers, I. Baran, P. Debevec, J. Popović, S. Rusinkiewicz & W. Matusik. *Dynamic Shape Capture using Multi-View Photometric Stereo*. ACM Transactions on Graphics, vol. 28, no. 5, page 174, 2009. 13

[Wand 07] Michael Wand, Philipp Jenke, Qixing Huang, Martin Bokeloh, Leonidas Guibas & Andreas Schilling. *Reconstruction of deforming geometry from time-varying point clouds*. In Proceedings of the Symposium on Geometry Processing, SGP'07, 2007. 85

[Wand 09] M. Wand, B. Adams, M. Ovsjanikov, A. Berner, M. Bokeloh, P. Jenke, L. Guibas, H.-P. Seidel & A. Schilling. *Efficient Reconstruction of Non-rigid Shape and Motion from Real-time 3D Scanner Data*. ACM Trans. Graphics (Proc. SIGGRAPH), vol. 28, 2009.

[Wardetzky 07] M. Wardetzky, S. Mathur, F. Kälberer & E. Grinspun. *Discrete Laplace Operators : No free lunch*. In Eurographics Symposium on Geometry Processing, 2007. 61

[Wedel 08] A. Wedel, C. Rabe, T. Vaudrey, T. Brox, U. Franke & D. Cremeres. *Efficient Dense Scene Flow from Sparse or Dense Stereo Data*. In European Conference on Computer Vision, 2008. 51

[Weinland 06] D. Weinland, R. Ronfard & E. Boyer. *Free Viewpoint Action Recognition using Motion History Volumes*. In Computer Vision and Image Understanding, 2006. 16, 37

[Xu 10] L. Xu, J. Jia & Y Matsushita. *Motion Detail Preserving Optical Flow Estimation*. In Computer Vision and Pattern Recognition, 2010. 49, 52, 62

[Zaharescu 09] Andrei Zaharescu, Edmond Boyer, Kiran Varanasi & Radu P. Horaud. *Surface Feature Detection and Description with Applications to Mesh Matching*. In Computer Vision and Pattern Recognition, 2009. 56

[Zaharescu 11] A. Zaharescu, E. Boyer & R. Horaud. *Topology-Adaptive Mesh Deformation for Surface Evolution, Morphing, and Multi-View Reconstruction*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, no. 4, 2011. 91, 92

[Zhang 01] Y Zhang & C. Kambhamettu. *On 3D Scene Flow and Structure Estimation*. In Computer Vision and Pattern Recognition, 2001. 51

[Zhang 04] Z. Zhang. *Camera Calibration with One-Dimensional Objects*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 26, no. 7, july 2004. 20

[Zheng 10] Q. Zheng, A. Sharf, A. Tagliasacchi, B. Chen, H. Zhang, A. Sheffer & D. Cohen-Or. *Consensus Skeleton for Non-rigid Space-time Registration*. Computer Graphics Forum, vol. 29, 2010. 84