

UNIVERSITÉ DE GENÈVE

Département d'informatique

FACULTÉ DES SCIENCES

Professeur Thierry Pun

FACULTÉ DES SCIENCES

ÉCONOMIQUES ET SOCIALES

Département de systèmes d'information

Professeur Nadia Magnenat-Thalmann

---

# **Anatomical and Kinematical Modelling of the Musculoskeletal System from MRI**

**THÈSE**

présentée à la Faculté des sciences de l'Université de Genève  
pour obtenir le grade de Docteur ès sciences, mention informatique

par

**Benjamin Gilles**

de

Bagnols-sur-Cèze (France)

Thèse N° XXXX

GENÈVE

Atelier de reprographie Uni-Mail

2007



# Acknowledgments

My first thanks and gratitude go to my supervisor Nadia Magnenat-Thalmann for her invaluable support during these five years at MIRALab. Many thanks to the jury members: Thierry Pun, Hervé Delingette and Gábor Székely to have accepted to be part of it and for manuscript reviewing.

I would like to thank the past and current phd students of the MIRALab medical group for their kindness and scientific collaboration: Caecilia Charbonnier, Clémentine Lo, Lydia Yahia-Cherif, MyungJin Kang and Lazhari Assassi. Thanks to Laurent Mocozet, Pascal Volino and Gwenael Guillard for their support and help. Great thanks to other MIRALab members for the friendly working atmosphere. I would also like to credit colleagues from EPFL - Lausanne: Daniel Thalmann, Ronan Boulic, Anderson Maciel, Sofiane Sarni and Ehsan Arbabi and from MEMCenter Bern: Stephen Ferguson and Salman Chegini for the enjoyable partnership within the CO-ME project all along these years.

This work would not have been possible without a very active participation from the clinical side: so, many thanks to Frank Kolo-Christophe, Duy N'Guyen, Jean-Paul Vallée and Rosalind Perrin from the Radiology Department, and Jacques Ménétreay and Hassan Sadri from the Orthopaedics Department of the University Hospital of Geneva for their enthusiasm and availability. Thanks (and sorry) to Laura Mueller for the hours spent on the manual segmentation. I am also grateful to all anonym volunteers that have accepted to take part in the studies.

Finally, I would like to sincerely thank my family and Caroline for their infallible support, trust and love despite the distance. Thanks to all of my friends from Lozère to have deflected me when it was necessary.

Thanks to the Swiss National Science Foundation for the financial support.

# Abstract

An advanced understanding of musculoskeletal disorders, through research and in-depth examination, is absolutely necessary in order to improve prevention and treatment. Indeed, musculoskeletal disabilities are some of the most spreading causes of pain. In the framework of the CO-ME (Computer Aided and Image Guided Medical Interventions) NCCR, our research group aims at understanding and visualising the functionalities of human articulations. Our current clinical case study is related to the search and treatment of early signs of hip osteoarthritis in young patients due to abnormal femoro-acetabular impingements (collisions). For this kind of disorder, a static analysis is not sufficient for the diagnosis and the pre-surgical planning for the reason that the pathology relates to the joint kinematics. This Phd work aims at investigating methods for the anatomical modelling of the different components of the musculoskeletal system (mainly bones, muscles, fat and skin) and for the kinematical analysis of the rigid and soft active motion of these components, from images. This includes the development of static and dynamic MRI processing techniques and their validation on lower limb joints of different individuals.

Nowadays, medical imaging technology allows the capture of internal motion with different modalities (CT, MRI, US). MRI is non invasive, and flexible as it is suitable for imaging both bones and soft tissues which is important in studying the musculoskeletal system. The anatomical and kinematical modelling of the musculoskeletal system from images deals with image segmentation which has been an intensive research field for many years. Direct approaches (thresholding, region growing, edge detection) fail to handle noisy images with a large amount of textural information such as MR images. Indirect approaches (registration, deformable models) use prior information (reference segmented datasets, anatomical and statistical information, texture database) to perform the segmentation. In this context, numerous methods based on image intensity, features or texture analysis with statistical and physical constraints were presented. However, for complex tasks, limitations of current techniques remain the absence of global analysis of the system (interrelationships between organs such as attachments and contacts are not taken into account). No existing technique allow a robust, time-efficient and quasi-automatic segmentation of bones/ muscles/ cartilage/ ligaments from MRI. Also, the lack of prior information and adequate geometric techniques prevent from using incomplete data such as multi-slices dynamic MRI.

In this thesis we will focus on discrete deformable models-based techniques allowing a good shape and motion control for large deformations, as well as light computational expenses for simulating complex and interrelated models. We show that various constraints (e.g. volume preserving, shape memory) can be added to the segmentation process with a great flexibility, through forces. The proposed work includes: 1) The definition of suitable representations for modelling musculoskeletal organs and their interrelationships 2) The introduction of proper internal and external forces to control models and allow a precise matching with image features 3) The investigation of evolution methods for fast and accurate convergence of several models (PDE resolution, multi-resolution approach, collision handling) 4) The introduction of high-level descriptors (medial surfaces) for compressing shape parameters 5) The testing of methods on various data (sequential MRI, real-time MRI, different protocols), different individuals, different postures, and different joints (hip and knee) in order to optimise acquisition protocols (dynamic planes position, resolution), to optimise image processing parameters (e.g. external forces parameters), to build a database of anatomical

and kinematical models and to validate the overall methodology. We demonstrate the efficiency of the proposed techniques over previous methods, in the following aspects: robustness, automation, accuracy, complexity and flexibility. In a long-term, we believe that our work could contribute to the medical field for the diagnosis, pre-planning and post-operative guide of musculoskeletal disorders related to both the morphology and the function. We also support the collection of registered models across the population which would be useful for anthropometric studies. We expect that the next step would be to bridge the simulation (prediction from physics) and modelling (estimation from data) domains through complexity-compliant models, the goal being to parameterise and validate personalised functional models.

# Résumé

## Problématique médicale

A cause de sa grande complexité géométrique, mécanique, chimique et physiologique, la machine humaine n'est toujours pas totalement comprise. Des événements de nature différente interviennent à des échelles spatiales (e.g. échanges chimiques/ interactions mécaniques) et temporelles (e.g. expression génétique/ transmissions neurales) très diverses. Par conséquent, l'étude biologique du fonctionnement humain a été fragmentée en différentes disciplines, malgré leurs interdépendances. Dans le cadre de cette thèse, nous nous intéressons aux aspects macroscopiques du système locomoteur, qui confère au corps la possibilité de se mouvoir dans des conditions favorables en fonction de stimulations neuromusculaires. L'étude du système locomoteur mêle des domaines scientifiques très différents: la médecine, la biologie, l'informatique, la mécanique et les sciences expérimentales.

Les pathologies liées au système locomoteur sont certainement les causes les plus courantes de douleur et de handicap physique à long terme, affectant plusieurs centaines de millions de personnes à travers le monde<sup>1</sup>. Grâce à la recherche, une compréhension avancée du système locomoteur et de ses troubles est absolument nécessaire pour améliorer la prévention et le traitement. Trois niveaux pathologiques peuvent être distingués : le premier est directement lié à la géométrie des organes (e.g. fracture, trauma, blessure musculaire/ ligamentaire) ; le deuxième concerne le niveau fonctionnel (e.g. dégénération articulaire, restriction du mouvement due à de trop courtes unités musculo-tendineuses) et le troisième niveau est lié au contrôle neuromusculaire (e.g. paralysie locale après accident cérébral). Dans le domaine du diagnostic et du traitement assisté par ordinateur, ces trois niveaux (anatomie, fonction et contrôle) correspondent à trois échelles de modélisation et de simulation. Comme le montre la Table 1, les modèles coopèrent avec les données à travers des processus de prédiction, de paramétrisation et de validation.

La capacité de prévoir le comportement locomoteur (e.g. cinématique squelettique, déformation des organes, dégénération des tissus, processus de reconstruction, etc.) à partir de la morphologie, de contraintes cinématiques, mécaniques et d'impulsions neuronales, aurait un impact très important sur la pratique médicale actuelle. En plus des applications en orthopédie (e.g. détection précoce d'arthroses, design de prothèses, planning d'ostéotomies, allongement de tendons, etc.), cela aidera en kinésiologie pour l'optimisation du mouvement (e.g. réduction des "tennis elbow"), la réhabilitation et l'ergonomie (minimisation de la fatigue physique). Dans le cadre de notre recherche, notre but médical est de rechercher les premiers signes d'arthrose de la hanche chez les jeunes patients. Notre objectif est de prédire les dégénération articulaires (principalement l'ossification du cartilage et du labrum) pour prévenir la pose de prothèse grâce à la chirurgie correctrice (résection de l'os). Notre hypothèse est que l'arthrose est due à des collisions répétées entre le fémur et l'acetabulum ("impingement syndrome"). Elles apparaissent lors de mouvements extrêmes et lorsque les os ont des morphologies pathologiques ("cam" ou "pincer"). Pour ce type de problème, une analyse statique n'est pas suffisante car la pathologie est liée à la cinématique (niveau fonctionnel).

---

<sup>1</sup><http://www.boneandjointdecade.org>

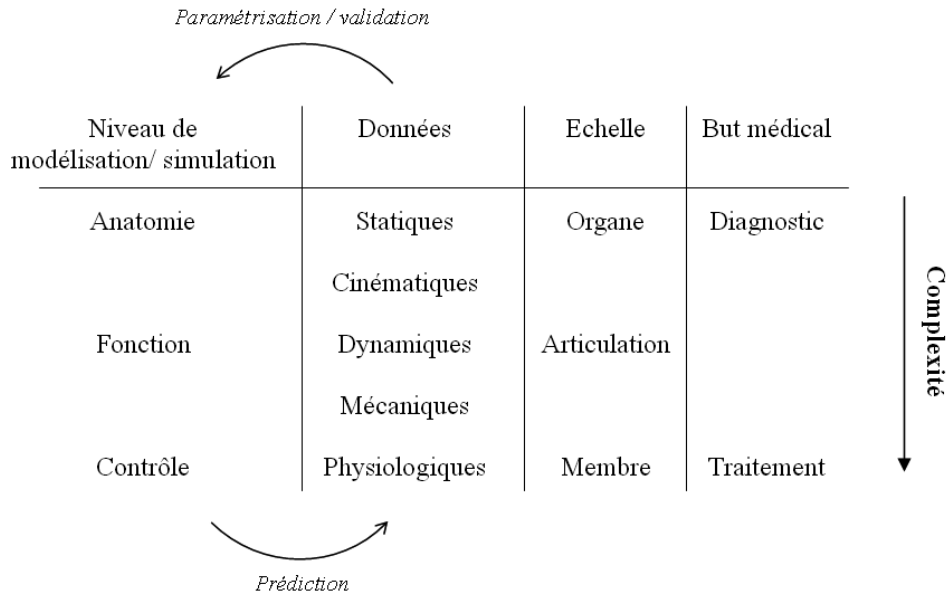


Table 1: Diagnostic et traitement assisté par ordinateur

Les outils de diagnostic utilisés de manière routinière en clinique, en particulier les scanners, deviennent de plus en plus précis, disponibles et standardisés, et également de moins en moins invasifs. Les données médicales sont obtenues de sources complémentaires: mis à part les images 3D multimodales traditionnelles (e.g. CT, IRM, US), les données cinématiques deviennent de plus en plus courantes (e.g. ciné IRM, IRM dynamique). Dans le cadre de leur travail, les radiologues doivent analyser une grande quantité de données liées à l'anatomie, la cinématique, la dynamique, la mécanique et la physiologie du système locomoteur, et doivent donc gérer et visualiser l'information à des niveaux de complexité grandissants. Dans ce contexte, les modèles anatomiques 3D (e.g. forme, surface, volume) et les modèles cinématiques 4D (e.g. angles articulaires, cartes de déformation) fournissent un meilleur aperçu que des piles d'images. Cela aide également à la fusion (le recalage) de données issues de différentes sessions/ modalités d'acquisition. La reconstruction de modèles virtuels grâce à la segmentation et le recalage d'images est par conséquent une étape fondamentale pour les outils futurs de diagnostic. La modélisation numérique a un double objectif: en plus du diagnostic, elle est utilisée pour la simulation (prédiction). Pour la plupart des applications, les variations entre les sujets doivent être capturées avec une grande précision pour évaluer correctement les problèmes locomoteurs (e.g. morphologie anormale). Cependant, certains paramètres ne changent pas beaucoup dans la population (e.g. lieu d'attachement des muscles), et des modèles génériques (normalisés) avec peu de paramètres spécifiques au patient sont suffisants pour certaines applications (e.g. optimisation du mouvement dans le sport).

Ce travail de thèse a pour but d'explorer des méthodes de modélisation et d'analyse cinématique des différents composants du système locomoteur (les os, les muscles, la graisse, la peau, le ligaments et les cartilages) en utilisant l'Imagerie par Résonance Magnétique (IRM). L'IRM est une technique non-invasive, et flexible car adaptée pour les os et les tissus mous, ce qui est important dans notre contexte. Ce travail inclut le développement et la validation de techniques de traitement d'images semi-automatiques et automatiques. Une priorité est mise sur l'applicabilité clinique de ces techniques (temps de calcul courts, utilisation de protocoles standards), et des cas d'étude sont présentés pour les membres inférieurs.

## Contexte technique

Hormis les méthodes empiriques, une pratique scientifique classique est de comprendre le fonctionnement humain à travers la création de modèles capables de reproduire de manière réaliste l'observable. La modélisation et la simulation se supportent mutuellement : les méthodes de simulation, basées sur des principes physiques, fournissent des schémas mathématiques dépendant de paramètres (par exemple, la forme des organes ou les paramètres physiques). Les techniques de modélisation ont pour but de quantifier ces paramètres dans un contexte particulier (individu, posture et instant). Certains paramètres ne sont pas significatifs à l'échelle spatiale et temporelle d'intérêt, et des hypothèses/ simplifications peuvent être faites. Cela est habituellement spécifié par l'application qui a pour but de répondre à une question bien définie avec une précision donnée. De plus, des contraintes techniques (puissance informatique, capacités de mesure) influencent la façon dont la complexité est gérée. Finalement, les méthodes de modélisation et de simulation sont évaluées par rapport aux données acquises grâce à des procédures de validation.

La modélisation anatomique et cinématique à partir d'images traite de segmentation d'images, qui est un domaine de recherche très actif depuis des années. Les approches directes (e.g. seuillage, accroissement de régions, détection de contours) ne permettent généralement pas de gérer des images bruitées possédant beaucoup d'information texturale comme les images IRM. Les approches indirectes (e.g. recalage, modèles déformables) sont plus robustes car elles utilisent une information à-priori (e.g. modèle de référence, information anatomique et statistique, base de données de texture). Dans ce contexte, de nombreuses méthodes basées sur l'intensité des images, sur des particularités structurelles ("features") ou sur les textures, utilisant des contraintes géométriques, statistiques et/ou physiques ont été présentées. Les contraintes de forme sont basées sur des hypothèses de régularité des surfaces, alors que les contraintes topologiques exploitent de l'information à-priori sur l'interrelation entre les organes. Lorsque des contraintes mécaniques sont utilisées, le pouvoir prédictif des méthodes de simulation est incorporé au processus de modélisation pour réduire considérablement le nombre de degrés de liberté. L'information statistique (exemples) peut être utilisée comme contrainte, en supposant qu'il y ait peu de variabilité dans les paramètres entre les individus et dans le temps.

La singularité du système locomoteur tient dans son grand nombre d'organes qui interagissent entre eux par contact et attachement (e.g. environ 20 muscles juste pour la cuisse), rendant la complexité du système en terme de topologie particulièrement significative. Pour l'instant, aucune méthode ne permet de segmenter de manière efficace le système locomoteur :

- Les méthodes de segmentation existantes sont principalement dédiées à des objets isolés, et peu d'attention a été donné à l'incorporation de contraintes topologiques (contacts et attachements) pour la segmentation multiple d'organes.
- Segmenter de manière robuste et automatique les os, les muscles, les cartilages, les ligaments à partir d'IRM n'est pour l'instant pas possible car les méthodes existantes manquent de contraintes pour les discriminer dans des images complexes et bruitées.
- Il n'y a pas d'outil robuste de segmentation du système locomoteur pour des images basse-résolution comme les IRMs statiques et dynamiques utilisées en clinique, où la résolution est réduite pour accélérer l'acquisition. En effet, les protocoles haute-résolution, utilisés dans de nombreuses études de recherche, ne sont pas applicables quotidiennement à cause des contraintes de disponibilité des scanners.

Concernant l'utilisation clinique de la segmentation du système locomoteur:

- Les outils de traitement d'images existants (semi-automatique) ne permettent pas un diagnostic assisté par ordinateur efficace en terme de temps de calcul. Par conséquent, la plupart des radiologues se basent encore uniquement sur les piles d'images.



- Des descripteurs de haut niveau pour les structures musculo-squelettiques ne sont pas disponibles pour exploiter efficacement les résultats de la segmentation. En particulier, l'extraction de paramètres quantitatifs et normalisés pour la comparaison inter-individus et dans le temps n'est pas disponible aujourd'hui.

## Objectifs

Les objectifs de cette recherche sont clairement dirigés par l'application. Comme expliqué ci-dessus, il y a un grand potentiel dans l'utilisation des nouvelles techniques d'acquisition pour la modélisation et la simulation du système locomoteur. Mais les méthodes de traitement d'images existantes manquent de précision, de robustesse, de vitesse et de flexibilité, car elles sont sous-contraintes. En effet, le but en segmentation est de réduire le nombre de degrés de liberté des modèles à travers l'utilisation de connaissance à-priori. En développant une nouvelle méthode de segmentation/ recalage, nous voulons permettre, à court terme :

- la segmentation rapide des muscles, des os, de la peau, de la graisse, des ligaments et des cartilages dans les IRMs conventionnels, pour le diagnostic assisté par ordinateur.
- le calcul des correspondances spatiales entre les individus (recalage inter-patient) pour l'analyse anthropométrique.
- la segmentation 4D d'IRM séquentiel et dynamique basse-résolution (recalage intra-patient) pour l'analyse cinématique des articulations.
- la description haut-niveau des formes pour l'analyse quantitative de la morphologie des organes.
- la génération de données du mouvement (cinématique des os, déformation des tissus mous) pour la validation et la paramétrisation de modèles biomécaniques.

L'objectif à long terme est de promouvoir:

- le diagnostic précoce des troubles locomoteurs, en soutenant les orthopédistes, les biomécaniciens et les kinésiologues avec des outils fiables.
- la formation médicale grâce à la création d'atlas 3D dynamiques, où les données sont organisées de manière sémantiques (ontologie liant les concepts médicaux et les données numériques).
- l'analyse morphologique et anthropométrique des organes, en rapport avec la collection de modèles de divers individus, créés par nos méthodes d'extraction de formes.
- le transfert de techniques en traitement d'images vers l'hôpital, grâce à l'utilisation de protocoles cliniques et de méthodes de traitement quasi-automatiques.
- la connexion entre les domaines de recherche comme la biomécanique, l'informatique graphique et l'imagerie médicale. En particulier, pour le système locomoteur, la compréhension de la relation physiologie/ dynamique passe par une recherche pluridisciplinaire, aujourd'hui fragmentée.

## Contribution

Dans cette thèse, nous nous concentrons sur les méthodes de segmentation basées sur des modèles déformables discrets, qui permettent un bon contrôle de la forme pour des larges déformations ainsi que des coûts de calcul légers pour la simulation de modèles complexes et inter-reliés. Nous montrons que des contraintes diverses (e.g. préservation de volume, mémoire de forme, etc.) peuvent être ajoutées au processus de segmentation avec une grande flexibilité, grâce à différentes forces internes. Les images permettent de calculer des forces

externes et d'attirer les modèles vers les zones d'intérêt. Nous démontrons cela par une étude utilisant des IRMs statiques et dynamiques. De plus, nous présentons des méthodes pour décrire l'évolution des formes dans le temps et entre les individus. Particulièrement, ce travail inclut la définition de :

- Représentations adaptées pour la modélisation des organes du système locomoteur et de leurs relations topologiques: modèles simplexes inter-connectés.
- Forces internes pour la régularisation des modèles (e.g. lissage de surface, conservation de volume, mémoire de forme)
- Forces externes permettant un recalage précis entre les modèles et les structures d'intérêt dans les images (frontières des organes) et le respect des contraintes topologiques (e.g. non-pénétration, attachements).
- Méthodes d'évolution multi-échelles pour la convergence rapide et précise des modèles (résolution des EDP, approche multi-résolution).
- Descripteurs haut-niveau (surfaces médiales) pour la compression des paramètres et l'amélioration de l'analyse des formes (analyse du mouvement, étude longitudinale, comparaison inter-patient, extraction de paramètres biomécaniques) et pour permettre une analyse statistique (approche basée sur des exemples).

Ce travail a été réalisé en étroite collaboration avec des médecins radiologues et orthopédistes (Hôpitaux Universitaires de Genève). Le développement de protocoles IRM a été fait conjointement avec nos techniques de traitement d'images, le but étant d'obtenir des images issues de protocoles cliniques portant suffisamment d'information pour nous. Dans le but d'optimiser les protocoles d'acquisition (e.g. position des plans dynamiques, résolution des images), d'optimiser les paramètres de traitement d'image (e.g. profils d'intensité), de créer une base de données anatomique et cinématique et de valider nos méthodes, nous avons testé nos techniques sur des images très diverses (e.g. IRMs radiaires. IRMs dynamiques temps-réel, différentes séquences), différentes articulations (hanche et genou) et différents individus (sains ou pas) dans différentes postures (neutres/ extrêmes)

Nos méthodes utilisent les travaux précédents sur les modèles déformables de Delingette et Montagnat [Del99] [MD00] [MD05], dont nous empruntons la définition mathématique des modèles simplexes, et certaines méthodes de calcul des forces internes et externes. Nous les appliquons à systèmes plus larges, et essayons de réduire la complexité à travers différents niveaux de détails, des représentations médiales et des techniques efficaces pour la simulation de systèmes de particules. De plus, notre travail s'inspire de la recherche en simulation d'habits (Volino et al. [VMT00b] [VMT00a]) où de grandes avancées ont été faites en simulation physique rapide de modèles déformables en contact.

En rapport avec les méthodes précédentes, nous revendiquons les avancées suivantes:

- la possibilité de segmenter le système locomoteur d'une façon quasi-automatique et rapide.
- l'introduction, dans le processus de segmentation, de nouvelles contraintes de topologie et de forme basée sur des considérations anatomiques (volume, épaisseur locale, non-pénétration, attachements).
- l'optimisation du compromis complexité/ précision grâce à des forces multi-résolution, un traitement des collisions hiérarchique et une approche globale/ locale pour le recalage.
- La réduction des paramètres pour la caractérisation des formes des tissus biologiques en utilisant une représentation basée sur les surfaces médiales qui est réversible.

## Résultats

Nous démontrons l'efficacité des techniques proposées par rapport aux méthodes précédentes dans les aspects suivants :

- **Robustesse:** Notre méthode est capable de gérer des déformations et des variabilités inter-patient relativement larges, et également des images incomplètes et sous-échantillonnées comme les IRMs dynamiques.
- **Automation:** à partir de peu de paramètres de l'utilisateur (radiologue) : environ 15 points placés en 2 minutes, notre méthode est capable de fournir des résultats précis. L'interaction est possible pour assister le processus automatique.
- **Précision:** à travers une validation adéquate, nous montrons que la précision de notre méthode est de l'ordre du millimètre. En combinant des contraintes radiaires et de forme, les erreurs de glissement de surface (recalage de frontières) sont réduites. Nous analysons la précision par rapport à des méthodes classiques en infographie (déformation basée sur le squelette, relaxation à partir de forces internes), et des méthodes de segmentation interactives/ manuelles.
- **Complexité:** Grâce à notre technique multi-résolution, la segmentation d'environ 20 muscles, 4 os, 2 cartilages, 4 ligaments et de la peau prend environ *30min* avec une précision de *1.5mm*. La représentation basée sur l'axe médian permet de représenter les tissus mous avec environ 15 fois moins de paramètres que des techniques classique basée sur les surfaces (erreur=*0.5mm*).
- **Flexibilité:** la méthode proposée est relativement générique en terme de type d'articulation et d'organe.

## Limitations et perspectives

Malgré les avancées attribuables aux méthodes que nous présentons, il y a un certain nombre de limitations liées à la méthodologie adoptée. L'expérience acquise durant ce projet nous permet de mettre en évidence des aspects critiques et des problèmes qui devront recevoir une attention particulière dans des travaux futurs. Plus de travail est également requis en terme de test et d'application.

- **Traitement d'images:** nous nous sommes concentré sur des aspects géométriques et numériques mais nous pensons que les aspects "traitement d'images" seraient une source importante d'amélioration, puisqu'ils sont la source principale d'erreurs et de coûts de calcul. Notamment, nous pensons que les aspects multi-résolution, statistiques et liés à la parallélisation seraient intéressant à explorer.
- **Géométrie:** nous avons noté certaines limitations dans des cas extrêmes: lorsque les différences sont telles que le recalage n'est plus robuste (postures extrêmes, larges variabilité inter-patient). L'utilisation d'un modèle de référence unique est une limitations importante qui pourrait être réduite par une base de donnée de référence et une étude statistique adéquate.
- **Simulation mécanique:** dans le but de relier la modélisation et la simulation, des nouvelles contraintes basées sur la physique des milieux continus devraient être introduites. Cela permettrait de valider et de paramétrer les modèles biomécaniques.
- **Application:** des tests plus complets devraient être effectués dans le but de fournir une base données exhaustive (plus d'exemples pathologiques et de patterns de stimulation musculaire sont nécessaires).

# Contents

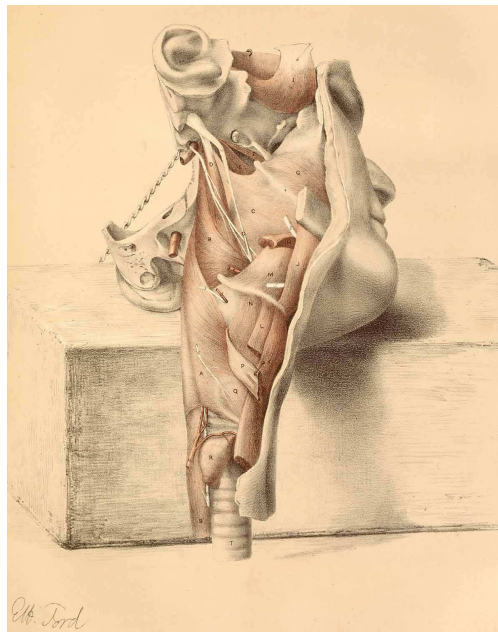
<b>1</b>	<b>Introduction</b>	<b>13</b>
1.1	Medical Problematic . . . . .	14
1.2	Technical context . . . . .	15
1.3	Objectives . . . . .	16
1.4	Contribution . . . . .	17
1.5	Organisation of this document . . . . .	18
1.6	Copyright notice . . . . .	18
<b>2</b>	<b>The human musculoskeletal system</b>	<b>20</b>
2.1	Musculoskeletal anatomy . . . . .	22
2.1.1	Overview . . . . .	22
2.1.2	Classification of joints . . . . .	23
2.2	Biological tissue modelling . . . . .	25
2.2.1	Tissue composition . . . . .	25
2.2.2	Constitutive models of biological tissue . . . . .	25
2.2.3	Mechanical simulation . . . . .	30
2.3	Acquisition modalities . . . . .	32
2.3.1	Static acquisition . . . . .	32
2.3.2	Kinematics and dynamics acquisition . . . . .	35
2.3.3	Physiology acquisition . . . . .	39
2.3.4	Mechanical testing . . . . .	39
2.3.5	Summary . . . . .	40
2.4	Bone modelling and simulation . . . . .	40
2.4.1	Anatomy . . . . .	40
2.4.2	Volumetric representations . . . . .	42
2.4.3	Surfacic representations . . . . .	43
2.4.4	Stick-figure representations . . . . .	43
2.4.5	Discussion . . . . .	44
2.5	Muscle/tendon modelling and simulation . . . . .	44
2.5.1	Anatomy . . . . .	44
2.5.2	Muscle actuation modelling . . . . .	45
2.5.3	Action line representations . . . . .	47
2.5.4	Volumetric representations . . . . .	48
2.5.5	Hybrid representations . . . . .	50
2.5.6	Discussion . . . . .	52
2.6	Cartilage modelling and simulation . . . . .	53
2.6.1	Cartilage Anatomy . . . . .	53
2.6.2	Cartilage representations . . . . .	53
2.7	Ligaments modelling and simulation . . . . .	54
2.7.1	Ligament anatomy . . . . .	54

2.7.2	Ligament representations . . . . .	55
2.8	Fat modelling and simulation . . . . .	56
2.8.1	Fat anatomy . . . . .	56
2.8.2	Fat representations . . . . .	56
2.9	Skin modelling and simulation . . . . .	56
2.9.1	Skin anatomy . . . . .	56
2.9.2	Skin representations . . . . .	57
2.9.3	Free-form skin deformation . . . . .	57
2.9.4	Skeleton-driven skin deformation . . . . .	58
2.9.5	Muscle-driven skin deformation . . . . .	58
2.9.6	Example-based deformation . . . . .	59
2.9.7	Summary . . . . .	60
2.10	Integrating musculoskeletal tissues . . . . .	60
2.11	Conclusion and discussion . . . . .	63
<b>3</b>	<b>Medical images analysis</b>	<b>65</b>
3.1	Introduction . . . . .	66
3.2	Direct segmentation . . . . .	67
3.2.1	Pre-processing . . . . .	67
3.2.2	Region detection . . . . .	68
3.2.3	Classification . . . . .	68
3.3	Image Registration . . . . .	69
3.3.1	Introduction . . . . .	69
3.3.2	Registration features . . . . .	70
3.3.3	Similarity measure: external forces/ energy . . . . .	75
3.3.4	Regularisation: parameterisation and internal forces/ energy . . . . .	76
3.3.5	Numerical resolution: model evolution . . . . .	82
3.4	Conclusion . . . . .	88
<b>4</b>	<b>Simplex meshes</b>	<b>89</b>
4.1	Introduction . . . . .	90
4.2	Background . . . . .	90
4.3	Duality . . . . .	91
4.4	Geometric and topological quality . . . . .	92
4.4.1	Basic operators . . . . .	92
4.4.2	The exchange operation . . . . .	94
4.4.3	Local resolution change . . . . .	95
4.4.4	Global resolution change: multi-resolution schemes . . . . .	98
4.4.5	Results and discussion . . . . .	99
4.5	Geometry . . . . .	105
4.5.1	Background . . . . .	105
4.5.2	Alternative definition for 2-simplex meshes . . . . .	105
4.5.3	Curvature . . . . .	107
4.6	Internal constraints . . . . .	109
4.6.1	Smoothing . . . . .	109
4.6.2	Shape memory . . . . .	111
4.6.3	Volume preservation . . . . .	111
4.6.4	Radial constraints . . . . .	112
4.7	External constraints . . . . .	116
4.7.1	Constraints from images . . . . .	116
4.7.2	Collision handling . . . . .	119
4.8	Evolution . . . . .	125

4.8.1	Forces . . . . .	125
4.8.2	Multi-resolution and global regularisation . . . . .	132
4.8.3	Tests . . . . .	133
4.9	Conclusion . . . . .	144
<b>5</b>	<b>Application to musculoskeletal MRI segmentation</b>	<b>145</b>
5.1	Problematic . . . . .	146
5.2	Acquisition protocols and description of the data . . . . .	146
5.2.1	Static acquisition . . . . .	147
5.2.2	Sequential acquisition . . . . .	148
5.2.3	Dynamic acquisition . . . . .	150
5.3	Pre-processing . . . . .	151
5.4	Construction of the generic model . . . . .	154
5.4.1	Interactive segmentation . . . . .	154
5.4.2	Definition of attachments . . . . .	154
5.4.3	Medial axis generation . . . . .	155
5.5	Musculoskeletal modelling from static MRI . . . . .	158
5.5.1	Processing scheme . . . . .	158
5.5.2	Optimising parameters . . . . .	159
5.5.3	Results and validation . . . . .	164
5.5.4	Morphological analysis . . . . .	175
5.6	Joint coordinate systems . . . . .	177
5.7	Functional hip joint center computation . . . . .	180
5.7.1	Hip joint center initialisation . . . . .	181
5.7.2	Collision detection and femoro-acetabular movement simulation . . . . .	182
5.7.3	Hip joint center estimation . . . . .	183
5.8	Bone motion extraction . . . . .	185
5.8.1	Overview . . . . .	185
5.8.2	Tracking in sequential MRI . . . . .	188
5.8.3	HJC validation . . . . .	188
5.8.4	Tracking in dynamic MRI . . . . .	190
5.9	Skinning . . . . .	194
5.10	Extraction of soft-tissue deformations . . . . .	195
5.11	Conclusion . . . . .	199
<b>6</b>	<b>Conclusion</b>	<b>202</b>
6.1	Discussion . . . . .	203
6.2	Benefits . . . . .	203
6.3	Limitations and future work . . . . .	204
6.3.1	Image processing . . . . .	204
6.3.2	Geometry . . . . .	204
6.3.3	Mechanical simulation . . . . .	205
6.3.4	Application and testing . . . . .	205
6.4	Long-term vision and technology transfer . . . . .	206

# Chapter 1

## Introduction



---

<sup>1</sup>George Viner Ellis and George Henry Ford. London, 1867, Chromolithograph, National Library of Medicine. [http://www.nlm.nih.gov/exhibition/dreamanatomy/da\\_g\\_II-C-03.html](http://www.nlm.nih.gov/exhibition/dreamanatomy/da_g_II-C-03.html)

## 1.1 Medical Problematic

Because of its huge geometrical, mechanical, chemical and physiological complexity, the human machine is not yet comprehensively understood. Events, various in nature, occur at very different spatial scales (e.g. chemical exchanges vs. mechanical interaction) and temporal scales (e.g. gene expression vs. neural transmission). Consequently, the study of human functioning has been compartmentalised in various disciplines, though these are not independent. In the framework of this thesis, we are interested in the musculoskeletal (or motor) system, composed of muscles, bones, and connective tissues, giving to joints (or articulations) the ability to move in favourable conditions, according to neuromuscular stimulations. The study of the musculoskeletal system encompasses various science domains ranging from the medical field to computer graphics, mechanics and experimental sciences.

Musculoskeletal disorders are certainly the most notorious and common causes of severe long-term pain and physical disability, affecting hundreds of millions of people throughout the world<sup>1</sup>. Through research, an advanced understanding of the musculoskeletal system and its disorders is absolutely necessary in order to improve prevention and treatment. Three pathological levels can be distinguished: the first is directly related to tissue geometry (e.g. bone fracture, muscle/ tendon/ ligament injuries); the second is related to musculoskeletal function (e.g. joint degeneration, movement restriction due to tight musculotendon units); and the third is related to neuromuscular control (e.g. cerebral palsy). In the domain of computer-aided diagnosis and treatment, those three levels (anatomy, function and control) correspond to three scales where modelling and simulation methods take place. As shown in Table 1.1, models cooperate with data through prediction, parameterisation and validation processes.

Modelling/ simulation level	Data	Scale	Medical outcome
Anatomy	Static	Organ	Diagnosis
	Kinematics		
Function	Dynamics	Joint	
	Mechanics		
Control	Physiology	Limb	Treatment

*Parameterisation/ validation*

*Prediction*

**Complexity**

Table 1.1: Computer-aided diagnosis and treatment

The ability to predict musculoskeletal behaviour (e.g. bone kinematics, tissue deformation, tissue degeneration, tissue reconstruction, etc.) from morphology, kinematical constraints, mechanical constraints or

<sup>1</sup><http://www.boneandjointdecade.org>



neuromuscular impulses would have a great impact to the current medical practice. In addition to applications in orthopaedics (e.g. early detection of osteoarthritis, prosthesis design, osteotomy planning, tendon lengthening, ligament reconstruction, etc.), this would aid in the kinesiology science for movement optimisation, (e.g. reduction of tennis elbow), rehabilitation and ergonomics (minimisation of physical fatigue under specific constraints). In the framework of our research, our interest lies in the search and treatment of early signs of hip osteoarthritis in young patients. Our goal is to predict joint degeneration (mainly cartilage and labrum ossification) early enough to prevent from total hip replacement (prosthesis implant), and to treat it through bone resection. Our hypothesis is that repetitive femoro-acetabular impingements (bone collisions) can lead to osteoarthritis. They may occur in extreme movements because of bone pathological morphology (collisions when reaching joint range of motion limits). For this kind of disorder, a static analysis is not sufficient for the reason that the pathology relates to the joint kinematics (functional level).

Diagnosis tools used in the daily medical practice, especially medical scanner, are becoming increasingly precise, available, standardised, as well as less and less invasive. Medical data may be obtained from different complementary sources. Besides traditional 3D images available from various modalities (e.g. CT, MRI, US), kinematical data are now getting more accessible (e.g. cine MRI, real-time MRI). Indeed, in the course of their work, radiologists are required to analyse large amounts of data related to musculoskeletal anatomy, kinematics, dynamics, mechanics and physiology, and must therefore manage and visualise information at increasing levels of complexity. In this context, 3D anatomical models (e.g. shape, surface, volume) and 4D kinematical models (e.g. joint angles, deformation maps) provide more insight than image stacks. They can also help in fusing (registering) data from diverse modalities or scan sessions. Virtual model reconstruction through medical image segmentation and registration is consequently fundamental in the forthcoming diagnosis tools. Computer modelling is two-fold: besides diagnosis, models may be used for prediction (computer simulation). For most application, inter-subject variations need to be accurately captured, in order to assess musculoskeletal problems correctly (e.g. abnormal morphology). However, some parameters do not change much across individuals (e.g. muscle attachments location), and generic (normalised) musculoskeletal models with a few patient-specific data are sufficient for some applications (e.g. movement optimisation in sport science).

This Phd work aims at investigating methods for the anatomical modelling and kinematical analysis of the different components of the musculoskeletal system (bones, muscles, fat, skin, ligaments and cartilages) from Magnetic Resonance Images (MRI). MRI is non invasive, and flexible as it is suitable for imaging both bone and soft tissues which is important in studying the musculoskeletal system. This work includes the development and validation of semi-automatic and automatic image processing techniques. A focus is put on the clinical applicability of the techniques (short processing time, use of standard MRI protocols) and case studies are presented for lower limb joints.

## 1.2 Technical context

Besides empiric methods, a common scientific practice is to understand human functioning through the creation of models, able to realistically reproduce the observable. Modelling and simulation are mutually supporting: simulation methods, based on physical principles, provide mathematical schemes depending on parameters (for instance, organ shapes, mechanical parameters, etc.). Modelling techniques aim at quantifying these parameters in a particular context (specific individual, posture and instant). Some parameters are not relevant at the spatial and temporal scale of interest, so that hypothesis and simplifications can be done. This is usually driven by the application (specifications), aimed at answering to a well-defined question with a targeted accuracy. Moreover, technical constraints (computer facilities, measurement capabilities) influence the way complexity is handled. Finally, modelling and simulation methods are evaluated towards acquired data through validation procedures.

Anatomical and kinematical modelling from images deals with image segmentation, which has been an intensive research field for many years. Direct approaches (e.g. thresholding, region growing, edge detection) generally fail to handle noisy images with a large amount of textural information such as MR images. Indirect approaches (e.g. registration, deformable models) are more robust through the use of prior information (e.g. reference models, anatomical and statistical information, texture database). In this context, numerous methods based on image intensity, features or texture analysis with geometrical, statistical or physical constraints have been presented. Shape constraints rely on assumptions about surface regularity, while topological constraints exploit prior knowledge about organ interrelationships. When using mechanical constraints, the predictive power of simulation methods is incorporated into the modelling process to reduce considerably the number of degrees of freedom. Statistical information (examples) may be used as constraints, assuming small intra subject variability along time and inter subject variability across the population.

The specificity of the musculoskeletal system lies in that a large number of organs are interacting together through contacts and attachments (e.g. about 20 muscles just for the thigh), making system complexity, in terms of topology, particularly significant. Currently, no method allows cost efficient musculoskeletal segmentation:

- Existing segmentation tools are mostly dedicated to single objects, and little attention is given to the incorporation of topological constraints (contact and attachments) for multiple organ segmentation.
- Robust and automatic segmentation of bones, muscles, cartilages and ligaments from MRI is not available. Current methods lack of constraints for discriminating them in complex and noisy images.
- There is no robust musculoskeletal segmentation tool from low-resolution images, such as clinical static and dynamic MRI data, where resolution is decreased to speed up acquisition (e.g. multi-slice real-time MRI). Indeed, high-resolution images, as used in many research studies, are often not achievable in the daily practice due to limited scanner availability.

Regarding the clinical use of musculoskeletal segmentation:

- Existing tools (semi-automatic) do not permit time-efficient, thus daily computer-aided diagnosis through image processing. Radiologists still mostly use image stacks.
- High-level descriptors of musculoskeletal structures are not available to efficiently exploit segmentation results. Particularly, the extraction of quantitative and normalised parameters for inter-subject shape and motion comparison is not possible today.

### 1.3 Objectives

The objectives of this research are clearly application-driven. As explained before, there is a great potential in using new acquisition techniques for musculoskeletal modelling and simulation. But existing image processing methods lack of accuracy, robustness, speed and flexibility, because they are under constrained. Indeed, the goal is to reduce the number degrees of freedom of musculoskeletal models in MRI through new prior information. Developing a new segmentation/ registration method, we aim at allowing, in short term:

- the fast segmentation of muscles, bones, skin, fat, ligaments and cartilages in conventional MRI, for computer aided diagnosis.
- spatial correspondences computation across individuals (inter-patient registration) for anthropometric analysis.
- 4D segmentation of sequential and low resolution dynamic MRI (intra-patient registration), for joint kinematical analysis.

- high-level shape description, for quantitative analysis of organs.
- motion data creation (bone kinematics, soft-tissue deformation), for validating/ parameterising biomechanical models.

The long-term objective is to promote:

- The early diagnosis of musculoskeletal disorders, by supporting orthopaedists, biomechanicians and kinesiologists with reliable tools.
- Medical training through the creation of 3D dynamic atlas, where data is semantically organised (ontology).
- Organ morphological and anthropometric analysis, with regards to a collection of models from diverse individuals, created by our shape extraction techniques.
- The transfer of image processing techniques to the hospital, through the use of clinical protocols and quasi automatic processing methods.
- The connection between biomechanics, computer graphics and medical imaging research. Especially for the musculoskeletal system, to understand the relation between physiology and dynamics with reference to measured data.

## 1.4 Contribution

In this thesis, we focus on discrete deformable models-based segmentation techniques allowing a good shape control for large deformations, as well as light computational expenses for simulating complex and interrelated models. We show that various constraints (e.g. volume preserving, shape memory) can be added to the segmentation process with a great flexibility, through forces. External forces can be also derived from images. This is demonstrated by a study of the musculoskeletal system with static and dynamic MRI. In addition, we present methods to describe shape changes across individuals and along time. Particularly, this work includes the definition of:

- Suitable representations for modelling musculoskeletal organs and their interrelationships: interconnected simplex meshes.
- Proper internal forces to regularise musculoskeletal models (e.g. surface smoothing, volume conservation, shape memory)
- External forces allowing a precise matching of models with image features (e.g. organ boundaries) and topological constraints (e.g. non-penetration, attachment).
- Evolution methods for fast and accurate coarse-to-fine convergence of models (PDE resolution, multi-resolution approach).
- High-level descriptors (medial surfaces) for compressing parameters, improving shape analysis (motion analysis, longitudinal studies, inter-patient comparisons, biomechanical parameters extraction) and allowing statistical methods to be applied (example-based approach to handle incomplete data such as multi-slice real-time MRI).

This work has been done in close collaboration with physicians from radiology and orthopaedics (University hospital of Geneva). MRI protocol developments have been performed with reference to our work in image processing. The goal was to obtain images carrying sufficient information with clinically achievable protocols

(fast). In order to optimise acquisition protocols (dynamic planes position, resolution), to optimise image processing parameters (e.g. external forces parameters), to build a database of anatomical and kinematical models and to validate the overall methodology, we have tested our methods on various images (e.g. radial MRI, real-time MRI, different protocols), dissimilar joints (hip and knee) and different individuals (healthy or not) in different postures (from neutral to extreme).

Our methods make use of previous work on deformable models by Delingette and Montagnat [Del99] [MD00] [MD05], from which we undertake the simplex meshes framework: mathematical definition, internal and external forces computation. We apply it to larger systems, and try to reduce complexity using levels of details (LODs), medial axis and efficient particle system simulation techniques. In addition, our work is inspired from cloth simulation research (Volino et al. [VMT00b] [VMT00a]) where there have been great advances in the efficient and physical simulation of colliding deformable surfaces.

With regards to previous methods, we claim the following advances:

- The ability to segment the musculoskeletal system quasi-automatically and in a time-efficient fashion.
- The introduction, in the segmentation process, of new topological and shape constraint forces based on anatomical considerations (volume, local thickness, non-penetration, attachments).
- The optimisation of the complexity/accuracy trade-off through multi-resolution forces, multi-resolution hierarchical collision handling and a coarse-to-fine approach.
- The reduction of parameters dimension for characterising the shape of biological soft-tissues using medial surfaces representation (reversible transformation).

## 1.5 Organisation of this document

This manuscript is composed of four nearly equal parts. The two firsts are dedicated to the existing work and broadly review the field. Chapter 2 is related to musculoskeletal modelling and simulation. In addition to the prerequisite anatomical know-how, it encompasses acquisition modalities and the various mathematical representations used to model musculoskeletal tissues (bones, muscles, cartilages, ligaments, fat and skin). Simulation methods, that predict tissue behaviour (using geometrically-based to physically-based constraints) are also discussed. In Chapter 3, we speak more specifically about processing techniques for segmenting and registering images (especially medical images). As most techniques are generic in terms of tissue to segment, we do not focus on musculoskeletal tissues but rather highlight relevant constraints to apply in this domain. Also, deformable model-based registration approaches are mostly detailed (our methods belongs to this class of techniques). The two remaining chapters deal with our work in the field of musculoskeletal MRI registration. The basic mathematical methods are presented in Chapter 4. We have enhanced and adapted the simplex mesh representation, and investigated new techniques for computing deformations. Finally, we evaluate them in Chapter 5 through extensive testing on clinical MRI datasets. Application specific techniques and limitations are presented along with the benefits in terms of accuracy, computational cost, flexibility and automation.

## 1.6 Copyright notice

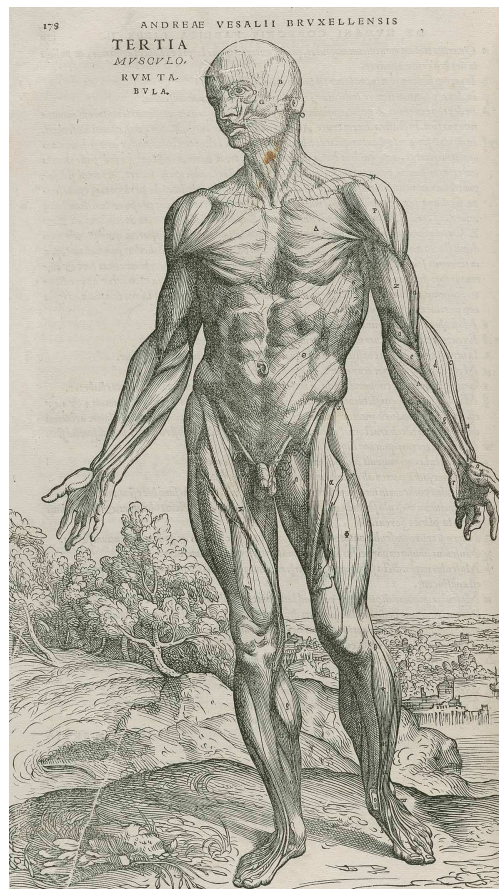
This document is ©2007 by Benjamin Gilles. The following text may be freely duplicated and distributed so long as no consideration is received in return, and this copyright notice remains intact.

This work has been supported by the CO-ME project (Computer Aided and Image Guided Medical Interventions) funded by Swiss National Research Foundation. Images from others are freely duplicated on the

courtesy of the cited authors/ editors. First page illustrations have been taken from the Dream Anatomy website from the U.S. National Library of Medicine. Developments have been performed in C++ using Microsoft Visual Studio .NET, and the open source toolkit VTK (©1993-2006 Ken Martin, Will Schroeder, Bill Lorensen, All rights reserved). This document has been compiled using L<sup>A</sup>T<sub>E</sub>X.

## Chapter 2

# The human musculoskeletal system



<sup>1</sup>Andreas Vesalius. De Humani Corporis Fabrica. Basel, 1543, Woodcut, National Library of Medicine. [http://www.nlm.nih.gov/exhibition/dreamanatomy/da\\_g\\_I-B-1-07.html](http://www.nlm.nih.gov/exhibition/dreamanatomy/da_g_I-B-1-07.html)

The musculoskeletal system is composed of numerous, complex, heterogeneous and imbricated organs. Chemical, electrical and mechanical phenomena contribute to the creation of movement. Muscles, submitted to neuro stimulations, contract and produce forces on bones, articulated at joints. Connective soft tissues, such as ligaments and cartilages, protect joints. By observing musculoskeletal geometry, tissue deformations, skeletal motions, and neuromotor electrical signals, scientists have tried to understand and predict (simulate) them through mathematical models, with reference to measured data (validation). From physiology to kinematics, different levels of prediction are possible as shown in Figure 2.1.

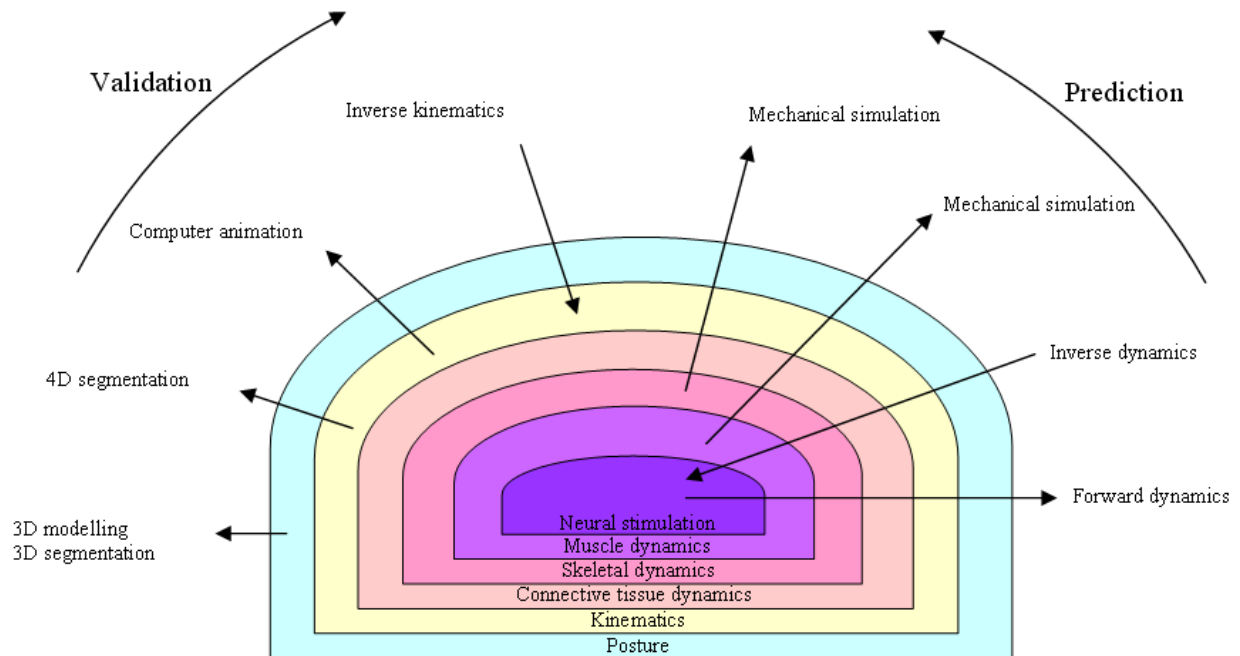


Figure 2.1: Diagram showing the nested nature of musculoskeletal functioning with the different level of modelling/ simulation

Musculoskeletal simulation methods typically deal with the trade-off between computational complexity and accuracy. Indeed, the musculoskeletal system has a complicated geometry, difficult to model realistically (e.g. muscle fibers direction), a complicated mechanical behaviour (viscoelastic, anisotropic, hyperelastic and highly non-linear behaviour), and complicated interactions (e.g. confined cartilages within articulations). Hence, the virtual reproduction of the musculoskeletal system is particularly challenging and requires computational weight to be accurate. For large systems (e.g. lower limbs) or fast simulations, assumptions need to be done through simpler representations. Body simulation is required in a wide range of applications imposing very different constraints, so the choice of the targeted accuracy/ computational complexity/ system size/ model representation is application-dependent. In computer graphics, the key issues are the plausibility and the computational efficiency of body movements. On the contrary, medical applications require an accurate simulation of internal organs based on valid physical parameters. Computer-aided diagnosis, pre-operative planning and image-guided surgery rely on a careful matching between models shape and patient organs. In neuromuscular biomechanics, the key issue is to precisely link muscles activation and joint torque for complex movement patterns (involving large musculoskeletal complexes).

After a short review of anatomical and mechanical concepts, we will broadly present acquisition modalities used in the field, while this work mainly deals with medical images (Chapter 3). Then, we will describe

the different theories and representations associated to musculoskeletal system modelling and simulation; highlighting in parallel, approaches from the different application domains that all assess body complexity but at different scale/accuracy and with their own background. We wish to stress possible future cross-interactions between side-domains such as biomechanics, kinesiology and computer graphics. The goal is also to give a broad and integrated review in musculoskeletal modelling/simulation for pedagogic purpose.

## 2.1 Musculoskeletal anatomy

### 2.1.1 Overview

This section presents the gross musculoskeletal anatomy and the main concepts in biological tissue simulation. The available information on the web such as the Gray's anatomy <sup>1</sup>, as well as artistic anatomy [Tho29] and medical [Sal98] literature have been used to gather this description.

The musculoskeletal system is an organ system that gives animals the ability to move to more favourable conditions by using muscles and skeletal system to engage in locomotion. The human musculoskeletal system consists of the human skeleton, made by bones attached to other bones with joints, and skeletal muscle attached to the skeleton by tendons. Bones are supported and supplemented by a structure of ligaments, tendons, muscles, cartilage and other organs. On top of the musculoskeletal system, the skin offers a protection to the external environment, and sensation/regulation/storage mechanisms. The skin is a three-layers organs mainly made of collagen. Collagen fibers are oriented according the Langer's lines (see Figure 2.2).

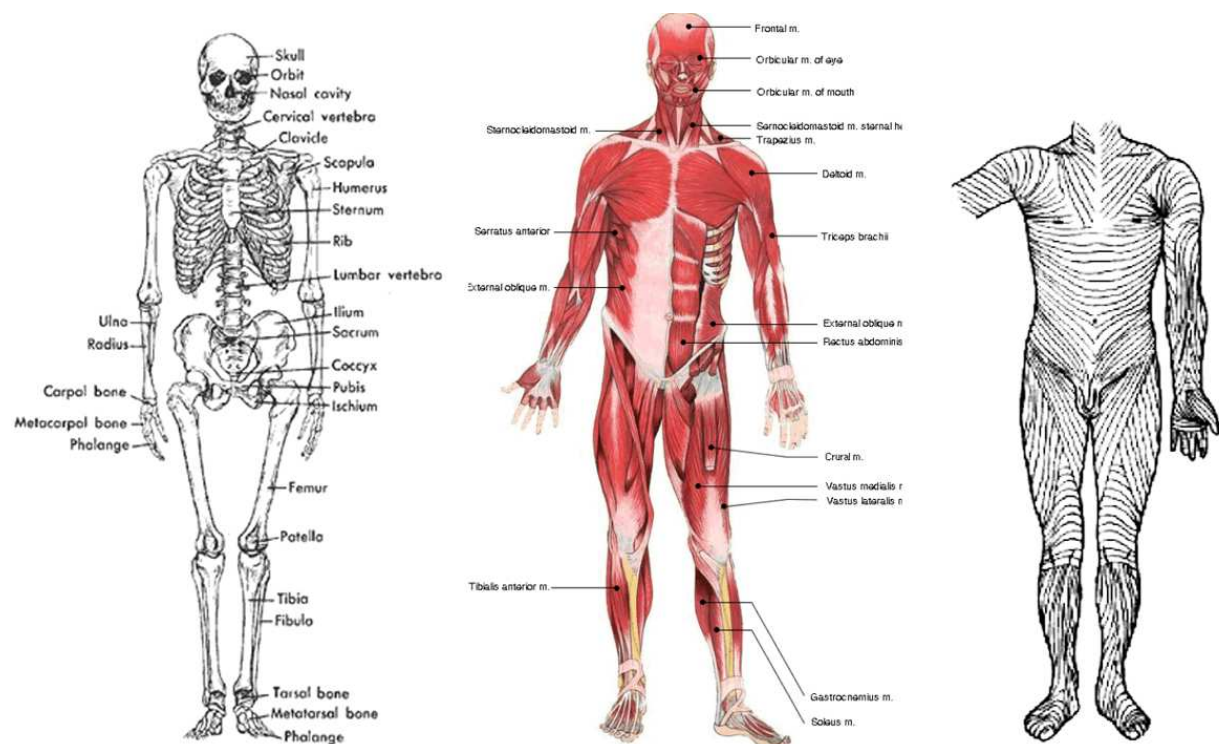


Figure 2.2: Skeletal system (from the Neurological Associates Pain Management Center, Florida), Muscle System and Langer's lines of the skin

<sup>1</sup>[http://en.wikipedia.org/wiki/Gray's\\_Anatomy](http://en.wikipedia.org/wiki/Gray's_Anatomy)



The human body is mainly shaped by bones, offering a rigid support and protection to internal soft organs. Moreover, bones are used to store energy for muscles (fat and minerals) and to produce blood cells. On average, the number of bones in the human skeleton is 206 (see Gray's Anatomy), without accounting for small sesamoid bones. The skeleton changes composition over a lifespan as a number of small bones (e.g. in skull and spine) fuse together during growth. Indeed, babies are born with approximately 270 bones. Skeletal muscles are the active part of the musculoskeletal system, generating forces on bones. There are around 650 skeletal muscles in the human body. Tendons attach skeletal muscles across joints, allowing muscle contraction to push/ pull bones. Muscles generally work in pairs to produce movement: when one muscle flexes (or contracts) the other relaxes, a process known as antagonism. Muscles have both electrical and chemical activity. There is an electrical gradient across the muscle cell membrane: the outside is more positive than the inside. Stimulus causes an instantaneous reversal of this polarity, causing the muscle to contract. The musculoskeletal system is articulated thanks to synovial joints linking bone segments. They contain ligaments that strengthen joints and hold bones in position. Inside the joint, the synovial fluid lubricates the articulation and prevents the two cartilage caps on the bones from rubbing together.

### 2.1.2 Classification of joints

There are three different types of joints found in the human body [TA75]:

**Fibrous joints** - bones are connected by fibrous connective tissue: in fibrous joints bones are joined by tight and inflexible layers of dense connective tissue, consisting mainly of collagen fibers. In adults, these are not designed to allow any movement; however, in children, fibrous joints have not solidified and are movable. Examples of fibrous joints are cranial sutures, joining the bones of the cranium, and gomphoses, the joints between the roots of the teeth and their sockets (or alveoli) in maxilla and mandible.

**Cartilaginous joints** - bones are connected by cartilage: in cartilaginous joints (also known as synchondroses) bones are connected entirely by cartilage. In comparison to synovial joints, cartilaginous joints allow only slight movement. Examples of cartilaginous joints are the pubic symphysis, the joints between the ribs and the sternum, and the cartilage connecting the growth regions of immature long bones. Another example is in the spinal column - the cartilaginous region between adjacent vertebrae.

**Synovial joints** - there is a space (synovial cavity) between the articulating bones: The term "Synovial joint" and "Diarthrosis joint" are often used interchangeably, although the first term refers to the structure and the second one to the function.

From a functional point of view, the different articulations of the body can be categorised such as:

**Synarthrosis joints** - permit no movement: Synarthroses are joints with very little (if any) mobility. They can be categorised by how the two bones are joined together: Syndesmoses are joints where the two bones are joined by one or more ligaments. Synchondroses are joints where the two bones are joined by a piece of cartilage. Synostoses are the fusion of two bones, to the point that they are practically one bone. In humans, the plates of the cranium, initially separate, fuse together as the child approaches adulthood.

**Amphiarthrosis joints** - permit little movement: amphiarthroses are slightly moveable joints where the two bone surfaces at the joint - both covered in hyaline cartilage - are joined by strands of fibrocartilage (e.g. junction between the tibia and fibula). Most amphiarthrosis joints are cartilaginous.

**Diarthrosis joints** - permit a variety of movements (e.g. flexion, adduction, pronation). Only synovial joints are diarthrosis. Diarthroses (sometimes called synovial joints and also diarthroidal joints) are the most common and most moveable type of joint in the body. The whole of a diarthrosis is contained by

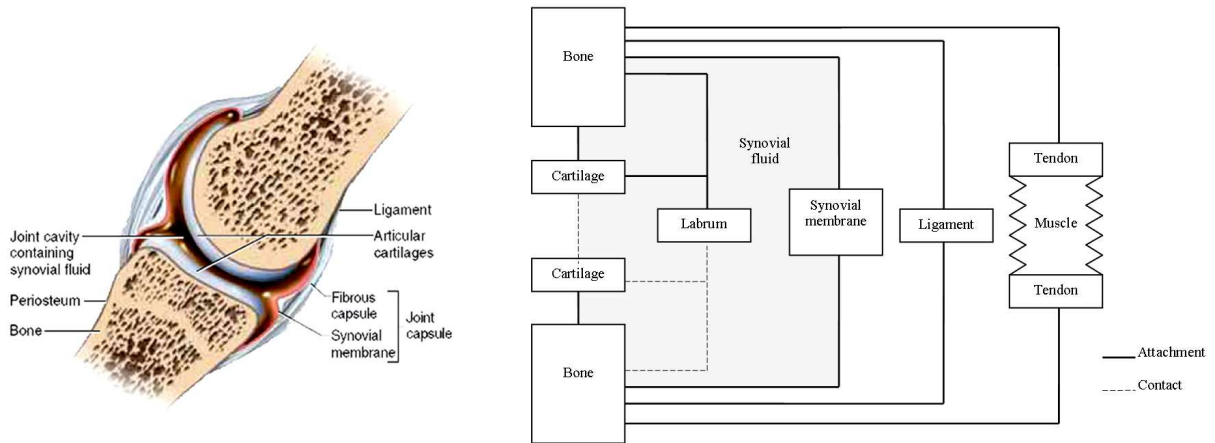








Figure 2.3: Synovial joint (from [Sal98]) and its corresponding multi-layer schematics. For some joints (e.g. shoulder, hip joints), a fibrous cartilage called labrum may protect bone edges

a ligamentous sac called the articular capsule. The surfaces of the two bones at the joint are covered in cartilage. The thickness of the cartilage varies with each joint, and sometimes may be of uneven thickness. In joints where the two surfaces do not fit snugly together, a meniscus or multiple folds of fibro-cartilage (labrum) within the joint correct the fit, ensuring stability and the optimal distribution of load forces. The synovium is a membrane that covers all the non-cartilaginous surfaces within the articular capsule. It secretes synovial fluid into the joint, which nourishes and lubricates the articular cartilage. The synovium is separated from the capsule by a layer of cellular tissue that contains blood vessels and nerves. Synovial joints can be further grouped by their shape, which controls the movement and the degrees of freedom or DOFs (images from [SPMC96]):

- 
  - Gliding joints (6 DOFs), such as in the carpals of the wrist and the tarsal and metatarsal bones (foot), allow a wide variety of movement, but with a little range of motion.
- 
  - Hinge joints (1 DOF), such as the elbow (between the humerus and the ulna), act like a door hinge, allowing flexion and extension in just one plane.
- 
  - Pivot joints (1 DOF), such as in the forearm (between the radius and the ulna), allow rotations around an axis parallel to bones.
- 
  - Ellipsoidal (or condyloid) joints (2 DOFs), such as the knee, are the union of egg-shaped heads and sockets, leading to two rotational degrees of freedom (no axial rotation). Some classifications make a distinction between condyloid and ellipsoidal joints.
- 
  - Saddle joints (2 DOFs), such as at the thumb (between the metacarpal and carpal), allow the same movements as ellipsoidal joints. The difference is in the shape: saddle joints present a double curvature.
- 
  - Ball and socket (or spheroidal) joints (3 DOFs), such as the hip joint, allow large rotations around the three axes. This is the most flexible type of joint, permitting a wide range of movement. However, range of motion (or directional limit) is restricted due the shape of the joint (depth of the socket), which influences the stability of the joint.

Directional limits of ball and socket joints have been widely studied in the past: [MWTT98] and [Aub02] have used spherical polygons for studying the shoulder, [KSMMT03] have analysed the hip joint through standard medical angles.

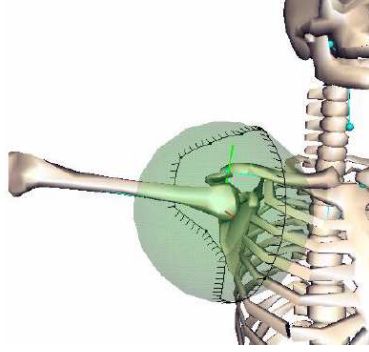


Figure 2.4: Directional limits of the shoulder as a spherical polygon (from [Aub02])

## 2.2 Biological tissue modelling

### 2.2.1 Tissue composition

The human body is mainly composed of water (60%), and to a least extent of starch, sugar, minerals, fat and protein. Human tissues are made of cells, although their structure relies on extra-cellular matrices, consisting of fibers (mainly proteins such as collagen and elastin) and a ground substance (mainly proteoglycans). The collagen is the most important material since it represents 30% of the total proteins and is contained in almost all tissues. Tissues are commonly classified into four groups:

**Connective tissues** - they include cartilage, tendons, ligaments, bone matrices, fat, skin, blood and lymph. Their characteristic is that their main component is the extra-cellular matrix. Fibrous arrangement is more-or-less regular: skin is composed of randomly interlaced proteins, whereas tendon and ligament fibers are dense and regular. There are also thin fibrous connective tissues for binding blood vessels, liver cells and muscle cells (basal membrane).

**Epithelia** - it is made of cells with little extra-cellular matrix, and covers the inside and outside surface of organs (e.g. lungs, stomach, intestines, blood vessels), on top of a supportive basal membrane.

**Muscle tissues** - their main component is the muscle cell. Three types of muscle tissues are distinguished: skeletal (striated or voluntary) muscles which produce articular movements, smooth muscles (e.g. digestive tract, bladder, arteries, and veins), and cardiac muscles (heart).

**Nerves** - nerve cells, or neurons, are designed to transfer signals from one part of body to another. A neuron has a cell body (soma), small branches (dendrites) binding neurons, and a long branch (axon) transmitting signals on long distances. At the end of an axon, there are numerous synapses, which pass the signals to other neurons or muscles.

### 2.2.2 Constitutive models of biological tissue

Similarly to fractals, the structure of tissues is hierarchical (see Figure 2.24), leading to a scale dependency of their mechanical property. Based on continuum mechanics and on the elasticity theory [Fun81], constitutive models of biological tissues aim at abstracting the different tissue layers. The goal is to be able to reproduce tissue behaviour at a macroscopic scale, taking into account the underlying structure. In the framework of this thesis, model deformations are derived from the geometry of the surface rather than continuum mechanics (no constitutive models) mainly because of flexibility, computational load and parameterisation issues (see Section 3.3.4). Indeed, constitutive models are better suited for simulation (predictive approach)

rather than for modelling (descriptive approach). Nevertheless, it is important to review the main concepts of continuum mechanics and constitutive modelling to understand the different approaches proposed in the literature to model and simulate musculoskeletal tissues.

Considering a deformed (Eulerian) state and a reference (Lagrangian) state, the continuum mechanics theory derives basic equations describing material strain in function of displacement, and internal forces in function of the stress. Numerical simulation can be performed through space/time discretisation and through the resolution of the movement differential equation, relating forces and displacement. For simplicity, the quasi-static and the small displacement cases are often assumed. While strain/stress formulations make no hypothesis about the material, tissue mechanical property is described through the relationship between stress and strain, known as the constitutive relationship. A constitutive equation is generally developed with the need to balance the accuracy of model behaviour under the considered range of loading with the simplicity of the formulation allowing numerical solving and experimental measurement (determination of equation constants). The main physical characteristic of a purely elastic material is that it stores energy under load. This energy, a scalar quantity, is often called strain energy:  $U = \sigma_{ij}\varepsilon_{ij}/2 = S_{ij}E_{ij}/2$  where:

- $\mathbf{x}'$ : undeformed position
- $\mathbf{x}$ : deformed position
- $\mathbf{u} = \mathbf{x} - \mathbf{x}'$ : displacement
- $E_{ij} = \frac{1}{2}(\frac{\partial u_i}{\partial x_{j'}} + \frac{\partial u_j}{\partial x_{i'}} + \frac{\partial u_k}{\partial x_{i'}} \frac{u_k}{\partial x_{j'}})$ : Green-Lagrange symmetric strain tensor (large deformations, reference configuration)
- $S_{ij}$ : 2nd Piola-Kirchoff symmetric stress tensor (large deformations, reference configuration)
- $\varepsilon_{ij} = \frac{1}{2}(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i})$ : Cauchy symmetric strain tensor (small deformations, deformed configuration)
- $\sigma_{ij}$ : Cauchy symmetric stress tensor (small deformations, deformed configuration)

Early developed constitutive models were uniaxial relationships between Cauchy stress and strain [KGD64]. Multidimensional models were established using a scalar strain energy function  $W$  (or Green-elastic function) for elastic materials such as:

$$\begin{aligned} S_{ij} &= \partial W(a_i, E_{ij}) / \partial E_{ij} && \text{(large/non-linear deformations)} \\ \sigma_{ij} &= \partial W(a_i, \varepsilon_{ij}) / \partial \varepsilon_{ij} && \text{(small/linear deformations)} \end{aligned} \tag{2.1}$$

Materials exhibit a highly non-linear stress/ strain relationship over the full range of loading (Figure 2.5). However, in a limited range, assumptions to such as the linear elastic material (constant Young's modulus) can be done. The linear strain assumption (not to be confused with linear elastic material), using the Cauchy strain, is valid for small deformations but is not robust for large deformations (ghost forces are produced for global rotation, producing distortions).

The linear, non-linear and viscoelastic models are the main constitutive models for describing the stress-strain relationship. From these general equations, assumptions are generally made about the structure of the material (symmetries) for decreasing the number of unknowns. These assumptions result into simplifications of the stress equation. Then, different materials (strain energy function formulations) have been proposed to reach a little number of parameters that can be determined experimentally.

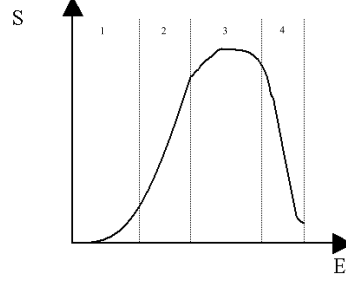


Figure 2.5: The four regions of the strain/ stress curve: 1) non-linear region; 2) quasi-linear region; 3) plasticity region; 4) rupture region

**Linear elasticity** - The linear elastic constitutive equation assumes that there is a linear relationship between stress and strain (Hooke's law), and that the stress depends only on the strain (not the strain rate). In this case, the Cauchy strain  $\varepsilon$  and stress  $\sigma$  are used. For linear elastic materials (e.g. bones), we have:

$$W = S_{ij}E_{ij}/2 = C_{ijkl}E_{ij}E_{kl}/2 \quad S_{ij} = C_{ijkl}E_{kl}$$

The material elastic tensor  $C_{ijkl}$  is obtained by taking the second derivative with respect to strain:

$$C_{ijkl} = \partial^2 W / \partial E_{ij} \partial E_{kl}$$

Due to strain/stress symmetries, the original 81 elastic coefficients of the elastic tensor are equivalent to 21 independent constants. In practice, they are simplified to 2, 5, or 9 constants, depending on material symmetry. If the material is isotropic, 2 coefficients, the Lamé coefficients  $\lambda$  and  $\mu$ , are used:

$$S_{ij} = \lambda \text{Tr}(E_{kl})\delta_{ij} + 2\mu E_{ij}$$

with  $\lambda = \nu E / [(1 + \nu)(1 - 2\nu)]$  and  $\mu = E / [2(1 + \nu)]$  ( $E$ : Young modulus,  $\nu$ : Poisson coefficient)

In the small deformation framework, the linear stress/ strain relation is valid since we consider infinitesimal variables change. For isotropic materials, we can express internal forces as a linear function of the displacement and Lamé coefficients. This is called the linear elasticity operator ( $\Delta$ : Laplacian;  $\nabla$ : gradient):

$$\mathbf{f}_{\text{int}} = \text{div}(\sigma) = \mu \Delta \mathbf{u} + (\lambda + \mu) \nabla(\text{div}(\mathbf{u}))$$

**Hyperelasticity (non-linear elasticity)** - While linear elasticity is a very good model for bone tissue, it does not serve well for soft tissue mechanics because most soft tissues undergo large deformations and because the stress/strain relationship is generally nonlinear (stiffness changes with deformation). The second derivative of the strain energy function with respect to the strain gives the tangential stiffness for the current deformation:

$$C_{ijkl}^t = \partial^2 W / \partial E_{ij} \partial E_{kl}$$

In many cases the strain energy function may be defined in terms of invariants of the right Cauchy deformation tensor  $\mathbf{C} = \mathbf{F}^T \mathbf{F}$  ( $F_{ij} = \partial x_i / \partial x_j'$ : deformation gradient). This is especially true if the material is isotropic. In that case, we need to apply the chain rule to determine the 2nd PK stress as:

$$S_{ij} = \frac{\partial W}{\partial C_{ij}} \frac{\partial C_{ij}}{\partial E_{ij}} = 2 \frac{\partial W}{\partial C_{ij}} \quad \text{with} \quad W = W[I1(C_{ij}), I2(C_{ij}), I3(C_{ij})] \quad (2.2)$$

Where  $I1$ ,  $I2$  and  $I3$  are the three first invariants of the tensor  $\mathbf{C}$ :

- $I1(\mathbf{C}) = Tr(\mathbf{C})$
- $I2(\mathbf{C}) = (Tr(\mathbf{C})^2 - Tr(\mathbf{C}^2))/2$
- $I3(\mathbf{C}) = Det(\mathbf{C})$

**Viscoelasticity** - Viscoelasticity incorporates aspects of both fluid (viscous) and solid behaviour (elastic). Most notably, we know that elastic materials store 100% of the energy due to deformation. However, due to internal friction (fibers and fibers/matrix friction), viscoelastic materials dissipate some of this energy. As a consequence, the loading stress-strain curve and unloading curve are different, the hysteresis area representing energy dissipation. Strain and stress of viscoelastic materials are time dependent: stress relaxation occurs for a fixed strain, whereas creep occurs for a fixed stress. For small strain, classical constitutive models of viscoelastic materials are derived from mechanical analog: the Maxwell model (serial dashpot and spring) and the Kelvin-Voight model (parallel dashpot and spring). In practice, the Kelvin-Voight gives more realistic results. Fung [Fun81] also introduced a viscoelastic model for large deformation in his quasi-linear viscoelasticity (QLV) theory. The stress relaxation function is separated into a time dependent part and an elastic part.

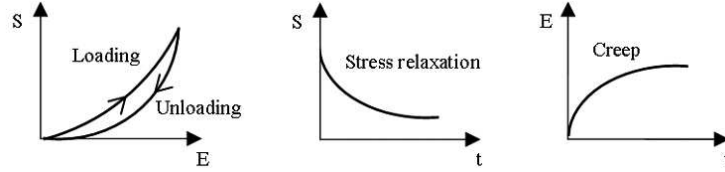


Figure 2.6: Characteristic curves of viscoelastic materials

*The poroelastic and biphasic theories* have been developed for modelling elastic solids with pores that are filled with fluid (e.g. cartilage), leading to a highly viscoelastic behaviour. They provide a quantitative description of the relative contributions of the solid and fluid phases in soft tissue [MKLA80]. In addition to the elastic constants normally measured, the permeability of the porous solid must also be measured.

**Isotropy** - Isotropy brings symmetry in the stress formulation, leading to simplification of Equation 2.2. It can be shown that, for isotropic materials, we obtain:

$$S_{ij} = 2 \frac{\partial W}{\partial C_{ij}} = 2 \frac{\partial W}{\partial I1} \delta_{ij} + 2 \frac{\partial W}{\partial I2} (I1 \cdot \delta_{ij} - C_{ij}) + 2 \frac{\partial W}{\partial I3} I3 \cdot C_{ij}^{-1} \quad (2.3)$$

*Polynomial isotropic material:* From Equation 2.3 and assuming incompressibility ( $I3 = 1$ ), it is common to approximate the material strain energy function by a general polynomial function of the invariants, where constants  $\mu_i$  are subsequently defined by fitting the model to experimental data, by:

$$W = \sum_i^N [\mu_1 (I1 - 3)^i + \mu_2 (I2 - 3)^i] + c$$

$$\text{Or in a reduced form: } W = \sum_i^N [\mu_i (I1 - 3)^i] + c$$

*Mooney-Rivlin/ Neo-Hookean isotropic material:* The Mooney-Rivlin model [Moo40] and the Neo-Hookean model are widely used for modelling incompressible material such as rubber, but are also employed in the framework of biological soft tissues [TBHF03] [SNF05]. They are 1st order Polynomial isotropic materials. The Mooney-Rivlin strain energy function is defined as:

$$W = \mu_1(I1 - 3)/2 - \mu_2(I2 - 3)/2$$

By simplification, we obtain the Neo-Hookean strain energy function:  $W = \mu_1(I1 - 3)/2$

*Veronda isotropic, non-linear material:* Mooney-Rivlin and neo-Hookean functions show a more-or-less linear force-elongation behaviour in uniaxial tension. In practice, the Veronda [VW70] material is preferred for soft-tissues because his exponential formulation is able to capture the different degrees of material non-linearity:

$$W = \mu_1(e^{\gamma(I1-1)} - 1)/\gamma + \mu_2(I2 - 3)/2$$

The reduced Veronda formulation is:  $W = \mu_1(e^{\gamma(I1-1)} - 1)/\gamma$

In practice,  $\gamma$  controls the material non-linearity, whereas  $\mu_1$  and  $\mu_2$  influence material stiffness.

**Transverse isotropy** - The isotropy assumption is often too coarse for soft-biological tissues as they generally exhibit a fibrous structure. Equation 2.2 is in practice simplified by assuming a directional or transverse isotropy. Given the fiber direction  $\mathbf{a}^0$  in the reference configuration, the 2nd PK stress is expressed by:

$$S_{ij} = 2 \frac{\partial W}{\partial C_{ij}} = 2 \frac{\partial W}{\partial I1} \delta_{ij} + 2 \frac{\partial W}{\partial I2} (I1 \cdot \delta_{ij} - C_{ij}) + 2 \frac{\partial W}{\partial I3} I3 \cdot C_{ij}^{-1} + 2 \frac{\partial W}{\partial I4} a_i^0 a_j^0 + 2 \frac{\partial W}{\partial I5} (a_i^0 C_{jm} a_m^0 + a_j^0 C_{im} a_m^0)$$

Where  $I4$  and  $I5$  are the 4th and 5th invariants:

- $I4(C_{ij}) = a_i^0 C_{ij} a_j^0 = \lambda^2$  (stretch of the reference fiber direction)
- $I5(C_{ij}) = a_i^0 C_{im} C_{mj} a_j^0$

*Fiber-reinforcement:* The presented isotropic material models model the ground substance contained in biological tissues. However, extra terms can be incorporated to simulate the fibrous nature of the tissues (fiber-reinforcement) [KL99]. As shown from the stress formulation of transversely isotropic materials, the 4th and 5th invariants are used to model anisotropy. Based on the Veronda non-linear material, the fiber-reinforcement term can be formulated as:

$$W = \mu_1(e^{\gamma(I4-1)} - 1)/\gamma + \mu_2(I4 - 3)/2$$

or:  $W = \mu_1(e^{\gamma(I4-1)} - 1)/\gamma$

**Incompressibility** - Most biological tissues are nearly incompressible [AD85] as they are mainly composed of water. In this case,  $W$  does not theoretically depends on  $I3$  since  $I3 = J^2 = 1$ . However, in the ideal case of total incompressibility, the stress-strain relationship is not an isomorphism. For instance, there is no deformation in the case of a constant (hydrostatic) force applied to the surface. This results in inaccuracy when numerically simulating the material (locking effect). In practice, pressure and displacement are treated separately: a term, function of  $I3$ , is added to the elastic strain energy function for ensuring quasi-incompressibility (pressure term), whereas elastic terms are altered through a reduced right Cauchy deformation tensor (thus reduced invariants). The reduction  $C^* = I3^{-1/3} C = J^{-2/3} C$  is widely used [SB87] [WVG96] [TBHF03], and leads to reduced invariants  $Ii^*$  to be incorporated in elastic strain energy function formulation. [SB87] defines the pressure term such as  $Wp = K(I3^* - 1)^2/2$ . In [WVG96] [TBHF03], the term  $Wp = K \ln(J)^2$  (dilatational response) is preferred.  $K$  is the bulk modulus of the material, allowing the modification of the degree of incompressibility.

### 2.2.3 Mechanical simulation

The numerical simulation of object mechanical behaviour has received a large attention in the computer graphics and computational mechanics communities, leading to a wide range of methods. The choice of a technique is largely driven by the application: the type of object and studied continuum guide the choice of the model representation; the range of dynamic variables to be explored allows assumptions in the constitutive model, accuracy/ computational time constraints define the numerical resolution method. Among the presented methods, we can distinguish the ones that consider objects as a continuum, applying discretisation techniques on continuum mechanics law (e.g. finite element); and the ones that consider objects as a set of (interconnected) mass points. The first is considered as more accurate (physically-based) and less sensitive to mesh orientation in the material, while the later is faster (numerically and geometrically-based). Some recent methods tries to unify these two families, through the expression of continuum mechanics law in the particle system framework (mass lumping process). Here, we briefly review the main existing approaches, focusing of solid object expressed from the Lagrangian point of view<sup>1</sup>. For more details, please refer to the recent exhaustive review on physically based deformable models by Nealen et al. [NMK<sup>+</sup>05]. In this thesis, we will focus on discrete surfacic models simulated as a particle system to optimise computational speed instead of accuracy. This is valid since models are driven by images rather than intrinsic properties.

**Continuum-based methods:** The finite-element method (FEM), the finite-difference method (FDM) and the boundary-element method (BEM) are well-established mechanical engineering techniques that have been widely used since the 70's for simulating tissues deformation [BPS72] [TPBF87]. Their main idea is to discretise the continuous equation describing energy equilibrium (virtual work principle) inside an object [Zie77] [Bat96]. The spatial domain is subdivided into elements such as triangles [MT95], bricks [CZ92] [LEHW01] or more frequently tetrahedrons [CDA99b] [OH99] [HFS<sup>+</sup>01] [TSB<sup>+</sup>05] [RGTC98], and boundary conditions are defined (Dirichlet or Neumann boundary conditions). While the change of variables across a regular spatial grid is straightforwardly evaluated by discrete derivatives with FDM ( $\partial f / \partial x \simeq (f(x + \Delta x) - f(x - \Delta x)) / 2$ ) [TPBF87] [TW88], FEM makes use of an interpolation function (e.g. B-splines, piecewise polynomial, linear basis, Bernstein-Bézier etc.), which is integrated in the energy equation. Therefore, FEM can exploit higher order derivatives than FDM and requires fewer nodes, not necessarily regularly sampled (better approximation of organ boundaries). BEM [JP99] [ALR<sup>+</sup>99] transform the differential operator defined in the domain to integral operators defined on the boundary (divergence theorem) by assuming that the material is linear and homogeneous. Using this idea for individual tetrahedrons, FVM [TBHF03] [TSB<sup>+</sup>05] makes use of the divergence theorem to simplify force and strain computation by handling tetrahedron triangles instead of the boundary of the Voronoi region associated to each node. Assumption in FVM is the constant strain tetrahedra and linear variables change (linear basis). Finally, all formulations lead to a partial differential equation (PDE) relating node acceleration  $\ddot{\mathbf{p}}$ , velocity  $\dot{\mathbf{p}}$ , position  $\mathbf{p}$  and external forces  $\mathbf{F}_{\text{ext}}$ , through mass  $\mathbf{M}$ , damping  $\mathbf{D}$  and stiffness matrices  $\mathbf{K}(\mathbf{p})$ :  $\mathbf{M}\ddot{\mathbf{p}} + \mathbf{D}\dot{\mathbf{p}} + \mathbf{K}(\mathbf{p})\mathbf{p} = \mathbf{F}_{\text{ext}}$ . The stiffness matrix expresses the strain-stress relationship and is consequently based on the underlying constitutive mechanical model.

Methods	Complexity	Assumptions
FEM	++	-- the basis function fits to variables change
FVM	+	- variables change linearly
FDM	-	+ variables change linearly (regular grid)
BEM	--	++ homogeneous linear material

Table 2.1: Continuum mechanics-based methods

<sup>1</sup>In the Lagrangian framework, changes are defined in terms of object deformation. As opposite, Eulerian methods are interested in space changes (object boundaries are implicit). This is useful for fluids/ gases/ implicit models simulation.



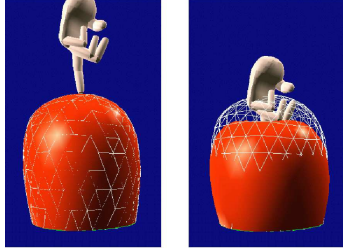


Figure 2.7: Boundary Element Method (BEM), from [JP99]

Authors have investigated different formulations of constitutive models for musculoskeletal tissues and their expression in the FEM/ FDM/ FVM/ BEM framework. As already said, biological tissues are, in general, non-linear, anisotropic, hyperviscoelastic and quasi-incompressible materials. For simplicity and because of time-step restriction, quasi-static situation is often assumed, so that the PDE becomes  $\mathbf{K}(\mathbf{p})\mathbf{p} = \mathbf{F}_{\text{ext}}$ , where  $\mathbf{K}(\mathbf{p})$  need to be inverted. Moreover, small strain is another widely made assumption [GMTT89]. Small strain leads to the use of Cauchy stress and strain and linear constitutive laws, thus decreasing computational cost ( $\mathbf{K}\mathbf{p} = \mathbf{F}_{\text{ext}}$ ). However, for large deformation (Green strain), strain non-linearities are paid by high computational cost (the stiffness matrix depends on the displacement). In their pioneer work, Terzopoulos and Witkin [TW88] combine rigid and non-rigid motion to reduce system stiffness. Recently, linearized version of this strain [DDCB01] and non-rotated frames [MG04] have shown

promising results. When interactions with models are surfacic only, which is the case when assuming no topological modifications such as cutting, internal nodes can be removed from computation, through a BEM formulation [JP99] or stiffness matrix condensation [Kar87]. Improvements are still possible through matrix pre-computation and efficient update methods in the case of few boundary conditions change. These techniques allow interactivity, with simple standalone models (one model, <10 000 nodes) [JP99] [BNC96] [CDA99b].

**Discrete methods:** The discretisation of the spatial continuous domain leads to a set of nodes that can be considered as particles with a certain mass, evolving in space under forces. As opposed to continuum-based methods, particle systems solve motion equation independently for each particle. This is a critical difference, allowing particle systems to be much more efficient in terms of computational speed. Indeed, particle systems, as they rely on a local description, lead to a much more sparse system than FEM, thus to a simpler resolution. In addition, topology change (e.g. cuts) can be more easily implemented. Regarding dynamics, particles are subject to the Newton law of motion (see Section 3.3.5). Forces are composed of internal forces (related to a local description of the material) and external forces (interaction forces with other objects). After restating the Newton equation into a 1st order differential system, time discretisation is performed. A finite difference formulation allows an explicit resolution of the system (explicit Euler, Runge-Kutta methods), but leads to tight time-step restrictions due to non-linearities as shown in [EEH00] [Hau04]. A Taylor expansion of the force is more suitable but requires the evaluation of force derivatives with regards to particles position and velocity and the inversion of a sparse matrix (see Section 3.3.5). This is known as the implicit method. It has shown good performances for stable real-time applications [BW98] [VMT00b], despite its possible inaccuracy (ghost numerical damping due to non-linear non-stiff behaviour [EEH00]). A particular implementation of particle systems is the well known mass-spring model, in which internal forces are straightforwardly computed from independent springs between particles. Mass-spring simulation is fast and can be easily parallelised and GPU accelerated [Mr05] (e.g. 100 000 particles at 100 frames/s). However material behaviour such as elongation and shearing cannot be modelled accurately with independent springs. Hence, mass-spring networks are particularly inaccurate, especially in the context of large deformations. Moreover spring parameterisation is difficult due to this non-correlation with material strain, even if it can be performed with reference to FEM using some optimisation methods (e.g. genetic algorithm) [BSSH04]. Mass-spring networks are in fact more suitable for enforcing geometric constraints in a mechanical system, such as smoothing or non-penetration effects. Other methods consider forces acting on elements such as tetrahedrons [CDA99a], or triangles [BW98] [VMT00b], taking into account more than particle pair interactions, in the evaluation of forces and force derivatives. They have shown promising results, in reproducing the various deformation modes of real materials, thanks to the formulation of the strain

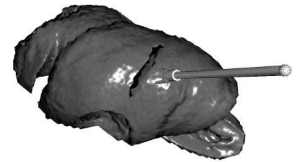


Figure 2.8: GPU-based mass-spring simulation, from [Mr05]

(small deformation strain or linearized version of the Green-Lagrange strain) at the element level. To reduce distortions due to linear strain formulation, Müller et al. [MG04] have used non-rotated configuration to allow large deformations with simple computation. They also present a meshless implementation [MHTG05] that makes use of point explicit registration (Section 3.3.5) to derive geometric internal forces and a stable integration scheme.

**Hybrid methods:** Bro-Nielsen and Cotin [BNC96] [CDA99b] achieved very fast simulation (with haptic feedback) thanks to the condensation technique and to the pre-computation of the effect of surface displacement (superposition technique), with application to hepatic virtual surgery and facial reconstruction (about 8000 elements at 15 frames/s). To handle cuts and mesh topology changes, they combine their method with a particle system ("mass-tensor" network), where internal forces are formulated with respect to the linear elasticity framework (as opposed to mass-spring networks) [CDA99b]. This work has been extended by Pincibono et al. [PDA03] for handling anisotropy and non-linear elasticity. Another approach by Debunne et al. [DDBC99] [DDCB01] is to use multi-resolution (space/time adaptive sampling) with a weighted finite difference integration technique to approximate the Laplacian and the gradient of the divergence operators of the linear elasticity operator. They use a version of the Green-Lagrange strain (invariant to rigid motion) allowing large deformations. Similarly, a local approximation of the resolution is performed in [OH99], through mass lumping, to reduce computational charge. This is known as *explicit FEM*. All these techniques are encouraging since they try to merge the simplicity of particle systems and the accuracy of finite element methods. However, no method currently allows realistic simulation (anisotropic non-linear viscoelastic materials under large deformations) of large multi-organs complexes (neighbourhood constraints). This is enhanced by the difficulty of validating and comparing such methods. A purpose of our work is to make available data for the validation of biomechanical models.



Figure 2.9: Mass-tensor network, from [CDA99b]

## 2.3 Acquisition modalities

When modelling and simulating real objects, data acquisition is essential. It aims at extracting model parameters and validating simulation output. Data can be divided into geometric data (e.g. medical images), kinematical data (e.g. optical motion capture data), mechanical parameters (e.g. strain-stress curves) and physiological data (e.g. muscle activation). Due to measurement difficulties (for instance, the access to bone kinematics, internal muscle activation patterns) and musculoskeletal system complexity, it is common to make assumptions about this data and use prior knowledge. Relying on these, computer models need to be validated correctly. Obviously, geometric models (shape) deal with few assumptions, whereas functional models require a lot of prior information for their simulation (Figure 2.1).

### 2.3.1 Static acquisition

**Cadaveric studies** - Until the  $XX^{th}$  century, cadaver analysis has been the only one way for assessing organ shapes and interrelationships (e.g. Gray's anatomy [Gra00]). The visible human [NLM99], a mixture of medical images and histological cuts of the same individual, is up to now the most complete dataset for anatomical description. It has supported many projects in the fields of medical image processing and

biomechanics. The measurement of submillimeter anatomical features such as in synovial joints still requires specimens [Fer00]. In [NTH00], Ng-Thow-Hing et al. use cadavers for measuring muscle fiber direction. However, medical imaging modalities, more and more accurate and standardised, tend to supplant these kind of studies.

**Radiography and Computed Tomography (CT)** - Historically, the radiography has been the first modality for in-vivo anatomical imaging of internal structure (Roentgen, end of *XIX<sup>th</sup>* century). It is based on the measurement of the intensity of X-rays traversing the body (not absorbed). Sensors are generally phosphor plates, that can be subsequently digitalised (Computed Radiography - CR), but new digital sensors are emerging (Digital Radiography - DR), though they are still expensive. The resulting 2D projections highlight anatomical structures, absorbing X-rays differently. Combining several projections (Computed Tomography - CT), volumes (typically 512x512x128 voxels) can be reconstructed with a resolution of about 0.5x0.5x1 mm in 30s. CT is a tomographic modality, meaning that output images, reconstructed from several projections, are cross-sections. The intensity in images is measured towards the Hounsfield scale, leading to a simple segmentation of tissues. However, different tissues can share the same value, and artefacts disturb this identification. It is known that CT is more dedicated to chest and bone investigations (about 70% of all imaging examinations), as bone-to-soft tissue contrast is high. Thresholding, few regional constraints and simple post-processing techniques (e.g. morphological operations) are generally sufficient to segment bones (see Figure 3.5), or structures highlighted with suited contrast agents (e.g. angiography). Direct volume rendering is also popular in the clinical environment [HPP<sup>+</sup>03]. In orthopaedics, where most of pathologies rely on bones (e.g. fracture, cartilage ossification), CT is the privileged modality, although it is invasive (high dose X-ray absorption can damage cells and cause cancer).

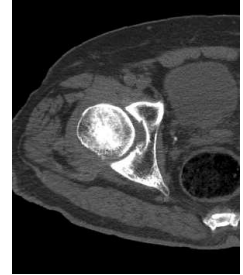


Figure 2.10: CT imaging of the hip

**Ultrasound (US)** - Ultrasounds propagate in homogeneous tissues but are partly reflected at tissue interfaces. The principle of US is to measure this reflection in time, the emission/reception time interval being proportional to the depth of the interface (assuming few wave velocity variations). US scanners generally operate in brightness mode (B-mode), that is grey-scale in images represent an amplitude map of the reflected sound. 2D cross-sections and 3D volumes can be interpolated by casting rays in different directions (ultrasound is a tomographic modality). Piezoelectric resonators are used as ultrasound emitters with frequencies commonly ranging from 3 to 30MHz. Either mechanical and free-hand probes are available [FDC01], allowing for some of them position tracking. General interpolation techniques may be used to resample unparallel slices such as in [TPGB99]. US is low-cost, fast and non-invasive. Unfortunately, image quality is low: speckle is present due to wave interference, geometric distortions occur due to wave velocity changes and ultrasound can be totally absorbed by some tissues (e.g. air, bones) and at large depth, leading to occlusions. Methods have been developed to correct geometric distortions [OR03]. US is typically used for heart and foetus dynamic imaging, but is also adequate for superficial musculoskeletal imaging [Adl99] [RJ92] [LAM99]. Particularly, muscle fibers direction can be visualised in-vivo (Figure 2.11).

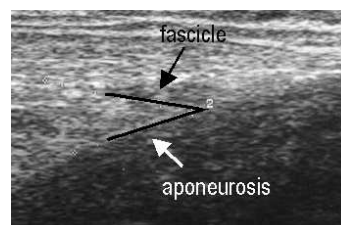


Figure 2.11: Ultrasound assessment of the pennation angle of the gastrocnemius muscle in a relaxed state, from University of California, Davis

**Magnetic Resonance Imaging (MRI)** - Atomic nuclei have a magnetic moment that can be oriented by an external magnetic field. This moment rotates about the axis of the field, with a frequency depending on the type of nucleus and on the strength of the magnetic field. In this condition, atoms can briefly jump from one equilibrium state (spin  $+1/2$ ) to another one (spin  $-1/2$ ), under radio impulses emitted at this frequency (resonance). When returning to the initial state, atoms generate a damped radio frequency signal (relaxation). MRI aims at measuring this signal from the resonance of hydrogen atoms, contained in all tissues (water). Indeed, tissues react differently, depending on the concentration of water (proton density), but also on the way that protons revert back to their resting states after the initial RF pulse (relaxation time). Relaxation time is measured in two directions: T1 (longitudinal) and T2 (transversal). In addition, fluid flow may decrease the signal. These four tissue parameters: proton density, T1 relaxation time, T2 relaxation time and fluid flow; determine the signal intensity on the MR images. By weighting these parameters, MRI is capable of producing various types of contrast, conferring to this modality a great versatility. T2\*-weighted imaging is an extension of T2-weighted imaging, where local magnetic field inhomogeneities is not compensated (refocussing). As a general rule, we have:  $T1 > T2 > T2^*$ . The two most important parameters for configuring a MR sequence are the repetition time (TR) and the echo time (TE). The TR is the time between consecutive RF pulses. The TE is the time between the initial RF pulse and the echo. Spatial acquisition is performed by locally modifying the magnetic field (magnetic gradient), thus the relaxation frequency. The desired relaxation frequency is subsequently selected. Image Encoding is performed in the Fourier domain (k-space). MRI provides images showing good soft tissue contrast and multi-planar imaging capabilities. Moreover, it is admitted that MRI does not produce any harmful ionizing radiation [SC04], which is important for clinical research commonly involving volunteers. Contrary to CT that is dedicated to bones, MRI is more flexible as it is possible to highlight desired organs, using tissue-specific sequences. Typical medical resolution is about  $1mm^3$ . Strong magnets ( $>1$  Tesla), which are horizontal tubes of about 60cm diameter, lead to a higher signal-to-noise ratio (SNR), and consequently to a higher resolution and shorter scan times than feeble magnets. However, open magnets ( $<1$  Tesla) are promising for musculoskeletal imaging, as they allow in-situ imaging (standing position, large range of motion)[BHB<sup>+</sup>99]. Moreover, specific surface coils can be used to improve acquisition time and SNR. In research, the spatial resolution can attain  $1\mu m^3$ , using powerful magnets (e.g. 4 Tesla, 7 Tesla).

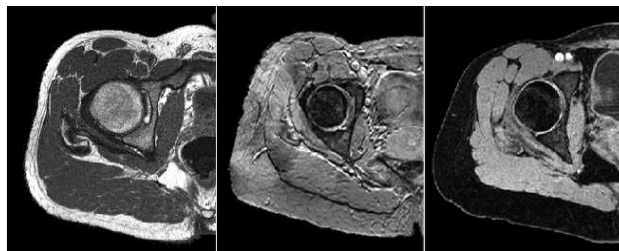


Figure 2.12: Different MRI sequences for imaging the hip: a) Spin-Echo T1, TR=578ms, TE=18ms; b) Gradient-Echo T2\*, TR=30ms, TE=14ms; c) Gradient-Echo T1, TR=20ms, TE=7ms

In biological tissues, the self-diffusion of water is restricted by membranes, and consequently relies on tissue geometry (e.g. fiber orientation). Fluid flow attenuates signal intensity in the direction of the magnetic gradient. Hence, by applying different gradient directions, fluid flow/ fiber direction can be measured with MRI. This is the principle of *diffusion MRI*. In Diffusion Tensor MRI (DtMRI), six gradient directions are applied, leading to a full determination of the diffusion tensor (6 unknowns). In Diffusion Weighted MRI (DWMRI), three gradient directions are applied, providing the average diffusivity of the tissue (trace of the diffusion tensor). In musculoskeletal imaging, it has been shown that the eigenvector with the largest eigenvalue (the principal eigenvector) of the diffusion tensor corresponds to the long axis of muscle fibers [DBN<sup>+</sup>96]. These local, voxel-based directions can be combined with a fiber-tracking algorithm (tractography) to reconstruct the paths of muscle fibers in the tissue [MZ02]. Skeletal muscles are increasingly analysed using DtMRI,

whereas brain remains the most studied structure. In [HSV<sup>+</sup>05], Heemskerk et al. measure essential biomechanical parameters of skeletal muscles (Section 2.5.2) such as the physiological cross-sectional area (PCSA), fiber length and pennation angle. They extract fiber direction in manually outlined regions-of-interest (ROI) corresponding to individual muscles.

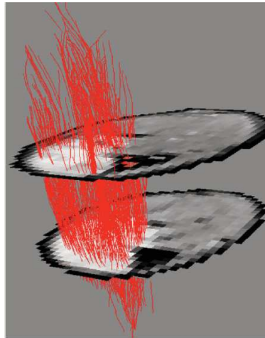


Figure 2.13: Skeletal muscle fibers tracked in a DtMRI volume, from [HSV<sup>+</sup>05]

**Body scanning** - For computer graphics and animation purposes, accurate skin models of the complete body are generally digitalised using commercial optical scanners (e.g. from TC2, CyberWare, Human Solutions). The reflection of white light or eye-safe laser is measured by mobile cameras, providing a point cloud of the scene. Recent automated scanners have a typical millimeter-based resolution, and take less than 30s for the full scanning. It is important to have a reduced scanning time in order to minimise motion artifacts. Subsequently to scanning, the point cloud need to be triangulated, which generally requires limb segmentation. Another approach uses stereo photogrammetry (e.g. 3dMd scanner) from at least two camera views. This technique is faster ( $\approx 1.5ms$ ) and more accurate ( $0.5mm$ ) than laser scanning when using efficient mesh reconstruction techniques, however this has not been applied to large scale (e.g. full body) yet.

**Bioelectrical impedance analysis (BIA)** - Assuming specific resistivities of limb constituents, a simple approach for muscle mass, volume and cross-sectional area assessment is the BIA method [BKNC88]. A small alternative current (e.g.  $800A$ ,  $50kHz$ ), traversing the body, is measured by voltage electrodes located along limbs, and the electrical resistance is calculated. BIA provides an overestimation of muscle volume, and cross-sectional muscle areas (about, respectively, 8% and 15% for the lower limb [FHG<sup>+</sup>99]), with a great inter-subject variability, but is suited for rough clinical evaluations.

### 2.3.2 Kinematics and dynamics acquisition

**Ultrasound (US)** - US is a fast imaging modality that can provide useful kinematic information, with a frequency up to 100 frames/s (e.g. foetal echography). 1D temporal analysis ( $x$  axis: time;  $y$  axis: depth), called M-mode, is useful for visualising highly sampled (about 1000 pulses/sec) superficial muscle contraction patterns or heart valves movements. Studying, strain rate variation in skeletal muscle, physical fatigue can be assessed such as in [WDO<sup>+</sup>04]. Direct tissue velocity visualisation is also possible with US, exploiting the Doppler effect (wave frequency changes when reflected, according to tissue velocity). This is particularly effective for measuring blood flow, muscle contraction velocity [GFSF95] and joint analysis [CSM<sup>+</sup>02]. Colour Doppler aims at visualising flow direction and velocity, whereas color power Doppler (CPD) only deals with velocity amplitude.

**Fluoroscopy and Computed Tomography (CT)** - Fluoroscopy produces real-time images of internal structures in a similar fashion to radiography. X-ray fluoroscopy is faster than CT, as it performs only one projection from a continuous input of X-rays, while CT requires multiple projections to reconstruct

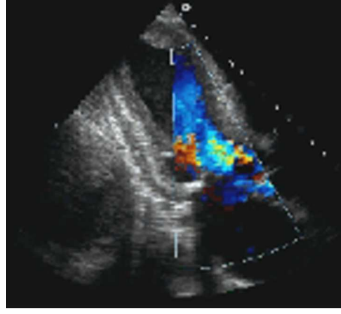


Figure 2.14: B-mode ultrasound slice with superimposed colour Doppler showing blood flow inside the heart (from AnaesthesiaUK)

cross-sections. Projections are suitable for motion tracking because projections contain 3D information, contrary to cross-sections. Consequently, fluoroscopy has been widely used for intra-operative bone tracking [ZGNW01] and biomechanical studies [YSAT01]. Accuracy can be improved using high-speed multiple plane systems such as in [TDG<sup>+</sup>95] [TA03] (250 frames/s). Implanted pins are used for accurate bone localisation [WZG<sup>+</sup>04], but non-invasive bone motion tracking can be obtained through 2D/3D registration between fluoroscopic images and CT data [ZGNW01] [YSAT01]. In [TA03], high precision knee bone tracking is achieved from CT data (<2 deg). CT volumes are converted into 2D projections using digitally radiographs reconstruction (DRR) techniques. New technical advances in CT scanner including large 2D detectors, fast rotating devices and fast systems for transmission and reconstruction, have recently permitted 4D CT acquisition. In [MET<sup>+</sup>04] [ETK<sup>+</sup>03], Mori et al. present a 256-slice scanner with a resolution of about  $1mm \times 1mm \times 0.5mm \times 1sec$  (field of view along  $z = 12cm$ ).

**Magnetic Resonance Imaging (MRI)** - Several MRI techniques has been made available for dynamic tissue analysis, through the decrease of MRI acquisition time. Particularly, joint kinematic studies with MRI have been reported. The passive sequential (or stepped) acquisition technique (i.e. loaded stationary joint during acquisition) is well-established [BHB<sup>+</sup>99] [RKW<sup>+</sup>98]. However, it has a limited utility in application to biomechanics, as demonstrated by Brossmann et al. [BMS<sup>+</sup>93] who reported the importance of acquiring joint motion actively, due to the existence of statistically significant variations between acquiring actively or passively. However, the problem of acquiring volumetric image data in real-time with MRI during active motion remains to be solved due to inherent trade-off in the MR imaging technique between Signal-to-Noise Ratio (SNR), spatial resolution and temporal resolution. To increase acquisition time, it is possible to reduce the repetition time (TR), the number of lines acquired in the k-space (reduced Fourier technique) and specific sequences such as trueFISP (or b-FFE, FIESTA). Quick et al [QLH<sup>+</sup>02] published results on the use of the trueFISP imaging sequence with for real-time imaging of active motion of the hand, ankle, knee and elbow on a single slice (6 frames/s). Parallel imaging techniques (e.g. SENSE, Philips Medical Systems), exploiting spatiotemporal correlations with reference images, can greatly speed up the acquisition as shown in [TBP03] (38 frames/s).

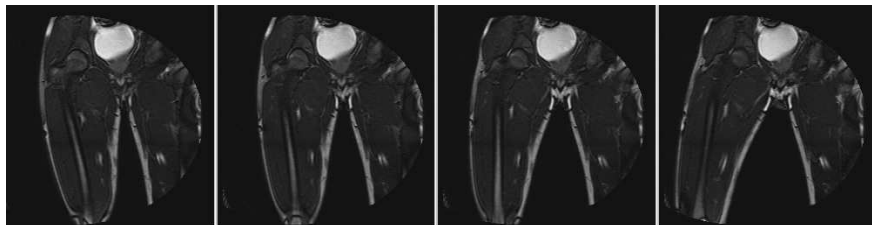


Figure 2.15: Real-time dynamic MRI of the thigh [GPMTV04]

For volumetric acquisitions, motion periodicity can be exploited by acquiring k-lines over several cycles (*cine-MRI*). This is particularly suited for the heart. However, cine-MRI is more complicated for joints as it requires an MRI-compatible positioning device with a method of producing repeatable motion [RS03] [SZD97] [BMS<sup>+</sup>93]. This technique uses a recorded metronome beat to synchronise the volunteer’s movements and an optical trigger on the foot to signal to the scanner the start of each cycle.

Because tissues keep their magnetic orientation for a time ( $\sim T_1$  time), it is possible to create temporary magnetic fiducial markers, or tags, within the tissues. The principle of *tagged MRI* is to use these tags to trace tissue deformations. Tagged MRI has been essentially used for cardiac motion assessment [MA92] [ZPR<sup>+</sup>88] (Figure 2.16), with a grid tag patterns.

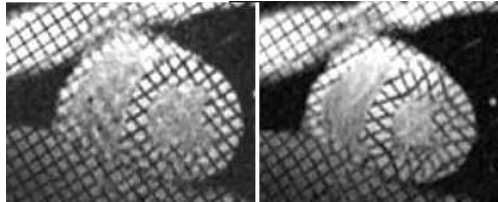


Figure 2.16: Cine tagged-MRI of the heart

Velocity imaging is possible using phase contrast sequences (*pcMRI*) [PHSE91] [DP94]. In pcMRI, two datasets are acquired: one sequence is flow sensitive, while the other one is flow compensated. Flow sensitiveness is controlled by the strength of the bipolar gradient pulse pair which is incorporated into the sequence. The raw data from the two data sets are subtracted. By comparing the phase of signals from each location in the two sequences the exact amount of motion induced phase change can be determined to have a map where pixel brightness is proportional to spatial velocity. pcMRI has been used for dynamic musculoskeletal imaging for assessing actuated parts of muscles through time (Figure 2.17). pcMRI can be coupled with cine-MRI as in [ABGD02] [DP94] for improving spatiotemporal resolution, or performed in real-time [ANB<sup>+</sup>03].

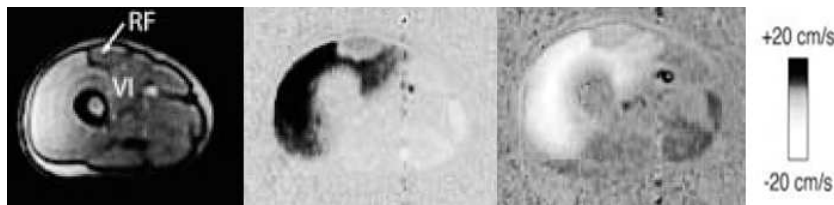


Figure 2.17: Cine phase-Contrast MRI of the thigh during knee flexion/extension [ABGD02]

**Motion Capture (MoCap)** - While medical imaging provides a precise but local measurement of musculoskeletal kinematics, motion capture (MoCap) techniques aim at acquiring more global data in terms of number of joints and ranges of motion. Avoiding the burden inherent to medical scanners, MoCap provides in-situ kinematical data (e.g. gait), and is compatible to other modalities for force (e.g. force plates) and physiological (e.g. EMG) measurement. In kinesiology, it is common to combine motion capture, force plates, EMG and a simplified musculoskeletal model (scaled bone surfaces and muscle action lines) for assessing musculoskeletal dynamics (see Section 2.10). Motion capture of the complete body is achieved through the recording of landmarks attached to the skin. In a post-processing phase, landmark positions are reconstructed in 3D, and rigid motions of the different body segments can be established, approximation skeletal movements. Three main motion capture systems can be distinguished [Men00]:

- The first (flock-of-bird) uses electromagnetic trackers. Low frequency electromagnetic signals are transmitted from sensors to receivers, both placed onto the body. These systems such as the MotionStar (Ascension Technology Corporation) can have 90 sensors and can record at  $140Hz$ . Motion reconstruction is very fast, and even real-time. However, the precision is low (typically  $0.5deg/2mm$ ), and the vicinity of surrounding ferromagnetic materials (metals) disturbs magnetic sensor measurements. This kind of system is not suitable for mobile motion analysis.
- The second system is based on electromechanical devices such as inclinometers, goniometers, accelerometers and gyroscopes. These devices do not require any external units for recording data and are therefore very promising since they can be used in a free environment (ambulatory motion analysis), the earth gravity and magnetic field being the reference. Moreover, thanks to electronics advancements, they are becoming smaller, cheaper, more stable and more accurate (smart sensors). Existing implementations of such systems range from simple miniature sensing systems which monitor only the amount of motion over several days (activity monitoring) using simple uni-axial sensors (e.g. IDEEA and Physilog systems) to methods which try to assess the full 3D kinematics [BKM<sup>+</sup>04] [ZHHH06]. Some motion capture suits or exoskeletons have been developed for the complete body (e.g. Gypsy, ShapeWrap). Limitations of such systems, with regards to optical ones are the relatively low sampling rate and accuracy (e.g.  $90Hz$  and  $0.5deg/2mm$  for ShapeWrap).
- The last system, which is currently the most used, is the optical motion capture (e.g. from Vicon Motion Systems Ltd). It is based on the acquisition, by several digital cameras, of reflective markers placed onto the body. The sampling rate typically ranges from  $30Hz$  to  $1000Hz$ . The motion reconstruction is very accurate ( $\sim 0.5mm$ ) but is not performed in real-time because occlusions arise when markers are seen by less than three cameras. Therefore, many adjustments are needed, in the recording phase (e.g. marker setup, camera configuration) and in the post-processing phase. In order to improve to accuracy, the capture volume can be adjusted (typically between  $1m^3$  and  $2m^3$ ).

Because only the skin deformation is measured with motion capture and because joint centers are unknown, bone motion need to be estimated, combining several redundant markers. This is done by taking into account joint degrees of freedom and by minimising skin/bone relative displacement (e.g. least square minimisation, 3.3.5) [LO99]. The error is exacerbated by the common use of generic skeleton models. However, in recent studies researchers have tried to reduce these errors using subject-specific models and skin/ bone displacement models [YCGMMT04] [YCMT06]. Other authors have used invasive bone implants as markers [LCSK92], to get rid of skin deformation artifacts [FLMM97]. This is also a common practice in surgical navigation.

**Video-based methods** - Instead of MoCap, raw video images or laser scans can be used to assess musculoskeletal kinematics. Whereas MoCap provides accurate and high-rate motion of the skin at sample points, it can be useful to assess the skin with more complete spatial information. However, video-based methods have a lower accuracy and frame-rate, and require efficient post-processing methods (computer vision techniques). In [DHW<sup>+</sup>04], a dynamic body scanner system ( $15Hz$ ) is used for spine dynamic analysis where internal anatomical structure are estimated for the shape of the skin. Commercial products using stereo photogrammetry (e.g. 3dMd) can achieve around  $60Hz$  for imaging body parts. In [ACP02], skin animation is performed from scattered scan data, through interpolation techniques.

**Force plates** - For dynamic simulations of the musculoskeletal system, it is necessary to evaluate external forces acting on the body, such as ground reaction forces (see Section 2.10). MoCap, EMG and force plates are typically associated. Force plates can record ground reaction forces (6 DOF) [DP02]. The subject should land in the middle part of the force plate (this is a reason for failed recordings and unnatural subject behaviour). Alternatively, one can use pressure measurement soles [WC03], leading to slightly different results (shoe/ground vs. foot/shoe interaction). Both systems can record at approximately  $500Hz$ , with a resolution of about  $1cm$  (e.g. Footscan systems).



### 2.3.3 Physiology acquisition

**Electromyography (EMG)** - The measurement of muscle stimulation patterns is essential to understand musculoskeletal dynamics and control (see Section 2.10), and consequently the causes of abnormal movements. Electromyography (EMG) consists in assessing electrical signals ( $0 - 6mV$ ,  $10 - 500Hz$ ) being generated by a muscle at the same time that it contracts [Cut93]. Surface EMG (sEMG), where electrodes are placed at the skin above muscles, is a non-invasive approach for muscle stimulation recording. sEMG allows superficial analysis contrary to invasive approaches (needles). It has been used in many studies regarding the forward dynamic simulation of the musculoskeletal system, the validation of inverse dynamic simulation, and the diagnosis of neuro-musculoskeletal disabilities [Pan01] [DL95]. Due to differences in electrode properties, configuration and location, which is problematic for the use and exchange of sEMG data, standard protocols have been recently established [HFDKR00]. Indeed, a good electrode/skin contact need to be insured to reduce noise, and a precise electrode placement to minimise crosstalk from neighbouring muscles. To tackle this, researchers have investigated more sophisticated electrodes such as bipolar electrodes and more recently electrode arrays. In addition to the difficult setup of EMG, the main drawback is that only active muscles are analysed, contrary to Magnetic Resonance Elastography. EMG data need to be post-processed to extract relevant information and to allow intra and inter subject comparison [DeL97]. Because there is no absolute signal value, EMG are generally normalised according to the maximum voluntary contraction or MVC (highest peak during maximum effort). Removal of measurement noise is performed through frequency filtering, window averaging, or the selection of relevant wavelet coefficients. Muscle activity can be measured by considering amplitudes, amplitude root mean squares (RMS), the number of zero crossing or the local integrated value. In addition, EMG analysis in the Fourier domain is particularly useful to separate the contribution from the different muscles, and analyse physical fatigue. Indeed, fatigue is characterised by an average frequency shift towards lower frequencies.

**Mechanomyography (MMG)** - MMG is a non-invasive technique for measuring muscle contraction speed of muscle groups [OLL<sup>+</sup>96] [BHJ<sup>+</sup>04], and is the mechanical equivalent to EMG (electrical activity). Its principle is to measure skin displacement under neuromuscular stimulation. Measurements are commonly made through a variety of transducers (condenser microphones, piezoelectric contact sensors or accelerometers) leading to various output names: muscle sound, soundmyogram, acousticmyogram, phonomyogram, accelerometermyogram or vibromyogram.

### 2.3.4 Mechanical testing

**Mechanical devices** - Biological tissues mechanical parameters (see Section 2.2.2) can be established from various modalities measuring the uniaxial/biaxial tissue strain response in relation to mechanical or ultrasound impulses. Indeed, the relationship between the strain (tension, compression, bending or torsion) and the stress (pressure or force) provides the elastic property of the material, while viscoelasticity can be analysed from stress and strain rates (stress-relaxation and creep testing) over multiple loading/unloading cycles. The biaxial stress-strain relation of a soft tissue can not be obtained by superimposing two individual uniaxial test results due to the large deformation, but data regression methods are usually performed to fit the constitutive model to data. Commonly, tissue specimens need preconditioning (several loading/unloading cycles) before obtaining reproducible results. Then, parameters are established from the loading and unloading curves. The use of dead specimens is problematic, as they may behave differently than living tissues: they need to be correctly hydrated (e.g. saline baths) and kept at a suitable temperature. In addition, tissue initial milieu is difficult to reproduce (e.g. pressurised synovial joints). In-vivo mechanical measurement of internal tissues poses extreme experimental difficulties, but is becoming possible through the development of miniaturised devices. Buckle transducers, telemetered pressure sensors and strain gauges have been investigated for in-vivo force, pressure and deformation measurement. Non-invasive approaches are also available for isotropic materials: Nava et al. [NMK<sup>+</sup>03] use an aspiration tool and analyse the shape of the sucked tissue. From the same group, Valtorta et al. [VM04] use a torsional resonator device to measure the viscoelasticity. Highly anisotropic materials (e.g. ligaments) are, however, difficult to measure in situ.

Implantable force transducers and video-based measurement techniques have been used, but the reference state (zero stress) remains problematic to determine [WG01].

**Magnetic Resonance Elastography (MRE)** - MRE is a new technique to measure material internal stiffness in vivo and non-invasively [MRL<sup>+</sup>96] [DRR<sup>+</sup>01]. It is based on the measurement of tissue response under shear waves, propagated from a transducer attached to the skin surface. The stiffer is the tissue, the more rapidly waves are propagated (longer wavelength). Wave displacement velocity is measured with Phase-Contrast MRI (pcMRI), whereas in ultrasound elastography, the measurement is based on Doppler techniques. MRE has been applied to measure muscle actuation, exploiting the fact that the more contracted is a muscle, the stiffer it is. MRE and EMG have shown good correlation [HJD<sup>+</sup>03]. In [PBHS05] muscle anisotropy (3 parameters) has been measured with MRE, based on a transversely isotropic linear material. MRE is a very promising approach allowing tension distribution within a muscle to be imaged. In addition, contrary to EMG, MRE can provide information about passive properties of muscle.

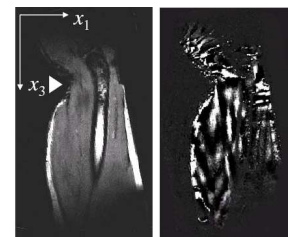


Figure 2.18: MRE measurement based on a harmonic shear excitation in  $x_1$ -direction, from [PBHS05]

### 2.3.5 Summary

Table 2.19 shows that a broad range of modalities are available for musculoskeletal acquisition. In fact, MRI appears to be the more flexible modality as it covers anatomy, dynamics and physiology measurement, while offering a good trade-off between resolution, acquisition time and signal-to-noise ratio, and no harmful radiation. However measurement limitations, in terms of accuracy (geometric and photometric distortions/artifacts) and reproducibility, need to be checked carefully (e.g. using phantoms) before proceeding to validation and parameter extraction experiments, from acquired data. Static and dynamic MRI will be extensively used in this thesis, and combined with other modalities (body scanner, MoCap) that offer larger field of view and range of motion.

## 2.4 Bone modelling and simulation

### 2.4.1 Anatomy

Bones are 95% solid (vs. water). The solid part is 65% mineral (mainly calcium, responsible of bone rigidity) and 35% organic (mainly collagen fibers responsible of the structure). They are commonly classified into five categories:

- *Long bones*, such as the femur and the tibia, are found in limbs, and consist of a curved and cylindrical body (or shaft, or diaphysis) with a central cavity (medullary canal) filled with marrow, and two extremities.
- *Short bones*, such as in the carpus and the tarsus, are combined to provide to the skeleton strength and compactness.
- *Flat bones*, such as the pelvis or skull bones, provide large surfaces for muscle attachment, as well as an extensive protection.
- *Sesamoid bones*, such as the patella, are small bones surrounded by tendon, used to transmit forces and modify pressure/friction.
- *Irregular bones*, such as the coccyx and the mandible, are the remaining bones that cannot be categorised, due to their unusual shape.

Model	Methods	Modalities		MRI								X-rays CT	US	Body scanner	BIA	Mocap	EMG	MMG	Force plates	Video- based methods	Mech. Device
		Data	Shape Fiber direction	Static MRI	Sequ. MRI	Dyn. MRI	Dt MRI	(Cine) Pc MRI	(cine) tagged MRI	MRE											
Anatomical model	Segmentation	Static	Shape Fiber direction	++	+							+	-								
Kinematical model	Registration	Kinematics	Motion		+	++	+					+		++							
Dynamic model	Inverse dynamic simulation	Dynamics	Ext. Forces Int. Forces															+			
Constitutive model	Physical- based simulation	Mechanics	Strain Stress		-		..			+			+						+	++	
Actuation model	Forward dynamic simulation	Physiology	Muscle actuation																	+	++
		<b>Invasiveness</b>										+									+

Figure 2.19: Musculoskeletal acquisition modalities

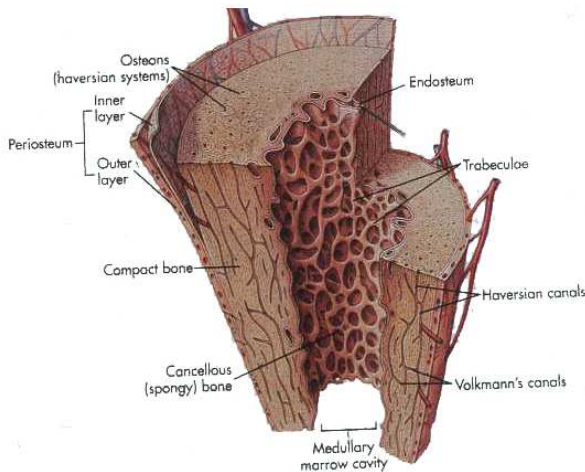


Figure 2.20: Long bone (Image from the comparative morphology center, University of Bristol)

Bones are composed of three types of tissue:

- *The compact tissue* is the harder, outer tissue of bones.
- *The cancellous (or trabecular) tissue* is the sponge-like and porous tissue inside bones.
- *The subchondral tissue* is the smooth tissue at the ends of bones, which is covered with cartilage.

Bone, except in its cartilaginous extremities, is enclosed in a fibrous membrane called the periosteum. Periosteum inner layer contains vessels (periosteal vessels) through which blood can nourish bone tissue. Ligaments and tendons are attached to the outer layer of the periosteum, which is made of compact bone. Both compact and trabecular bone tissues commonly operate in the linear elastic domain. The compact bone is stiff, approximately linear elastic, transversely isotropic and relatively homogenous (Figure 2.21). However, if bone is subject to very rapid loads that would produce a viscoelastic response or very large loads that would cause fracture, linear elasticity would

not be a valid assumption. Bone Young's modulus is typically of 12 GPa (longitudinally) and 13.4 GPa (transversely). A exhaustive review of bone mechanics has been done in [Cow01].

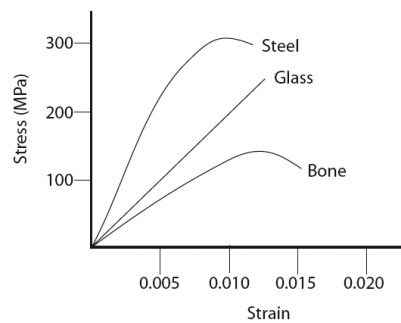


Figure 2.21: Bone stress-strain curve, from [Haw02]

## 2.4.2 Volumetric representations

Historically, FEM was first introduced to biomechanics in 1972 [BPS72] for 2D stress analysis of the femur. Since then, FEM has been increasingly used for prosthesis design, fracture fixation, bone remodelling, fracture assessment, in addition to stress analysis. The linear/small strain behaviour of bones makes them particularly suited for FEM analysis. In FEM, bones are represented by volumetric elements (usually tetrahedrons or bricks) and possibly surfacic elements. As an example: Dalstra and Huiskes [DH95] use 2600 bricks to model trabecular/subchondral plevic tissue and 600 shell elements to represent the compact membrane of the pelvis, and assess load transfer. 10 years later, Majumder et al. [MRP04] use 72000 tetrahedrons/17000

shells to dynamically simulate the pelvis during gait, including muscle effects. Beillas et al. [BPTY04] present a knee simulation based on FEM including cartilage, ligaments and muscle effects, for contact modelling, with comparison to in-vivo data. Computation time is high ( $\sim 12h$  for the computation of a impact of  $0.4s$  with a time-step of  $0.3\mu s$ ).

Another widely used type of representation of bone is the voxel-based representation, since one can directly use medical images or segmented images (no reconstruction). CT/ X-ray imaging provides an excellent bone contrast with regards to other tissue, leading to simple bone extraction and volume visualisation techniques (Section 3.2). On the contrary, MRI requires more robust methods such as level-sets [KHC05] or deformable models [GPV<sup>+</sup>04] (Section 3.3). Statistical methods based on shape estimation from scattered points (e.g. intra-operatively digitalised or acquired superficially from US) have been also presented [RST<sup>+</sup>06] (Section 3.3.4). The Voxel-Man [HPP<sup>+</sup>03] is an example of voxel-based anatomical atlas using visible human data [NLM99]. In [PLP<sup>+</sup>02], Pflessler et al. developed an interactive application for bone virtual surgery (drilling) based on voxel volumes.

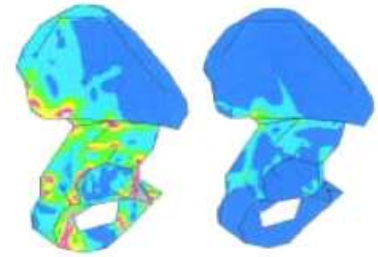


Figure 2.22: Max/min stress distribution on the pelvic membrane during gait [MRP04]

In some computer graphics applications, manually designed implicit models (Section 3.3.2) are used for fast collision detection [SPCM97] [TT93]. In these examples, bones are made of implicit primitives (ellipsoids, spheres, cylinders, tori, hyperboloids, bilinear patches, etc).

### 2.4.3 Surfacic representations

The most intuitive way of representing rigid objects is using surfaces. Indeed, the internal part is not useful for most applications, since bone mechanical behaviour is generally not considered (infinitesimal strain). A surface model is a realistic bone representation, allowing the support of other tissues, and the analysis of bone shape and joint kinematics. Generic discrete triangle models have been widely used [PN98] [WG97] [Aub02] [NTH00] [MWT98] since they are efficiently supported by graphics hardware. However, some researchers have preferred parametric surfaces such as B-Splines or bicubic-Hermite surfaces [FMT<sup>+</sup>03], to enforce surface regularity and smoothness from a minimal number of control points, that can be digitalised from medical images. In biomechanics, the estimation of muscle moment arms highly depends on the position of joint centers [PL05]. Joint limits are also important for realistic virtual human animation and, in orthopaedics, to quantify possible joint degeneration [KDG91]. Considering only bone-to-bone surface contacts, there are some attempts to functionally determine joints centers and ranges of motion from patient-specific models [KSMMT03] [MT00]. These techniques can be integrated in motion analysis systems, reducing errors (up to  $30mm$ ) of traditional methods based on motion capture data and stick figure skeleton used for their calculation [YCGMMT04] [YCMT06].

### 2.4.4 Stick-figure representations

The structure of the body is principally attributable to its skeleton, which is a set of interconnected rigid objects. As a consequence, hierarchical sets of line segments oriented in space have been used since earliest drawings for representing human and human movements. On these segments, more precise bone representation (surface, volume) can be wrapped (see Figure 2.23). Eshkol et al. [EW58] in 1958, have first formalised these models. Later, articulated figures have been naturally used in computer science [Wit70] [BS79] [MTT90]. In stick-figure representation, the different segments represent limbs and are connected at joints. So, contrary to orthopaedics and biomechanics where a precise modelling of bone surface is required, computer graphics applications only require precise joint centers. Joint center estimation is achieved

from optical motion capture data through error minimisation [YCMT06], from dynamic medical images through motion tracking methods [TA03] [GPMTV04], or from anatomical models through functional methods [KSMMT03]. Joint degrees of freedom are exploited to derive or analyse postures of the articulated figure. The H-Anim standard [HA] is a widely used hierarchical structure for representing humanoids based on articulated skeletons. The current H-Anim implementation consists of 94 joints. Though a stick-figure is not a realistic human representation, it is generally handled as an underlying representation of the skeleton on which a body surface is wrapped (see Section 2.9). In traditional computer animation, joint angles are user-specified at key frames, whereas they are interpolated (using spherical quaternions, matrix blending or similar techniques) for intermediate frames. Alternatively, inverse kinematics [BMT97] can be used to derive joint angles based on a set of position and orientation constraints applied to the skeleton. Another technique is to measure skin motion with goniometers, flock of birds or optical techniques and establish from it, joint centers and angles, using least square error minimisation (Section 3.3.5). [MBT99] and [SMEH99] are examples of motion capture-based skeleton animation techniques.

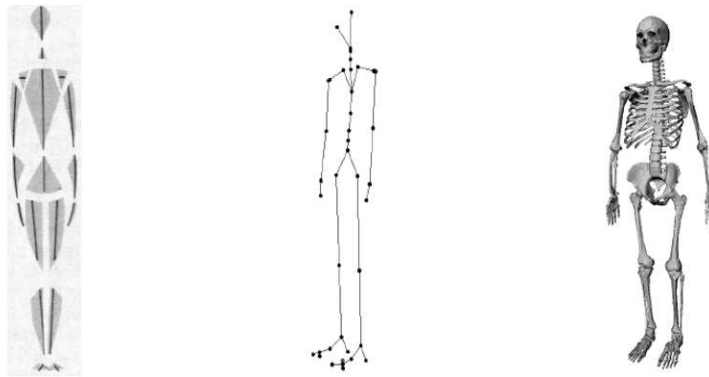


Figure 2.23: Eshkol model [EW58], stick-figure skeleton [PN98] and skeleton surface [PN98]

### 2.4.5 Discussion

We have seen that, from truthful but partial representations using volumes to comprehensive but simplified representations using line segments, the skeleton can be represented at different scales of complexity. Joint animation can be also multi-levels: for instance the knee can be considered as a hinge (ideal) joint, an ellipsoidal joint or an ellipsoidal joint with sliding axis. In fact, there is a trade-off between computational load and perception/ accuracy constraints. A current research topic in computer graphics, is the investigation of scalable skeletons (levels of articulation) [HA] [DJGMT03]. In this thesis, a discrete surfacic representation will be applied to capture patient-specific bone shapes. From them, volumetric models can be derived to simulate bone mechanical behaviour. We use standards bone coordinate systems based on anatomical features [WSA<sup>+</sup>02] to derive medically relevant angles (e.g. flexion, abduction, etc.).

## 2.5 Muscle/tendon modelling and simulation

### 2.5.1 Anatomy

There are approximately 650 skeletal muscles in the human body, making up from 40 to 45% of the total body weight (largest fraction of body mass in nonobese adults). Their function is to produce articular movement. They are attached to tendons through which they can exert forces on bones. Bone/tendon attachments can be large, and their site is correlated to bone shape with few inter-individual variability [KdH04]. Skeletal muscles generally have two attachment sites known as the origin (closest to the mass center) and the insertion. Thanks to their hierarchical structure (Figure 2.24), muscles can produce forces at a

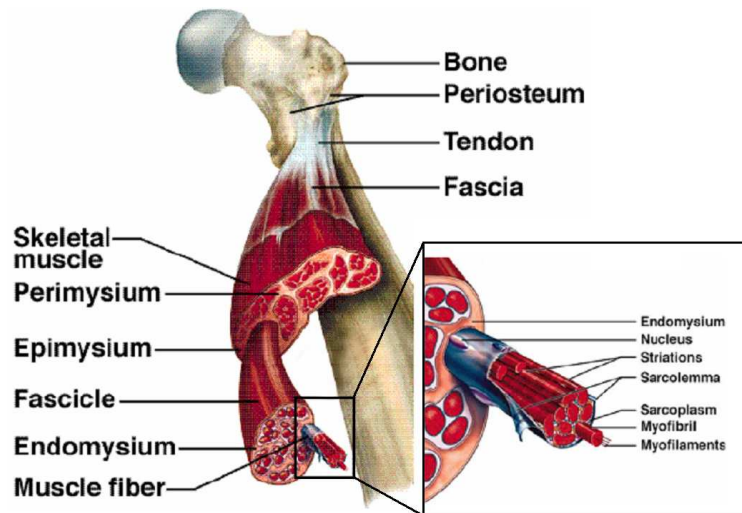


Figure 2.24: Muscle layers [Sal98]

macroscopic scale due to microscopic events. According to the prevalent cross-bridge theory [McM87], muscle force is due to chemical bonds at the myofilament level. Consequently, muscle fiber orientation (pennation) has a direct impact on the production of skeletal forces. There are different fibre arrangement patterns in skeletal muscles, which are classified as in Figure 2.25. Muscles are incompressible, hyperelastic, viscoelastic and fiber-reinforced materials. Tendons (see Figure 2.38), made up of parallel-aligned collagen fibres, are much more stiffer than muscles (usually elongation  $<5\%$ ) and are used to store elastic energy. Muscles can undergo isometric contractions (no change in muscle length, when muscle forces exactly compensate external forces) or isotonic contractions (change in muscle length). During isotonic contractions, muscle can shorten (concentric contraction) or lengthen (eccentric contraction).

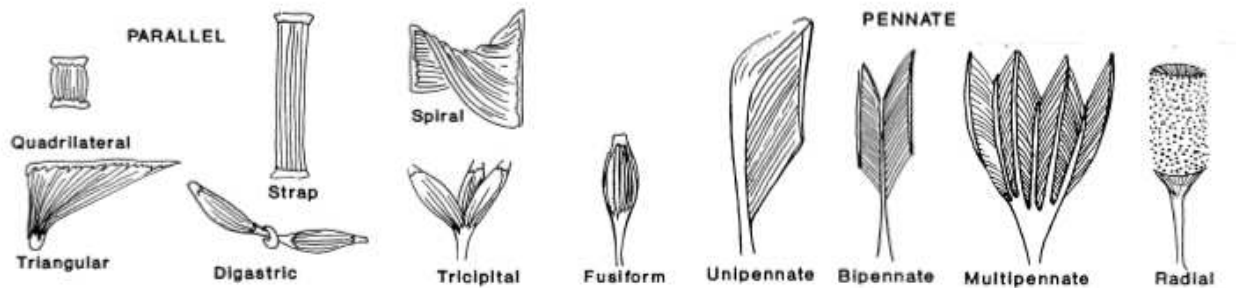


Figure 2.25: Muscle fiber arrangements (pennations) [Cut93]

## 2.5.2 Muscle actuation modelling

Within a muscle, all fibers may not be stimulated at the same time, as they are excited by different motor neurons, and because muscle electrical property is anisotropic [GdJB84] [EF83]. For simplicity, a muscle is commonly associated to a scalar activation level  $a^M(t)$ , comprised between 0 and 1 (full activation). This activation is driven by the frequency of neural excitation pulses, corresponding to a certain neural stimulation  $u^M(t)$ , comprised between 0 and 1 (full stimulation). The delay between the neural stimulation signal  $u^M(t)$

and the muscle activation level  $a^M(t)$  can be modelled as a first-order process [Zaj89]:

$$\tau_a \dot{a}^M(t) + (\tau_a + (\tau_d - \tau_a)u^M(t)) \cdot a^M(t) / \tau_d = u^M(t) \quad \text{where } \tau_a \text{ and } \tau_d \text{ are time-constants}$$

Typically,  $\tau_a$  range from 12-20 ms (activation) and  $\tau_d$  ranges from 24-200 ms (deactivation) [Pan01]. We can assume they are 15 and 50ms, respectively, such as in [Zaj89].

Given a particular actuation level, the muscle force generating property is highly non-linear. That is the force is not proportional to the elongation. A fundamental model, still widely used, has been established in 1938 by Hill [Hil38]. Later, Zajac [Zaj89] has extended this model to account for the pennation angle. They make the hypothesis that muscles can be represented by a series element simulating tendons elasticity, a parallel element accounting for muscle elasticity at rest, and a contractile element, freely extensible at rest but capable of shortening when activated. All these assumptions are not true in reality, however Hill-based models is relatively simple and have shown a good correlation with experimental data for a range of muscles with different architectures [PZ91]. The model is described through three relationships (Figure 2.26): 1) the force-length curve accounts for muscle contraction with regards to the activation level (linear combination between an active and a passive quadratic behaviour); 2) the force-velocity hyperbolic curve describes the viscoelastic damping and the muscle loss in tension when shortening; 3) the tendon force-strain quadratic relationship accounts for tendon elasticity, predominant with regards to sarcomere elasticity. Zajac has normalised these relationships (dimensionless curves), for further scaling from muscle-specific parameters:

- $F_0^M$ : maximum isometric muscle force
- $l_0^M$ : optimal muscle fiber length
- $l_S^T$ : tendon slack length
- $\alpha$ : pennation angle
- $v_{max}^M$ : maximum contraction velocity of muscle.

We commonly assume the following [Pan01] [NTH00]:

- $F_0^M = 25N/cm^2 \times PCSA$  where  $PCSA$  is the physiological cross-sectional area
- $v_{max}^M = 10s^{-1}$

$l_0^M$ ,  $l_S^T$  and  $\alpha$  need to be estimated from literature, cadaver dissections or medical images (MRI, dtMRI). Tendon rest length  $l_S^T$ , who has a significant impact on actuator performances [Pan01], is however difficult to measure.  $PCSA$  can be obtained from a simplified geometric muscle model taking into account muscle volume, length and pennation [NTH00]. As a result, with the Hill's model, the musculotendon dynamics is described by a single, nonlinear, differential equation relating musculotendon force  $F^{MT}$ , musculotendon length  $l^{MT}$ , musculotendon shortening velocity  $v^{MT}$ , and muscle activation  $a^M$ :  $F^{MT} = f(f^{MT}, l^{MT}, v^{MT}, a^M)$

The fundamental Hill's actuation model is typically adapted to mechanical models accounting for muscle structure, and spatial constraints: fiber local direction, attachment areas, volume preservation and non-penetration constraints with bones and other muscles. Due to computational limits and depending on the application domain, a wide range of models and simulation methods have been developed, from simplified/fast to accurate/slow ones. Simple models, as a direct application of the Hill's model, make the assumption of parallel internal forces (identical sarcomeres), leading to 1D muscle geometric models (action lines) [DLH<sup>+</sup>90]. Whereas, complex volumetric models accounts for internal force local direction [WMG96] [BD05]. Between the two, some hybrid models combine muscle shape models (surfaces) and action lines [NTH00] [PN98]. In computer graphics, the muscle force generating property is ignored in most cases (no actuation model). However, geometric methods have been developed to reproduce realistically muscle deformation using hybrid models, where radial forces are applied to surfaces depending on the length of underlying action lines [Aub02] [PN98].



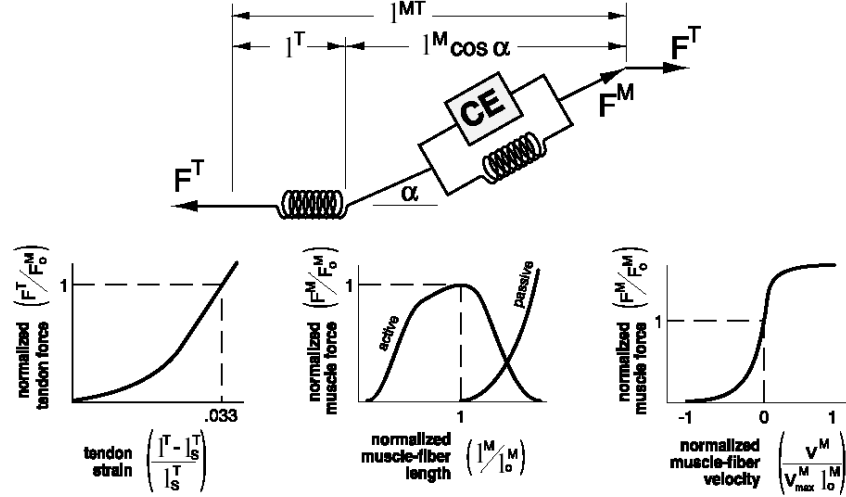


Figure 2.26: Zajac's dimensionless musculotendon model [Zaj89]

### 2.5.3 Action line representations

The action line is the most used representation for muscles, since it provides a simple abstraction of muscular fiber complex, in terms of geometry (muscles have a tubular-like shape), action (muscles exert forces along their principal axis) and reaction (muscles primarily resist to tensile forces along their principal axis). Indeed, representing muscles as a linear or non-linear (Hill's model) spring drastically reduces the complexity of the system. Hill's based action lines have been successfully applied to simulate tendon transfer [BT87], tendon lengthening [DSC95], osteotomies [SACD99], muscle compensation [KN04], and for rehabilitation and educational training [TSF05]. Action lines are used either as a full representation of muscle, either as a support of a wrapping surface on which radial constraints are applied (Section 2.5.5). An action-line aims at representing both the muscle path and the muscle force direction, however, this concept is not strict, and different ways of modelling action-lines have been presented as stressed by Jensen and Davy [JD75]:

- **Straight line approach:** muscle attachment centroids are bound by a segment. This method is easy to implement but is not accurate, since muscle wrapping around bones or other muscles are not reproduced.
- **Muscle centroid approach:** a piece-wise line is defined by muscle cross-section centroids. This approach is more precise, but the extraction of muscle centroids is difficult.

The centroid approach has been applied in most frameworks for musculoskeletal dynamics analysis [Cha03] [BT87] [DLH<sup>+</sup>90] [DL95] [RDS<sup>+</sup>03] (see Section 2.10). To tackle the problem of measuring muscle centroids, most studies make use of cadaveric specimens and/ or medical images, in order to reconstruct a static generic model [ASAD00]. This model can be further scaled according to individual measurements. In [ABD01], Arnold et al. stress the importance of this individualisation part for computer-aided surgery in patient with cerebral palsy. They compare scaled and unscaled generic models for the determination of normalised muscle length. Given an individualised model, another problem is to determine how muscle path changes during motion. One way of approximating centroid motion is to introduce few via points defined from wrapping primitives where muscle path is assumed to remain fixed relatively to bones [DLH<sup>+</sup>90] [DL00] [ASAD00] (Figure 2.27). In [GP00], Garner and Pandy allow action lines to slide freely over bones (obstacle-set method), in order to avoid some discontinuities in the computation of moment arms. In [ASAD00], it has been shown that moment arms determined with action lines over a range of hip/ knee motion remain within 10% of experimental data.

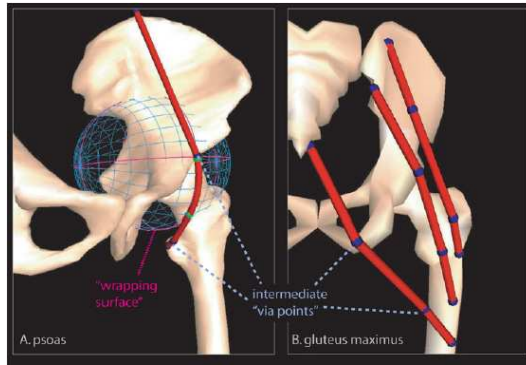


Figure 2.27: MRI-based action-lines, wrapping surfaces and via-points reconstruction, from Arnold et al. [ASAD00]

## 2.5.4 Volumetric representations

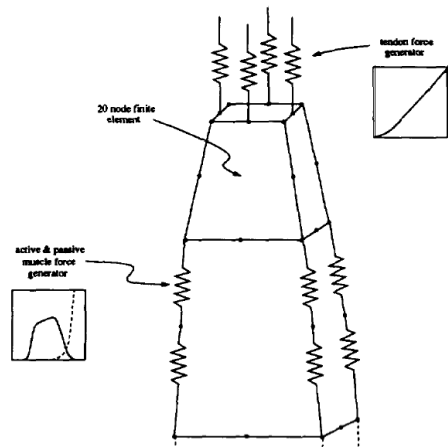


Figure 2.28: FEM muscle model, from Chen et al. [CZ92]

Volumetric models, including finite elements, particle networks and implicit models, are supports for object mechanical behaviour estimation, through internal matter representation. In addition to a high accuracy, volumetric representation leads to simplified volume and contact computation, which is useful for applying volume preservation and non-penetration constraints. Volume discretisation produce interconnected particles, forming elements (tetrahedrons, bricks, etc.) on which physical-based approaches are applied (see Section 2.2.3).

Chen et al. [CZ92] developed a simplified linear finite element model of skeletal muscle based on Zajac's dimensionless model. Passive muscle properties are approximated, considering muscles as a homogeneous, incompressible, linear, isotropic and viscoelastic material. To decrease computational complexity, they use a simple 20 nodes finite element brick lattice, in which the muscle surface is embedded. Muscle deformation is derived from lattice deformation by Free-Form-Deformation techniques (see Section 2.9.3). Zhu et al. [ZCK98] have used a similar method with a voxel-based brick lattice, corresponding to segmented medical images (visible human [NLM99]), and a re-sampling technique for scalable simulation. Contacts are not modelled with these methods, and muscle internal architecture is not taken into account, leading to unrealistic models.

Hirota and al. [HFS<sup>+</sup>01] have presented a study on the lower limb (visible human data) with a few muscles based on tetrahedral FEM. Muscles are modelled by a Veronda material (non-linear, isotropic, hyperelastic) with an increasing stiffness near tendons, whereas connective tissues are based on a Mooney-Rivlin material (non-linear, isotropic, hyperelastic). The originality of the method lies in the contact penalty force computation based on material distance map. They are able to model fold and crease patterns. However, the force generating property and muscle anisotropy are not modelled. Moreover, there is no validation of their method (not quantitative comparison with dynamic data).

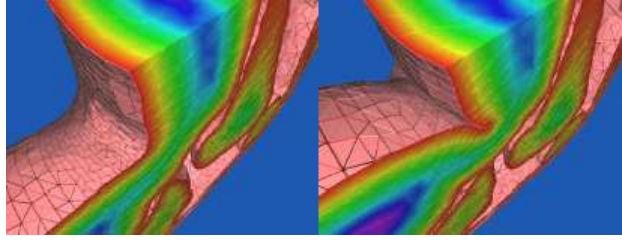


Figure 2.29: FEM musculoskeletal system, from Hirota et al. [HFS<sup>+</sup>01]

In addition to common material elasticity, viscoelasticity, fiber-reinforcement and volume preservation properties, constitutive relations have been established for muscle actuation [WMG96]. The Hill's relationships is expressed locally, through the local fiber stretching, instead of the global musculotendon length. Some authors have applied these constitutive relations to volumes, where fiber direction are obtained through the wrapping of a template fiber arrangement model (see Figure 2.25) [BD05] [JMB00] [LEHW01] [MWTT98]. Blemker et al. [BD05] validate their prediction model towards manually reconstructed models from MRI (4 muscles, 2 positions, 1 subject) and the mean distance between surfaces was  $3mm$ . However, computation charge is high for this kind of methods (one resolution FEM and no speeding up through geometric methods).

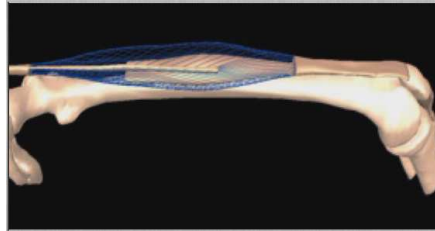


Figure 2.30: Rectus femoris volumetric model including template fiber arrangement, from Blemker et al. [BD05]

The most advanced musculoskeletal simulation framework, in terms of comprehensiveness and computational speed/ complexity trade-off, is probably the one presented by Teran et al. in [TBHF03] [TSB<sup>+</sup>05]. They use a finite-volume method (FVM), which is equivalent to a first order finite-element method, assuming constant strain tetrahedra. In addition, the simulation takes advantage of a new robust finite element technique that handles both degenerate and inverted tetrahedra [TSIF05]. State-of-the-art action lines-based inverse dynamic simulation [GP01] is used for moment arms and activation patterns computation, allowing motion specification from motion capture or traditional key-frame animation. In [SNF05], they determine muscle activation patterns by minimising the distance between simulation results and motion capture data. They embed a constitutive muscle model (transversely isotropic and quasi-incompressible) based on Green-Lagrange strain and Cauchy or 2nd PK stress. Their model includes the Mooney-Rivlin rubber-like model for representing the isotropic ground substance [Moo40], an incompressibility term from [WMG96], and the active and passive muscle fiber response from the Zajac's model [Zaj89]. However they neglect muscle history dependent changes in elasticity (strain hardening), muscle force/velocity relationship and the anisotropic shear behaviour relative to the fiber axis. They model muscles fibers using B-spline solids similarly to Ng-Thow-Hing et al. [NTH00] and assign fiber directions to individual tetrahedrons. This permits the modelling of a non-uniform distribution of fibers within a single muscle. In [TSB<sup>+</sup>05], they extend their method to reduce the complexity intrinsic to a large musculoskeletal complex such as the upper limb. They present a dynamic Free-Form-Deformation embedding scheme based on a non-manifold body-centered cubic (BCC) tetrahedral lattice. It allows a 10-fold reduction of tetrahedrons number (from

10 millions to 1 millions tets to represent 30 muscles) and the time step restriction for stability is relaxed by a factor of 25 (due to the regular structure of the BCC tetrahedra and the elimination of poorly shaped elements). The dynamic simulation is performed by a semi-implicit scheme. Contacts and fascia effect are modelled through anchors and links constraining BBC lattice deformation. The overall simulation runs at the rate of 4 minutes per frame. Nevertheless, their method lacks of validation (no comparison towards dynamic images) and the individualisation of their model was not explored (manually reconstructed model from visible human data).

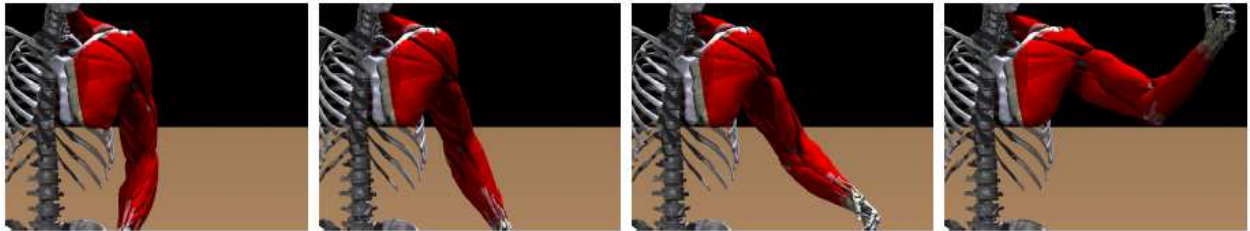


Figure 2.31: FVM-based upper limb muscles simulation, from Teran et al. [TSB<sup>+</sup>05]

Manually designed implicit surfaces also referred as "Metaballs" or "Blobbies" have been widely used for representing and simulating muscles or muscle complex, in a computationally effective fashion. They are defined as an isovalue of a potential field:  $F(x, y, z) : \mathbb{R}^3 \rightarrow \mathbb{R}$ , leading to simple inclusion tests (e.g. skin modelling in [TT93] [TSC96] [SPCM97]). Their blending capability allows the potential field to be calculated from several primitives (sources) such as  $F(x, y, z) = \sum_i F_i(x, y, z)$ . Geometric primitives, although they do not fit to individual organ boundaries, can realistically represent the whole muscle complex through primitive blending. Some authors have used anatomy-based implicit primitives to model muscle shapes. Ellipsoids are particularly convenient for simulating fusiform muscles because isometric and isotonic contractions can be easily reproduced through principal axis modification. Given a local coordinate system, the ellipsoid equation is:  $f(x, y, z) = x^2/a^2 + y^2/b^2 + z^2/c^2 - 1 = 0$ . Principal axis modification is done through  $a$ ,  $b$  or  $c$  alteration. Volume conservation is simply performed by keeping the product  $abc$  constant, the volume being  $4\pi abc/3$ . This technique has been applied by Scheepers et al. [SPCM97] and Wilhelms et al. [WG97] for modelling human and animal musculature. Scheepers et al., in addition to ellipsoids, use piecewise lines to model tendons. Kahler et al. [KHS01] [KHYS02] use interconnected ellipsoids along a piece-wise line for modelling facial muscles. These methods have not been validated so far.

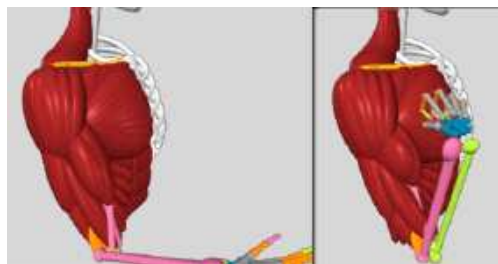


Figure 2.32: Implicit model-based upper limb muscles simulation, from Scheepers et al. [SPCM97]

### 2.5.5 Hybrid representations

Hybrid representations for muscles, including a surface (shape) and internal features (springs representing fibers or action-lines), have been developed to allow cost effective muscle dynamic simulation, and particularly

the reproduction of isometric and isotonic contractions. Interaction forces are applied between the surface and internal features. Surfaces are more flexible, allow simpler interaction and can be more efficiently controlled than implicit surfaces (Section 3.3.2). Moreover, they are obviously cheaper than volumetric meshes in terms of memory and simulation time.

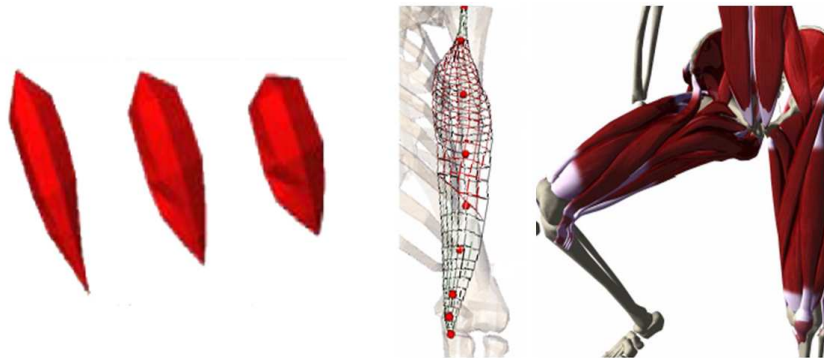


Figure 2.33: From left to right: Fusiform muscle wrapping onto an action line, from Porcher-Nedel et al. [PN98] and muscle surface deformation driven by underlying action-lines, from Aubel et al. [Aub02]

Porcher-Nedel et al. [PN98] use the idea of representing muscles as piece-wise linear function from biomechanics. They introduce a method to wrap a surface according to its action-line. Their muscle models are triangle meshes enhanced with elastic and angular springs to control muscle local deformation. However, their method is limited to fusiform muscles (from the visible human), and has not been validated. Aubel et al. [Aub02] extend this method, using a multilayered approach composed of triangle meshes. They use cubic Bezier splines to represent action lines and can model any kind of muscles (several action lines per muscle). Muscle vertices displacement is derived from action line node local frame coordinates and a radial scaling function from cross-sections (simulating volume conservation and isometric contractions). A similar approach is presented in [ZLW03]. A muscle vertex can be influenced by two action lines using bilinear interpolation of the displacement (to insure relatively smooth constraints along muscle surface, although not rigorously continuous). Muscles with large attachment areas and/ or several origins/ insertions require many action lines, that need to be designed interactively. Collisions handling is performed between action lines and bones from a repulsive force field. However, they do not compute exact contacts and volumes. They do not simulate the force generating property of muscle and do not validate their method (computer graphics application). Thanks to a reversible medial representation (medial surface), we present in this thesis a new hybrid model from which exact contacts as well as continuous radial constraints can be computed.

Ng-Thow-Hing et al. [NTH00] model muscles and ligaments using parametric surfaces (B-Spline) and present appropriate methods for computing control points from spatial constraint points. They introduce a continuous volume sampling function to define muscle fiber control points with generalised coordinates. A Lagrangian evolution is applied to a viscoelastic (spring) network composed of interconnected surface or internal spatial points. They can simulate exact volume preservation as well as muscle fiber-driven deformation. The originality is to use anatomical-based internal constraints (fibers) from cadaveric specimens instead of geometric medial lines. To obtain more realistic fiber forces, they apply the Hill model instead of linear springs, but do not compare computed deformations with real data. In [FMT<sup>+</sup>03], Fernandez et al. also use parametric surfaces (tricubic Hermite splines) to model muscles. Since they are smooth, muscles can be represented with a minimum number of control points.

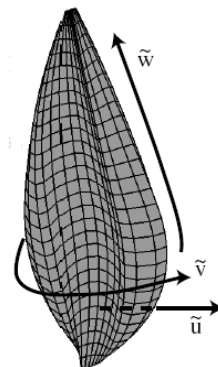


Figure 2.34: Parametric muscle model, from Ng-Thow-Hing et al. [NTH00]

Extended action lines (with local frames computation) that are able to handle full 3D elasticity problems (e.g. twisting, bending) and not only elongation, have been introduced, particularly for rope and hair modelling. Strands, a primitive introduced by Pai [Pai02] have shown to be particularly efficient for dynamic simulation. Strands have been used in [PSW05] for fast musculoskeletal simulation (including thin muscles of the forearm). They incorporate muscle/ bone and muscle/ muscle contacts through wrapped generalised cylinders. Since strands allow multi-resolution (strain/ stress are evaluated at user-defined discrete locations), results in terms of computational speed are very promising. However, there is no evidence that muscle anisotropy and local muscle fiber directions can be incorporated (strands actuation rely on a Hill-based model).

## 2.5.6 Discussion

Despite simple action line models, scaled according to individual measurements, have shown good correlation with experimental results (moment arms, muscle length) and have lead to valuable clinical studies, highly accurate and cost effective 3D dynamic muscle simulation is not currently possible due to computational complexity (high number of interacting organs, complex mechanical behaviour). However, computational speed is improved by making assumptions about object mechanical behaviour and simple geometric tricks can be used to cheat with regards to physical laws. Particularly, geometric constraints such as smoothing, radial springs and collision handling can visually replicate musculoskeletal behaviour, although they still need to be validated from measurements or from more accurate predictive methods (e.g. FEM). The force/ deformation relationship is better known through adequate tissue experiments, leading to sophisticated constitutive laws. However due to various assumptions commonly done (fiber direction approximation, quasi-static simulation, small number of elements), in-vivo validation still need to be performed to correlate dynamic simulations and real deformations. This has been done for the heart (biomechanics-based segmentation) [VGW06], but this is more difficult for skeletal muscles, since voluntary actuation is not known and dynamic images are more difficult to obtain (no repetitive motion, larger region, etc.). In addition, improvement in terms of individualisation need to be achieved towards automatic shape, fiber direction and mechanical parameters extraction, which is possible from MRI images (Section 2.3). The investigation of the force/ actuation relationship is limited, due to the lack of reliable experimental data. In most studies, the actuation is defined as a scalar value, which can be correlated to EMG measurement, but the spatial repartition of activated fibers is not considered. This has been done for the heart [NH01], but not for skeletal muscles.

## 2.6 Cartilage modelling and simulation

### 2.6.1 Cartilage Anatomy

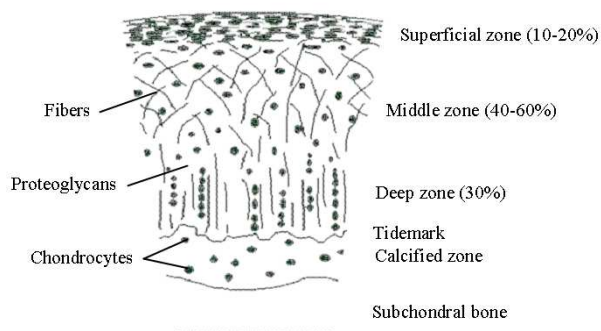


Figure 2.35: Cartilage layers

Articular cartilage (or hyaline cartilage) is a dense connective tissue aimed at reducing friction in bone contacts (synovial joints), and distributing load onto wider bone surfaces. Cartilage is biphasic: it is mainly composed of water (decreasing from 85% to 70% during life). Its solid part is made collagen fibers responsible of cartilage tensile strength, a ground matrix (proteoglycans) that bond water and cells (chondrocytes) that secret and maintain the matrix. Articular cartilage is multi-layered (anisotropic): the superficial layer (highest concentration of collagen/lowest concentration of proteoglycans) is very resistant to shear stresses, and ensure smooth articular contacts. The intermediate layer is mechanically designed to absorb shocks and distribute the load efficiently. The deepest layer is highly calcified, and anchors the articular

cartilage to the bone. Much of the foetal skeleton consists of hyaline cartilage which is subsequently replaced by bone. A mineralisation front advances through the base of the hyaline articular cartilage at a rate dependent on cartilage load and shear stress. Mineralisation can cause joint degeneration (osteoarthritis). Fibrocartilage (e.g. the labrum) is a modified articular cartilage, containing a higher proportion of collagen fibers, almost parallel.

Cartilage is biphasic and porous, that is a pressurised fluid flows out/ in during compression/ relaxation, leading to a high stress/ strain rate dependence (see Figure 2.36). This transient mechanical behaviour is influenced by an intrinsic viscoelasticity mainly due to collagen fibers, while the proteoglycans impart for elastic behaviour [SZM93] [LSAB03]. Furthermore, fluid flow and pressurisation were found to be the dominant mechanisms of load support in confined compression. However, current theories still experience difficulties in determining the roles of fluid-flow and viscoelastic properties in the mechanical behaviour of articular cartilage [Mil98]. It appears that collagen viscoelasticity plays a minor role if significant fluid flow is present [LSAB03].

### 2.6.2 Cartilage representations

It is important to assess cartilage degeneration process in order to improve osteoarthritis treatment, and significantly reduce the amount of prosthesis replacements. However, due to the complex behaviour of articular cartilages (non-linear, fiber-reinforced, bi-phasic and viscoelastic behaviour in confined condition), precise mechanical models have been difficult to develop. In [RM77], a first attempt to characterise cartilage elasticity (solid phase) has been performed with FEM. In [RGST79], the lubrication (fluid flow) phenomenon has been studied assuming nonporous cartilage. Later, the cartilage has been assumed to have a poroelastic behaviour [MKLA80] [McC82]. Current studies focus on fiber-reinforced poroviscoelastic (FRPVE) models [LSAB03]. A FEM implementation of this model, using non-homogeneous brick finite elements has been done in [Jul04]. Ferguson et al. [Fer00] have studied the labrum, imparting for the sealing function of the pressurised layer within the hip joint. Indeed, labrum damage can contribute to hip joint degeneration (no fluid layer, less uniform pressure distribution) [KDG91]. Idealised joints (e.g. ball and socket hip joint) has been mainly applied. However, more specific cartilages based on joint geometry/ function have been recently studied from cadaveric specimens and in-vivo static and dynamic data, especially for the knee [BPTY04] [Pir05]. In [Mac05], Maciel et al. use linear isotropic hip cartilage layers (particle systems) on top of MRI reconstructed bone models to assess strain/ stress distribution, for computer aided corrective surgery

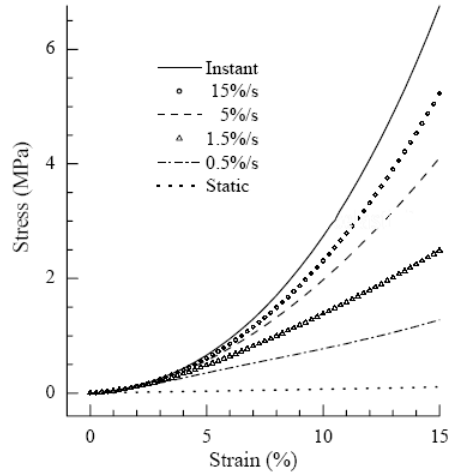


Figure 2.36: Cartilage stress/strain curve under compressive axial strain at different rate [LSAB03]

planning. Indeed, joint impingement, due to bone abnormal shape, is a possible cause of degeneration [Fer00] [KSMMT03]. Cartilage static examination (quantification of the degeneration) is mainly performed through cartilage thickness extraction from MRI and coverage analysis in the thickness map [WTGW03].

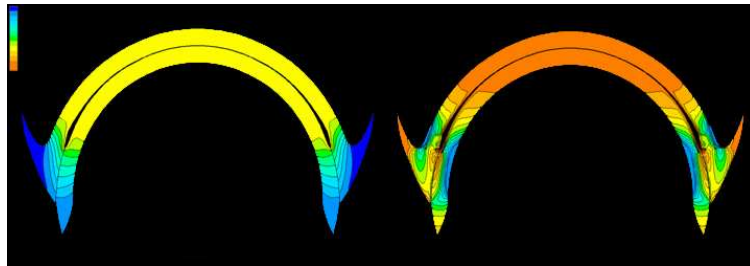


Figure 2.37: Pore pressure and principal strain distributions in the sealed hip joint, from Ferguson et al. [Fer00]

## 2.7 Ligaments modelling and simulation

### 2.7.1 Ligament anatomy

Ligaments are bands of fibrous, slightly elastic tissue that connect contiguous bones, across joints. They guide joint motion, augment joint mechanical stability, and prevent from excessive movements and subluxations. Ligaments are composed of 20% cells (fibroblasts), and 80% matrix (30% solid, mainly collagen fibers and 70% water). Ligaments and tendons have almost similar properties, although there is slightly more collagen in tendons than in ligaments. In addition, fibers interlace with a predominant direction in ligaments, whereas they are parallel in tendons. The elastin/collagen proportion can vary: in elastic ligaments (e.g. vocal cords) there is twice more elastin. However, articular ligaments are collagenous ligaments.

Ligaments are transversely isotropic, viscoelastic (due to the water) and nonlinear materials [CN89] [WG01] [HGGV02]. They can undergo large deformations and help to store elastic strain energy. However, they can



be damaged (for strain higher than about 10%) or completely disrupted when restricting abnormal extreme motions. The time-dependant (or viscoelastic, or hysteresis) behaviour of ligaments has been studied in [BLLSW05] [PLCJ02]. The conclusion is that there is a small but significant increase of ligament stiffness with regards to loading rate increase. On the contrary, like most biological soft-tissues, ligaments tend to be less stiff after several loading/unloading cycles (history dependence). Therefore, stretching exercises can prevent from ligament injuries. Due to fluid exchange, ligaments, such as cartilages, are not completely incompressible [TVG95]. Ligament mechanical behaviour can also vary with age and temperature.

## 2.7.2 Ligament representations

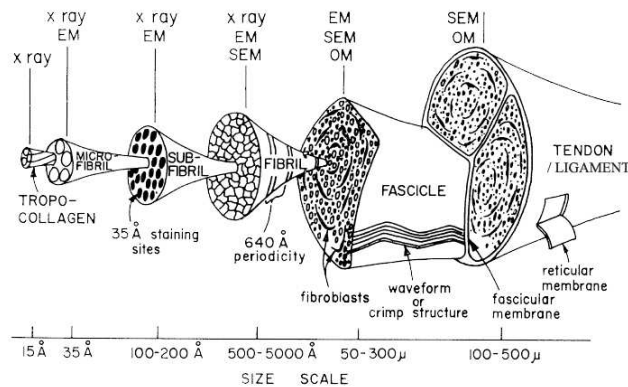


Figure 2.38: Ligament/tendon layered structure [KGB78]

of multiple lines (corresponding to fibre bundles) which model ligament inhomogeneities more accurately, one dimensional representations are not able to predict 3D stress and strain resulting from a combination of tension, shear, bending and compression [WG01]. Moreover, load transfer is not accurately assessed with discrete insertion points, and approximated ligament principal directions. To tackle this, 3D ligament models have been studied. Penrose et al. [PHBH02] present a complete knee FE model where tissue dynamics and joint kinematics are estimated through complex contact computation, difficult to validate. In [WGL<sup>+</sup>05], Weiss et al. propose a finite-element ligament model based on an incompressible, fiber-reinforced, hyperelastic constitutive model [WMG96]. They test a single ligament (manually reconstructed from CT images) in the quasi-static condition, with prescribed bone positions and validate their method using a video-based strain analysis method. Their model is composed of shells [HL81] which are more accurate than hexahedral elements for simulating the bending behaviour of thin objects. The reference state (zero stress) remains problematic to determine but can be estimated from optimisation approaches.

For interactive applications and the coarse/ fast determination of joint limits and ligament failure points, particle system models of ligaments have been developed based on simple elastic laws. In [Mac05], Maciel et al. apply a linear isotropic dynamic model to hip ligaments. Though not physically accurate, their model is able to simulate the main elastic behaviour of ligaments, through cost-effective computational methods. Particularly, they can reproduce the extensive folding that occurs in many capsular ligaments. Ng-Thow-Hing et al. [NTH00] model ligaments with B-Splines embedded in a viscoelastic network. Due to their geometric nature, to their simple constitutive behaviour, and to the lack of in-vivo dynamic data, these methods remain unvalidated, and inappropriate for acute clinical studies.

In addition to the desire of understanding ligaments functions, surgical applications (prosthesis design, ligament reconstruction) have motivated the development of finite-element models able to predict equilibrium positions under loading and kinematic constraints. In fact, knee analysis has received the highest interest due to common knee ligament pathologies (e.g. anterior cruciate ligament or ACL injuries). Similarly to muscles, ligaments have been mostly represented by one dimensional models (straight lines, multiple lines, wrapping piece-wise lines) [BH91] [BPTY04]. Indeed, the main property of ligaments is the tensile reaction along their principal axis. The line representation allows a simple mechanical description of the ligaments: one parameter for stress and one parameter for strain. Despite the use

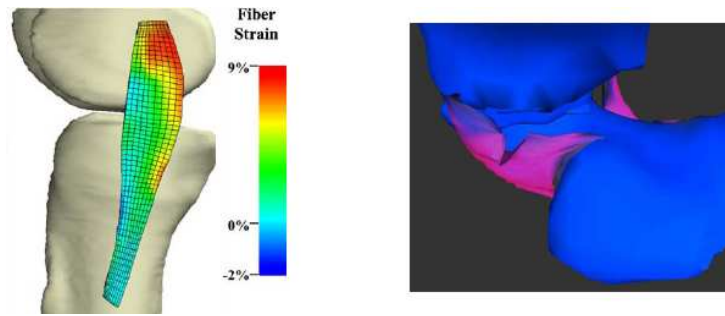


Figure 2.39: From left to right: Medial collateral ligament strain computed with a 3D anisotropic FE model [WGL<sup>+</sup>05] and particle system simulation of the ischiofemoral ligament [Mac05]

## 2.8 Fat modelling and simulation

### 2.8.1 Fat anatomy

Fatty tissues (triglycerides) aim at storing metabolic energy over extended periods of time, but also at protecting body organs against shock, at maintaining body temperature, and at promoting healthy cell function. They can be found between the skin and the fascia and between deep organs (e.g. muscles). Fat is found in various quantities across individuals, depending on age, gender, exercise, etc. Mechanically, fatty tissues are viscoelastic, non-linear, isotropic and nearly incompressible.

### 2.8.2 Fat representations

Even if fat plays an important role in body shape, there are only few studies regarding fat modelling and simulation. In body modelling, fat effect can be modelled through a surface relaxation-based geometric method such as in [SMT03]. In [SPCM97], Scheepers et al. model fatty connective tissues by adjusting the radius of influence of the underlying implicit muscle models. This can be similarly done with a voxelisation technique [WG97]. The subcutaneous fatty tissue can also be modelled as an extra skin membrane layer with possibly volume preservation such as in [LTW95] [KGG96] [TT93] [NTH00]. Researchers also investigated physical-based approaches for fat simulation. Aubel et al. [Aub02] apply the linear elastic particle system method of Debunne et al. [DDCB01] to simulate incompressible breast and subcutaneous fat tissue. A linear and incompressible FEM layer is used for simulating fat in [KRG<sup>+</sup>02] [Neb01]. Hirota et al. [HFS<sup>+</sup>01] model fat with a Mooney-Rivlin material. Finally, some studies in medical image registration, such as [SBRP99], apply finite-element models for simulating breast deformation in mammography, the goal being to find spatial correspondences with undeformed states such as in MRI/CT images.

## 2.9 Skin modelling and simulation

### 2.9.1 Skin anatomy

The skin consists of two layers: the epidermis (50 to 100  $\mu m$ ) mainly composed of keratin, and the dermis (1 to 3  $mm$ ) made of collagen fibers (72%), elastin fibers (4%) and other structures (blood vessels, nerves, hair follicles, smooth muscle, glands and lymphatic tissue) [Dan73]. The epidermis layer is responsible for keeping water in the body and keeping other harmful chemicals and pathogens out, whereas the dermis regulates temperature and supplies the epidermis with nutrient-saturated blood. Dermis collagen fibers are oriented according to Langer's lines (see Figure 2.2) and the skin tends to be less stiff in these directions. Under the dermis, the hypodermis (subcutaneous fatty tissue with variable thickness) primarily composed of cells and elastin fibers, attaches the skin to underlying bone and muscle through fascia, as well as supplies it with

blood vessels and nerves. The skin is a non-linear, anisotropic, viscoelastic and incompressible material. It is characterised by its extensibility (up to 50%), resistance to friction, response to lateral compressive loading and plasticity. Skin modelling is difficult due its great variability around the body. Constitutive equations for skin have been established for its two layers (the fatty hypodermis is often removed due to its low resistance). Uniaxial models [KGD64] and elastic membrane [Dan73] models have been first developed. Later, Lanir et al. established a fundamental biaxial model in [Lan79] [Lan87]. They consider both collagen (undulated) and elastin (linear) fibers. Shoemaker et al. [SSLF86] have applied the Lanir's model with Fung's viscoelasticity theory (QLV).

## 2.9.2 Skin representations

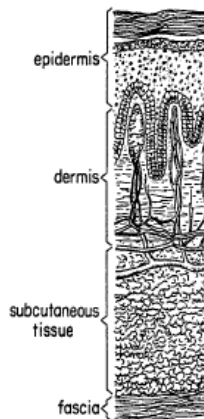


Figure 2.40: Skin layers, from [Dan73]

In addition to medical application where a precise modelling of the skin is required, representing and animating virtual human envelope is fundamental in computer graphics. Run-time skin animation is required for interactive applications, through simple geometric methods. In biomechanics, the skin is essentially modelled using 2D or 3D finite elements [Neb01] [HFS<sup>+</sup>01] [KG96], whereas in computer graphics, polygonal surfaces are used [LTW95] [WG97] [Aub02] [NTH00]. There are also examples with implicit surfaces [BW90] [Yos92] [SPCM97]. DeRose et Al. [DKT98] extend Catmull-Clark subdivision surfaces to handle creases and corners. Skin surface patches are smooth, parameterisable and straightforwardly controllable by designers through an overlying control mesh. Lee et al. [LTW95] propose a sophisticated skin models, composed of several anchored polygonal meshes representing the different skin and subskin layers (epidermis, dermis, sub-cutaneous connective tissue and fascia). Parke et al. use parametric surfaces [Par82]. Skin is a passive connective tissue which deformation relies on underlying organs. Skin deformation techniques, from geometric-based methods to anatomy-based methods, fall into the following categories: 1) lattice-based free-form deformation (FFD); 2) skeleton-driven deformation (SSD or skinning); 3) muscle-driven deformation; 4) example-based approach.

## 2.9.3 Free-form skin deformation

Free-Form Deformations (FFD) have been introduced by Sederberg and Parry [SP86] as a technique for deforming objects from a lattice in which objects are embedded. The lattice, composed of interconnected control points, is deformed by manipulating control points. A spatial correspondence is established according to lattice deformation, resulting in the wrapping of embedded objects. Original Sederberg's FFD is based on a parallelepiped lattice composed of tricubic Bezier hyperpatches leading to a straightforward computation of the mapping function. FFD have been widely used for animating virtual humans in real-time [CHP89] [MT97] [SK00]. A multi-layer body model was introduced by Chadwick et al. [CHP89] where the muscle layer is represented by a Sederberg's rectangular lattice on which biomechanical parameters such as elasticity and contractility are applied. Extensions have been proposed to allow more flexibility in the lattice representation to better match the underlying model: Moccozet et al. [MT97] have proposed Dirichlet FFD based on a Voronoi structure; Singh et al. [SK00] have presented surface-oriented FFD where deformations are directly derived from neighbouring surface elements. Another recent example of lattice-based skin deformation technique, proposed by Capell et al. [CGC<sup>+</sup>02], makes use of FEM and bone constraints to derive the deformation of an anatomy-based lattice. Given a coarse lattice ( $\sim 500$  cells), the computation can be quickly computed ( $\sim 100Hz$ ), thanks to an implicit integration scheme [BW98]. This provides physically consistent deformation (isotropic, linear material with non-linear strain computation), contrary to most geometric-based methods.



Figure 2.41: surface-oriented FFD overlaid on a virtual character [SK00]

### 2.9.4 Skeleton-driven skin deformation

Skeleton-driven deformation is very popular in virtual human animation, as it allows simple computation of skin deformation according to character pose. In literature, skeleton-driven deformation is often referred as Sub-Space Deformation (SSD) or blend skinning. The idea is to compute skin transformation as a weighted sum of joint transforms (matrix blending), weights being defined according to joint influences. In this context, Magnenat-Thalmann et al. [MTLT88] introduced the concept of joint-dependent local deformation (JLD). To prevent from usual matrix degeneration, joint rotation interpolation can be carried out using linear quaternion interpolation or spherical quaternion interpolation [KZ05], so-called Slerp [Sho85]. However, skeleton-driven deformation suffers from collapsing artefacts when blending very different transformations [LCF00]. To tackle this, and to exploit the cylindrical topology of human limbs, authors also presented skinning methods based on cross-sections or sweep surfaces, for which the orientation according to the skeleton is interpolated from joint angles [KMTM<sup>+</sup>98]. It is possible to smoothly blend sweep surfaces along the complete body hierarchy as shown in [HYC<sup>+</sup>05].

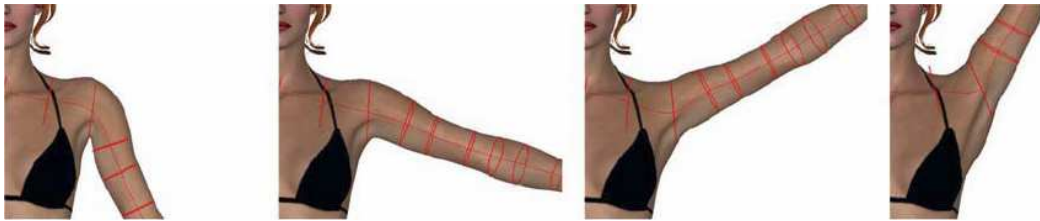


Figure 2.42: Sweep surfaces of the arm [HYC<sup>+</sup>05]

### 2.9.5 Muscle-driven skin deformation

In reality, skin deformation is driven by underlying muscle/fat. Consequently, it is possible to model and animate skin by studying muscle/fat boundaries. Thalmann et al. [TSC96] sample the skin by casting ray from the skeleton within cross-sections. Scheepers et al. [SPCM97] use the blending aspect of implicit functions defining their muscles. Wilhelms et al. [WG97] use a voxelisation technique. Another approach is to compute skin deformation using springs anchored to the underlying layer, such as in [Aub02], [NTH00], [LTW95], [WG97], [TT93] and [KHS01]. Purely muscle-driven skin deformation (only springs-dependent) may not be sufficiently realistic for particular joints or postures. The skin layer can be mechanically simulated to add elastic constraints such as in [LTW95] or [TT93]. Extra features can be added to the skin layer, such as dynamic effects and volume preservation [LTW95]. Wrinkles can be simulated by tuning spring stiffness and using texture bump maps [WKMT99] [KHYS02]. In the field of facial surgery, Larrabee et al. [Lar86] model skin as a linear elastic FEM membrane with regularly spaced nodes, connected by linear springs to

subcutaneous attachments. Roth, Koch et al. [KG96] [RGTC98] [KRG<sup>+</sup>02] improved this model to higher order FEM, and proposed a framework for maxillofacial surgery planning from medical images. Keeve et al. [KGG96] use anchored elastic membranes.

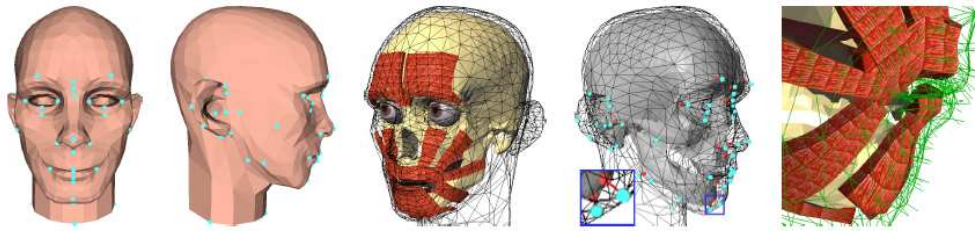


Figure 2.43: Muscle-driven skin deformation, from [KHYS02]

Instead of an elastic membrane, Hirota et al. [HFS<sup>+</sup>01] add a FEM skin/fat volumetric layer based on the Mooney-Rivlin material (see Figure 2.29). Thick 4-nodes shell elements [HL81] have been also used: Maurel et al. [MWTT98] have applied two-layered thick composite shell elements with a different constitutive relationship for each of the layers.

### 2.9.6 Example-based deformation

Example-based approaches consist in data estimation from scattered sample data (either acquired, simulated or designed). New data (for instance body poses) can be computed as an interpolated value between initial samples or between representative samples (or principal variations, or variation modes). Representative samples can be obtained through a suitable statistical analysis of the dataset. The difficulty lies in the definition of model parameters (or sample dimension, or number of degrees of freedom) that can be very large. Pentland et al. [PW89] pioneered the work on constrain deformation from samples (modal analysis). For body animation, joint transforms are generally taken as model parameters [LCF00], although high-level parameters can be used such as the emotional state of a character [KGSMT03]. Similarly, in image segmentation, a relevant set of variation modes may be used to constraint model deformation (see Section 3.3.4).

**Data Interpolation:** The Shepard’s method is a popular approach for interpolation. The interpolated value is a weighted sum of samples divided by the sum of the weights (sample weights are defined as a power of the inverse distance from the sample). Shepard’s method suffers from serious problems when interpolating at sample values (singular values) or far from samples (convergence to sample mean). Radial Basis Functions (RBF) are widely used for interpolation as they do not experience these difficulties [Pow87]. The interpolated value is defined as the weighted combination of non-linear functions (or radial basis) of the distance from samples. A popular radial basis is the Gaussian radial basis, where weights are chosen to minimise the square distance error from samples. The Bookstein’s thin-plate-spline (TPS) [Boo89] minimises the bending energy of a deformed surface, and has been extended in 3D.

**Statistical Analysis:** The most popular statistical analysis method is the Principle Component Analysis (PCA), also known as the Karhunen-Loeve transform [Jol86]. PCA aims at reducing data dimensionally by projecting it to lower dimensional space while preserving as much information as possible. Principal orthogonal variations are the eigenvectors of the covariance matrix between samples. PCA is well suited when samples are correlated (otherwise, one can use Independent Component Analysis (ICA), aiming at reducing data dimensions from Gaussianity). When dealing with high dimensional space (large number of samples  $n$  compared to sample dimension  $d$ ), computational problems arise with PCA because of the resulting covariance matrix that can be very large (size  $n \times n$ ). Singular Value Decomposition (SVD) of the

covariance matrix intends to tackle this problem, and is often used as an underlying technique for computing principal components.



Figure 2.44: Example-based skinning from [MG03]

Example-based skinning has been widely studied, since [LCF00]. In this work, Lewis et al. introduce pose-space deformation (PSD). Gaussian RBF are used to interpolate new body shapes (joint local deformations) from sculpted key-frame poses. The interpolation domain is composed of joint angles or any other user parameters. In [ACP02], Allen et al. use a similar technique but from uncorrelated range-scan data, with appropriate techniques to derive body poses. Skeleton-driven deformation has been fit to example-based approaches in [MG03] and [JT05] in order to calculate influence weights, hence reducing required data for run-time applications. Kry et al. [KJP02] propose an hardware implementation of such example-based skinning. Kurihara et al. [KM04] propose a weighted PSD approach based on CT data of the hand. In addition to the skin

animation of articulated characters, example-based approaches have been widely applied: Seo et al. [SMT03] have used Gaussian RBF to generate new models from scattered static bodies. New shapes are interpolated within principal shape (PC) variations due to anthropometric variations. In [KHYS02], facial deformation is obtained through TPS interpolation between source and target scattered landmarks. Consequently, they are able to analyse the head growing process by registering longitudinal data. Kshirsagar et al. [KGSMT03] analysed the facial deformation/emotion relation with PCA. In [GBT04], principal variations of angular motion vectors in walking are analysed with regards to speed and body size changes, and new sequences are obtained by interpolation/ extrapolation.

## 2.9.7 Summary

	Accuracy	Complexity	Assumptions/ Prior knowledge
SSD	-	--	+
FFD	+	-	-
Muscle-driven	++	+	--
Example-based	++	--	++

Table 2.2: Comparison of skin deformation algorithm

In this thesis, skin deformation is derived from images (see Section 5.10). However, as an initialisation process, SSD is applied, allowing a measurement of the accuracy and a parameterisation of this technique (see Section 5.9). The modelling process is enhanced by skin/ muscle anchors.

## 2.10 Integrating musculoskeletal tissues

Complete dynamic simulation of the musculoskeletal system (at the limb level 1.1) is very important, as studies on isolated parts of the system cannot answer most of clinical questions: how variations in surgical parameters affect moment arms, force-generating capacities, contact forces inside joints, ranges of motion, etc. Dynamic simulation is based on the analysis of the movement of rigid segments composing the virtual skeleton. Using generalised coordinates accounting for the different degrees of freedom, skeletal dynamics can be described through force equilibrium derived from solid mechanics laws. Namely, the motion of the rigid parts of the body is related to forces applied to them according to:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q})\dot{\mathbf{q}}^2 + \mathbf{G}(\mathbf{q}) + \mathbf{R}(\mathbf{q})\mathbf{F}^{\mathbf{MT}} + \mathbf{E}(\mathbf{q}, \ddot{\mathbf{q}}) = \mathbf{0} \quad (2.4)$$

Where:

- $\mathbf{q}$ ,  $\dot{\mathbf{q}}$  and  $\ddot{\mathbf{q}}$ : vectors of the generalized coordinates, velocities, and accelerations
- $\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}$ : vector of inertial forces and torques where  $\mathbf{M}(\mathbf{q})$  is the system mass matrix
- $\mathbf{C}(\mathbf{q})\dot{\mathbf{q}}^2$ : vector of centrifugal and Coriolis forces and torques
- $\mathbf{G}(\mathbf{q})$ : vector of gravitational forces and torques
- $\mathbf{R}(\mathbf{q})\mathbf{F}^{\mathbf{MT}} = \mathbf{T}^{\mathbf{MT}}$ : vector of musculotendon torques where  $\mathbf{R}(\mathbf{q})$  is the matrix of muscle moment arms and  $\mathbf{F}^{\mathbf{MT}}$  is a vector of musculotendon forces
- $\mathbf{E}(\mathbf{q}, \ddot{\mathbf{q}})$ : vector of external forces and torques. Its evaluation requires a careful modelling of interactions with the environment (contacts).

In this equation, it is interesting to determine unknowns, such as muscle forces and joint reaction torques (part of external forces) to better understand the generation of human motion but also for ergonomics applications, rehabilitation and movement optimisation. These values cannot be measured non-invasively, so they are estimated. As a first step of forces prediction, muscles and possibly connective tissues are modelled on top of a skeleton model. To reduce complexity of the pipeline (measurement, modelling and simulation), simplifications are done in terms of joint degrees of freedom, number of models and organ representation as shown in the previous sections. Given a simplified musculoskeletal model and a set of input parameters such as body motions or muscle activity patterns (Section 2.5.2), Equation 2.4 is solved to determine unknown parameters, minimising some criterions. As shown in Figure 2.45, two main approaches are considered in biomechanics:

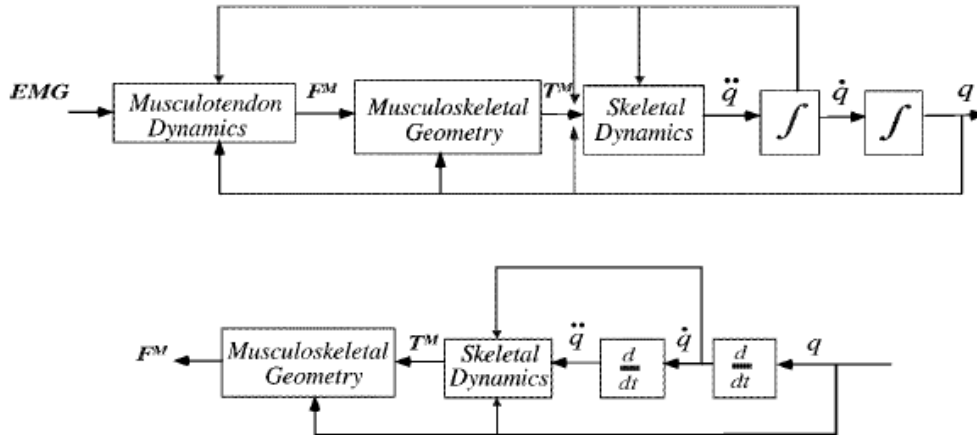
- Inverse dynamics where body motions (position, velocity, and acceleration of each segment) and external forces are used as inputs.
- Forward dynamics where muscle stimulation or muscle activations are used as inputs.

Inverse kinematics (or IK) may be applied to complete unknown body motions by prioritising position and velocity constraints. This is particularly useful in computer graphics, where virtual humans can be animated from a small set of input parameters (typically, the position/ velocity of some limb extremities) such as in [BMT97]. In [SNF05], facial muscle activations are automatically determined from motion capture data through an energy optimisation technique and an accurate biomechanical model [TSB<sup>+</sup>05]. Unfortunately, the Equation 2.4 remains undetermined, as muscles, operating in complex agonist/antagonist schemes, own more DOFs than joints. In other words, the force developed by each muscle cannot be determined uniquely. Solutions are found through optimisation strategies (least action principle):

- Static optimisation [CB81] minimises criterions based on instantaneous parameters (e.g. the sum of muscle forces, stresses, work).
- Dynamic optimisation [PZ91] iteratively minimises criterions based on parameters history over a period of interest (e.g. energy consumption, acceleration derivatives).

Static optimisation is relevant for inverse dynamics (descriptive approach), while dynamic optimisation is valid for forward dynamics (predictive approach). They produce consistent results as shown in [AP01] [TSF05]. Because dynamic optimisation is computationally costly, recent approaches such as the Computed Muscle Control (CMC) technique [TA06] have been developed to decrease computational charges using feed-forward and feedback mechanisms (the goal being to generate muscle patterns that closely match measured kinematics through forward simulation), thus combining inverse and forward dynamics. Another interesting approach is to use training techniques to allow automatic motion planning [GT95]. Recently, Lee and Terzopoulos [LT06] have proposed automatic muscle actuation pattern computation, through neural network

### Forward Dynamics



### Inverse Dynamics

Figure 2.45: Comparison of forward and inverse dynamics methods commonly used to determine muscle force (from [Pan01])

training and feedback controls (Figure 2.46). They show impressive results on autonomous head animation. However, due to computational complexity, all these methods use simplified models (muscle action lines, simple muscle and mechanical actuation models) and ideal joints. In turn, the real musculoskeletal system exhibits complex dynamic interactions between the different tissues that are not taken into account.

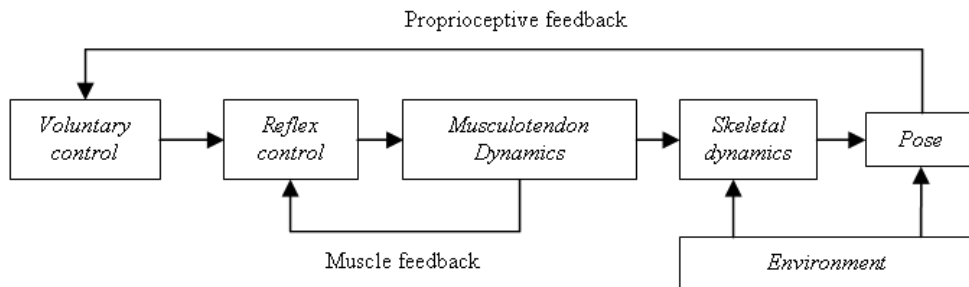


Figure 2.46: Autonomous biomechanical control through multiple feedbacks [LT06]

Static and dynamic optimisation approaches are naturally integrated in musculoskeletal frameworks. From initial frameworks [BT87] [DLH<sup>+</sup>90] to more recent ones [TSF05] [Cha03] [DL95] [RDS<sup>+</sup>03], complete parameterisation and interaction strategies have been made available. In particular the prevailing SIMM and Anybody graphics-based platforms have shown promising results in the computation of pre and post-operative muscle lengths and moment arms.

A limitation of such systems is the use of idealised joints, assuming a uniform distribution of contact forces and ignoring abnormal joint motion such as the subluxation. Moreover, as stressed in [TSB<sup>+</sup>05], [NTH00]





Figure 2.47: Lower limb simulation with the SIMM platform, from Delp et al. [DLH<sup>+</sup>90] [DL95]

and [ASAD00], usual simplified muscles (action lines) are not able to take into account large attachment areas, complex fiber arrangement, as well as global constraints such as muscle volume preservation and non-penetration constraints, which have an influence on system dynamics. Indeed, discrete one-dimensional models cannot capture three-dimensional force transfer at insertion points and contact areas. Therefore, the relationship between musculoskeletal dynamics and organ deformations need to be better studied through image segmentation, and efficient 3D deformable tissue models.

## 2.11 Conclusion and discussion

Between skin and bones, soft biological tissues are contributing to the active creation of the skeletal motion (musculotendon units) and to the passive linkage between the different organs (ligaments, cartilages, fascia and fat). We have seen that, on one side, simplified models (e.g. average muscle action lines) aim at simulating a large region of the body with approximated mechanical behaviour and contacts. On the other side, methods based on realistic FEM models (volumetric) simulate a small organ complex but are more realistic due to their explicit contact representation as well as their use of continuum mechanics laws. An intermediate solution is the use of surface models on which popular geometric methods can be applied. As stressed by Pandy et al. [Pan01], ligaments and cartilages are not modelled in most studies regarding large-scale body dynamics and body animation. Indeed, their purpose is to study muscle function and to plausibly reproduce body motion. For body dynamics study (see Section 2.10), extra torque, increasing near joint limits, may be added to simulate ligaments [Pan01]. However, cartilage tissue is not necessary as it is part of the joint idealisation (no friction). For more detailed joint studies, related to specific pathologies (osteoarthritis, ligament failure), precise tissue characterisation and 3D simulation is necessary to improve prevention, diagnosis and treatment. FEM is currently the standard approach for the application of constitutive laws, and the comparison with experimental results. However, particle systems, more flexible in terms of geometry thus closer to the model reconstruction process, start to be used for fast simulation of individualised joints. FEM is difficult to use in daily clinical diagnosis mainly due to computational expenses, whereas accurate particle systems accounting for anisotropy, viscoelasticity and hyperelasticity are not currently available. In addition, there is a lack of reliable reconstruction/ individualisation methods from medical images, leading to difficulties in the implementation of ligaments/ cartilage simulation in clinics. Indeed, a more effective analysis of the relation between joint morphology and function, coupled with an appropriate validation is required. In this thesis, we aim at bringing closer the musculoskeletal modelling and simulation part, through cost-effective registration techniques based on deformable models.

In the next chapter, we will review the main segmentation and reconstruction techniques used to obtain 3D and 4D anatomical data from medical data. We will see that, to be robust, fast and accurate, they need to be constrained using prior knowledge. In this chapter, we have seen that geometrical knowledge has been used

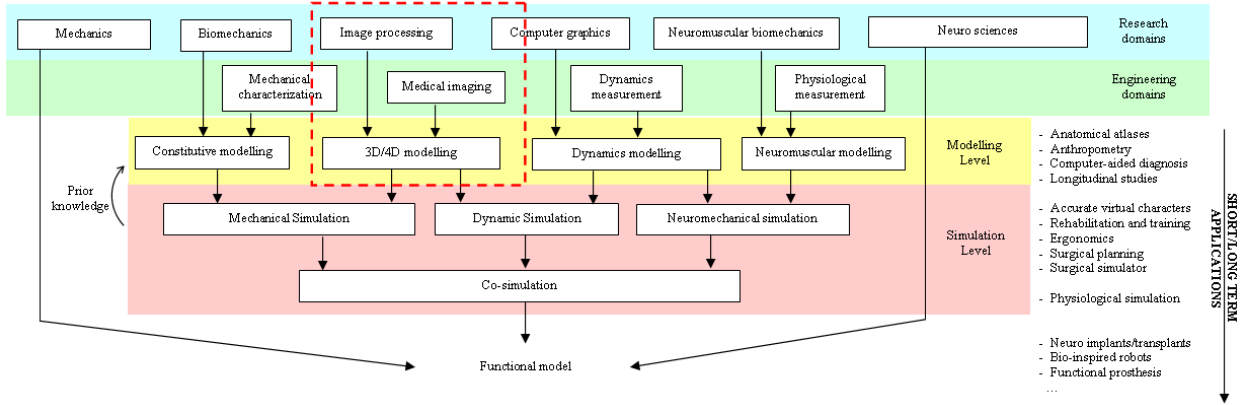


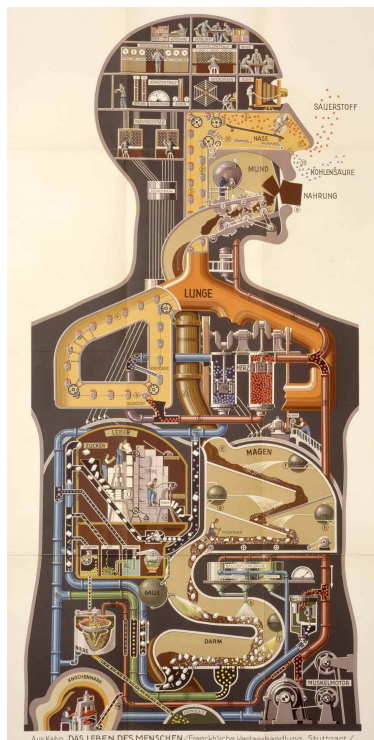
Figure 2.48: Place of modelling in the scope of musculoskeletal research

to speed up computational time of musculoskeletal simulation (medial representations, wrapping surfaces, example-based approaches, lattice deformation, multiresolution, etc.). We aim at incorporating it in the segmentation process. More precisely, researchers have investigated methods for averaging/ approximating organ geometry (shape, internal structure and attachments) and mechanical actions. Tissues are quasi-incompressible (mainly composed of water) and fairly smooth. These aspects can be simulated using simple geometric methods acting on elastic surfaces. Mechanical features due to muscles dynamics and system kinematics, such as muscle isotonic and isometric contractions, inertial effects (jiggles) or non-penetration constraints, can be also reproduced. Muscles and ligaments generally have a tubular-like shape, with an origin and an insertion on two different bones. Hence, they can be efficiently represented by underlying piece-wise action lines, where isotonic contractions are modelled through action line shrinking/ stretching, and isometric contraction through radial constraints applied to a wrapped surface. In most applications in computer graphics, the force-generating property of muscles are not modelled. Deformations obtained through joint angles changes (isotonic contractions) may be suitable for visualising believable virtual humans, whereas it is not appropriate for biomechanical simulations.

All these cost-effective techniques should be used for segmentation. Indeed the hybrid representation, combining a surface and a medial axis, appears to fit well to our needs. Unlike stick-figures that describe organs in an abstract or conceptual way, surfaces represent explicitly their boundaries (that can be correlated to acquired data). Also, visualisation of organ shape is a keep issue for many applications in computer graphics and medical informatics (diagnosis tools, 3D atlases, etc.). Due to its undemanding storage and straightforward visualisation, surfaces are certainly the most common way of representing organs. Moreover, they can well represent thin objects such as the skin layers, fascias or interosseous membranes (thickness is neglected or implicitly represented using minimum distance constraints), and stiff objects such as bones for which only boundaries are appropriate.

## Chapter 3

# Medical images analysis



<sup>1</sup>Fritz Kahn. Der Mensch als Industriepalast (Man as Industrial Palace). Stuttgart, 1926, Chromolithograph, National Library of Medicine. [http://www.nlm.nih.gov/exhibition/dreamanatomy/da\\_g\\_IV-A-01.html](http://www.nlm.nih.gov/exhibition/dreamanatomy/da_g_IV-A-01.html)

### 3.1 Introduction

Nowadays, medical imaging devices are able to produce a large amount of information, such as high-resolution volumes and temporal sequences, more-and-more difficult to analyse and visualise. In this context, higher-level information such as 3D or 4D models is increasingly required to support medical diagnosis. As shown in Chapter 2, such models also have predictive capabilities (for computer-aided treatment, prevention, rehabilitation and ergonomics). In the context of simulation, the use of medical images is important, as they allow the production of patient-specific models (anatomical models), expressing subtle inter-subject variability (possible cause of pathology), and the validation of mechanical prediction through temporal studies (kinematical models). The anatomical and kinematical modelling of the musculoskeletal system deal with image segmentation, that is the identification, in 2D, 3D or 4D images, of the region or contour of interest. Segmentation has been an intensive research field for many years, and is generally associated to registration (the search of spatial correspondences across datasets). Indeed, registration can indirectly perform segmentation as soon as a generic/ reference dataset has been segmented (the generic model is iteratively deformed/ registered to images). On the contrary, direct segmentation can be achieved with few assumptions about the problem (organ shape, image intensities, deformation, etc.). Direct approaches such as thresholding, region growing or edge detection are generally ad-hoc techniques that fail to handle noisy images with a large amount of textural information such as medical images, while indirect approaches use prior information (reference segmented datasets, generic models, anatomical and statistical information, texture database) to perform the segmentation.

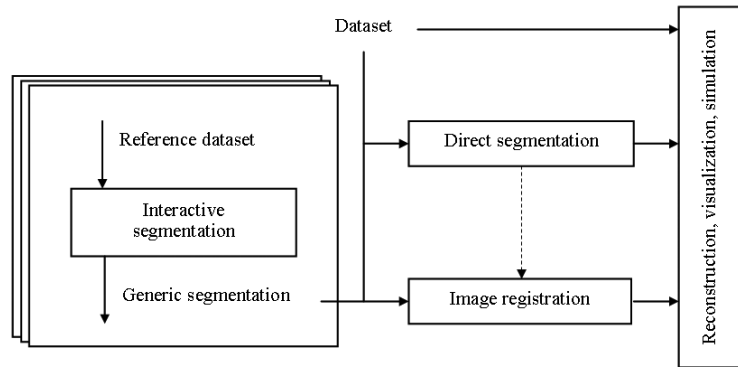


Figure 3.1: Image segmentation

Given ordinary clinical images (low-resolution images with anisotropic voxels, noise and partial volumes), the output of segmentation (binary volume), even manually obtained, is a more-or-less correct representation of organ shapes. A post-processing step is necessary to reconstruct a model, from the binary map, that can be simulated and efficiently visualised. The Marching Cubes algorithm, originally proposed by Lorensen et al. [LC87], is considered to be a standard approach to the problem of extracting iso-surfaces from a volumetric dataset. However it produces aliased meshes, with a poor geometrical quality (irregular triangles) and an excessive number of triangles. Constrained reconstruction techniques (such as [Del99]) where surface smoothness and mesh topology can be controlled are more efficient. In this context, deformable models are nowadays a widely spread approach for segmentation, since they allow geometrically constrained image segmentation (no post-processing).

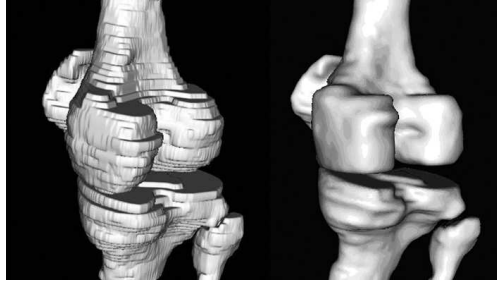


Figure 3.2: marching cubes reconstruction of the knee (left) and constrained reconstruction using deformable models (right)

## 3.2 Direct segmentation

Direct segmentation of anatomical structures in medical images is particularly challenging because medical images are noisy and generally complex from a textural point of view. Moreover, the ratio between image resolution and the size of structures to segment can be high. Low resolution images produce partial volume effects, making frontiers between organs unclear. Direct segmentation makes very little assumption about organ shape, organ size or image intensity, so is very generic. For complex problems, direct segmentation is noise-sensitive, not robust and quite inaccurate. However it can help in speeding up manual segmentation and enhancing/ extracting image features for further algorithms (e.g. registration).

Generally speaking, direct segmentation is an iterative process involving a "detection" step where regions are identified in images and a *classification* step where regions are combined to create new regions. A pre-processing step may initially filter images to normalise them, reduce artefacts and noise and enhance specific features. After segmentation, the resulting atlas can be converted into surfacic or volumetric model through reconstruction techniques.

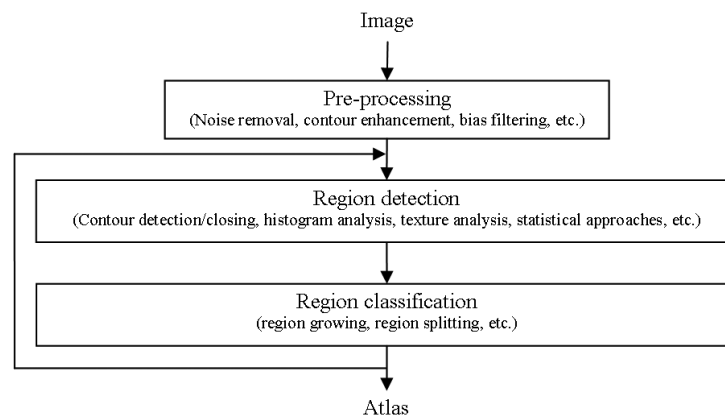


Figure 3.3: Direct segmentation process

### 3.2.1 Pre-processing

Well-known filtering algorithm may be applied to images in a pre-processing step. Noise can be attenuated through low pass filtering. Anisotropic filtering [PM90] is particularly suited, as it does not alter edge sharpness. When segmenting several datasets from different patients or sessions, intensity normalisation

may be applied to get more robust results (because most algorithms rely on image intensity). This is frequent to fit image contrast and brightness to a reference image. Normalisation is generally performed by an (piecewise) affine transform of the intensities. Between cross-sections, intensity bias can be present in medical volumes and attenuated through intensity normalisation across slices.

### 3.2.2 Region detection

Regions in images can be defined by their internal voxels or by their boundaries. Similarly, segmentation methods aim at discriminating regions, looking at image intensity levels and their derivatives. Another way is to assess the probability of voxels to belong to a region, considering local intensity values.

- **Edge-based approaches:** Organ interfaces are characterised by a more or less steep intensity level variation in images and are quantified through gradient amplitude and orientation. Various convolution filters have been proposed for gradient measurement. Among them, 1st order derivative filters such as the Sobel or Prewitt filter, and 2nd order derivative filters such as the Laplacian filter are particularly used.
- **Region-based approaches:** When regions are more or less homogeneous, voxels and their neighbours look similar, except at boundaries. Voxel intensities, standard deviations or gradients are computed and compared to the ones of neighbouring voxels (classification step) in an ascending or descending fashion (e.g. split and merge algorithm). When regions are not homogeneous, but present repetitive patterns, texture analysis may be performed. It is based on local parameters (e.g. frequency or statistics), extracted within a specific window. Alternatively, cross-correlation with prior texture primitives may be performed. An example of the use of statistical parameters is the cooccurrence method [Har79].
- **Statistical approaches:** Bayesian approaches try to maximise the knowledge of the scene (classification) given the image, through its probability density function  $p(S|I)$ . According to the Bayes rule:  $p(S|I) = p(I|S)p(S)/p(I)$ , this is an optimisation process (see Section 3.3.5) involving the likelihood (probability of the image given the scene) and the prior (probability of the scene). It leads to  $\tilde{S} = \operatorname{argmax}(\log(p(S)) + \log(p(I|S)))$  (Maximum A Posteriori or MAP solution). For expressing the likelihood, additive measurement noise (usually Gaussian) is considered as well as a signal model (such as the Gibbs distribution, considering that the signal is a Markov field). It is also common to incorporate a model of spatial interaction to account for signal continuity in the image. The Bayesian approach has been used for muscle classification [MCM<sup>+</sup>05] using a partial volume model [SSLS<sup>+</sup>01]. It is a very general framework that can be used also in deformable segmentation [IMU<sup>+</sup>00] and registration [SD92], as we will see later.

### 3.2.3 Classification

Detected regions, or voxels are classified into classes (e.g. tissue type) using homogeneity criterions. When using one parameter, thresholding is performed at a specific level. The criterion is generally image intensity level or probability density. The threshold may be automatically computed through histogram analysis (e.g. Gaussian modelling of grey level distribution). Alternatively, classification can be performed through multiple parameters (multidimensional classification), such as the region mean level, the standard deviation, the contour mean curvature or the object principal axis. Unsupervised classification tries to delineate automatically the different classes in the parameter space using proximity criterions, whereas supervised classification makes use of prior knowledge about parameter values of the different classes.

In gradient images, binary maps can be obtained by contour tracking algorithms. Some spurious edges are removed with non-maximum suppression (e.g. in the Canny edge detector). To close contours, morphological operators may be used in a post process stage. Edge detection is particularly sensitive to noise (high frequency intensity variations) and requires an adequate pre-processing (smoothing). Prior knowledge about edge orientation, amplitude and thickness is specified through filter parameters (e.g. direction, size).

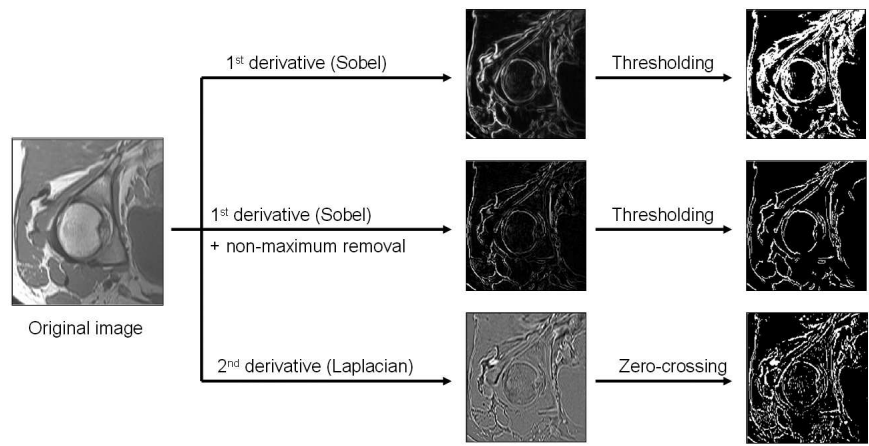


Figure 3.4: Edge detection filters applied to an MR image of the hip

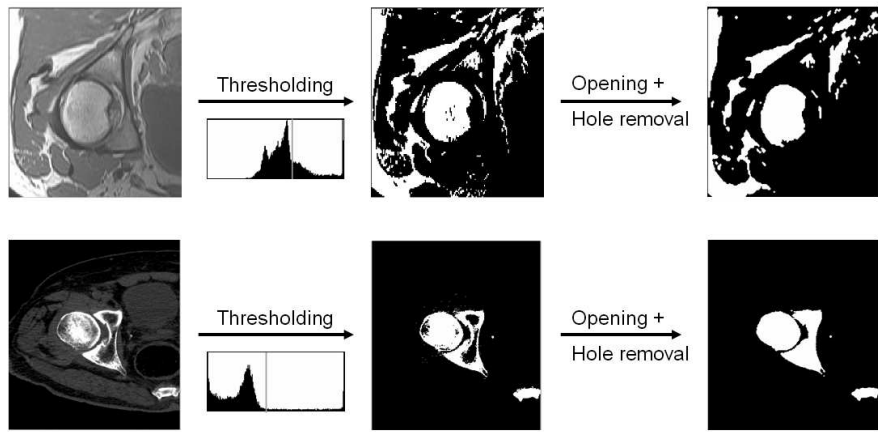


Figure 3.5: Histogram analysis for bone segmentation from MRI (top) and CT (bottom)

Two strategies are possible regarding the detection/ classification loop. The ascending strategy (split algorithm) starts from under-segmented regions and make them grow iteratively by detecting similar neighbouring regions. In the descending strategy (merge algorithm) the image is split into many regions (over-segmentation) which are subsequently merged through homogeneity/ proximity evaluation and decision making (e.g. quadtree partition). The two may be mixed (split and merge algorithm) [HP78].

### 3.3 Image Registration

#### 3.3.1 Introduction

Image registration, that is the computation of spatial correspondences between images, is one of the main fields in image processing. Indeed, registration is required in various domains such as image fusion (multimodal registration), segmentation (template registration), motion tracking (temporal registration) and navigation (viewpoint registration). The anatomical and kinematical modelling of the musculoskeletal system from MRI can be viewed as a multi-registration process (see Figure 5.1). The diversity of problems, approaches and comparisons has made any attempt to exhaustively classify registration methods difficult.

The various existing reviews [Bro92] [MV98] [FHM00] [Cac02] have focused on different more-or-less independent criterions such as:

- **The nature of features:** Features, also called primitives or landmarks, are the information, either geometric or iconic, extracted from images to be registered.
- **The similarity measure:** this is the criterion upon which the registration is performed.
- **The regularity of the problem:** the type of transformation that we are looking for (e.g. rigid, elastic, etc.)
- **The resolution method:** how the optimal transformation, maximising the similarity between features, is found.

Image registration aims at finding the displacement field that maximises the similarity between images, while preserving a certain regularity of the displacement. Hence, it can be seen as an energy optimisation problem, where the external energy represents the attraction potential field from images, and the internal energy, the regularity of the displacement. Alternatively, image registration can be viewed as a force (energy derivative) equilibrium process where external and internal forces self-balance.

### 3.3.2 Registration features

Features, also called primitives or landmarks, are the information derived from images that is registered. We commonly distinguish geometric features (shape) and icons (image), but it is possible to combine them.

#### Iconic features

Icons rely on the (pre-processed) photometric information of a voxel (e.g. optical flow [HS81]) or a region (e.g. template matching [DGS01], intensity profile [MD00]). With iconic registration, there is no pre-segmentation or geometric feature extraction. However, images may be pre-processed to remove noise and bias, and normalise intensities. Iconic registration has shown to be very efficient in multimodal registration (higher relevance of intensity distribution correlation over extracted features matching).

#### Geometric features and models

There are two main approaches when using geometric features for image registration. On one hand, they can be segmented separately in the two datasets and then, registered geometrically (using geometrical distance as the similarity measure) [AFP00]. On the other hand, they can be extracted from the source dataset, and subsequently used to segment the target one. With this approach, geometric features (the model) are rigidly or elastically matched, in an iterative process. These features are called *deformable models*. With deformable models, segmentation, registration and reconstruction are performed at the same time, leading to a possible combination between iconic (image-based) and geometric registration. Moreover, they allow spatially coherent regularisation mechanisms.

Geometric features of lowest dimension are *points*. Point registration can lead to rigid or elastic registration. Indeed, from known spatial correspondences at discrete locations, it is possible to estimate the global rigid movement or the surrounding elastic displacements using error minimisation and adequate interpolation methods (see Section 2.9.6) such as radial basis functions (e.g. TPS). We differentiate artificial (or extrinsic) and anatomical (or intrinsic) fiducial points [FHM00]. Artificial markers are fixed invasively (e.g. bone pins) or non-invasively (e.g. skin markers) to organs for inter-patient image registration, intra-patient image registration or motion tracking in image sequences. As discussed in [FHM00], their discrete locations are extracted from marker regions, using image processing methods (Figure 3.6). Alternatively, these regions of



interest can be directly registered through iconic registration (block matching). Apart from optical motion capture (MoCap, see Section 2.3.2), extrinsic markers are widely used for highly accurate intra-operative bone tracking [WZG<sup>+</sup>04] (navigation) and statistical modelling. Intrinsic landmarks rely on salient anatomical features. They can be extracted from images, but robust point extraction (automatic or manual) is often a difficult task [RFS99] [FL93]. Points can also be computed from already extracted features (e.g. surfaces) through geometric methods.

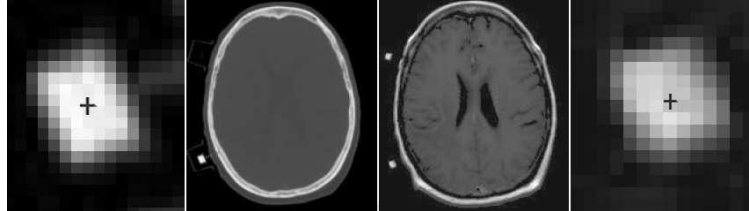


Figure 3.6: extrinsic landmark-based CT-MR registration, and extracted points from marker regions, adapted from [FHM00]

Interpolation methods are usually not realistic at locations distant from landmarks (Section 2.9.6). So that accurate point-based registration is generally limited to rigid registration (tracking applications). For rigid registration, a *coordinate system* can be used instead of a set of landmarks. This can be obtained through gravity center, principal axis and moment calculation from grey-scale or binary images [ABKC90], but also from organ shape (intrinsic anatomical axes). Such axes are standardised for bones [WSA<sup>+</sup>02]. They are mostly used for rough initialisation due to their shape sensitivity aspect. To overcome unrealistic point-based elastic registration, landmarks orientation and anisotropic error computation have been incorporated in the classical Thin-Plate-Spline (TPS) interpolation method [RFS99] (extended landmark registration). This provides better results, especially at organ interfaces where the registration transformation is not continuous.

Higher dimensional geometric features, such as curves, surfaces and volumes, have various mathematical representations. The type of representation is a key-point with deformable models, as it influences geometric descriptors, the evolution process and the visualisation methods associated to them. McInerney et al. [MT96], Montagnat et al. [MD01], Singh et al. [SGT98] present detailed reviews and classifications of deformable models. *Curves*, such as crest lines, are perceptive features that may be used for registration; however, *surfaces* remain the most used geometric primitives, since boundary is commonly the most noticeable object feature. In images, homogeneous objects, are, in fact, only characterised by their shape. We broadly present the various representations used in image registration, and to a larger extent in tissue modelling and simulation.

There are three distinct families of deformable models: *continuous models* are definite through the mapping of parameter ranges in space; *discrete models* are defined at discrete locations in space and connectivity relationships; and *implicit models* are indirectly defined by function isovalues. Continuous models offer the ability to compute differential quantities (e.g. normals, curvatures) easily and are well controllable through their limited number of parameters (intrinsic geometric regularisation) but their shape is limited by the parameterisation. On the contrary, discrete models are more flexible but difficult to constrain geometrically. Implicit models can benefit from their volumetric nature, and merging capabilities, but their abstract nature makes spatial interactions with them, difficult.

**Continuous models:** For deformable models, the continuous representation has been proposed first. In 1988, Kass et al. [KWT88] propose the well-known *snake* representation (also called active contour, or

explicit contour). A snake is an explicit mapping of a real parameter to 2D coordinates. This has been generalised to 3D case by Terzopoulos et al. [TF88] and adapted by McInerney et al. to handle topology change [MT99] (T-snake). Parametric models are continuous models where the mapping is made through particular mathematical functions, defined by few parameters (degrees of freedom of the model). In 3D, a continuous model is defined by:

$$\begin{aligned} S : [0, 1]^p &\rightarrow \mathfrak{R}^3 \\ \mathbf{u} &\rightarrow [x(\mathbf{u}), y(\mathbf{u}), z(\mathbf{u})]^T \end{aligned}$$

where  $p$  is the number of parametric coordinates ( $p = 1$  for curves,  $p = 2$  for surfaces,  $p = 3$  for volumes),  $\mathbf{u}$  the vector of material coordinates, and  $x$ ,  $y$  and  $z$  the mapping functions towards the Cartesian coordinate space.

Continuous models are popular in medical image analysis since the regularity of their shapes is simply evaluated; examining mapping function derivatives with regards to material coordinates. Moreover, parametric models are intrinsically regularised thanks to their limited number of parameters (DOFs). As shown in Section 3.3.4, regularisation ensures robustness against noise, local solutions and aperture problems. B-splines [NTH00] [Gue93], (bicubic) Hermite surfaces [FMT<sup>+</sup>03], and superquadrics [TM91] [BCA96] are the most used parametric model/ functions, since they can describe a wide range of shapes, while offering physically significant regularisation/ deformation. Shape limitation can be overcome by adding local shape perturbations [TM91], or by applying FFD subsequently to registration [BCA96]. In [LMT06], explicit and parametric representations have been united in a finite element framework.

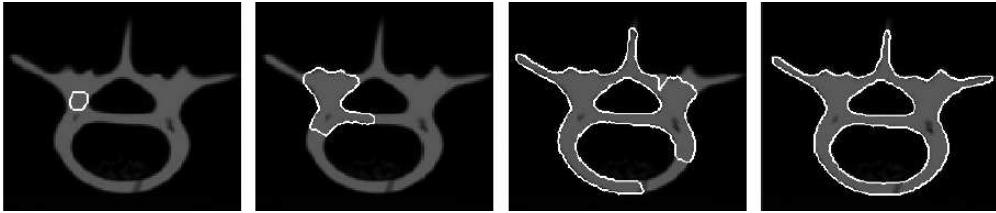


Figure 3.7: topology adaptive continuous model (T-snake) based segmentation from [MT99]

Continuous models are particularly suited for statistical analysis, because their mapping functions can be efficiently decomposed into a linear combination of variation modes [PW89]. Different basis (e.g. Fourier, harmonic, principal components) have been exploited in image segmentation [SD92] [CT01] [SKBG95] for constraining model evolution into relevant deformations, that can be learnt from examples. Based on PCA, Cootes et al. [CT01] use a statistical model of grey-level appearance in addition to a statistical model of shape variation.

Continuous models usually represent curves and surfaces. However, they admit some volumetric representations: superquadrics have an implicit formulation (e.g. ellipsoid) and most parametric surfaces can be extended to volumes ( $p = 3$ ). B-splines have been extended to B-splines solids, allowing interesting volumetric constraints [NTH00]. Continuous models need to be discretised for evolving (Section 3.3.5). With parametric models, control points (part of the parameters) may be directly used.

The main problem with this kind of models remains the inverse problem (defining shape functions and parameters from a set of locations), which is required for reconstructing/ initialising models and interacting with them. This is carried out through energy functional minimisation (Section 3.3.5), as a geometric registration process [FMT<sup>+</sup>03] [NTH00].

**Discrete models:** A discrete mesh is represented by vertices and connectivity relationships. Connected points form faces and cells. In discrete contours (respectively triangle meshes, tetrahedral meshes and rectangular grids), each cell contain exactly 2 (resp. 3, 4 and 8) vertices. They are characterised by a constant cell connectivity. Dually, simplex meshes [Del94a] [Del99] are characterised by a constant point connectivity (in  $p$ -simplex meshes, each vertex is connected to exactly  $p + 1$  vertices). Discrete surfaces are very popular in the field of computer graphics, as they are flexible and memory cheap. Moreover, dedicated hardware has led to very efficient visualisation pipelines. [Aub02] and [PN98] are examples of the use of triangle meshes for modelling the musculoskeletal system. In image registration and segmentation, deformable discrete surfaces have been widely used, since they are visually correlated to the anatomical structures. Among the numerous methods, [LM99] [LM01] [PMTK01] use triangle meshes, while [FWG<sup>+</sup>99] use tetrahedral meshes, [CPCZ03] use hexahedral elements, [MD00] [MD05] use 2-simplex meshes. Hamarneh et al. [HAGM04] propose discrete deformable organisms with radial springs. Abstract volumetric deformation grid have shown to be an efficient support for image deformation (FFD) and hence registration [BCA98] [HRS<sup>+</sup>99]. In [LRMK99] Lötjönen et al. present a coarse-to-fine approach with multi-resolution grids. Szeliski et al. [SL96] propose an octree grid, to ensure better fitting with anatomical structure. However, FFD-based registration methods, often fail to handle displacement discontinuities at boundaries. There are also examples of the use of mass-spring network [NA96] for segmentation.

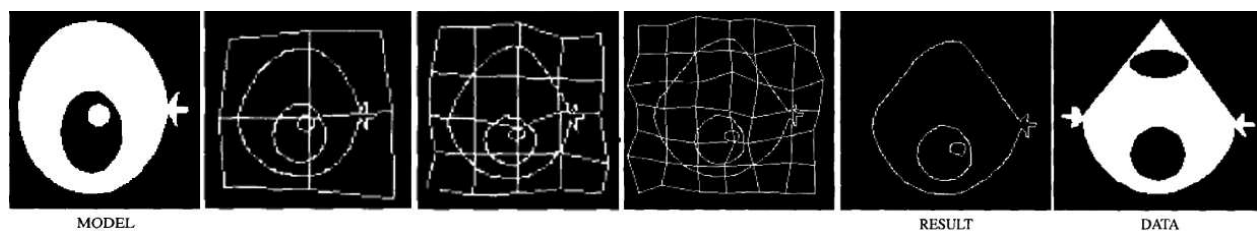


Figure 3.8: Multi-resolution FFD-based registration from Lötjönen et al. [LRMK99]

Particle systems<sup>1</sup> are discrete models with non-constant connectivity between particles. Such as real atomic or electromagnetic forces, the connectivity is defined through forces, dependant on distance between particles. Particle systems are topology adaptive and suited for tissue simulation and object reconstruction [SET02] [MHTG05]. However they remain difficult to control from a macroscopic point of view, because geometric properties are difficult to compute. Müller et al. [MHTG05] compute deformations from the explicit rigid registration with a reference particle state 3.3.5. There is no known work on segmentation and registration using particle systems.

**Implicit representation:** Models are defined implicitly by the zero of a potential field  $F$  taking spatial coordinates as parameters. In 3D, we have:

$$F : \mathbb{R}^3 \rightarrow \mathbb{R}$$

$$S = \{\mathbf{p} \in \mathbb{R}^3 | F(\mathbf{p}) = 0\}$$

As presented in Chapter 2, implicit models are widely used in computer graphics [CGD97] and body modelling [SPCM97]. Particularly, complex shapes can be obtained by merging simple implicit primitives, defined by polynomial function, such as superquadrics or hyperquadrics. Ellipsoids are part of superquadrics. Superquadrics also have a parametric formulation and have been widely used in image segmentation such as in [BCA96]. Level-set [OS88] [MSV95] is the most used implicit model representation in medical image

<sup>1</sup>This is the particle system representation, in opposition to particle system simulation as in 2.2.3 and 3.3.5

analysis. It has become popular since they allow a great flexibility in terms of shape and topology, through higher dimensional parameters. However, as emphasised by Montagnat et al. [MD01], level-sets are computationally heavy. In general, implicit models are difficult to render graphically and to constrain spatially. This is especially difficult to interact with them and to register them. In [VYCL03], Vemuri et al. use a level-set framework to derive image deformation for the registration problem.

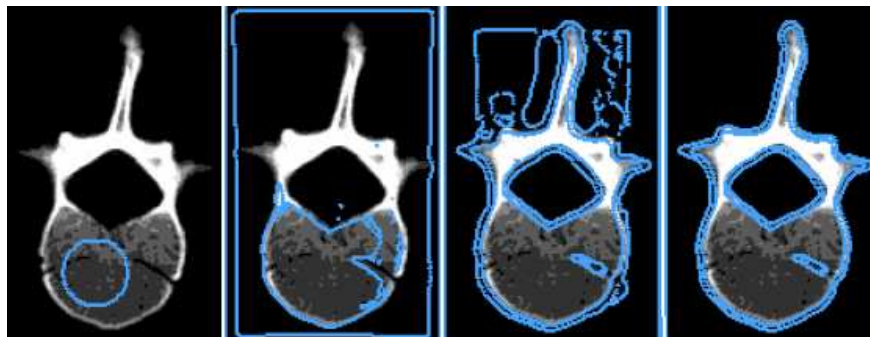


Figure 3.9: Level-set based segmentation from Montagnat et al.

Table 3.1 summarises geometric features, used in medical image analysis and in this thesis:

Feature	Representations	Examples	Examples in the thesis
Points	Discrete locations	Anatomical landmarks	Bone landmarks, Section 5.6
		Artificial markers: bone pins, skin markers	Skin optical markers, Section 5.10
Basis	Coordinate system	Principal axes and moment (image-based), Anthropometric axes (anatomy-based)	Bone anatomical coordinates, Section 5.6
	Extended landmarks		
Deformable Curves	Explicit curves	2D snakes	
	Parametric curves	Splines	Cardinal spline registration (soft-tissues attachment on bones), Section 5.4.2
	Discrete curves	1-simplex models	
Deformable surfaces	Explicit surfaces	3D snakes	
	Parametric surfaces	Splines, superquadrics	
	Discrete surfaces	triangle meshes, 2-simplex meshes, mass-spring surfaces	2-simplex models for organ registration, Section 5.5.3
Deformable volumes	Implicit models	Level-sets, superquadrics	
	Parametric volumes	Splines, superquadrics	
	Discrete volumes	Mass-spring networks, FEM networks, 3-simplex models	2-simplex models + medial axis for organ registration, Section 5.5.3

Table 3.1: Geometric features

### 3.3.3 Similarity measure: external forces/ energy

The similarity (or external energy) between images (or between geometric features) is a measure of the matching quality. It is an important parameter upon which the searched transformation is computed.

In geometric registration, the similarity is derived from the distance between features: for instance, the Euclidian distance or the p-order Minkowski distance between two points, the Hausdorff distance or Mahanalobis distance between polygons, etc. In [PCS<sup>+</sup>89], heads are registered using point-to-surface Euclidian distance. In the well known Iterative Closest Point (ICP) algorithm [BM92], points are first paired, minimising their mutual distance; then, the algorithm tries to minimise this distance over all points through a rigid transformation (see Section 3.3.5). Global external forces and moments can be defined through the examination of energy variation according to the transformation parameters (standard approach). Local external forces correspond to feature pairing (pair and smooth) approaches. A review of geometric registration techniques in medical image processing can be found in [AFP00].

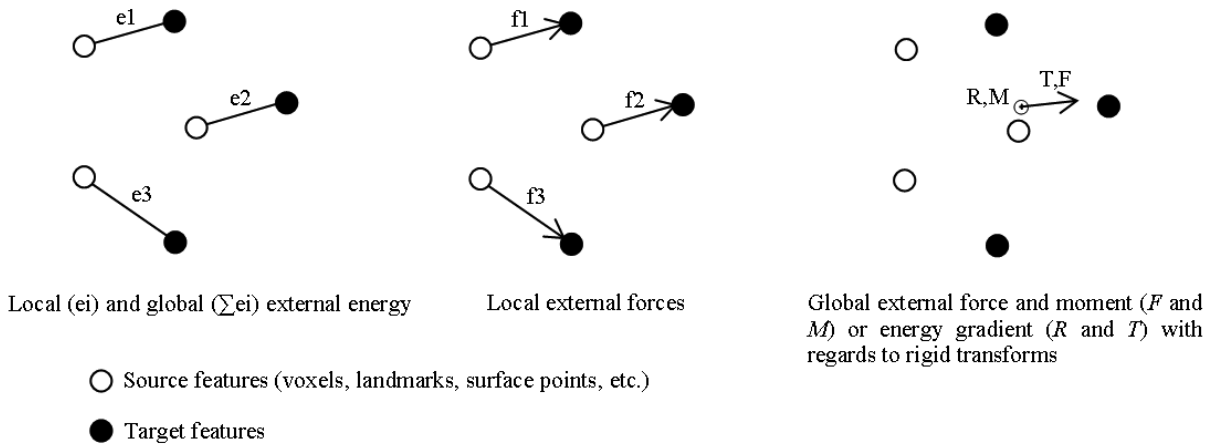


Figure 3.10: Registration external energy and forces

A common methodology, when using deformable surfaces or curves for segmentation, is to derive the external energy from image gradient values [KWT88]. In this case, models converge to locally maximal gradients, but may be attracted by undesired noisy edges. This can be partly corrected by regularising the gradient vector field such as in [XP98] (Gradient Vector Flow - GVF). Nevertheless, it is more robust to use prior information about image intensity and gradient through intensity (iconic) registration. In iconic registration, the similarity relies on the correlation between the intensities of aligned voxels or regions. Depending on the assumptions about it [RMA00], many similarity measures have been proposed [HHD<sup>+</sup>00] [FHM00] [MV98] [BNC96]. Intensity conservation along displacement, which can be valid in motion analysis, is assumed when using the sum of absolute differences (SAD), the sum of square differences (SSD), or the cross-correlation (CC) measures. Between these measures, CC is the least noise-sensitive but the heaviest in terms of computation. In mono-modal registration, the hypothesis of affine relationship between intensities from the two registered datasets improves robustness with regards to global intensity change across acquisition protocols and sessions. In this case, the cross-correlation normalised by the standard deviation (NCC) measure [HHD<sup>+</sup>00] [RMA00] eliminates the affine terms. In multimodal registration, assuming that intensities values can be mapped between datasets through a certain function (functional relationship), measures have been proposed, such as the correlation ratio (CR) [RMPA98] or the variance of intensity ratio (VIR) measure [WCM92]. Finally, for multimodal registration, information theory has lead to popular and flexible measures, based on the joint entropy. Minimising the joint entropy (or joint histogram spreading) is equivalent

to optimise intensity value prediction from one image to another, without a-priori knowledge of the functional relationship. The (normalised) mutual information (NMI, MI) results from the normalisation of the joint entropy [CMD<sup>+</sup>95] [Vio95] [WVA<sup>+</sup>96] [MCV<sup>+</sup>97]. MI is robust with regards to change in the size of the overlapping region. In [RMA00], Roche et al. unify the MI and CR measures within a maximum likelihood (ML) framework. The well known Displaced Frame Difference (DFD or optical flow) such as in [HS81] (1st order development of the DFD) incorporates pairing (assuming infinitesimal displacement) and similarity measurement (assuming intensity conservation such as SAD or SSD) through the displacement  $\mathbf{u} = (T - S)\nabla\mathbf{T}$  ( $T$ : target image;  $S$ : source image;  $\nabla\mathbf{T}$ : gradient of  $T$ ). The demon formulation [Thi95] is a modified DFD allowing large displacement through an iterative algorithm.

Prior to similarity computation, intensity filtering can be performed to decrease noise (e.g. anisotropic smoothing) or enhance particular features. Particularly, some authors have used image gradient magnitudes [MD00] or vectors [GPMTV04] instead of basic intensities.

Measure type	Measure	Assumptions	Limitations
Geometric	Distance	Rely on feature pre-detection and pairing	
Probabilistic	NMI	Correlation btw. intensity values	
	MI	Correlation btw. intensity values	
	CR	Functional correlation btw intensity values	
Intensity Correlation	VIR	Affine correlation btw intensity values	One modality
	NCC	Affine correlation btw intensity values	One modality
	CC	Intensity conservation	One modality, one protocol
	SSD	Intensity conservation	One modality, one protocol
	SAD	Intensity conservation	One modality, one protocol
Optical flow	Demon	Intensity conservation	One modality, one protocol
	DFD	Intensity conservation	One modality, one protocol, small displacements
Intensity	Gradient	Invariant and locally unique gradient along boundaries	Clean images, small displacements

Table 3.2: Similarity measures

### 3.3.4 Regularisation: parameterisation and internal forces/ energy

Due to noise, local solutions and the aperture problem (apparent motion), the registration process need to be constrained through degrees of freedom minimisation and geometric regularisation. Making hypothesis about the regularity of the problem and the form of the solution (initial distance between the two datasets) is a key-point in a registration process, for at least two reasons: 1) it determines the solution search space (dimension, order of magnitude) and consequently rules most of the computational charge (parameterisation of the resolution); 2) it decreases the uneven effect of external forces and energy through intrinsic constraints. Assumptions are generally application-dependent: for instance, rigid transformations are involved in intra-patient multimodal fusion, bone tracking or viewpoint registration problems.

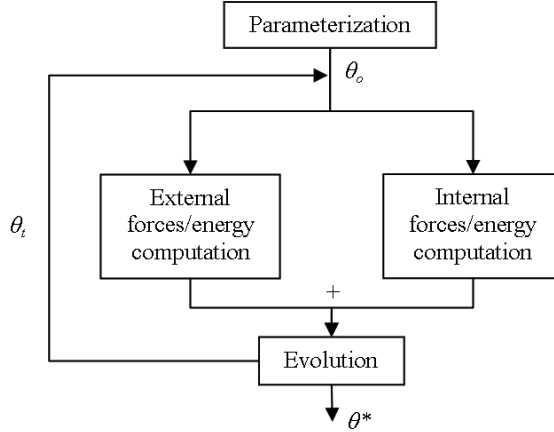


Figure 3.11: Regularisation in the registration process is twofold: parameterisation and internal forces/energy.  $\Theta_0$ : initial conditions;  $\Theta_t$ : temporary solution;  $\Theta^*$ : solution

### Parameterisation

Parameterisation of the resolution aims at limiting solution search to relevant degrees of freedom. Indeed most of the difference between datasets can be recovered from a limited number of parameters. The researched transformations can be global or local, whether which subset of images is concerned. Coarse-to-fine approaches increase the number of DOF and the localness of the transformation during the registration process, avoiding fall into local minima of the energy. There are basically two ways or parameterising a registration process, as pointed out by Cachier at al. [Cac02]. The standard approach maximises the similarity directly in the space of admitted transformations (global parameterisation), while the Pair and Smooth (P&S) approach locally maximises the similarity (pairing) according to a local parameterisation, and subsequently smooth the result (through standard smoothing of the displacement or explicit resolution 3.3.5). Table 3.3 summarises the main transformations found in image registration:

In the standard approach, global parameterisation can be performed using homogeneous transforms. Indeed, standard 3D geometric transformations can be formulated with a  $4 \times 4$  homogeneous matrix, where values may be dependant. These matrices are computed from the minimal set of transformation parameters, through successive basic transformations (e.g. rotation around one axis, shear in one direction, etc.). For instance, changes in the three hip standard angles (centred rigid transform) are equivalent to a femur homogeneous transform matrix (see Section 5.6).

Geometrically continuous elastic deformations can be derived (interpolated) from the deformations of a lattice (structured control point set of dimension  $N_x \times N_y \times N_z$ ), such as in FFD [SP86] (see Section 2.9.3). Lattice deformation is defined through the displacement of a limited number of control points ( $\delta \mathbf{p}_{i,j,k}$ ). The interpolant function  $f_{i,j,k}$  is a triple tensor product of 1D (recursively defined) weighting functions. The use of cubic splines is particularly popular since they allow various continuity constraints: Hermite, Bezier and Catmull-Rom enforce C1 continuity, while B-Splines are C2 continuous. Hence, B-splines are widely used in image registration [RSH<sup>+</sup>99] [SCW01]. The use of a regular lattice can be limiting for handling displacement discontinuities at organ boundaries. Indeed, a careful matching between model geometry and anatomy provides more realistic deformations. From unstructured data, parameterisation is still possible: as extensions of B-splines, octree B-splines [SL96] allow the use of more flexible lattice in terms of mesh topology; and triangular B-splines [DS92] [WHQ05] have shown good results for handling displacement discontinuities through unstructured triangular/ tetrahedral control mesh.

Transform	DOF	General form	Limitations
Centred rigid	3	$\delta\mathbf{p} = \begin{bmatrix} \mathbf{R} - \mathbf{I} & \mathbf{0} \\ \mathbf{0}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix}$	One patient, bones
Rigid	6	$\delta\mathbf{p} = \begin{bmatrix} \mathbf{R} - \mathbf{I} & \mathbf{t} \\ \mathbf{0}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix}$	One patient, bones
Similarity	7	$\delta\mathbf{p} = \begin{bmatrix} s\mathbf{R} - \mathbf{I} & \mathbf{t} \\ \mathbf{0}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix}$	
Affine	12	$\delta\mathbf{p} = \begin{bmatrix} \mathbf{A} - \mathbf{I} & \mathbf{t} \\ \mathbf{0}^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix}$	
Projective	15	$\delta\mathbf{p} = \begin{bmatrix} \mathbf{A} - \mathbf{I} & \mathbf{t} \\ \mathbf{x}^T & v - 1 \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix}$	
FFD (e.g cubic splines)	$3 \times N_x \times N_y \times N_z$	$\delta\mathbf{p} = \sum_{i,j,k}^{N_x, N_y, N_z} f_{i,j,k}(\mathbf{p}) \delta\mathbf{p}_{i,j,k}$	
Local (e.g. RBF)	$3 \times N$	$\delta\mathbf{p} = \sum_i^N \mathbf{w}_i(\delta\mathbf{p}_i) \phi(\ \mathbf{p} - \mathbf{p}_i\ ) + \mathbf{f}(\mathbf{p})$	
Constrained pairing	User-defined	-	
Example-based	Sample size $N$	$\delta\mathbf{p} = \sum_i^N \delta w_i \mathbf{p}_i$	Samples need to be representative

Table 3.3: Different transformations used for parameterising image registration

	Centred rigid	Rigid	Similarity	Affine	Projective
<b>Transformation</b>					
Rotation	x	x	x	x	x
Translation		x	x	x	x
Uniform scaling			x	x	x
Non-uniform scaling				x	x
Shear				x	x
Perspective projection					x
Composition of projections					x
<b>Invariants</b>					
Length	x	x			
Angle	x	x	x		
Ratio of length	x	x	x		
Parallelism	x	x	x	x	
Incidence	x	x	x	x	x
Cross ratio	x	x	x	x	x

Table 3.4: Properties of homogenous transforms (from W. Lee on-line course)

Besides, deformation from unstructured control points can be performed through radial basis function - RBF (see Section 2.9.6). The interpolated displacement is defined as the weighted combination of non-linear functions (or radial basis  $\phi(\cdot)$ ) of the distance from control points ( $\mathbf{p}_i$ ) plus an affine or polynomial term  $\mathbf{f}(\mathbf{p})$ . Weights  $\mathbf{w}_i$  are pre-computed according to control point displacements  $\delta\mathbf{p}_i$  such as [LCF00]:  $\mathbf{W} = (\Phi^T \Phi)^{-1} \Phi^T \delta\mathbf{P}$ , where  $\Phi_{i,j} = \phi(\|\mathbf{p}_i - \mathbf{p}_j\|)$ . The TPS radial basis [RSS+96] is popular for interpo-



lating displacement since it minimises the bending energy; moreover it can handle different local continuity constraints [RFS99] and is well adapted when the number of control points (landmarks) is small (however, it turns to heavy computation when it grows). The use of RBF is difficult in a variational framework, because, weights need to be recomputed at each control point displacement. In general, with RBF, scattered spatial correspondences are manually selected, or semi-automatically segmented; and not iteratively found in an optimisation framework. However, in [RAD03] Rohde et al. use a compactly supported radial basis allowing weights to be optimised locally and independently for each group of 8 points.

In classical P&S registration approaches, feature local pairing is constrained through local parameterisation. Because image registration highly relies on object boundaries registration, it is particularly suitable to constrain deformation in the normal direction of these boundaries. [Del99] [MD00] [LM01] constrain external forces along deformable surface normals and within a certain range (similarity is maximised in this search space). Based on the weighting of the displacement amplitude of control points within deformation spheres, Lötjönen et al. [LM01] locally interpolate displacement (Figure 3.12). However, this may be unrealistic because displacement direction is constant inside spheres. In [HS81] [Thi95] [Cac02], pairing is done in the direction of image gradient, with a certain magnitude (optical flow). With the DFD [HS81], the displacement is computed as  $\mathbf{u} = (T - S)\nabla\mathbf{T}$ . While, with the demon algorithm [Thi95] [Cac02] we obtain  $\mathbf{u} = (T - SoM)\nabla\mathbf{T}/(\|\nabla\mathbf{T}\|^2 + \alpha^2(T - SoM)^2)$  ( $SoM$  is the current transformed source image within the iterative process), leading to a normalisation of the displacement (its amplitude remains below the constant  $1/(2\alpha)$ ). Ghanei et al. [GSZW98] use both surface normal and image gradient directions.

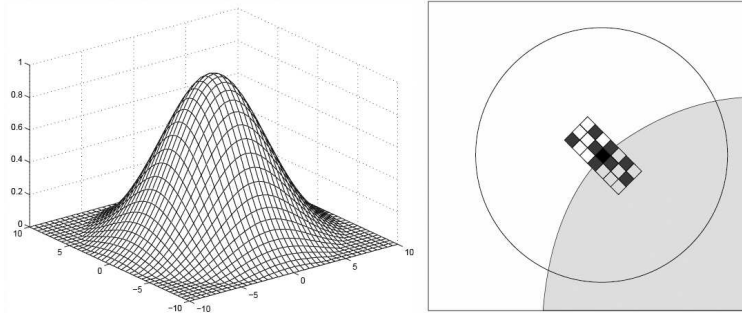


Figure 3.12: Local parameterisation from [LM01]: A 2D weight function for the deformation and 2D search space visualised in the center of the sphere

The last class of parameterisation techniques is the one based on statistics. That is new shapes are obtained by a linear combination of  $N$  sample shapes. Consequently, deformations are limited in a space of  $N$  dimensions. Relevant samples are generally obtained by PCA (see Section 2.9.6). The example-based approach is quite popular in image segmentation and registration (brain) since it takes into account shape and appearance (image intensity) variability within the population [CT01] (Active Shape Model - ASM and Active Appearance Model - AAM) [NA96] [WS00] [DSMK01] [SKBG95] [SD00] [CKP00].

During the registration process, a widespread approach is to increase the number of DOF and the localness of the transformation in order to improve robustness and computational speed; this is the coarse-to-fine approach. In [MD00] and [RSH<sup>+</sup>99], geometric transforms, from rigid to affine, are successively applied. In [LM01], deformation spheres become smaller and smaller. Multi-resolution lattices are also used such as in [SL96] [SCW01] [LRMK99] [RAD03]. To decrease computational weight, Rohde et al. [RAD03] optimise regions of interest independently through RBF-based deformation. Shen and Davatzikos [SD00] propose a hierarchical model to allow a scalable control of it. McNerney, Park and Lachaud [MT95] [PMTK01] [LM99] locally adapt their deformable surface during segmentation, and progressively increase image resolution. Ho

and Gerig [HG04] propose to use of multi-scale reference intensity profiles based on anisotropic smoothing.

### Internal forces/ energy

Intrinsic regularisation aims at ensuring a certain continuity of shapes and displacements by penalising other configurations during optimisation. Because the image registration problem is ill-conditioned (no stability) due to noisy external terms, extra regularisation terms (stabilisers) are needed within the resolution process. Either forces or regularisation energy terms can be applied depending on the chosen evolution process (see Section 3.3.5). Basically, they are of three types: smoothing, physically-based and example-based.

Smoothing energy or forces enforce shape continuity through curvature, elastic energy and/or bending energy minimisation. It relies on the assumption that biological tissues are relatively smooth. Most of regularisation terms, used in the literature, are based on 2nd order Tikhonov differential stabilisers [TPBF87] [TWK87] [MT95], since they are a generic expression of shape deformation through differential terms up to the second order. Given continuous models, material parameters and weights, p-order Tikhonov stabilisers are defined by:

- **Curves:**  $E_{reg} = \int \sum_{1 \leq i \leq p} w_i(u) \left\| \frac{\partial^i \mathbf{C}(u)}{\partial u^i} \right\|^2 du$
- **Surfaces:**  $E_{reg} = \int \sum_{1 \leq i+j \leq p} \frac{(i+j)!}{i!j!} w_{ij}(u, v) \left\| \frac{\partial^{i+j} \mathbf{S}(u, v)}{\partial u^i \partial v^j} \right\|^2 dudv$
- **Volumes:**  $E_{reg} = \int \sum_{1 \leq i+j+k \leq p} \frac{(i+j+k)!}{i!j!k!} w_{ijk}(u, v, w) \left\| \frac{\partial^{i+j+k} \mathbf{V}(u, v, w)}{\partial u^i \partial v^j \partial w^k} \right\|^2 dudvdw$

1<sup>st</sup> order terms are related to the elasticity of the model, while 2<sup>nd</sup> terms deal with bending. Forces derived from the minimisation of the energy can be obtained from the energy using the Euler-Lagrange theorem. Elastic forces minimise the curvature (Laplacian smoothing), while bending forces average the curvature. Table 3.5 summarises the different type of shape regularisation energy/ forces for parametric models [MD01] (examples given for curves):

Type	Energy	Force
Elastic	$\int \left\  \frac{\partial \mathbf{C}}{\partial u} \right\ ^2 du$	$\frac{\partial^2 \mathbf{C}}{\partial u^2}$
Bending	$\int \left\  \frac{\partial^2 \mathbf{C}}{\partial u^2} \right\ ^2 du$	$-\frac{\partial^4 \mathbf{C}}{\partial u^4}$

Table 3.5: Shape energy/ forces of parametric deformable curves

When using explicit or parametric models, the differential terms can be easily computed due to their continuous nature. This is more complicated for discrete models where geometric approximations need to be made. Standard Laplacian smoothing forces (attraction of vertices towards the barycenter of their neighbours) minimise the elastic energy but produce model shrinking. This well known problem can be tackled by compensating them with surfacic balloon forces [Coh91] or removing their normal contributions [MD01]. Their tangential contributions prevent from surface stretching. Bending forces rely on the averaging, in the normal direction, of the discrete curvature. There are several ways for estimating the curvature: Park et al. [PMTK01] average the normal contribution of Laplacian forces; Montagnat et al. [MD01] smooth the simplex angle (see Section 4.6.1); Ghanei et al. [GSZW98] smooth the Gaussian curvature (see Section 4.5.3).

Similarly to shape, deformation can be regularised [TPBF87] [LM01], by enforcing continuous variation of the deformation across space. For instance, in a deformable volume where  $u$ ,  $v$  and  $w$  are the material parameters and  $\mathbf{U}(u, v, w)$  the deformation, by minimising:

$$E_{reg} = \int w_{200} \left\| \frac{\partial^2 \mathbf{U}}{\partial u^2} \right\|^2 + w_{020} \left\| \frac{\partial^2 \mathbf{U}}{\partial v^2} \right\|^2 + w_{002} \left\| \frac{\partial^2 \mathbf{U}}{\partial w^2} \right\|^2 + 2w_{110} \left\| \frac{\partial^2 \mathbf{U}}{\partial u \partial v} \right\|^2 + 2w_{011} \left\| \frac{\partial^2 \mathbf{U}}{\partial v \partial w} \right\|^2 + 2w_{101} \left\| \frac{\partial^2 \mathbf{U}}{\partial w \partial u} \right\|^2 dudvdw$$

where  $w_{ij}$  are the rigidity weights.

In [MD05] Montagnat et al. apply weighted global forces to 2-simplex meshes based on the closest homogeneous transform. They also use the same constraints on discrete curves (1-simplex meshes) representing vertex trajectories in time (4D regularisation). In [LM01], Lötjönen et al. minimise an energy based on the excursion of model normals (dot product of deformed and original normals). It forces deformations to remain orthogonal to the surface, thus reducing the aperture problem and mesh over stretching.

Radial regularisation of surfaces is useful for constraining models possessing a medial representation. Indeed, this provides surfaces with a volumetric aspect and higher level control parameters through simple mechanisms such as radial springs [HAGM04] [FVMO04] [PFJ<sup>+</sup>03] [TWK87]. Radial regularisation will be detailed in Section 4.6.4.

Smoothing forces, minimising the bending and/or the elastic energy can be considered as weak prior information [MD05] because biological tissue generally satisfy this condition. On the contrary, example-based regularisation provides strong prior information as it may be specific to a particular problem, organ, individual, instant, etc. Montagnat et al. [MD05] add shape and trajectory constraints through forces attracting shapes towards a reference configuration. In contrast, example-based parameterisation (Section 3.3.4) combines sample deformations to compute model deformation.

Deformable models, fitting to organ boundaries, implicitly handle deformation discontinuities at boundaries. However, this is not the case with FFD-based registration, where boundaries are embedded into the deformable volumetric lattice, so that Tikhonov stabilisers would not suit (isotropic regularisation). Different anisotropic stabilisers, reviewed in [DKA95], have been developed for handling discontinuities. They basically provide anisotropic smoothing perpendicularly to image gradient.

The minimisation of the elastic strain energy ( $E_{reg} = \int W dv$ , Section 2.2.2) has been used in several studies [Chr94] [CRM96] [LHH97] [HRS<sup>+</sup>99] [CP00] [PVC<sup>+</sup>01] [CPCZ03] [WS00] [BN96] [VGW06], since it provides physically-based regularisation (one-to-one mapping, no negative volume). Equivalently, one can use body forces (divergence of the stress), in a force equilibrium evolution process 3.3.5, where an equilibrium with external forces (image-based) is iteratively found. In the linear elasticity framework, body forces [TG70] are expressed as (linear elasticity operator):

$$\text{div}(\sigma) = \mu \Delta \mathbf{u} + (\lambda + \mu) \nabla(\text{div}(\mathbf{u})) = -\mathbf{f}_{\text{ext}}$$

where  $\lambda$  and  $\mu$  are the Lamé coefficients and  $\mathbf{u}$  the displacement. This linear elasticity operator has been used by Christensen et al. [Chr94] [CRM96] for image registration. They decompose lattice displacement on a basis formed by the truncated eigenvectors  $\Phi_{\mathbf{r}}$  of the linear elasticity operator ( $\mathbf{u} = \sum_{i,j,k,r=1}^{Nx, Ny, Nz, 3} \Phi_{\mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{r}}(\mathbf{p}) w_{i,j,k,r}$ ). The external force field is then regularised by computing coefficients  $w_r$ , obtained through the projection of forces onto eigenvectors. Since infinitesimal displacement is not a valid assumption, they introduce the fluid registration, where the internal stress is reinitialised at each iteration. To speed up the rather slow algorithm from Christensen et al. (several hours for 2D registration) Bro-Nielsen

et al. [BN96] implement a fast filtering method from the impulse response of the linear elasticity operator. As a natural extension, authors have investigated hyperelastic registration to allow large displacement in the elastic framework (no stress re-initialisation). Nowadays, sophisticated constitutive models are used for registration such as in [VGW06], where Veress et al. incorporate an anisotropic hyperviscoelastic constitutive model for heart registration. In addition to the large computational weight of such methods, there are some restrictions in using mechanical parameters, making physically-based regularisation not necessarily more robust than geometric regularisation: first, external forces from images do not have any physical meaning, so that they are difficult to blend with body forces. There are also topological problems due to the linearization of the elasticity operator. In addition, inter-patient registration based on mechanics does not have much sense (there is no matter deformation). Finally, the mechanical model is difficult to fit to the data because of assumptions (small displacement, linear isotropic behaviour, etc.) and unknown mechanical parameters. The problem can be over constrained such as the over penalisation of large rotations when using linear elasticity.

In multiple object segmentation, it is valuable to add extra constraints, preventing from interpenetrations. Few authors have actually incorporated neighbouring constraints and collision handling within an image registration process. In [YSD04], Yang et al. use a neighbour prior model within a Bayesian and level-set framework, to enforce relative organ positions according to a sample set (example-based approach). In [MT99] McInerney et al. check self-penetration of their T-snake, using spatial hashing, to allow topology change. Non self-intersecting forces are applied in [PMTK01] [MKAE00] on discrete models, by checking non-neighbouring triangles that self penetrate. Recent advances in collision handling from the computer graphics community [TKH<sup>+</sup>05] have however not been applied in image registration. Similarly, model interrelationship constraints such as organ attachment have not been applied to registration, since multiple object segmentation has remained marginal, considering the predominant applications (heart and brain). However it is of major importance in musculoskeletal segmentation.

### 3.3.5 Numerical resolution: model evolution

The solution of the registration problem is found by minimising the energy of the model, composed of an internal/ regularisation term and an external/ image-based potential energy:  $\hat{\mathbf{P}} = \underset{\mathbf{P}}{\operatorname{argmin}} E(\mathbf{P})$  ( $\mathbf{P}$ : parameters). At minimum, competing forces equilibrate and we have  $\mathbf{F} = -\nabla E = \mathbf{0}$ . In most cases, because of the non-convex aspect of the energy (noise, multiple local minima, etc.), there is no analytical solution. After discretisation, the energy is minimised iteratively using traditional multivariate optimisation strategies. Alternatively, when conferring a mechanical aspect to energy/ forces, the iterative process can be performed through a dynamic evolution of the system. Another class of evolution method is the Eulerian evolution, where the space is transformed rather than the model itself. The solution (model shape) is hence implicitly defined by the space state. Eulerian evolution has been investigated for the front propagation of implicit models (level-sets [OS88]) and will not be discussed here.

#### Explicit resolution

In some simple cases, where the system has a small number of DOFs, and the external energy a simple form, it is possible to analytically find the global solution. In general, this is the case in geometric registration because the external energy has a spatial interpretation and can incorporate the parameterisation of the transformation (no internal energy). In [AHB87] [Pen96], the best rigid, similarity or affine transform that minimise the distance (error in the least square sense) between paired point-sets is found, by applying the quaternion method. This is particularly useful in body motion capture, where the rigid motion of the different body segments need to be estimated from noisy marker motion. When not available, pairing can be performed by minimising locally the distance (e.g. closest point). The pairing is then updated at each iteration, where the optimal transform is found, until the convergence (P&S approach). This is the principle of the popular Iterative Closest Point algorithm [BM92].

Let:

- $\mathbf{x}$ : source points
- $\bar{\mathbf{x}} = \frac{1}{N} \sum_i \mathbf{x}_i$ : gravity center of source points
- $\underline{\mathbf{x}} = \mathbf{x} - \bar{\mathbf{x}}$
- $\mathbf{y}$ : target points
- $\bar{\mathbf{y}} = \frac{1}{N} \sum_i \mathbf{y}_i$ : gravity center of target points
- $\underline{\mathbf{y}} = \mathbf{y} - \bar{\mathbf{y}}$
- $\mathbf{m} = \underline{\mathbf{y}} - \underline{\mathbf{x}}$
- $\mathbf{p} = \underline{\mathbf{y}} + \underline{\mathbf{x}}$

Optimal transforms are given in Table 3.6.

Type	Problem	Optimal transform	Comments
Translation	$\tilde{\mathbf{t}} = \operatorname{argmin}(\sum_i \ \mathbf{y}_i - \mathbf{A}\mathbf{x}_i - \mathbf{t}\ ^2)$	$\tilde{\mathbf{t}} = \bar{\mathbf{y}} - \tilde{\mathbf{A}}\bar{\mathbf{x}}$	$\mathbf{A}$ : any homogeneous transform
Rigid transform	$(\tilde{\mathbf{R}}, \tilde{\mathbf{t}}) = \operatorname{argmin}(\sum_i \ \mathbf{y}_i - \mathbf{R}\mathbf{x}_i - \mathbf{t}\ ^2)$	The optimal rotation quaternion is the eigen vector with the smallest eigen value of the matrix: $\sum_i \begin{bmatrix} 0 & m_i^x & m_i^y & m_i^z \\ -m_i^x & 0 & p_i^z & -p_i^y \\ -m_i^y & -p_i^z & 0 & p_i^x \\ -m_i^z & p_i^y & -p_i^x & 0 \end{bmatrix}^2$	Translation: same than above
Similitude	$(\tilde{\mathbf{R}}, \tilde{\mathbf{t}}, \tilde{s}) = \operatorname{argmin}(\sum_i \ \mathbf{y}_i - s\mathbf{R}\mathbf{x}_i - \mathbf{t}\ ^2)$	$\tilde{s} = \frac{\operatorname{Tr}(\tilde{\mathbf{R}} \sum_i \mathbf{y}_i \mathbf{x}_i^T)}{\sum_i \ \mathbf{x}_i\ }$	Rigid transform: same than above
Affine transform	$(\tilde{\mathbf{A}}, \tilde{\mathbf{t}}) = \operatorname{argmin}(\sum_i \ \mathbf{y}_i - \mathbf{A}\mathbf{x}_i - \mathbf{t}\ ^2)$	$\tilde{\mathbf{A}} = \sum_i \underline{\mathbf{x}}_i \underline{\mathbf{y}}_i^T (\sum_i \underline{\mathbf{x}}_i \underline{\mathbf{x}}_i^T)^{-1}$	Translation: same than above

Table 3.6: Explicit computation of the registration transform

### Energy minimisation

The minimisation, aimed at seeking the equilibrium, is driven by forces (energy spatial derivatives). Applying the Euler-Lagrange equation, the energy minimisation problem is turned into a force equilibrium equation (stationary equation) that can be used to derive the evolution of the model. Space discretisation is performed by the finite difference method [TPBF87] [TW88], the finite element/volume method [CC93] [MT95] [PVC<sup>+</sup>01] or any parameterisation scheme (e.g. B-Splines) that lumps system degrees of freedom to control nodes. After discretisation, we obtain  $-\nabla \mathbf{E}(\mathbf{P}) = \mathbf{F}_{\text{int}}(\mathbf{P}) + \mathbf{F}_{\text{ext}}(\mathbf{P}) = \mathbf{K}\mathbf{P} + \mathbf{F}_{\text{ext}}(\mathbf{P})$  ( $\mathbf{K}$ : the stiffness matrix) which is zero at equilibrium (local energy minimum).

Instead of minimising the energy (maximum likelihood scheme), some authors maximise the probability of model shape given the image (maximum a posteriori scheme), through a Bayesian framework [SD92] [WS00]

[TS92] [CKP00]. This can be viewed as an extra layer to which optimisation is performed. The Bayesian approach is interesting for incorporating statistical shape information (example-based approach) and image noise.

**Local optimisation:** To find the minimum of the energy, exhaustive research methods, or quasi-exhaustive methods (e.g. multigrid) may suit when the number of degrees of freedom is small (e.g. translation). In most cases, the complete inspection of the search space is too costly, and the search must be oriented. A popular oriented search method is the Downhill Simplex method [NM65] that provides N-dimensional bracketing of the solution (by N+1 points). The associated figure (simplex) is successively transformed through reflections, expansions and contractions, until a tolerance (simplex volume) is reached. Amoeba is an implementation of the downhill simplex method [PTVF92]. Another way to seek a local minimum is to descend according to the gradient of the energy, when it is available. In other words, the model evolves in the steepest energy direction with a certain speed. The gradient-descent method is widely used in the field [CP00] [Thi95] [CKP00] [SD00]. However, the convergence might require a large number of iterations (oscillations often occur around the solution). To speed up the convergence, one may adapt the time-step and the direction of the evolution. In the conjugate gradient algorithm, the direction of the evolution is forced to be orthogonal to the previous ones. The Powell's methods provide conjugate directions without computing the derivatives. In the Newton algorithm, the time-step is computed, assuming that the energy is quadratic (2nd order Taylor development of the energy). The Levenberg-Marquardt method smoothly combines the Newton and the gradient descent algorithm. In fact, the Newton method is more suited near a local minimum, whereas the gradient descent works better far away from the solution. Newton method involves the expensive evaluation of the Hessian matrix of the energy. Quasi-Newton methods have been proposed to update an Hessian matrix estimate at each iteration, considering energy and energy gradient change [Bro69]. Vemuri et al. use a pre-computed Hessian matrix in [VHS<sup>+</sup>97]. Considering the first order development of the energy that we want to be null:  $E(\mathbf{P}_{t+dt}) \simeq E(\mathbf{P}_t) + \nabla E(\mathbf{P}_t) \cdot (\mathbf{P}_{t+dt} - \mathbf{P}_t) = 0$ , and assuming a displacement along the gradient  $\mathbf{P}_{t+dt} - \mathbf{P}_t = -\nabla E(\mathbf{P}_t) dt$ , we get the Newton-Raphson method where only first order energy derivatives (forces) need to be computed. In [MHHR06], Müller et al. use this method within a mechanical system to derive its dynamic evolution.

Method	Evolution equation
Gradient descent	$\mathbf{P}_{t+dt} - \mathbf{P}_t = -\nabla E(\mathbf{P}_t) dt$
Newton	$\mathbf{P}_{t+dt} - \mathbf{P}_t = -\nabla^2 E(\mathbf{P}_t)^{-1} \nabla E(\mathbf{P}_t)$
Quasi-Newton	$\mathbf{P}_{t+dt} - \mathbf{P}_t = -A^{-1} \nabla E(\mathbf{P}_t)$ where $A \simeq \nabla^2 E(\mathbf{P}_t)$
Levenberg-Marquardt	$\mathbf{P}_{t+dt} - \mathbf{P}_t = -(\nabla^2 E(\mathbf{P}_t) + \rho \mathbf{I})^{-1} \nabla E(\mathbf{P}_t)$
Newton-Raphson	$\mathbf{P}_{t+dt} - \mathbf{P}_t = -\ \nabla E(\mathbf{P}_t)\ ^{-2} E(\mathbf{P}_t) \nabla E(\mathbf{P}_t)$

Table 3.7: Local energy minimisation schemes

**Global methods:** Above methods are local, meaning they work well when the energy has few local minima (convex energy). If not, stochastic optimisation methods such as the simulated annealing or evolutionary algorithms can be used to find a global solution, despite their significant computational load. The simulated annealing (or stochastic relaxation) randomly changes parameter values at each iteration, and allows energy increase depending on the temperature (the higher the temperature, the higher rise is allowed). Acceptance probability is given by the metropolis function:  $p = \min\{1, -\exp(\delta E/T)\}$  where  $T$  is the temperature. The temperature decreases until the instant freezing, where allowed perturbations only make a reduction of the energy. Iterated Conditional Modes (ICM) or greedy algorithm is a simplification of simulated annealing, where  $T$  is set to zero (therefore a local minima can be reached). The Gibbs sampler [Sto94] uses an initial probabilistic distribution of parameters  $p(\mathbf{P})$ , and new model state is drawn from it and automatically

accepted. The Mean Field Annealing (MFA) replaces the random search with a deterministic search based on image characteristics [BMM<sup>+</sup>88] [SLS<sup>+</sup>92]. Evolutionary algorithms (e.g. genetic algorithms [Koz98]) are based on the Darwinian principle where a population of agents (parameters) evolves according to a fitness function (energy) and gene crossing principles. In [MH06], a genetic algorithm is used along with deformable models for medical image segmentation.

Dynamic programming can be used to find a global solution. It is based on the decomposition of the energy into independent terms. Solving these sub-problems, the global solution can be recovered by finding the shortest path in the energy graph. Amini et al. [AWJ90] have applied dynamic programming to 2D snakes. But, this has not been extended in 3D, the critical point being the ability to decompose the energy.

### Dynamic evolution

In a dynamic system, forces drive the model, that is the position and velocity of control points. This is particularly suited for discrete models that are commonly considered as lumped mass particles, moving according to forces (see Section 2.2.3). Dynamic systems are judged against their stability and accuracy, resulting from the numerical integration of the differential equation. The Newtonian law of motion leads to a first-order differential equation system relating the force vector  $\mathbf{F}$  to the particle velocity vector  $\mathbf{V}$  and position vector  $\mathbf{P}$ :

$$\begin{aligned}\frac{\partial \mathbf{P}}{\partial t} &= \mathbf{V} \\ \mathbf{M} \frac{\partial \mathbf{V}}{\partial t} &= \mathbf{F}(\mathbf{P}, \mathbf{V})\end{aligned}$$

The force vector  $\mathbf{F}(\mathbf{P}, \mathbf{V})$  depends on particle position (internal and external forces/ energy derivative) and velocity (damping representing the viscosity of the milieu/ energy dissipation). When discretising the system with finite differences, we obtain:

$$\begin{aligned}\mathbf{u} = \mathbf{P}_{t+dt} - \mathbf{P}_t &= \mathbf{V} dt \\ \mathbf{V}_{t+dt} - \mathbf{V}_t &= \mathbf{M}^{-1} \mathbf{F}(\mathbf{P}, \mathbf{V}) dt\end{aligned}\tag{3.1}$$

The Lagrangian evolution, that assumes massless particles and unitary isotropic viscosity [TM91] ( $\mathbf{F}(\mathbf{P}, \mathbf{V}) = -\nabla \mathbf{E}(\mathbf{P}_t) - \mathbf{IV} = \mathbf{0}$ ) leads to  $\mathbf{P}_{t+dt} - \mathbf{P}_t = -\nabla \mathbf{E}(\mathbf{P}_t) dt$ , which is equivalent to the gradient descent algorithm.

In the forward Euler method, velocities and forces are taken at time  $t$  ( $\mathbf{V} = \mathbf{V}_t$  and  $\mathbf{F}(\mathbf{P}, \mathbf{V}) = \mathbf{F}(\mathbf{P}_t, \mathbf{V}_t)$ ), so that the new state vector ( $P_{t+dt}, V_{t+dt}$ ) is explicitly defined in the system 3.1. However it goes forward quite blindly (no notice of force derivatives), so that the stability is ensured for very small time steps  $dt$ . A higher order forward method is the well known Runge-Kutta method [PTVF92]. Its principle is to use force multiple evaluation to better extrapolate the new state vector through higher order Taylor expansion. As an example, the 2nd order Runge Kutta (or mid-point method) is given by:

$$\begin{aligned}\mathbf{k1} &= \mathbf{M}^{-1} \mathbf{F}(\mathbf{P}_t, \mathbf{V}_t) dt \\ \mathbf{k2} &= \mathbf{M}^{-1} \mathbf{F}(\mathbf{P}_{t+dt/2}, \mathbf{V}_t + \mathbf{k1}/2) dt \\ \mathbf{V}_{t+dt} &= \mathbf{V}_t + \mathbf{k2} + O(dt^3)\end{aligned}\tag{3.2}$$

A simple way for improving stability, with no extra computational cost, is to take the implicit formulation of the first equation of 3.1:  $\mathbf{P}_{t+dt} - \mathbf{P}_t = \mathbf{V}_{t+dt}dt$ , where  $\mathbf{V}_{t+dt}$  can be obtained with the forward step described above. Combining the two equations, we get:

$$\begin{aligned}\mathbf{P}_{t+dt} &= 2\mathbf{P}_t - \mathbf{P}_{t-dt} + \mathbf{M}^{-1}\mathbf{F}(\mathbf{P}_t, \mathbf{V}_t).dt^2 \\ \mathbf{V}_{t+dt} &= (\mathbf{P}_{t+dt} - \mathbf{P}_t)dt^{-1}\end{aligned}\tag{3.3}$$

The Verlet integration method [Ver67] is obtained by modifying the second equation by:  $\mathbf{V}_{t+dt} = (\mathbf{P}_{t+dt} - \mathbf{P}_{t-dt})(2dt)^{-1}$ . It is a popular forward integration method that has been recently applied to deformable models due to its simplicity [THMG04] (only one force evaluation per time-step). It takes into account the previous time step  $t - dt$  and is more accurate than the explicit Euler ( $O(dt^4)$  precision). Unfortunately, forward methods still suffer from tight time-step restriction (even with adaptive time-stepping) which is prohibitive for real-time applications.

In the implicit methods, the idea is to estimate state vector temporal derivatives at  $t + dt$  using its spatial derivatives. Let  $\mathbf{Q}_t = [\mathbf{P}_t, \mathbf{V}_t]$  be the state vector, and  $\alpha$  the implicitness of the method, the system 3.1 can be rewritten as [VMT05]:

$$\mathbf{Q}_{t+dt} - \mathbf{Q}_t = \mathbf{Q}'_{t+\alpha dt}dt$$

The first-order Taylor expansion leads to:

$$\mathbf{Q}_{t+dt} - \mathbf{Q}_t = \mathbf{Q}'_t dt + \alpha \frac{\partial \mathbf{Q}'}{\partial \mathbf{Q}} (\mathbf{Q}_{t+dt} - \mathbf{Q}_t) dt\tag{3.4}$$

In the explicit Euler method,  $\alpha = 0$ , while in the implicit Euler method,  $\alpha = 1$ . The implicit midpoint corresponds to  $\alpha = 1/2$ . Implicit methods are called backward since a forward step from new state vector at  $t + dt$  brings it back to the initial state vector at  $t$ . Hence, results are always consistent, allowing larger time-steps. Since [BW98], implicit methods have become popular in computer graphics. However implicit methods are not necessarily more accurate than explicit methods, due to numerical damping. A comparison of explicit/implicit methods in terms of stability and accuracy is done in [VMT01] [VMT05] [HE01]. Applied to the mechanical system, the implicit scheme 3.4 leads to:

$$\begin{aligned}\mathbf{P}_{t+dt} - \mathbf{P}_t &= (\mathbf{V}_t + \alpha(\mathbf{V}_{t+dt} - \mathbf{V}_t))dt \\ \mathbf{V}_{t+dt} - \mathbf{V}_t &= \mathbf{H}^{-1}\mathbf{Y}\end{aligned}$$

where:

$$\begin{aligned}\mathbf{H} &= \mathbf{I} - \alpha\mathbf{M}^{-1}\frac{\partial \mathbf{F}}{\partial \mathbf{V}}dt - \alpha^2\mathbf{M}^{-1}\frac{\partial \mathbf{F}}{\partial \mathbf{P}}dt^2 \\ \mathbf{Y} &= \mathbf{M}^{-1}\mathbf{F}(\mathbf{P}_t, \mathbf{V}_t)dt + \alpha\mathbf{M}^{-1}\frac{\partial \mathbf{F}}{\partial \mathbf{P}}\mathbf{V}_tdt^2\end{aligned}\tag{3.5}$$



Consequently, the implicit resolution is equivalent to the resolution of a large sparse linear equation system (inversion of  $\mathbf{H}$ ). The more stiff (large number of eigenvalues) is the system, the more complicated is the resolution (more solver iterations). In [DSB99], only the linear part is resolved, through pre-computed inverse matrix. In [EEH00], explicit resolution is applied to the non-stiff non-linear part, while implicit resolution is applied to the stiff linear part (IMEX method), which improves accuracy (less numerical damping due to implicit resolution) and computational speed. In the above system, the implicitity parameter  $\alpha$  governs the trade-off between stability and computational speed (small  $\alpha$  value reduces system stiffness) [VMT05].

Supposing that  $\alpha = 1$  (Euler implicit scheme),  $\mathbf{F}(\mathbf{P}_{t+dt}, \mathbf{V}_{t+dt}) \simeq \mathbf{K}(\mathbf{P}_t)\mathbf{P}_{t+dt} + \mathbf{F}_{\text{ext}}(\mathbf{P}_t) - \mathbf{C}(\mathbf{V}_{t+dt} + \mathbf{V}_t)/2$  (isotropic damping),  $\frac{\partial \mathbf{F}}{\partial \mathbf{P}} \simeq \mathbf{K}(\mathbf{P}_t)$  and  $\frac{\partial \mathbf{F}}{\partial \mathbf{V}} \simeq -\mathbf{C}/2$ , we obtain from Equation 3.5, a semi-implicit scheme, since  $\mathbf{F}_{\text{ext}}$  is evaluated at  $t$ . This is equivalent to the scheme used in the foundational work on deformable models by Terzopoulos et al. [TPBF87] [TW88]:

$$\begin{aligned}\mathbf{H} &= \mathbf{I} + \mathbf{M}^{-1}\mathbf{C}dt/2 - \mathbf{M}^{-1}\mathbf{K}(\mathbf{P}_t)dt^2 \\ \mathbf{Y} &= \mathbf{M}^{-1}(\mathbf{K}(\mathbf{P}_t)\mathbf{P}_t + \mathbf{F}_{\text{ext}}(\mathbf{P}_t) - \mathbf{C}\mathbf{V}_t)dt + \mathbf{M}^{-1}\mathbf{K}(\mathbf{P}_t)\mathbf{V}_tdt^2\end{aligned}$$

To increase accuracy, authors have used history states to better approximate the new state. Particularly, Backward Differentiation Formulas (BDF) have been recently applied to deformable models [HE01] [Hau04]. They are computationally inexpensive, since they only require one system resolution at each step, contrary to Runge-Kutta methods. BDF are less robust in the cases of nonlinear systems, since they are sensitive to sudden state change (e.g. collision response) due to their dependence with previous states. In practice, BDF are limited to their second order (first order is equivalent to the implicit Euler). The progressive BDF-2 is formulated as [VMT05]:

$$\mathbf{Q}_{t+dt} - \mathbf{Q}_t = \beta(\mathbf{Q}_t - \mathbf{Q}_{t-dt}) + \mathbf{Q}'_{t+\alpha dt}\delta t$$

with  $\beta = \frac{2\alpha-1}{2\alpha+1}$  and  $\delta t = \frac{2}{2\alpha+1}dt$ .  $\alpha = 1/2$  corresponds to the implicit midpoint ( $\beta = 0$ ,  $\delta t = dt$ ). In the framework of the mechanical system, we have:

$$\begin{aligned}\mathbf{P}_{t+dt} - \mathbf{P}_t &= \beta(\mathbf{P}_t - \mathbf{P}_{t-dt}) + (\mathbf{V}_t + \alpha(\mathbf{V}_{t+dt} - \mathbf{V}_t))\delta t \\ \mathbf{V}_{t+dt} - \mathbf{V}_t &= \mathbf{H}^{-1}\mathbf{Y}\end{aligned}$$

where:

$$\begin{aligned}\mathbf{H} &= \mathbf{I} - \alpha\mathbf{M}^{-1}\frac{\partial \mathbf{F}}{\partial \mathbf{V}}\delta t - \alpha^2\mathbf{M}^{-1}\frac{\partial \mathbf{F}}{\partial \mathbf{P}}\delta t^2 \\ \mathbf{Y} &= \beta(\mathbf{V}_t - \mathbf{V}_{t-dt}) + \mathbf{M}^{-1}\mathbf{F}(\mathbf{P}_t, \mathbf{V}_t)\delta t + \alpha\mathbf{M}^{-1}\frac{\partial \mathbf{F}}{\partial \mathbf{P}}(\mathbf{V}_t\delta t + \beta(\mathbf{P}_t - \mathbf{P}_{t-dt}))\delta t\end{aligned}\tag{3.6}$$

The resolution of the system always involves the inversion of a large sparse matrix. Terzopoulos et al. [TPBF87] resolve the linear system with a Choleski decomposition and a relaxation method (Gauss-Seidel). Recent applications use the Conjugate Gradient (CG) algorithm, which is particularly adapted to sparse systems [PTVF92]. The CG algorithm works on symmetric positive-definite systems which is not the case

of 3.5 or 3.6, due to mass-modified particles. Post-multiplication [VMT00b] or pre-multiplication [BW98] by the inverse mass matrix can be performed to recover global symmetry. In [HE01] [Hau04], the CG algorithm is integrated in the Newton resolution method, allowing error control and adaptive time stepping.

The Conjugate Gradient can be accelerated using matrix preconditioning [BE79] [She94]:  $\mathbf{V}_{t+dt} - \mathbf{V}_t = \mathbf{H}^{-1}\mathbf{Y}$  is equivalent to  $\mathbf{V}_{t+dt} - \mathbf{V}_t = (\mathbf{U}^{-1}\mathbf{H})^{-1}\mathbf{U}^{-1}\mathbf{Y}$  where  $\mathbf{U}$  is the symmetric positive-definite preconditioning matrix. Indeed, if  $\mathbf{U}^{-1}\mathbf{H}$  contains less non-zero elements than  $\mathbf{H}$ , its inversion is cheaper. The problem is that  $\mathbf{U}^{-1}\mathbf{H}$  is not necessarily symmetric positive-definite. To overcome this,  $\mathbf{U}$  is decomposed into  $\mathbf{U} = \mathbf{E}\mathbf{E}^T$ . Then, it can be shown [She94] that the problem is equivalent to solving  $\mathbf{E}^{-T}(\mathbf{V}_{t+dt} - \mathbf{V}_t) = (\mathbf{E}^{-1}\mathbf{H}\mathbf{E}^{-T})^{-1}\mathbf{E}^{-1}\mathbf{Y}$  where  $(\mathbf{E}^{-1}\mathbf{H}\mathbf{E}^{-T})$  is symmetric positive-definite, thus invertible with the CG algorithm.  $\mathbf{E}$  may not be computed, as clever substitutions can be done within the CG process, so that only the ability of computing the effect of applying  $\mathbf{U}^{-1}$  to a vector is required (this is called the Untransformed Preconditioned Conjugate Gradient method). Popular preconditioners are the Jacobi preconditioner ( $\mathbf{U}_J = \text{diagonal of } \mathbf{H}$ ) and SSOR (Symmetric Successive Overrelaxation Method) conditioner ( $\mathbf{U}_{\text{SSOR}} = (\mathbf{U}_J + \mathbf{L})\mathbf{U}_J^{-1}(\mathbf{U}_J + \mathbf{L})^T$  where  $\mathbf{L}$  is the lower triangular part of  $\mathbf{H}$ ). Following the idea of Terzopoulos et al. [TW88], forces can be applied on rigidly registered models (non-rotated references), leading to more sparse system, especially when using linear Cauchy strain [MG04]. This can be viewed as a particular preconditioning method.

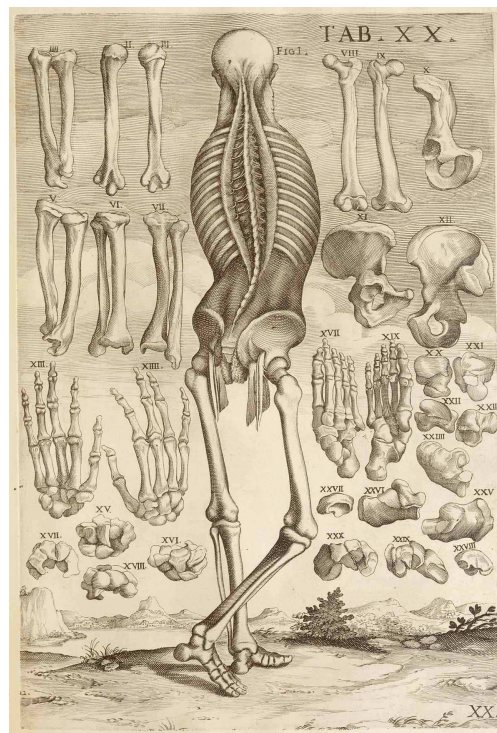
In [PTVF92] and [Hau04], other integration methods are presented, particularly the Rosenbrock method. This method requires many stages making it not competitive from a computational speed point of view. Comparisons between the different integration techniques can be found in [VMT01] [Hau04] [VMT05].

### 3.4 Conclusion

Most proposed registration techniques remain generic and fairly independent from the targeted applications. The lack of specificity makes methods difficult to parameterise, difficult to evaluate and generally holds back their implementation in the clinical environment. However, it is reducing due to more-and-more established physical-based simulation methods, available data for example-based approaches and cross-validation studies. Unifying tissue simulation and segmentation could lead to a better specificity for daily clinical use. Moreover, physically-based constraints, taking into account not only pixel intensity but high-level parameters such as the shape, the function and the mechanics of tissues, improve the accuracy and the robustness of registration methods by reducing problem degrees of freedom. Transformations can be limited to admissible deformations through examples [DSMK01] or/ and prediction methods [VGW06]. This is particularly relevant for intra-patient registration, while, for inter-patient registration or longitudinal studies, it is required to study how high-level parameters change across individuals or through time. For inter-patient registration, we expect that external forces would play a more significant role than for intra-patient registration (image-driven versus physics-driven). In this context, deformable surfaces offer a moderate complexity and a good flexibility to tune intrinsic, geometrical and data constraints. This is clearly valuable for musculoskeletal modelling, as shown in Chapter 2. Besides, physically-based volumetric models are more suited for patient-specific tissue simulation and intra-patient registration.

## Chapter 4

# Simplex meshes



<sup>1</sup>Pietro Berrettini da Cortona. *Tabulae anatomicae*. Rome, 1741, Copperplate engraving, National Library of Medicine. [http://www.nlm.nih.gov/exhibition/dreamanatomy/da\\_g\\_I-D-3-16.html](http://www.nlm.nih.gov/exhibition/dreamanatomy/da_g_I-D-3-16.html)

## 4.1 Introduction

In this chapter, we propose general methods for discrete deformable model-based segmentation. This embraces the different aspects of the process: the construction and adaptation of models, the analysis of their geometry, the computation of relevant internal and external constraints and the evolution of model state (vertex positions and velocities). Although our methods can be applied/ adapted to any type of discrete model (and most parametric models), we focus on simplex meshes since they have proven to be particularly efficient in terms of flexibility and computational cost, thanks to their simple geometric description. We extend and generalise current mesh topological operators for fitting vertex positions and connectivity relationships to object shapes at a particular resolution. We express different geometric-based internal and external constraints, relevant for musculoskeletal modelling, in terms of desired particle positions and velocities. Particularly, we try to confer a volumetric aspect to simplex surfaces through the computation of reversible medial axis, in order to enhance regularisation mechanisms at a moderate cost. Due to the multi-object aspect of the musculoskeletal system, we introduce scalable methods to efficiently handle collisions. Finally, we investigate evolutions schemes and try to optimise the trade-off between accuracy, robustness and speed.

## 4.2 Background

Simplex meshes have been introduced by H. Delingette [Del99] [Del94a] [Del94b] for constrained 3D shape reconstruction and segmentation. Later, they have been extended by J. Montagnat [Mon99] [MD00] [MD05] for 4D segmentation, through the introduction of global spatiotemporal constraints. A  $k$ -simplex mesh is a  $k$ -manifold discrete mesh where vertices have exactly  $k + 1$  distinct neighbours. Neighbouring vertices are connected by edges, edge closed successions form faces and faces closest successions form cells. Faces/cells may correspond to holes (mesh ends). Ends of  $k$ -simplex meshes correspond to  $(k - 1)$ -simplex meshes (see Figure 4.2). Depending on their connectivity  $k$ , simplex meshes can represent various objects such as curves ( $k = 1$ ), surfaces ( $k = 2$ ) or volumes ( $k = 3$ ), with any kind of topology [Del99] (see Figure 4.1). Simplex meshes are submitted to the Euler-Poincaré rule:  $F(v, e, f, c, H, g) = 0$  where  $v$  is the number of vertices,  $e$  the number of edges,  $f$  the number of faces (for  $k > 1$ ),  $c$  the number of cells (for  $k > 2$ ),  $g$  the genus (topology of the solid e.g.  $g = 0$  for a sphere,  $g = 1$  for a torus) and  $H$  the number of holes/ ends. The Euler-Poincaré formula is simplified for simplex meshes due to the constant vertex connectivity. This is summarised in Table 4.1.

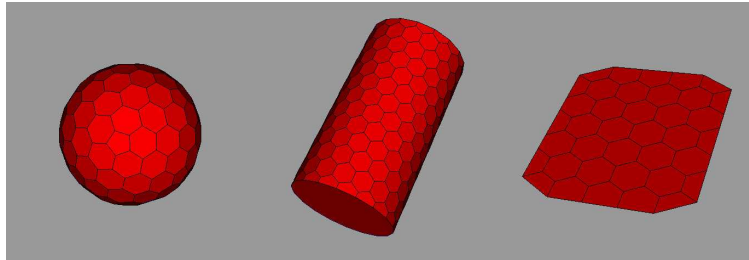


Figure 4.1: Different 2-simplex primitives

Vertices are oriented according to neighbours ordering (see Figure 4.2). When vertices are coherently oriented, face orientation can be defined. For 3-simplex meshes, face orientation depends on the considered cell: let be a vertex  $\mathbf{P}$  and its ordered neighbours  $\{\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3, \mathbf{P}_4\}$  (see Figure 4.2). Its neighbouring cell  $C_1$ , not containing  $\mathbf{P}_4$ , is a 2-simplex mesh of which faces are oriented coherently with regards to  $\mathbf{P} : \{\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3\}$ . If we consider the cell  $C_2$ , not containing  $\mathbf{P}_1$ , we perform a circular permutation of the neighbours:  $\{\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3, \mathbf{P}_4\} \longrightarrow -\{\mathbf{P}_2, \mathbf{P}_3, \mathbf{P}_4, \mathbf{P}_1\}$ . Consequently,  $C_2$  faces are oriented according to

$k$	Euler-Poincaré rule
1	$v = e$ or $v = e + H$
2	$f + H - v/2 = 2(1 - g)$
3	$v + c + H - f = -2g$

Table 4.1: The Euler-Poincaré rule for  $k$ -simplex meshes

the vertex ordering  $-\{\mathbf{P}_2, \mathbf{P}_3, \mathbf{P}_4\}$  that is equivalent to  $\{\mathbf{P}_2, \mathbf{P}_4, \mathbf{P}_3\}$ . Orientation in 3-simplex meshes is important to check cell inversion (self-collision), resulting in negative cell volumes.

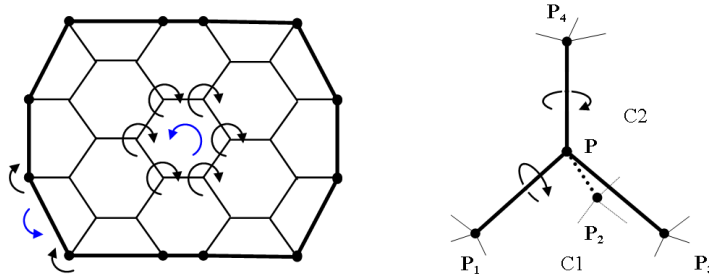


Figure 4.2: A) Closed 2-simplex mesh with one hole (outside). Its boundary corresponds to a 1-simplex mesh (in bold). Vertex orientation results in face orientation (in blue). B) 3-simplex mesh vertex orientation

### 4.3 Duality

$k$ -simplex meshes are topologically dual to classical  $k$ -solid meshes (e.g. polyline, triangle mesh, tetrahedral mesh) as shown in Figure 4.3. Polygonal meshes are characterised by a constant number of vertices per face, while simplex meshes exhibit a constant vertex connectivity. Dual polygonal meshes are obtained by linking the centers of adjacent simplex edges ( $k = 1$ ), faces ( $k = 2$ ) or cells ( $k = 3$ ). However, different polygonal meshes are realisable for a unique simplex meshes (e.g. different triangulations for a given simplex surface, Figure 4.4). Consequently, the polygonal mesh/ simplex mesh correspondence is purely topological (no unique geometrical equivalence). Note that simplex meshes do not correspond to the Voronoi diagram of their dual solids (obtained by linking face/ cell circumscribed circle/ sphere centers) except for a special configuration called *centroidal Voronoi tessellation* (CVT) [DW03] where each vertex is the center of its own Voronoi cell.

It is important to be able to convert simplex meshes into polygonal meshes, since triangle meshes are supported by graphics hardware and tetrahedral meshes by FEM software. To better preserve the geometric quality of initial simplex meshes, we can include their initial vertices, and link them with face/ cell centers (the resulting mesh is the dual of the higher resolution simplex mesh according to the multi-resolution scheme A, presented in Section 4.4.4). Figure 4.4 shows the method for simplex surfaces. For 3-simplex meshes, the method works similarly by adding faces and cell centers. Resulting meshes conserve the curvature information and shape information at boundaries, but have a higher number of vertices ( $v' = v + f$  for  $k = 2$ ) than dual meshes ( $v' = f$  for  $k = 2$ ).

For  $k = 2$ , equilateral triangles are obtained from regular hexagons, where simplex vertices are the centers of their neighbours. This corresponds to the CVT configuration. It has been shown that the optimal triangulation (optimal vertex repartition and optimal connectivity) is obtained from it [DW03]. Consequently,

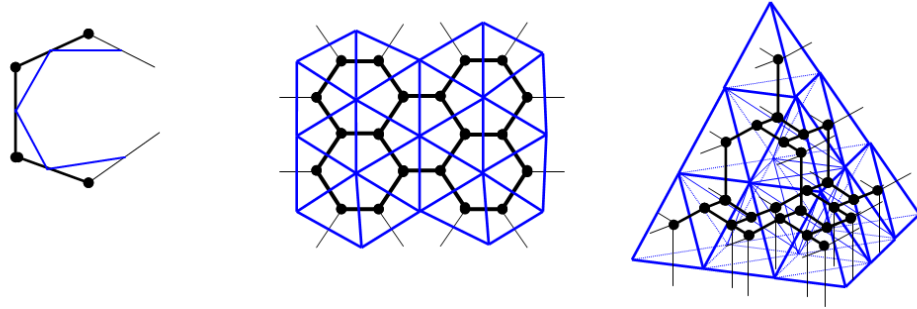


Figure 4.3: Opened simplex meshes (black) and their dual polygonal meshes (blue): a) 1-simplex mesh/ polyline; b) 2-simplex mesh/ triangle mesh; c) 3-simplex mesh/ tetrahedral mesh

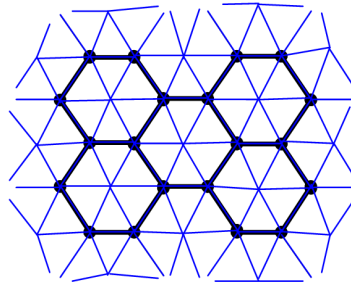


Figure 4.4: A triangulation scheme that preserves shape features of 2-simplex meshes. The resulting mesh is the dual of the higher resolution simplex mesh

remeshing algorithms that iteratively minimise the distance towards it, through variational approaches, have been developed [SAG03]. We expect similar results from the triangulation of quasi-regular simplex meshes (see Section 4.4.5 for results). Note that, with ordinary surfaces, homeomorphic to spheres ( $g = 0$ ), it is not possible to have hexagons only. According to the Euler rule, we can show that, if they are composed of pentagons and hexagons, the number of pentagons is exactly 12 and the number of hexagons is free.

For  $k = 3$ , there is no regular configuration, leading to regular tetrahedra since tetrahedra do not tile space (their dihedral angle is  $\arccos(1/3)$ ). A regular configuration of simplex vertices (leading to regular dual tetrahedra) is obtained when vertices are the barycenter of their 4 neighbours. In this case, edges form angles of  $\pi - \alpha$ . Since the closest cell type is the regular dodecahedron cell (edge-edge angle of  $3\pi/5 \simeq \alpha$ ), we expect to obtain mostly dodecahedrons when proceeding to mesh optimisation. Note that gathered dodecahedrons are not 3-simplex meshes. They must be completed with other types of cells [Del94a].

## 4.4 Geometric and topological quality

In this section, we examine how simplex mesh connectivity can be improved to reach regular meshes.

### 4.4.1 Basic operators

The geometric quality (uniformity of vertices repartition) and topological quality (uniformity of edge/ face number among faces/ cells) of simplex meshes are important to ensure a good matching of the model with the object we want to represent. H. Delingette [Del99] [Del94a] [Del94b] shows that they can be easily improved through basic operators that change mesh connectivity and topology. There are six basic operators from

which all possible mesh connectivity changes are realisable. Of course, the Euler-Poincaré rule remains valid. Tables 4.2, 4.3 and 4.4 summarise these operators for  $0 < k < 4$ . Restrictions on the use of these operators are related to degenerated configurations (e.g. faces with two vertices, cells with three faces, faces with three common vertices). These correspond to non-complete meshes [Del94a]. During mesh optimisation, we have to check carefully that we do not achieve these configurations.



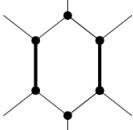
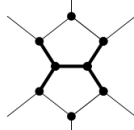
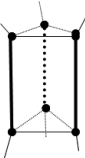
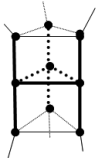
$k$	$TO_1 \leftarrow$	$\rightarrow TO_2$	$\Delta v$	$\Delta e$	$\Delta f$	$\Delta c$	$\Delta g$	$\Delta H$
1			1	1	-	-	-	0
2			2	3	1	-	0	0
3			n	2n	n+1	1	0	0

Table 4.2:  $TO_1$  and  $TO_2$  operators

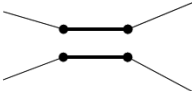
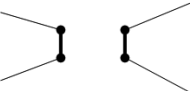
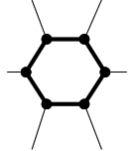
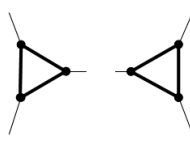
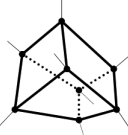
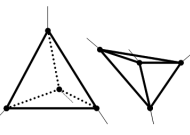
$k$	$TO_4 \leftarrow$	$\rightarrow TO_3$	$\Delta v$	$\Delta e$	$\Delta f$	$\Delta c$	$\Delta g$	$\Delta H$
1			0	0	-	-	-	0
2			0	0	2	-	-1	0
3			0	0	0	2	-1	0

Table 4.3:  $TO_3$  and  $TO_4$  operators

$k$	$TO_6 \longleftarrow$	$\longrightarrow TO_5$	$\Delta v$	$\Delta e$	$\Delta f$	$\Delta c$	$\Delta g$	$\Delta H$
1			0	-1	-	-	-	1
2			0	0	-1	-	0	1
3			0	0	0	-1	0	1

Table 4.4:  $TO_5$  and  $TO_6$  operators

#### 4.4.2 The exchange operation

To improve the topological quality of simplex meshes, an interesting operation is to change the repartition of edges across faces (and faces across cells). The exchange operator is a macro-operator that combines  $TO_1$  and  $TO_2$  without changing the total number of vertices, edges and faces. This operation has been defined by Delingette et al. [Del94b] for two adjacent vertices of 2-simplex meshes (1<sup>st</sup> order operation). For the dual triangle mesh, it corresponds to the well known swapping operator applied to two adjacent triangles. We extend the exchange operation to the 2<sup>nd</sup> order by considering vertex 2-neighbourhood (see Table 4.5). The 2<sup>nd</sup> order exchange operation can be decomposed into four successive 1<sup>st</sup> order exchanges. When proceeding to an exchange on 2-simplex meshes, we want to decrease the deviation of the number of edges per face with regards to a target number of edges per face  $\tilde{n}$ . Given the number of edges  $n_i$  of the face  $i$ , the exchange operation is valuable if  $\sum(n_i + \Delta n_i - \tilde{n})^2 < \sum(n_i - \tilde{n})^2$ . In this expression,  $\tilde{n}$  can be simplified (see Table 4.5). In other words, the criterion upon which the exchange is performed is independent of the target number of edges per face.

$k$	Order		Criteria	
2	1		$\longrightarrow n_1 + n_2 - n_3 - n_4 > 2$ $\longleftarrow n_3 + n_4 - n_1 - n_2 > 2$	
	2		$\longrightarrow n_1 + n_2 + n_3 - n_4 - n_5 - n_6 > 6$ $\longleftarrow n_4 + n_5 + n_6 - n_1 - n_2 - n_3 > 6$	

Table 4.5: The exchange operators for 2-simples meshes



For 3-simplex meshes, the definition of the exchange operation is not straightforward because  $TO_1$  applied to a face created by  $TO_2$  generally changes the total number of vertices, faces and edges. The swapping operation applied to dual tetrahedra [CDM04] leads to an ambiguous definition of edges. There is an exception with faces composed of four vertices (see Table 4.6), the condition being  $n_1 + n_2 - n_3 - n_4 > 2$  where  $n_i$  is the number of faces of the cell  $i$ . We have found that another  $TO_1/TO_2$  combination is valuable for 3-simplex meshes. It consists in converting triangles into edges (see Table 4.6), thus removing such undesirable small faces.

$k$		$\longrightarrow TO_2$	$\longrightarrow TO_1$	Criteria
3				$n_1 + n_2 - n_3 - n_4 > 2$

Table 4.6: Two exchange operations for 3-simples meshes

#### 4.4.3 Local resolution change

To improve the geometric quality, it is possible to locally add/ remove a face (resp. cell) using the  $TO_2/TO_1$  operators, according to the surface (resp. volume) and elongation of the face (resp. cell).

Given the target face surface  $\tilde{S}_f$  of a 2-simplex mesh and the surface  $sf$  of a face, we subdivide it with  $TO_2$  if  $2(sf/2 - \tilde{S}_f)^2 < (sf - \tilde{S}_f)^2 \Leftrightarrow sf > \tilde{S}_f\sqrt{2}$ . Reciprocally, we remove the edge separating the face 1 and 2 with  $TO_1$  if  $(sf_1 - \tilde{S}_f)^2 + (sf_2 - \tilde{S}_f)^2 > (sf_1 + sf_2 - \tilde{S}_f)^2 \Leftrightarrow sf_1 \times sf_2 < \tilde{S}_f^2/2$ . These criteria are analogous for 1 and 3-simplex meshes (see Table 4.7).

$k$	$TO_1 \longleftarrow$	Criteria	$\longrightarrow TO_2$
1		$\longrightarrow l > \tilde{l}\sqrt{2}$ $\longleftarrow l_1 \times l_2 < \tilde{l}^2/2$	
2		$\longrightarrow sf > \tilde{S}_f\sqrt{2}$ $\longleftarrow sf_1 \times sf_2 < \tilde{S}_f^2/2$	
3		$\longrightarrow vc > \tilde{V}_c\sqrt{2}$ $\longleftarrow vc_1 \times vc_2 < \tilde{V}_c^2/2$	

Table 4.7: Improvement of the geometric quality through  $TO_1$  and  $TO_2$

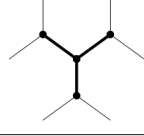
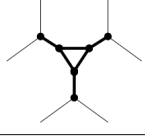
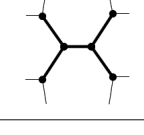
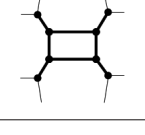
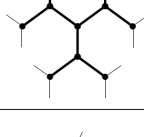
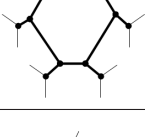
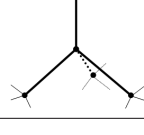
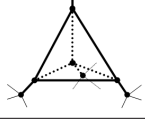
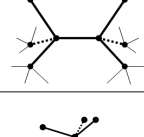
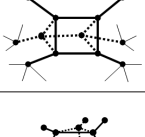
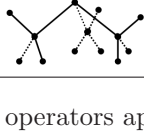
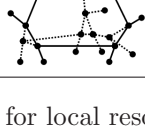
$k$	Order		Criteria	
2	1		$\rightarrow \sum_3 sf_i > 2\sqrt{3}\tilde{S}_f$ $\leftarrow \sum_4 sf_i < 2\sqrt{3}\tilde{S}_f$	
	2		$\rightarrow \sum_4 sf_i > 2\sqrt{5}\tilde{S}_f$ $\leftarrow \sum_5 sf_i < 2\sqrt{5}\tilde{S}_f$	
	3		$\rightarrow \sum_6 sf_i > \sqrt{42}\tilde{S}_f$ $\leftarrow \sum_7 sf_i < \sqrt{42}\tilde{S}_f$	
3	1		$\rightarrow \sum_4 vc_i > 2\sqrt{5}\tilde{V}_c$ $\leftarrow \sum_5 vc_i < 2\sqrt{5}\tilde{V}_c$	
	2		$\rightarrow \sum_5 vc_i > \sqrt{30}\tilde{V}_c$ $\leftarrow \sum_6 vc_i < \sqrt{30}\tilde{V}_c$	
	3		$\rightarrow \sum_8 vc_i > 6\sqrt{2}\tilde{V}_c$ $\leftarrow \sum_9 vc_i < 6\sqrt{2}\tilde{V}_c$	

Table 4.8: Different macro operators applied to neighbouring vertices for local resolution change

To change locally the resolution, we can also combine multiple  $TO_1$  and  $TO_2$  operations into macro operators. It is possible to replace a vertex with a new cell connecting its corresponding edges (equivalent to  $TO_2$  applied to adjacent edges). We can do the same by considering  $2^{nd}$  order neighbourhood (equivalent to  $TO_2$  applied to edges separated by one edge),  $3^{rd}$  order neighbourhood and so on. These operators are summarised in Table 4.8. The most interesting order is the order 3 since it leads to new faces/ cells close to hexagons/ dodecahedrons. Instead of taking this "vertex point of view", new macro operators can be defined through face/ cell subdivision (Table 4.9). They consist in inserting vertices on all edges of a face and connecting them through a new face. For  $k = 3$ , we perform this operation on all faces of a cell and we connect the new vertices through a new cell. This operator works for any size of face/ cell.

All these operations replace  $a$  elements (vertices, edges, faces or cells) with  $b$  elements inside a given domain of the space  $\Omega$ , that is a length, a surface or a volume (we have:  $\sum_a \Omega_i = \sum_b \Omega_i = \Omega$ ). The improvement criterion is  $\sum_b (\Omega_i - \tilde{\Omega})^2 < \sum_a (\Omega_i - \tilde{\Omega})^2$ . Supposing that elements are equally distributed in space (similar size), the criterion is reduced to  $b(\Omega/b - \tilde{\Omega})^2 < a(\Omega/a - \tilde{\Omega})^2$ . Expanding the expression, we finally obtain the following improvement criterion:

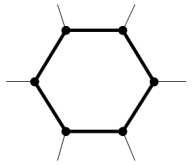
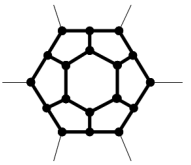
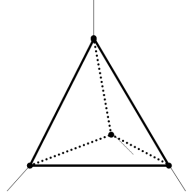
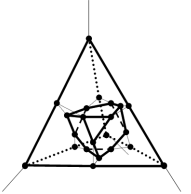
$k$		Criteria	
2		$\rightarrow \sum_1 sf_i > \tilde{S}_f \sqrt{n+1}$ $\leftarrow \sum_{n+1} sf_i < \tilde{S}_f \sqrt{n+1}$	
3		$\rightarrow \sum_1 vc_i > \tilde{V}_c \sqrt{n+1}$ $\leftarrow \sum_{n+1} vc_i < \tilde{V}_c \sqrt{n+1}$	

Table 4.9: Macro operators applied to face/ cells for local resolution change

$$\Omega < \tilde{\Omega} \sqrt{ab}, \text{ if } (b - a) > 0 \text{ (resolution decrease)}$$

$$\Omega > \tilde{\Omega} \sqrt{ab}, \text{ if } (b - a) < 0 \text{ (resolution increase)}$$

Criteria based of the surface/ volume are not sufficient to ensure a good geometric quality, as faces can stretch in one direction. Montagnat et al. [Mon99] propose to subdivide the face  $f$  of a 2-simplex mesh if  $\max(l_i)/\text{median}(l_i) > \tilde{e}$ , where  $\tilde{e}$  is a preset ratio and  $l_i$  the length of the  $i^{\text{th}}$  edge of the face. The problem is that faces can stretch without significant edge length deviation (see Table 4.10). We prefer to compute the 2 principal lengths  $L_1$  and  $L_2$  of the face and subdivide it along the  $2^{\text{nd}}$  direction if  $2(L_1/2 - L_2)^2 - (L_1 - L_2)^2 < 0 \leftrightarrow L_1 > L_2 \sqrt{2}$  (see Table 4.10). This is analogous for 1 and 3-simplex meshes.

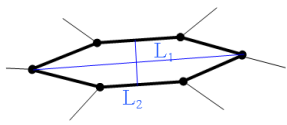
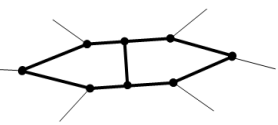
$k$		Criterion	$\rightarrow TO_2$
1, 2, 3		$\rightarrow L_1 > L_2 \sqrt{2}$	

Table 4.10: Increase of the resolution according to the elongation

When proceeding to a topological optimisation, we must define the criteria upon which the local refinement/decimation will take place. In other words, we need to establish the target geometric parameters (i.e. edge lengths, face surfaces and cell volumes). Because models will interact (collisions) and because their shape will change, we wish to have an equal repartition of vertices on the surface or volume. The target resolution is then defined according to a target edge length  $\tilde{l}$ . The most natural faces/ cells being the regular hexagon ( $k = 2$ ) and the regular dodecahedron ( $k = 3$ ), we calculate the target face surface  $\tilde{S}_f$  and cell volume  $\tilde{V}_c$  according to them (see Table 4.11). In addition, each vertex  $i$  is associated to a surface (resp. volume)  $S_i$  (resp.  $V_i$ ) for which their target values can be also calculated. At a given time, we can compute the target number of vertices  $\tilde{v}$  according to the current curve length  $L$ , surface  $S$  or volume  $V$ . Other criteria are

possible such as the curvature of the surface [Mon99]. Indeed, it is interesting to refine regions of highest curvature for some applications such as mesh compression or visualisation.


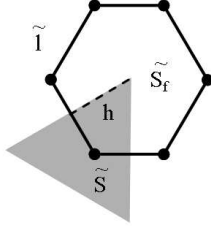
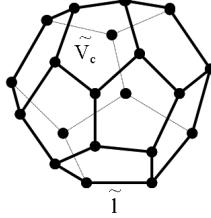
$k$		Target parameters
1		$\tilde{v} = \frac{L}{l}$
2		$\tilde{S} = \frac{3h\tilde{l}}{2} = \frac{3\sqrt{3}}{4}\tilde{l}^2$ $\tilde{S}_f = 2\tilde{S}$ $\tilde{v} = \frac{S}{\tilde{S}}$
3		$\tilde{V} = \frac{1}{20}(15 + 7\sqrt{5})\tilde{l}^3$ $\tilde{V}_c = 5\tilde{V}$ $\tilde{v} = \frac{V}{\tilde{V}}$

Table 4.11: Target parameters for the topological optimisation

#### 4.4.4 Global resolution change: multi-resolution schemes

Given a mesh, it is interesting to change its resolution globally in order to adapt the complexity of the system according to the complexity of the problem. In image registration, the global-to-local approach (or coarse-to-fine approach) is widely used since it increases robustness at a lower computational cost (see Section 3.3.4). With simplex meshes, it is possible to define different level of details through multi-resolution schemes that preserve the constant vertex connectivity. H. Delingette and J. Montagnat [Del94b] [Mon99] present a global multi-resolution scheme inspired from cell decimation (see Table 4.9), where the number of points is tripled for each resolution increase (method A of Table 4.12). The problem with this method is that shape features are not preserved because low resolution vertices are not conserved in the higher resolutions (curvature averaging). In addition, this scheme does not support the constraining of model vertices at specific locations, which is problematic when dealing with model attachments (e.g. bone/ tendon attachments). We propose a new multi-resolution scheme (method B of Table 4.12) that is based on the tessellation of the dual triangle mesh. Despite its sharper vertex/ face number increase, this scheme is superior since it better preserve shape features. Of course, the change in the number of vertices and faces respects the Euler-Poincaré rule of Table 4.1 ( $\Delta f - \Delta v/2 = 0$ ). When increasing the resolution, edge lengths are approximately divided by two (this is exact for regular hexagons).

These two schemes are systematic and very efficient in terms of computational speed because vertices are defined as a linear combination of the vertices of the lower/ higher resolution mesh. Table 4.12 states these combinations, where  $\mathbf{p}_i^r$  the position of the point  $i$  of the LOD  $r$  and  $\bar{\mathbf{p}}_l^r$  the barycenter of the face  $l$  of the LOD  $r$ . Method B is a bit more efficient since resolution decrease is immediate (memory copy); it does not require any computation.

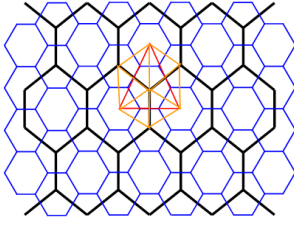
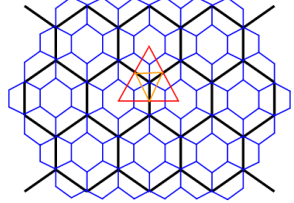
Method		Property changes	Linear combinations
A		$v^{r+1} = 3v^r$ $f^{r+1} = f^r + v^r$	$\mathbf{p}_i^{r+1} = (\mathbf{p}_j^r + \mathbf{p}_k^r + \overline{\mathbf{p}}_i^r)/3$ $\mathbf{p}_i^r = \overline{\mathbf{p}}_i^{r+1}$
B		$v^{r+1} = 4v^r$ $f^{r+1} = f^r + 3v^r/2$	$\mathbf{p}_i^{r+1} = (\mathbf{p}_j^r + \overline{\mathbf{p}}_i^r)/2$ or $\mathbf{p}_i^{r+1} = \mathbf{p}_i^r$ $\mathbf{p}_i^r = \mathbf{p}_i^{r+1}$

Table 4.12: Two multi-resolution schemes applied to a simplex surface (in black: resolution  $r$ ; in blue: resolution  $r + 1$ ) and the operation on the dual triangle mesh (in red: resolution  $r$ ; in orange: resolution  $r + 1$ )

Multi-resolution schemes works the same way for 1-simplex and 3-simplex meshes (same linear combinations). Figure 4.5 shows the application of method B to simplex volumes. Here, we have  $v^{r+1} = 5v^r$ ,  $f^{r+1} = 2f^r + 4v^r$  and  $c^{r+1} = c^r + f^r$ . The Euler-Poincaré rule remains valid ( $\Delta v + \Delta c - \Delta f = 0$ ).

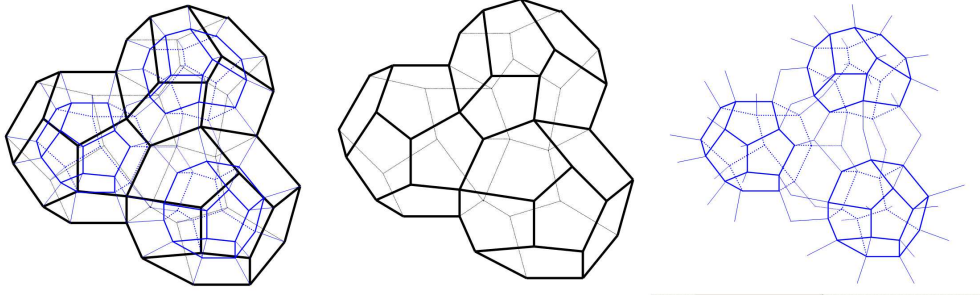


Figure 4.5: Multi-resolution scheme B applied to a 3-simplex mesh

#### 4.4.5 Results and discussion

In this section, we perform some mesh optimisation tests for  $k = 2$  and  $k = 3$  and evaluate the geometric and topological quality of resulting meshes.

**Surfacic optimisation** The first representative test consists in fitting a simplex surface to a given reference surface (femur), while optimising the mesh to obtain a regular vertex repartition. The femur has a particularly irregular shape compared to other organs. We have verified that the method operates similarly or better for other organs and resolutions. The fitting is based on smoothing, matching and volume preservation (set to 1.4 times the reference volume) forces. These forces and the evolution process will be detailed in the next sections. The straightforward matching of a simplex sphere, that is without any optimisation of

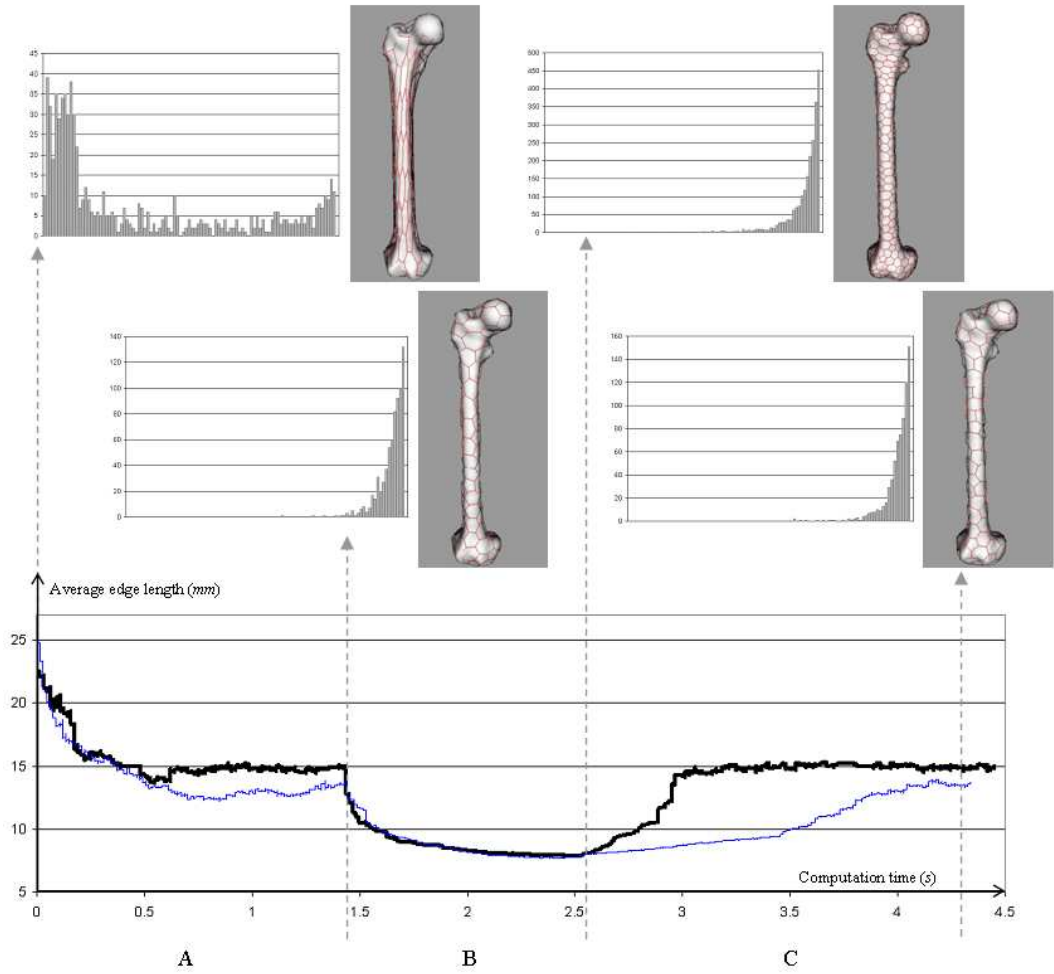
the mesh, leads to undesirable folding and overstretched faces as shown in Figure 4.6. Given a target edge length  $\tilde{l}$ , we subdivide/ merge faces using  $TO_1$  and  $TO_2$  according to the face surface and elongation criteria defined before. We also perform 1<sup>st</sup> and 2<sup>nd</sup> order exchanges (Table 4.5), 3<sup>rd</sup> vertex-based resolution changes (Table 4.8) and face-based resolution changes (Table 4.9) when they are valuable. We randomly select the vertex, edge or face to check and subdivide. The number of tests per iteration is limited in order to ensure stability (between two subdivisions, forces need to restore the mesh to a smooth configuration). We also limit acceptance criteria through a tolerance parameter (set to 1.1) that slightly increases lower sides of criterion equations. In order to analyse both resolution increase and decrease, we fix the target edge length to 15mm, to 8mm and then back to 15mm. Results are summarised in Table 4.13, and Video 4.1 . The table shows the average edge length in function of the computation time spent on mesh topological optimisation. We compare the use of  $TO_1$  and  $TO_2$  only (in blue) and the use of all operators (in black). In order to evaluate the quality of the resulting triangle mesh, the radius ratio metric is applied:  $2r/R$  where  $r$  is the radius of the inscribed circle and  $R$  the radius of the circumscribed circle of a given triangle. Using Heron’s formula, we have  $2r/R = 16s^2/(abc(a+b+c))$  ( $s$ : triangle area;  $a$ ,  $b$  and  $c$ : edge lengths). It is a well-accepted metric since it tends to 0 for degenerated triangles, and to 1 for regular triangles.



Figure 4.6: Left: The non-optimised femur model presents a poor topological and geometrical quality. Right: desired model with quasi-regular faces

These results show that our mesh optimisation scheme is effective since target parameters are reached with low deviations, in a computationally efficient way. Moreover, resulting triangle meshes possess quasi-regular elements (histograms close to 1). Operation frequencies (number of subdivisions/ number of tests) are an indicator of the usefulness of the defined operations. The 3<sup>rd</sup> order vertex-based subdivision scheme, we have introduced, is clearly valuable since it significantly reduces computation time for performing resolution adaptation. In addition, target parameters are better approximated (which is more significant for a small number of vertices) since it allows a more global evaluation than  $TO_1/ TO_2$ . Cell-based subdivision and 2<sup>nd</sup> order exchange are not very useful as shown by their low frequencies. We believe that the cell-based subdivision criterion is too tight because neighbouring cell should not be ignored due to vertex spreading.

**Volumetric optimisation** The second test consists in tetrahedralising a simplex surface using the adaptation of a 3-simplex mesh. As an initialisation, we duplicate the initial 2-simplex surface and link corresponding vertices to create a 3-simplex mesh (see Figure 4.7). Given a target edge length  $\tilde{l}$ , we subdivide/ merge cells using  $TO_1$  and  $TO_2$  according to the cell volume and elongation criteria defined before, we perform exchange operations on 4-vertices faces when they are valuable and we remove triangles as described above. Note that, when performing  $TO_2$ , cells are cut in half, according to their principal directions. We have found that the vertex-based and cell-based subdivision schemes are not valuable for 3-simplex meshes since they create/ remove a too high number of small faces (i.e. triangles). Contrary to 2-simplex meshes, new faces cannot be spread efficiently across cells through the exchange operation. We illustrate our results with two



	A	B	C
Target edge length/ vertex surface	15mm 292.3mm <sup>2</sup>	8mm 83.1mm <sup>2</sup>	15mm 292.3mm <sup>2</sup>
Frequ. of 1 / 2-exchanges	0.3%/0.0006% 0.4%/0.0005%	0.2%/0.0003% 0.1%/0.0003%	0.2%/0.001% 0.4%/0.0008%
Frequ. of TO2 / TO1	12%/24% 19%/38%	24%/7% 20%/4%	5%/14% 19%/68%
Frequ. of vertex-based increase / decrease	0.1%/15% —	20%/5% —	0.02%/8% —
Frequ. of cell-based increase / decrease	0%/0% —	1%/0% —	0%/0.5% —
Average edge length (std. dev.) in mm	15.0(1.7) 13.6(1.4)	7.9(1.1) 7.9(2.5)	14.93(1.92) 13.4(4.2)
Average vertex surface (std. dev.) in mm <sup>2</sup>	271.6(28.8) 191.8(31.4)	76.9(11.6) 75.5(9.7)	268.63(26.80) 217.3(33.5)

Table 4.13: Top: Images and radius ratio histograms at the different steps of the process. Middle: Average edge length variation (in blue: with  $TO_1$  and  $TO_2$ ; in black: with all operators). Bottom: Operation frequencies and resulting simplex mesh properties

representative examples: a sphere and a femur (Table 4.14 and Video 4.2 ). The tolerance parameter is set to 1.5 to prevent from instability.

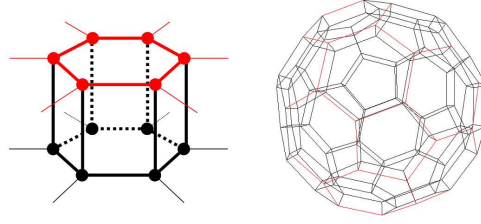


Figure 4.7: Automatic creation of a 3-simplex volume from a 2-simplex surface (in red) through surface duplication

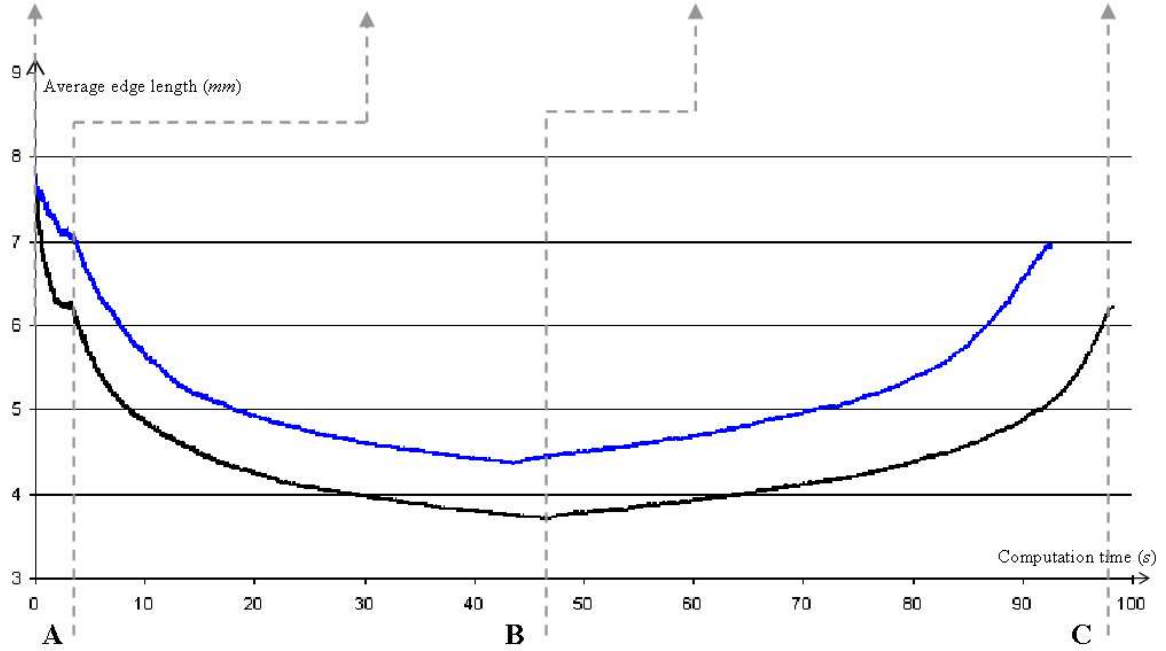
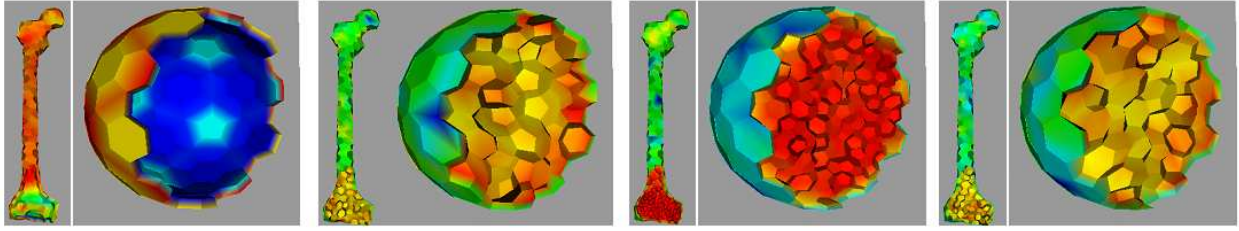
As expected, cells tends to dodecahedrons (12 pentagons). The resulting meshes have regular elements (little standard deviations). However, the method generally underestimate target values (like for 2-simplex meshes) because only  $TO_1/TO_2$  operations are used. The underestimation is worse than showed in the table since border elements are accounted (they have a fixed size, bigger than the target ones). The computation time is quite high compared to the surfacic case, and this could be prohibitive for large models or high resolutions.

After 3-simplex mesh adaptation, we tetrahedralise it in order to obtain models that can be further simulated with FEM software. As expected, the dual method leads to more homogeneous tetrahedra than the *cell partitioning* method of Section 4.3. In order to fit the dual tet mesh to the object at the boundaries, we lift border vertices by replacing border cell centers by border face centers. To minimise distortions, we also alter the weights of border vertices during adaptation (these weights will be defined in Section 4.6.1). We compare our method with a state-of-the-art tetrahedralisation method from Alliez and Desbrun [ACSYD05] (see Table 4.4.5). Similarly to triangles, we evaluate the quality of the resulting tetrahedral mesh using the radius ratio metric:  $3r/R$  where  $r$  is the radius of the inscribed sphere and  $R$  the radius of the circumscribed sphere. Let  $\{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}, \mathbf{e}, \mathbf{f}\}$  be the edge vectors of a given tetrahedron (where  $\{\mathbf{a}, \mathbf{f}\}$ ,  $\{\mathbf{b}, \mathbf{e}\}$  and  $\{\mathbf{c}, \mathbf{d}\}$  are the pairs of opposite edges). After reduction, we obtain:

$$\frac{3.r}{R} = \frac{[\mathbf{a}, \mathbf{b}, \mathbf{c}]^2}{(\|\mathbf{a} \wedge \mathbf{d}\| + \|\mathbf{a} \wedge \mathbf{e}\| + \|\mathbf{d} \wedge \mathbf{e}\| + \|\mathbf{b} \wedge \mathbf{c}\|) \cdot \sqrt{(cd + af + eb)(cd + af - eb)(af + eb - cd)(cd - af + eb)}}$$

Despite the generation of high quality meshes with almost regular elements, our method works a bit worse than the one of Alliez et. al (histogram closer to 1). This is mainly because our optimisation is local (the difference is exacerbated for high resolutions). We can point out that almost no degenerated tetrahedron is generated. Contrary to the CVT [DW03] technique that does not penalise degenerate tetrahedra (e.g. slivers), our method does because of the smoothing forces we use. They are based on barycentric coordinates weighted by neighbouring vertex volumes (see Section 4.6.1). The CVT tries to displace vertices towards the center of their Voronoi domain, but does not account for vertex occupancy (this would be equivalent of using the Laplacian smoothing forces of 4.6.1). Somehow, the way we displace vertices is closer to the technique of [ACSYD05], despite it is performed on the dual mesh. The advantage of our method is the flexibility: we can handle any kind of geometric constraints. For instance, boundaries can be fixed in advance and internal frontiers can be defined (e.g. separation between cortical and spongy bone). We can also specify the size of the elements by altering weights when computing smoothing forces. This is useful for simulating inhomogeneous materials.





	A	B	C
Target edge length/ vertex volume	7mm 525.7mm <sup>3</sup>	4mm 98.1mm <sup>3</sup>	7mm 525.7mm <sup>3</sup>
Frequ. of exchanges/ triangle removals	0.1%/0.03% 0.05%/0.01%	0.1%/0.02% 0.06%/0.01%	0.5%/0.06% 0.2%/0.03%
Frequ. of TO2 / TO1	4%/1% 2.2%/1%	13%/0.3% 11%/0.6%	0.2%/24% 0.9%/16%
Mean number of vertices per face (std. dev.)	5(0.7) 4.9(0.8)	5(0.6) 5(0.7)	5(0.7) 4.9(0.8)
Mean number of faces per cell (std. dev.)	11.7(2) 10.3(1.8)	13.1(1.1) 12.2(1.5)	11.7(2) 10.5(1.6)
Average edge length (std. dev.) in mm	6.2(1.5) 7.1(2)	3.7(1.2) 4.3(1.9)	6.2(1.5) 6.9(2)
Average vertex volume (std. dev.) in mm <sup>3</sup>	415(163) 533(278)	88(65) 134(134)	404(161) 503(272)

Table 4.14: Top: Images of cutted volumes at the different steps of the process (the color mapping shows vertex volumes; blue: maximum; red: minimum). Middle: Average edge length variation (in blue: femur example; in black: sphere example). Bottom: Operation frequencies and resulting simplex mesh properties

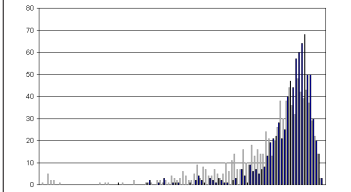
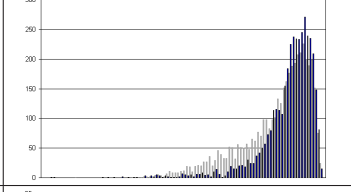
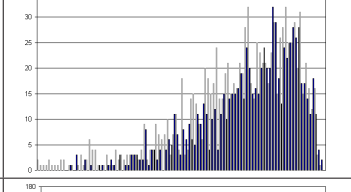
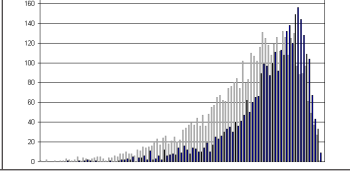
Models	Mean edge length (std. dev.) in <i>mm</i>		Radius ratio histograms
Sphere - low res ( $\sim 200$ vertices)	15.4(2.7)	15.7(2)	
Sphere - high res ( $\sim 800$ vertices)	9.2(2.1)	9.5(1.2)	
Femur - low res ( $\sim 300$ vertices)	15.9(3.7)	17.3(4.2)	
Femur - high res ( $\sim 900$ vertices)	10(3)	11.2(2.3)	

Table 4.15: Tetrahedralisation results on the different models. In grey: our method based on the dual mesh; In blue: the method from Alliez et al. [ACSYD05]

## 4.5 Geometry

### 4.5.1 Background

The constant vertex connectivity of simplex meshes leads to a simple local description of their geometry. The global three degrees of freedom of vertices can be turned into three local parameters. From these parameters, vertices are uniquely described with regards to their neighbours. It can be shown that, given a set of simplex parameters, a model is uniquely defined by means of a similitude transform (translation, rotation and isotropic scaling) [Del94b]. In other words, simplex parameters are invariants through translation, rotation and scaling. This is a nice property since geometric descriptors are likely to have it. We distinguish the metric and angular parameters: the linear combination of neighbours positions  $\mathbf{P}_i$  with weights  $\varepsilon_i$  (the *metric parameters*) defines the orthogonal projection  $\mathbf{P}_\perp$  onto the neighbourhood domain. *Angular parameters* (two parameters if  $k = 0$ , one if  $k = 2$ , zero if  $k = 3$ ) represents the elevation of  $\mathbf{P}$  above  $\mathbf{P}_\perp$ . Hence, they describe the curvature (and torsion) of the surface (curve). The elevation is performed along the normal vector  $\mathbf{n}$  that is obtained from the neighbours. Table 4.16 summarises the fundamental geometric definitions of simplex meshes stated by Delingette [Del94b].

### 4.5.2 Alternative definition for 2-simplex meshes

For 2-simplex meshes, the use of the simplex angle  $\phi$  is restrictive because  $\mathbf{P}$  is uniquely defined only if  $d \leq r$  (the projection of  $\mathbf{P}$  lies inside the circumscribed circle of its neighbours). Figure 4.8 illustrates that two different elevations  $h$  can be defined for a given simplex angle when  $d > r$ . Note that, expanding the inequality  $d \leq r$ , we get  $(\varepsilon_1 \varepsilon_2 \|\mathbf{P}_1 \mathbf{P}_2\| + \varepsilon_1 (1 - \varepsilon_1 - \varepsilon_2) \|\mathbf{P}_1 \mathbf{P}_3\| + \varepsilon_2 (1 - \varepsilon_1 - \varepsilon_2) \|\mathbf{P}_2 \mathbf{P}_3\|) \geq 0$ . This restriction is problematic since a lot of shapes can not be modelled using simplex parameters (e.g. the plane in Figure 4.1) due to this ambiguity. Moreover, during the deformation process, this leads to restrictions when applying forces or constraining vertex positions (e.g. attachments between objects, collision response, etc). A specific filtering algorithm is then necessary to insure that all  $\mathbf{P}_\perp$  lie inside their corresponding circumscribed circles. Such algorithm has not been described by Delingette and Montagnat. Nevertheless, to keep mesh flexibility, we have decided to use the parameter  $h$  directly, and more exactly the parameter  $h_n = h \cdot St^{-1/\alpha}$  where  $St$  is the area of the triangle formed by the neighbours and  $\alpha$  a parameter that tunes the scale invariant aspect. With  $\alpha = 2$ , the definition is similitude invariant; with  $\alpha = \infty$ , the definition is only invariant through rigid transforms. In Section 4.6.2, reference shape parameters are used to derive shape memory forces. For them, it is important to allow surface flexibility in terms of local scaling (to better account for inter-subject differences) but also to limit surface distortion. In practice, we take  $\alpha = 4$  which as proven to be an adequate trade-off. Note that, with our new definition, the computation of the simplex angle (thus the discrete curvature 4.5.3) is still valid (unique), despite it does not ensure reversibility. Our alternative definition is summarised in Equation 4.1.

$$\begin{aligned} \mathbf{P}(\varepsilon_1, \varepsilon_2, h_n) &= \varepsilon_1 \mathbf{P}_1 + \varepsilon_2 \mathbf{P}_2 + (1 - \varepsilon_1 - \varepsilon_2) \cdot \mathbf{P}_3 + h_n \cdot St^{1/\alpha} \cdot \mathbf{n} \\ \text{with } \mathbf{n} &= \frac{\mathbf{P}_1 \mathbf{P}_3 \wedge \mathbf{P}_1 \mathbf{P}_2}{\|\mathbf{P}_1 \mathbf{P}_3 \wedge \mathbf{P}_1 \mathbf{P}_2\|} \text{ and } St = \frac{\|\mathbf{P}_1 \mathbf{P}_3 \wedge \mathbf{P}_1 \mathbf{P}_2\|}{2} \end{aligned} \tag{4.1}$$

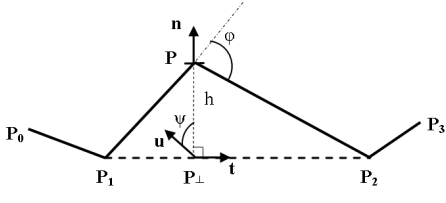
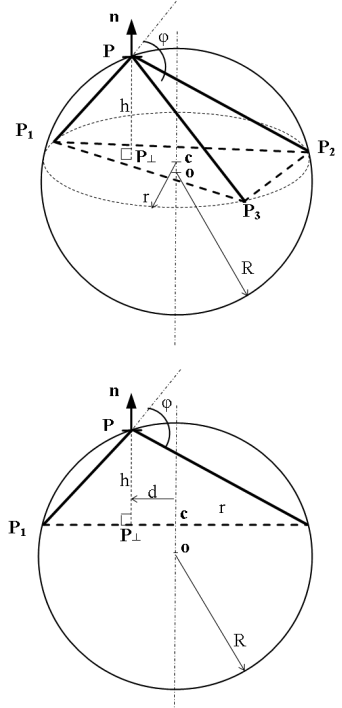
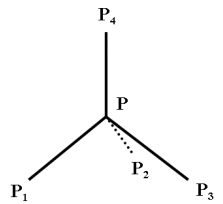
$k$	Local geometry	Geometric parameters
1		$\mathbf{P}(\varepsilon, \psi, \phi) = \varepsilon \mathbf{P}_1 + (1 - \varepsilon) \cdot \mathbf{P}_2 + h(\phi) \cdot \mathbf{n}(\psi)$ $\mathbf{n}(\psi) = \cos(\psi) \cdot \mathbf{u} + \sin(\psi) \cdot \mathbf{t} \wedge \mathbf{u}, \quad \psi \in [-\pi, \pi]$ $\mathbf{u} = \frac{[\mathbf{P}_0 \mathbf{P}_1, \mathbf{P}_1 \mathbf{P}_2, \mathbf{P}_2 \mathbf{P}_3]}{\ [\mathbf{P}_0 \mathbf{P}_1, \mathbf{P}_1 \mathbf{P}_2, \mathbf{P}_2 \mathbf{P}_3]\ }, \quad \mathbf{t} = \frac{\mathbf{P}_1 \mathbf{P}_2}{\ \mathbf{P}_1 \mathbf{P}_2\ }$ $h(\phi) = \frac{r(-1 + \varepsilon \sqrt{1 + 4\varepsilon_1(1 - \varepsilon_1) \tan^2(\phi)})}{\tan(\phi)}, \quad \phi \in [0, \pi]$ $r = \ \mathbf{P}_1 \mathbf{P}_2\ /2$ $\varepsilon = \begin{cases} 1, & \text{if } \phi <  \pi/2  \\ -1, & \text{if } \phi >  \pi/2  \end{cases}$
2		$\mathbf{P}(\varepsilon_1, \varepsilon_2, \phi) = \varepsilon_1 \mathbf{P}_1 + \varepsilon_2 \mathbf{P}_2 + (1 - \varepsilon_1 - \varepsilon_2) \cdot \mathbf{P}_3 + h(\phi) \cdot \mathbf{n}$ $\mathbf{n} = \frac{\mathbf{P}_1 \mathbf{P}_3 \wedge \mathbf{P}_1 \mathbf{P}_2}{\ \mathbf{P}_1 \mathbf{P}_3 \wedge \mathbf{P}_1 \mathbf{P}_2\ }$ $h(\phi) = \frac{(r^2 - d^2) \tan(\phi)}{r + \varepsilon \sqrt{r^2 + (r^2 - d^2) \tan^2(\phi)}}, \quad \phi \in [-\pi, \pi], \quad d \leq r$ $r = \ \mathbf{C} \mathbf{P}_1\ , \quad d = \ \mathbf{C} \mathbf{P}_\perp\ $ $\varepsilon = \begin{cases} 1, & \text{if } \phi <  \pi/2  \\ -1, & \text{if } \phi >  \pi/2  \end{cases}$
3		$\mathbf{P}(\varepsilon_1, \varepsilon_2, \varepsilon_3) = \varepsilon_1 \mathbf{P}_1 + \varepsilon_2 \mathbf{P}_2 + \varepsilon_3 \mathbf{P}_3 + (1 - \varepsilon_1 - \varepsilon_2 - \varepsilon_3) \cdot \mathbf{P}_4$

Table 4.16: Simplex meshes geometry

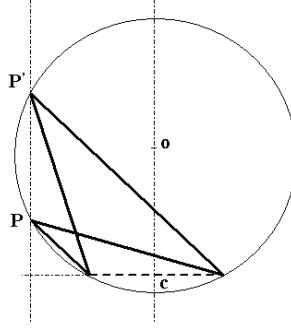


Figure 4.8: In some cases, the simplex angle is ambiguous for defining vertex positions

### 4.5.3 Curvature

The curvature is an important geometric descriptor, as it expresses object regularity through  $2^{nd}$  order computation. The mechanical analog is the elastic energy (its derivative is the bending energy), see Section 3.3.4. The curvature is intrinsically a continuous notion since it is the infinitesimal variation of the tangent vector. For surfaces, two principal curvatures can be extracted (maximal and minimal) by considering all tangent directions. It can be shown that these directions are orthogonal [dC76] (together with the normal vector, they constitute the *Darboux frame*). The mean curvature  $\kappa_H$  and the Gaussian curvature  $\kappa_G$  have been defined from the two principal curvatures (see Table 4.18). To distinguish convex and concave parts of surface, we generally sign the curvature according to the surface normal direction. For discrete models, the curvature is estimated, paying attention that it should converge to the continuous definition when mesh density increases. In [Del94a] [Del94b], H. Delingette defines a discrete curvature for simplex meshes. For triangle meshes, discrete differential operators have been defined through spatial averaging around vertices. Desbrun et al. [DMSB00] have proposed operators by considering 1-ring Voronoi regions to guarantee error bounds. These regions tile the surface without any overlap, so that the Gauss-Bonnet (continuous) theorem is satisfied. Definitions of the curvature are summarised in Tables 4.17 (curves) and 4.18 (surfaces).

Representations	Local geometry	Curvature definitions
Continuous curve		$\kappa = \kappa \mathbf{n} = \frac{\partial^2 \mathbf{C}}{\partial s^2} = \frac{\partial \mathbf{t}}{\partial s}$
Discrete curve		$\kappa = \frac{2 \cdot \sin(\phi) \mathbf{n}}{\ \mathbf{P}_1 \mathbf{P}_2\ }$

Table 4.17: Curvature definition for curves

For triangulating a simplex surface, we have proposed to insert face center vertices (Section 4.3). However,

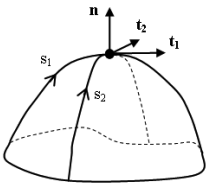
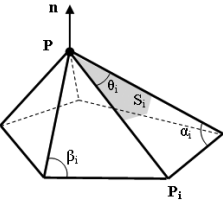
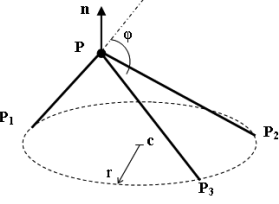
Representations	Local geometry	Curvature definitions
Continuous surface		$\kappa_1 = \frac{\partial t_1}{\partial s_1}, \quad \kappa_2 = \frac{\partial t_2}{\partial s_2}$ $\kappa_H = (\kappa_1 + \kappa_2)/2$ $\kappa_G = \kappa_1 \cdot \kappa_2$
Triangle surface		$\kappa_H = \kappa_H \mathbf{n} = \pm \sum (\cot \alpha_i + \cot \beta_i) (\mathbf{P} - \mathbf{P}_i) / (4 \cdot S)$ $\kappa_G = \kappa_G \mathbf{n} = (2\pi - \sum \theta_i) \mathbf{n} / S$ $S = \sum S_i = \sum (\cot \alpha_i + \cot \beta_i) \ \mathbf{P} - \mathbf{P}_i\ $
Simplex surface		$\kappa_H = \kappa_H \mathbf{n} = \frac{\sin \phi}{r} \mathbf{n} = \pm \frac{\mathbf{n}}{R}$ $\mathbf{n} = \frac{\mathbf{P}_1 \mathbf{P}_3 \wedge \mathbf{P}_1 \mathbf{P}_2}{\ \mathbf{P}_1 \mathbf{P}_3 \wedge \mathbf{P}_1 \mathbf{P}_2\ }$

Table 4.18: Curvature definition for surfaces

without any curvature continuity constraints, resulting meshes are low quality and difficult to exploit (see Figure 4.10, middle figure). We propose to adjust face centers according to the average neighbour curvature, in an iterative fashion (because changing their position changes this average curvature). Given a face center  $\mathbf{C}$  of curvature  $\kappa$ , its  $N$  neighbours  $\mathbf{P}_i$  of curvature  $\kappa_i$  and normal  $\mathbf{n}_i$  (see Figure 4.9), we calculate the average curvature  $\bar{\kappa} = \sum \kappa_i / N$  and average normal  $\mathbf{n} = \sum \mathbf{n}_i / \|\sum \mathbf{n}_i\|$ . At each iteration, we move  $\mathbf{C}$  in the direction of  $\mathbf{n}$  if  $\bar{\kappa} > \kappa$  (resp. in the direction of  $-\mathbf{n}$  if  $\bar{\kappa} < \kappa$ ) and we adapt (divide by two) the amount of displacement  $d$  when the direction changes. The process quickly converges as shown in Figure 4.10, despite oscillations that are produced by too large displacement steps (in the example, steps are initialised to  $d = 1mm$ ). We have checked that using the Gaussian and the mean curvature gives similar results.

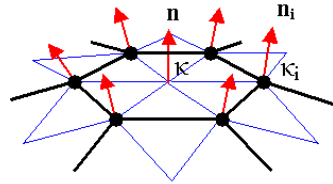


Figure 4.9: Face center adaptation for curvature averaging

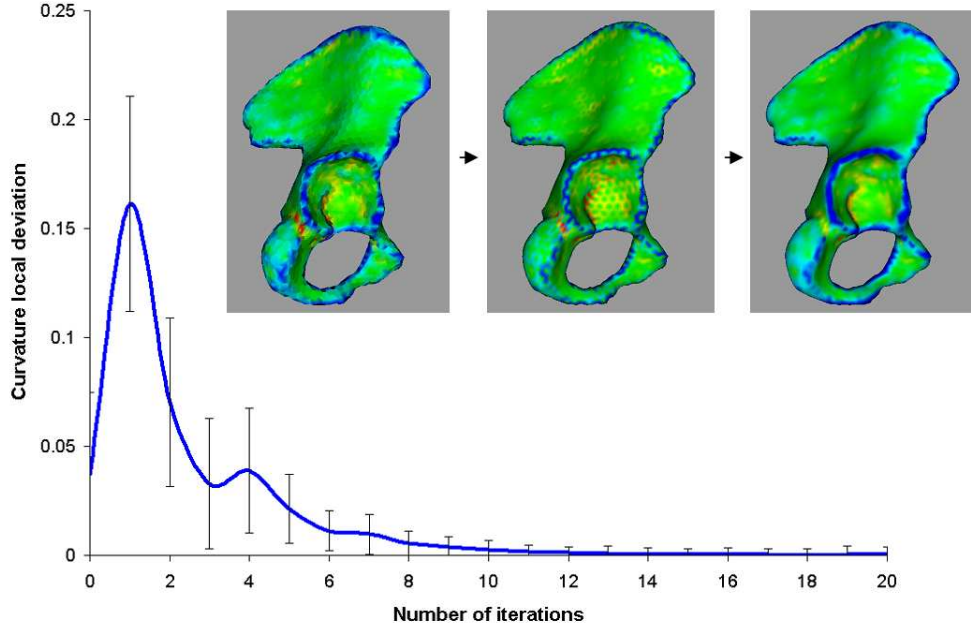


Figure 4.10: Plot of the curvature deviation: the curvature smoothing process converges in a few iterations. Pictures show the mean curvature (blue: max, red, min) of the simplex mesh, the triangulated mesh before smoothing and after smoothing

## 4.6 Internal constraints

As already discussed in Section 3.3.4, deformable models need intrinsic forces to ensure shape continuity against external noise (e.g. image forces, collisions, etc.). In this Section, we will examine different formulations of these forces for simplex meshes. Note that, contrary to a variational approach where a global energy is minimised, we have chosen a dynamic evolution (Section 3.3.5) to derive model deformation from forces, which is more suited for discrete models. We will see that, due to their simple local geometric description, forces will take simple and computationally efficient formulations. We will particularly be interested in smoothing, shape memory, volume conservation and radial forces, since they are relevant in the context of medical image segmentation and anatomical modelling. Forces acting on a given vertex  $\mathbf{P}$  will be expressed according to its target position  $\tilde{\mathbf{P}}$  as we will see in Section 4.8. For the time being, we will only derive the expression of  $\tilde{\mathbf{P}}$ .

### 4.6.1 Smoothing

Smoothing is a soft constraint that is important in medical image segmentation. Indeed, to face noise, the first step is to exploit the fact that biological tissues have a smooth shape. Using this prior knowledge, we can derive forces that enforce 1<sup>st</sup> or 2<sup>nd</sup> order shape continuity. 1<sup>st</sup> order elastic forces aim at minimising the curvature. The popular Laplacian smoothing attracts vertices towards the barycenter of their neighbours:  $\tilde{\mathbf{P}} = \langle \mathbf{P}_i \rangle_\eta$  where the  $\langle \cdot \rangle_\eta$  operator denotes local averaging in the neighbourhood  $\eta$  of  $\mathbf{P}$  (note that 2<sup>nd</sup> or 3<sup>rd</sup> order neighbours can be used in order to give a more global effect to the smoothing). Its tangential component is appropriate because it enforces a regular spreading of the vertices over the entire length/surface. However its normal component produces undesirable effect for curves and surfaces since it globally reduces the length/surface of the model. To overcome this, several authors [GSZW98] [PMTK01] have proposed to replace the normal component by a 2<sup>nd</sup> order term that averages the curvature (bending force). In the context of simplex meshes, Montagnat et al.[MD05] uses a local average of the simplex angle to

compute the goal elevation  $\tilde{h} = h(\langle \phi_i \rangle_\eta)$ . However, as explained previously in 4.5.2, simplex angles are ambiguous for certain configurations. We prefer to use a local average of the elevation  $\tilde{h} = \langle h_i \rangle_\eta$  which will produce a comparable effect.

The Laplacian smoothing takes neighbour centers in order to locally distribute vertices evenly. However, globally, it does not necessarily produce regularly spaced vertices as it does not account for vertex occupancies. This is particularly obvious when faces and cells are irregular (see Figure 4.11). In fact, the center is a particular case of the barycenter for which all weights are equal. Here, we propose to use different weights in order to average the size of the domains associated to vertices (rather than their spatial positions). Let be  $\Omega_i$  the size of the domain covered by a vertex  $\mathbf{P}_i$  (by contraction we denote it as vertex length  $l_i$  (for  $k = 1$ ), vertex surface  $S_i$  ( $k = 2$ ) or vertex volume  $V_i$  ( $k = 3$ )). Note that these domains tile model space (for instance, the total surface of a 2-simplex mesh is equal to  $\Omega = \sum_i \Omega_i = \sum_i S_i$ ). Now, we can introduce the new target position for  $\mathbf{P}$  as the barycenter of its neighbours, weighted by the size of their associated domains:  $\tilde{\mathbf{P}} = \sum_\eta \Omega_i \mathbf{P}_i / \sum_\eta \Omega_i$ . As a result, vertices will be attracted by the regions of highest size. This technique is more global than the Laplacian method since it takes into account neighbours of higher order. Moreover it prevents from element degeneration (e.g. infinitesimally small faces, sliver tetrahedra, etc.). In [ACSYD05] Alliez et al. turns up to a similar expression, from a variational approach, to optimise vertex placement in tetrahedral mesh generation (generalised barycenter).

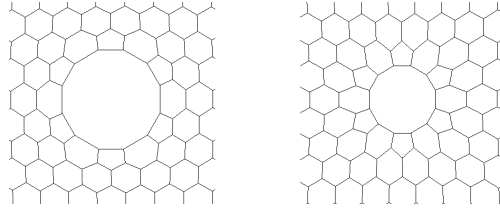


Figure 4.11: Comparison of *Laplacian* (left) and *weighted barycenter* (right) constraints on an irregular 2-simplex mesh

Like for Laplacian smoothing, we have to discard the normal component of the weighted barycenter to prevent from global shrinking. For simplicity, we restrict the neighbouring domain to the order 1 to compute the tangential component, so that no extra computation is required (i.e. vector decomposition). However, we use neighbours up to the  $2^{nd}$  order for the normal component (bending). In practice, this has proven to be adequate in terms of computational time/regularisation power. We summarise the smoothing constraints to be applied to simplex meshes in Table 4.19.

$k$	Laplacian smoothing [Del94b] [MD05]	Weighted barycenter
1	$\tilde{\mathbf{P}} = \frac{\mathbf{P}_1 + \mathbf{P}_2}{2} + h(\langle \phi_i \rangle_2) \mathbf{n}(\langle \psi_i \rangle_2)$	$\tilde{\mathbf{P}} = \frac{l_1 \mathbf{P}_1 + l_2 \mathbf{P}_2}{l_1 + l_2} + h(\langle \phi_i \rangle_2) \mathbf{n}(\langle \psi_i \rangle_2)$
2	$\tilde{\mathbf{P}} = \frac{\mathbf{P}_1 + \mathbf{P}_2 + \mathbf{P}_3}{3} + \langle h_i \rangle_2 \mathbf{n}$	$\tilde{\mathbf{P}} = \frac{S_1 \mathbf{P}_1 + S_2 \mathbf{P}_2 + S_3 \mathbf{P}_3}{S_1 + S_2 + S_3} + \langle h_i \rangle_2 \mathbf{n}$
3	$\tilde{\mathbf{P}} = \frac{\mathbf{P}_1 + \mathbf{P}_2 + \mathbf{P}_3 + \mathbf{P}_4}{4}$	$\tilde{\mathbf{P}} = \frac{V_1 \mathbf{P}_1 + V_2 \mathbf{P}_2 + V_3 \mathbf{P}_3 + V_4 \mathbf{P}_4}{V_1 + V_2 + V_3 + V_4}$

Table 4.19: Smoothing constraints



## 4.6.2 Shape memory

With simplex meshes, it is possible to simply enforce hard constraints on shapes by predefining their simplex parameters (see Table 4.16) such as in [Del94b] [MD05]. Indeed, we know that organs have quasi-similar shapes across individuals, so it is particularly effective to use a reference model (prior shape information through the reference shape parameters  $\tilde{\varepsilon}_i$ ,  $\tilde{\psi}$ ,  $\tilde{\phi}$  and  $\tilde{h}_n$ ) during segmentation. Shape memory constraints are consequently suited for deformable segmentation and registration. The resulting target vertex positions are summarised in Table 4.20:

$k$	Shape memory
1	$\tilde{\mathbf{P}}(\tilde{\varepsilon}, \tilde{\psi}, \tilde{\phi}) = \tilde{\varepsilon}\mathbf{P}_1 + (1 - \tilde{\varepsilon})\cdot\mathbf{P}_2 + h(\tilde{\phi})\cdot\mathbf{n}(\tilde{\psi})$
2	$\tilde{\mathbf{P}}(\tilde{\varepsilon}_1, \tilde{\varepsilon}_2, \tilde{h}_n) = \tilde{\varepsilon}_1\mathbf{P}_1 + \tilde{\varepsilon}_2\mathbf{P}_2 + (1 - \tilde{\varepsilon}_1 - \tilde{\varepsilon}_2)\cdot\mathbf{P}_3 + \tilde{h}_n\cdot St^{1/4}\cdot\mathbf{n}$
3	$\tilde{\mathbf{P}}(\tilde{\varepsilon}_1, \tilde{\varepsilon}_2, \tilde{\varepsilon}_3) = \tilde{\varepsilon}_1\mathbf{P}_1 + \tilde{\varepsilon}_2\mathbf{P}_2 + \tilde{\varepsilon}_3\mathbf{P}_3 + (1 - \tilde{\varepsilon}_1 - \tilde{\varepsilon}_2 - \tilde{\varepsilon}_3)\cdot\mathbf{P}_4$

Table 4.20: Shape memory constraints

We have seen in the Section 4.5 on geometry that a set of simplex parameters defines a unique shape up to a similarity transformation. This scale, translation and rotation invariance properties of simplex parameters are worth as the absolute position, orientation and scale of the structure to be segmented are not known in principle. Enforcing shape memory constraints, the resulting segmented model will have smooth differences with the reference model, in terms of scale of the elements constituted by neighbouring vertices.

## 4.6.3 Volume preservation

Most biological tissues are nearly incompressible. This is a useful information for constraining shapes during intra-patient registration (e.g. motion tracking). Note that this is not necessarily worth for longitudinal studies where the goal is precisely to quantify volume changes. Volume conservation is applied to closed simplex surfaces through normal forces (in the case of simplex volumes, forces are applied to the boundary, so on a simplex surface). We perform volume computation of a triangulated simplex surface by applying the divergence (or Gauss') theorem. Let  $\mathbf{F}(x, y, z) = z$  be a vector field,  $\sum V$  the total volume of the domain  $\Omega_V$  of boundary  $\Omega_S$  and  $A_i$  the surface of the  $x - y$  projection of the triangle  $i$  formed by the vertices  $\mathbf{P}_1^i = [x_1^i, y_1^i, z_1^i]^T$ ,  $\mathbf{P}_2^i = [x_2^i, y_2^i, z_2^i]^T$  and  $\mathbf{P}_3^i = [x_3^i, y_3^i, z_3^i]^T$  ( $\{\mathbf{P}_1^i, \mathbf{P}_2^i, \mathbf{P}_3^i\}$  being oriented counter clock wise on the surface). We have:

$$\begin{aligned} \sum V &= \int \int \int_{\Omega_V} dV = \int \int \int_{\Omega_V} \text{div}(\mathbf{F})dV = \int \int_{\Omega_S} \mathbf{F}\cdot d\mathbf{S} = \int \int_{\Omega_S} z\cdot\mathbf{P}_1^i\mathbf{P}_3^i \wedge \mathbf{P}_1^i\mathbf{P}_2^i = \sum_i A_i\bar{z} \\ &= \sum_i \frac{z_1^i + z_2^i + z_3^i}{3} ((x_3^i - x_1^i)(y_2^i - y_1^i) - (x_2^i - x_1^i)(y_3^i - y_1^i)) \end{aligned}$$

This method is very fast since we use simplified triangle meshes (they are not filtered by the method presented in Section 4.5.3): face centers are simply added and linked to simplex vertices to form oriented triangles. The resulting triangle mesh is already a good approximation of the volume. Indeed, a fast method is mandatory because volumes need to be computed at each iteration. This method is sensitive to self-collisions (inverted models will have negative volumes), but this is not prohibitive because smoothing/ shape memory forces will prevent from remnant self-collisions. Nevertheless, this may be applied to 3-simplex meshes for checking inverted cells (a 3-simplex cell is a 2-simplex mesh). Now, we have to derive the goal position of a vertex  $\tilde{\mathbf{P}}$ ,

given a target volume  $\sum \tilde{V}$ , a current volume  $\sum V$  and a current surface  $\sum S$ . The assumption of a similar displacement  $\Delta h$  for all vertices along the normal yields to  $\Delta h = (\sum \tilde{V} - \sum V) / \sum S$ , whereas the assumption of a similar volume variation contribution for all vertices yields to  $\Delta h_i = \Delta V / S = (\sum \tilde{V} - \sum V) / (S.v)$  ( $v$ : total number of vertices;  $S$ : surface associated to  $\mathbf{P}$ ). The resulting target positions are given in Table 4.21. The constant displacement method is in practice more efficient because it produces less distortions than the other technique. These distortions are not necessarily recovered by shape memory forces because they are scale invariant.

$k$	Constant displacement	Constant volume variation
2	$\tilde{\mathbf{P}} = \mathbf{P} + \frac{(\sum \tilde{V} - \sum V)}{\sum S} \mathbf{n}$	$\tilde{\mathbf{P}} = \mathbf{P} + \frac{(\sum \tilde{V} - \sum V)}{S.v} \mathbf{n}$

Table 4.21: Volume preservation constraints

#### 4.6.4 Radial constraints

Soft-tissues of the musculoskeletal system are, most of the time, modelled and simulated using 1D or 2D medial representations accounting for both position and thickness. Indeed, muscles generally have a smooth and tubular shape, that can be represented by an underlying piece-wise action line [ND98] [AT01] [TSB<sup>+</sup>05], where isotonic contractions are modelled through action line shrinking/ stretching, and isometric contractions through radial constraints applied to a wrapped surface (Section 2.5). However, muscles with large attachment areas and/ or several origins/ insertions require many action lines [NTH00] [AT01]. Ligaments are thin tissues effectively represented by FEM shells [WGL<sup>+</sup>05], multiple lines being not accurate to assess load transfer [BH91] (Section 2.7.2). Cartilages are mostly analysed through their thickness [WTGW03] (Section 2.6.2). Looking at the different approaches presented in the literature, we see that authors use medial representations both to abstract objects (to decrease the number of parameters to represent them) and reversely, to constrain them (functional aspect of medial representations). However, representations that are used are *not* reversible, meaning that they are not able to capture the shape and to replicate the function accurately. Moreover, their construction is subjected to user interaction (no unique representation). On the contrary, the medial axis, in the geometric sense, is unique and is able to reconstruct exactly any object. The medial axis transform (MAT) has been introduced in 1964 by H. Blum as the computation of maximal balls inside on object [Blu64]. The medial axis is the union of these balls (center and radius). Three main properties characterise the medial axis:

- **Homotopy equivalence:** An object and its medial axis are connected the same way.
- **Good localisation:** each point of the medial axis is equidistant from at least two points of the surface.
- **Reversibility:** An object is reconstructed exactly by the union of maximal balls of its medial axis.

In this section, we propose a new method to compute medial surfaces of musculoskeletal tissues automatically, which will better obey to these properties than previous methods. It leads to enhanced mechanisms for constraining surfaces (continuous constraints) and approximating/ characterising shapes. The medial axis transform is not straightforward to compute and several methods have been presented for approximating it, based on Voronoi diagrams [ACK01], on distance maps [Bor84], or on thinning (see [ABE07] for a complete review of these methods). In fact medial axis are very sensitive to the smoothness of the surface (instability of the medial axis transform): small bumps create new branches and spikes on the medial axis (see Figure 4.15). This is quite undesirable as the goal is precisely to have a simpler representation of an object thanks to its medial axis. To tackle this well known problem for noisy surfaces, two general approaches have been explored to create minimal medial axis:

- **Pruning** [ACK01] [DGB03] [ABE07]: An accurate (complex) medial axis is iteratively simplified by suppressing redundant or irrelevant parts. The resulting medial axis is generally no more homotopy equivalent.
- **Shape constraints** [TWK87] [PFJ<sup>+</sup>03] [FVMO04]: A medial axis with a predefined (simple) shape is iteratively fitted to the center of the object. Extra parameters may be added to the medial axis (excursion parameters).

While the first approach makes no assumptions about the final structure of the medial axis, the second starts from a "desired" shape and is therefore less demanding. For muscles, ligaments and cartilages, we can make the hypothesis that medial sheets will be homeomorphic to planes (no branching); and this will be corroborated by the results. We will consequently focus on the second approach. In [TWK87] [PFJ<sup>+</sup>03], medial curves and planes (M-reps) are used to represent complex objects. In addition to radius parameters, they add shape perturbations to better approximate objects. We will show that such parameters are not useful for our purpose since our objects are much more regular

To approximate its true medial axis, our goal is to iteratively fit a plane (*the medial axis*) to the center of an object (*the model*) through forces. These forces aim at minimising the distance between the reconstructed surface from the medial axis and the model. Reciprocally, forces will be applied to the model in order to constrain it radially. The model is a closed 2-simplex surface without any hole ( $g = 0$  and  $H = 0$  in the Euler-Poincaré formula). The medial axis is a 2-simplex surface with genus zero and one hole ( $g = 0$  and  $H = 1$ ). A radius is associated to each one of its vertices, the reconstructed surface being the boundary of the union of the resulting balls. Because our medial axis is discrete, we linearly interpolate the radii along its surface in order to reconstruct a smooth and continuous model (see Figure 4.12). We have previously shown in [DGB03] that radius interpolation is suited and efficient for representing objects with a minimal number of medial points.

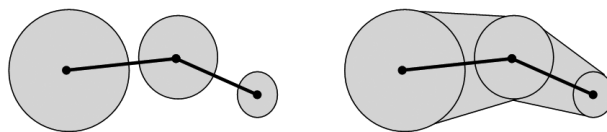


Figure 4.12: A 2D illustration of the discrete medial axis and the reconstructed model (in grey) without radius interpolation (left) and with radius interpolation (right)

The goal is to reach a state where model vertices  $\mathbf{P}_i$  ( $i$  indexes model vertices) lie on the surface of the reconstructed model (see Figure 4.13). In other words, we want that the distance between  $\mathbf{P}_i$  and its orthogonal projection on the medial axis  $\mathbf{P}_{i\perp}$  corresponds to the interpolated radius  $r_i$ . Let  $\mathbf{Q}_j$  be the medial axis vertices of radius  $R_j$  ( $j$  indexes medial vertices) and  $w_{ij}$  the barycentric coordinates of the projection of  $\mathbf{P}_i$ . We have  $\mathbf{P}_{i\perp} = \sum_j w_{ij} \cdot \mathbf{Q}_j$  and  $r_i = \sum_j w_{ij} \cdot R_j$ .

In a deformed state, we want to displace medial axis vertices or model vertices to reach the desired state. The error can be calculated by  $err = \langle |r_i - P_i P_{i\perp}| \rangle_i$  (the  $\langle . \rangle_i$  operator denotes averaging over all points). If a desired reference state has been defined before, projections and radii ( $w_{ij}$  and  $R_j$ ) are known; otherwise, weights have to be recomputed at each iteration by projecting  $\mathbf{P}_i$  orthogonally, and radii need to be approximated. We have tested different methods for estimating radii, minimising the error:

- **Closest point:**  $R_j = Q_j P_i^*$  where  $\mathbf{P}_i^*$  is the closest model vertex to  $\mathbf{Q}_j$ .
- **Weighted mean:**  $R_j = \sum_i w_{ij} \cdot P_i P_{i\perp} / \sum_i w_{ij}$ .

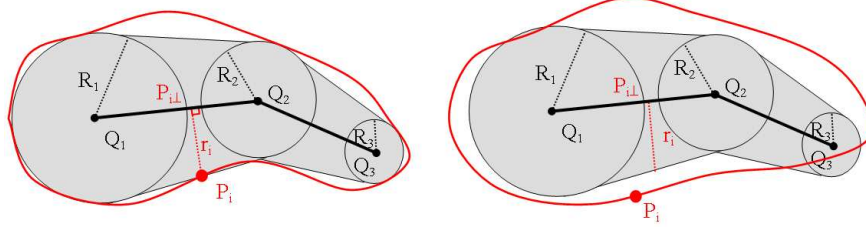


Figure 4.13: A 2D illustration of the desired state (left) and deformed state (right). In red, the model; in black, the medial axis

- **Iterative minimisation of the error:** For a random vertex  $\mathbf{Q}_j$ , we compute the derivative of the error  $\partial err / \partial R_j = \sum_i \text{sign}(r_i - P_i P_{i\perp}) w_{ij}$ . If it is positive, we decrease  $R_j$  by an infinitesimal value  $\varepsilon$ , otherwise we increase  $R_j$  by  $\varepsilon$ . We iterate the process until convergence.

In practice, we use the *weighted mean* method which gives very close results to the optimal ones (*iterative minimisation* technique) without excessive computational costs. The closest point technique was found to be too inaccurate. Now, given that all  $w_{ij}$  and  $R_j$  are known, we have to derive the desired positions  $\tilde{\mathbf{Q}}_j$  of  $\mathbf{Q}_j$  (approximation of the true medial axis) and/ or  $\tilde{\mathbf{P}}_i$  of  $\mathbf{P}_i$  (radial regularisation of the model) that minimise the error. Medial axis vertex positions  $\tilde{\mathbf{Q}}_j$  are, in fact, defined indirectly through the desired displacement of  $\mathbf{P}_{i\perp}$ :

$$\delta \mathbf{P}_{i\perp} = \tilde{\mathbf{P}}_{i\perp} - \mathbf{P}_{i\perp} = \sum_j w_{ij} \cdot (\tilde{\mathbf{Q}}_j - \mathbf{Q}_j) = \sum_j w_{ij} \cdot \delta \mathbf{Q}_j$$

We will see in Section 4.8 that when applying forces on the set of particles  $\mathbf{Q}_j$ , we must comply with the momentum conservation principle. Assuming that all forces  $\mathbf{F}_j$  are collinear and proportional to  $\mathbf{P}_{i\perp} \tilde{\mathbf{P}}_{i\perp}$ , we get  $\delta \mathbf{Q}_j = w_{ij} \mathbf{P}_{i\perp} / \sum_j w_{ij}^2$ . This is true if we consider only one projection. However, since each  $\mathbf{Q}_j$  shares multiple  $\mathbf{P}_{i\perp}$ , we average their contribution. The final formula for calculating the desired medial vertex positions is thus:

$$\tilde{\mathbf{Q}}_j = \mathbf{Q}_j + \langle w_{ij} \mathbf{P}_{i\perp} \tilde{\mathbf{P}}_{i\perp} / \sum_j w_{ij}^2 \rangle_i$$

We have investigated several methods for determining desired displacements that are summarised in Table 4.22. The *Closest point* method is based on the straightforward projection on medial spheres, without radius interpolation. Contrary to the two other methods, this leads to undesirable distortions of the surfaces as well as inaccurate medial axis localisation and object reconstruction (see Figure 4.15). With the *radius* method, the goal is to seek interpolated reference radii; whereas with the *radius and barycentric coordinates* method, the goal is to seek reference radii and positions. The last method does not allow any sliding of model vertices along the reconstructed surface. For model boundaries, because the direction of the projection is not normal, this method is not applicable and the *radius* method must be used. We could remedy this through angular parameters, as used by Pizer et al. [PFJ<sup>+</sup>03]. However, we have experienced that smoothing forces favourably prevents from excessive sliding (without requiring extra parameters). Note that these two methods are equivalent when no reference state is available, since orthogonal projections are performed at each time-step. When using a reference state, the *radius and barycentric coordinates* method requires an extra parameter  $\lambda_i$  for each model vertex: the side with regards to the medial surface normal on which they are projected. This side determines the sign in the expressions of Table 4.22. However, in

Method	Illustrations	Medial axis constraints	Model constraints
Closest point		$\tilde{\mathbf{Q}}_j = \mathbf{P}_i^* + R_j \frac{\mathbf{P}_i^* \mathbf{Q}_j}{\ \mathbf{P}_i^* \mathbf{Q}_j\ }$ $\mathbf{P}_i^*: \text{closest point to } \mathbf{Q}_j.$	$\tilde{\mathbf{P}}_i = \mathbf{Q}_j^* + R_j \frac{\mathbf{Q}_j^* \mathbf{P}_i}{\ \mathbf{Q}_j^* \mathbf{P}_i\ }$ $\mathbf{P}_i^*: \text{closest point to } \mathbf{Q}_j$
Radius		$\tilde{\mathbf{Q}}_j = \mathbf{Q}_j + \langle w_{ij} \frac{\mathbf{P}_{i\perp} \mathbf{P}_{i\perp}}{\sum_j w_{ij}^2} \rangle_i$ $= \mathbf{Q}_j + \langle w_{ij} \frac{\mathbf{P}_{i\perp} \mathbf{P}_i + r_i \frac{\mathbf{P}_i \mathbf{P}_{i\perp}}{\ \mathbf{P}_i \mathbf{P}_{i\perp}\ }}{\sum_j w_{ij}^2} \rangle_i$	$\tilde{\mathbf{P}}_i = \mathbf{P}_{i\perp} + r_i \frac{\mathbf{P}_{i\perp} \mathbf{P}_i}{\ \mathbf{P}_{i\perp} \mathbf{P}_i\ }$
Radius and barycentric coordinates		$\tilde{\mathbf{Q}}_j = \mathbf{Q}_j + \langle w_{ij} \frac{\mathbf{P}_{i\perp} \mathbf{P}_{i\perp}}{\sum_j w_{ij}^2} \rangle_i$ $= \mathbf{Q}_j + \langle w_{ij} \frac{\mathbf{P}_{i\perp} \mathbf{P}_i + \lambda_i r_i \mathbf{n}_{i\perp}}{\sum_j w_{ij}^2} \rangle_i$	$\tilde{\mathbf{P}}_i = \mathbf{P}_{i\perp} + \lambda_i r_i \mathbf{n}_{i\perp}$

Table 4.22: Radial constraints

With:

- $\mathbf{P}_{i\perp} = \sum_j w_{ij} \cdot \mathbf{Q}_j$
- $\mathbf{n}_{i\perp} = \sum_j w_{ij} \cdot \mathbf{n}_j / \|\sum_j w_{ij} \cdot \mathbf{n}_j\|$  where  $\mathbf{n}_j$  is the normal at  $\mathbf{Q}_j$ .
- $r_i = \sum_j w_{ij} \cdot R_j$
- $\lambda_i = \text{sign}(\mathbf{P}_{i\perp} \cdot \mathbf{P}_i \cdot \mathbf{n}_{i\perp})$  in the reference configuration (*side* parameter).

practice, this parameter is also used in the *radius* method. The reason is that we experienced some flipping when radii are small, even if reference radii are reached (see Figure 4.14). Using the *side* parameter, we can detect those undesirable auto-intersections and remove them through reflection with regards to the medial axis. The reflection is performed by:  $\delta \tilde{\mathbf{P}}_i = 2(\mathbf{P}_{i\perp} \cdot \mathbf{P}_i \cdot \mathbf{n}_{i\perp}) \mathbf{n}_{i\perp}$  (note that this is also valid for boundary vertices).



Figure 4.14: Auto-intersections happen when not using the *side* parameter

We will see that the application of the technique to the musculoskeletal system (Section 5.4.3) is valuable, since organ shapes can be well and quickly approximated by the medial axis (*err*  $\simeq 0.6\text{mm}$ ). The localisation is good because no extra excursion or perturbation parameter (that would bump the surface) is necessary, and our technique preserves homotopy (a plane is homotopic to a closed surface). We will test more in details the two methods we have presented. Their main difference lies in the way to constrain the overlying model. We will see if smoothing forces can recover the shape (without sliding). This would mean that barycentric

coordinates of projected model vertices do not need to be enforced.

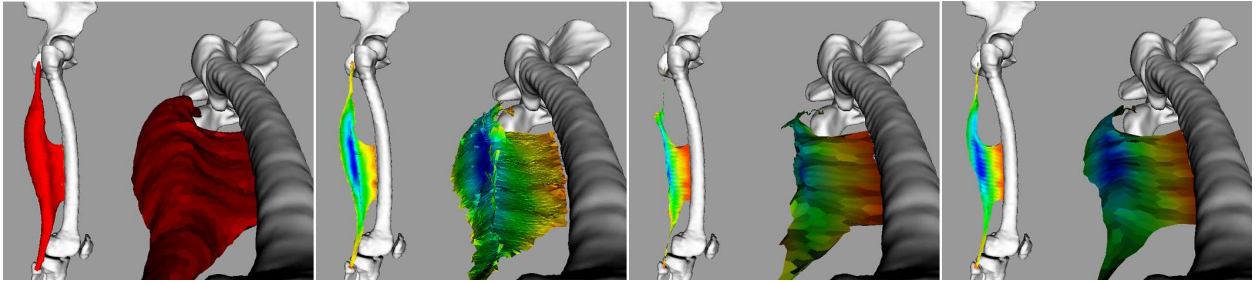


Figure 4.15: From left to right: original model; its medial axis constructed with the powercrust method [ACK01]; medial axis constructed with our method *closest point*; medial axis constructed with our method *radius*. Radii are color mapped (blue=max, red=min)

## 4.7 External constraints

### 4.7.1 Constraints from images

In Section 3.3.3, we have exposed the different similarity measures that have been used for segmentation and registration (geometric and iconic). In our context, discrete deformable simplex meshes will be embedded into MRI volumes. We need to derive vertex constraints that will drive the model to the boundaries of the structure to segment. We only focus on the matching of simplex surfaces that has proven to be efficient. But of course, other techniques based on volumetric models (3-simplex meshes or medial axis) could be imagined based on volumetric homogeneity criteria. For direct segmentation, the main criterion that discriminates organ boundaries is the gradient that is maximal there. On the contrary, for registration, the criterion is based on the similarity between two datasets. We have seen that it is, in general, more robust to use prior knowledge about intensity distributions at boundaries (the gradient is ambiguous to differentiate objects, so is suited for small displacements only). Let be  $\mathbf{S}$  the source (reference) dataset that have already been segmented and  $\mathbf{T}$  the target dataset we want to segment. In our notation, we implicitly include the transformation: for a given vertex  $\mathbf{P}$  in an undeformed position, the intensity value in the reference image is  $S(\mathbf{P})$ . In the deformed configuration, the intensity value in the target image at  $\mathbf{P}$  is  $T(\mathbf{P})$ . Because vertices are in floating positions, we need to interpolate image intensities. We will always use tri-linear interpolation that provides a good trade-off between computational speed and accuracy. For special (non-volumetric) images such as radial MRI and dynamic MRI (see Section 5.2) we use the equivalent technique: the radial interpolation (in one direction, the linear interpolation is performed using angles) and the bilinear interpolation respectively. The registration feature size is related to the neighbourhood of each vertex. In the standard fashion [LM01] [HG04] [MD05], we use the profile along the surface normal. Indeed, this is the direction where we expect intensity changes (at organ boundaries, intensities do not vary much in tangent directions). Intensities in that neighbourhood are called *intensity profiles*. Montagnat et al. [Mon99] have proposed to use tangent patches around the normal direction to smooth the intensity and remove noise. This is, in fact, computationally weighty. We prefer to pre-process the entire dataset with anisotropic smoothing or NL-mean (see Section 5.3) which gives quasi-similar results.

Given a vertex  $\mathbf{P}$  of normal  $\mathbf{n}$  and a step size  $s$ , we regularly sample the image in the normal direction to build intensity profiles. We consider two different profile lengths:  $N_o$  is outside the model and  $N_i$  is inside the model. The intensity profile is defined for  $\mathbf{P}$  as a vector  $[S_{-N_i}, \dots, S_i, \dots, S_{N_o}]$  where  $S_i = S(\mathbf{P} + s.i.\mathbf{n})$  (the total number of values is  $N = N_o + N_i + 1$ ). For the target dataset, we consider intensity profiles shifted by an integer value  $j$ . Indeed, the goal is to search the best vertex position by maximising the similarity

within a search space  $-l \leq j \leq l$ . The total number of values in the target intensity profile is hence  $N + 2l$ . Figure 4.16 illustrates these different concepts:

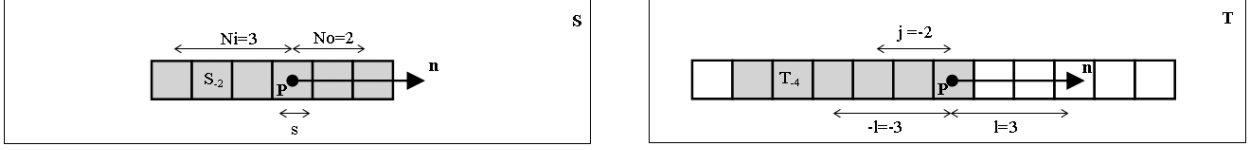


Figure 4.16: Illustration of the intensity profile of  $\mathbf{P}$  in the reference image (left) and target image (right)

Now, we present the different techniques we have tested for computing image constraints. Indeed, for MRI images, no method as proven to be the most effective in the general case, because the modality is very flexible in terms of contrast, signal-to-noise ratio and resolution. We will test all the techniques in the context of our specific protocols, in Section 5.5.2.

**Diffusion-based constraints** Optical flow [HS81] is dedicated to the registration of images from the same protocol and is sensitive to global intensity changes. The assumption is that intensities can be mapped between the two images through the displacement function (hypothesis of intensity conservation). An extension of the basic formulation is the demon algorithm [Thi95] that is more robust (bounded displacement) and adapted to large displacements (iterative method). In our framework, we want the displacement to be collinear with the normal. The demon algorithm has been defined for one voxel. In our case, we average the displacement over the entire intensity profile. We finally obtain:

$$\tilde{\mathbf{P}} = \mathbf{P} + d\mathbf{n} = \mathbf{P} + \left\langle \frac{(S_i - T_i)\nabla\mathbf{S}_i \cdot \mathbf{n}}{\|\nabla\mathbf{S}_i\|^2 + \alpha^2(S_i - T_i)^2} \right\rangle_i \mathbf{n}$$

The displacement is bounded by  $1/(2\alpha)$ . For consistency with intensity profile-based constraints (where the displacement is bounded by  $s.l$ ), we set  $\alpha = 1/(2sl)$ .

**Intensity profile-based constraints (pair and smooth approach)** Here we compute the displacement by finding the best offset  $\tilde{j}$  that maximises a similarity measure (or minimises a distance  $\Delta$ ). This approach is similar to the one of Montagnat et al. [MD05]. The target position is obtained by:

$$\tilde{\mathbf{P}} = \mathbf{P} + s \cdot \tilde{j} \cdot \mathbf{n} = \mathbf{P} + s \cdot \underset{-l < j < l}{\text{Argmin}} \Delta(\mathbf{S}, \mathbf{T}, j) \cdot \mathbf{n}$$

Here we do not consider any probabilistic measures (Section 3.3.3) because the number of voxels  $N$  is not statistically significant to allow a good estimate of the functional relationship between intensities. The simplest method is to maximise the gradient (direct segmentation). With  $\Delta_{NG}$ , we maximise the gradient magnitude while, with  $\Delta_G$ , we project the gradient vector in the normal direction. The sign is obtained ad-hoc from the structure to segment. If not known, one can use the absolute value of the dot-product.

$$\begin{aligned} \Delta_{NG}(\mathbf{S}, \mathbf{T}, j) &= -\|\nabla\mathbf{T}_j\| \\ \Delta_G(\mathbf{S}, \mathbf{T}, j) &= \pm \nabla\mathbf{T}_j \cdot \mathbf{n} \end{aligned}$$

For image registration, we first consider the sum of absolute differences (SAD). This is the simplest similarity measure that consists in computing the Euclidean distance between images. One can use intensity gradients (vectors) instead of intensities (scalars). Distances to minimise are consequently:

$$\begin{aligned}\Delta_{AD}(\mathbf{S}, \mathbf{T}, j) &= \sum_i |S_i - T_{i+j}| \\ \Delta_{GAD}(\mathbf{S}, \mathbf{T}, j) &= \sum_i \|\nabla \mathbf{S}_i - \nabla \mathbf{T}_{i+j}\|\end{aligned}$$

To take into account possible global intensity changes (i.e. different protocol), we also test the cross-correlation measure. Developing the sum of squared Euclidean distance  $\Delta(\mathbf{S}, \mathbf{T}, j) = \sum_i [S_i - T_{i+j}]^2 = \sum_i [S_i^2 + T_{i+j}^2 - 2S_i T_{i+j}]$ , we notice that the squared terms are approximately constant. Thus the distance to minimise is reduced to  $\Delta(\mathbf{S}, \mathbf{T}, j) = -\sum_i S_i T_{i+j}$ . To decrease sensitivity to the feature size  $N$ , to the variation of  $\sum_i S_i^2$  in the image and to the affine change of intensities, researchers have introduced the normalised cross-correlation (NCC) that uses centered intensities divided by the standard deviation.  $NCC$  is formulated as:  $\Delta(\mathbf{S}, \mathbf{T}, j) = -\sum_i (S_i - \bar{S})(T_{i+j} - \bar{T}_j)(\sum_i (S_i - \bar{S})^2 (T_{i+j} - \bar{T}_j)^2)^{-1/2}$ . Now, we have  $-1 \leq \Delta(\mathbf{S}, \mathbf{T}, j) \leq 1$ . Expanding the expression, we can easily improve computational speed:

$$\begin{aligned}\Delta_{NCC}(\mathbf{S}, \mathbf{T}, j) &= -\frac{\sum_i T_{i+j}(S_i - \bar{S}) - \bar{T}_j \sum_i (S_i - \bar{S})}{\sqrt{\sum_i (S_i^2 + \bar{S}^2 - 2S_i \bar{S})(T_{i+j}^2 + \bar{T}_j^2 - 2T_{i+j} \bar{T}_j)}} \\ &= -\frac{\sum_i T_{i+j}(S_i - \bar{S})}{\sqrt{(\sum_i S_i^2 - N\bar{S}^2)(\sum_i T_{i+j}^2 - N\bar{T}_j^2)}}\end{aligned}$$

Here, the terms  $(S_i - \bar{S})$  and  $(\sum_i S_i^2 - N\bar{S}^2)$  related to the reference image can be pre-computed. For using gradient vectors, we propose to extend the formulation to three dimensions as follows:

$$\Delta_{GNCC}(\mathbf{S}, \mathbf{T}, j) = -\frac{\sum_i \nabla \mathbf{T}_{i+j} \cdot (\nabla \mathbf{S}_i - \nabla \bar{\mathbf{S}})}{\sqrt{(\sum_i \|\nabla \mathbf{S}_i\|^2 - N\|\nabla \bar{\mathbf{S}}\|^2)(\sum_i \|\nabla \mathbf{T}_{i+j}\|^2 - N\|\nabla \bar{\mathbf{T}}_j\|^2)}}$$

It is also possible to account for the gradient orientation difference only as this has proven to be efficient in multimodal registration [HM06]:

$$\Delta_{GO}(\mathbf{S}, \mathbf{T}, j) = -\sum_i \frac{\nabla \mathbf{T}_{i+j}}{\|\nabla \mathbf{T}_{i+j}\|} \cdot \frac{\nabla \mathbf{S}_i}{\|\nabla \mathbf{S}_i\|}$$

All measures based on scalar values can be used with the magnitude of the gradient (that can be pre-computed by central difference). The other formulations deal with gradient vectors images that can be also pre-computed (source and target). A problem occurs when using gradient vectors from the source and target datasets, as they are not registered in the same frame (given the transformed source image  $SoM$ , we have  $\nabla \mathbf{S} \circ \mathbf{M}_i \neq \mathbf{M}(\nabla \mathbf{S}_i)$ ). When the transformation  $M$  is a rigid transform, this can be easily overcome by correcting gradient vectors back. However in the elastic case, we would need to estimate the displacement field (which is problematic as we are working in the discrete domain to gain computational time). Because



images will be oriented similarly, we will check the use of the above expression. A transformation-invariant measure is to use *sinus* a *cosine* values of the gradient vectors with regards to the normal  $\mathbf{n}$  (we come to two scalars that we can use with scalar similarity measures). The *cosine* corresponds to the directional gradient. To avoid high computational cost (computation, interpolation and projection of gradient vectors), it is possible to filter intensity profiles by convoluting them with 1D discrete masks in order to enhance particular features. For instance the Prewitt mask  $[-1, 0, 1]$  will compute the gradient in the direction of the intensity profile (surface normal), the mask  $[-1, 2, 1]$  will enhance contours, while the mask  $[1, 2, 1]$  will smooth intensities.

To smooth intensity profile-based constraints and reduce the influence of noise, it is interesting to take into account vertex neighbourhood. Like [MD05], we can directly average the target positions in the neighbourhood  $\eta$  of  $\mathbf{P}$ :  $\tilde{\mathbf{P}} = \mathbf{P} + s \cdot \langle \tilde{j}_i \cdot \mathbf{n}_i \rangle_{\eta}$ . An alternative technique we propose, is to average neighbourhood similarities:  $\Delta(\mathbf{S}, \mathbf{T}, j) = \langle \Delta(\mathbf{S}_i, \mathbf{T}_i, j) \rangle_{\eta}$ . Because they are correlated, we expect noise reduction. We will check the accuracy of these techniques in Section 5.5.3.

### 4.7.2 Collision handling

In multi-organ segmentation (and more generally in model simulation), collision handling is important to guarantee that objects do not self-penetrate. Indeed, unlike real ones, virtual objects can occupy the same place in space, which has to be overcome in two steps: a *collision detection* step to identify in space and time inter-object and self penetrations; and a *collision response* step to restore the system into a correct state.

#### Collision detection

Collision detection is often considered as the bottleneck for simulation, since it involves pair-wise penetration tests for *all* primitives (points, edges, faces, cells) of *all* models involved in the simulation. Obviously, this is a computationally costly process that often prevents from real-time performances, even with relatively small systems, and this complexity increases steeply when adding new primitives/ objects. Since the mid 80's, researchers have tried to reduce this complexity through different algorithmic approaches. The systematic nature of collision detection has recently led to numerous methods based on graphics hardware. Nevertheless, collision detection still remains a major and active research topic in computer graphics, due to the critical importance of its computational burden. We concisely review the different families of methods to motivate our choices. More details can be found in the recent review on the topic by the main contributors to the domain [TKH<sup>+</sup>05].

- **Basic techniques** : a naïve approach is to check for each primitive the closest point to the surface of each object, the sign of the dot product between the distance vector and the surface normal indicating whether the primitive is inside or outside the object. This method is, in fact, sensitive to sudden normal inversion (e.g. self-collisions) leading to instabilities. An improvement is to consider crossing primitives within two successive time-steps (independent to normal direction) [Pro97]. But still, these approaches are very time-consuming and inappropriate for interactive applications. Nevertheless, they are used as a final step of the *bounding volume hierarchy* method, after the quick elimination of non-colliding primitives.
- **Distance fields**: the main idea with this approach is to pre-process and store the distance towards the surface of closed objects for all positions in space. Then, collisions and penetration depths are quickly computed by checking this distance. This approach is especially suited for implicit models (see Section 3.3.2) and rigid objects (distance field calculation is only performed once). However, it is not intended for deformable objects (too high computational cost of distance calculation and update). To improve data storage from using a uniform grid, various data structures (e.g. octree, adaptively sampled fields) have been proposed. Collision detection accuracy is sensitive to the resolution of the distance field.

- **Spatial subdivision:** Such as with distance fields, the *spatial subdivision* technique maps objects to a sample grid. The difference is that the grid is not explicitly stored and distances are not computed. Objects are hashed and primitives/ grid correspondences are stored in a hash table. Comparing entries in the hash table, collisions can be detected. This approach has shown to be suited for real-time applications and computational time can be controlled by adjusting the size of grid cells. However the accuracy is sensitive to inhomogeneous primitive size.
- **Image-space techniques :** here, projected images of the scene are rendered using graphics hardware and 1D collisions are checked within the layered depth image (LDI). This approach is fast and general (works with particles, surfaces and volumes) but its precision depends on the LDI resolution and rendering direction.
- **Bounding volume hierarchies :** the idea is that, if enclosing volumes of two objects are not intersecting, objects are not colliding. It means that quick checks can be performed by considering very simple *bounding volumes*. In a pre-processing step, all primitives of each object are partitioned in space in a recursive fashion, until each leaf contains a single primitive. Then, during simulation, hierarchies are recursively tested for overlap. If two leafs overlap, a one-to-one basic collision detection technique is applied. Hierarchies are also quickly updated at each time-step (refitting) taking into account the displacement. Their structure may be updated in case of large deformations or topology change, despite high computational cost (about 10 times slower than refitting). Various bounding volumes have been proposed (oriented bounding boxes, convex hulls, spherical shells, spheres, etc.). Particularly, *k*-Discrete Oriented Polytopes (*k*-DOPs) [KHM<sup>+</sup>98] [MKE03] have shown to perform the best in terms of computational speed (construction and overlap test) and compactness with regards to ordinary objects.
- **Stochastic methods :** the time-critical aspect of collision detection which can be prohibitive for real-time applications, has lead to inexact methods that ensure control over computational time and collision detection quality. There are two main classes of stochastic methods: the *average-case approach* estimates the probability of intersecting primitives within a bounding volume hierarchy using the spatial distribution of primitives inside a grid, while the approach based on *selected primitives* only checks a subset of bounding volume hierarchies using spatial coherence, random selection, multi-resolution and/ or ad-hoc techniques (e.g. local distance minima in thin structures, sliding objects in permanent contact, etc.).

The *distance field* method with uniform grids is superior for closed rigid models, in terms of computational speed, since collision query can be performed very fast. In our case, soft-tissues/ bones collisions need to be checked when performing segmentation. For bones, the distance field calculation is performed as follows:

1. A uniform grid is initialised around the object (bounding box) with a *10mm* tolerance (for proximity detection) and a *1mm* isotropic resolution.
2. All triangles of the triangulated model are scanned to "switch on" the grid voxels corresponding to the surface (note that this is similar to the *spatial subdivision* approach). The scanning is based on the Bresenham algorithm in 3D. Typical computational time for this step is *7.5s* with a model composed of 80k vertices, but this could be improved using parallelisation.
3. Euclidean distances are propagated from the surface using the parallelised VTK implementation of the algorithm by Saito et al. [ST94] (computational time  $\sim 1.5s$ ).
4. Distances are signed by propagation according to the border of the grid that is outside the object (computational time  $\sim 2.5s$ ).

To detect collisions and proximities between a bone and a deformable model, we first check if bounding boxes overlap. If yes, we check the distance (with trilinear interpolation) from the grid for all vertices of the deformable model. Collision direction is computed by estimating the gradient of the distance map [TKH<sup>+</sup>05][PDTNA]. Gradients and positions may be rigidly transformed to take into account possible change in bone pose from the initial configuration. We found that  $1mm$  resolution was sufficient in practice. To improve accuracy, an offset in the distance need to be added to prevent from edges/ faces collisions.

We perform collision detection between deformable models using bounding volume hierarchies, since they have proven to be efficient and flexible (no dependence with graphics hardware and primitive size, scalability, possibility of adding ad-hoc optimisations). Moreover, it is well adapted to our model type (medial axis) as we will see later. Particularly, we use a 18-DOP quadtree [MKE03] that exhibits good properties (fast tests and updates, compact bounding volumes). A  $k$ -DOP is a convex polyhedron defined by  $k$  half-spaces  $H_i$ ,  $i \in \{0..k-1\}$  :  $H_i = \{\mathbf{P} \in \mathbb{R}^3 | \mathbf{n}_i \cdot \mathbf{P} \leq b_i\}$  where normal vectors  $\mathbf{n}_i$  are chosen by opposite pairs to form  $k/2$  intervals (see Figure 4.17). A DOP is defined through the bound vector  $\mathbf{b} = [b_0, \dots, b_{k-1}]^T$ . To speed up its computation, normal vectors are chosen according to basis vectors. For instance, with  $k = 18$ , we consider the 18 vectors  $\pm[1, 0, 0]^T$ ,  $\pm[0, 1, 0]^T$ ,  $\pm[0, 0, 1]^T$ ,  $\pm[1, 1, 0]^T$ ,  $\pm[1, 0, 1]^T$ ,  $\pm[0, 1, 1]^T$ ,  $\pm[1, -1, 0]^T$ ,  $\pm[1, 0, -1]^T$  and  $\pm[0, 1, -1]^T$ . Consequently, the bounds  $[b_0, b_1]$  of a given set of vertices  $\mathbf{P}_i$ , for example, is simply given by  $[\min_i(P_{ix}), \max_i(P_{ix})]$ . During collision detection, the intersection test between two different DOPs  $\mathbf{b}$  and  $\mathbf{b}'$  is quickly performed by:  $b_{2j} \leq b'_{2j+1}$  and  $b'_{2j} \leq b_{2j+1}$ ,  $\forall j \in [0, k/2]$  ( $k$  tests in total). We generally need to offset intersection tests for proximity detection. DOPs are inflated by modifying the bounds according to:  $\delta b_i = \pm \varepsilon \|\mathbf{n}_i\|$  (note that the norm of normal vectors has to be taken into account, which has not been noticed in [MKE03]).

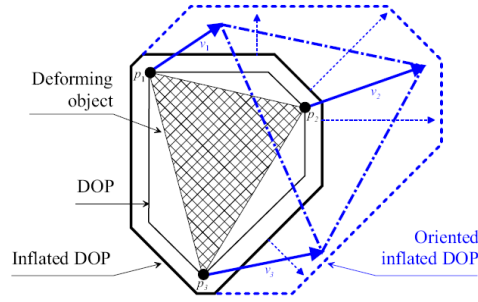


Figure 4.17: 2D illustration of DOP inflation (from [MKE03])

We generate the quadtree using a top-down approach: we first generate a binary tree by subdividing recursively each parent in two along the largest interval direction and by distributing the same number of primitives in the two children DOPs. We stop the recursive subdivision when all leaves contain only one face. Then, we construct the quadtree by taking half of the levels of the binary tree. The total computation time for this initial build is  $\sim 200ms$  for a model with 6500 primitives. During simulation, the quadtree is refitted in the bottom-up fashion: each leaf is updated according the primitives, and parents are recursively updated by calculating the DOP enclosing all children DOPs. This is very fast ( $\sim 10ms$  for the same model).

We distinguish static collision detection (detection of primitives that are contained into objects) and dynamic collision detection (detection of crossing primitives between two successive time steps). In the static case, children DOPs of colliding primitives do not necessarily intersect other DOPs (see Figure 4.18). This is true if we consider closed surfaces (contrary to volumetric models). For open surfaces (e.g. cloth), where the inside and outside is not defined, only the dynamic collision detection is possible (an alternate static approach, is to detect crossing primitives and segment the surface from them, by propagation). Here, we consider closed

2-simplex meshes that are the base of our work, and the direct detection approach (no propagation). We perform static collision detection as a special case of dynamic collision detection: by considering that one object was infinitely far in the former time-step. In the dynamic case, we use inflated DOPs covering two successive time-steps such as in [MKE03] (that are the DOPs enclosing the current DOPs and the former ones). We exploit the fact that if two primitives are penetrating during this time interval, their inflated DOPs overlap. By traversing quadtrees of two objects top-down [TKH<sup>+</sup>05], we can identify overlapping DOPs in which collisions possibly occurred.

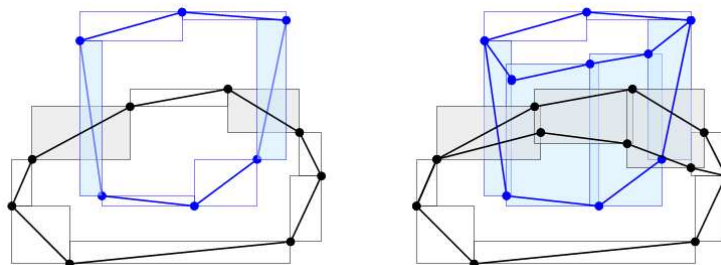


Figure 4.18: A 2D illustration of static collision detection based on bounding primitives (left) and dynamic collision detection based on inflated primitives (right). Crossing primitives are shaded

Now, we have to determine whether primitives intersected within two successive time-steps and if so, where and when collision happened (this is referred as continuous collision detection). For this, we consider point/triangle collisions. We can show that this equivalent to find roots of third degree polynomial equations [Pro97] [BFA02] that can be easily solved. By counting the number of collisions in total for a given vertex, we can determine if it lies inside or outside the object. Note that with static collision detection where one of the objects comes from an infinite position and the other is still, simplifications can be done in terms of DOP overlapping tests (13 tests instead on 18) and primitive intersection tests (first order equation). Contrary to the dynamic case where collision time and location are computed with this method, static collision detection involves an infinite virtual movement so that collision locations have no meaning. We thus project colliding points to the surface of the object to determine the collision vector that will be used for the response.

Now we show how we improve this standard collision detection scheme between closed surfaces using medial axis. We have described, in Section 4.6.4, how to generate medial surfaces that approximate objects through maximal balls. In fact these balls represent explicitly the inside (contrary to closed surfaces where the inside is implicit). Collision detection, like for any volumetric model, is thus simpler because colliding primitives result in overlapping (non-inflated) DOPs. We thus modify our algorithm as follows:

```

// DOP tree refitting
for all leaves A
  for all  $i \in [0, 17]$  do
     $b_i^A = \max_j(\mathbf{n}_i \cdot \mathbf{P}_j + R_j)$  //  $j$  indexes model vertices
    if  $i$  is odd then  $b_i^A = b_i^A + \varepsilon \|\mathbf{n}_i\|$  else  $b_i^A = b_i^A - \varepsilon \|\mathbf{n}_i\|$ 
  end for
end for
while root DOP is not updated do
  for all parent DOP A having updated children do
    for all  $i \in [0, 17]$  do
      if  $i$  is odd then  $b_i^A = \max_c(b_i^c)$  else  $b_i^A = \min_c(b_i^c)$  //  $c$  indexes children of A
    end for
  end for
end while

// DOP overlapping test
traverse(A,B)
if DOP A and B do not overlap ( $\exists j \in [0, 8] / b_{2j}^A \geq b_{2j+1}^B$  or  $b_{2j}^B \geq b_{2j+1}^A$ ) then return false
if A and B are leaves then collisiondetection(A,B) collisiondetection(B,A)
else for all children  $A_i$  and  $B_j$  do traverse( $A_i, B_j$ ) end for

// Collision test
collisiondetection(A,B)
for all vertices  $\mathbf{P}_i$  of the primitives enclosed in A do
   $\mathbf{P}_{i\perp} = \text{Closestpoint}(\mathbf{P}_i, \text{faces enclosed in } B)$ 
   $r_i = \text{InterpolatedRadius}(\mathbf{P}_{i\perp})$ 
  if  $\mathbf{P}_i$  is a medial axis point  $r_i = r_i + \text{Radius}(\mathbf{P}_i)$ 
  if  $P_i P_{i\perp} < r_i$  then addcollision( $\mathbf{P}_i, \mathbf{P}_{i\perp}$ )
end for

addcollision( $\mathbf{P}, \mathbf{P}_{\perp}$ )
if  $\mathbf{P}$  already collides with the object owning  $\mathbf{P}_{\perp}$  then
  if  $PP_{\perp} < r_P$  then  $r_P = PP_{\perp}$  and store( $\mathbf{P}_{\perp}$ )
else  $r_P = PP_{\perp}$  and store( $\mathbf{P}_{\perp}$ )

```

This pseudo code allows static collision detection (no continuous detection). Despite the unknown exact time and location of the collision, the detected collision vectors are in practice sufficient to ensure a precise response. Our algorithm is significantly faster than the one based on surfaces: 1) The number of vertices is reduced from the surface to the medial axis, thus reducing the overall number of DOP and tests; 2) DOP do not need to be inflated with the previous state, thus reducing the number of overlapping DOPs and collision detection tests; 3) Colliding vertices are only projected to a subset of the medial axis. In Section 4.8.3 we perform collision tests that show the benefits of our method.

Out of the collision detection process, a collision is expressed as a linear combination of vertex positions:  $\mathbf{Pc} = \mathbf{PP}_{\perp} = \sum_i w_i \mathbf{P}_i - \mathbf{P}$ . When permanent contact is expected a fast collision detection scheme consists in updating already detected collisions and proximities by re-computing  $\mathbf{Pc}$  with the same weights and new positions (non-sliding contact) or by re-computing collision parameters completely (sliding contacts). The state (collision or proximity) is updated by checking the length of  $\mathbf{Pc}$  (medial axis-based collision detection) and/or the dot-product with the normal (model-based collision detection).

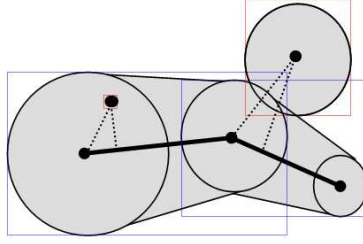


Figure 4.19: In this example where a vertex and a medial point (in red) is tested towards a medial surface (blue), our medial axis-based collision detection method detects 4 collisions

### Collision response

Collision response aims at constraining particles to reach a non-penetrating state. Typically, from a current state where collisions are detected, it consists in altering particle positions, velocities and accelerations (forces) to cancel collisions in the current state and to prevent from collisions in the next states. A good scheme has to respond smoothly to particle changes. This is why a proximity region is commonly considered to allow a gradual collision correction scheme and prevent from particle jumping at object interfaces. Collision response schemes (and contact models) typically fall into four categories: *constraint dynamics* [WGW90] incorporate forces into the system (usually rigid bodies in contact) so that constraints about position and velocities at contact points are satisfied; *penalty methods* [HFS<sup>+</sup>01] enforces non-penetration through stiff springs applied to penetrating surfaces (that can result in instabilities); *exact non-penetration methods* [Bar92] analytically compute penalty terms by resolving non-penetration equations involving the relative acceleration between bodies; and finally *impulse-based methods* [MC94] [Hah98] apply velocity impulses to guaranty non-penetration. The latest treat contact as multiple independent microcollisions, while the firsts use different physical models for computing resting, frictional and collision effects, and exploit contact coherence to resolve equations with multiple solutions. In this section, we derive several position/velocity/acceleration correction terms for a single collision (penalty and impulse-based approaches). We will see in the Section 4.8 how to combine different complementary/antagonist constraints into an evolution scheme. Such as in [VMT00a], we express a collision  $c$  using a state vector  $\mathbf{Qc}$  that denotes either its position ( $\mathbf{Pc}$ ), velocity ( $\mathbf{Vc}$ ) or acceleration ( $\mathbf{Ac}$ ). The collision state is a linear combination between a set of particle states:  $\mathbf{Qc} = \sum_i w_i \mathbf{Q}_i$  where the weights are obtained from the collision detection step. Because we only consider point/face collisions, the collision state is:  $\mathbf{Qc} = \mathbf{Q}\mathbf{Q}_\perp = \sum_i w_i \mathbf{Q}_i - \mathbf{Q}$  where  $\mathbf{Q}$  is the state of the point,  $\mathbf{Q}_\perp$  the state of its projection onto the face (or entry point) and  $\mathbf{Q}_i$  the state of face vertices. A gap (or offset) between objects may be enforced through the vector  $g\mathbf{n}$  ( $\mathbf{n}$  is the surface normal at  $\mathbf{P}_\perp$  and  $g$  the offset). In medial axis-based collision detection, radii are part of this gap.

Now, we propose collision correction terms to be applied to the collision position ( $\Delta\mathbf{Pc} = \tilde{\mathbf{Pc}} - \mathbf{Pc}$ ), velocity ( $\Delta\mathbf{Vc}$ ) and/or acceleration ( $\Delta\mathbf{Ac}$ ). According to the collision normal  $\mathbf{n}$ , we decompose vectors into a normal and a tangent component ( $\mathbf{u} = \mathbf{u}_t + \mathbf{u}_n = \mathbf{u}_t + (\mathbf{u}\cdot\mathbf{n})\mathbf{n}$ ). We introduce the friction coefficient  $0 \leq \mu \leq 1$  in order to simulate rubbing objects. In the proximity region, gradual collision detection is performed through the weight coefficient  $0 \leq \lambda \leq 1$ . We consider the following types of response:

- **Spring:** this kind of response pushes particles towards a non-colliding configuration.
- **Reaction:** when particle are in the proximity region, this type of response is applied in order to avoid future collisions (it cancels normal components).
- **Reference:** From a reference configuration where point-to-face information has been stored, this type of constraint is used to enforce it in order to restore the relative position between particles.

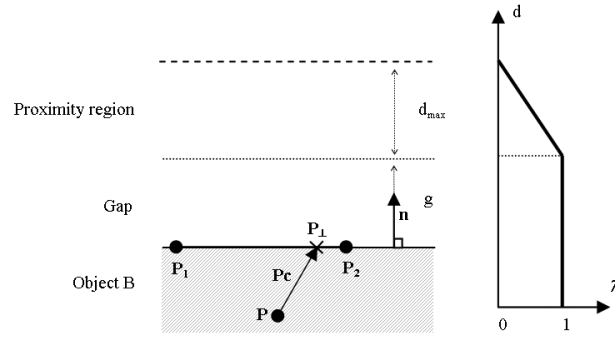


Figure 4.20: 2D illustration of a vertex/ face collision

Where:

- $\mathbf{P}$ ,  $\mathbf{V}$ ,  $\mathbf{A}$ : position, velocity and acceleration of the colliding vertex (object  $A$ )
- $\mathbf{P}_i$ ,  $\mathbf{V}_i$ ,  $\mathbf{A}_i$ : position, velocity and acceleration of colliding vertices (object  $B$ )
- $\mathbf{P}_\perp$ : projection (or "entry point") of  $\mathbf{P}$  on (in)  $B$
- $\mathbf{n}$ : normal vector at  $\mathbf{P}_\perp$
- $g$ : gap or offset
- $d_{max}$ : proximity distance
- $\lambda = 1 + (\mathbf{Pc} \cdot \mathbf{n} + g)/d_{max}$ : proximity factor ( $\lambda = 1$  if  $\mathbf{Pc} \cdot \mathbf{n} > -g$ ;  $\lambda = 0$  if  $\mathbf{Pc} \cdot \mathbf{n} < -g - d_{max}$ )
- $\mathbf{Pc} = \mathbf{P}\mathbf{P}_\perp = \sum_i w_i \mathbf{P}_i - \mathbf{P}$ : collision position
- $\mathbf{Vc} = \sum_i w_i \mathbf{V}_i - \mathbf{V}$ : collision velocity
- $\mathbf{Ac} = \sum_i w_i \mathbf{A}_i - \mathbf{A}$ : collision acceleration

Collision constraints are given in Table 4.23. Their goal is to impose a certain distance between  $\mathbf{P}$  and its projection  $\mathbf{P}_\perp$ . The  $2^{nd}$  order spring comes from the Taylor development of positions and velocities at time  $t + 2dt$  [VMT00a] and impose both a desired position (taken in the spring row) and a desired velocity (reaction row) through acceleration correction. As we will see in Section 4.8, there are basically two ways of applying these constraints: soft constraints consists of weighted forces that are added to the dynamic evolution. Another way is to alter directly positions, velocities and accelerations ( $\mathbf{Qc} = \tilde{\mathbf{Qc}}$ ). Note that when performing position correction, the detected collisions need to be updated to take into account position changes. The use of the three corrections sequentially with thresholds is in practice the most robust method. In the next chapters, we will see what type of response is relevant in the framework of musculoskeletal segmentation.

## 4.8 Evolution

### 4.8.1 Forces

We have previously expressed geometric constraints to be applied to the models, in terms of desired particle positions. Now, we need to derive particle forces that would satisfy these constraints, and integrate them into a dynamic evolution scheme (Section 3.3.5). A common way to derive forces is to express a potential energy

	Position correction	Velocity correction	Acceleration correction
<b>Spring</b> $\lambda = 1$	$\tilde{\mathbf{P}}\mathbf{c} = (1 - \mu)\mathbf{P}\mathbf{c}_t - g\mathbf{n}$	$\tilde{\mathbf{V}}\mathbf{c} = \Delta\mathbf{P}\mathbf{c}/dt$ $= ((1 - \mu)\mathbf{P}\mathbf{c}_t - g\mathbf{n} - \mathbf{P}\mathbf{c})/dt$	$1^{st}$ order: $\tilde{\mathbf{A}}\mathbf{c} = \Delta\mathbf{P}\mathbf{c}/dt^2$ $= ((1 - \mu)\mathbf{P}\mathbf{c}_t - g\mathbf{n} - \mathbf{P}\mathbf{c})/dt^2$ $2^{nd}$ order: $\tilde{\mathbf{A}}\mathbf{c} = \Delta\mathbf{P}\mathbf{c}/dt^2 - (0.5\Delta\mathbf{V}\mathbf{c} + 2\mathbf{V}\mathbf{c})/dt$
<b>Reaction</b> $0 < \lambda < 1$	-	if $(\mathbf{V}\mathbf{c} \cdot \mathbf{n} > 0)$ $\tilde{\mathbf{V}}\mathbf{c} = (1 - \lambda)\mathbf{V}\mathbf{c}_n + (1 - \lambda(1 - \mu))\mathbf{V}\mathbf{c}_t$ else $\tilde{\mathbf{V}}\mathbf{c} = \mathbf{V}\mathbf{c}_n + (1 - \lambda(1 - \mu))\mathbf{V}\mathbf{c}_t$	if $(\mathbf{A}\mathbf{c} \cdot \mathbf{n} > 0)$ $\tilde{\mathbf{A}}\mathbf{c} = (1 - \lambda)\mathbf{A}\mathbf{c}_n + (1 - \lambda(1 - \mu))\mathbf{A}\mathbf{c}_t$ else $\tilde{\mathbf{A}}\mathbf{c} = \mathbf{A}\mathbf{c}_n + (1 - \lambda(1 - \mu))\mathbf{A}\mathbf{c}_t$
<b>Ref.</b> $0 < \lambda \leq 1$	$\tilde{\mathbf{P}}\mathbf{c} = (1 - \mu)\mathbf{P}\mathbf{c}_t + d_{ref}\mathbf{n}$ with $d_{ref} = \mathbf{P}\mathbf{c}_{ref} \cdot \mathbf{n}_{ref}$	$\tilde{\mathbf{V}}\mathbf{c} = \Delta\mathbf{P}\mathbf{c}/dt$ $= ((1 - \mu)\mathbf{P}\mathbf{c}_t + d_{ref}\mathbf{n})/dt$	$\tilde{\mathbf{A}}\mathbf{c} = \Delta\mathbf{P}\mathbf{c}/dt^2$ $= ((1 - \mu)\mathbf{P}\mathbf{c}_t + d_{ref}\mathbf{n})/dt^2$

Table 4.23: Collision constraints

to be minimised. Let us consider the energy  $E(\mathbf{X})$  that depends on a set of particles  $\mathbf{P}_i$  (their positions are concatenated into the vector  $\mathbf{X}$ ). Instead of a scalar energy term, it is possible to use a constraint vector  $\mathbf{C}(\mathbf{X})$  that we want to be null such as in [BW98]:  $E(\mathbf{X}) = \frac{1}{2}\mathbf{C}(\mathbf{X})^T\mathbf{C}(\mathbf{X})$ . Given the target position  $\tilde{\mathbf{P}}$  of a vertex  $\mathbf{P}$ , a coherent expression of the energy for this single constraint is:  $E(\mathbf{X}) = \frac{1}{2}P\tilde{P}^2$  (the constraint vector is  $\mathbf{C}(\mathbf{X}) = \mathbf{P}\tilde{\mathbf{P}}$ ). Forces acting of the particle  $\mathbf{P}_i$  with a certain stiffness  $\alpha$  are computed through energy derivatives:

$$\mathbf{F}_{\mathbf{P}_i} = -\alpha \frac{\partial E(X)}{\partial \mathbf{P}_i} = -\alpha \frac{\partial \mathbf{C}(\mathbf{X})}{\partial \mathbf{P}_i} \mathbf{C}(\mathbf{X}) \quad (4.2)$$

For implicit schemes, we need to compute force derivatives with regards to particle positions and velocities. These  $3 \times 3$  symmetric matrices can be decomposed into an isotropic term (proportional to  $\mathbf{I}$ ) and anisotropic term (proportional to  $\mathbf{F}\mathbf{F}^T/F^2$ , where  $\mathbf{F}$  is the force) [VMT00b]. Indeed, forces derived from a target position (shape and smoothing) are isotropic, while forces derived from a target direction (volume conservation, image forces, collision response, radial forces) are anisotropic. However, perfectly anisotropic variation leads to instability, and a small isotropic contribution, that would act as a damping factor, need to be added in practice [VMT00b]. For instance, for an anisotropic force applied to a single particle, we use the linear combination:  $\partial\mathbf{F}/\partial\mathbf{P} = \partial\alpha\mathbf{P}\tilde{\mathbf{P}}/\partial\mathbf{P} \simeq -\alpha k\mathbf{I} - \alpha(1 - k)\mathbf{P}\tilde{\mathbf{P}} \cdot \mathbf{P}\tilde{\mathbf{P}}^T / P\tilde{P}^2$  where  $k$  is a constant equals to 0.9.

### Independent forces

External forces are applied to particles independently, meaning that the corresponding energies  $E(\mathbf{P})$  only depend on a single particle. These forces are directed towards the target position:  $\mathbf{F}_{\mathbf{P}} = \alpha\mathbf{P}\tilde{\mathbf{P}}$ . This follows the Hookean spring law (with a null rest length) and is analogous to a gradient descent scheme where the time-step corresponds to the stiffness. We will see at the end of this section how the stiffness parameter can be weighted to allow a more direct minimisation of the energy.

### Damping

Damping is a particular force that is inversely proportional to particle velocity. It reduces considerably instabilities and oscillations that can occur in particle systems. In the real world, it has a mechanical



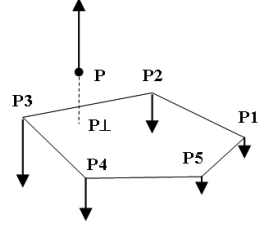
interpretation: the energy dissipation due to air, object viscosity and friction. The straightforward isotropic formulation  $\mathbf{F}_{\mathbf{P}} = -\gamma \frac{\partial \mathbf{P}}{\partial t} = -\gamma \mathbf{V}$  (where  $\gamma$  is the stiffness) leads to a global damping of the system that penalises rigid motion (which is undesirable because oscillations appear locally). A better way is to project velocities onto the direction of the force that is applied [NMK<sup>+</sup>05]:  $\mathbf{F}_{\mathbf{P}} = -\gamma \mathbf{V} \cdot \mathbf{F} \mathbf{F} / F^2$ . We can consider all forces to be applied to  $\mathbf{P}$ . But in practice the use of the total force (weighted-sum of individual forces) is sufficient. The derivative of the damping force with regards to the velocity is given by:  $\partial \mathbf{F} / \partial \mathbf{V} = -\gamma \mathbf{F} \mathbf{F}^T / F^2$

### Forces applied to a set of particles

Most internal constraints result from the interaction between several particles (e.g.  $k + 2$  neighbouring particles for smoothing constraints). The main restriction, when applying forces to a set of particles, is to respect the conservation laws of mechanics. Especially, forces need to have a null linear and angular momentum. If not, external *ghost* forces would produce undesirable translational and rotational strain.

$$\sum_i \mathbf{F}_{\mathbf{P}_i} = \mathbf{0} \quad \text{and} \quad \sum_i \mathbf{P}_i \wedge \mathbf{F}_{\mathbf{P}_i} = \mathbf{0} \quad (4.3)$$

We have seen that, for radial and collision constraints, the goal is to enforce a certain distance between a particle  $\mathbf{P}$  and its projection  $\mathbf{P}_{\perp} = \sum_i w_i \mathbf{P}_i$  on a surface. Let  $\mathbf{C} = \widetilde{\mathbf{P}} \mathbf{P}_{\perp} - \mathbf{P} \mathbf{P}_{\perp}$  be the constraint vector. Here, we assume that there is no dependence between  $\widetilde{\mathbf{P}} \mathbf{P}_{\perp}$  and  $\mathbf{P} \mathbf{P}_{\perp}$ . Applying Equation 4.2, we obtain forces summarised in Table 4.24. The linear momentum is null and the angular momentum is null only if  $\widetilde{\mathbf{P}} \mathbf{P}_{\perp}$  is collinear with  $\mathbf{P} \mathbf{P}_{\perp}$ . For radial constraints, we have used this method on the set of medial axis vertices  $\mathbf{Q}_j$  (see Section 4.6.4). In reality, we should have included model vertices, because they are dependent (we have decoupled them in order to handle medial axis and model constraints independently). We can notice that if we would have done



it, the *radius* method would lead to the same results (null angular momentum). However, this is not the case for the *radius and barycentric coordinate* method (we thus expect some imprecision with this method, see tests in Section 4.8.3). For collision constraints (Table 4.23), target parameters depend on the present position (through the normal vector), which has been neglected. For instance, for position correction with a friction of 1, the *spring* method would lead to  $\mathbf{P}\mathbf{c}(t + dt) = -g\mathbf{n}(t) \neq -g\mathbf{n}(t + dt)$  and the *reference* method to  $\mathbf{P}\mathbf{c}(t + dt) = (\mathbf{P}\mathbf{c} \cdot \mathbf{n})_{ref} \mathbf{n}(t) \neq (\mathbf{P}\mathbf{c} \cdot \mathbf{n})_{ref} \mathbf{n}(t + dt)$ . Also, the angular momentum is generally not null due to the friction coefficient and to the normal that misalign  $\widetilde{\mathbf{P}}\mathbf{c} = \widetilde{\mathbf{P}} \mathbf{P}_{\perp}$  and  $\mathbf{P}\mathbf{c} = \mathbf{P} \mathbf{P}_{\perp}$ . However, because we use small collision distances ( $\mathbf{P}\mathbf{c} \cdot \mathbf{n}$  is small) and high friction coefficients, inaccuracies are not prohibitive. The cost of taking into account normal variation (see Table 4.26) is not worth. The *reference* matching method can be viewed as applying shape constraints between two objects. However it is not an accurate method because the variation of the normal has been neglected. In fact, this will be applied to softly attach objects (to take coarsely into account proximity information in the segmentation). Accurate internal forces will compensate inaccuracies of interaction constraints.

Concerning 3-simplex meshes internal constraints, we have seen in Table 4.19 and 4.20 that the general form is:  $\widetilde{\mathbf{P}} = \sum_{0 < i \leq 4} \varepsilon_i \mathbf{P}_i$  with  $\sum_4 \varepsilon_i = 1$  (internal constraint applied to a vertex  $\mathbf{P}$  of neighbours  $\mathbf{P}_i$ ). Applying Equation 4.2 with the constraint vector  $\mathbf{C} = \mathbf{P} \widetilde{\mathbf{P}}$ , we get the linear forces and force derivatives summarised in Table 4.25. These forces respect momentum conservation laws 4.3.

For 2-simplex meshes, local internal constraints deal with the interaction between four different particles  $\widetilde{\mathbf{P}} = \sum_3 \varepsilon_i \mathbf{P}_i + h\mathbf{n}$  with  $\sum_3 \varepsilon_i = 1$  and  $\mathbf{n} = \mathbf{P}_1 \mathbf{P}_3 \wedge \mathbf{P}_1 \mathbf{P}_2 / \|\mathbf{P}_1 \mathbf{P}_3 \wedge \mathbf{P}_1 \mathbf{P}_2\|$ . The use of the constraint vector

Constraint	Forces	Force derivatives
$\mathbf{C} = \widetilde{\mathbf{P}}\mathbf{P}_\perp - \mathbf{P}\mathbf{P}_\perp$ $\sum_i w_i = 1$	$\mathbf{F}_\mathbf{P} = -\alpha(\widetilde{\mathbf{P}}\mathbf{P}_\perp - \mathbf{P}\mathbf{P}_\perp)$ $\mathbf{F}_{\mathbf{P}_i} = \alpha w_i(\widetilde{\mathbf{P}}\mathbf{P}_\perp - \mathbf{P}\mathbf{P}_\perp)$	$\frac{\partial \mathbf{F}_\mathbf{P}}{\partial \mathbf{P}} = -\alpha \mathbf{I}$ $\frac{\partial \mathbf{F}_\mathbf{P}}{\partial \mathbf{P}_i} = \frac{\partial \mathbf{F}_{\mathbf{P}_i}}{\partial \mathbf{P}} = \alpha w_i \mathbf{I}$ $\frac{\partial \mathbf{F}_{\mathbf{P}_i}}{\partial \mathbf{P}_j} = \frac{\partial \mathbf{F}_{\mathbf{P}_j}}{\partial \mathbf{P}_i} = -\alpha w_i w_j \mathbf{I}$

Table 4.24: Interaction forces

$k$	Constraint	Forces	Force derivatives
3	$\mathbf{C} = \mathbf{P}\tilde{\mathbf{P}} = \sum_4 \varepsilon_i \mathbf{P}_i - \mathbf{P}$ $\sum_4 \varepsilon_i = 1$	$\mathbf{F}_\mathbf{P} = \alpha \mathbf{P}\tilde{\mathbf{P}}$ $\mathbf{F}_{\mathbf{P}_i} = -\alpha \varepsilon_i \mathbf{P}\tilde{\mathbf{P}}$	$\frac{\partial \mathbf{F}_\mathbf{P}}{\partial \mathbf{P}} = -\alpha \mathbf{I}$ $\frac{\partial \mathbf{F}_\mathbf{P}}{\partial \mathbf{P}_i} = \frac{\partial \mathbf{F}_{\mathbf{P}_i}}{\partial \mathbf{P}} = \alpha \varepsilon_i \mathbf{I}$ $\frac{\partial \mathbf{F}_{\mathbf{P}_i}}{\partial \mathbf{P}_j} = \frac{\partial \mathbf{F}_{\mathbf{P}_j}}{\partial \mathbf{P}_i} = -\alpha \varepsilon_i \varepsilon_j \mathbf{I}$

Table 4.25: Local internal forces for simplex volumes

$\mathbf{C} = \mathbf{P}\tilde{\mathbf{P}}$  leads to non-linear equations and complex derivatives expressions (first line of Table 4.26), due to the cross-product  $\mathbf{n}$ . The linear momentum is null but not the angular momentum (despite the constraint rotational invariance). It results in *ghost* spinning and to non-symmetric force derivative matrices which is problematic for implicit integration that uses the conjugate gradient method (see Section 3.3.5). We propose another formulation through analogy with the volumetric case. Assuming that forces are proportional to  $\mathbf{P}\tilde{\mathbf{P}}$  (with the scales 1 and  $-\lambda_i$  for  $\mathbf{P}$  and the  $\mathbf{P}_i$  respectively), and injecting them into the momentum equations, we get a unique solution for the  $\lambda_i$  (see Table 4.26). In fact, solutions correspond to the coordinates of the goal position  $\tilde{\mathbf{P}}$  in the basis  $(\mathbf{P}, \mathbf{P}\mathbf{P}_1, \mathbf{P}\mathbf{P}_2, \mathbf{P}\mathbf{P}_3)$ . In the volumetric case, they were equals to the metric parameters  $\lambda_i = \varepsilon_i$  since  $\sum_i \varepsilon_i \tilde{\mathbf{P}}\mathbf{P}_i = 0 \iff \sum_4 \varepsilon_i \mathbf{P}\mathbf{P}_i = \mathbf{P}\tilde{\mathbf{P}}$ . For  $k = 2$ , we have to compute them at each iteration. This time, the method ensures momentum conservation but still, the global derivative matrix is not symmetric. Symmetry can be recovered by replacing the derivative matrices by the isotropic matrices obtained with for the volumetric case (Table 4.25). This would have a minor impact on the accuracy and stability of the implicit integration.

### Weighting the stiffness parameters

Such as for the gradient descent method, the forces we have formulated are oriented towards the region of lower energy. However, due to the arbitrary stiffness, time-step and mass, there is no evidence that the energy will be actually reduced. This is a common and well-known problem of Hookean springs: a too large time-step or too small mass would undesirably overshoot the position (we can see it from Equation 3.1), resulting in an increase of the potential energy and instability for explicit schemes. To adjust the stiffness, we perform a first-order expansion of the constraint energy or vector that we want to be null at next time-step:

$k$	Constraint	Forces	Force derivatives
2	$\mathbf{C} = \mathbf{P}\tilde{\mathbf{P}}$ $= \sum_3 \varepsilon_i \mathbf{P}_i + h\mathbf{n} - \mathbf{P}$ $\sum_3 \varepsilon_i = 1$	$\mathbf{F}_{\mathbf{P}} = \alpha\mathbf{P}\tilde{\mathbf{P}}$ $\mathbf{F}_{\mathbf{P}_i} = -\alpha(\varepsilon_i\mathbf{I} + h\frac{\partial\mathbf{n}}{\partial\mathbf{P}_i})\mathbf{P}\tilde{\mathbf{P}}$	$\frac{\partial\mathbf{F}_{\mathbf{P}}}{\partial\mathbf{P}} = -\alpha\mathbf{I}$ $\frac{\partial\mathbf{F}_{\mathbf{P}}}{\partial\mathbf{P}_i} = \frac{\partial\mathbf{F}_{\mathbf{P}_i}}{\partial\mathbf{P}} = \alpha(\varepsilon_i\mathbf{I} + h\frac{\partial\mathbf{n}}{\partial\mathbf{P}_i})$ $\frac{\partial\mathbf{F}_{\mathbf{P}_i}}{\partial\mathbf{P}_j} = -\alpha((\varepsilon_i\mathbf{I} + h\frac{\partial\mathbf{n}}{\partial\mathbf{P}_i})(\varepsilon_j\mathbf{I} + h\frac{\partial\mathbf{n}}{\partial\mathbf{P}_j})^T + \frac{\partial^2\mathbf{n}}{\partial\mathbf{P}_i\partial\mathbf{P}_j}\mathbf{P}\tilde{\mathbf{P}})$
		$\mathbf{F}_{\mathbf{P}} = \alpha\mathbf{P}\tilde{\mathbf{P}}$ $\mathbf{F}_{\mathbf{P}_i} = -\alpha\lambda_i\mathbf{P}\tilde{\mathbf{P}}$	$\frac{\partial\mathbf{F}_{\mathbf{P}}}{\partial\mathbf{P}} = -\alpha\mathbf{I}$ $\frac{\partial\mathbf{F}_{\mathbf{P}}}{\partial\mathbf{P}_i} = \alpha(\varepsilon_i\mathbf{I} + h\frac{\partial\mathbf{n}}{\partial\mathbf{P}_i})$ $\frac{\partial\mathbf{F}_{\mathbf{P}_i}}{\partial\mathbf{P}} = \alpha\lambda_i\mathbf{I}$ $\frac{\partial\mathbf{F}_{\mathbf{P}_i}}{\partial\mathbf{P}_j} = -\alpha\lambda_i(\varepsilon_j\mathbf{I} + h\frac{\partial\mathbf{n}}{\partial\mathbf{P}_j})$

Table 4.26: Local internal forces for simplex surfaces using a classical method based on energy derivation (first line) and our method (second line)

Where:

- The derivative of the cross-product is:  $\frac{\partial\mathbf{n}}{\partial\mathbf{u}} = \frac{\partial(\mathbf{u}\wedge\mathbf{v}/\|\mathbf{u}\wedge\mathbf{v}\|)}{\partial\mathbf{u}} = \frac{1}{\|\mathbf{u}\wedge\mathbf{v}\|}(\mathbf{v}^* + (\mathbf{n}\wedge\mathbf{v})\mathbf{n}^T)$
- $\mathbf{v}^* = \begin{bmatrix} 0 & -v_z & v_y \\ v_z & 0 & -v_x \\ -v_y & v_x & 0 \end{bmatrix}$  is the cross-product operator ( $\mathbf{v}^*.\mathbf{x} = \mathbf{v}\wedge\mathbf{x}$ )
- $\begin{bmatrix} \lambda'_1 \\ \lambda'_2 \\ \lambda'_3 \end{bmatrix} = \begin{bmatrix} pp_{1x} & pp_{2x} & pp_{3x} \\ pp_{1y} & pp_{2y} & pp_{3y} \\ pp_{1z} & pp_{2z} & pp_{3z} \end{bmatrix}^{-1} \mathbf{P}\tilde{\mathbf{P}}$  and  $\lambda_i = \frac{\lambda'_i}{\sum_3 \lambda'_i}$

$$E(\mathbf{X} + \Delta\mathbf{X}) \simeq E(\mathbf{X}) + \sum_i \left(\frac{\partial E(\mathbf{X})}{\partial\mathbf{P}_i}\right)^T \Delta\mathbf{P}_i = 0$$

$$\mathbf{C}(\mathbf{X} + \Delta\mathbf{X}) \simeq \mathbf{C}(\mathbf{X}) + \sum_i \left(\frac{\partial\mathbf{C}(\mathbf{X})}{\partial\mathbf{P}_i}\right)^T \Delta\mathbf{P}_i = \mathbf{0}$$

When we compute forces using constraint derivation (Equation 4.2) which is the case in general, and approximating the displacements by  $\Delta\mathbf{P} \simeq \mathbf{M}^{-1}\mathbf{F}dt^2$ , we come to a formulation that is a generalisation of the Newton-Raphson method (see Section 3.3.5). We can compute a stiffness vector  $\boldsymbol{\alpha}$  that makes the energy zero. Here, we also introduce the reduced masses  $m$  and  $\mathbf{m}$  for a given constraint, as shown in the equations:

$$\begin{aligned}
\mathbf{F}_{\mathbf{P}_i} &= -\alpha \frac{\partial E(\mathbf{X})}{\partial \mathbf{P}_i} \\
&= -\alpha \frac{\partial \mathbf{C}(\mathbf{X})}{\partial \mathbf{P}_i} \mathbf{C}(\mathbf{X}) = -\alpha w_i \mathbf{C}(\mathbf{X}) \\
\alpha &= E(\mathbf{X}) \left( \sum_j \left( \frac{\partial E(\mathbf{X})}{\partial \mathbf{P}_j} \right)^T \mathbf{M}_j^{-1} \frac{\partial E(\mathbf{X})}{\partial \mathbf{P}_j} \right)^{-1} \mathbf{I} dt^{-2} = E(\mathbf{X}) m \mathbf{I} dt^{-2} \\
&= \left( \sum_j w_j^2 \mathbf{M}_j^{-1} \right)^{-1} dt^{-2} = \mathbf{m} dt^{-2} \\
\Delta \mathbf{P}_i &\simeq -\mathbf{M}_i^{-1} m E(\mathbf{X}) \frac{\partial E(\mathbf{X})}{\partial \mathbf{P}_i} \\
&\simeq -\mathbf{M}_i^{-1} \mathbf{m} \mathbf{C}(\mathbf{X}) w_i
\end{aligned}$$

Integrated into any evolution scheme, we see that the mass and time-step terms are cancelled, resulting in a more stable behaviour (goal positions are not overestimated). These expressions are made through simplifications, realisable only if: we use a scalar energy  $E$  (it leads to a scalar stiffness  $\alpha$  that can be factorised), or if we use a constraint vector  $\mathbf{C}$  that is a *linear* combination of particle positions (leading to isotropic derivatives  $\frac{\partial \mathbf{C}}{\partial \mathbf{P}_i} = w_i \mathbf{I}$  that can be factorised). For scalar masses and energies, we get the same expression than [MHHR06]. We apply directly position corrections  $\Delta \mathbf{P}$  (position-based dynamics). Indeed, it is possible to use either position corrections (regular Newton-Raphson scheme) or forces (dynamic evolution) as already discussed in Sections 3.3.5 and 3.3.5. We will compare the two approaches in Section 4.8.3. In our case, we get the following values for the corrected stiffness:

- **Independent force:**  $\alpha = \mathbf{M} dt^{-2}$  or  $\alpha = \mathbf{0}$  (if  $\mathbf{M}^{-1} = \mathbf{0}$ ).
- **Interaction forces:**  $\alpha = (\sum_i w_i^2 \mathbf{M}_i^{-1} + \mathbf{M}^{-1})^{-1} dt^{-2}$
- **3-simplex internal forces:**  $\alpha = (\sum_4 \varepsilon_i^2 \mathbf{M}_i^{-1} + \mathbf{M}^{-1})^{-1} dt^{-2}$
- **2-simplex internal forces (first method):**  $\alpha = \frac{1}{2} P \tilde{P}^2 (\sum_3 [(\varepsilon_i \mathbf{I} + h \frac{\partial \mathbf{n}}{\partial \mathbf{P}_i}) \mathbf{P} \tilde{\mathbf{P}}]^T \mathbf{M}_i^{-1} [(\varepsilon_i \mathbf{I} + h \frac{\partial \mathbf{n}}{\partial \mathbf{P}_i}) \mathbf{P} \tilde{\mathbf{P}}] + \mathbf{P} \tilde{\mathbf{P}}^T \mathbf{M}^{-1} \mathbf{P} \tilde{\mathbf{P}})^{-1} dt^{-2}$
- **2-simplex internal forces (second method):**  $\alpha = (\sum_3 \varepsilon_i \lambda_i \mathbf{M}_i^{-1} + \mathbf{M}^{-1})^{-1} dt^{-2}$

For independent forces, our results are similar to Müller et al. [MHTG05], that have proposed a shape matching-based method for computing particle dynamics. We also obtain the same expression for interaction forces than Volino et al. [VMT00a] (collision response). For the first (non-linear) method dealing with 2-simplex surfaces, we had to use the scalar energy  $E = \frac{1}{2} P \tilde{P}$ . With the second method, the energy does not strictly go to zero since we have neglected the non-linear derivative of the normal.

### Handling inter-constraint dependencies

In practice, several constraints are combined to compute the new particles state (for instance, collision and shape constraints). But they are redundant: resolving a particular constraint can resolve some others. A common practice is to use a weighted sum of forces presented above (or alternatively, a weighted sum of the position correction terms):

$$\begin{aligned}
\Delta \mathbf{P}_i \simeq \mathbf{M}_i^{-1} \mathbf{F}_{\mathbf{P}_i} dt^2 &= -\frac{\mathbf{M}_i^{-1}}{\sum_{l \ni i} \gamma_l} \sum_{l \ni i} \gamma_l m_l E_l \frac{\partial E_l}{\partial \mathbf{P}_i} \\
&= -\frac{\mathbf{M}_i^{-1}}{\sum_{l \ni i} \gamma_l} \sum_{l \ni i} \gamma_l \mathbf{m}_l \Delta \mathbf{C}_k w_{il}
\end{aligned}$$

Here, to allow a control of force contributions, the new stiffness parameter  $\boldsymbol{\alpha}' = \gamma \boldsymbol{\alpha}$  where  $\gamma$  is a scalar in  $[0, 1]$  and  $\boldsymbol{\alpha}$  the parameter defined above, is used. With this approach (*weighted sum*), equilibrium is found during the evolution process when all constraints compensate, meaning that the total energy is at a local minimum (similarly to a real mechanical system). We already used this approach in Section 4.6.4 when averaging model vertices contribution at medial axis points.

Instead of using a weighted sum of the different constraints, we can think about a more accurate approach that would compute the corrections to be applied in order to satisfy *all* constraints globally. Here, individual constraints are not cancelled independently as before. Instead, we apply a certain correction term ( $\Delta E_l$  or  $\Delta \mathbf{C}_l$ ) for each constraint and sum the different contributions. The total variation for a given constraint  $k$  is then:

$$\begin{aligned}
E_k(\mathbf{X} + \Delta \mathbf{X}) &\simeq E_k(\mathbf{X}) + \sum_{i \in k} \left( \frac{\partial E_k}{\partial \mathbf{P}_i} \right)^T \Delta \mathbf{P}_i = E_k(\mathbf{X}) + \sum_{i \in k} \left( \frac{\partial E_k}{\partial \mathbf{P}_i} \right)^T \sum_{l \ni i} \mathbf{M}_i^{-1} m_l \Delta E_l \frac{\partial E_l}{\partial \mathbf{P}_i} = 0 \\
\mathbf{C}_k(\mathbf{X} + \Delta \mathbf{X}) &\simeq \mathbf{C}_k(\mathbf{X}) + \sum_{i \in k} w_{ik} \Delta \mathbf{P}_i = \mathbf{C}_k(\mathbf{X}) + \sum_{i \in k} w_{ik} \sum_{l \ni i} \mathbf{M}_i^{-1} \mathbf{m}_l \Delta \mathbf{C}_k w_{il} = 0
\end{aligned} \tag{4.4}$$

The total (desired) variation ( $\widetilde{\Delta E}_k = -E_k(\mathbf{X})$  or  $\widetilde{\Delta \mathbf{C}}_k = -\mathbf{C}_k(\mathbf{X})$ ) is hence a linear combination of the corrections ( $\Delta E_l$  or  $\Delta \mathbf{C}_l$ ) applied to individual constraints. If we concatenate corrections and reduced masses into a single vector, we finally get:

$$\begin{aligned}
\mathbf{Y} &= \mathbf{H} \mathbf{W} & \text{With: } Y_k &= \widetilde{\Delta E}_k = -E_k(\mathbf{X}) \\
H_{kl} &= \frac{\partial E_k}{\partial (m_l E_l)} = \sum_{i \in k \cap l} \left( \frac{\partial E_k}{\partial \mathbf{P}_i} \right)^T \mathbf{M}_i^{-1} \frac{\partial E_l}{\partial \mathbf{P}_i} \\
W_l &= m_l \Delta E_l \\
\text{or: } Y_{\mathbf{k}} &= \widetilde{\Delta \mathbf{C}}_{\mathbf{k}} = -\mathbf{C}_{\mathbf{k}}(\mathbf{X}) \\
\mathbf{H}_{\mathbf{k}l} &= \frac{\partial \mathbf{C}_{\mathbf{k}}}{\partial (\mathbf{m}_l \mathbf{C}_l)} = \sum_{i \in k \cap l} w_{ik} \mathbf{M}_i^{-1} w_{il} \\
W_l &= \mathbf{m}_l \Delta \mathbf{C}_l
\end{aligned}$$

These general formulations can be mixed to allow the use of constraints based on energy and constraints based on vector within a unique system. It is possible to resolve this sparse system with standard methods. The vector-based approach leads to a symmetric matrix  $\mathbf{H}$  for which the conjugate gradient method is particularly suited [VMT00a]. In fact, the size of the system can be quite large if we apply several constraints

on each particle. Instead of resolving the system, a faster approach is to resolve each constraint sequentially, and take into account state changes for processing the next constraints. These three resolution methods (weighted-sum, sequential resolution and matrix inversion) can be applied to compute forces, position correction terms (position-based approach) or velocity correction terms (impulse-based approach). Accurate correction computation through the combination of all constraints is especially relevant for position correction due to stability. For forces and velocities, stability is ensured by the integration in time and the use of derivatives with regards to particle position (implicit scheme). In this case, a simple summation would lead to energy minimum quite quickly and smoothly (no oscillation). Hence, we have implemented the sequential and matrix inversion (conjugate gradient) methods, described above, for collision response. Also, as we are testing a position-based method (see Section 4.8.3), we have used the sequential approach for the whole simulation: we resolve each constraint one after the other through vertices displacement, and update model parameters (velocities, normals, volumes and surfaces) continuously (the response of each constraint is taken into account by the others). Indeed, in the framework of our system (many interacting models at different scales with different types of constraints), inverting the matrix would be too costly.

## 4.8.2 Multi-resolution and global regularisation

The use of different levels of details (LODs) is valuable for segmentation (reduction of the system complexity and sensitivity to local solutions). Also, for dynamic simulation, complexity adaptation has shown a good potential for interactive applications [DDCB01]. In Section 4.4.4, we have presented a global multi-resolution scheme for simplex meshes that preserves shape features. Here, we propose to use the different scales to derive multi-resolution constraints. A straightforward approach is to use them successively, as done in most multi-scale registration methods [LRMK99] [SL96] [RAD03] [SCW01] [HG04] [MT95] [PMTK01]. It means that LODs are used independently, one after the other, to allow a coarse-to-fine optimisation. On the contrary, our approach combines all LODs at the same time. The idea is to quickly propagate constraints from lower resolutions to a current simulation level. This is simply done by linear interpolation using the same technique than for vertices. But here, this is an approximation because vertices are no longer located exactly midway between face centers and vertices (vertices have moved relatively to their neighbours, since resolution increase). However, this approximation is pertinent, assuming that shape constraints have enforced mesh local regularity. Given a vector  $\mathbf{U}_i^r$  attached to the point  $\mathbf{P}_i$  at resolution  $r$  and the mean vector  $\overline{\mathbf{U}}_l^r$  related to the cell (or face)  $l$ , the vector interpolation scheme is given by:

$$\begin{aligned} \text{Resolution increase: } \mathbf{U}_i^{r+1} &= (\mathbf{U}_j^r + \overline{\mathbf{U}}_l^r)/2 \text{ or } \mathbf{U}_i^{r+1} = \mathbf{U}_i^r \\ \text{Resolution decrease: } \mathbf{U}_i^r &= \mathbf{U}_i^{r+1} \end{aligned}$$

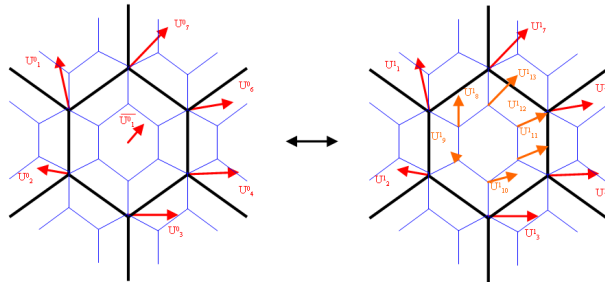


Figure 4.21: Vector interpolation between two resolutions

The Figure 4.21 illustrates how it works for 2-simplex meshes, but the method is similar for curves and volumes. We can use forces, displacements or velocity corrections as vectors. So, all constraints can be finally summed into a unique LOD. It is possible to tune the influence of each scale by modifying the weights

of the interpolation scheme, for instance:  $\mathbf{U}_i^{r+1} = W_r[(\mathbf{U}_i^r + \overline{\mathbf{U}}_i^r)/2]$  where  $W_r \in [0, 1]$  is the weight related to the scale  $r$ . Because our position-based method is sequential, we compute displacements from coarse to fine LODs, model parameters and interpolated displacement being updated continuously. For implicit schemes that make use of force derivatives, we apply the same linear interpolation to the  $3 \times 3$  matrices  $\partial \mathbf{F}_{\mathbf{P}_i} / \partial \mathbf{P}_i$  and sum all the contributions at the current resolution. Here, we can notice that the use of the Jacobi preconditioner (that cancels all non-diagonal terms) is worth because the number of terms in the derivative matrix would increase significantly with only a few LODs, slowing down excessively the CG resolution (for instance, from 4 to 78 for shape forces applied to a 2-simplex surface with 2 LODs). Also, for non-diagonal terms, the definition of the average would not be simple. The most significant benefit of our multi-scale method is related to collision handling: collision detection can be performed at coarser LODs, while the response is being passed to the current resolution. Because collision detection is a very time-consuming task that explodes when the number of vertices becomes high, the advantage is great. Indeed, exact contact computation is most of the time not relevant as fat separates organs. The collision response (using either the weighted sum, sequential or matrix inversion methods) is computed at the collision detection resolution level, and only particle correction (either position correction, velocity correction or forces) is interpolated across LODs. However, we could imagine mixing all constraints into a unique equation to take into account the constraints redundancies across scales. Vertex displacement linearly depends on the displacement of other vertices according to the multi-resolution scheme:  $\Delta \mathbf{P}_i = \sum_{j \in \eta(P_i)} \lambda_j \Delta \mathbf{P}_j$  (where  $\eta(P_i)$  is the set of points related to  $\mathbf{P}_i$  across resolutions). By substitution into Equation 4.4, we obtain the global variation of a constraint  $k$ :

$$\begin{aligned}
E_k(\mathbf{X} + \Delta \mathbf{X}) &\simeq E_k(\mathbf{X}) + \sum_{i \in k} \left( \frac{\partial E_k}{\partial \mathbf{P}_i} \right)^T \sum_{j \in \eta(P_i)} \lambda_j \Delta \mathbf{P}_j \\
&= E_k(\mathbf{X}) + \sum_{i \in k} \left( \frac{\partial E_k}{\partial \mathbf{P}_i} \right)^T \sum_{j \in \eta(P_i)} \lambda_j \sum_{l \ni j} \mathbf{M}_j^{-1} m_l \Delta E_l \frac{\partial E_l}{\partial \mathbf{P}_j} = 0 \\
\mathbf{C}_k(\mathbf{X} + \Delta \mathbf{X}) &\simeq \mathbf{C}_k(\mathbf{X}) + \sum_{i \in k} w_{ik} \sum_{j \in \eta(P_i)} \lambda_j \Delta \mathbf{P}_j \\
&= \mathbf{C}_1(\mathbf{X}) + \sum_{i \in k} w_{ik} \sum_{j \in \eta(P_i)} \lambda_j \sum_{l \ni j} \mathbf{M}_j^{-1} \mathbf{m}_l \Delta \mathbf{C}_k w_{jl} = 0
\end{aligned}$$

Still, this is a linear combination that can be solved through matrix inversion. But here, constraints can be defined at different scales. By the way, we did not choose this approach for computational speed purpose (we use the weighted-sum technique for the force-based approach and the sequential technique for the position-based approach). But, it could fit to other applications.

This multi-resolution framework allows a regularisation of the constraints that is scalable (weighting the resolutions changes the rigidity of the models). However, for local constraints such as image constraints, this is not suited (they do not depend on neighbour vertices). Also, because image constraints are noisy, we need to regularise them globally. Table 3.6 gives us the homogenous transforms that approximate the best a given vector field. Such as in [Mon99] [MD05], we regularise target positions through the linear combination:  $\tilde{\mathbf{P}}_i = \mathbf{P}_i + \lambda d_i \mathbf{n}_i + (1 - \lambda) \mathbf{M} \mathbf{P}_i$  where  $\mathbf{M}$  is the closest transform (translation, rigid, similitude or affine) with regards to all displacements  $d_i \mathbf{n}_i$ .  $\lambda \in [0, 1]$  is a parameter that can regulate the effect of image constraints (from global to local).

### 4.8.3 Tests

In this Section, we are testing different evolution processes that compute the new particle state vector  $\mathbf{Q}_{t+dt} = [\mathbf{P}_{t+dt}, \mathbf{V}_{t+dt}]$  given the forces  $\mathbf{F}_t$ , the derivatives  $\partial \mathbf{Q}' / \partial \mathbf{Q}$ , the masses  $\mathbf{M}$ , the time-step  $dt$  and

the old state vectors. In Section 3.3.5, we have reviewed the existing evolution methods. Particularly, explicit methods advance blindly in time and require small time-steps to remain stable. Implicit methods, on the contrary, use force derivatives to ensure stability for any time-step. Explicit methods are fast for computing each iteration (direct calculation) representing a small advance in time, while implicit methods are slow (inversion of a large sparse matrix) for every large progression in time. So, there is no evidence of which method is the best in terms of its maximal ratio  $dt/dct$  ( $dct$  is the computation time for each iteration). Also, we need to assess the convergence speed (accuracy of the method) since large time-steps are not especially accurate (e.g. more collisions, larger penetration depth). We are testing the following evolution methods:

- **Explicit Euler:** This is the fully explicit scheme where velocities and forces are taken at time  $t$  (see Equation 3.1).
- **Semi-implicit Euler:** Velocities are taken at time  $t+dt$  for expressing the new positions (see Equation 3.3). This scheme is used in [PMTK01] and [MD05] for deformable model segmentation.
- **Verlet:** Former positions at time  $t - dt$  are used to compute the new velocities.
- **Runge-Kutta 2:** This is the standard explicit mid-point method (see Equation 3.2), including two force evaluations.
- **Runge-Kutta 4:** It uses four evaluations of the forces to better extrapolate velocities and positions.
- **Implicit Euler:** This is the general implicit scheme described by Equation 3.5 with  $\alpha = 1$ .
- **Implicit Midpoint:** This is the general implicit scheme described by Equation 3.5 with  $\alpha = 1/2$ .
- **BDF:** This is the general BDF scheme described by Equation 3.6 with  $\alpha = 1/\sqrt{3}$  (third order accuracy [VMT05]).
- **Newton-Raphson:** Here we successively minimise the constraint energy (or vector) through Newton-Raphson steps [MHHR06] and apply constraints directly on particles. Several iteration loops are applied for internal constraints, contrary to external constraints that are computed once. Subsequently, velocities are computed with a standard explicit step.

All methods make use of a weighted sum of forces, except the last one that compute sequentially position corrections (see previous section). For all implicit methods, we use the Jacobi preconditioner to speed-up the computation as we are more interested by speed than realism. We have verified that the convergence speed was not significantly affected by the loss of accuracy (the Jacobi preconditioner removes all non-diagonal terms  $\partial F_i/\partial P_j$ ,  $i \neq j$  of the stiffness matrix). The damping factor is kept equal to  $\gamma = 0.5$  for all tests. For explicit schemes, we apply isotropic damping which is more stable. Anisotropic damping (more accurate) is used for implicit schemes. As described in Section 3.3.5, the conjugate gradient algorithm is used to invert the system  $\mathbf{HX} = \mathbf{Y}$ . The error factor  $\beta$  decreases quickly until a predefined threshold  $\varepsilon = 0.05$  is reached. We give the general pseudo-code of the conjugate gradient method. This is similar to [VMT00b] and [She94]:

```
// Conjugate gradient algorithm
 $\beta = 0$ ;  $\mathbf{X} = \mathbf{0}$ ;  $\mathbf{R} = \mathbf{Y}$ 
while  $\beta > \varepsilon$ 
     $\alpha = \mathbf{R}^T \mathbf{R}$ 
    if ( $\beta \neq 0$ ) then  $\mathbf{T} = \mathbf{R} + (\alpha/\beta)\mathbf{T}$  else  $\mathbf{T} = \mathbf{R}$ 
     $\beta = \mathbf{T}^T \mathbf{H} \mathbf{T}$ 
     $R = R - (\alpha/\beta)\mathbf{H} \mathbf{T}$ 
     $X = X + (\alpha/\beta)\mathbf{T}$ 
     $\beta = \alpha$ 
```



We look at several scenarios that cover the typical situations encountered in musculoskeletal segmentation. Note that for all timing tests, it is important to use high precision clocks such as <sup>1</sup>.

### Shape denoising:

We consider a 2-simplex sphere (3840 vertices and 1922 faces) perturbed with noise and check how it recovers its initial shape through shape and volume conservation forces (See Figure 4.22 and Video 4.3 ). To ensure stability of some integration methods, the stiffness  $\alpha' = \gamma\alpha$  of shape memory constraints is altered through the parameter  $\gamma \in [0, 1]$ . The contribution of volume conservation constraints is kept to  $\gamma = 0.01$ . Convergence is established when the average vertex displacement is  $< 1\mu m$ . The model being composed of 4 level of details, we assess the use of multi-resolution forces (with equal stiffness for all levels). In addition, we compare the application of independent constraints on each particle and the application of constraints on multiple particles (second method) as described above. To prevent from instabilities due to singularities in the computation of the  $\lambda_i$ , we switch to the *independent constraints* mode ( $\lambda_i = 0$ ) when  $\sum_3 \lambda'_i < 0.2$ . Convergence times are summarised in Table 4.27.

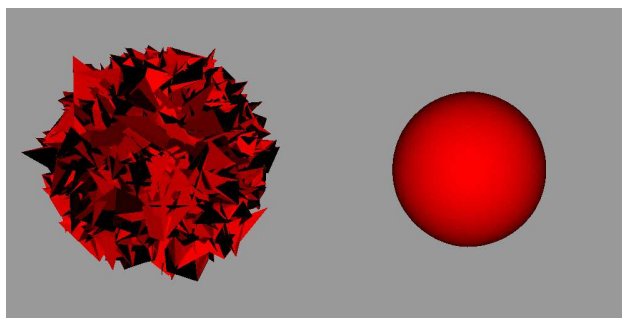


Figure 4.22: Shape denoising test (left: initial state; right: state after convergence)

	Computation time per it. (in ms)	Convergence time (in s)	$\gamma$
Explicit Euler	3.26	11.7	0.01
Semi-implicit Euler	3.26	2.7	0.2
Verlet	3.11	2.9	0.2
Runge-Kutta 2	6.55	6.3	0.1
Runge-Kutta 4	13.21	12	0.3
Implicit Midpoint	4.14	2.8	0.01
BDF	4.37	3	0.01
Newton-Raphson	7.76	2.4	1
Newton-Raphson (multires)	12.39	2	1
Newton-Raphson (multires and multiparticle)	21.06	4.8	1
Implicit Euler	5.74	1.2	1
Implicit Euler (multires)	6.21	0.9	1
Implicit Euler (multires and multiparticle)	9.03	2	1

Table 4.27: Convergence times (in s) for the shape denoising test (sphere)

In this simple test, we see that the Newton-Raphson and all implicit and semi-implicit schemes are competi-

<sup>1</sup><http://www.codeproject.com/datetime/perftimer.asp>

tive. The use of multi-resolution forces slightly improves the computation time. Because noise produces local shape perturbations, local constraints applied to the highest resolution are relevant (we will see later that our multi-resolution scheme is more useful in case of global perturbations). The expression of constraints for several particles is not useful here (the gain in accuracy does not compensate the increase of complexity).

Now, we perform the same test with a model owning a medial representation (biceps femoris muscle with three LODs) and compare shape memory and radial constraints (See Figure 4.23 and Video 4.4 ). The model is composed of 3808 (1906) vertices (faces) and its medial axis of 302 (152) vertices (faces). So, shape memory constraints are based on  $3808 \times 3 = 11424$  shape parameters while radial constraints are based on  $302 \times 4 = 1208$  parameters (about 10 fold reduction of prior shape information). We are testing the three different methods for computing radial constraints (see Section 4.6.4). These forces are global (no dependence with vertex neighbourhood), so their application across levels of details has no sense. We measure the accuracy of the matching as the average vertex displacements from the initial (unnoised) configuration (see Table 4.28).

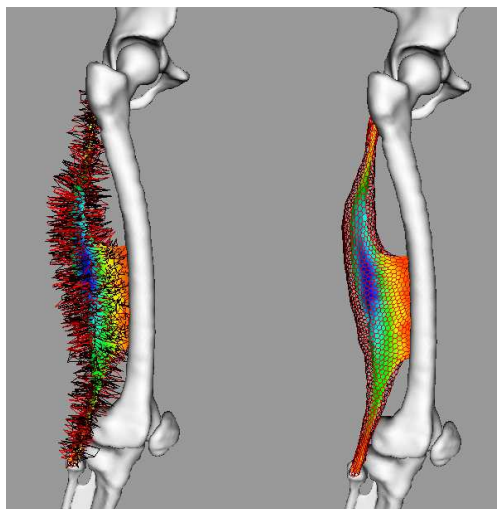


Figure 4.23: medial-axis-based shape denoising test (left: initial state; right: state after convergence)

	Shape memory	Rad. constraints (closest pt)	Rad. constraints (rad.)	Rad. constraints (rad. and bary. coord.)
Implicit Midpoint	0.9s	19s	1.9s	0.3s
BDF	1s	13.2s	2.1s	0.3s
Newton-Raphson	2.2s	15.7s	5.3s	0.4s
Newton-Raphson (multires)	1.7s	-	-	-
Implicit Euler	1s	12.8s	1.9s	0.7s
Implicit Euler (multires)	1.8s	-	-	-
Accuracy $\pm$ stddev. (in mm)	$0.3 \pm 0.2$	$2 \pm 1.5$	$1 \pm 0.6$	$2.3 \pm 2.7$

Table 4.28: Convergence times (in s) for the shape denoising test (biceps femoris)

The three methods for computing radial constraints are increasingly stiff. With the *closest point* methods (resp. *radius* method), vertices are allowed to slide on maximal spheres (resp. on maximal interpolated

spheres). So, smoothing forces are applied with a small contribution (i.e.  $\gamma = 0.1$ ) to ensure local regularity. As expected, the *closest point* method is not enough accurate and stiff. Results show that the *radius* method is more accurate, even if slower than the *radius and barycentric coord.* method. This last method is very stiff (exact global positions are imposed) and is, in fact, less stable than the others. It starts to be excessively inaccurate when simulating model bending or other configurations that are far from the reference one. So, we will choose the *radius* method for segmentation to allow a better local continuity. We think that its slightly lower accuracy and higher computational weight compared to shape memory-based constraints remains acceptable and is worth given that we tremendously reduce the amount of information required to represent shapes. Concerning numerical integration methods, the *implicit Euler* method seems the best one for our purpose in terms of accuracy/ computational speed/ stability. In practice, this has been corroborated for modelling and segmenting the musculoskeletal system from MRI. In the remaining of this these, we will adopt this method.

### Shape matching:

Here, we assess how a model can recover its shape when it is initially deformed with a global transform (See Figure 4.24 and Video 4.5 ). The femur model is composed of 8224 (4114) vertices (faces) at highest resolution (out of three resolution levels). We apply shape memory forces ( $\gamma = 1$ ) distributed or not across levels of details. After two minutes of computation, we compute the error with respect to the reference (undeformed) model. Because shape memory and volume conservation forces are invariant through rigid transforms, the reference model is rigidly registered with the iterative closest point method before computing the average distance towards it. The use of our multi-resolution framework greatly improves the results (error of  $2mm$  vs.  $13mm$ ) as shown in Figure 4.24 because global and local shape forces are used simultaneously. Once again, shape forces formulated for several particles was found not useful (the gain in accuracy is not relevant compared to the increase of computational time). Shape forces applied independently on each particle are stable when models are regular (no ghost spinning or strain) meaning that momentum are more or less conserved. For irregular meshes, the use of shape constraints may produce some oscillations and spinning due to large metric and angular parameters (small geometrical changes produce large variations in the forces). However, the models we are considering are regular (their construction has been made by adequately applying smoothing forces). Thus, we will always consider 2-simplex mesh internal forces as independent forces. On the contrary we will keep using the formulation with multiple particles for interaction forces in order to ensure a precise collision response and radial constraints.

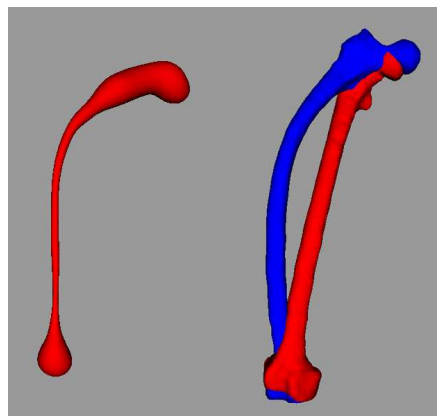


Figure 4.24: Shape recovering through local (blue) and multi-resolution (red) shape memory forces. Left: initial state; Right: state after 2 minutes computation

We have tested the use of 3-simplex meshes for shape matching (see Figure 4.25). We initially attach a

point to perturb shapes and then simulate models with shape memory forces. Because these constraints are scale invariant for 3-simplex meshes, we apply volume conservation forces at their boundaries (which is a 2-simplex mesh). We consider forces from the interaction between neighbouring particles as expressed in Table 4.25. Considering independent forces only ( $\mathbf{F}_{\mathbf{P}_i} = \mathbf{0}$ ) leads to close results than 2-simplex meshes (initial sphere). Multiple particle forces, on the contrary, converges to a shape that is physically more plausible if we imagine a soft object under plastic deformation (see Figure 4.25). Weighting the two results (infinitely stiff material and plastic material) could lead to interesting methods for fast physically-based simulation (moreover, simplex meshes offer simple expressions for simulating anisotropy). For our purpose (segmentation and modelling), we will only consider simplex surfaces that are much more efficient in terms of complexity.

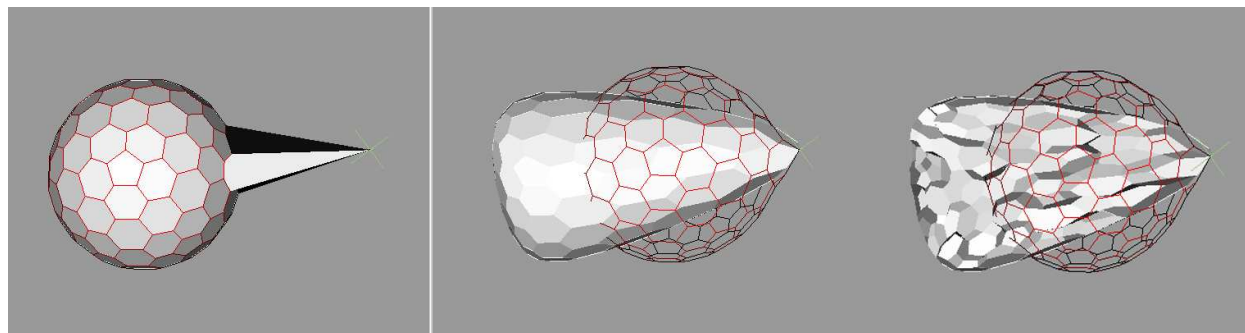


Figure 4.25: Comparison of shape recovering processes using a 2-simplex or a 3-simplex mesh. Left: initial state; Right: Final state

Now, we introduce external forces to fit a deformable model to a reference model through *closest point* constraints (see Figure 4.26 and Video 4.6 ). The goal is to align two models from two different individuals to find anatomical correspondences. The straightforward application of external constraints, without any regularisation, leads to undesirable results as shown in the figure (right image). On the contrary, a pair and smooth approach that regularises external forces with a gradual increase of the number of degrees of freedom provides consistent results. In this example, we successively apply a rigid regularisation, an affine regularisation and finally no regularisation. From this simple case, we can see how useful can be a global regularisation for geometric or iconic registration. We will extensively use it in the next chapter.

### Collision handling:

Here, we compare the different collision detection and response methods. The first test consists in trapping a sphere (3840/1922 vertices/ faces for the highest LOD out of 3 LODs) between two immobile planes (see Figure 4.27 and Video 4.7 ). The sphere should converge to a state where it is perfectly motionless. The proximity region is set to  $d_{max} = 2mm$  and the gap to  $g = 0$ . Response thresholds are set to  $10mm$ ,  $10mm.s^{-1}$  and  $10mm.s^{-2}$ . We apply the *spring* and *reaction* types of response as described in Section 4.7.2 when vertices are in the collision and proximity regions respectively. We measure the convergence time as the time spent on collision detection/ response before vertices become still (average displacement  $< 5\mu m$ ). As shown in Table 4.29, some methods do not strictly converge because vertices oscillate (displacement  $\simeq 30\mu m$ ). Force alteration is performed smoothly (we add a force of stiffness 1 to other forces rather than replacing them).

In this simple test, all collision correction methods are more or less equivalent with a preference on velocity correction (impulse-based method). We see that the improvement in terms of computational time is about 3 times when decreasing resolution. The accuracy (average point-to-point distance) with regards to the finest

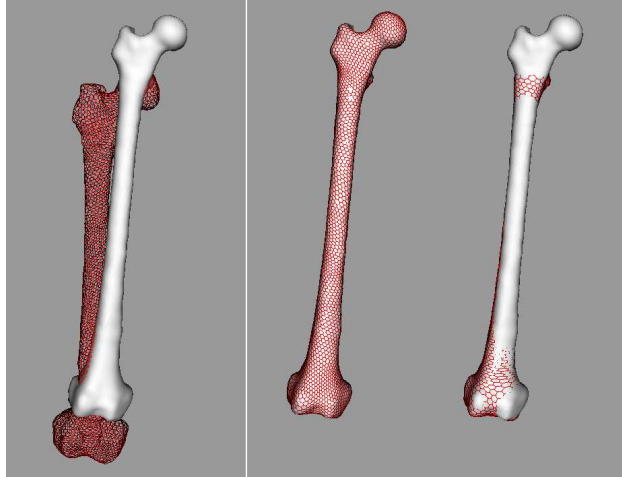


Figure 4.26: Elastic matching between two individual femurs. Left: initial state; Right: final states from respectively global and local regularisation

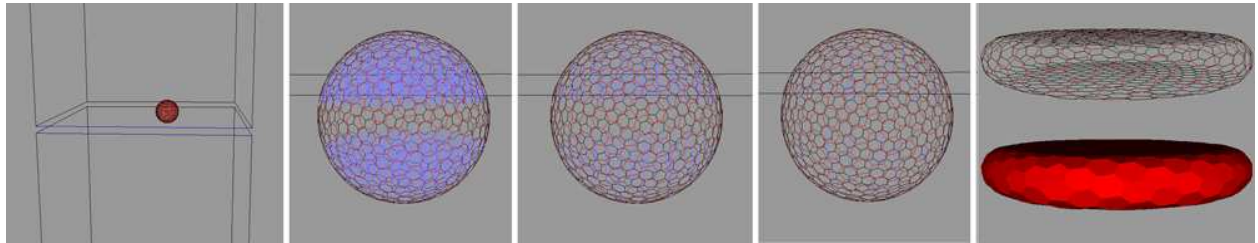


Figure 4.27: From left to right: scene overview; detected collision position vectors at the three different resolutions; final state

LOD	Response type	Weighted sum	Sequential	Conjugate Grad.
0	Position	-	-	-
	Velocity	2.46	5.45	5.72
	Force (1 <sup>st</sup> order)	2.61	5.94	5.61
	Force (2 <sup>st</sup> order)	3.50	7.12	9.37
	Velocity+ Force (1 <sup>st</sup> order)	3.68	10.25	9.26
1	Position	-	-	-
	Velocity	0.62	0.82	0.74
	Force (1 <sup>st</sup> order)	0.77	1.00	0.76
	Force (2 <sup>st</sup> order)	0.83	1.10	1.13
	Velocity+ Force (1 <sup>st</sup> order)	0.77	1.24	1.09
2	Position	-	-	-
	Velocity	0.28	0.29	0.22
	Force (1 <sup>st</sup> order)	0.40	0.43	0.20
	Force (2 <sup>st</sup> order)	0.41	0.44	0.26
	Velocity+ Force (1 <sup>st</sup> order)	0.35	0.38	0.30

Table 4.29: Convergence times (in *s*) for the collision detection test

level is  $2 \pm 0.75mm$  and  $4.5 \pm 1.5mm$  for the resolution 1 and 2 respectively (weighted sum and sequential methods). The conjugate gradient method provides a higher accuracy ( $2 \pm 0.3mm$  and  $2.2 \pm 0.9mm$ ) but is less stable (the final vertex displacement is about  $2\mu m$  compared to  $1\mu m$  with other methods) which is not prohibitive. Because this scenario is simple (no multiple antagonist collisions), a simple resolution method such as the weighted sum works fine, and this would be in general the case in musculoskeletal segmentation. For coarsest LODs, the conjugate gradient can benefit from its accuracy to converge quicker (despite its higher computational time for each iteration). We think that the use of velocity correction with a weighted sum resolution is the best in practice, especially for large system, as the trade-off between accuracy, convergences speed and stiffness is larger. The position correction is not necessary as it introduces undesired oscillations.

Now, three deformable objects in contact (8960/4486 vertices/ faces in total) owning medial axis (360/183 vertices/ faces in total) collide under the action of volume preservation forces (see Figure 4.28 and Video 4.8 ). We set target volumes to be 1.5 times the initial volumes and simulate the system until convergence (average displacement  $< 5\mu m$ ). Medial axis stiffness are set to 0.5 for shape memory and radial forces. Model stiffness are set to 0.6, 0.3 and 0.1 for radial, smoothness and volume conservation forces respectively. Medial axis radii are updated at each iteration (360 parameters). Note that the system is uniquely defined through medial axis shape parameters ( $360 \times 4 = 1440$  parameters). We compare convergence times (time spent on collision handling), the accuracy (average point-to-point and point-to-surface distances with regards to the most accurate case) and the stability (remnant average vertex displacement) of different collision detection methods and resolutions. Particularly, we check different features for detecting collisions between objects (each object is tested towards other objects resulting is 6 object-to-object tests): the standard *model-to-model* collision detection, our *model-to-medial axis* scheme (each model point is tested towards medial axis) and our *medial axis-to-medial axis* collision detection method (see Section 4.7.2). We also check three types of collision detection schemes: the *full* detection, the *update* detection (from on initial full detection, we only check colliding/ proximity points in the next detections) and the *reference* matching (from on initial full detection, we enforce objects relative positions as described in Section 4.7.2). The two last schemes rely on the hypothesis of permanent sliding contacts and permanent non-sliding contacts respectively. Results are given in Table 4.30.

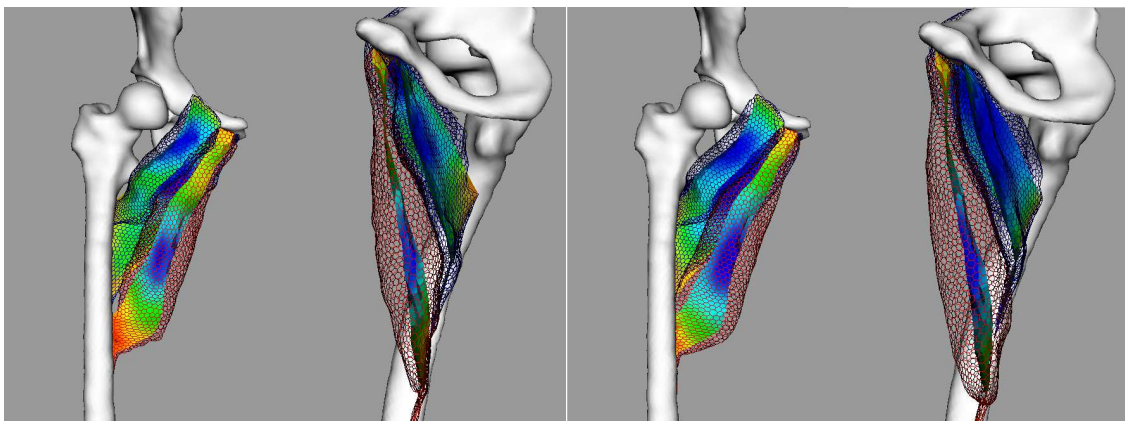


Figure 4.28: Medial axis-based collision detection example. Left: initial state; right: final state

The computation of updated and reference collisions are about 3 times faster than the full detection. So it is valuable to use them when contacts are permanent (e.g. during the adaptation of a generic model), the full detection being useful in the initial step of generic models construction. In this example, where models do not need to slide on each other, the update and reference modes are more-or-less equivalent in

Collision features	Collision detection method	LOD	Comp. time per it. (in <i>ms</i> )	Convergence time (in <i>s</i> )	Stability (in $\mu m$ )	Pt-to-pt error (in <i>mm</i> )	Pt-to-surf error (in <i>mm</i> )
Model-Model	Full	0	327.37	71.70	0.30	0.00	0.00
		1	56.46	8.71	0.16	1.41	0.24
		2	26.51	1.40	0.09	2.21	0.64
	Update	0	150.44	26.77	0.24	1.67	0.37
		1	31.82	2.67	0.19	1.64	0.41
		2	21.83	0.52	0.10	1.92	0.63
	Ref	0	155.48	29.41	0.10	1.71	0.41
		1	31.94	2.49	0.09	1.71	0.44
		2	22.00	0.52	0.10	1.91	0.61
Model-MA	Full	0	164.39	41.61	0.43	2.11	0.96
		1	47.71	6.50	0.26	2.36	0.81
		2	27.10	1.87	0.10	2.38	1.00
	Update	0	66.63	11.77	0.21	2.21	0.70
		1	28.50	2.40	0.67	3.40	0.83
		2	22.76	0.98	0.10	3.67	0.98
	Ref	0	71.02	-	51.50	2.56	1.10
		1	29.34	2.37	0.18	3.37	0.79
		2	22.67	0.95	0.10	3.67	1.06
MA-MA	Full	0	35.48	3.36	0.34	3.36	1.58
	Update	0	21.76	0.53	0.19	3.36	1.54
	Ref	0	21.44	0.53	0.18	3.12	1.41

Table 4.30: Results for the medial axis-based collision test

terms of stability and accuracy. Moreover, here, there is no fat separating organs, so standard collision detection enforcing a null inter-organ distance is valuable. Once again, we see the utility of multi-resolution collision detection (computational time improvement of a factor from 3 to 10 times) without prohibitive inaccuracies. The medial axis-based collision detection also improves greatly computational times. The use of model-MA collision detection approximately divides computational costs by 2, while MA-MA detection divides them by about 40. The last ones gives approximately the same results than the standard detection with coarsest LODs but is less accurate (because the error due to medial axis approximation is higher than the approximation by resolution decrease).

Here, we compare our collision detection method between deformable and rigid objects (distance field-based method) and the method between deformable and deformable models (bounding volume hierarchy). We take, as an example, a muscle model (4896/2450 vertices/ faces) that collides with two bones (21248/10626 vertices/ faces and distance field of  $1mm$  resolution) under the action of volume preservation forces (see Figure 4.29 and Video 4.9 ). We use a gap of  $2mm$  between bones and muscles. We check the stability, the accuracy and convergence times as shown in Table 4.31.

Our distance field-based collision detection scheme shows a clear improvement over the bounding volume hierarchy methods (about 4 times improvement in terms of computational time, while accuracy and stability are comparable). We have checked that increasing the resolution of the distance field does not improve results much, since the tri-linear interpolation provides a good estimate of such continuous data.

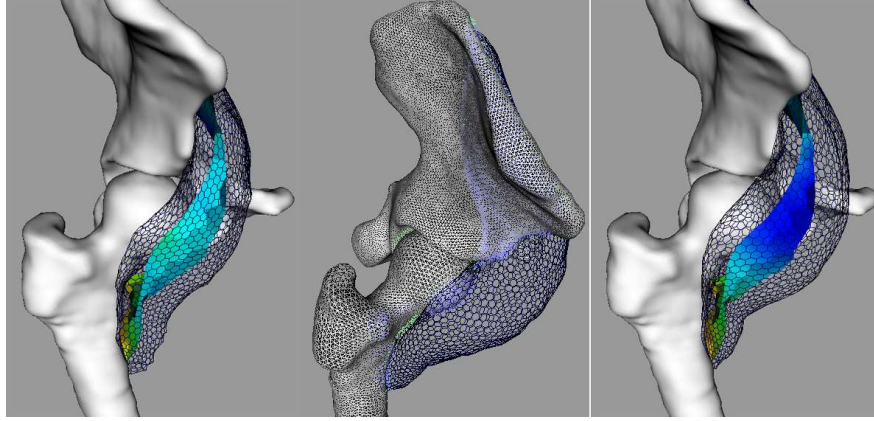


Figure 4.29: Scenario for demonstrating our distance field-based collision detection scheme. Left: initial state; middle: collisions (blue) and proximities (green) if collisions would not be handled; right: final state after collision handling

Collision detection method	LOD	Computation time per it. (in ms)	Convergence time (in s)	Stability (in $\mu m$ )	Error (in mm)
BVH	0	513.55	23.18	0.10	0.00
	1	81.79	7.06	0.10	0.38
	2	24.32	2.15	0.10	0.99
distance field	0	49.11	8.93	0.10	0.47
	1	16.98	1.52	0.10	0.19
	2	13.74	0.53	0.10	0.42

Table 4.31: Results for the distance field-based collision test

### Full simulation:

In this last test, we simulate all muscle models of the hip and the thigh: 21 models possessing medial axis (69184/35708 and 4272/2157 vertices/ faces in total). We perturb the system by simulating a global movement (20deg of flexion) using our skinning algorithm (Section 5.9) and check the recovering of the initial vertex positions under the action of internal forces only (see Figure 4.30 and Video 4.10 ). We use MA/MA and model/bone reference positions, and velocity correction to handle collisions. The different resolutions are applied sequentially to speed up the convergence. This time medial axis are driven by shape parameters rather than model surfaces. So, for medial axis, force stiffness are turned to  $\gamma = 0.9$  and 0.1 for shape memory and radial constraints respectively.

Our model converges after about 2500 iterations corresponding to 160s of computation. In the video, visualisation is performed every 10 iterations, which allows acceptable interactivity: the computation times per iteration are 0.25s, 0.06s and 0.02s for the three resolutions respectively. Figure 4.31 shows the repartition of the computational time across the different tasks. Because we use reference inter-organ positions, collision detection time is very low. The collision response time is higher (about 15% higher) when using model-model detection because the number of collisions increases significantly. When keeping a low resolution, model-model and MA-MA collision detections are equivalent. We have found that the (low cost) *weighted-sum* collision handling method provides sufficient accuracy. The *conjugate gradient* resolution becomes inaccurate (oscillations) and computational costly when the system is stiff (such as the one using reference positions). The final error measured with the initial unwrapped model is  $0.8 \pm 1.0mm$  which roughly corresponds to



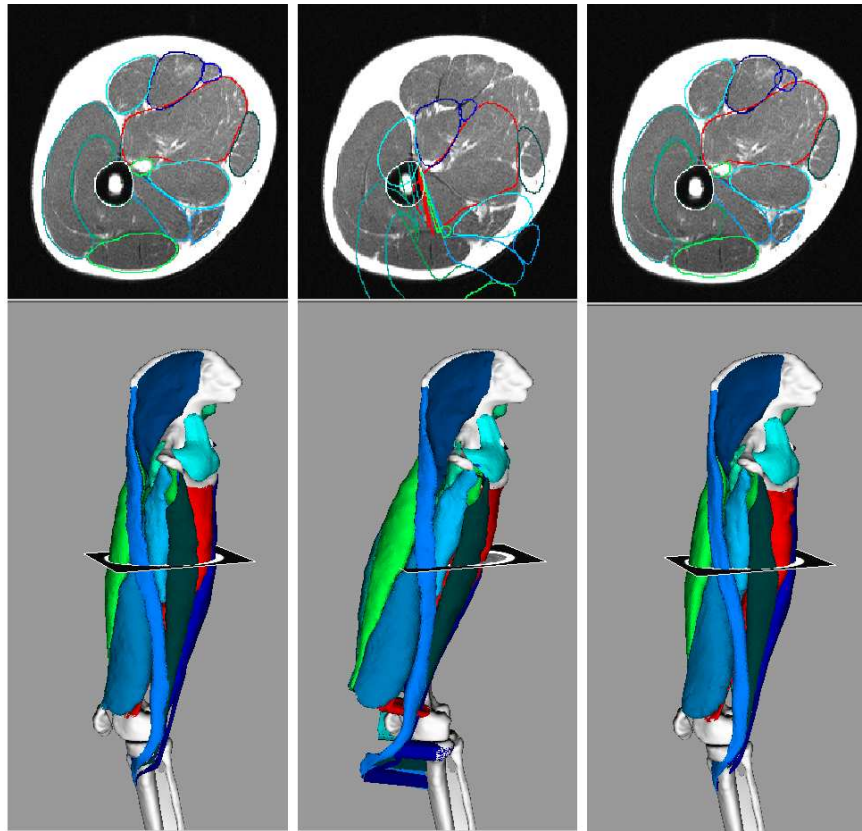


Figure 4.30: Full musculoskeletal segmentation test. Left: target reference models; Middle: initially perturbed models; Right: final state after applying medial axis-based internal forces

the medial axis approximation error (see Section 5.4.3). For registration, we expect that this error will be recovered through external image-based forces.

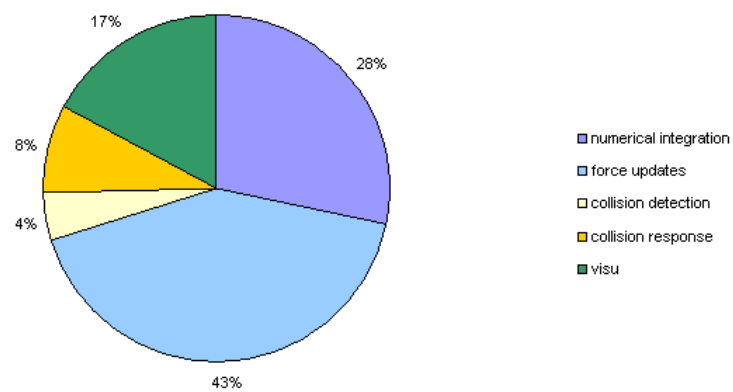


Figure 4.31: Repartition of computational time for the different tasks when using reference collisions

## 4.9 Conclusion

The different facets of discrete deformable models have been developed in this chapter: their construction (mesh connectivity optimisation), their geometric description, the constraints that can enforce their regularity and fitting to image data, and finally their simulation. We have focussed on the aspects that are useful for fast musculoskeletal segmentation [GMMT06]: the relevant geometric features to be enhanced such as model smoothness, volume preservation and non-penetration, the fast computation of particle positions from constraints and the reduction of degrees of freedom through regularisation. Indeed, the reduction of parameter space is very important in segmentation in order to speed up convergence and represent the state (geometry) of the system with the fewest independent values (to allow efficient shape analysis and statistical comparisons across time or individuals). A natural extension of our work is to add a statistical layer to would compute the relevant prior knowledge to use (medial axis shape parameters) from examples in order to better parameterise the problem. We have abstracted the local nature of shape representation through two different mechanisms: multi-resolution models (hypothesis that the importance of shape information decreases when details increase) and medial representation (hypothesis of thickness continuity and topological conservation).

In our study, we have chosen the simplex mesh representation. But, of course, equivalences of most of our methods can be found for other representations such as triangle/ tetrahedral meshes and even parametric models. We believe that simplex meshes offer a better flexibility and efficiency due to its simple geometric description. This is not necessarily true if we think about other applications that would extensively use graphics hardware, physical constraints (e.g. FEM) or simple shapes (that can be accurately modelled with parametric functions). We have tried to bridge volumetric and surfacic representations into the same framework: 3-simplex mesh deformations are derived from internal volumetric forces *and* external boundary forces. We think that this is a particularly promising domains, not yet deeply explored. Indeed, modelling (surfacic models and geometric constraints) and simulation (volumetric models and physically-based constraints) tasks are generally separated. Expressing shapes and constraints within the same framework, would allow to tune the data-driven/ predictive aspects of model deformations and therefore to better parameterise/ validate physically-based models.

From our experience, it is clear that the multi-disciplinary nature of medical image processing is increasing. Connections between different science domains are more and more expected: particularly computer graphics (discrete geometry and visualisation), numerical analysis (optimisation and numerical integration), information theory (statistics), physics (continuum mechanics) and signal processing (image analysis). In future, a big challenge is to understand how lower scale and physiological phenomena (e.g. biological interactions, gene expressions, electrophysiological process) lead to our macroscopic observations. Particularly, understanding the links (more-or-less continuous) between the different scales (through complexity compliant models) should receive a growing attention. This is an essential aspect of future diagnosis and therapy tools.

## Chapter 5

# Application to musculoskeletal MRI segmentation



---

<sup>1</sup>Jacques Gamelin. Nouveau recueil d'ostologie et de myologie. Toulouse, 1779, Etching, National Library of Medicine [http://www.nlm.nih.gov/exhibition/dreamanatomy/da\\_g\\_III-B-15.html](http://www.nlm.nih.gov/exhibition/dreamanatomy/da_g_III-B-15.html)

## 5.1 Problematic

We have seen in chapter 2 that orthopaedics and biomechanicians are in need of computer-aided tools for musculoskeletal examination (from the analysis and prediction points of view). Current techniques are not well integrated in the clinical daily practice because not enough automatic (most of them are difficult to parameterise for non-computer scientist users), robust (lack of flexibility with regards to image and morphology variations), fast and accurate. As opposed to the neurology and cardiology application domains, musculoskeletal research has not yet taken the benefit of the recent advances in medical image acquisition due to its complexity (large displacements/ variability and large scales, large number of interdependent organs difficult to delineate in images). As a result, the current clinical practice remains mainly empiric, and the origin of most diseases (e.g. arthritis, muscle injuries) is not known yet. A better understanding of musculoskeletal function is directly related to the advances in the medical image processing field. Current clinical procedures (diagnosis, pre-operative planning and post-operative guides) could be greatly improved by visualising and analysing individual musculoskeletal geometry extracted by image processing software. The next step is to use extracted models for prediction (biomechanical simulation). For it, *real observed kinematical data* is useful for validation and parameterisation. The challenge in musculoskeletal extraction is to maximise the trade-off between robustness (noise-sensitivity reduction through regularisation techniques) and flexibility (to much regularisation should not prevent from extracting individual features and abnormal morphology). In parallel to shape extraction, statistical studies are needed to characterise musculoskeletal variability. Note that establishing spatial correspondences between models and reducing shape parameters during segmentation (registration) facilitate further statistical analysis studies.

In the previous chapter, we have introduced generic computer graphics techniques that can be applied to image-based musculoskeletal modelling and geometric-based musculoskeletal simulation. We have presented how to compute and integrate forces from iconic and geometric constraints. Geometric simulation is in principle faster but less accurate than physical-based simulation (forces are computed from applying constitute laws to strain), so we want to evaluate it with respect to measurements. In this chapter, we propose to test our algorithms on different kinds of MRI data. Particularly, we will assess their accuracy (comparison with manual segmentation), their robustness (test on different individuals, postures and MRI sequences), their predictive power (difference between image-driven and geometry-driven musculoskeletal modelling) and their performance (computational weight). In the framework of our project, static, sequential and dynamic MR images have been acquired to improve standard hip examination procedures. Because of scanner availability restrictions, the spatial resolution is decreased when the temporal resolution is increased. To regularise segmentation of low resolution images, we use prior knowledge from previous registrations (see Figure 5.1).

In this chapter, we also introduce specific techniques for hip morphological analysis. The goal is to enhance and automate standard clinical measures which is possible when performing registration (known anatomical correspondences between individuals and through time).

## 5.2 Acquisition protocols and description of the data

MRI protocols definition has been done in close collaboration with physicians from radiology and orthopaedic departments (HUGE - Geneva). The goal was to obtain images carrying sufficient information with clinically achievable protocols (fast). Acquisition has been performed on a 1.5T Intera MRI system (Philips Medical Systems, Best NL). All images were made available in the standard DICOM format<sup>1</sup>. We load them using the well established ITK GDCM loader<sup>1</sup>. To gather image data, we save resulting volumes as VTK *structured points*. Note that intensity values (grey levels) are kept in their original form (*unsigned short*). For non-parallel slices (e.g. radial and dynamic images), we code image orientation (available in the DICOM header)

---

<sup>1</sup><http://medical.nema.org/>

<sup>1</sup><http://www.itk.org/>

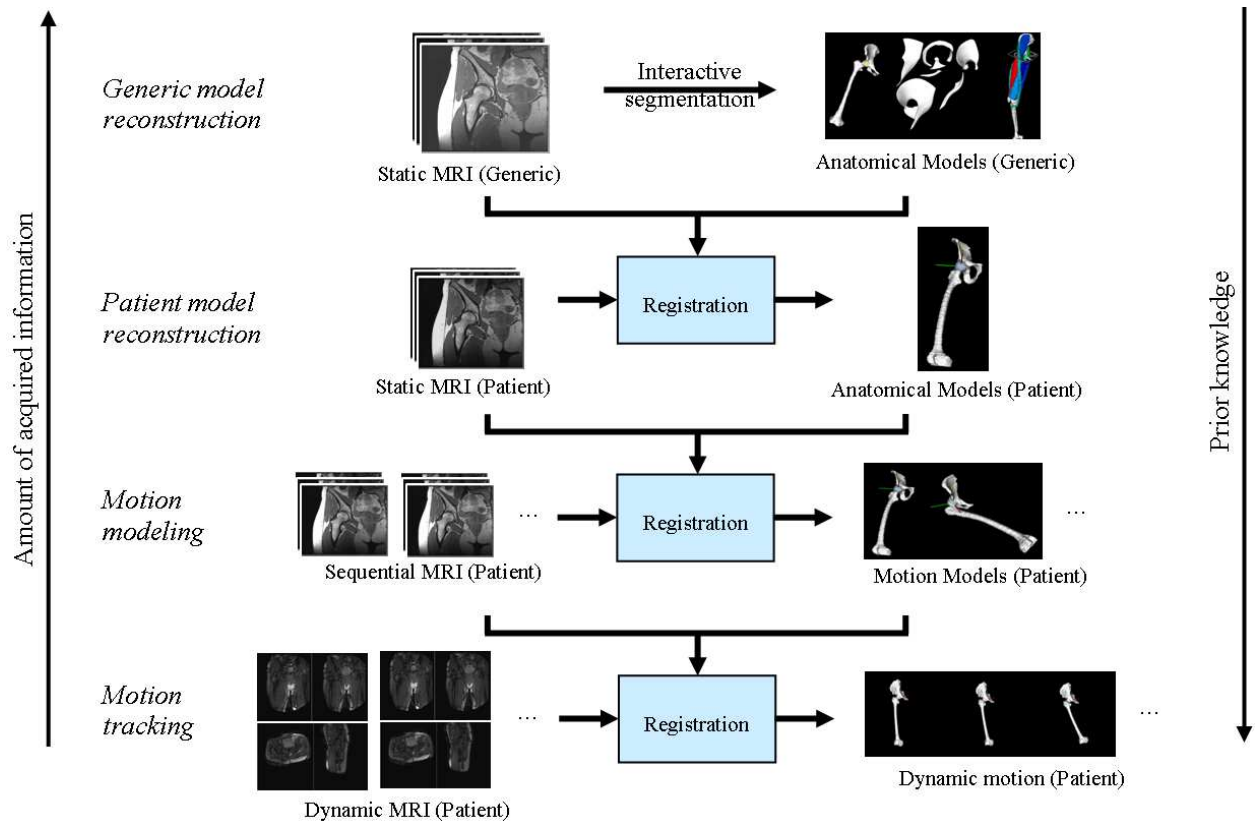


Figure 5.1: Registration processes involved in musculoskeletal modelling

in separate files.

### 5.2.1 Static acquisition

We adapt the standard imaging protocol to achieve the full acquisition of the hip and the thigh. Indeed, the complete bones of the hip joint (pelvis and femur) are required to be able to place the standard coordinate systems based on anatomical landmarks [WSA<sup>+</sup>02] (see Section 5.6). Due to restricted acquisition time in the clinical environment, it is not possible to achieve a precise (isotropic) acquisition over the full region. Instead, we adjust the slice thickness (from  $2mm$  to  $10mm$ ) according to the region of interest (see Figure 5.2). Another sequence, specific to the cartilage, is also achieved. In total, they make around 250 slices and the acquisition lasts about  $40min$ . During acquisition, it is important to check that series overlap so that no information is missing (especially with regards to shutter effects), and that volunteers remain still during and between the different series. To improve image quality (SNR), a surface coil around the hip is used. We have applied this protocol on 13 healthy and young volunteers (7 male, 6 female). We have also tested the same protocol for the knee, on 2 generic volunteers (in this case, the acquisition is composed of 3 series: one for the knee ( $2mm$ ), one for the tibia ( $10mm$ ) and one for the ankle ( $4mm$ )).

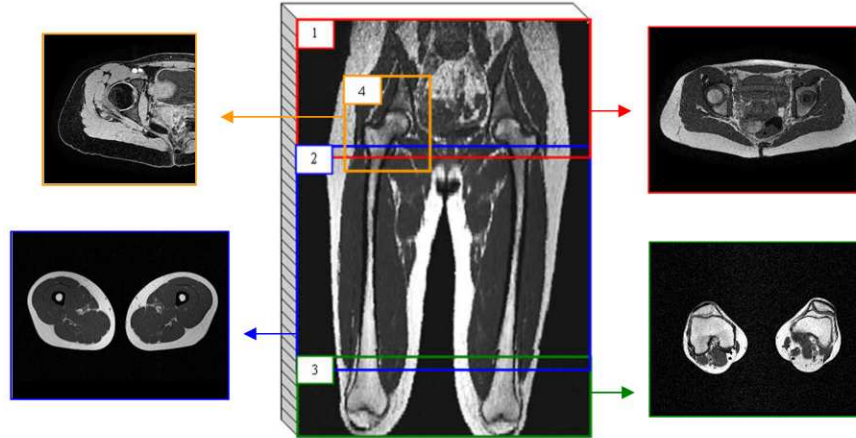


Figure 5.2: Static MRI protocol for the hip

- Axial 2D T1 Turbo Spin Echo (TSE), TR = 578ms, TE = 18ms, FOV = 40cm, Matrix =  $512 \times 512$ , thickness = 2mm (hip (1)), 4mm (knees (3)) or 10mm (thigh (2)), gap = 0mm, FA = 90deg, resolution =  $0.78 \times 0.78$ mm.
- Axial 3D T1 Gradient-Echo (GE), TR = 20ms, TE = 7ms, FOV = 20cm, Matrix =  $256 \times 256$ , thickness = 2mm (4), gap = 0mm, FA = 50deg, NSA = 2, resolution =  $0.78 \times 0.78 \times 2$ mm.

### 5.2.2 Sequential acquisition

Besides these high-resolution acquisitions that provide a set of high-quality models, we have performed a clinical study (with ethics approval from the hospital of Geneva) on young female dancers, where several poses are analysed: the neutral position and the right and left splits (see Section 5.10). Because of tight acquisition time restrictions, the former protocol is not applicable for each posture. A faster (but low-resolution) protocol is proposed consisting in two axial 3D series (acquisition time: 3min). A high resolution sequence centered on joints with isometric voxels is run to improve cartilage/ligament and bone models near articulations (acquisition time: 4min per hip). Moreover, because the purpose is to study the coxo-femoral conflict and labrum deformations, we use radial acquisition that provides highly detailed images cutting through the labrum around the acetabulum (acquisition time: 6min per hip).

Another study has been performed on 6 young volunteers, consisting in analysing bone motion in real-time dynamic MRI during an abduction motion pattern. To validate the tracking, a sequential acquisition is performed at stepped positions, where the two scans are run (stationary joint during the scan). At the same time, in order to assess skin/ bone relative movement, which is important for error reduction in motion capture (see Section 2.3.2), optical markers with injected contrast agent are placed onto the skin and acquired [YCGMMT04] [YCMT06] (see Figure 5.4). To be able to reproduce the same motion patterns across volunteers, a positioning device is used to block the leg at the different step. A fast MRI protocol allowing accurate bone tracking is applied based on a fast gradient echo sequence (acquisition time: 2.5min).

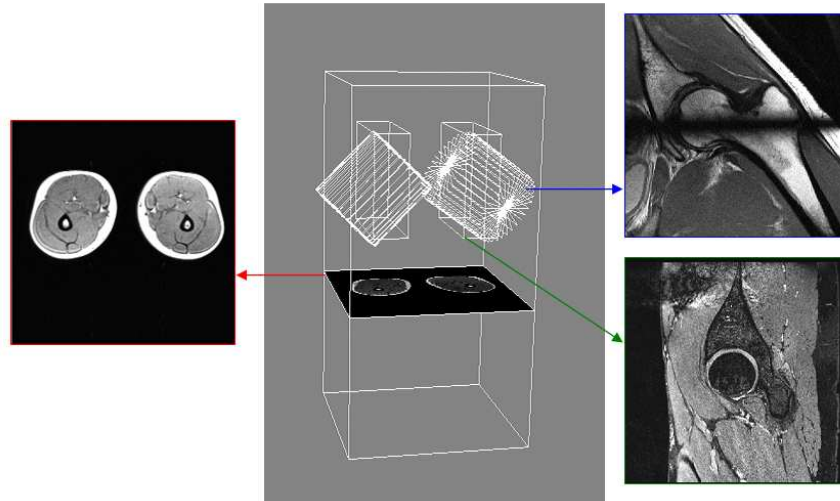


Figure 5.3: Sequential MRI protocol for the analysis of coxo-femoral conflicts

- Low-resolution: axial 3D T1, TR = 4.15ms, TE = 1.69ms, FOV = 35cm, Matrix = 256 × 256, thickness = 5mm, gap = 0mm, FA = 10deg, resolution = 1.37 × 1.37 × 5mm.
- High resolution: sagittal 3D T2\* TrueFISP, TR = 10.57ms, TE = 4.63ms, FOV = 20cm, Matrix = 384 × 384, thickness = 0.6mm, gap = 0mm, FA = 28deg, resolution = 0.52 × 0.52 × 0.6mm.
- Radial acquisition: 2D TSE, TR = 2180ms, TE = 13ms, FOV = 16cm, Matrix = 384 × 384, thickness = 3mm, gap = 0mm, FA = 180deg, resolution = 0.41mm × 0.41mm × 10deg, number of slices = 18.

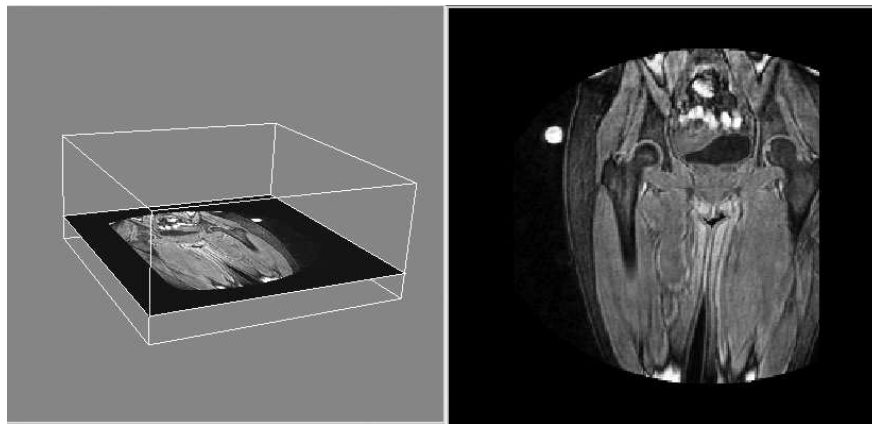


Figure 5.4: Sequential MRI protocol for bone motion analysis

- Low-resolution: Coronal 3D T1 fast gradient echo, TR = 6.4ms, TE = 3.1ms, FOV = 50cm, Matrix = 512 × 512, thickness = 4mm, gap = 0mm, FA = 15deg, resolution = 1 × 1 × 4mm.

### 5.2.3 Dynamic acquisition

Currently, the hip is examined in static MRI volumes and by manual inspection. However, the literature suggests that dynamic MR imaging would improve the ease of diagnosis of joint abnormalities [QLH<sup>+</sup>02]. However, there are technical limitations with the use of dynamic MRI. The speed of acquisition is limited by gradient strength and slew rate, which cannot exceed a safe limit. There is also a trade-off between Signal-to-Noise ratio, spatial resolution and temporal resolution. The state of the art frame rate for 2D real-time MRI imaging (for an in-plane spatial resolution of around  $2mm$ ) remains at around  $4frames/s$ . However, in order to acquire adequate functional information it would be necessary to acquire a volume. This would reduce the frame rate to an unacceptably low level for motion to be captured.

#### In-vitro study

In a first step, the b-FFE (balanced Fast Field Echo, Philips Medical Systems, Best NL) imaging sequence (aka. trueFISP) is quantitatively compared to four other sequences, including Turbo Spin Echo (TSE), RF-spoiled FFE (T1-FFE) and a Field Echo, Echo Planar Imaging (FE-EPI) sequence. In order to quantify sequence performance, a phantom consisting of tubes of Gd-DTPA (Schering AG, Germany) at varying concentrations is used. Using this phantom, measurements of signal-to-noise ratio (SNR) can be made for a range of physiological T2/ T1 values. The b-FFE sequence is found to outperform all other ultra-fast MR sequences available on the scanner in terms of SNR divided by the acquisition time, SNR<sub>t</sub>. The SNR and CNR (between muscle and fat) is optimal at a flip angle of 90deg for b-FFE sequence. Partial Fourier acquisition in the read-out direction is possible without significant reduction in image quality. This enabled the scan time to be reduced by  $\sim 30\%$ . This study is detailed in [PIKV04] and [GPMTV04].

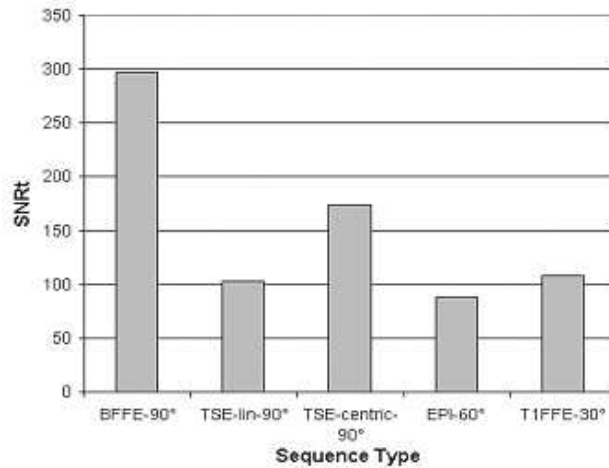


Figure 5.5: Plot showing the relative performance of different ultra-fast sequences in terms of SNR divided by frame acquisition time (SNR<sub>t</sub>) for the oil tube phantom. These were balanced fast field echo (BFFE), turbo spin echo (TSE) with centric and linear k-space trajectories, Echo planar imaging (EPI) and T1 weighted Fast Field Echo (T1FFE)

#### In-vivo study

The trade off in image quality with FOV and matrix was investigated qualitatively on six healthy volunteers in order to achieve the optimum resolution, contrast and frame acquisition time. As scan duration is proportional to the phase encode matrix, the phase encode matrix is maintained  $< 100$  at the shortest repetition time possible (TR 3.5ms). It was found that reducing the FOV and hence the phase encode matrix, maintaining an in-plane resolution of  $2mm$ , is not an effective way to reduce frame acquisition time,



due to the need to use fold-over suppression to avoid aliasing in the phase encode direction. A parallel imaging technique, SENSE (Philips Medical Systems, Best NL), was found to reduce the scan time by a factor of 2 without significant reduction in image quality. A reference scan is acquired prior to the SENSE MR sequence to measure the sensitivity profile of the phased-array coil. The same reference scan is used for all the images of the dynamic series. Figure 5.6 shows images from two dynamic protocols at a certain time. The first one was performed on three slices without the SENSE technique ( $1.75frames/s$ ), and the second on four slices with the SENSE technique ( $6.67frames/s$ ) which produces more artefacts. In Section 5.8.4, we will present how the planes are defined with reference to our bone tracking technique.

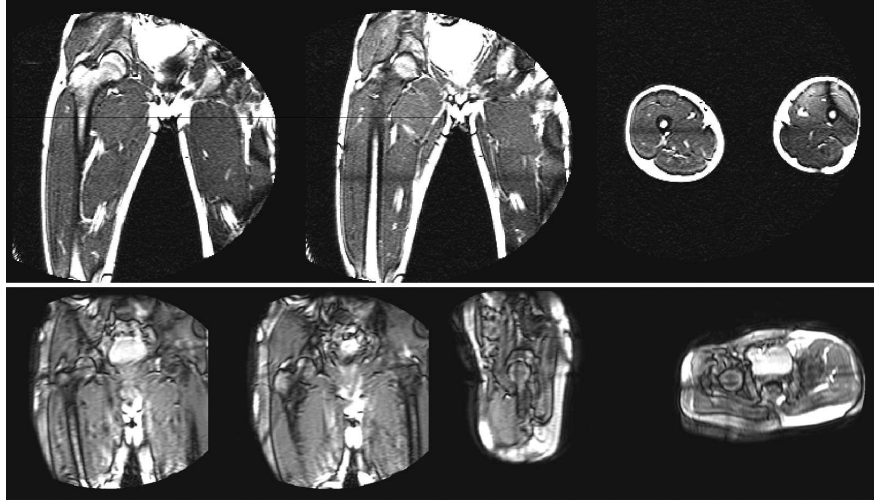


Figure 5.6: Dynamic MRI protocols for the hip

- First row: bFFE, TR =  $3.5ms$ , TE =  $1.1ms$ , FOV =  $450mm$ , Matrix =  $256 \times 256$ , thickness =  $10mm$ , FA =  $80deg$ , resolution =  $1.8mm \times 1.8mm$ .  $57sec$ , Reduced Fourier scan: 153 Phase Encoding steps.
- Second row: bFFE, TR =  $2.9ms$ , TE =  $1.1ms$ , FOV =  $450 \times 500mm$ , Matrix =  $96 \times 192$ , thickness =  $10mm$ , FA =  $30deg$ , resolution =  $4.7mm \times 2.6mm$ .  $15sec$ , Reduced Fourier scan: 81 Phase Encoding steps, SENSE factor 2 encoding

### 5.3 Pre-processing

In MRI, data is acquired in a complex form, known to be corrupted with white Gaussian noise. Commonly, only the magnitude (intensity) image is kept, due to its physical meaning. The phase information is discarded, but recent studies [BFS<sup>+</sup>06] have shown that it could be valuable for cartilage segmentation. The computation of the magnitude change the data distribution to a Rician distribution [SdDA<sup>+</sup>98]. The second moment of the intensity is given by:  $E[I^2] = \langle I_i^2 \rangle_i = 2\sigma^2 + \mu^2$  where  $I$  is the intensity variable,  $\sigma$  is the standard deviation of the Gaussian noise and  $\mu = \sqrt{\mu_R^2 + \mu_I^2}$  is the mean signal value (as opposed to the Gaussian distribution where  $E[I^2] = \sigma^2 + \mu^2$ ). A common measure of the noise level in images is the signal-to-noise ratio that is given by the ratio of the signal standard deviation and noise standard deviation:  $SNR = \sigma_s / \sigma_n$ . To compute the noise variance, we take background voxels where the mean signal is known to be null [KKCO89]. An unbiased estimation of variance is thus obtained by averaging squared intensity values of background points:  $\sigma_n^2 = \langle I_i^2 \rangle_{i \in B} / 2$ . Background points can be obtained by a simple threshold on the intensities. An estimate of the signal standard deviation is obtained from all voxels  $\sigma_s^2 = \langle I_i^2 \rangle_i - \langle I_i \rangle_i^2$ . The  $SNR$  is finally given by:

$$SNR^2 = 2 \frac{\langle I_i^2 \rangle_i - \langle I_i \rangle_i^2}{\langle I_i^2 \rangle_{i \in B}}$$

An alternate way is to subtract (resp. correlate) two images acquired successively (in this case, only the noise (resp. signal) remains) [SdDA<sup>+</sup>98]. In order to remove noise in MRI images, we are comparing two standard methods: the Gaussian isotropic filtering, and the anisotropic filtering [PM90]. Anisotropic filtering is known to better preserve edges which is important in segmentation (see Section 4.7). Here, we apply the VTK implementations of the isotropic filter with  $\sigma = 1mm$  and of the anisotropic filter with diffusion factor=  $0.5mm$ , number of iterations= 4 and threshold= 500. We are also testing the non-local means (NL-means) algorithm by Buades et al. [BCM05] that has proven to be very efficient for MRI images [CYB06]. In the NL-means algorithm, the filtered value  $I'_j$  of a voxel  $j$  is a weighted average of *all* voxel intensities, where weights are defined by a similarity function:

$$I'_j = \frac{\sum_i w_{ij} I_i}{\sum_i w_{ij}} \quad \text{where} \quad w_{ij} = e^{-\frac{\|I_{v(i)} - I_{v(j)}\|^2}{(h\sigma_s)^2}}$$

Here,  $h\sigma_s$  is a smoothing factor, proportional to the noise level (we set  $h = 3$ ). The similarity measure that we use is the weighted Euclidean distance between the intensities in the neighbourhood of  $i$  and  $j$ :  $\|I_{v(i)} - I_{v(j)}\|^2 = \sum_{k \in v} (I_{i+k} - I_{j+k})^2 / V$  where  $v$  is the neighbourhood domain (typically a cubic domain of size  $V = N^3 = 3^3$  around voxels). Because the raw computation time of this algorithm is very large (up to several hours). We discard voxels pairs that are likely to be dissimilar: first we restrict the search of pairs in a squared domain of size  $M^3 = 21^3$  in which we take  $n = 500$  random samples, and second we discard voxels pairs that have dissimilar local means and standard deviation. Local means and standard deviation are computed in the neighbourhood  $v$ :  $\mu_i = \langle I_i \rangle_{i \in v(i)}$  and  $\sigma_i^2 = \langle I_i^2 \rangle_{i \in v(i)} - \langle I_i \rangle_{i \in v(i)}^2$ . The test is performed through the thresholds:  $0.8 < \mu_i / \mu_j < 1.2$  and  $0.25 < \sigma_i^2 / \sigma_j^2 < 2.25$ . This approach is almost similar to the one of Coupé et al. [CYB06].

Our test volume is presented in Figure 5.7, where noise is highlighted through gradient images. We see in the filtered images (Figure 5.8) that, as expected, the NL-means (third column) outperforms the two other methods. Removed noise (RMS difference between original and restored volumes) is more widely spread over the image (no visible geometric structure), meaning that the initial details of the images are better conserved. Looking at the different gradient images, showing the homogeneity of the regions, it is visible that the NL-means method better reduces the noise level. High-frequency components of the noise are removed while high-frequency component of the image are conserved. On the contrary, the isotropic smoothing (first column) removes all high-frequencies, losing image details (larger contours) and suppressing the possibility of delineating some thin structures. The anisotropic filter (second column) does not remove noise in the regions of high gradient (black regions in the noise images). Thus, it homogenises already homogeneous regions (low gradients) but does not improve contour information. Also the threshold level has to be parameterised ad-hoc by first analysing the gradient level of the details to be conserved. In contrast, parameters of the NL-means are generic. Its main drawback is, in fact, the huge computational time required (about 1 hour for a standard MRI volume), despite the improvements presented above. This could be prohibitive in the clinical environment if an immediate processing is required. However, as this task does not need any user interaction, we believe this could be applied automatically between the acquisition and processing steps.

After noise filtering, MRI volumes for the same acquisition (same pose) are merged into a unique volume: they are first re-sampled according to the highest resolution, and then rigidly registered according to volume origin/ dimensions contained in the DICOM files. During merging, intensities are normalised linearly: for all volumes, intensities are rescaled according to the max/ min intensities of the entire stack.

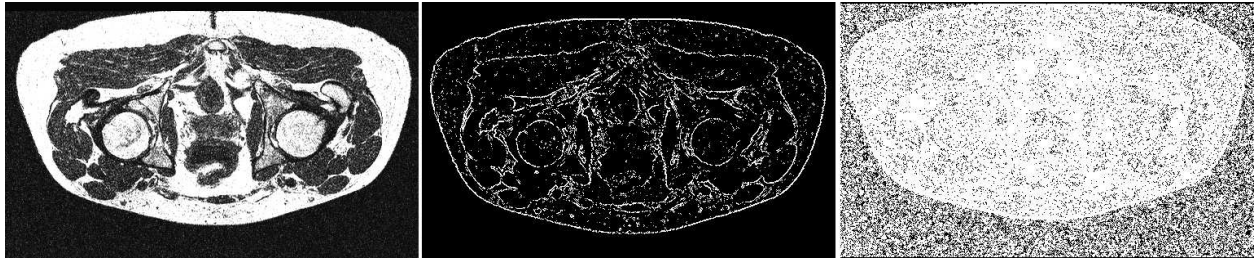


Figure 5.7: *Left*: Sample image from the original volume ( $SNR = 3.1$ ). *Middle*: gradient image, thresholded at 500. *Right*: gradient image, thresholded at 100

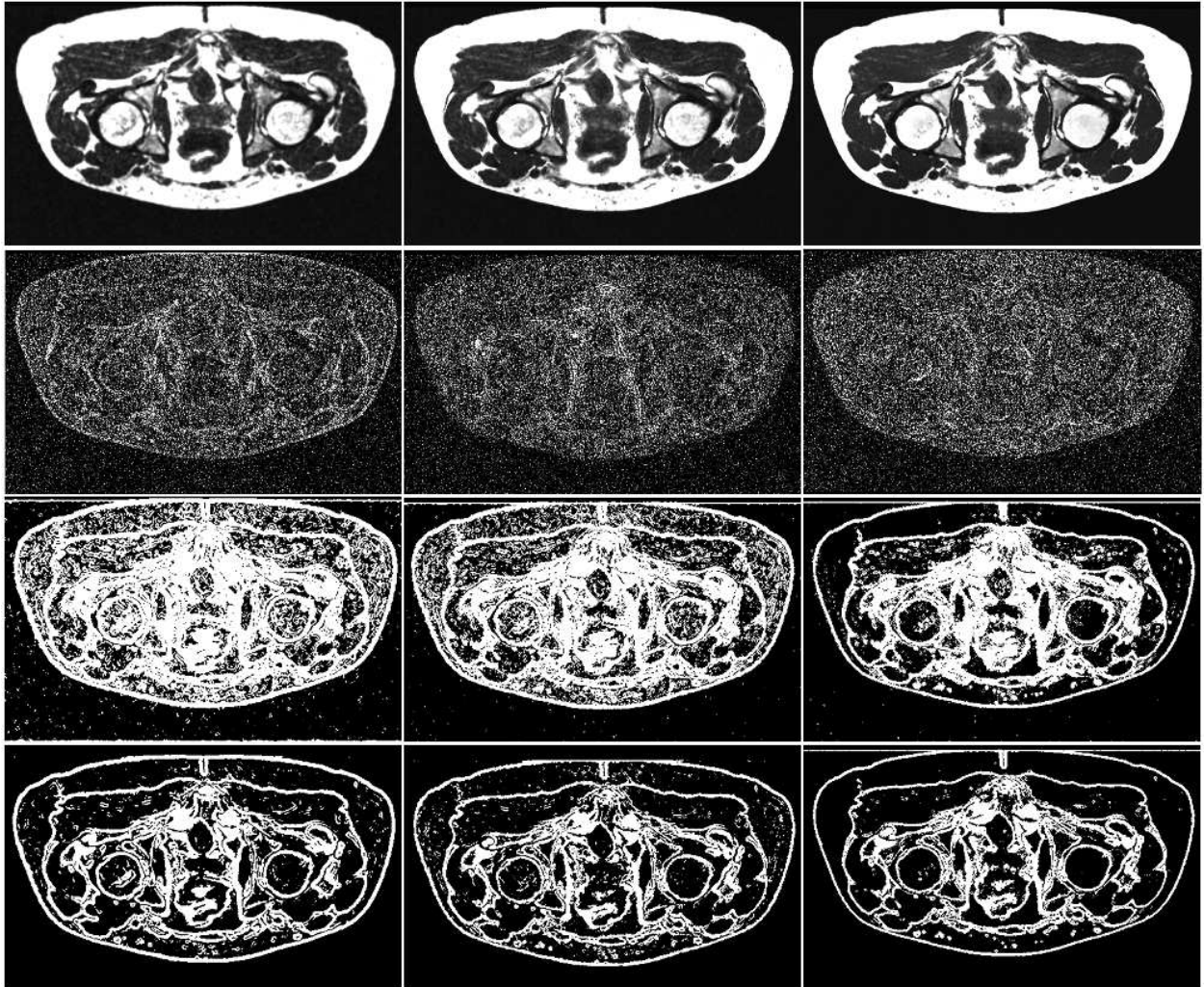


Figure 5.8: *First row*: Sample images from the filtered volume. *Second row*: noise images. *Third row*: gradient image thresholded at 100. *Fourth row*: gradient image thresholded at 50. *First Column*: Isotropic smoothing ( $SNR = 3.4$ ). *Second column*: Anisotropic smoothing ( $SNR = 3.4$ ). *Third column*: NL-means method ( $SNR = 3.4$ )

## 5.4 Construction of the generic model

### 5.4.1 Interactive segmentation

Our fast collision handling method allows to constraint models according to user-defined positions in space. By manually putting internal/ external/ boundary constraint points, we have reconstructed generic models to be used as reference models for automatic segmentation. For all organs, a simplex sphere or a cylinder is used as a starting point (Figure 4.1). The mesh is progressively optimised (Section 4.4) to improve its geometric and topological quality with regards to the shape we want to approximate. Alternative techniques can be used: binary volumes (i.e. segmented data) can be drawn with standard segmentation software. After their conversion to triangle meshes (using the marching cubes algorithm), we can fit simplex models using our deformable ICP algorithm (see example in Section 4.8.3). We can also use image forces from binary volumes using gradient information. All these methods lead to similar results, but, to avoid the time consuming step of binarising the data, we prefer to place constraints points during mesh adaptation which is faster in practice. If needed, we can convert our simplex meshes back to segmented volumes by using the voxelisation technique presented in Section 4.7.2. As presented below, we can adapt the generic models interactively in order to generate reference models of any individual in any posture. In the following sections, interactively reconstructed simplex surfaces are used as a gold standard for the validation of our automatic techniques. The power of constraining shapes during adaptation is that exact geometric correspondences are obtained (i.e. morphological features have the same vertex indexes across individual models).

### 5.4.2 Definition of attachments

The human musculoskeletal anatomy exhibits various organs interrelationships: muscles are attached to bones, they can merge into common tendon units, fascia binds muscles and enforces frictionless contact between them, etc. This prior topological information can be taken into account to improve the segmentation. Soft attachments (fascia and fat) are simulated using collision handling based on the reference relative position between organs (Section 4.7.2). For hard constraints (bone/ muscle, cartilage/ bone and ligament/ bone attachments) we have developed a method based on cardinal splines to generate attachment areas, so that their placement and adjustment have a reduced number of degrees of freedom. Two splines, corresponding to the two models to be attached, are used to establish spatial correspondences through their curvilinear coordinates. We project splines on model surfaces and store the barycentric coordinates of the spline control points. Thus, attachment areas can be easily warped from one individual to another (assuming there are small inter-subject variations in organ attachment areas). The process to select vertices to be constrained is done via these splines:

1. First, faces are selected by projecting spline points (obtained by cardinal spline interpolation between control points).
2. Selected faces are filled in to produce the attachment area.
3. Border vertices are selected by tracking the attachment area contour (simplex surface orientation 4.2 is helpful to guarantee a correct vertex ordering wrt. spline coordinates)
4. Internal vertices are selected from attachment faces and border vertices.

This process has proven to be robust and consistent (see Figure 5.9), since it guarantees that no vertex is missed (this would happen if we choose to select vertices directly from the splines). We handle differently border and internal attachment vertices when we do the binding: border vertices positions are altered (by matching points of the two splines) and constrained by mass modification [BW98]:  $\mathbf{M}^{-1} = \mathbf{0}$ . Internal vertices are kept free for a moment (surface relaxation), and then projected to the attachment surface (they are also constrained by mass alteration). To allow further attachment modification, the two splines are

merged in one and the selection process is updated on one model only (see Video 5.1 ). This process is fine as soon as we consider soft-tissue/bone attachments because bone models are rigid (the attachment process is performed after bone segmentation). For merging two deformable models (e.g. common tendons), the process is slightly different because attachments are mobile: first, we pair attachment vertices from the two models. Then, during simulation, we consider vertex pairs as one vertex (masses and force contributions from the two models are summed). In practice, this is also used for modelling specific parts with high curvature, where smoothing forces are not appropriate (e.g. attachment between the adductor magnus and its inferior tendon, see Figure 5.8).

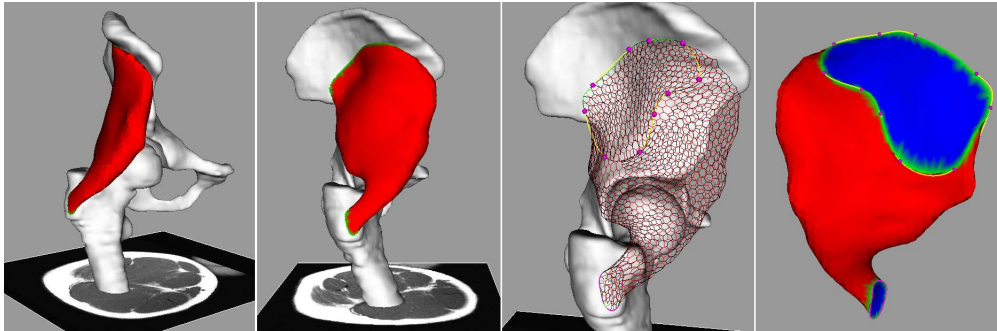


Figure 5.9: An example of spline-based muscle/ bone attachment (gluteus minimus)

We have defined about 50 generic attachment splines for the hip and the thigh (see Figure 5.10) mainly from the literature on anatomy. Using MRI, it is difficult to precisely validate attachment areas because there are almost not visible. However, by checking the continuity of organ shapes near interfaces, we have corrected the attachments (minimisation of the distortions). It is also important to check that no attachment intersect (to prevent from permanent collisions), which is possible with our selection scheme (see above).

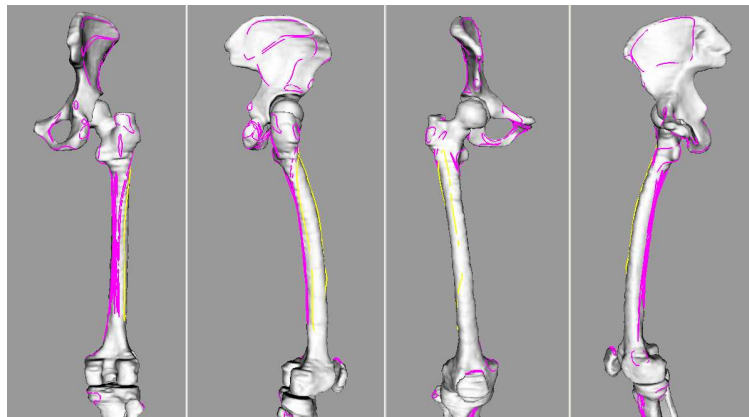


Figure 5.10: Attachment splines for the hip and the thigh

### 5.4.3 Medial axis generation

In Section 4.6.4, we have described how to compute constraints that drive a medial axis to the center of closed surface. Here we present how we use this technique in the context of the musculoskeletal system. Because we want simple medial axis (no spike) that have a reduced number of DOFs, we initially build

simplex planes. We have tested several connectivity schemes to generate simplex planes (see Figure 5.11). We use the first method (on the left) because it creates less distortions on the surface.

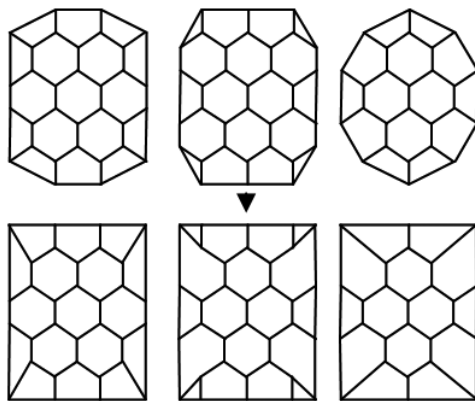


Figure 5.11: Several methods for generating a simplex surface with one hole.

Now, we explain how medial axis are fully automatically generated from a model that is attached onto bones (we only consider muscles, ligaments and cartilage models, due to the relevance of their medial representation, see Section 4.6.4). Our idea is to initialise simplex planes near models using attachments areas (see Section 5.4.2). Then, these simplex planes are cropped and iteratively adjusted to object centers. More precisely, the process involves the following steps:

1. We generate attachment centers with an approach based on the distance map: all attachment faces (blue and green faces in Figure 5.12.A) are initialised with a distance equals to 1. Distances are iteratively modified until convergence: all faces that have a distance  $d$ , which neighbouring faces have also a distance  $d$ , are incremented to  $d + 1$ . The center faces are obtained by taking the local maxima of this distance map (green faces in Figure 5.12.A). This approach is similar to most approaches for medial axis generation from voxel maps [Bor84]. It works fine, assuming that faces are homogeneous in terms of size which is the case.
2. We initialise a simplex plane by rigid transform using attachment centers and the model principal directions: two attachments (e.g. gluteus minimus) give one direction (by linking the central vertices of the center areas) and the second direction is obtained by taking the model principal direction orthogonal to the first direction. Some models have three attachments (e.g. biceps femoris) which give directly the two plane directions.
3. Medial attachment points are selected by taking the closest border points to the center extrema.
4. The plane is transformed by thin plate splines interpolation according to the original and target attachment positions (see Figure 5.12.C).
5. The plane is cropped by projecting the model (see Figure 5.12.D). First, two faces groups, without holes, are selected from the plane by projecting all model vertices. Then, all external faces are removed. Finally, degenerated vertices (with two neighbours) and faces (with three vertices) are removed. This guarantees a correct connectivity (three neighbours per vertex) and topology ( $g = 0$  and  $H = 1$ ).
6. The medial axis is finely matched (see Figure 5.12.E, Figure 5.14 and Video 5.2 ) by simulation using medial constraints (see Section 4.6.4) and smoothing constraints (see Section 4.6.1). Attachment vertices are projected at each iteration to allow sliding on model attachment areas. During the simulation, collision handling is performed to restrict the medial axis to the inside of the model.

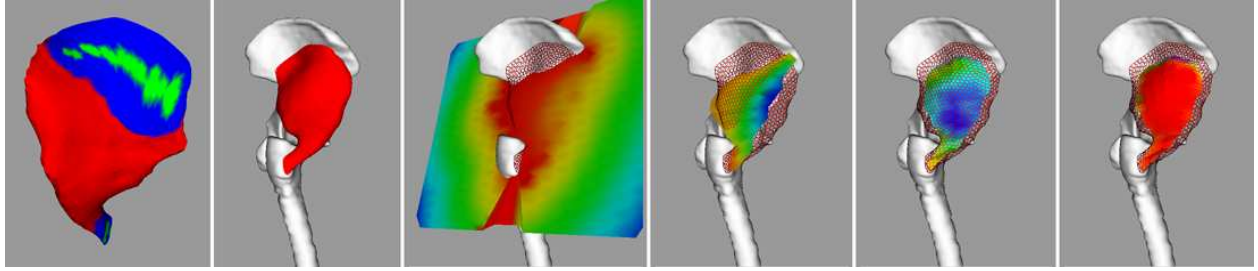


Figure 5.12: Medial axis generation steps. From left to right: A. Attachment center computation; B. initial attached model; C. Simplex plane initialisation.; D. Plane after cropping; E. Medial axis after fitting; F. Reconstruction error (red: min, blue: max). In C, D and E colours show the radius (red: min, blue: max)

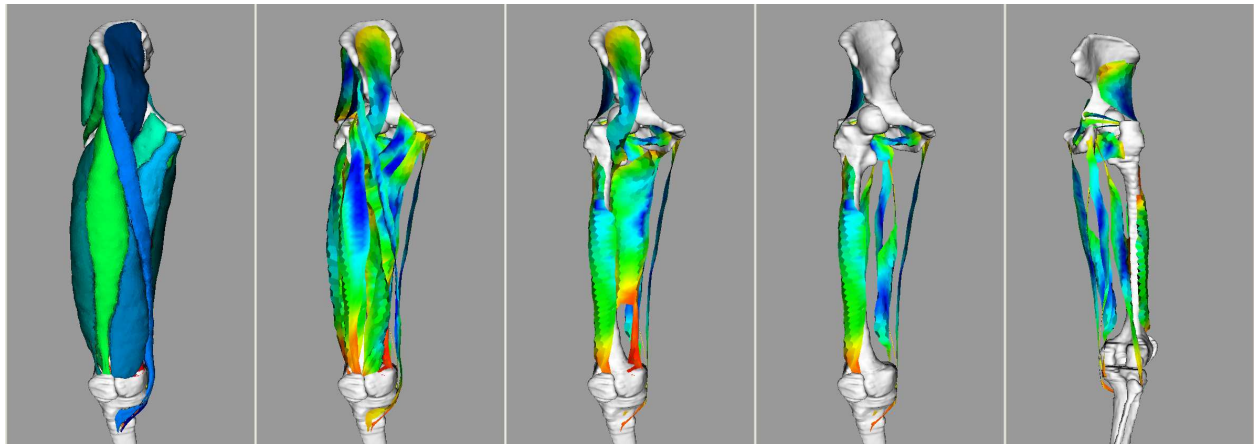


Figure 5.13: Models (left) and Medial axis (right) of the thigh. Different layers and views are shown

This fully automatic process has proven to be robust for all the models we have tested (21 muscles, 3 cartilages, 3 ligaments). Figure 5.13 illustrates our results for the hip and the thigh. For the cartilages and the labrum, a simplex sphere has been used instead of a plane to better fit the shape of the joint. We have noticed that, with our method, simplex faces are almost not distorted, meaning that a step to improve their topological and geometrical quality (Section 4.4.5) is not needed here (we have verified that the quality would not be significantly improved). As explained in Section 4.6.4, it is possible to compute the error associated to a medial axis by averaging the distance between the model and the reconstructed model. Figure 5.14 shows the typical evolution of this error during the matching process (it corresponds to the matching shown in Video 5.2 ). Obviously, we can reduce the final error by increasing the resolution of the medial axis (i.e. the number of vertices). However, the goal is to minimise the amount of information for representing a model. So, there is a trade-off between the compression factor and the error. The compression factor is the ratio between the number of model vertices times 3 (three spatial coordinates) and the number of medial vertices times 4 (three spatial coordinates plus the radius). Figure 5.15 illustrates the dependence between the compression factor and the error. In practice, we have decided to use an initial edge length of  $8mm$  for planes, leading to an average compression factor of 14 and an average error of  $0.61 \pm 0.65mm$ . Considering the type of shapes we have, we believe that this is a good trade-off. It allows, on one side, a precise shape representation (for shape analysis, radial regularisation and medial axis-based collision handling) and, on the other side, a significant reduction of the required information (about 14 fold reduction).

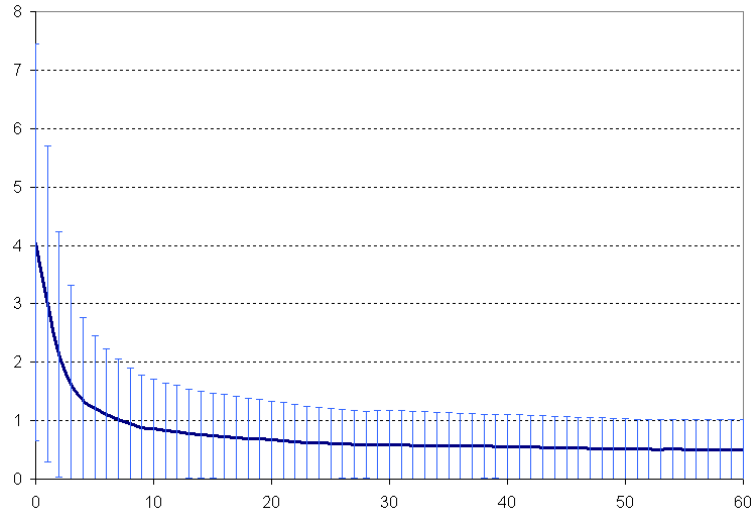


Figure 5.14: Evolution of the medial axis matching error (in  $mm$ ) in function of the iteration number for a representative object (biceps femoris)

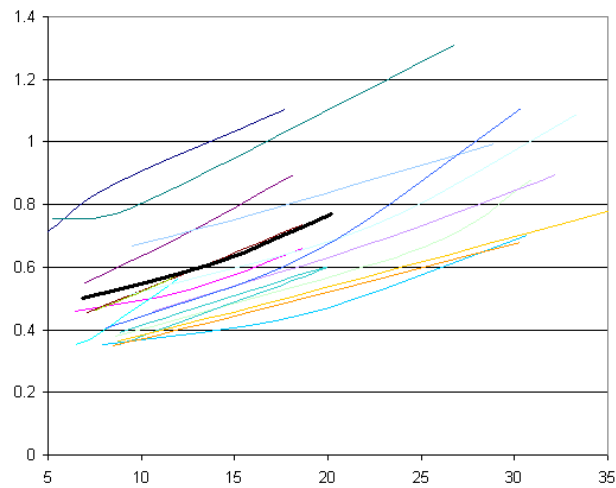


Figure 5.15: Final error in function of the compression factor for the tested models, and the average (in bold)

## 5.5 Musculoskeletal modelling from static MRI

In this section, we describe how we can perform an automatic registration of the generic musculoskeletal models onto individual static MRI data. The prior knowledge that is used is composed of shape information (local and global geometry of the multi-resolution generic model) and image information (intensity profiles of the generic model in the generic MRI dataset).

### 5.5.1 Processing scheme

Our individualisation scheme is made of the following steps:

1. Pre-processing of individual MRI data (volume extraction from DICOM images, noise removal, rigid



registration and re-sampling of MRI series) as described in Section 5.3.

2. Initialisation of the generic bones near individual bones. This can be done either by thin-plate-spline interpolation (with few manual placed landmarks), or by rigid transform using MRI volumes coordinates.
3. Bone models fitting based on image, shape and inter-model constraints.
4. Extraction of bone standard coordinates from anatomical landmarks as described in Section 5.6.
5. Initialisation of soft-tissue attachment splines based on the local barycentric coordinates of generic spline control points.
6. Initialisation of generic soft-tissue from individual joint angles. Our generalised skinning algorithm will be presented in Section 5.9.
7. Soft-tissue models fitting based on image, shape and inter-model constraints.

The two key steps are the bone and soft-tissue fitting phases (3. and 7.). For them, different parameters need to be established, such as intensity profile lengths and force contributions.

## 5.5.2 Optimising parameters

### Intensity profiles

It is important to optimise image forces to better discriminate organ contours in images. Indeed, no similarity measure has proven to perform the best in the general case. Also, the size of the neighbourhood and the masks to process them has to be established in our context. In [SLP06], Skerl et al. propose five evaluation criteria to measure the quality of a similarity measure. It is based on a gold standard registration, around which the variation of the similarity measure is examined. In our case we have manually segmented models from different subjects (Section 5.4.1). The accuracy of these gold standards is millimetre-based (voxel size). We want to compare the intensity profiles of two generic models by checking the variation of the similarity. Contrary to [SLP06] (rigid registration), our parameter space consists of vertex 1D excursions  $j$  along normals (see Section 4.7.1). Based on it, we build similarity profiles for each vertex. The similarity is the opposite of the function  $\Delta$  presented before (that we want to minimise). Figure 5.16 shows the similarity profile for a sample point  $i$  which is globally maximal at  $\tilde{j}_i$ . Positive gradients are equal to the similarity gradient if  $\Delta(j-1) < \Delta(j)$  ( $j < \tilde{j}$ ) or  $\Delta(j+1) > \Delta(j)$  ( $j > \tilde{j}$ ), and to 0 otherwise. Figure 5.17 shows typical intensity profiles and similarity profiles. The similarity measure evaluation criteria are:

- **Accuracy** ( $ACC = 1/N \sum_i |s.j_i^* - s.\tilde{j}_i|$ ): the mean distance between the gold standard excursions and the ones of maximal similarity (for a better understanding of  $ACC$ , we change the original definition from [SLP06] that uses the mean *square* distance).
- **Distinctiveness** ( $DO = 1/(2.k.s.N) \sum_i [\Delta(\tilde{j}_i + k) + \Delta(\tilde{j}_i - k) - 2\Delta(\tilde{j}_i)]$ ): the average change of the similarity around the global maximum.
- **Capture range** ( $CR = s/N \cdot \sum_i |\tilde{j}_i - \tilde{j}'_i|$ ): the smallest distance between the global maximum and a local maximum.
- **Number of minima** ( $NOM$ ): The cumulative number of similarity minima (i.e. number of maxima of  $\Delta$ ).
- **Risk of nonconvergence** ( $RON = 1/(2.k.s.N) \sum_i \sum_{j=\tilde{j}_i-k}^{\tilde{j}_i+k} dj_i$ ): the average of positive gradients within distance  $k.s$  from global maxima.

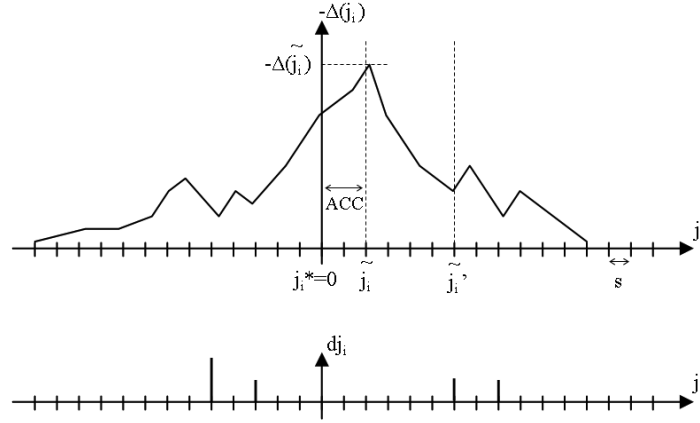


Figure 5.16: Similarity profile at point  $i$  and positive gradients

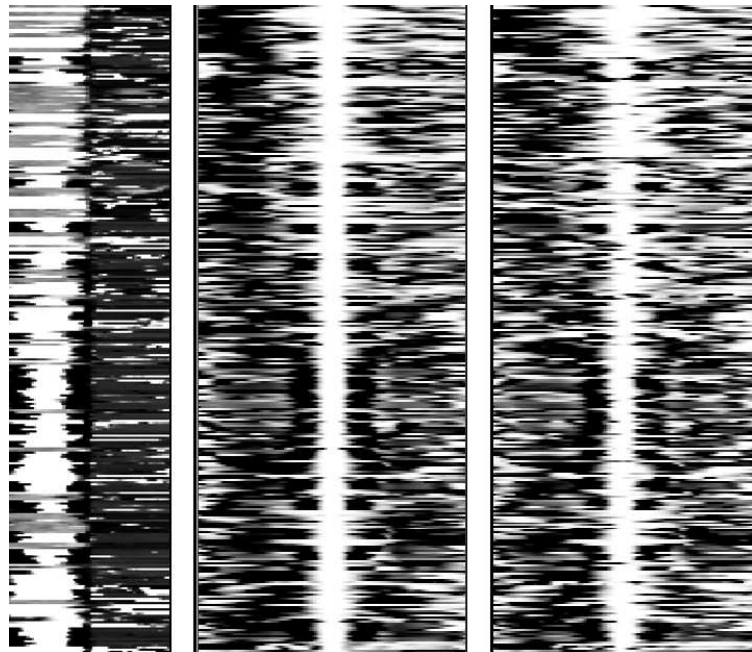


Figure 5.17: a. Sample intensity profiles (femur); b. the corresponding self-similarity profile (NCC measure); c. a inter-subject similarity profile (NCC measure)

The better a similarity measure, the smaller the values of ACC, NOM and RON, and the larger the values of CR and DO. In a first step, we optimise the size of intensity profiles by minimising the value of ACC which is the most important indicator. Figure 5.18 is a typical example of the accuracy variation with respect to profile lengths. We see that, after a certain point, increasing the length does not improve the accuracy. Also, being more variable, the external part of intensity profiles is less relevant (we believe that the best profile size is intrinsic to the tissue we want to segment, and not to the MRI sequence). Table 5.1 summarises the values of internal length  $s.Ni$  and external length  $s.No$  obtained for different models (femur, pelvis and all muscles) from two generic individuals and different similarity measures. In practice, we decide to take, as a general case,  $s = 0.5mm$ ,  $Ni = 25$  and  $No = 5$  for bones and  $Ni = 20$  and  $No = 10$  for soft tissues.

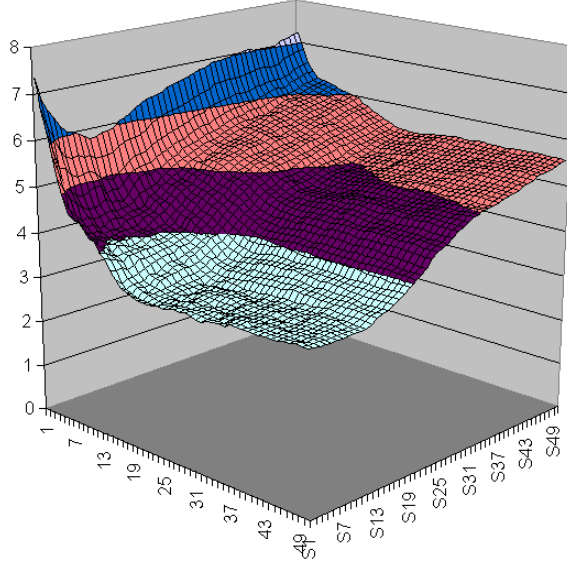


Figure 5.18: Similarity measure accuracy in function of intensity profile internal and external lengths (in half millimeters)

Similarity measures	Femur			Pelvis			Muscles		
	$s.Ni$ (mm)	$s.No$ (mm)	$ACC$ (mm) $\pm$ std dev	$s.Ni$ (mm)	$s.No$ (mm)	$ACC$ (mm) $\pm$ std dev	$s.Ni$ (mm)	$s.No$ (mm)	$ACC$ (mm) $\pm$ std dev
AD	11	3.5	$1.93 \pm 2.03$	1	0.5	$2.75 \pm 3.07$	0.5	0.5	$4.25 \pm 3.26$
NCC	10	1	$1.53 \pm 1.69$	8	0.5	$2.44 \pm 2.75$	7	3	$3.50 \pm 3.01$
AD (gradient magnitude)	20.5	0.5	$2.36 \pm 2.51$	12	2	$3.44 \pm 3.24$	7	1	$3.82 \pm 3.20$
NCC (gradient magnitude)	22	1.5	$2.05 \pm 2.20$	9	3	$3.08 \pm 2.98$	9.5	3	$3.67 \pm 3.09$
GAD	25	0.5	$1.99 \pm 2.13$	12.5	0.5	$3.27 \pm 3.17$	0.5	0.5	$4.09 \pm 3.10$
GNCC	14	1	$1.53 \pm 1.53$	8.5	1	$2.71 \pm 2.88$	10	5	$3.44 \pm 2.94$
AD $[-1, 0, 1]$	25	0.5	$1.90 \pm 2.07$	12	0.5	$3.28 \pm 3.18$	0.5	1	$3.92 \pm 3.04$
NCC $[-1, 0, 1]$	14	2.5	$1.51 \pm 1.59$	8.5	5.5	$2.52 \pm 2.64$	9.5	5	$3.33 \pm 2.85$
AD $[-1, 2, 1]$	12	3.5	$1.89 \pm 1.99$	1	0.5	$2.77 \pm 3.05$	1	0.5	$4.26 \pm 3.26$
NCC $[-1, 2, 1]$	11	1	$1.51 \pm 1.67$	8	0.5	$2.44 \pm 2.73$	7	2.5	$3.49 \pm 2.99$

Table 5.1: Optimised intensity profile length

Now, we want to compare the different similarity measures through the criteria proposed in [SLP06], in order to select the best one. Our similarity profile length (search space) is fixed to  $2cm$  ( $1cm$  backwards and  $1cm$  inwards) with a step size of  $0.5mm$ .  $DO$  is computed with  $k = 2$  and  $RON$  over the full similarity profile. Tables 5.2, 5.3 and 5.4 summarise the results for the femur, the pelvis and muscles. They show that the best similarity measure is the normalised cross-correlation ( $NCC$ ). The use of gradient images is not worth for this type of mono-modal registration. However the discrete mask  $[-1, 2, 1]$  (gradient enhancement) slightly improves the results without significant extra computational cost. We prefer it from the mask

$[-1, 0, 1]$  (gradient magnitude in normal direction) because it is more accurate far from the solution due to its continuous term. Here, intensity profiles are computed in images from the same protocol (the static protocol of Section 5.2.1). We have tested the use of different protocols (turbo spin echo T1 vs. truefisp T2\*) which also indicates that the best similarity measure is the normalised cross-correlation (*NCC*). The use of the gradients directly (*G* and *NG* measures) leads to good results in the close neighbourhood of the solution (large number of extrema) and is not sensitive to errors in the generic models. Therefore, we usually combine a measure based on intensity profile (*NCC*) with a large search space, and a measure based on the gradient with a reduced search space (see Section 5.5.3).

Similarity Measures	ACC	DO	NOM	RON	CR
AD	2.06	50	1.47	1.4	0.71
NCC	1.61	0.07	1.59	0.01	0.64
AD (gradient magnitude)	2.52	26.35	3.41	1.99	0.79
NCC (gradient magnitude)	2.2	0.22	3.19	0.02	0.63
GAD	2.41	16391.44	2.6	1985.47	0.81
GNCC	1.49	0.21	2.64	0.02	0.55
AD $[-1, 0, 1]$	2.23	35.7	3.23	2.41	0.71
NCC $[-1, 0, 1]$	1.52	0.26	3.06	0.03	0.55
AD $[-1, 2, 1]$	1.98	104.59	1.56	3.38	0.65
NCC $[-1, 2, 1]$	1.6	0.08	1.65	0.01	0.63
G	4.11	252.63	5.25	10.32	0.68
NG	3.01	230.09	4.56	18.73	0.75

Table 5.2: Similarity measure evaluation (femur)

Similarity Measures	ACC	DO	NOM	RON	CR
AD	3.58	21.6	2.07	1.74	1.48
NCC	2.4	0.15	2.08	0.02	0.82
AD (gradient magnitude)	3.25	12.45	3.25	0.97	0.9
NCC (gradient magnitude)	2.89	0.2	3.13	0.02	0.82
GAD	3.18	9094.44	3.49	1215.37	0.86
GNCC	2.49	0.26	3.47	0.03	0.7
AD $[-1, 0, 1]$	3.16	24.59	4.12	2.27	0.81
NCC $[-1, 0, 1]$	2.44	0.34	3.91	0.04	0.63
AD $[-1, 2, 1]$	3.49	48.91	2.3	3.84	1.39
NCC $[-1, 2, 1]$	2.36	0.16	2.28	0.02	0.79
G	4.53	189.36	5.69	10.27	0.76
NG	4.34	235.92	4.92	18.5	0.74

Table 5.3: Similarity measure evaluation (pelvis)

The last test deals with the convergence of forces based on similarity measures (see Table 5.5). We simulate a generic model with image-based forces from another generic model and check vertex displacements after 20 iterations (time-step:  $dt = 1$  and damping:  $\gamma = 0.5$ ). In the ideal case, the model should remain still. The depth of the search space is set to  $1cm$  ( $0.5cm$  outside and  $0.5cm$  inside) and the step size to  $0.5mm$ .

Similarity Measure	ACC	DO	NOM	RON	CR
AD	4.52	20.74	1.89	0.94	1.54
NCC	3.54	0.12	2.37	0.02	0.98
AD (gradient magnitude)	4.04	14.39	2.79	1.11	1.25
NCC (gradient magnitude)	3.69	0.18	3.03	0.02	0.92
GAD	4.5	7679.18	3.17	1023.14	1.22
GNCC	3.42	0.23	3.55	0.03	0.77
AD [-1, 0, 1]	4.25	19.18	3.95	1.53	1.07
NCC [-1, 0, 1]	3.32	0.35	4.16	0.04	0.68
AD [-1, 2, 1]	4.55	44.05	2.15	2.19	1.5
NCC [-1, 2, 1]	3.53	0.14	2.61	0.02	0.93
G	4.68	156.49	5.52	54716.88	0.95
NG	3.76	166.58	4.78	8.57	0.8

Table 5.4: Similarity measure evaluation (muscles)

Similarity Measures	Force Smoothing	Energy Smoothing	Error (in <i>mm</i> ) $\pm$ std dev		
			Femur	Pelvis	Muscles
G			2.91 $\pm$ 3.44	3.02 $\pm$ 3.89	2.59 $\pm$ 2.94
G	x		2.78 $\pm$ 2.42	2.28 $\pm$ 2.24	2.41 $\pm$ 2.24
NG			2.55 $\pm$ 4.18	3.02 $\pm$ 3.86	2.35 $\pm$ 2.74
NG	x		2.14 $\pm$ 2.72	2.29 $\pm$ 2.21	2.10 $\pm$ 2.03
DEMON			1.37 $\pm$ 1.40	1.44 $\pm$ 1.53	2.18 $\pm$ 2.15
DEMON	x		1.40 $\pm$ 1.17	1.41 $\pm$ 1.25	2.20 $\pm$ 1.97
AD			1.82 $\pm$ 3.11	2.64 $\pm$ 4.15	3.23 $\pm$ 3.72
NCC			1.48 $\pm$ 2.12	1.94 $\pm$ 3.00	2.65 $\pm$ 3.82
NCC		x	2.74 $\pm$ 3.23	2.18 $\pm$ 3.11	3.48 $\pm$ 4.27
NCC	x		1.24 $\pm$ 1.24	1.40 $\pm$ 1.63	2.23 $\pm$ 2.61
NCC [-1, 2, 1]	x		1.22 $\pm$ 1.20	1.38 $\pm$ 1.62	2.24 $\pm$ 2.62
NCC [-1, 2, 1] AFFINE	x		0.90 $\pm$ 0.62	0.71 $\pm$ 0.51	0.54 $\pm$ 0.39

Table 5.5: Similarity measure convergence tests

Once again, the normalised cross-correlation seems the most adequate measure. Intensity difference-based measures (*AD* and *DEMON*) are not accurate with MRI because of the global intensity changes that occur across acquisitions (even if using the same sequences), although they seem adequate in the close neighbourhood of the solution (see last test). In practice, far from the solution, they do not perform well. The demon method is intrinsically regularised (smoothed) through displacement averaging along profiles. It ensures low amplitude forces at each iteration, but an increase of the number of time steps exhibits its divergence (also the use of force smoothing does not improve the results, meaning that the method is globally inaccurate in our context). The *energy smoothing* local regularisation method is not satisfactory, due to a uncorrelation of similarity profiles in the vertex neighbourhood (probably because of normal direction changes). On the contrary *force smoothing* is valuable as it filters outlier forces. We see that the affine regularisation improves the results a lot, meaning that enforcing shape regularity effectively reduces sensitivity to noise. We expect that internal forces will improve the accuracy of the matching in the same way.

## Force contributions

We have to set the contribution of the different forces at the different resolutions through the stiffness parameters  $\alpha' = \gamma\alpha$  (see Section 4.8). At the simulation level, we want that the sum of the force contributions equals to 1 (no overshoot nor underestimate of target positions). In practice, we have found that setting the total contribution of external forces to 30% of the total force contributions, lead to good results. Also, across the different resolutions, we set all shape memory forces with a equal contribution. To tune the global/ local effect of internal forces, we tried to use a rigidity parameter that increase/ decrease the contribution of low resolution forces with respect to high resolution forces. But we did not notice any significant improvement from using the same contributions for all resolutions. Let  $r_i$  be the current simulation resolution,  $r_0$  the finest resolution and  $r_{max}$  the coarsest resolution. We have:

- Shape memory forces: for each resolution  $r_j$  ( $r_i \leq r_j \leq r_{max}$ ),  $\gamma = 0.7/(r_{max} - r_i + 1)$
- External forces: for the resolution  $r_i$ ,  $\gamma = 0.3$ .

Additionally, volume preservation and smoothing forces may be added depending on the context (see next sections). When models own a medial representation, shape memory forces are applied to medial axis, and radial forces to both medial axis and models. Radial forces are not used in the multi-resolution fashion because they do not depend on the local mesh resolution. The distribution of radial contributions depends on what we want to do: either a shape memory-driven deformation (radial forces are higher for models) or a surface/ image driven deformation (radial forces are higher for medial axis).

### 5.5.3 Results and validation

#### Bone modelling

As described in the Section 5.2.1 and 5.2.2 on MRI acquisition, we are using two different protocols for bone model reconstruction. The first one is composed of one high resolution spin echo volume covering the whole femurs and pelvises. The second one is composed of five overlapping volumes: a low resolution volume (from knee to iliac crests), two small high resolution truefisp volumes centered on the hip joints and two radial volumes centered on the hip joints. Being more accurate, the first protocol has been used to reconstruct the generic models (reference for shape and intensity profiles during segmentation). We now present the matching of one femur generic model and one pelvis generic model to the 12 remaining MRI volumes. To validate the technique, a medical student (Laura Müller) has used the interactive technique (Section 5.4.1) to reconstruct accurately all the models (right side). The accuracy of those interactively reconstructed models is close to the image resolution ( $\simeq 1mm$ ) We compute the average distance and standard deviation to monitor the error of the automatic method (Table 5.6). During the process, the affine regularisation is applied to external forces with a decreasing contribution: the parameter  $\lambda$  that balances the contribution of local forces with respect to global forces varies from 0 to 0.7 (see Figure 5.19, red curve). Also the search distance for computing intensity profile-based forces is progressively reduced during the process: from  $1cm$  ( $0.5cm$  outside and  $0.5cm$  inside) to  $0cm$  (see Figure 5.19, green curve). Here we only use forces based on the NCC measure ( $[-1, 2, 1]$  mask) that we smooth in vertex neighbourhood. Collision handling is applied between the femur and the pelvis (sequential resolution) with a gap of  $4mm$ . Figure 5.19 (blue curve) reports the typical variation of the error during the process (that lasts about 1 minute, including both the femur and the pelvis) and Video 5.3 shows the matching process. Due to inaccuracy in the gold standard models, it is in fact not possible to assess errors below  $1mm$ . By manually comparing gold standard and automatically reconstructed models superimposed onto MRI (Figure 5.20), we do not observe any significant and systematic errors (the automatic models were all in the zone of ambiguity/ partial volume), indicating that our method works well for bones. There is an exception near iliac spines where large differences have been observed for some models (see Figure 5.20 middle). This is due to the morphological difference between men and women (the generic model is from a male) and to the big amount of noise in this region (close to the field of view

limit). Errors could be reduced by using generic models from the two genders. The method is not sensitive to small initial errors in the placement of the 15 landmarks, due to the global regularisation that is applied at the beginning of the process. The generic intensity profiles have been generated from a pre-processed volume (with the NL-means algorithm). For the volunteers, we did not notice significant differences in using denoised or initial volumes, showing that our algorithm is little noise-sensitive. However, it is important to use denoised reference intensity profiles to avoid systematic matching errors. We have checked that the use of multi-resolution forces is relevant as it prevents from large displacements when using fine resolutions (the systematic difference in the error is about  $0.5mm$ ). The difference is particularly obvious in Figure 5.21 where the curvature is visualised.

Volunteer ID	Mean error (mm)	Std. dev (mm)
1	1.41	1.21
2	2.04	1.93
3	1.48	1.19
4	1.25	1.06
5	2.13	2.12
6	1.49	1.53
7	2.03	1.66
8	1.39	1.31
9	1.53	1.35
10	1.98	1.78
11	2.04	1.89
12	1.70	1.68
Mean	1.71	1.62

Table 5.6: Distances between interactively and automatically reconstructed bone models

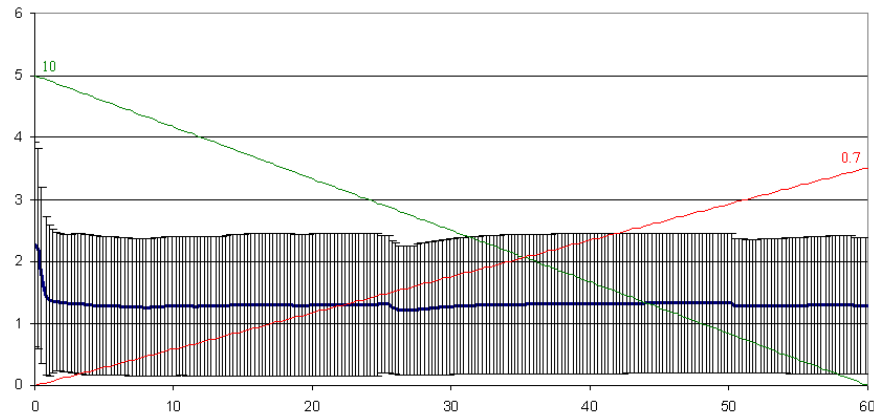


Figure 5.19: In blue: error variation during the bone segmentation process (std. dev.); in red: contribution of local external forces; in green: search distance for intensity profile-based constraints

Here, we present bone segmentation results when using the protocol for coxo-femoral conflict examination (Section 5.2.2). The main difference is that several overlapping MRI volumes are used to improve the quality of joint imaging. We propose to combine them to improve the segmentation. The first (low resolution) serie (Figure 5.22 left) is used for a coarse matching in the parts where no other serie is present (i.e. knee

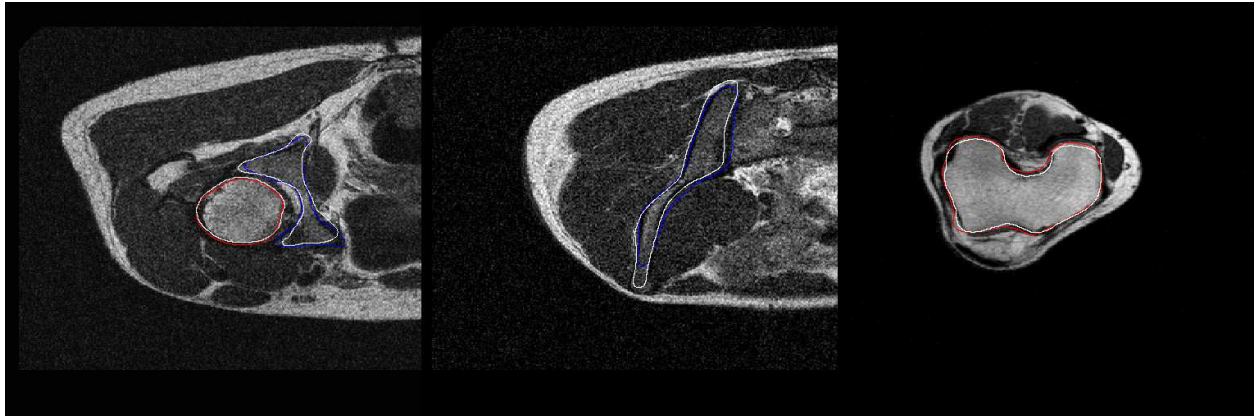


Figure 5.20: Difference between gold standard bone models (in white) and automatically reconstructed models (in red and blue)

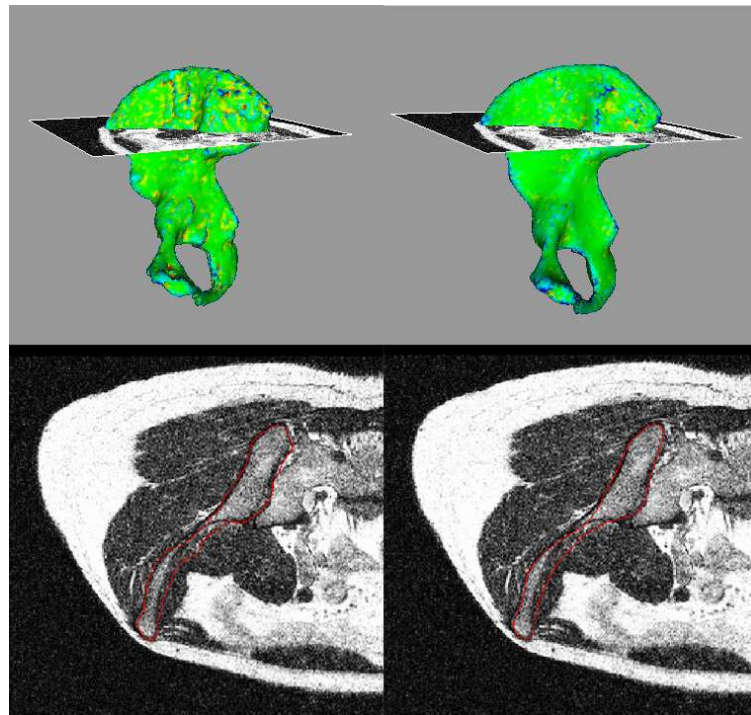


Figure 5.21: Improvement when using multi-resolution forces (right). Colours show model curvature

parts). The radial serie (Figure 5.22 right) is used for bone modelling in the articular regions. Those two series exhibit similar contrasts than the spin echo reference serie (where generic intensity profile have been generated), so that matching force based on intensity profile similarity are relevant. However, the last serie (truefisp, Figure 5.22 middle) shows very different contrasts. For it we use a force based on the gradient ( $NG$  measure). Hence, external forces contribution is split into intensity profile forces (series number 1 and 3) and gradient forces (serie 2), with an equal contribution and search distance. By using the same parameters than before, we can perform an accurate and automatic matching of the generic bones, as illustrated in Video 5.4 . The typical computational time is  $2min$  for the two hips.



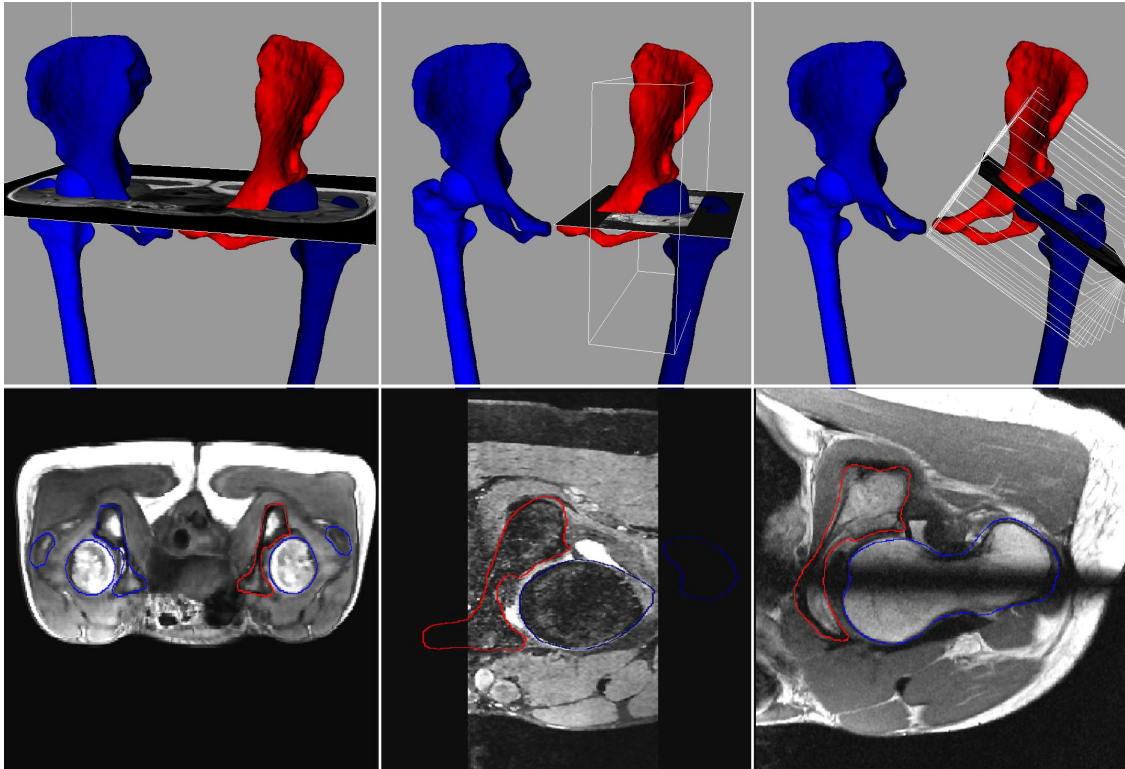


Figure 5.22: Result of bone matching using three overlapping datasets

### Skin modelling

Skin is probably the simplest tissue to segment because the background is clearly different from the rest of the image (low amplitude signal and noise). Therefore, a simple similarity measure based on the gradient with a large search space is sufficient for accurate skin reconstruction. Because fat (hypodermis tissue) appears in white in most MRI sequences (T1) due to its high water content, and because the epidermis thickness is negligible (see Section Section 2.9), skin delineation in MRI is even more simple. In fact, the surface of the different skin layers (epidermis, dermis, hypodermis) are not discernible with our resolution ranges; so, we abstract the skin in a single surface. The epidermis (resp. dermis) could be simulated using an average thickness of  $0.1mm$  (resp.  $2mm$ ).

Figure 5.23 and Video 5.5 demonstrate the application of our framework for automatic skin segmentation. From previous bone segmentation, the generic skin is automatically initialised based on our skinning algorithm (see Section 5.9). Because there is no much geometrical feature on the skin, model correspondences between individuals are not very accurate. In fact, they mainly rely on the skinning algorithm that uses joint transforms. We apply skin-bone collision handling due to some parts where bone are salient (knee, iliac crests). Also, because the body is cut through by MRI volume boundaries, we have to constrain our skin model (which is a closed surface) not to go inside. For boundary points (that are the same for any individual), instead of a penalty method (stiff and not exact), we use mass modification to constraint their displacement on MRI planes as introduced by Baraff et al. [BW98]:  $\mathbf{M}^{-1} = m^{-1}(\mathbf{I} - \mathbf{nn}^T)$  where  $\mathbf{n}$  is the plane normal and  $m$  the mass scalar value. The full skin segmentation process takes typically  $40s$  (90% of the time being spent on force computation). The skin is composed of 3744/1874 vertices/ faces (the part of the skin we model includes the pelvis and the femur). We did not validate our algorithm as we did not notice visually any significant and systematic error. Manual delineation is probably less precise (manual

segmentation vs. detection of high gradient contours).

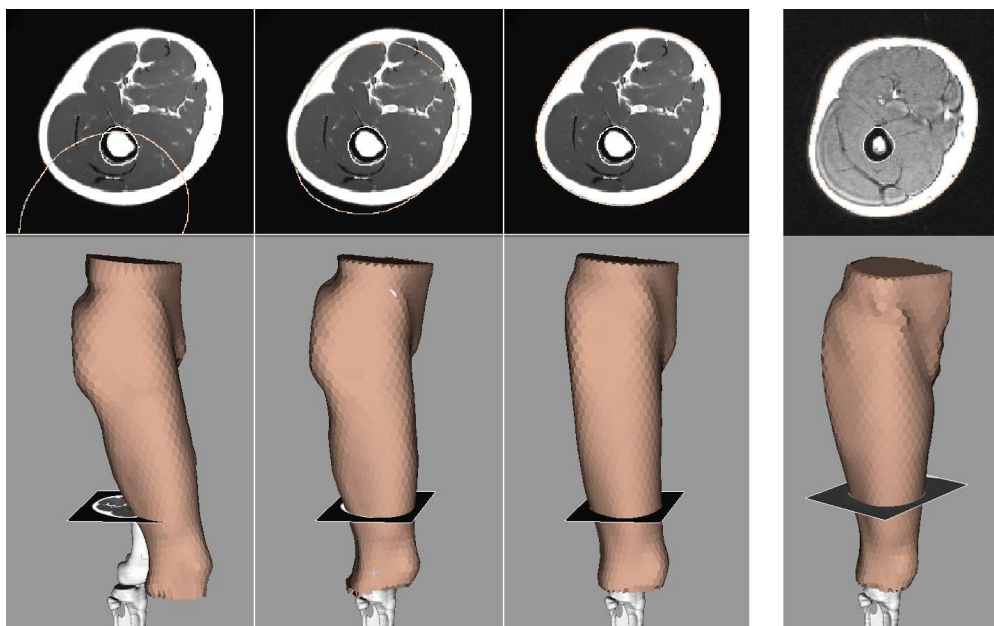


Figure 5.23: Result of skin automatic segmentation. Left: generic skin model; skin after initialisation from bone poses (skinning); Individual skin after deformable matching. Right: segmented skin from low resolution MRI

When using a low resolution MRI protocol such as the one presented in Figure 5.3 (resolution =  $1.37 \times 1.37 \times 5mm$ ), skin segmentation is also precise because skin is smooth (image resolution decrease has little influence) and because the noise level is low in this type of images. we could use exactly the same parameters for the two different protocols.

### Cartilage and ligament modelling

Cartilages and ligaments are thin periarticular tissues. Due to the highest resolution achievable with conventional MRI ( $\sim 0.5mm$ ), the use of image information is not sufficient to allow an accurate reconstruction of such tissues. Geometric and topological constraints start to be more reliable. For the hip joint, we know from anatomy literature that femoral and acetabular cartilages fill the inter-articular space, the interface being very smooth. The labrum links the femoral head and the acetabular rim to which it is attached. Ligaments have a quasi-constant thickness. Using our methods, it is possible to enforce such constraints: stable and accurate contact management imposes inter-organ constraints while the radial representation provides a volumetric control of the shape. Such as FEM shells, our medial axis represent thin tissues through orientation and thickness information.

To simplify cartilage medial axis construction, we have selected cartilage coverage regions on the bones themselves. So instead of a mobile medial axis, we use a fixed one where only radii change. This is coherent with most cartilage geometry studies where coverage is analysed using a thickness map [WTGW03]. Only external model vertices are active for radius computation and radial constraining. To ensure quasi-constant thickness [SS99], we smooth medial axis radii locally (such as curvature, we average radii in a certain neighbourhood for each vertex). The inter-articular distance should be the sum of the cartilage thickness, therefore, we split it into radii of facing medial axis vertices. For parts of the femoral cartilage outside the

joint, we use a minimum thickness of  $1.5mm$ . To enhance some topological information (cartilage perfect contact, labrum/ femoral cartilage contact), we use forces based on vertex projection (with  $\gamma = 0.2$ ). Other model forces are divided into smoothing (weighted barycentre and curvature averaging method) and radial forces. The balance between the two allows us to tune surface smoothness against thickness continuity. Concerning the labrum, we found that radial constraints are not suited for controlling shape because its medial axis is too free (it is tied to bone only at one extremity). Therefore we apply shape memory forces to the labrum model itself and radial forces to medial axis only (medial axis is still use to analyse labrum shape). For ligaments we fix the thickness uniformly to  $3mm$ . Ligament medial axis are simulated with shape memory and smoothing forces (equal contribution). Contrary to the labrum one, they are here the active elements (shape and collision constraints come from them). Ligament surfaces are simulated passively from medial axis position, through radial and smoothing forces (equal contribution). Collision handling is performed using medial axis/ medial axis and medial axis/ bone full collision test. Model sizes are summarised in Table 5.7.

Organs	Nb of model vertices (faces)	Nb of medial axis vertices (faces)
acetabular cartilage	1728 (866)	186 (94)
femoral cartilage	3040 (1520)	324(162)
labrum	1176 (590)	176 (89)
iliofemoral ligament	2944 (1474)	344 (173)
pubofemoral ligament	992 (498)	102 (52)
ischiofemoral ligament	704 (354)	84 (43)

Table 5.7: Properties of periarticular tissue models

As shown in Figure 5.24 and Video 5.6 , the fitting is performed sequentially from internal stiff organs to external elastic organs: cartilages are deformed first, then the labrum, then ligaments. At each step, the previous models are constrained with infinite mass. The complete fitting is obtained in  $\sim 3min$  with a minimum framerate of  $10frames/s$ . Thus, interaction (visualisation and constraint point placement) is, by far, possible during the segmentation process.

Because cartilage and ligaments are difficult to identify in MRI, our geometrical-based algorithm is difficult to validate. In general, we have found a good agreement between 3D models and anatomical structures, but a quantitative validation would be required from more suitable images (i.e. arthro-MRI using contrast agent injected inside the capsule). The advantages of our algorithm are the low computational time and the flexibility. Indeed, it can be easily tuned with different default thickness, different attachment areas and with user constraint point located inside or outside models. The high computational speed allows a fast pre-segmentation of the anatomy to be used by radiologists and orthopaedists during diagnosis and planning. We will see in section 5.10 that for unusual postures (e.g. split position), it is difficult to reconstruct mentally where tissues should be. Our software provides a good individualised estimation fully automatically and almost immediately using geometric constraints only.

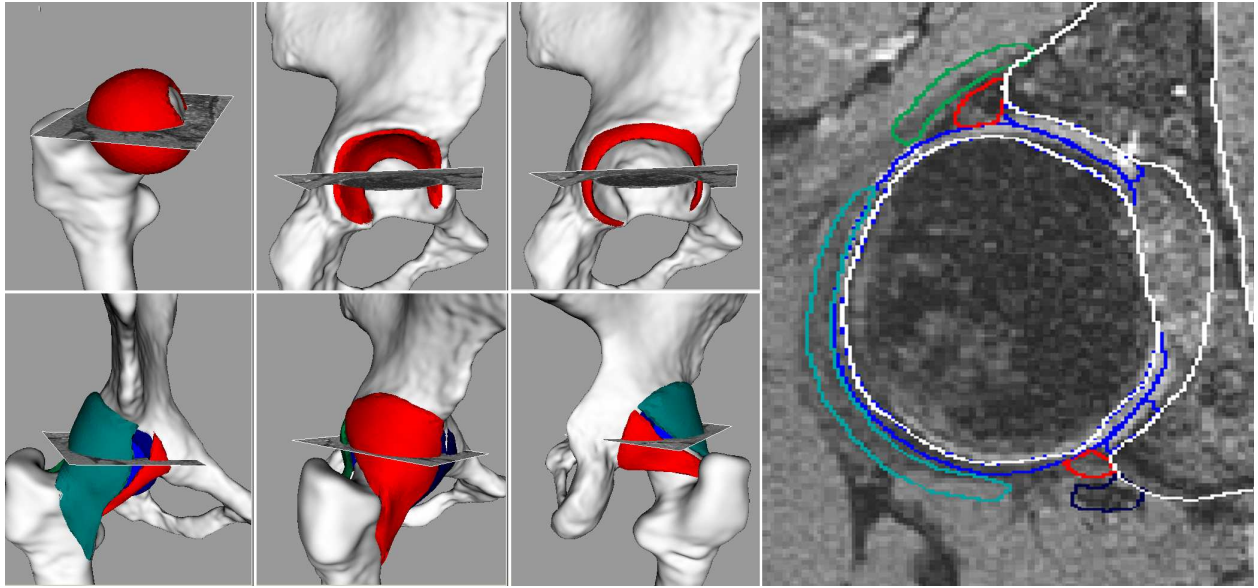


Figure 5.24: Automatically segmented periarticular tissues using our geometric method. Left (in red): femoral cartilage, acetabular cartilage, labrum, pubofemoral ligament, iliofemoral ligaments, ischiofemoral ligament; Right: MRI cutplane with overlaid models.

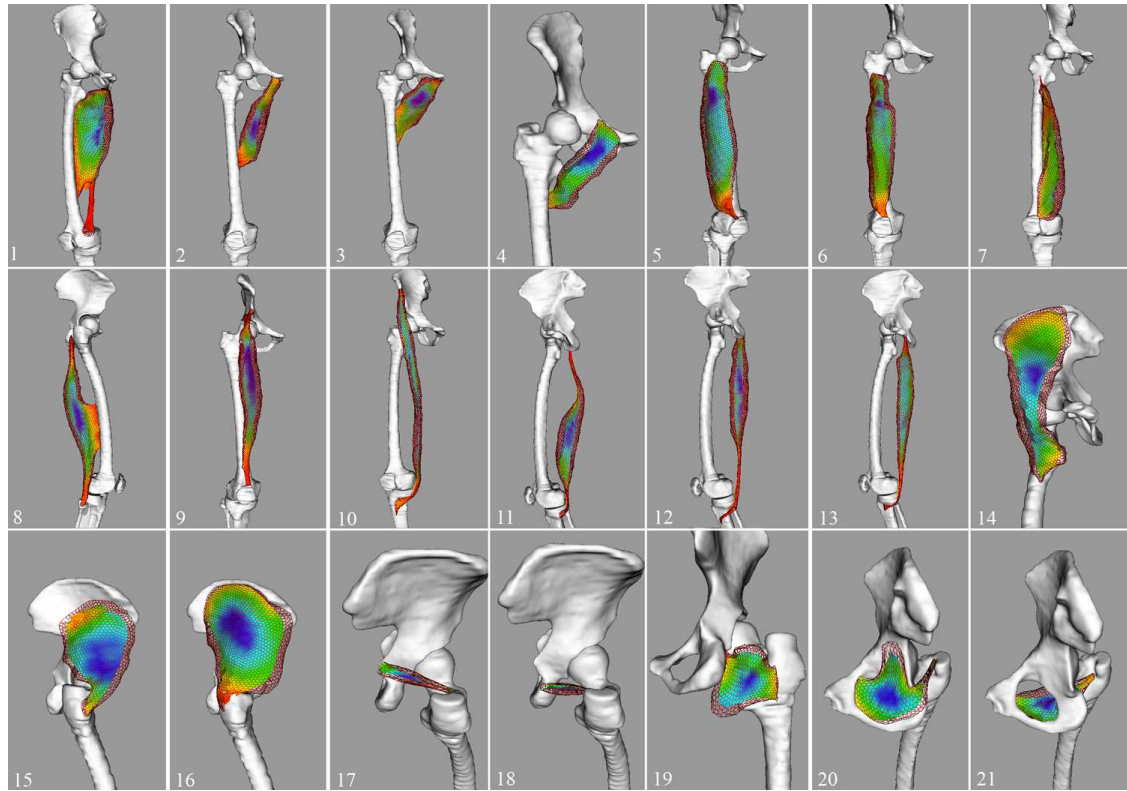
### Muscle modelling

We have modelled the 21 skeletal muscles of the hip and thigh as shown in Table 5.8 (71328 model vertices 4272 medial axis vertices in total). Information related to muscle actions is from the University of Washington<sup>1</sup>. Our methodology allows to represent the variety of shapes and topology they exhibit. Therefore we expect that it can be generalised to other human skeletal muscles (to do this, one has to segment image data through constraint points, place attachment splines, fit a model while optimising its mesh and finally compute the medial axis. It takes in total about one hour per organ).

We can use a different amount of prior information to perform muscle segmentation. The first level consists of using segmented bones only. Such as for cartilage and ligaments, we wrap generic attachments onto the segmented bones using barycentric coordinates of spline control points. Then we apply our skinning algorithm (Section 5.9) to initialise generic muscles near the contour to segment. We have tested different initialisation schemes that we compare in this section:

- **No initialisation:** only attachment points are wrapped.
- **TPS transform:** we apply the thin-plate-spline algorithm to transform generic muscles according to source (generic) and target (patient) attachment points. We have tested the individual wrapping of models according to their own attachment points (method *TPS (indiv.)* of Table 5.9) and the wrapping of all models according to all attachment points (method *TPS (all)*), which prevents from model interpenetrations. Results show that this is the most accurate initialisation method for such neutral postures. For high amplitude movements (Section 5.10), our skinning algorithm gives better results.
- **Skinning algorithm:** we transform model vertices from joint transforms and limb segment lengths (Section 5.9). We test different blending parameters  $\lambda$ . For such low amplitude differences in terms of

<sup>1</sup><http://www.rad.washington.edu/atlas2/>



Muscles	Action	Nb of model/ MA vertices
1 - Adductor magnus	Powerful thigh adductor; superior horizontal (vertical) fibers also help flex (extend) the thigh	4576 / 412
2 - Adductor longus	Adducts and flexes the thigh, and helps to laterally rotate the hip joint	3424 / 144
3 - Adductor brevis	Adducts and flexes the thigh, and helps to laterally rotate the thigh	3392 / 152
4 - Pectineus	Adducts the thigh and flexes the hip joint	2144 / 64
5 - Vastus lateralis	Extends the knee	6112 / 330
6 - Vastus intermedius	Extends the knee	5760 / 356
7 - Vastus medialis	Extends the knee	4288 / 346
8 - Biceps femoris	Flexes the knee, and also rotates the tibia laterally; long head also extends the hip joint	3808 / 302
9 - Rectus femoris	Extends the knee	5152 / 208
10 - Sartorius	Flexes and laterally rotates the hip joint and flexes the knee	3488 / 330
11 - Semimembranosus	Extends the thigh, flexes the knee, and also rotates the tibia medially, especially when the knee is flexed	3488 / 208
12 - Semitendinosus	Extends the thigh and flexes the knee, and also rotates the tibia medially, especially when the knee is flexed	4544 / 324
13 - Gracilis	Flexes the knee, adducts the thigh, and helps to medially rotate the tibia on the femur	3232 / 394
14 - Iliopsoas	Flex the torso and thigh with respect to each other	4896 / 158
15 - Gluteus minimus	Abducts and medially rotates the hip joint	2752 / 108
16 - Gluteus medius	Major abductor of thigh; anterior (posterior) fibers help to rotate hip medially (laterally)	4736 / 178
17 - Superior gemellus	Rotates the thigh laterally; also helps abduct the flexed thigh	416 / 42
18 - Inferior gemellus	Rotates the thigh laterally; also helps abduct the flexed thigh	352 / 16
19 - Quadratus femoris	Rotates the hip laterally; also helps adduct the hip	1600 / 54
20 - Obturator internus	Rotates the thigh laterally; also helps abduct the flexed thigh	1472 / 72
21 - Obturator externus	Rotates the thigh laterally; also helps adduct thigh	1696 / 74

Table 5.8: Muscle model description

joint angles, the blending parameter is not very important and we did not measure much differences. We will see that for kinematical analysis during high amplitude movements, it plays a major role.

The second level consists of using generic muscle shape parameters and the generic relative positions between them (*reference* collision handling). So here, only internal forces are computed and models deform to look similar to the generic ones. As discussed previously, we use medial axis shape parameters which are more compact and relevant. Until this point, we have not used image information. But we expect that medial axis deformation will be also driven by the overlying models (contrary to the last test of Section 4.8.3). So, as a general rule, we set medial axis force stiffness to  $\gamma = 0.6$  and  $0.4$  for shape memory and radial constraints respectively. For the models, radial and smoothing stiffness are set to  $0.4$  and  $0.2$ . We also add small volume conservation forces ( $\gamma = 0.05$ ) to prevent from model crushing when updating medial axis radii. This force will be increased for intra-patient segmentation (Section 5.10) since muscle are nearly incompressible. At this step, an important parameter is the maximum distance we use for computing the inter-object constraints (proximity parameter in Section 4.7.2) since it will drive object relative positions and most of the collision handling computational cost. Results show that using shape forces only is not accurate (method *internal forces*). Also it is important to take into account the skin in proximity detection (see method *no skin*). In Table 5.9, we test different distances and different collision primitives (model/ model or MA/ MA collision handling). We have found that a collision proximity distance of  $10mm$  provides the best results for the coarsest LOD, while we use a distance of  $4mm$  for the other LODs to decrease computational costs. Also, for the fine level of details, we use model/ model collision handling to improve the quality of contacts (to reduce errors due to medial axis approximation).

At the third level of prior information, we introduce image forces (method *external forces*). As already discussed, we split them in two: we combine a measure based on intensity profile (*NCC*) with a large search space (up to  $2cm$  backwards and  $2cm$  inwards), and a measure based on the gradient (*NG* measure) with a reduced search space (up to  $5mm$  backwards and  $5mm$  inwards). We apply a global affine regularisation from  $100\%$  to  $20\%$  and we smooth external forces in the local vertex neighbourhood. During the whole segmentation process, we progressively increase the number of degrees of freedom while reducing the magnitude of admissible deformation (see Figure 5.26). This is done through different means: first, we successively increase model level of details (geometric aspect). Second, we reduce image force search space and the global regularisation parameter (image processing and regularisation aspects). Finally, we update, at the end of the segmentation, generic shape parameters through medial axis radii updates (update of prior information).

Until now, our muscle segmentation method is fully automatic. But a fourth level of prior information can be added from user inputs. Indeed, during the segmentation, it is possible to interactively place constraint points on the images, to get a faster matching and a more accurate segmentation. Using our collision detection scheme, deformable models are forced to include or exclude these constraint points. A medical student has used this interactive segmentation to segment accurately 7 datasets. It provides gold standard datasets for validating our automatic segmentation method (we compute the average distance between the automatically and interactively segmented surfaces). Typically, about 2000 constraint points are necessary for a very accurate segmentation (about 30 minutes of manual work, method *gold standard*). But a little number of points can prevent from reaching most of the local minima (method *constraint points*) that are shown in Figure 5.28. So, we evaluate the application of about 100 points placed during the fitting process (the frame rate is always  $> 1frames/s$ , which allows interaction). Table 5.9 shows the average error of our method over the 7 different subjects (3 males and 4 females, different in terms of size and origin). These tests have been performed on the high resolution static images described in Table 5.2. Figure 5.26 and 5.27 show the typical error variation and computational time repartition. The segmentation process is also illustrated in Figure 5.25 and Video 5.7 .

Our segmentation method has proven to be valuable for muscle segmentation, since it provides a good

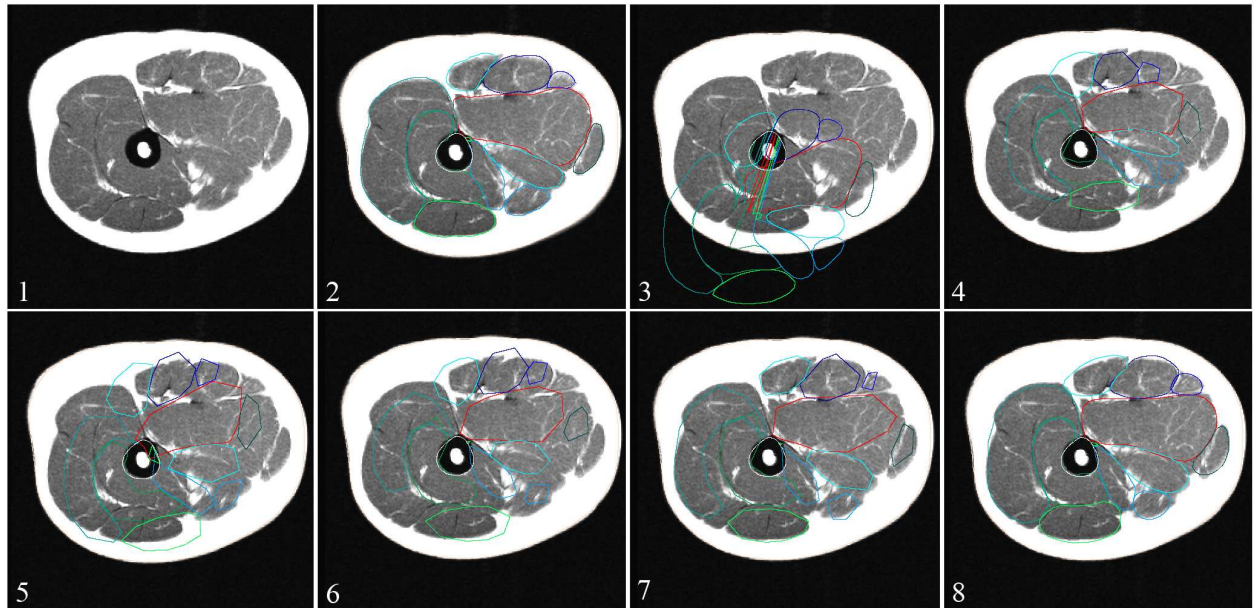


Figure 5.25: Typical segmentation results on a sample slice for different methods (see text)

Prior knowledge	Method	Image	Error $\pm$ std dev (in mm)	Comp. time (in s)
Individual bones + generic muscle models	Attachment wrapping	3	24.69 $\pm$ 22.59	0
	TPS (indiv.)		4.33 $\pm$ 4.75	1
	TPS (all)	4	4.34 $\pm$ 4.73	1
Joint transforms	Skinning ( $\lambda = 100$ )		5.95 $\pm$ 5.98	1
	Skinning ( $\lambda = 10$ )	5	5.75 $\pm$ 5.68	1
	Skinning ( $\lambda = 1$ )		5.56 $\pm$ 5.38	1
Shape parameters	Internal forces		5.48 $\pm$ 5.82	15
Inter-object proximities	Model-model ( $d = 20$ ), no skin		4.25 $\pm$ 4.22	20
	Model-model ( $d = 20$ )		3.73 $\pm$ 3.61	20
	Model-model ( $d = 10$ )		3.84 $\pm$ 3.67	20
	Model-model ( $d = 4$ )		4.18 $\pm$ 4.04	20
	MA-MA ( $d = 20$ )		3.79 $\pm$ 3.79	20
Images	MA-MA ( $d = 10$ )	6	3.79 $\pm$ 3.68	20
	MA-MA ( $d = 4$ )		4.01 $\pm$ 3.88	20
	External forces	8	1.58 $\pm$ 1.92	500
User inputs	Constraint points		1.37 $\pm$ 1.56	500
	Gold standard	2	0 $\pm$ 0	2000

Table 5.9: Muscle segmentation results

estimate of muscle shape (average error  $< 2mm$ ) in a time efficient way. However, it is not always robust in presence of false or fuzzy edges, and when anatomical variability from the generic model is too high (see Figure 5.28). This can be overcome through manual interactions. But the automatic method could be

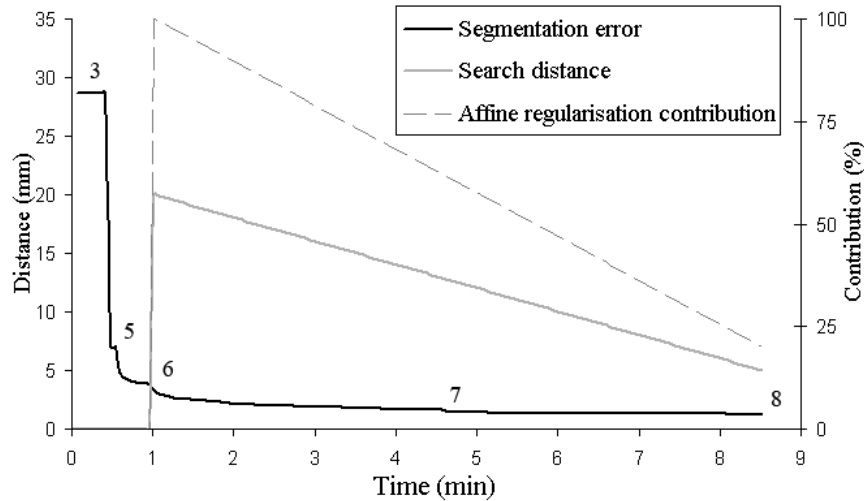


Figure 5.26: Error variation during the automatic muscle segmentation process. Numbers refer to images of Figure 5.25

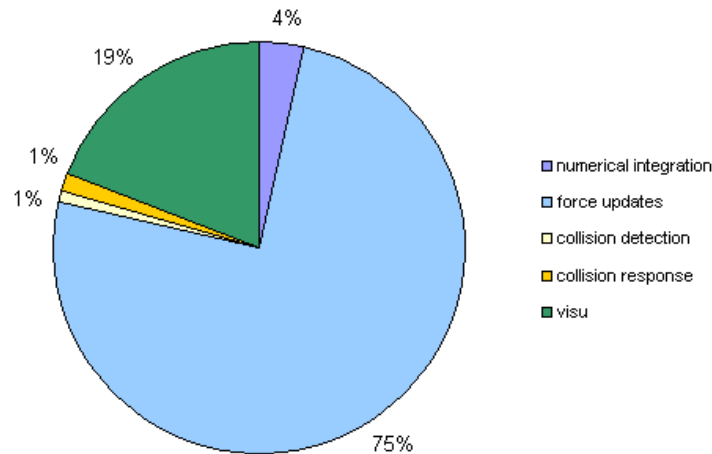


Figure 5.27: Computational weight repartition for our automatic muscle segmentation method

improved in future by combining statistically several generic models. We have evaluated the predictive power of our method (without using image information). Internal forces provides significantly better results than standard skinning algorithm based on joint transforms only. This is interesting since muscle shape could be estimated from other modalities than MRI: Motion capture data provides joint angles so our skinning algorithm could be applied if bones have been previously acquired from MRI or CT (the error would be about  $6mm$ ). Body scanners give skin models which can be used to derive internal forces (the error would be about  $4mm$ ). We did not individualise attachments since they mostly rely on bone geometrical features [KdH04]. Visually, it is difficult to identify muscle attachments on MRI, and we believe that it would be difficult to have an automatic technique for adjusting generic attachments.

We have also tested our algorithm on low resolution MRI (axial 3D T1, resolution =  $1.37 \times 1.37 \times 5mm$ , see Figure 5.3). Because contrasts are particularly different from the high resolution MR images, we have



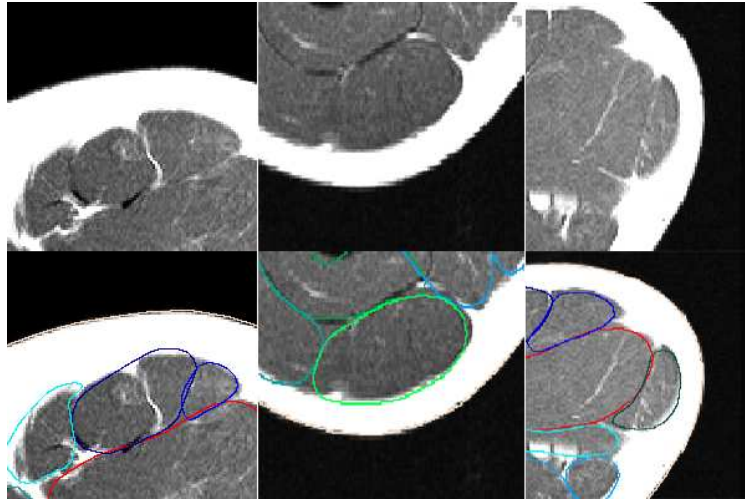


Figure 5.28: Typical errors encountered during the automatic muscle segmentation process

build a new generic model with this protocol, based on the same technique than above (construction of gold standard models), to derive the new reference intensity profiles. Validation tests on two datasets (two female dancers) have shown that the accuracy of our method is approximately the same than with high resolution MRI (see Figure 5.29). However, results are biased because gold standard models are less accurate due to a more difficult manual delineation of muscles (partial volume effects). The benefit of our method is to be able to provide, from geometric constraints, a good estimation of individual muscle position from images with low resolution and contrast.

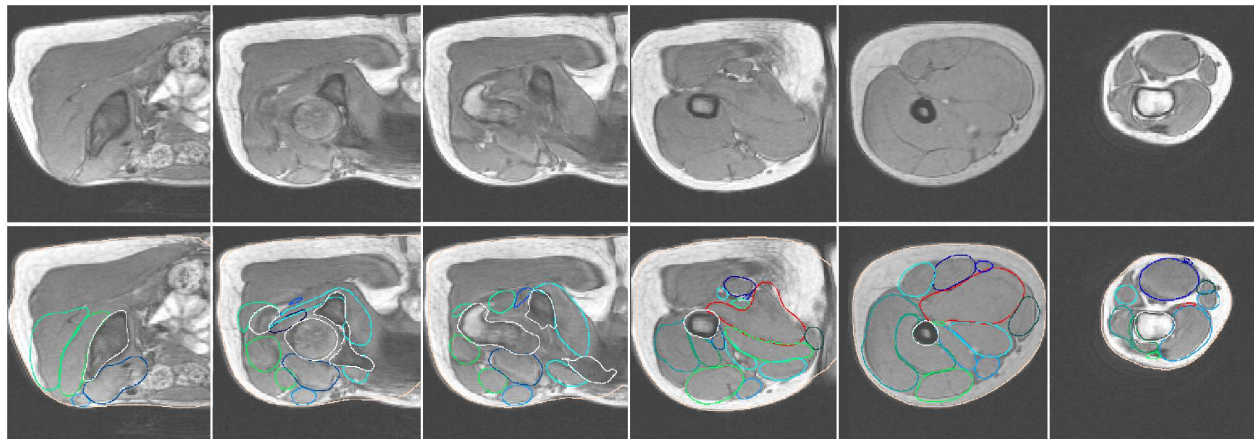


Figure 5.29: Muscle automatic segmentation from low resolution MRI

#### 5.5.4 Morphological analysis

Because our segmentation method deforms a generic shape and locally enforces its geometric features, we have exact correspondences across individuals (model registration). So, it is possible to compare their morphology which is medically important. Also, we can directly identify geometric features useful for standard axis computation (see Section 5.6). Under a collaboration with radiologists and orthopaedists, we have developed tools to quantify morphological features related to the hip joint. Bone shape analysis is

important in order to diagnose osteoarthritis problems and plan bone resection surgery. A first work was related to the computation of the version of the acetabulum, which can be a cause of pain (pincer impingement due to anteversion or retroversion [PMD<sup>+</sup>06]). We have implemented the standard measurement method from [RLK99], in a fully automatic fashion. It is based on the selection of acetabular rim points (obtained from the generic model) and their connection in axial planes (see Figure 5.30). Then, we measure the angle between these lines and the frontal direction. Contrary to [RLK99], we use normalised directions according to anatomical points instead of image directions. It eliminates the dependence with patient position inside the scanner. A standard parameter related to femur geometry is the alpha angle which is an important indicator of cam impingements [PMD<sup>+</sup>06] but requires the identification of the hip joint center (see Figure 5.31). Our method automatically fits a sphere to approximate the femoral head to simplify and reduce errors in the common manual measurement.

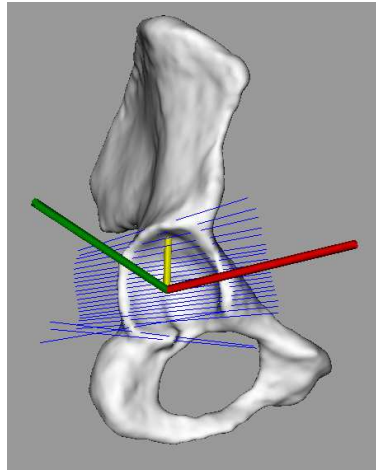


Figure 5.30: Computation of the version of the acetabulum

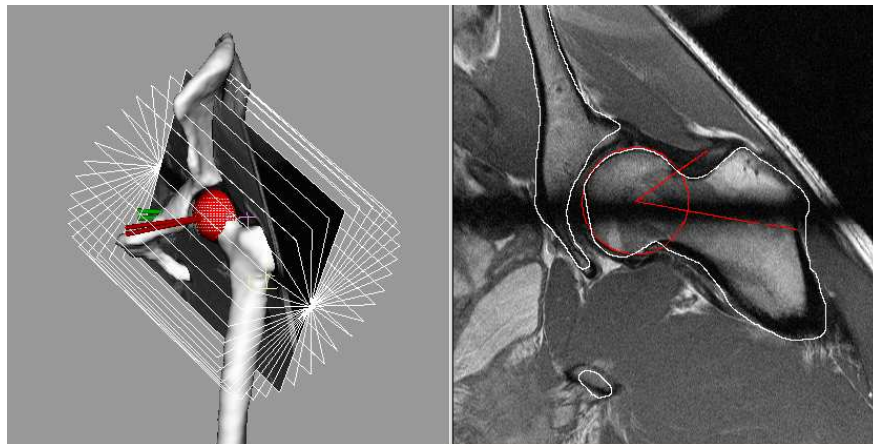


Figure 5.31: Computation of the femur alpha angle

High-level descriptors such as the medial axis convey more information than local descriptors (curvature). Soft-tissue thickness can be simply analysed through medial axis radii comparison, as shown in Figure 5.32. Using approximated geodesic distances to attachments (Dijkstra algorithm), we compute normalized

coordinates  $X$  and  $Y$  along medial surfaces, from which a thickness profile can be extracted (maximum radius in  $Y$  direction). For some muscles showing thickness steep changes, tendons lengths (which is an important biomechanical parameter) can be automatically extracted.

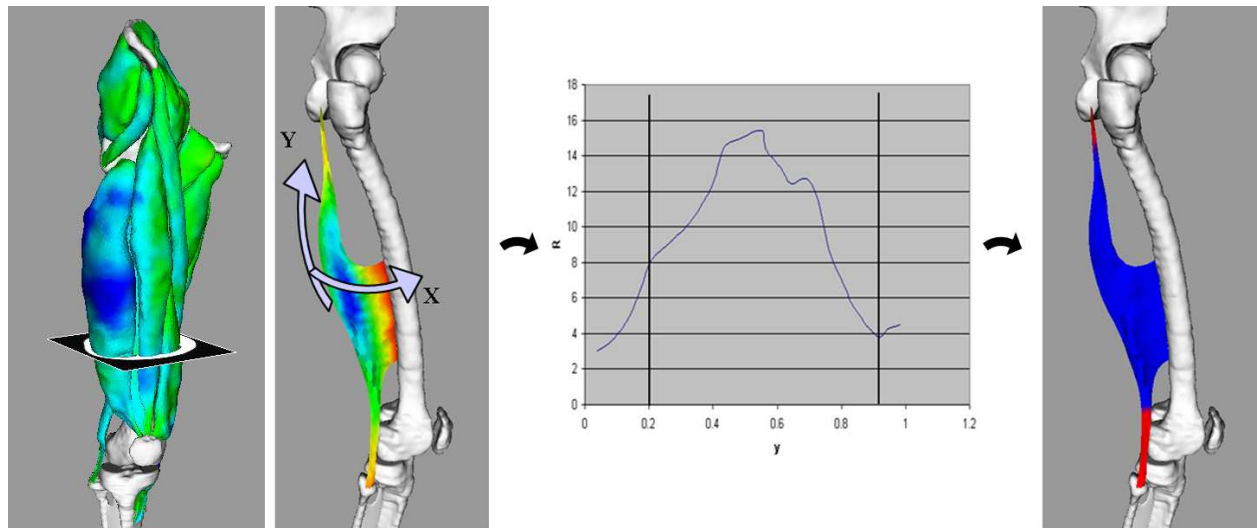


Figure 5.32: Medial axis-based morphological analysis. Left: muscle thickness comparison between two subjects. Right: thickness profile analysis along  $Y$  geodesic direction and tendon extraction

## 5.6 Joint coordinate systems

To report joint motion in a repeatable way, that is independently to the acquisition frame, standard joint coordinate systems have been defined based on anatomical landmarks only [GS83] [WSA<sup>+</sup>02]. They have been chosen to describe joint motion in clinically relevant terms (i.e. for the hip joint: flexion/ extension, internal/ external rotation, abduction/ adduction) allowing a better communication among researchers and clinicians. For each joint, a standard for the local axis system in each articulating bone is generated. These axes then standardise the joint coordinate system. We use the definitions proposed by the Standardisation and Terminology Committee (STC) of the International Society of Biomechanics (ISB) [WSA<sup>+</sup>02] (ankle and hip) and from the seminal work of Grood and Suntay [GS83] (knee). Here we recall them for the lower joints. All bone coordinate systems and landmarks are expressed in the world system (scanner frame). Orthonormal bone systems are equivalent to  $4 \times 4$  homogenous transform matrices (rigid transforms, see Section 3.3.4).

The patella coordinate systems  $\mathbf{MPt}_R$  and  $\mathbf{MPt}_L$  has not been defined in the standards. Therefore, we use the tibia coordinate system defined for the knee ( $\mathbf{MPt}_R = \mathbf{MT}_R^K$  and  $\mathbf{MPt}_L = \mathbf{MT}_L^K$ ). In doing so, we suppose that the patella is in neutral position during the acquisition.

The main benefit of using non-orthonormal joint coordinate systems is that transformations are sequence independent (that is the final transformation is independent to the order of the rotations). On the contrary, Euler angles use one of the two orthonormal frame as a reference, which introduces some confusion relative to nomenclature. As described in [GS83], standard medical angles and shifts can be easily converted into (or defined from) homogenous matrices. These matrices represent the relative transform between one bone to another, from the neutral position (e.g.  $\mathbf{RPF}$  represent the transform from the pelvis to the femur and is equivalent to the hip angles/ shifts). Now, we link the different transforms in the world coordinate system (scanner frame). As shown in Figure 5.33, bone global transforms are denoted by  $\mathbf{T}$ , bone systems in the

Bones	Right side	Left side
Pelvis	<ul style="list-style-type: none"> <li>• <math>\mathbf{A}_R</math>: right anterior superior iliac spine</li> <li>• <math>\mathbf{P}_R</math>: right posterior superior iliac spine</li> <li>• <math>\mathbf{HJC}_R</math>: right hip center of rotation (attached to the pelvis)</li> </ul>	<ul style="list-style-type: none"> <li>• <math>\mathbf{A}_L</math>: left anterior superior iliac spine</li> <li>• <math>\mathbf{P}_L</math>: left posterior superior iliac spine</li> <li>• <math>\mathbf{HJC}_L</math>: left hip center of rotation (attached to the pelvis)</li> </ul>
	$\mathbf{P} = (\mathbf{P}_R + \mathbf{P}_L)/2$ : mid posterior superior iliac spine	
Femur	<ul style="list-style-type: none"> <li>• <math>\mathbf{FE}_{LR}</math>: right femoral lateral epicondyle</li> <li>• <math>\mathbf{FE}_{MR}</math>: right femoral medial epicondyle</li> <li>• <math>\mathbf{FE}_R = (\mathbf{FE}_{LR} + \mathbf{FE}_{MR})/2</math>: midpoint of the right femoral epicondyle</li> </ul>	<ul style="list-style-type: none"> <li>• <math>\mathbf{FE}_{LL}</math>: left femoral lateral epicondyle</li> <li>• <math>\mathbf{FE}_{ML}</math>: left femoral medial epicondyle</li> <li>• <math>\mathbf{FE}_L = (\mathbf{FE}_{LL} + \mathbf{FE}_{ML})/2</math>: midpoint of the left femoral epicondyle</li> </ul>
Tibia/ fibula	<ul style="list-style-type: none"> <li>• <math>\mathbf{M}_{MR}</math>: tip of the right medial malleolus</li> <li>• <math>\mathbf{M}_{LR}</math>: tip of the right lateral malleolus</li> <li>• <math>\mathbf{C}_{MR}</math>: the most medial point on the border of the right medial tibial condyle</li> <li>• <math>\mathbf{C}_{LR}</math>: the most lateral point on the border of the right lateral tibial condyle</li> <li>• <math>\mathbf{TT}_R</math>: right tibial tuberosity</li> <li>• <math>\mathbf{M}_R = (\mathbf{M}_{MR} + \mathbf{M}_{LR})/2</math>: the right inter-malleolar point</li> <li>• <math>\mathbf{C}_R = (\mathbf{C}_{MR} + \mathbf{C}_{LR})/2</math>: the right inter-condylar point</li> </ul>	<ul style="list-style-type: none"> <li>• <math>\mathbf{M}_{ML}</math>: tip of the left medial malleolus</li> <li>• <math>\mathbf{M}_{LL}</math>: tip of the left lateral malleolus</li> <li>• <math>\mathbf{C}_{ML}</math>: the most medial point on the border of the left medial tibial condyle</li> <li>• <math>\mathbf{C}_{LL}</math>: the most lateral point on the border of the left lateral tibial condyle</li> <li>• <math>\mathbf{TT}_L</math>: left tibial tuberosity</li> <li>• <math>\mathbf{M}_L = (\mathbf{M}_{ML} + \mathbf{M}_{LL})/2</math>: the left inter-malleolar point</li> <li>• <math>\mathbf{C}_L = (\mathbf{C}_{ML} + \mathbf{C}_{LL})/2</math>: the left inter-condylar point</li> </ul>
Calcaneus	<ul style="list-style-type: none"> <li>• Plane <math>PA_R</math>: plantar aspect of the right foot</li> <li>• <math>\mathbf{LA}_R</math>: long axis of the second metatarsal of the right foot</li> </ul>	<ul style="list-style-type: none"> <li>• Plane <math>PA_L</math>: plantar aspect of the left foot</li> <li>• <math>\mathbf{LA}_L</math>: long axis of the second metatarsal of the left foot</li> </ul>

Table 5.10: Anatomical landmarks

acquisition frame by  $\mathbf{M}$  and bone relative transforms by  $\mathbf{R}$ . All these matrices are  $4 \times 4$  rigid transform matrices (see Section 3.3). Here are the different relationships (same expressions for the right and left sides):

Hip transforms:

- $\mathbf{RPF} = \mathbf{MP}^{-1}.\mathbf{TP}^{-1}.\mathbf{TF}.\mathbf{MF}^H$
- $\mathbf{RFP} = \mathbf{RPF}^{-1}$

Knee transforms:

- $\mathbf{RFT} = (\mathbf{MF}^K)^{-1}.\mathbf{TF}^{-1}.\mathbf{TT}.\mathbf{MT}^K$
- $\mathbf{RTF} = \mathbf{RFT}^{-1}$
- $\mathbf{RTPt} = (\mathbf{MT}^K)^{-1}.\mathbf{TT}^{-1}.\mathbf{TPt}.\mathbf{MPt}$

Bones	Right side	Left side
Pelvis	$MP_R:$ <ul style="list-style-type: none"> <li>• <math>O = HJC_R</math></li> <li>• <math>Z = \frac{A_L A_R}{\ A_L A_R\ }</math></li> <li>• <math>Y = \frac{P_{AR} \wedge P_{AL}}{\ P_{AR} \wedge P_{AL}\ }</math></li> <li>• <math>X = Y \wedge Z</math></li> </ul>	$MP_L:$ <ul style="list-style-type: none"> <li>• <math>O = HJC_L</math></li> <li>• <math>Z = \frac{A_L A_R}{\ A_L A_R\ }</math></li> <li>• <math>Y = \frac{P_{AR} \wedge P_{AL}}{\ P_{AR} \wedge P_{AL}\ }</math></li> <li>• <math>X = Y \wedge Z</math></li> </ul>
Femur	$MF_R^H$ (hip) and $MF_R^K$ (knee): <ul style="list-style-type: none"> <li>• <math>O = HJC_R</math> in neutral position (hip) or <math>FE_R</math> (knee)</li> <li>• <math>X = \frac{HJC_R FE_{LR} \wedge HJC_R FE_{MR}}{\ HJC_R FE_{LR} \wedge HJC_R FE_{MR}\ }</math></li> <li>• <math>Y = \frac{FE_R HJC_R}{\ FE_R HJC_R\ }</math></li> <li>• <math>Z = X \wedge Y</math></li> </ul>	$MF_L^H$ (hip) and $MF_L^K$ (knee): <ul style="list-style-type: none"> <li>• <math>O = HJC_L</math> in neutral position (hip) or <math>FE_L</math> (knee)</li> <li>• <math>X = \frac{HJC_L FE_{ML} \wedge HJC_L FE_{LL}}{\ HJC_L FE_{ML} \wedge HJC_L FE_{LL}\ }</math></li> <li>• <math>Y = \frac{FE_L HJC_L}{\ FE_L HJC_L\ }</math></li> <li>• <math>Z = X \wedge Y</math></li> </ul>
Tibia/ fibula	$MT_R^K$ (knee) and $MT_R^A$ (ankle): <ul style="list-style-type: none"> <li>• <math>O = M_R</math> (ankle) or <math>FE_R</math> (knee)</li> <li>• <math>Z = \frac{M_{MR} M_{LR}}{\ M_{MR} M_{LR}\ }</math> (ankle) or <math>X \wedge Y</math> (knee)</li> <li>• <math>X = \frac{C_R M_{LR} \wedge C_R M_{MR}}{\ C_R M_{LR} \wedge C_R M_{MR}\ }</math></li> <li>• <math>Y = Z \wedge X</math> (ankle) or <math>\frac{M_R C_R}{\ M_R C_R\ }</math> (knee)</li> </ul>	$MT_L^K$ (knee) and $MT_L^A$ (ankle): <ul style="list-style-type: none"> <li>• <math>O = M_L</math> (ankle) or <math>FE_R</math> (knee)</li> <li>• <math>Z = \frac{M_{LL} M_{ML}}{\ M_{LL} M_{ML}\ }</math> (ankle) or <math>X \wedge Y</math> (knee)</li> <li>• <math>X = \frac{C_L M_{LL} \wedge C_L M_{ML}}{\ C_L M_{LL} \wedge C_L M_{ML}\ }</math></li> <li>• <math>Y = Z \wedge X</math> (ankle) or <math>\frac{M_L C_L}{\ M_L C_L\ }</math> (knee)</li> </ul>
Calcaneus	$MC_R:$ <ul style="list-style-type: none"> <li>• <math>O = M_R</math> in neutral position</li> <li>• <math>Y</math> axis perpendicular to the plantar aspect of the right foot</li> <li>• <math>X</math> long axis of the second metatarsal of the right foot</li> <li>• <math>Z = X \wedge Y</math></li> </ul>	$MC_L:$ <ul style="list-style-type: none"> <li>• <math>O = M_L</math> in neutral position</li> <li>• <math>Y</math> axis perpendicular to the plantar aspect of the left foot</li> <li>• <math>X</math> long axis of the second metatarsal of the left foot</li> <li>• <math>Z = X \wedge Y</math></li> </ul>

Table 5.11: Bone coordinate systems

Ankle transforms:

- $RTC = (MT^A)^{-1} . TT^{-1} . TC . MC$
- $RCT = RTC^{-1}$

When registering a generic model, shape constraints are applied (Section 4.6.2) to derive anatomical correspondences, so that landmarks can be automatically find from the generic ones (such as attachment splines, Section. 5.4.2). Thus except from the hip joint center (see next section), all parameters needed for bone coordinate system computation are obtained automatically by storing generic landmark positions on the generic bone surfaces (barycentric coordinates).

Hip	$\mathbf{e}_1 = \mathbf{Z}_{\text{pelvis}}$ : <ul style="list-style-type: none"> <li>• Flexion/ extension <math>\alpha = \widehat{\mathbf{X}_{\text{pelvis}}\mathbf{e}_2}</math>: rotation around <math>\mathbf{e}_1</math></li> <li>• Mediolateral translation <math>\mathbf{q}_1 = \mathbf{O}_{\text{pelvis}}\mathbf{O}_{\text{femur}}\cdot\mathbf{e}_1</math>: translation along <math>\mathbf{e}_1</math></li> </ul> $\mathbf{e}_3 = \mathbf{Y}_{\text{femur}}$ : <ul style="list-style-type: none"> <li>• Internal/ external rotation <math>\gamma = \widehat{\mathbf{X}_{\text{femur}}\mathbf{e}_2}</math>: rotation around <math>\mathbf{e}_3</math></li> <li>• Proximo-distal translation <math>\mathbf{q}_3 = \mathbf{O}_{\text{pelvis}}\mathbf{O}_{\text{femur}}\cdot\mathbf{e}_3</math>: translation along <math>\mathbf{e}_3</math></li> </ul> $\mathbf{e}_2 = \mathbf{e}_3 \wedge \mathbf{e}_1$ : <ul style="list-style-type: none"> <li>• Adduction/ abduction <math>\beta = \pi/2 - \widehat{\mathbf{e}_3\mathbf{e}_1}</math>: rotation around <math>\mathbf{e}_2</math></li> <li>• Antero-posterior translation <math>\mathbf{q}_2 = \mathbf{O}_{\text{pelvis}}\mathbf{O}_{\text{femur}}\cdot\mathbf{e}_2</math>: translation along <math>\mathbf{e}_2</math></li> </ul>
Knee	$\mathbf{e}_1 = \mathbf{Z}_{\text{femur}}$ : <ul style="list-style-type: none"> <li>• Flexion/ extension <math>\alpha = \widehat{\mathbf{X}_{\text{femur}}\mathbf{e}_2}</math>: rotation around <math>\mathbf{e}_1</math></li> <li>• Mediolateral translation <math>\mathbf{q}_1 = \mathbf{O}_{\text{femur}}\mathbf{O}_{\text{tibia}}\cdot\mathbf{e}_1</math>: translation along <math>\mathbf{e}_1</math></li> </ul> $\mathbf{e}_3 = \mathbf{Y}_{\text{tibia}}$ : <ul style="list-style-type: none"> <li>• Internal/ external rotation <math>\gamma = \widehat{\mathbf{X}_{\text{tibia}}\mathbf{e}_2}</math>: rotation around <math>\mathbf{e}_3</math></li> <li>• Proximo-distal translation <math>\mathbf{q}_3 = \mathbf{O}_{\text{femur}}\mathbf{O}_{\text{tibia}}\cdot\mathbf{e}_3</math>: translation along <math>\mathbf{e}_3</math></li> </ul> $\mathbf{e}_2 = \mathbf{e}_3 \wedge \mathbf{e}_1$ : <ul style="list-style-type: none"> <li>• Adduction/ abduction <math>\beta = \pi/2 - \widehat{\mathbf{e}_3\mathbf{e}_1}</math>: rotation around <math>\mathbf{e}_2</math></li> <li>• Antero-posterior translation <math>\mathbf{q}_2 = \mathbf{O}_{\text{femur}}\mathbf{O}_{\text{tibia}}\cdot\mathbf{e}_2</math>: translation along <math>\mathbf{e}_2</math></li> </ul>
Ankle	$\mathbf{e}_1 = \mathbf{Z}_{\text{tibia}}$ : <ul style="list-style-type: none"> <li>• dorsiflexion/ plantarflexion <math>\alpha = \widehat{\mathbf{X}_{\text{tibia}}\mathbf{e}_2}</math>: rotation around <math>\mathbf{e}_1</math></li> <li>• Mediolateral shift <math>\mathbf{q}_1 = \mathbf{O}_{\text{tibia}}\mathbf{O}_{\text{calcaneus}}\cdot\mathbf{e}_1</math>: translation along <math>\mathbf{e}_1</math></li> </ul> $\mathbf{e}_3 = \mathbf{Y}_{\text{calcaneus}}$ : <ul style="list-style-type: none"> <li>• Internal/ external rotation <math>\gamma = \widehat{\mathbf{X}_{\text{calcaneus}}\mathbf{e}_2}</math>: rotation around <math>\mathbf{e}_3</math></li> <li>• Compression/ distraction <math>\mathbf{q}_3 = \mathbf{O}_{\text{tibia}}\mathbf{O}_{\text{calcaneus}}\cdot\mathbf{e}_3</math>: translation along <math>\mathbf{e}_3</math></li> </ul> $\mathbf{e}_2 = \mathbf{e}_3 \wedge \mathbf{e}_1$ : <ul style="list-style-type: none"> <li>• Inversion/ eversion <math>\beta = \pi/2 - \widehat{\mathbf{e}_3\mathbf{e}_1}</math>: rotation around <math>\mathbf{e}_2</math></li> <li>• Antero-posterior drawer <math>\mathbf{q}_2 = \mathbf{O}_{\text{tibia}}\mathbf{O}_{\text{calcaneus}}\cdot\mathbf{e}_2</math>: translation along <math>\mathbf{e}_2</math></li> </ul>

Table 5.12: Right (or left) joint coordinate systems

## 5.7 Functional hip joint center computation

In most biomechanical studies, the hip joint is considered as a ball and socket joint, meaning that no shifts are allowed (perfect joint). The location of the hip joint center (HJC) is a important parameter for computing joint angles and moment arms, and for pre-surgical planning (implant placement). In the literature, two main approaches have been presented to estimate the HJC:

- The predictive approach [BPB90] estimates the HJC as a relative position of anatomical landmarks (static).

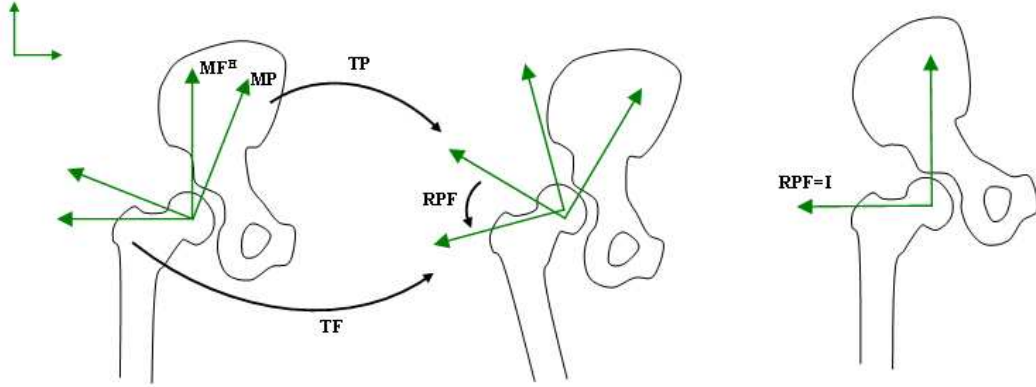


Figure 5.33: Bone and joint transforms in the reference (acquisition) position (left), in a user position (middle) and in the neutral position (right)

- The functional approach estimates the HJC from recorded [Cap84] or simulated [KSMMT03] movements of the joint.

It has been reported that the functional approach is more accurate since it accounts for joint dynamics [WSA<sup>+</sup>02]. However, the true HJC is unknown in kinematics studies because they mostly use external movement (motion capture data). In [SD06], the functional method is tested with a known HJC (phantom), and a smallest error of  $2.2 \pm 0.2mm$  is reported (circumduction motion pattern). In [KSMMT03], a hip joint 3D model, reconstructed from MRI, is used to estimate the HJC. In their method, an initial HJC is placed manually (by fitting a sphere inside the acetabulum). Then, the proposed algorithm adjusts the HJC by testing points around the initial guess, the goal being to minimise pelvis/ femur collisions during low amplitude circumduction. The algorithm was not validated since motion was only simulated. Also, it turns to heavy computation (several hours) for an accurate estimate and the manual initialisation is a source of error. An interesting idea in using patient specific 3D models instead of generic rigid bodies ideally articulated, is that the hip joint is no more considered as a perfect ball and socket. Indeed, it has been reported that both the femur and the pelvis are not strictly spherical [Men97]. Especially, for extreme movements, it is interesting to know if joint incongruity has any impact on joint degeneration through joint subluxation and excessive labral deformations. So, the estimation and the validation of femoro-acetabular movements are important. Here we propose to extent the method of Kang et al. (automation and speeding up of the process) and further confront it to real motion data. This is done in the following steps:

### 5.7.1 Hip joint center initialisation

It is possible to find the best sphere fitting to a set of points by least square fitting of data [SE03]. We manually select the points of interest (acetabulum/ femoral head) on the generic bones to automatically obtain them from any individual models. For these points, the function to minimise is given by:  $\sum_{i < N} (\|\mathbf{P}_i \mathbf{C}\| - r)^2$  where  $\mathbf{C}$  is the HJC and  $r$  the radius of the sphere. Setting the function derivatives with regards to  $\mathbf{C}$  and  $r$ , we obtain the following iterative process that quickly converges:

$$\begin{aligned}
r_j &= \frac{1}{N} \sum_i \|\mathbf{P}_i \mathbf{C}_j\| \\
\mathbf{C}_{j+1} &= \mathbf{C}_0 + \frac{r_j}{N} \sum_i \frac{\mathbf{P}_i \mathbf{C}_j}{\|\mathbf{P}_i \mathbf{C}_j\|} \\
\text{With: } \mathbf{C}_0 &= \langle \mathbf{P}_i \rangle_{i=1}^N = \frac{1}{N} \sum_i \mathbf{P}_i
\end{aligned}$$

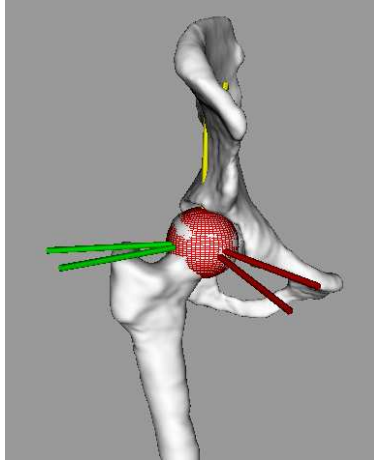


Figure 5.34: Hip joint center estimation through femoral head spherical approximation

Using acetabulum points and using femoral head points do not lead to the same results (about  $2mm$  difference) due to an inhomogeneous inter-articular distance. Extending the above method with two spheres centered on the same point provides the best estimate of the HJC, assuming a constant inter-articular distance (perfect ball and socket). In Section 5.8.3, we will show that this is a more accurate prediction than using one sphere only. Let  $\mathbf{P}_i, i < N$  be the acetabulum points and  $\mathbf{Q}_i, i < M$  the femoral head points. The fitting process is given by:

$$\begin{aligned}
r1_j &= \frac{1}{N} \sum_{i < N} \|\mathbf{P}_i \mathbf{C}_j\| \\
r2_j &= \frac{1}{M} \sum_{i < M} \|\mathbf{Q}_i \mathbf{C}_j\| \\
\mathbf{C}_{j+1} &= \mathbf{C}_0 + \frac{1}{N+M} \left( r1_j \sum_{i < N} \frac{\mathbf{P}_i \mathbf{C}_j}{\|\mathbf{P}_i \mathbf{C}_j\|} + r2_j \sum_{i < M} \frac{\mathbf{Q}_i \mathbf{C}_j}{\|\mathbf{Q}_i \mathbf{C}_j\|} \right) \\
\text{With: } \mathbf{C}_0 &= \frac{1}{N+M} \left( \sum_{i < N} \mathbf{P}_i + \sum_{i < M} \mathbf{Q}_i \right)
\end{aligned}$$

### 5.7.2 Collision detection and femoro-acetabular movement simulation

Given the initialised hip joint center and bone coordinate systems described above ( $\mathbf{MF}^H$  and  $\mathbf{MP}$ ), it is now possible to check femur/ pelvis collisions given certain joint angles and shifts (expressed through the



matrix  $\mathbf{RFP}$ ). In Section 4.7.2, we have presented a fast method for collision detection between rigid object that we are using here. For it, we need to pre-compute femur and pelvis distance maps (they are associated to the acquisition coordinate system). To check collisions for a given posture, we use the transformation  $\mathbf{M} = \mathbf{MF}^H \cdot \mathbf{RFP} \cdot \mathbf{MP}^{-1}$  that maps any point of the pelvis to the femur distance map (reciprocally,  $\mathbf{M}^{-1}$  maps any point of the femur to the pelvis distance map). This is illustrated in Figure 5.35.

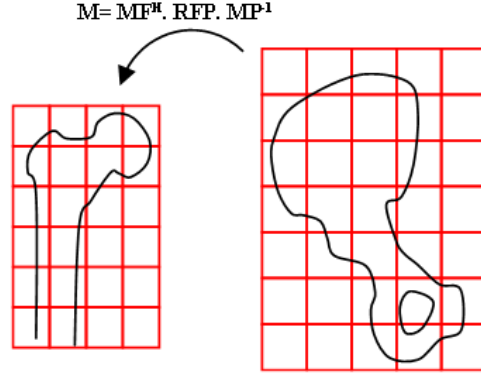


Figure 5.35: Femur/ pelvis collision detection

In order to simulate the femoro-acetabular movement, our goal is to stick to a reference inter-articular distance. We can set it to a constant value (radius difference between the two initialisation spheres  $d_i^{ref} = r_2 - r_1, \forall i < M$ ) or to the initial distance (see Figure 5.36.a). We will check the difference in Section 5.8.3. Some femoral head points go in/ out from the acetabulum, therefore, we try to seek  $d_i = d_i^{ref}$  for pelvis points only (see Figure 5.36.b). For a given joint transform, we minimise the differences  $|d_i - d_i^{ref}|$  in an iterative process through pelvis infinitesimal translations. At each iteration, the rigid transform matrix  $\mathbf{M}$  is translated by:

$$\mathbf{t} = \frac{\varepsilon}{M} \sum_{i < M} (d_i - d_i^{ref}) \mathbf{M}_R \mathbf{n}_i$$

where  $\mathbf{n}_i$  is the normal of the pelvis at  $i$  (in the acquisition frame),  $\mathbf{M}_R$  is the rotational part of  $\mathbf{M}$  and  $\varepsilon$  a parameter that weight the translation (set to 0.2). The process typically converges in 300 iterations ( $\|\mathbf{t}\| < 1\mu m$ ) for a total translation  $< 3mm$ . The total computation time is about 0.2s for 2877 acetabulum points.

### 5.7.3 Hip joint center estimation

The algorithm presented above can optimise any joint transform in terms of shifts to seek a reference inter-articular distance. In [SD06] and [KSMMT03], it has been shown that a circumduction motion pattern is the better trajectory to estimate the hip joint center because all types of rotations are involved. A circumduction motion pattern is a conical movement characterised by two parameters: the elevation  $\alpha$  and the angle  $0 \leq \beta < 2\pi$  (see Figure 5.37). The joint transform is express by the unit quaternion:  $\mathbf{q} = [\cos \alpha, \cos \beta \sin \alpha, 0, \sin \beta \sin \alpha]$  (the multiplication by  $\sin \alpha$  comes from the normalisation to unit length). We straightforwardly obtain the joint transform matrix  $\mathbf{RFP}$  by converting this quaternion.

Given the set of transforms  $\mathbf{M}_i, i < N$  corresponding to a circumduction where translations have been optimised, our goal is now to find the point in the pelvis frame that moves the less in the femur frame

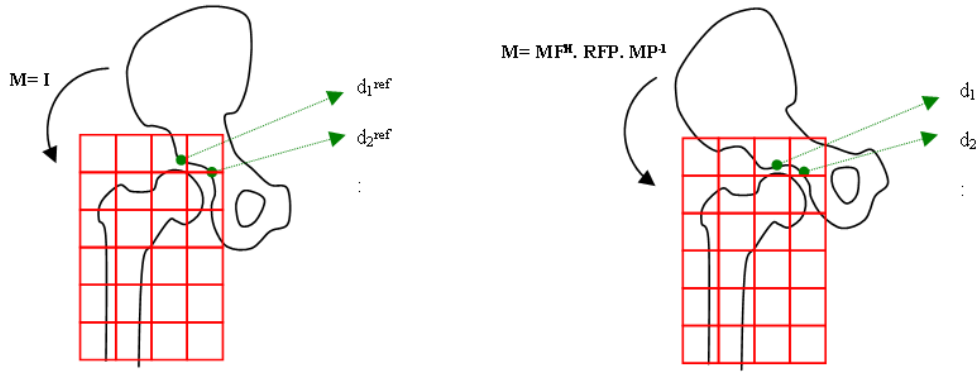


Figure 5.36: a. Reference inter-articular distance (acquisition posture) and b. inter-articular distance for a given joint transform

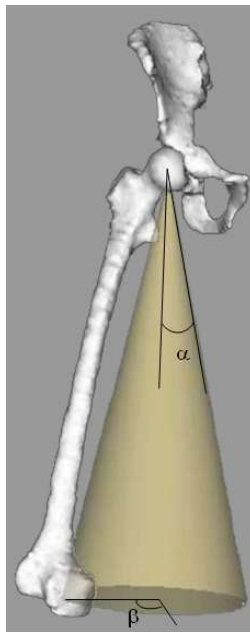


Figure 5.37: Circumduction motion pattern of the hip

(the HJC estimate  $\tilde{\mathbf{C}}$ ). The number of transforms  $N$  is determined by the angular resolution. The optimal center is then found through the minimisation:  $\tilde{\mathbf{C}} = \operatorname{argmin}(\sum_{i < N} \|(\mathbf{M}_i - \mathbf{I})\mathbf{C}\|)$ . Applying the amoeba minimisation technique (see Section 3.3.5), the global minimum is quickly found because the initial HJC is already a good guess and because the function to minimise is convex (see Figure 5.38).

In this section, we have presented a complete method to automatically compute the HJC and its movement, assuming a constant inter-articular distance during circumduction motion pattern. In Section 5.8.3, we will validate it.

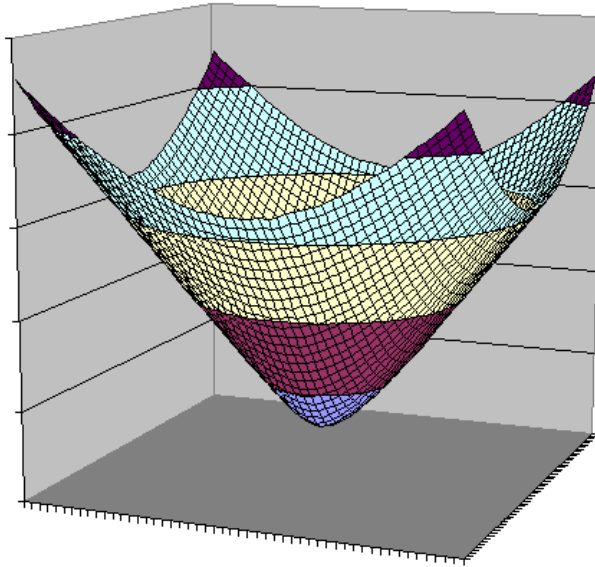


Figure 5.38: Joint center displacement function in X-Y directions within a 5mm distance. Other directions give similar results.

## 5.8 Bone motion extraction

### 5.8.1 Overview

Bone tracking in sequential or dynamic MRI is an intra-patient registration process parameterised by rigid transformations. As already discussed in Section 3.3.4, two approaches can be applied to find the six optimal parameters:

- The standard technique globally minimises the distance (or maximise the similarity) through slight iterative changes in the transform parameters.
- The pair and smooth approach locally minimises the distance (at vertex positions) and find the global transform that best approximates local displacements through an explicit resolution method (Section 3.3.5).

Here, we propose to test and compare these two approaches. Three poses are involved in the registration process (Figure 5.39): in the reference reconstruction frame, models, distance maps and static MRI are defined. Registration features (the model and images we are registering) are identified in the source (moving) dataset. The transformation  $\mathbf{T}_0$  between the reference and source dataset is known. The goal is to align registration features to the fixed (target) images through the rigid transform  $\mathbf{T}$ , the mapping between the source and target datasets being done with the transformation  $\mathbf{T}\mathbf{T}_0^{-1}$ .

The unknown transformation  $\mathbf{T}$  can be expressed in terms of joint transformations: for instance, the femur transform is expressed by  $\mathbf{TF} = \mathbf{TP.MP.RPF.(MF^H)^{-1}}$ . So instead of optimising directly the transformation  $\mathbf{T}$ , it is possible to optimise  $\mathbf{RPF}$ . By optimising the rotational part only and keeping a null translation, we can constrain the joint center to remain still. This is particularly useful when images do not provide sufficient information for tracking joint translations accurately (e.g. very low resolution images in dynamic MRI). In this case, the assumption of a fixed joint center is more accurate than tracking the two bones independently. For the standard registration approach,  $\mathbf{RPF}$  parameters are modified. However, for

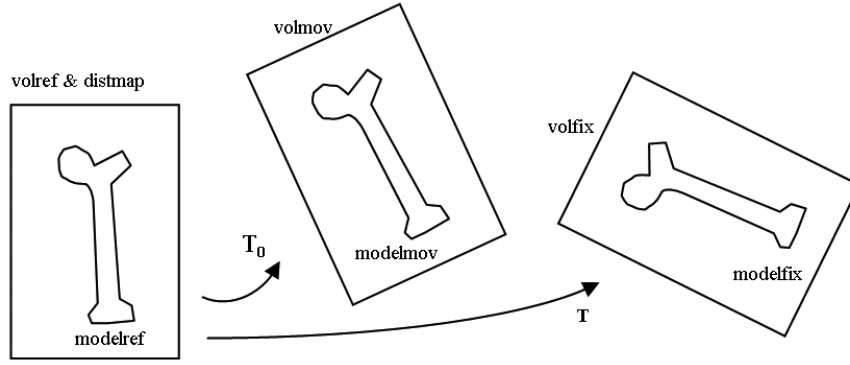


Figure 5.39: From left to right: reference, source (moving) and target (fixed) datasets

the pair and smooth approach, we need an explicit scheme that computes the best *centered* rigid transform, given a displacement field. We extend the method presented in Section 3.3.5 for rigid transformations: by translating all vertices in a coordinate system centered on the joint center  $\mathbf{C}$  and by optimising the rotations only:

$$\begin{aligned}
 \tilde{\mathbf{R}} &= \operatorname{argmin}(\sum_i \|\mathbf{y}'_i - \mathbf{R}\mathbf{x}'_i\|^2) \\
 &= \operatorname{argmin}(\sum_i \|\mathbf{y}_i - \mathbf{C} - \mathbf{R}(\mathbf{x}_i - \mathbf{C})\|^2) \\
 &= \operatorname{argmin}(\sum_i \|\mathbf{y}_i - \mathbf{R}\mathbf{x}_i + (\mathbf{R} - \mathbf{I})\mathbf{C}\|^2)
 \end{aligned}$$

Therefore, for centered rigid transforms, rotations are given in the same way than for rigid transforms. The difference is that we impose the optimal translation to be:  $\tilde{\mathbf{t}} = -(\mathbf{R} - \mathbf{I})\mathbf{C}$  where  $\mathbf{C}$  is the joint center coordinate.

Bones are rigidly registered by aligning a region of interest in the source and target images through similarity measure maximisation. To save computational time, we consider intensity profiles in the neighbourhood of the surface instead of the complete region inside bones (Figure 5.40). Intensity profiles rely on regions and directions of maximal intensity changes to better and faster discriminate similar/ dissimilar registered images (Section 4.7.1). For bones, we have optimised intensity profile sizes to  $s = 0.5mm$ ,  $N_i = 25$  and  $N_o = 5$  (Section 5.5.2). Due to the strong regularisation (parameterisation with rigid transforms), we can extend the search space  $l$  to  $5cm$  (compared to bone elastic registration) without increasing sensitivity to noise.

For the standard approach, the rigid transform parameters to modify are based on the quaternion formulation [Sho85]. Indeed, anatomical angles are not suited because they are *dependent* (this is even worse with Euler angles). The conversion of a rigid transform into a translation and a quaternion leads to a translation vector  $\mathbf{t} = [t_x, t_y, t_z]^T$  and a unit quaternion  $\mathbf{q} = [\cos \theta/2, \sin \alpha \cos \beta \sin \theta/2, \sin \alpha \sin \beta \sin \theta/2, \cos \alpha \sin \theta/2]$ . The quaternion is equivalent to a rotation of  $\theta$  around an axis  $\mathbf{u}$  (see Figure 5.41). We define  $\mathbf{u}$  through the spherical coordinates  $\alpha$  and  $\beta$ , the norm of  $\mathbf{u}$  ( $\sin \theta/2$ ) being given by the normalisation of the quaternion to unit length. So, the six *independent* parameters (degrees of freedom) for the standard registration approach are  $t_x, t_y, t_z, \alpha, \beta$  and  $\theta$ . For the evolution (maximisation of the similarity through parameter modification), we use the VTK implementation of the amoeba resolution method (downhill simplex method)<sup>1</sup>. The

<sup>1</sup><http://www.vtk.org/doc/release/5.0/html/a01152.html>

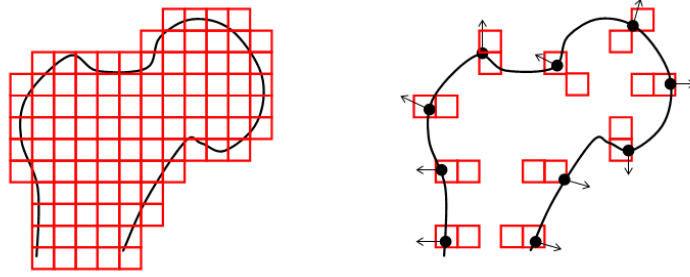


Figure 5.40: Template vs. intensity profile-based matching approaches

parameters scales are set to  $5mm$  and  $0.1rad$  ( $\sim 6deg$ ) and the number of iterations to 50. The registration is run until converge. These parameters have proven to cover the range of displacements typically found in our application.

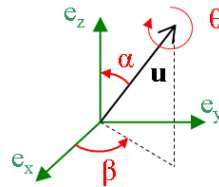


Figure 5.41: A rotation expressed with quaternions leads to the three *independent* parameters  $\alpha$ ,  $\beta$  and  $\theta$

To compare the standard and the pair and smooth approach, we use the dataset where bones have been reconstructed and we apply a certain rigid transform to the bones. Therefore, we know the transformation  $\mathbf{T}_{\text{ref}} = \mathbf{I}$  to recover through registration (gold standard). Thus, the registration error is given by the matrix  $\mathbf{T}$  (see Figure 5.39). Decomposing  $\mathbf{T}$  into a translation and a unit quaternion leads to two error values: the norm of the translation vector, and the amplitude of the rotation  $\theta$ . Here, we show three different representative scenarios where the femur is registered from three different initial conditions: the femur is rotated with regards to the pelvis; the femur is translated with regards to the pelvis; the femur is rotated and translated with regards to the pelvis. Error evolution is given in Figure 5.42. In these tests (and more generally in all tests we have performed), the pair and smooth method was more accurate, faster and more convergent than the standard approach. This is because, the global energy is not convex (lots of local minima) contrary to the set of local displacements. Also, even if performing the registration of a volume with itself (the hypothesis of intensity conservation is valid), we note that the normalised cross-correlation measure ( $NCC$ ) performs better than the absolute differences ( $AD$ ). The typical final registration errors with the selected technique (pair and smooth and  $NCC$  measure) is  $0.5mm$  for translations and  $0.05deg$  for rotations. Reducing the intensity profiles step size from  $0.5mm$  to  $0.2mm$  or  $0.1mm$  improves the translation error until about  $0.2mm$ . Here, we reach the sub-pixel precision realisable with the image resolution we use ( $0.78 \times 0.78 \times 2mm$ ). This kind of accuracy is obtained in the simple case of self registration. In the general case, it is reasonable to expect a error of about  $0.5mm/0.05deg$ . The increase of model resolution does not improve the accuracy significantly, however the convergence is improved at the cost of computational time (this acts as a smoothing of the energy).

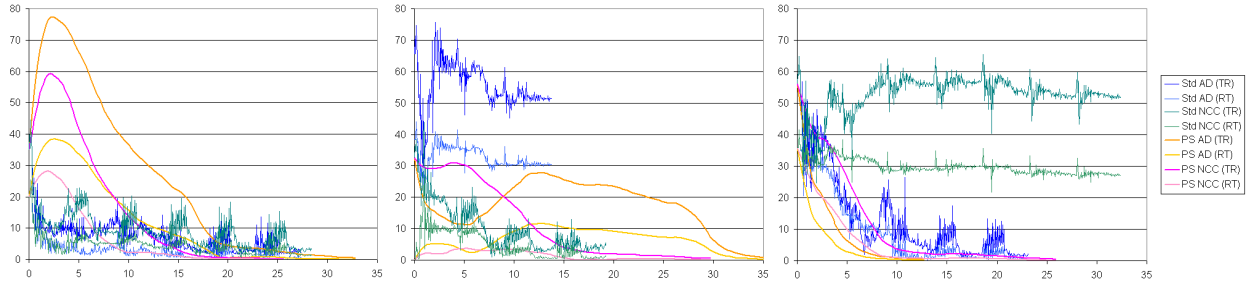


Figure 5.42: Matching error (translations in  $mm$  and rotations in  $deg$ ) in function of computation time (in seconds) for different registration methods and different scenarios (from left to right)

### 5.8.2 Tracking in sequential MRI

The static and sequential MRI protocols we are using (Sections 5.2.1 and 5.2.2) present various contrasts and resolutions. Therefore, the similarity measure is not always reliable. For instance, bones appear in black with the truefisp or gradient echo sequences, which is different than in the reference spin echo sequence where models have been reconstructed (black: cortical bone, white: trabecular bone). Then, the gradient-based similarity measures ( $G$  measure) seems better suited. From a registration based on the gradient (different protocols), reliable registrations can be subsequently performed based on intensity profile similarity ( $NCC$  measure) as shown in Figure 5.43. When performing sequential acquisition, the first volume is taken in a neutral position (very close to the static acquisition posture), therefore we expect no movement of the hip joint center. To reduce translational errors due to the registration based on the gradient, we impose a pure joint rotation between the reference and the first sequential positions (this is done by tracking the femur first, which is more reliable, and then the pelvis with centered rigid transforms). Then we track bones independently in the remaining sequential volumes, using the pair and smooth approach and  $NCC$  measure. The Video 5.8 illustrates a typical tracking process in sequential MRI.

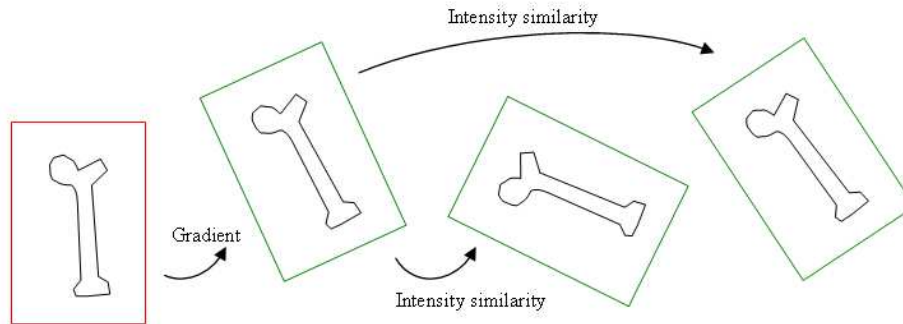


Figure 5.43: Registration strategy when using different protocols

### 5.8.3 HJC validation

From bone motion, tracked in sequential volumes, we want to analyse the error we would make if we assume a fixed hip joint center. This is important in order to compare the different methods for computing the joint center (either predictive or functional). Knowing the joint center (and eventually how it moves) would allow an accurate estimate of joint motion from angular inputs (e.g. motion capture data) or low-resolution images (e.g. real-time dynamic MRI). Also, we would like to assess if there is any correlation between large joint

translations and the pathological state of the articulation. A good hip joint center estimate should minimise the translation of the joint transforms **RPF** obtained in the different MRI sequences. Because the tracked translations are in the order of the expected tracking accuracy ( $\sim 0.5mm$ ), the validity of our results is based on their statistical significance. Indeed, we perform our study on a relatively large amount of volunteers (17) and postures ( $\sim 80$ ). We distinguish results from the two sequential protocols: the first one deals with low resolution images and low amplitude motion realised in the lying position (Figure 5.4); and the second one with high resolution images and extreme postures (split) (Figure 5.3). We expect a higher accuracy with the second study since the same MRI protocols are used in the reference (neutral) and extreme positions (the tracking is based on intensity similarity only), and since image resolution is better (i.e. truefisp acquisition  $0.52 \times 0.52 \times 0.6mm$ ).

We are comparing the hip joint centers obtained from the following methods:

- **HJC from acetabulum sphere:** Center of the sphere fitted to acetabular points (see Section 5.7.1)
- **HJC from femoral head sphere:** Center of the sphere fitted to femoral head points (see Section 5.7.1)
- **HJC from double sphere:** Center of the spheres fitted to acetabular and femoral head points (see Section 5.7.1)
- **HJC from constant distance:** Less moving point during circumduction motion (elevation of  $20deg$ ) assuming a constant inter-articular distance corresponding to the radius difference of the two fitted spheres (see Section 5.7.3)
- **HJC from reference distance:** Less moving point during circumduction motion (elevation of  $20deg$ ) assuming a reference inter-articular distance (see Section 5.7.3)

To compare the movement using these methods with the measured one, we check the amplitude of the relative displacement between the femur and the pelvis (that should be zero). We also compare two methods, presented in Section 5.7.2, where the hip joint center is allowed to move. We can assume either a constant inter-articular distance (*constant distance* method) or a reference inter-articular distance (*reference distance* method). To compare the movement with the measured one, we compute the amplitude of the translation imposed by the inter-articular distance adjustment. Our first intention was to calculate the point that moves the less in the different sequences and compare it to the estimated hip joint centers. Unfortunately, the too little number of postures and the high "in-plane" nature of the movements (mainly abduction) do not allow a precise and unique estimation of the measured hip joint center. Tables 5.13 and 5.14 summarise the displacement obtained with low amplitude and high amplitude postures.

	Min	Mean $\pm$ stdev	Max
HJC from acetabulum sphere	0.30	$0.75 \pm 0.50$	2.20
HJC from femoral head sphere	0.22	$0.51 \pm 0.27$	1.02
HJC from double sphere	0.25	$0.43 \pm 0.24$	0.95
HJC from constant distance	0.48	$0.92 \pm 0.47$	1.77
HJC from reference distance	0.24	$0.55 \pm 0.25$	1.08
Constant distance	1.18	$1.86 \pm 0.65$	3.01
Reference distance	0.24	$0.56 \pm 0.25$	1.11

Table 5.13: Errors in terms of joint translation obtained from different femoro-acetabular movement prediction methods (low amplitude postures)

For low amplitude movements (angles  $< 30deg$ ), we can see that the best methods provide translational error in the order of the magnitude of the expected tracking error ( $\sim 0.5mm$ ). The assumption of a fixed point with pure rotations around it is valid. We see that methods are more or less equivalent (in the region of estimated hip joint centers, there is no much movement), except for the *constant distance* method. This is due to deviations in the measured inter-articular distance in the reference position (close to the different other postures) that produces a systematic translation error.

	Min	Mean $\pm$ stdev	Max
HJC from acetabulum sphere	1.86	$3.77 \pm 1.11$	5.78
HJC from femoral head sphere	1.54	$2.65 \pm 0.70$	3.70
HJC from double sphere	2.02	$2.98 \pm 0.83$	4.65
HJC from constant distance	0.90	$3.61 \pm 2.26$	7.87
HJC from reference distance	1.61	$2.32 \pm 0.59$	3.34
Constant distance	2.43	$3.22 \pm 0.52$	4.13
Reference distance	1.49	$2.18 \pm 0.36$	2.57

Table 5.14: Errors in terms of joint translation obtained from different femoro-acetabular movement prediction methods (extreme postures)

For extreme flexions, (flexion =  $135 \pm 11deg$ , abduction =  $31 \pm 11deg$ , internal rotation =  $23 \pm 14deg$ ), measured femoro-acetabular translations are significant with regards to the tracking accuracy. The displacement is at least of  $1.5mm$  and is repeated very systematically across the different volunteers (low standard deviations). We see that a functional method that enforce inter-articular distances to be close than reference ones (neutral posture) is valuable compared to other predictive techniques based on a static analysis (error reduction of about  $0.5mm$ ).

#### 5.8.4 Tracking in dynamic MRI

Real-time dynamic MRI datasets are composed of several non-orthogonal slices (Section 5.2.3). To map source and moving datasets, slice relative positions (which remain fixed during acquisition) need to be taken into account. As shown in Figure 5.44, the rigid registration is performed between a moving dataset (volume) and a coordinate system that is attached to the dynamic images. From this coordinate system, dynamic slices are located using the fixed rigid transform matrices  $\mathbf{O}_i$ ,  $i < Nb$  of *dynamic slices*. The mapping between the source and target datasets is done through the transformation  $\mathbf{O}_i \mathbf{T} \mathbf{T}_0^{-1}$ . To compute intensity profiles in dynamic images, we need to map 3D spatial positions into 2D coordinates, through the matrices  $\mathbf{O}_i \mathbf{T}$ . The  $z$  coordinate corresponds the distance to the dynamic planes. Hence, to decide if we are lying or not in a plane, we check if  $|z| < 2mm$ . Model normals (used in the pair and smooth approach) have to be transformed through the rotational part of  $\mathbf{O}_i$  and projected to the planes. When using gradient vector-based measures, we register gradient vectors in the same system (so we need to transform them also).

#### Parameter optimisation and validation of the tracking

A study (Sections 5.2.2 and 5.2.3) was run with six healthy volunteers to optimise and evaluate the robustness of the registration-MRI protocol combination without the introduction of motion artifacts. The slice positions of the dynamic slices are required to be adjusted to intersect appropriate bony landmarks on each volunteer. These planes are set initially and maintained throughout the sequential motion protocol. To validate the tracking, a sequential acquisition is performed at stepped positions, where the two scans are run (stationary



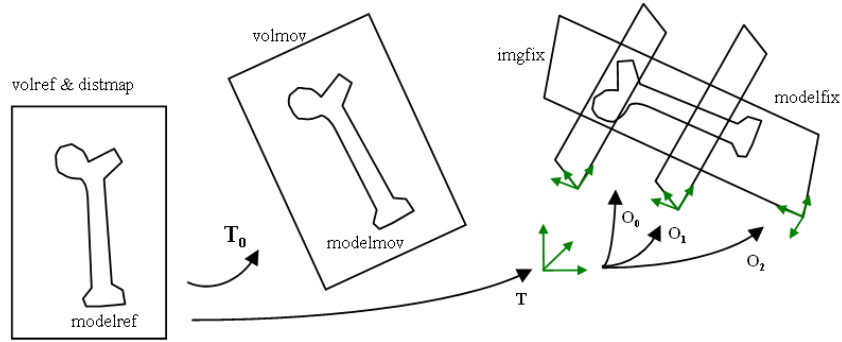


Figure 5.44: From left to right: reference, source (moving) and target (fixed) datasets

joint during the scan). 3D sequential acquisition gives a gold standard of bone positioning as it provides high spatial resolution.

We use seven imaging planes with the configuration shown in Figure 5.45. Because image contrast look similar to the static MRI where bones have been reconstructed, we measure the similarity between the two ( $\mathbf{T}_0 = \mathbf{I}$ ). The same initialisation and same optimisation method (pair-and-smooth approach,  $s = 0.5mm$ ,  $N_i = 25$ ,  $N_o = 5$  and  $l = 10$ ) are used for each similarity measure. Table 5.15 shows that the normalised cross-correlation measure based on gradient vector images performs the best tracking in terms of accuracy. Errors corresponds to the differences in terms of joint transform (norm of the shift and quaternion angle) between the two acquisitions. Also *GNCC* is found to be the most robust metric: Figure 5.46 shows the robustness test of the femur relative position tracking for a sample position and a sample volunteer.

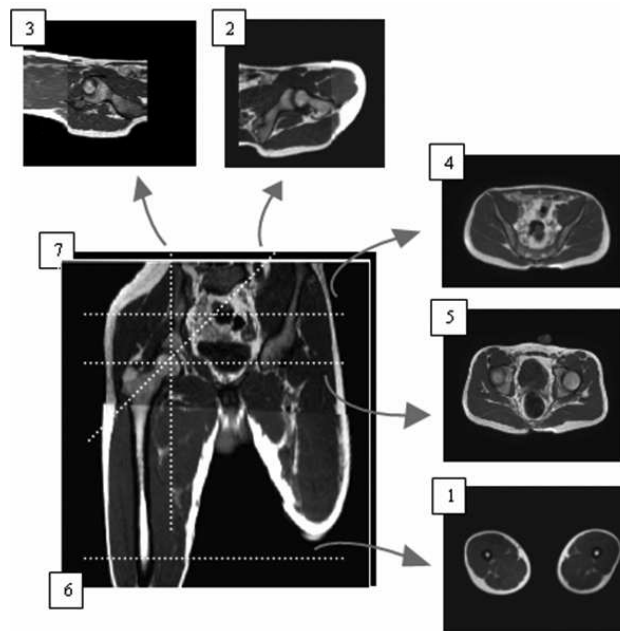


Figure 5.45: Dynamic imaging plane configuration

Measures	Mean error in translation $\pm$ Std. dev. (in mm)	Mean error in rotation $\pm$ Std. dev. (in deg)
GNCC	1.8 (1)	1.3 (0.7)
MI	2.3 (1.3)	3.5 (3.8)
GAD	3.0 (1.2)	5.3 (1.4)
NG	5.6 (5.3)	3.1 (2.9)
NCC	4.5 (2.1)	6.1 (3.7)
AD	9.9 (11.6)	8.4 (6.5)
G	4.4 (3.1)	11.9 (6.8)

Table 5.15: Comparison of the different measures for tracking bones in dynamic MRI

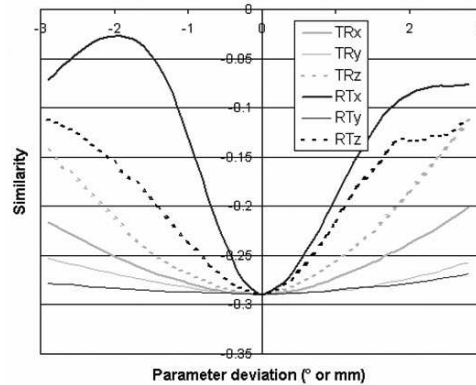


Figure 5.46: Similarity variation around solution. An example of a robust similarity metric: GNCC

Out of all planes that have been acquired, three best planes have to be selected in order to achieve real-time dynamic acquisition. We have tested different combination of three planes for the initial seven planes shown in Figure 5.45. Tests have been achieved with the GNCC metric and the same tracking parameters. Table 5.16 shows that the planes 2, 5 and 6 give the most accurate tracking. In order to increase the acquisition frame rate, dynamic plane resolution can be decreased until tracking errors become too large ( $< 3deg$  in rotation). With GNCC metric and planes 2, 5 and 6, we have tested three different resolutions as shown in Table 5.17 A resolution of  $4 \times 4mm$  was found to be the limit.

Planes	Mean error in translation $\pm$ Std. dev. (in mm)	Mean error in rotation $\pm$ Std. dev. (in deg)
2, 5, 6	2.4 (1)	2.1 (1.1)
2, 5, 7	2.2 (1)	2.7 (1.5)
2, 4, 6	3.1 (1.6)	2.6 (1.1)
3, 4, 6	2.8 (1.2)	4.1 (2.6)
3, 5, 6	2.9 (1.5)	4.4 (3.5)

Table 5.16: Comparison of plane configurations

### Application to real-time dynamic MRI

We have applied our method on real-time dynamic sequences (with motion artifacts) and obtained visually satisfactory results. For a particular frame  $t$ , we initialise bones from preceding positions. The pelvis remains nearly immobile during movement implying that the user initialisation for the first frame is suitable for the

Resolution	Mean error in translation $\pm$ Std. dev. (in mm)	Mean error in rotation $\pm$ Std. dev. (in deg)
2 x 2 mm	2.4 (1)	2.1 (1.1)
4 x 4 mm	3.3 (1.7)	3.3 (1.5)
8 x 8 mm	5.7 (1.9)	5.1 (3.7)

Table 5.17: Comparison of plane resolutions

others. To initialize the femur, we make the assumption that the movement is planar and uniform by using the spherical quaternion interpolation (so-called Slerp [Sho85]) between frames  $t - 1$  and  $t - 2$  with an interpolation parameter equals to 2. The dynamic protocol was a fast gradient echo sequence with balanced gradients. This protocol provides sufficient morphological data for bone tracking to be carried out. In case of a free abductive motion, with no positioning device, it was difficult to constrain the femur to remain in the coronal plane 6. Also we attained a satisfactory sequence acquisition time near real-time (less than 1s) with four imaging planes. Thus we used the combination of planes 2, 5, 6 and 7. The final dynamic MRI protocol used attained a frame rate of 6.7frames/s. Figure 5.47 and Video 5.9 show tracking results.

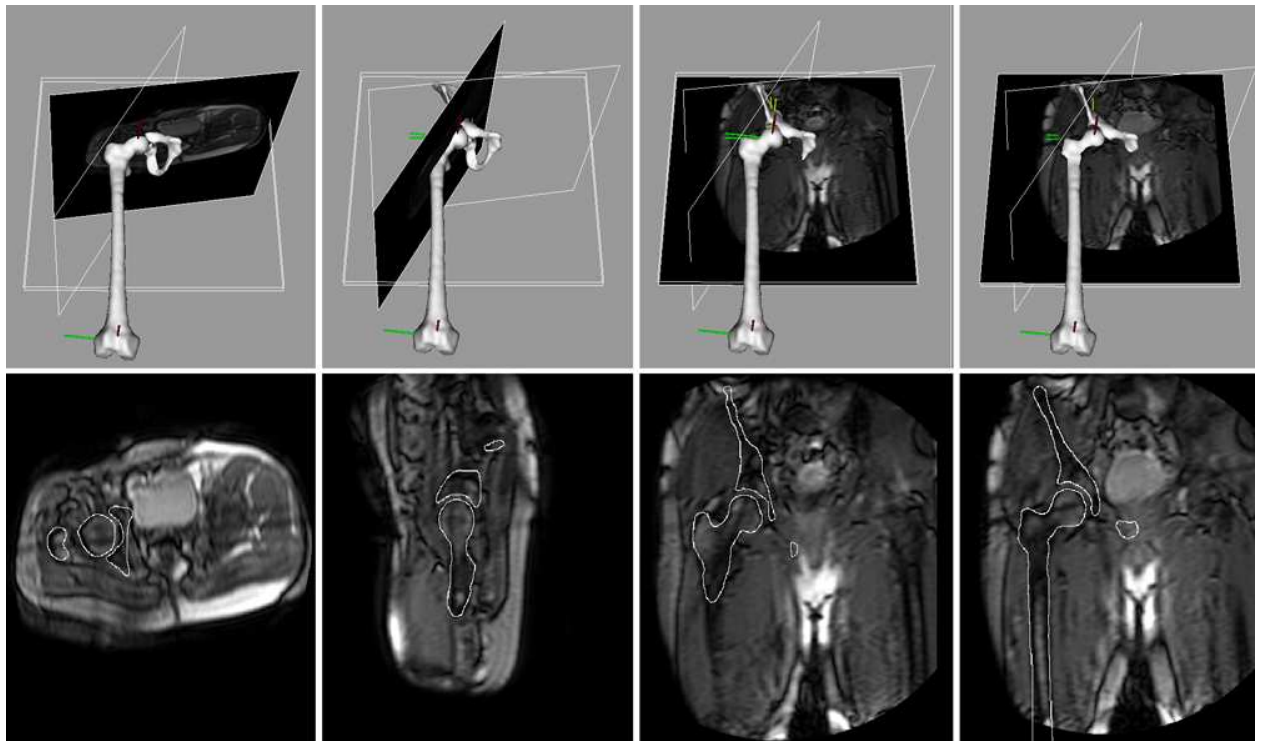


Figure 5.47: Bone tracking results in real-time dynamic MRI, at a sample frame

## 5.9 Skinning

From joint angles, it is possible to estimate the deformation of internal soft-tissues using a skinning algorithm (see Section 2.9). It is valuable for both intra-patient and inter-patient registration to provide a good initialisation of the models before image-based matching. Moreover, it is interesting to measure how accurate can be a skeleton-driven deformation algorithm through comparisons with real data. We use a method based on sweep surfaces such as in [KMTM<sup>+</sup>98] [HYC<sup>+</sup>05], to exploit the cylindrical shape of the leg. For clarity, we illustrate the method on one articulation, but it is used on the three articulations of the leg. To be able to wrap models from one individual to another, we use a reference neutral posture with a reference length for each segment (see Figure 5.48). In the deformed state, bone segments are resized and rigidly transformed. A reference point  $\mathbf{C} = [0, 0, z]^T$  (associated to the segment 1 for instance) is transformed to  $\mathbf{T}_1 \cdot \mathbf{M}_1 \cdot [0, 0, z \cdot L_1 / L_1^{ref}]^T$  (where  $\mathbf{T}_1$  is the bone transform and  $\mathbf{M}_1$  the bone coordinate system).

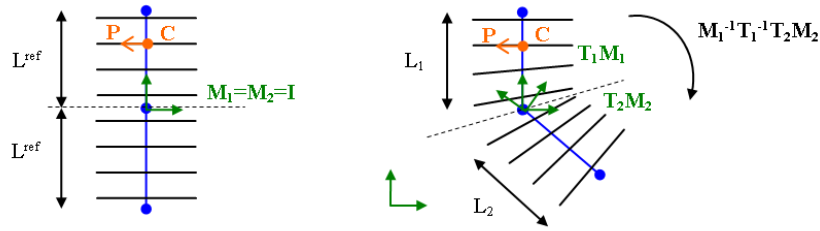


Figure 5.48: Sweep surfaces for one articulation in the neutral (left) and deformed posture (right)

Planes are oriented by weighting the two transforms. We use the spherical quaternion (slerp) technique that interpolates rotations accurately: the vector  $\mathbf{CP} = [x, y, 0]^T$  is transformed to  $Qslerp(t, \mathbf{T}_1 \cdot \mathbf{M}_1, \mathbf{T}_2 \cdot \mathbf{M}_2) \cdot [x, y, 0]^T$  where  $t$  ( $0 \leq t \leq 1$ ) is the weighting parameter.  $t$  depends on the distance  $z$  of the point  $\mathbf{C}$  to the joint center in the reference configuration. This is done through the bijective mapping function  $f : z \in \mathbb{R} \rightarrow t \in [0, 1]$ . Each point should be influenced by two segments only, therefore the boundary conditions for the function that maps  $z$  to  $t$  are:  $f(L^{ref}/2) = 0$  and  $f(-L^{ref}/2) = 1$ . We choose an arctangent function to be able to tune the smoothness of the function (from rigid transform to linear blending) thanks to a parameter  $\lambda$ :

$$t = f(z) = \frac{1}{2} \left( 1 - \frac{\text{atan}(\lambda \cdot z / L^{ref})}{\text{atan}(\lambda / 2)} \right)$$

Figure 5.49 shows the blending curves for different  $\lambda$ . In practice,  $\lambda = 10$  has shown to be a good choice for inter and intra-patient registration (small angles). Figure 5.50 presents two representative results for intra and inter-patient registration based on our skinning algorithm. In addition, we have made a sample animation (Video 5.10) to show the common use of skinning for geometric-based body animation between user defined postures.

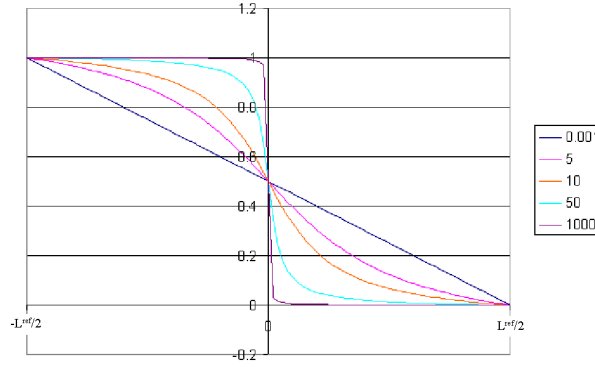


Figure 5.49: The skinning interpolation weight in function of the distance  $z$  to the joint center. The different curves are obtained from different  $\lambda$

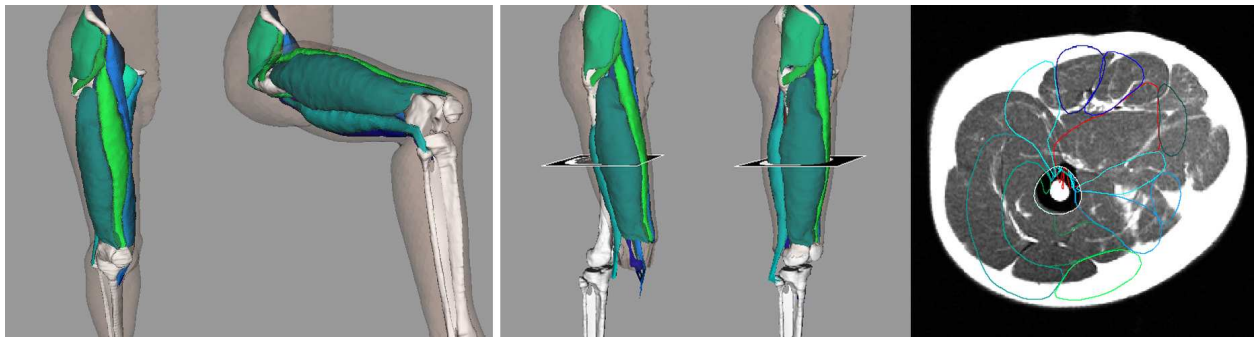


Figure 5.50: Left: Example of intra-patient skinning (a: original posture, b: simulated posture). Right: example of inter-patient skinning (a: before; b: after; c: resulting initialisation on a sample slice)

## 5.10 Extraction of soft-tissue deformations

Such as bone motion tracking, the extraction of tissue deformations is an intra-subject image registration problem. Our goal is to provide a high resolution estimation from low resolution information. From the inter-subject case, new constraints can be added to the system:

- Because individual models have been already segmented in the static images, the reference models used for the matching are now subject-specific (see Figure 5.1). Deformations are computed from the individual neutral positions.
- Reference proximities are also taken from the individual neutral positions
- Organs are nearly incompressible, so we increase the contribution of volume conservation forces, the reference volumes being the ones in the neutral position.

Sequential and dynamic MRI images are low resolution making external forces less reliable. The extra internal constraints represent new prior knowledge that is precious for keeping an acceptable accuracy. Indeed, sequential MRI is difficult to segment manually, especially regarding the delineation of muscles. Even if this is not possible to segment them in the classically way (on a slice-by-slice basis), we believe that they contain sufficient information for model adaptation. So, a well constrained method is necessary

to decrease model flexibility and noise-sensitivity. For some specific postures, we expect that using shape/ proximities from the neutral position would not be adequate, in this case generic models can be taken from another individual in such posture. We need to assess which solution would be the best.

### Extrinsic landmark tracking

In order to assess skin/ bone relative movement we have tracked optical skin markers in sequential MRI (Figure 5.4 and 5.51). We manually place them in the first sequence. In the remaining ones, they are automatically initialised through our skinning algorithm. Then, their positions are refined through template registration such as in [FHM00] (Figure 3.6). Results are reported in [YCGMMT04] [YCMT06].

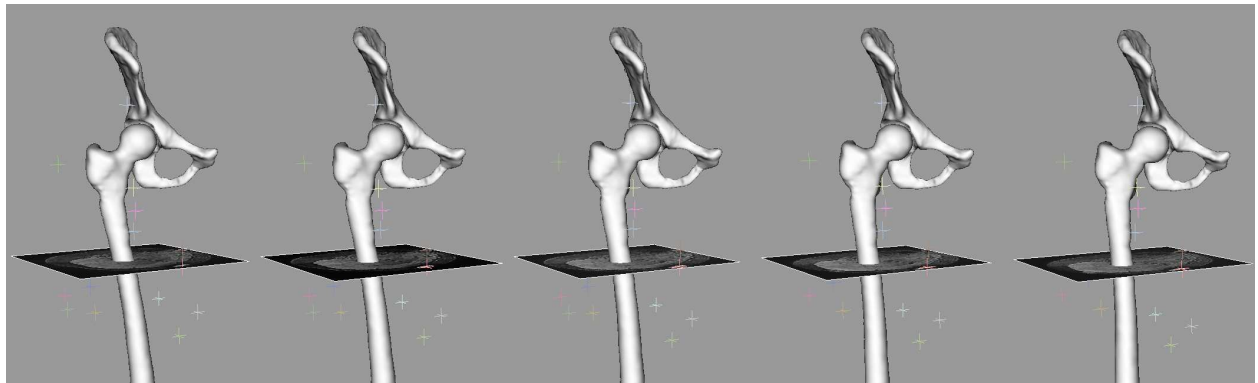


Figure 5.51: Landmark tracking in sequential MRI, during an abduction motion pattern

### Skin tracking

For a small amplitude postures ( $< 20deg$ ), we apply our skinning algorithm and our skin matching method (Section 5.5.3). Even if the skin/ background contrast is low with this MRI protocol (Figure 5.4), the matching remains accurate (we have however to crop the volume to get rid of the shutter effect present near the field of view limits). As expected, the error from skin initialisation (skinning algorithm) is low:  $1.62 \pm 2.33mm$  is this example. Figure 5.52 shows the two skins, modelled from joint angles only (red) and images (blue).

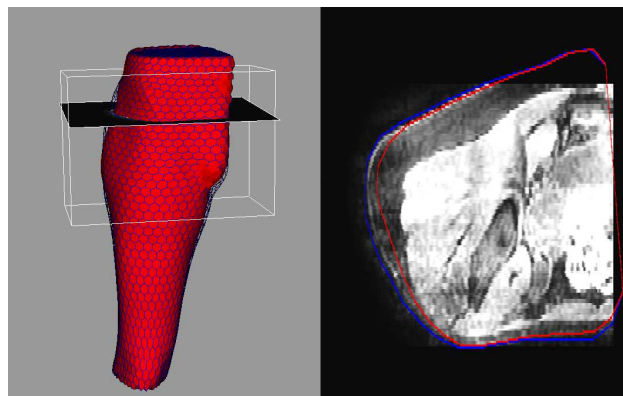


Figure 5.52: Skin tracking from sequential MRI (small amplitude posture): initialisation (red) and fitted surface (blue)

For large amplitude postures (split), we use an interactively segmented reference model, since the direct skin segmentation is not acceptable. From it we undertake the initial shape (for inter-patient skinning initialisation), shape parameters and proximities (with bones). The automatic segmentation using this reference model is satisfactory (see red model in Figure 5.53), despite some errors inside the body and in the crease zone. We have compared the intra-patient skinning (from the neutral position) and inter-patient skinning (both in extreme position). The latest leads to better results ( $7.09 \pm 7.05mm$  vs.  $7.24 \pm 8.75mm$ ), but we see that the error to be recovered using images is still large.

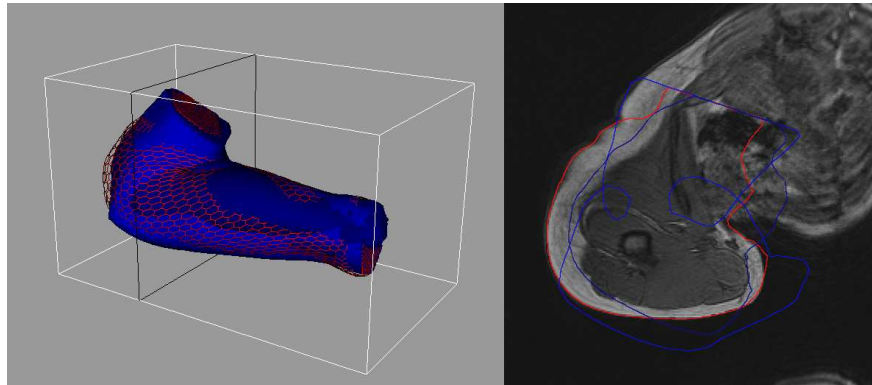


Figure 5.53: Skin tracking from sequential MRI (extreme posture): fitted surface (red), intra-patient initialisation (blue) and inter-patient initialisation (dark blue)

### Cartilage and ligament tracking

For cartilages and ligaments, we apply exactly the same technique than in Section 5.5.3, since it works fine even for extreme poses. Once again, the matching relies on bone relative position and internal forces, rather than image information. Figures 5.54, 5.55 and 5.56 and Video 5.11 illustrate the matching results for the split position.

By superimposing models and MRI, We generally found a good agreement between the two (Figure 5.54). However, since manual cartilage/ ligament delineation is difficult, our method cannot be quantitatively validated without more adequate images or a mechanical model of the tissues. The next step will be the comparison with the deformations obtained from FEM. From geodesic distances on medial axis, it is possible to compute the compression/ elongation of the organs from the neutral position. This is done by calculating the distances on the surface to the closest attachment points, and summing them for the two attachments. As illustrated in Figure 5.57, in the reference neutral position, a point  $\mathbf{P}$  is associated to the closest points  $\mathbf{P1}$  and  $\mathbf{P2}$  and the reference "fiber length"  $d1 + d2$  is stored. In the deformed position, the compression/ elongation percentage is then calculated by  $100 \times (d1' + d2' - d1 - d2)/(d1 + d2)$ .

To enhance particular parameters, we perform colour mapping on the models and medial axis (parameters are mapped from red to blue). It is interesting to analyse high labrum and cartilage deformations as it might be pathological. In Figure 5.55, cartilage compression due to a lower inter-articular distance is shown in green. On the right image, the blue colour (thickness increase of  $1mm$ ) demonstrates a high compression of the labrum in the region of the femoro-acetabular conflict. During split, the ligament elongation is up to 30% (iliofemoral ligament).

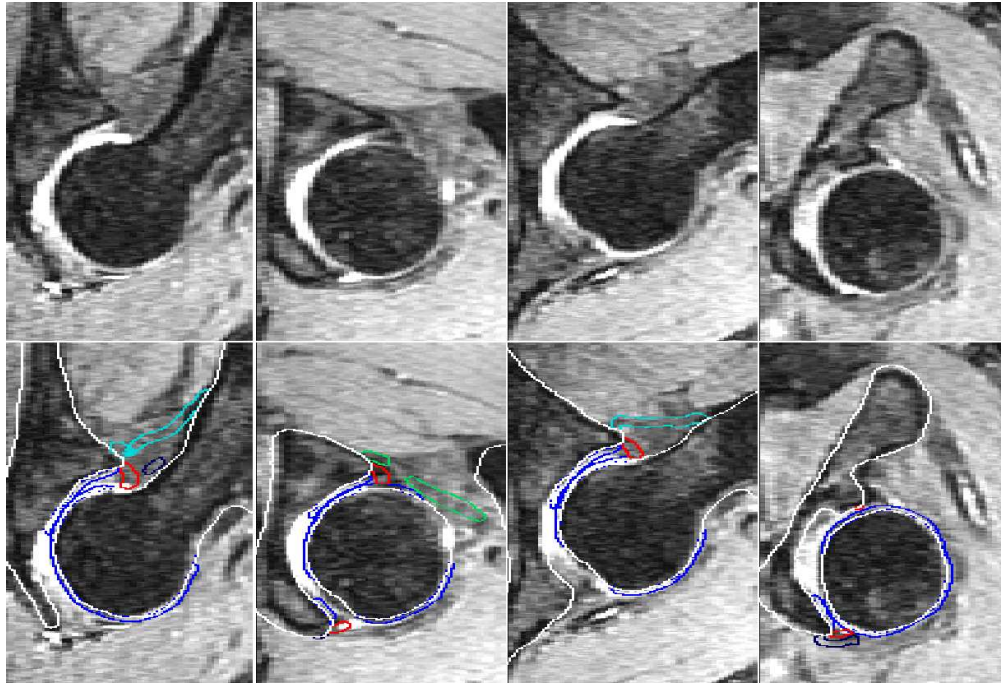


Figure 5.54: Cartilage/ ligament automatic segmentation on sample MRI slices around the hip (split posture)

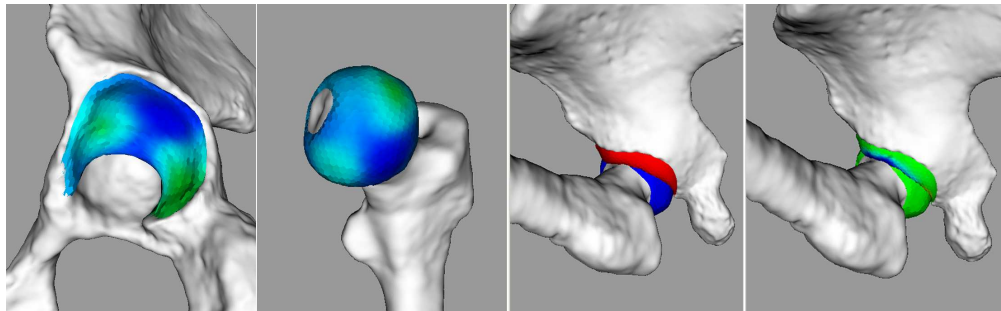


Figure 5.55: From left to right: acetabular cartilage (colours show local thickness) , femoral cartilage (colours show local thickness), labrum (in red), labrum (colours show radius variation from the neutral pose)

### Muscle tracking

The first trial consists in segmenting muscles during low amplitude abductive movement. Such as for the skin, very little correction is to be performed from the intra-patient skinning initialisation (Figure 5.58, top right). This time, we use image forces from gradient vectors only ( $G$  measure), since images have very different contrasts than the reference ones. We add also volume preserving and shape forces from the individualised muscles in the neutral position. The adaptation gives a displacement of  $2 \pm 1.9mm$ . The matching seems correct on images (even if it is almost impossible to validate). The elongation percentage from the neutral posture gives very interesting results: gluteal and adductor muscles are the ones that deform the most (up to 10% in length change) as shown in Figure 5.58, middle. It is in agreement with functional anatomy knowledge (these muscles are responsible of adduction/ abduction as written in Table 5.8).

As before, we now asses deformations in split position. And, once again we perform inter-patient registration



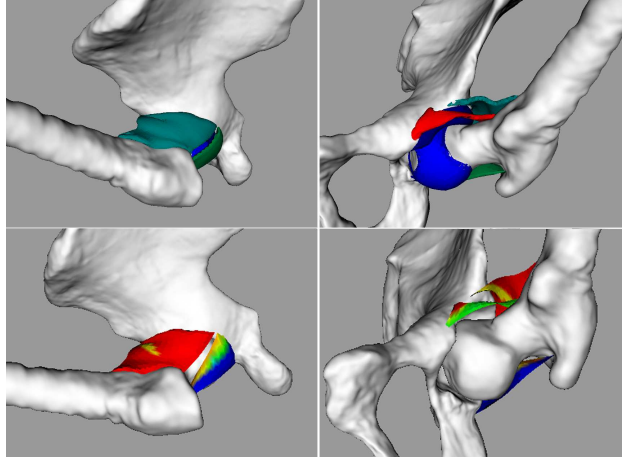


Figure 5.56: Top figures: ligament models; Bottom: ligament medial axis where colours represent the elongation percentage (clamped at  $-10\%$  (red) and  $+10\%$  (blue))

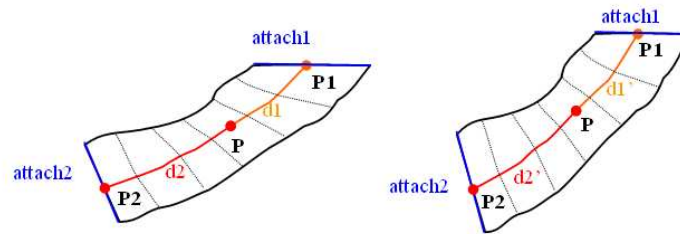


Figure 5.57: Elongation computation at  $P$  between a reference state (left) and a deformed state (right)

rather than intra-patient registration because more relevant prior information is conveyed by the extreme posture from another individual than by the individual models from the neutral position.

The analysis of the elongation gives also interesting results since anterior and posterior muscles and the ilio-psoas are highlighted (they are responsible for flexion/ extension). The elongation is in the range of  $\pm 30\%$  (Figure 5.59, left). The change in muscle thickness, on the contrary does not give much information (right figure)

## 5.11 Conclusion

The generic methods of Chapter 4 have been adapted to the musculoskeletal domain and we have developed the necessary application-specific aspects [GMMT06]:

- musculoskeletal acquisition has constraints related to contrasts, spatial/ temporal resolution, acquisition time and signal-to-noise ratio; that need to be taken into account.
- medical knowledge (e.g. attachments, volume conservation) has to be incorporated to add constraints to the system.
- the various topologies revealed by muscles bones ligaments cartilages skin need to be represented and covered by our techniques. The particular variety in organ shapes demands high flexibility.

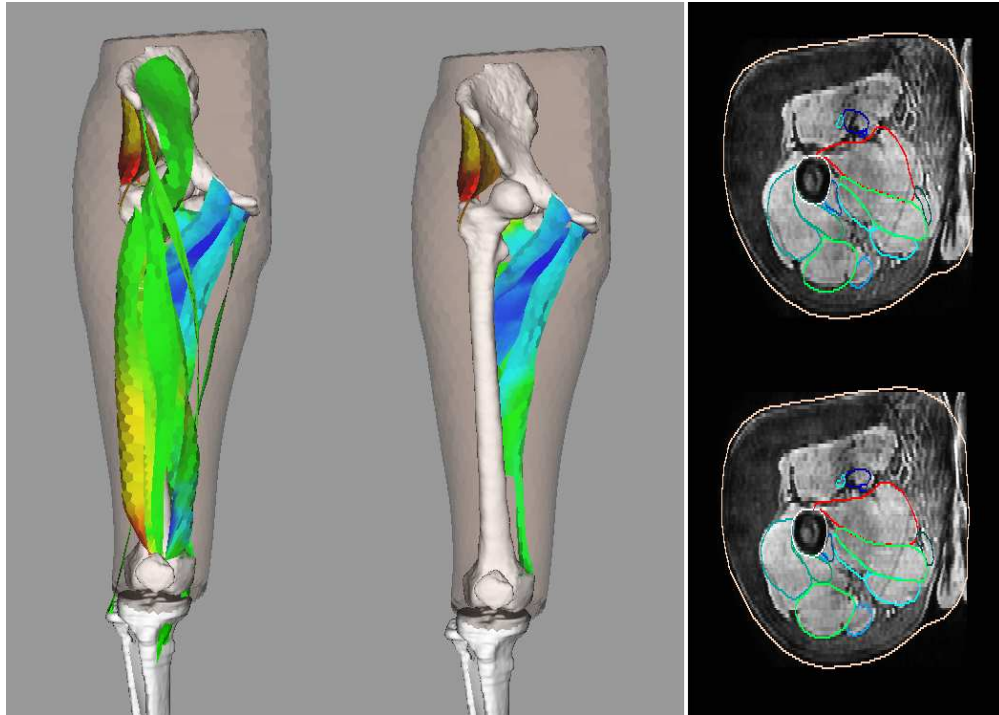


Figure 5.58: Segmented muscle during low amplitude abductive movement. Left: elongation percentage (clamped at  $-10\%$  (red) and  $+10\%$  (blue)). Right: surface adaptation from initialisation (top)

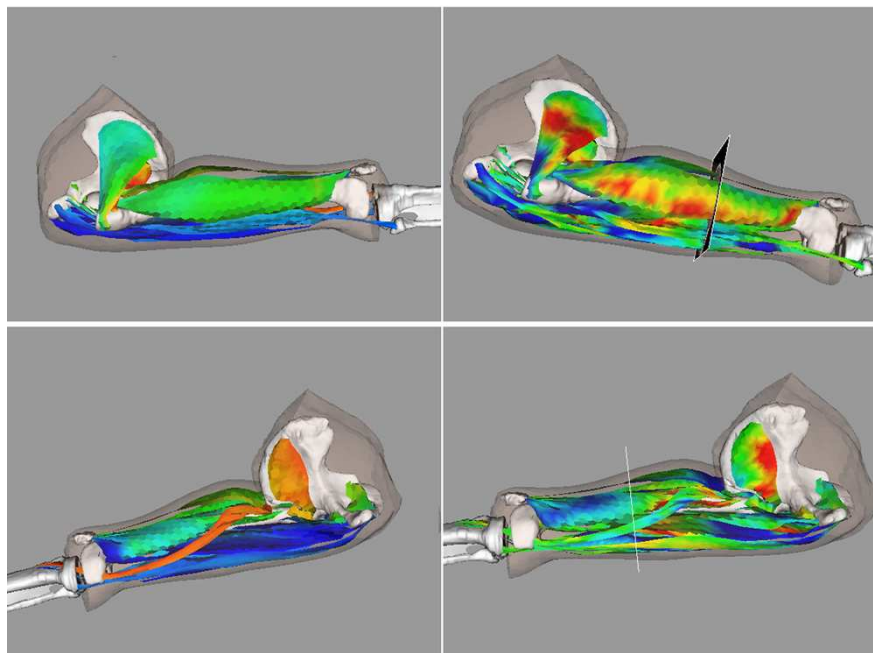


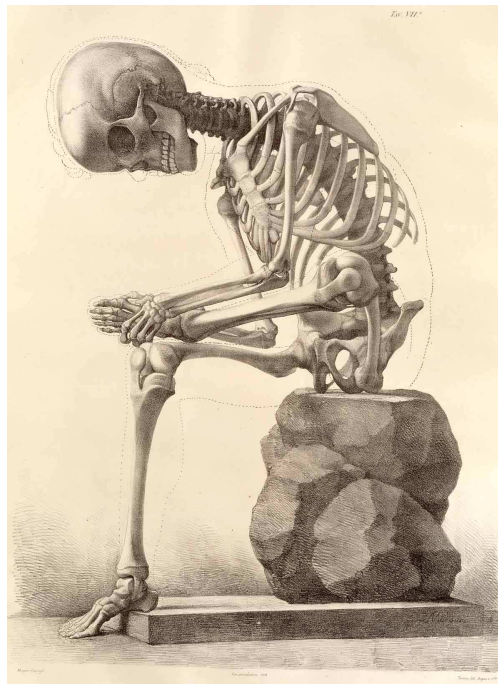
Figure 5.59: Segmented muscle from the split position. Left: elongation percentage (clamped at  $-30\%$  (red) and  $+30\%$  (blue)). Right: local thickness changes

- manual work is required for model building, validation and testing.
- all parameters have to be tuned and optimised to improve robustness
- user-specific techniques and interfaces has to be designed and code has to be optimised for performance improvement.

In the medical field especially, because of its obvious societal impact, an important phase is the validation. The implementation in the clinical environment is long, and supplement the research and development period. That is why it is important to train clinicians to control new tools, so that they can support large scale studies and data collection. Then, specific clinical goals can be answered through statistical evidences.

## Chapter 6

## Conclusion



---

<sup>1</sup>Francesco Bertinatti and Mecco Leone. *Elementi di anatomia fisiologica applicata alle belle arti figurative*. Turin, 1837-39, Lithograph, National Library of Medicine. [http://www.nlm.nih.gov/exhibition/dreamanatomy/da\\_g\\_III-A-12.html](http://www.nlm.nih.gov/exhibition/dreamanatomy/da_g_III-A-12.html)

## 6.1 Discussion

Our work represents one the first attempt to partly model the musculoskeletal system from medical images, in a time-efficient way [GPMTV04] [GMMT06] (most existing methods are interactive, do not perform registration, and/ or make use of non-clinical images such as the visible human dataset). Indeed, besides the well established segmentation techniques (CT-based bone/ lung segmentation, MRI-based brain/ heart/ liver segmentation) musculoskeletal segmentation and registration has received a minor attention in the community, principally because of its complexity but also of its lower impact (musculoskeletal disabilities are not life-threatening even if the first cause of long-term pain). The potential impacts and benefits are however substantial. On the other side, musculoskeletal biomechanical simulation has been a active research topic for many years, with side collaboration in the neurology (muscular control), robotic (functional prosthesis) and sport/ safety (ergonomics) domains. It is however using over-simplified models and despite its success, the musculoskeletal system is not yet comprehensively understood, especially regarding the cause of degenerative diseases, the volumetric organ behaviour and the outcome of inter-subject variability.

Nowadays, medical images offer the ability to study the four dimensional behaviour of the musculoskeletal system. Additional complementary modalities such as motion capture and EMG can exhibit large scale movements and muscle actuations coordination/ patterns. The fragmentation of research areas has lead to partial models. The main challenge is to be able to fuse the different multimodal data through complexity-compliant models. Within a discrete deformable model framework, we have tried to provide scalable geometric methods (mesh topology adaptation/ medial representation / multi-resolution scheme) for controlling shapes and deformations according to different contexts (inter/ intra patient, iconic/ geometric, global/ local registration). We have shown that simple (computationally cheap) constraints are valuable in the framework of the musculoskeletal system, since its behaviour is mainly driven by simple geometric rules such as volume preservation, curvature/ thickness smoothness and non-penetration. We have checked how far some purely geometric methods could predict musculoskeletal behaviour. Now, to allow a full and continuous scalability from modelling to simulation, physically based methods should complete our methods. Surfacic models need to be converted into volumetric models. Anisotropy is an important aspect that has to be taken into account, even if fiber direction is difficult to measure.

## 6.2 Benefits

We have demonstrated the efficiency of the proposed techniques over previous methods, in the following aspects:

- **Robustness:** our method is able to handle relatively large deformations and inter-patient variability, as well as incomplete and under-sampled images such as dynamic MR images.
- **Automation:** from little inputs from the user (radiologist): about 15 landmarks placed in 1 or 2 minutes, our method is able to achieve accurate results. Real-time interaction is possible to assist the automatic process.
- **Accuracy:** through an adequate validation we have shown that the accuracy of our method is millimetre-based. Combining shape and radial constraints, surface sliding errors (boundary registration) are reduced. We have analysed the accuracy towards computer graphics methods (skeleton-driven deformation, model relaxation from internal forces), and interactive/ manual segmentation methods.
- **Complexity:** thanks to our multi-resolution scheme, a typical segmentation of about 20 muscles, 4 bones, 2 cartilages, 4 ligaments and a skin, takes about 30min with an accuracy of 1.5mm. Medial axis representation allows to represent soft-tissues with about 15 times less parameters compared to traditional surface representation with an error of 0.5mm.
- **Flexibility:** the proposed methods are quite generic in terms of types of articulations and organs.

## 6.3 Limitations and future work

Despite the advances attributable to the presented methods, they present a certain number of limitations from the methodology point of view. The experience that we have acquired during this project let us to highlight some critical aspects and problems that should receive a particular attention in future research. More work is also required in the testing and application areas. We are now reviewing weaknesses of our techniques and are proposing new directions per research area:

### 6.3.1 Image processing

While we have focussed on geometric and numerical issues, image processing aspects could receive a greater attention. We believe that this would be the larger source of benefits since external forces are the first cause of error and larger computation burden. In addition, too low resolution images such as dynamic MRI were difficult to handle.

- To improve computational speed, multi-resolution intensity profiles could be introduced. Our tests in this domain (combination of external forces at different scales) were not convincing, but another way is to use multi-resolution images that would preserve most image features (for instance using wavelet transform).
- For global transformations (e.g. for bone tracking and first elastic registration steps), we could introduce stochastic methods based on information theory to reduce computational cost for similarity measurement (in other words, we could use a subset of the images to estimate the whole similarity).
- Faster process would be achievable through parallelisation techniques. They are technically applicable since particles and voxels are mainly considered individually.
- To increase accuracy, the mixture on forces from region homogeneity criteria, contours and iconic similarity should be studied. Our medial representation could be the basis to quickly extract internal features (3D textures).
- We noticed some difficulties in weighting external forces (based on gradient and/ or image correlation) because the different regions have various confidences. We would need large scale analysis (many examples) to be able to spatially weight forces in a relevant way.
- More relevant similarity measures for musculoskeletal images could be developed and combined (for instance, the ones based on MRI phase). In order to segment tissue anisotropy (fibers), measures based on tensor similarity need to be extended to use of dtMRI.
- The example-based approach for reference intensity profiles (shape appearance models) could have some benefits for handling more variability in MRI sequences and image contrasts.

### 6.3.2 Geometry

We have noticed a limited use of our methods in extreme cases: when deformation (extreme poses) and morphological variability are too large. The unique prior shape and the weakness of the initialisation (skinning) are obstacles for recovering large differences during registration. To a least extent, limitations are related to the representation itself.

- We would need to add a statistical layer to handle prior shape information and statistical inter-object distances. For this, we believe that a higher level of representation to gather, compare and analyse heterogeneous data (different individuals/ poses and possibly from different acquisition modalities) would be valuable. This could be achieved through ontology support. In a long-term, cross-comparison studies (e.g. genome/ physiome expression) should be targeted.

- A first idea to improve the skinning would be to sweep surfaces around bone centroids instead of anatomical axis. This would reduce distortions near bones. But other skinning techniques (e.g. example-based skinning) could be beneficial.
- At some places without anatomical features, bone vertices could slide and produce errors in attachments location. We would need to develop attachment individualisation techniques based on surface continuity constraints (tendons). New MRI sequences could be used to better highlight these parts.
- In order to model organs more realistically, the representation has to be extended: various pathologies need to be modelled to be able to segment them accurately and to track their evolution. They could be tissue lesions, muscle injuries, cartilage calcification or bone fractures. Regarding the structural aspect of our models, we may incorporate scalable fiber models or anisotropic features through a volumetric representation.

### 6.3.3 Mechanical simulation

We have seen that using geometric/ image-based forces and efficient integration techniques were beneficial for cost-efficient model registration. But, in order to bridge the modelling and the simulation domains, particularly for the parameterisation and the validation of biomechanical models, new types of constraints have to be integrated.

- Models that are progressively driven from geometry to mechanics are not yet available. We need to smoothly incorporate physical-based constraints derived from continuum mechanics within the same framework used to integrate forces and velocities. Anisotropy and muscle actuation are among the most important features to add.
- Organ interactions are currently roughly simulated through springs between surfaces. Contacts could be improved through a particle system simulating the fat, and through a better knowledge of the physical process behind organ friction.
- Research results from computer graphics/ simulation are directly useful in our context, particularly the field of fast collision handling and particle system simulation (using for instance GPU, PPU or parallelisation techniques).
- The relationship between muscle actuation and muscle deformation is not yet fully understood. Tissue mechanical simulation and image segmentation need to be coupled with large-scale studies including motion capture and physiological acquisition (e.g. EMG). External measurements such as the range of motion would allow characterising ligaments/ muscles maximal length (partly attributable to tissue mechanical parameters).

### 6.3.4 Application and testing

In the previous sections related to the research, we can note that statistics is an horizontal aspect. In order to build a relevant database, extensive testing and data collection is mandatory, which could not be exhaustively achieved during this thesis.

- New data should cover: more individuals, more postures, more muscle actuation states (acquire loaded/unloaded joints). The goal is to constitute a comprehensive and representative database.
- New MRI sequences/ protocols/ scanners should be considered to enhance more facets of the musculoskeletal system: for instance PcMRI, MRE, arthro MRI, open MRI or T3 scanners.
- As already said, data from other modalities should be merged in order to relate large scale biomechanical studies and image-based analysis studies.

- Concerning the validation aspect, more cross-comparisons between more accurate (but more specific) studies are needed. It includes established techniques such as bone reconstruction from CT or studies on cadavers/ phantoms, but also more comparison between results from high-resolution and low-resolution MRI.

## 6.4 Long-term vision and technology transfer

Besides the above technical improvements in fundamental and applied research, the perspective is to contribute to the medical field through technology transfer and software engineering. This should go through extensive testing in the clinical context and standardisation procedures. The different potential applications domains include:

- **Diagnosis, post-operative guide and rehabilitation:** our techniques can be applied to specific medical contexts such as the longitudinal analysis of organ reconstruction processes (e.g. bone reconstruction, muscle healing) and pathology detection (e.g. bone necrosis, muscle injuries). More clinical tools are necessary to provide medically established morphological measurements.
- **Pre-operative planning:** based on the reconstructed virtual organs, specific tools replicating surgical procedures can be used. For instance, osteotomy, bone resection or tendon lengthening methods. The ability to simulate virtually the outcome of the procedure through tissue mechanical simulation is very promising.
- **Intra-operative navigation:** besides the need of accurate 3D models, fast methods for geometric registration and prediction of tissue deformation are necessary. Our research supports this purpose.
- **Training:** realistic and validated shapes and motion patterns could improve medical training regarding the anatomy and functions of the musculoskeletal system. We believe that dynamic atlases will supplement traditional two dimensional instruments. For surgery simulation, bridging geometry-based modelling and physically-based simulation will permit to better know the validity of geometric methods and therefore improve simulation speed.
- **Anthropometric/ biomedical studies:** data collection is necessary for studying the anthropometric and genomic causes of inter-subject variability (and possibly pathologies). The more automatic methods we propose for morphology extraction can support it. Moreover, we foresee that the study of the functional aspects of the musculoskeletal and neuro-motor systems will increasingly make use of individualised virtual models.



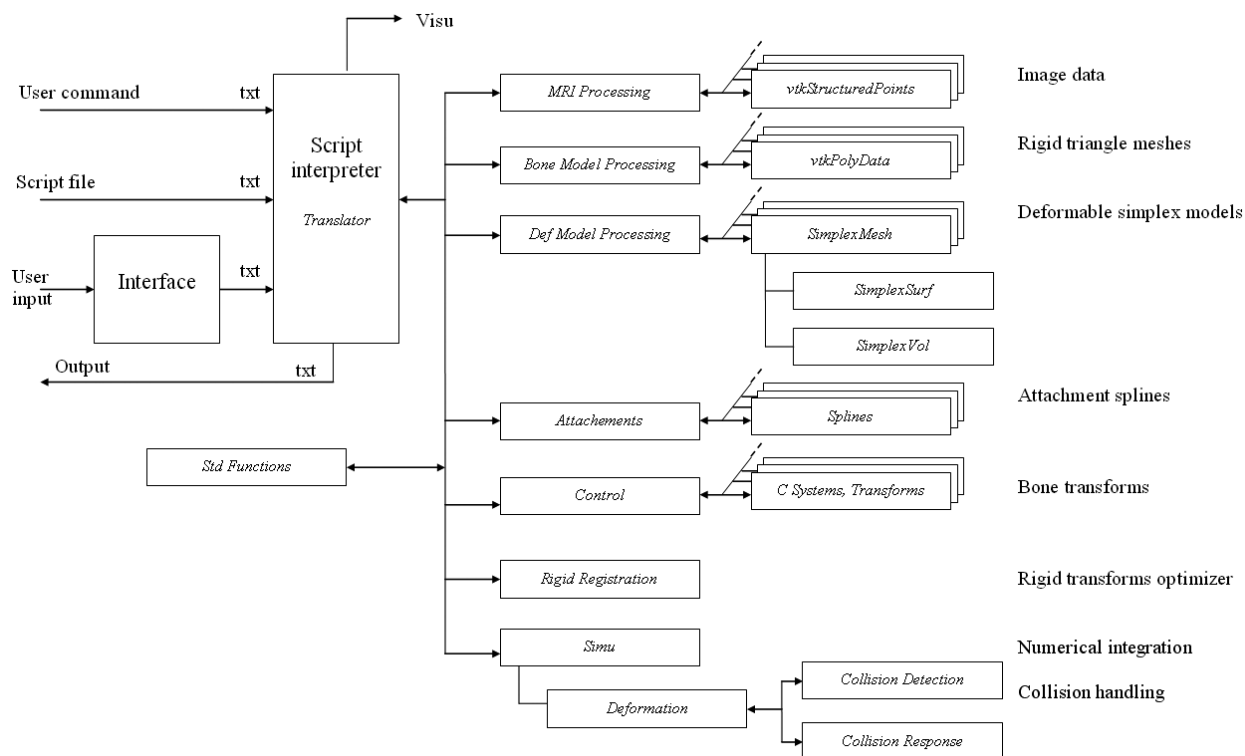
# Annex 1: Software implementation and functionalities

## Introduction

Developments have been performed in conventional C++ using Microsoft Visual Studio .NET, and the open source toolkit VTK (©1993-2006 Ken Martin, Will Schroeder, Bill Lorensen, All rights reserved). All tests presented in this thesis have been performed on a standard PC: Pentium IV 3.4GHz with 1Gb of memory and a Quadro FX 1300 graphics card, under Microsoft Windows XP.

To allow future use of our application, we have chosen to completely separate the interface from the methods. All functions can be accessed via scripts (text commands) to facilitate the parameterization and the automation (template script files with generic parameters can be launched by non-expert users).

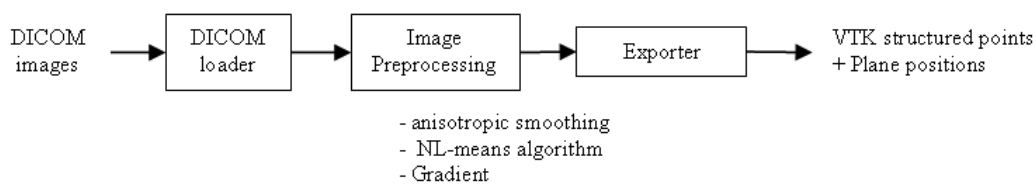
The following diagram shows the basic relationships between the developed C classes (in italic).



## Functionalities

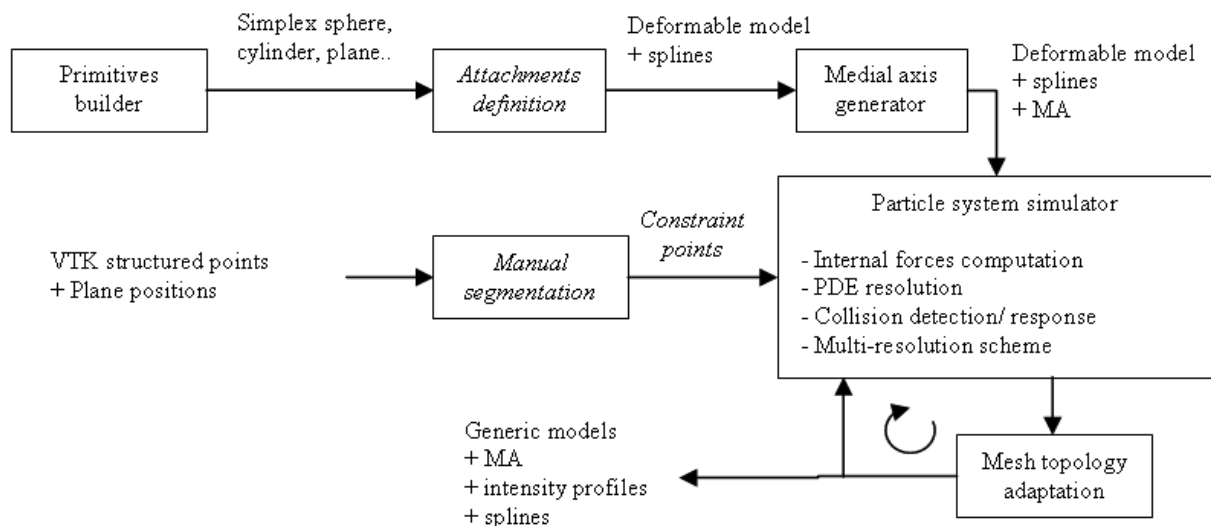
The segmentation process can be viewed as a multi-registration process. An interactively reconstructed generic model is deformed to match patient-specific images. Then, patient models are registered (rigidly for bones, elastically for soft-tissues) to temporal images for the kinematical analysis. As described in the thesis, we make use of discrete surface meshes with constant vertex connectivity (simplex meshes). They are deformed by image-based external forces and geometric-based internal forces. The deformation (new vertex positions and velocities) are obtained by numerically integrating forces (Newton law). Our application can be broken down into several components:

### Medical data acquisition (5.2, 5.3)



We need to import the DICOM files (containing image intensities and headers) and combine them to VTK image files. Each image corresponds to a slice, and all slices are not necessarily parallel. Our loader uses the GDCM vtk loader, that can extract image data, and plane resolutions and orientations. We combine all slices in a VtkStructuredPoints volume (3 dimensional image) and code intensities with unsigned short precision (original precision). If slices are not parallel, we write a file containing slices origins/ orientations (4x4 homogeneous matrices) and resolutions (3d vector). For radial acquisitions, we reorganize slices in the right order to be able to perform cylindric interpolation.

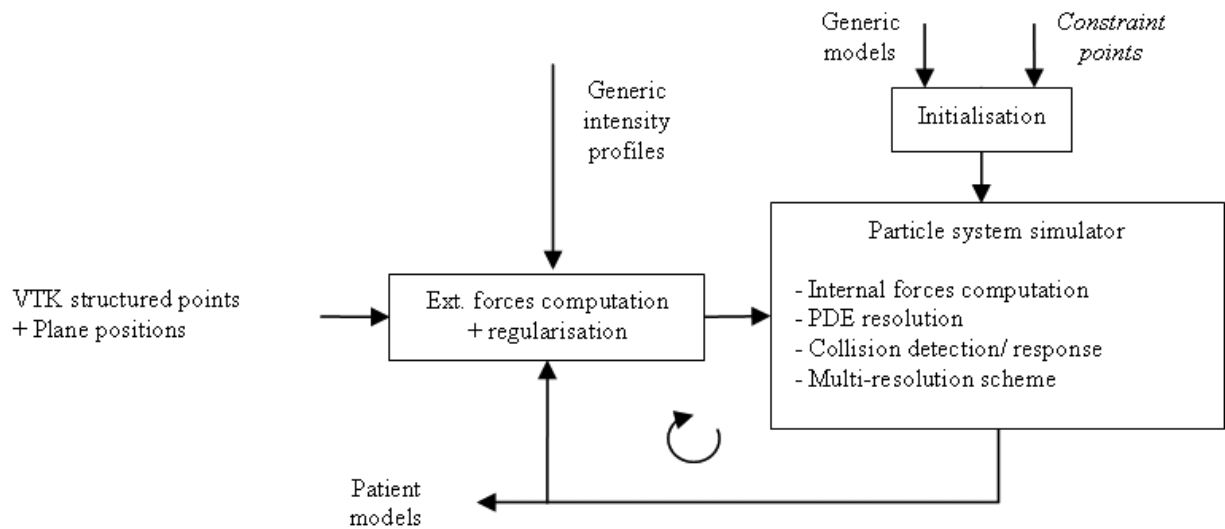
### Generic model reconstruction (5.4)



For this task, manual work is required (in italic) to adjust the primitive to the region of interest through constraint points placement (obtained by clicking onto the MRI slices). Gradient forces with small search

space is generally added for an accurate fitting to image boundaries. Also, attachment splines need to be adjusted on the two sides (bone model and deformable model). During the deformation process, mesh topological operators are executed from time to time, to finally produce the generic mesh (coarsest resolution). The outputs are then generated: higher resolution meshes and intensity profiles.

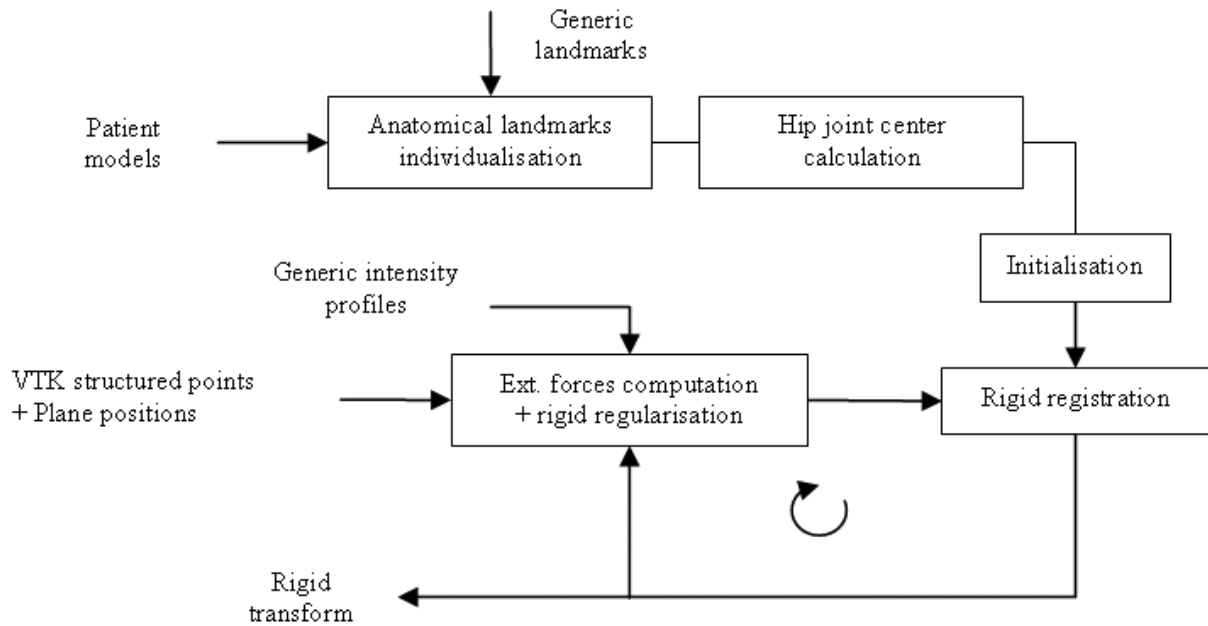
### Model individualisation (5.5)



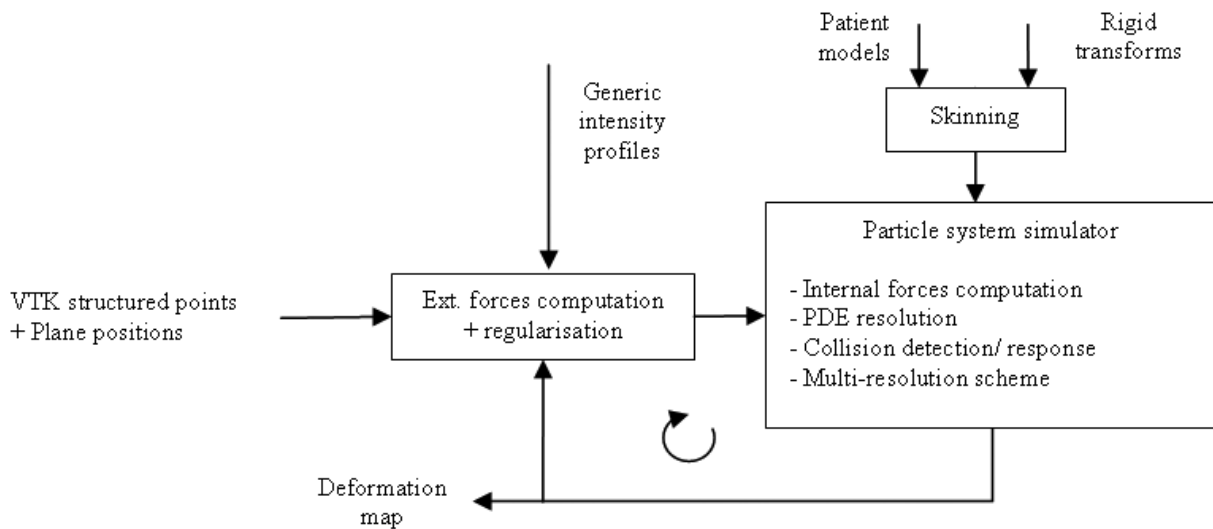
Inter-patient registration is performed by computing the deformation of the generic model that aligns intensity profiles. An initialisation step is necessary to place generic models near the structure to segment. This is done by TPS using manual placed landmarks or alternatively through the inter-patient skinning algorithm.

### Bone motion analysis (5.6, 5.7, 5.8)

Once patient bones have been reconstructed, anatomical landmarks are found automatically from the generic landmarks (through their barycentric coordinates on the surface). The hip joint center is then computed from this. After manual initialisation, rigid registration is performed with the pair and smooth approach (rigid regularisation of intensity profile-based forces). The output is the rigid transform from the initial position.



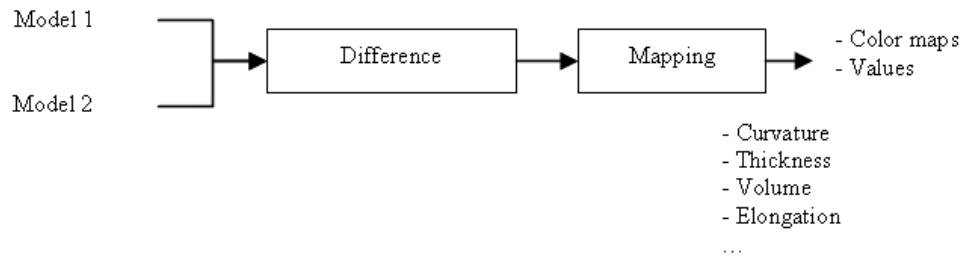
### Soft-tissue deformation analysis (5.9, 5.10)



For intra-patient soft tissue registration, an automatic initialisation is performed using the intra-patient skinning algorithm from registered bones. Generic intensity profiles are obtained from generic models in that position or patient models in a reference position.

### Morphological analysis (5.5.4)

The morphological analysis is based on the comparison between two models (from two patients or two positions). Since geometric features are registered, a simple difference between values (for instance the



curvature) at vertex positions is sufficient. The mapping into colours requires a transfer function that warps the min value to 0 and the max value to 1. Outlier values may be also clamped.

# Annex 2: Clinical background

From the College Faculty of the University of Washington School of Medicine, Seattle, Washington (Courtesy of Gastonia Orthopedics) <sup>1</sup>

## Musculoskeletal Examination Benchmarks

### Inspection:

- Opposite extremities comparison (shape and joint alignment)
- Deformity examination: erythema, swelling/effusions, masses, and/or ecchymoses or other discolorations.
- Muscle bulk inspection

### Palpation:

- Systematic palpation of symptomatic areas and other related areas (joint above and below)
- Warmth assessment of the symptomatic joint (medial and lateral line), muscles and bony prominences
- Comparison of the asymptomatic and symptomatic sides

### Bilateral vascular status assessment

### Sensation assessment to exclude sensory nerve injury

### Radiographs acquisition as clinically indicated

### Motor / functional examination:

- Range of motion (ROM) if no fracture:
  - Active ROM assessment of joints (the ROM the patient can accomplish)
  - Passive ROM assessment of joints if active range of motion is painful or limited (the ROM the clinician can accomplish).
  - Comparison of the asymptomatic and symptomatic sides and normal ROM from manuals
- Strength assessment of the major muscle groups (also part of Neurological Examination)
- Specific provocative/stability testing for the abnormal joint(s) if clinically indicated.

---

<sup>1</sup>[http://www.orthogastonia.com/index.php/fuseaction/patient\\_ed.categories](http://www.orthogastonia.com/index.php/fuseaction/patient_ed.categories)

## Musculoskeletal pains

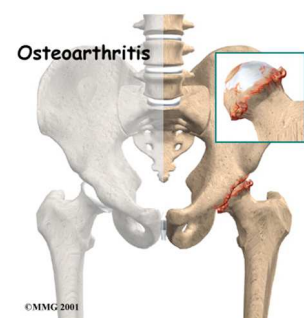
- **Referred pain:** injured or inflamed soft tissues. Pain, unrelated to musculoskeletal system, may refer pain to a musculoskeletal site.
- **Radiating pain:** pattern of pain related to a particular nerve or nerve root irritation (e.g., sciatica)
- **Mechanical problems:** worse with use; improve with rest; stiffness better in AM after bedrest but increases with time, focally tender joints, sometimes swollen joints, but joints are not warm; and there is no joint erythema
- **Inflammatory problems:** are worse with use, persist with rest; stiffness worse in AM after bedrest but better with time and movement as the morning proceeds; diffusely tender and swollen involved joints; joints are usually warm; skin erythema may accompany joint complaints.
- **Suspected fracture:** point tenderness, deformity, step-off, or crepitus
- **Joint swelling:** may be from osteophytes or synovial swelling or effusion
- **Tenderness:** normal joints can be palpated with considerable force without causing pain. Tenderness on joint palpation is abnormal.
- **Warmth:** normal joints are generally cooler than surrounding areas. A warm joint is abnormal.
- **Joint Crepitus:** sensation of grinding with palpation when a joint is moved. This may come from abnormalities of cartilage, bone-on-bone pathology, or abnormalities of tendon function.

## Osteoarthritis (OA) pathology

Osteoarthritis (sometimes referred to as degenerative, or wear-and-tear, arthritis) is a common problem for many people after middle age. OA commonly affects the hip joint. The main problem in OA is degeneration of the articular cartilage. Articular cartilage is the smooth lining that covers the surfaces of the ball-and-socket joint of the hip. The cartilage gives the joint freedom of movement by decreasing friction. When the articular cartilage degenerates, the subchondral bone is uncovered and rubs against bone. Small outgrowths called bone spurs or osteophytes may form in the joint.

**Causes:** OA of the hip can be caused by a hip injury earlier in life. Changes in the movement and alignment of the hip eventually lead to excessive wear and tear on the joint surfaces. The alignment of the hip can be altered from a fracture in the bones around or inside the hip. Cartilage injuries, infection, or bleeding within the joint can also damage the joint surface of the hip. Scientists believe genetics makes some people prone to developing OA in the hip. Problems in the subchondral bone may trigger changes in the articular cartilage: some medical conditions can make the subchondral bone too hard or too soft, changing how the cartilage normally cushions and absorbs shock in the joint. Avascular necrosis (AVN) is another cause of degeneration of the hip joint. In this condition, the femoral head (the ball portion of the hip) loses a portion of its blood supply and actually dies.

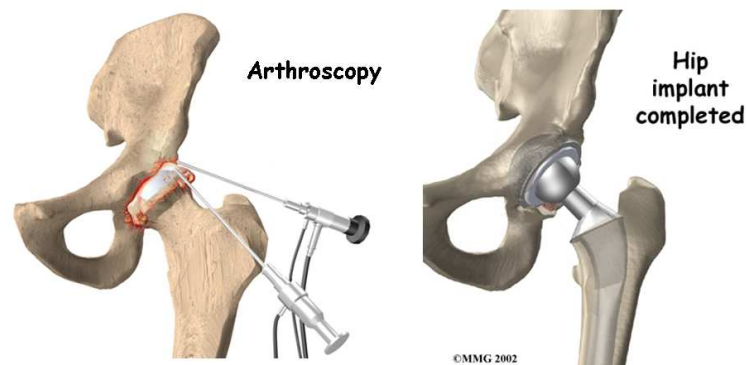
**Symptoms:** the symptoms of hip OA usually begin as pain while putting weight on the affected hip. The affected hip is felt stiff and tight due to a loss in its range of motion. As the condition becomes worse, pain may be present all the time.



**Nonsurgical Treatment:** OA cannot be cured, but medicines are available to ease symptoms and to slow down the degeneration of the joint: mild pain reliever and anti-inflammatory medication. Cortisone injection may be prescribed but may actually speed up the process of degeneration. Physical therapy can help to learn how to control symptoms: rest, heat, topical rubs, range-of-motion, strengthening and stretching exercises. A cane be needed to ease pressure when walking.

### **Surgery:**

- **Arthroscopy:** Surgeons can use an arthroscope (miniature TV camera inserted into the joint through a small incision) to check the condition of the articular cartilage in a joint. At the same time, joint cleaning can be performed by removing loose fragments of cartilage. Another method involves simply flushing the joint with a saline solution, after which some patients report relief. Hip arthroscopy is relatively new, and it is unclear at this time which patients will benefit.
- **Bone resurfacing:** Femoral bone resurfacing can last from 10 to 20 years and is easier on the patient than total hip replacement. Particularly for younger people and persons who have complications from taking steroids, this is a good option. Total hip replacement is often viewed as a last resort and femoral bone resurfacing can delay the need for hip replacement for many years.
- **Osteotomy:** When the alignment of the hip joint is altered from disease or trauma, more pressure than normal is placed on the surfaces of the joint. This extra pressure leads to more pain and faster degeneration of the joint surfaces. Angle realignment (osteotomy) aims at spreading forces over a larger surface in the hip joint. In this procedure, the bone of either the pelvic socket or femur is cut, and the angle of the joint is changed. In some cases, it can result in shifting pressure to the other healthier parts of the hip joint.
- **Artificial Hip Replacement:** this is the ultimate solution for advanced hip OA. Surgeons prefer not to put a new hip joint in patients less than 60 years old. This is because younger patients are generally more active and might put too much stress on the joint, causing it to loosen or even crack. A revision surgery to replace a damaged joint is harder to do, has more possible complications, and is usually less successful than a first-time joint replacement surgery.



**Rehabilitation:** Shortly after surgery, walking is practiced using walker or a pair of crutches Exercises are used to improve muscle tone and strength in the hip and thigh muscles and to help prevent the formation of blood clots. In addition, a therapist helps the patient to maximize hip strength, and restore a normal walking.



# Bibliography

- [ABD01] A.S. Arnold, S.S. Blemker, and S.L. Delp. Evaluation of a deformable musculoskeletal model for estimating muscle-tendon lengths during crouch gait. *Annals of Biomedical Engineering*, 29(3):263–274, 2001.
- [ABE07] D. Attali, J.D. Boissonnat, and H. Edelsbrunner. Stability and computation of medial axes: a state of the art report. In *B. Hamann T. Mller and B. Russell, editors, Mathematical Foundations of Scientific Visualization, Computer Graphics, and Massive Data Exploration. Springer-Verlag, Mathematics and Visualization*, 2007.
- [ABGD02] D.S. Asakawa, S.S. Blemker, G.E. Gold, and S.L. Delp. In vivo motion of the rectus femoris muscle after tendon transfer surgery. *J. of Biomechanics*, 35(8):1029–1037, 2002.
- [ABKC90] N.M. Alpert, J.F. Bradshaw, D. Kennedy, and J.A. Correia. The principal axis transformation a method for image registration. *Journal of nuclear medicine*, 31:1717–1722, 1990.
- [ACK01] N. Amenta, S. Choi, and R.K. Kolluri. The power crust, unions of balls, and the medial axis transform. *Computational Geometry: Theory and Applications*, 19(2–3):127–153, 2001.
- [ACP02] B. Allen, B. Curless, and Z. Popovic. Articulated body deformation from range scan data. *ACM Transactions on Graphics*, 21(3):612–619, 2002.
- [ACSYD05] P. Alliez, D. Cohen-Steiner, M. Yvinec, and M. Desbrun. Variational tetrahedral meshing. *Proc. of SIGGRAPH’05, ACM Transactions on Graphics*, 24(3):617–625, 2005.
- [AD85] M.L. Audu and D.T. Davy. The influence of muscle model complexity in musculoskeletal motion modeling. *J. Biomech. En.*, 107:147–157, 1985.
- [Adl99] R.S. Adler. Future and new developments in musculoskeletal ultrasound. *Radiol Clin North Am*, 37:623–631, 1999.
- [AFP00] M. Audette, F. Ferrie, and T. Peters. An algorithmic overview of surface registration techniques for medical imaging. *Medical Image Analysis*, 4(3):201–217, 2000.
- [AHB87] K.S. Arun, T.S. Huang, and S.D. Blostein. Least-squares fitting of two 3-d point sets. *IEEE Transactions on pattern analysis and machine intelligence*, 5:698–700, 1987.
- [ALR<sup>+</sup>99] C. Monserrat Aranda, M.C. Juan Lizandra, M. Alcaniz Raya, V. Grau Colomer, and C. Knoll. Deformation simulation algorithms of elastic tissues in ”real-time” based in elasticity theory. *Stud Health Technol Inform.*, 62:21–22, 1999.
- [ANB<sup>+</sup>03] D.S. Asakawa, K.S. Nayak, S.S. Blemker, S.L. Delp, J.M. Pauly, D.G. Nishimura, and G.E. Gold. Real-time imaging of skeletal muscle velocity. *J. of Magnetic Resonance Imaging*, 18:734–739, 2003.
- [AP01] F.C. Anderson and M.G. Pandy. Static and dynamic optimization solutions for gait are practically equivalent. *J. of Biomechanics*, 34:153–61, 2001.

- [ASAD00] A.S. Arnold, S. Salinas, D.J. Asakawa, and S.L. Delp. Accuracy of muscle moment arms estimated from mri-based musculoskeletal models of the lower extremity. *Computer Aided Surgery*, 5:108–119, 2000.
- [AT01] A. Aubel and D. Thalmann. Interactive modeling of the human musculature. *Proc. of Computer Animation*, 2001.
- [Aub02] A. Aubel. Anatomically-based human body deformations. *Phd Thesis, EPFL*, 2573, 2002.
- [AWJ90] A.A. Amini, T.E. Weymouth, and R.C. Jain. Using dynamic programming for solving variational problems in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(9):855–867, 1990.
- [Bar92] D. Baraff. Dynamic simulation of non-penetrating rigid bodies. *PhD thesis, Department of Computer Science, Cornell University*, 1992.
- [Bat96] K. Bathe. Finite element procedures. *Prentice Hall*, 1996.
- [BCA96] E. Bardinet, L.D. Cohen, and N. Ayache. Tracking and motion analysis of the left ventricle with deformable superquadrics. *Medical Image Analysis*, 1(2):129–149, 1996.
- [BCA98] E. Bardinet, L.D. Cohen, and N. Ayache. A parametric deformable model to fit unstructured 3d data. *Comp. Vision Graph. Image Understand.*, 7(1):39–54, 1998.
- [BCM05] A. Buades, B. Coll, and J.M. Morel. A non local algorithm for image denoising. *Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR'05)*, 2:60–65, 2005.
- [BD05] S.S. Blemker and S.L. Delp. Three-dimensional representation of complex muscle architectures and geometries. *Annals of Biomedical Engineering*, 33(5):661–673, 2005.
- [BE79] A. Björck and T. Elfving. Accelerated projection methods for computing pseudoinverse solutions of systems of linear equations. *BIT*, 19:145–163, 1979.
- [BFA02] R. Bridson, R. Fedkiw, and J. Anderson. Robust treatment of collisions, contact and friction for cloth animation. *Proc. of SIGGRAPH'02*, pages 594–603, 2002.
- [BFS<sup>+</sup>06] P. Bourgeat, J. Fripp, P. Stanwell, S. Ramadan, and S. Ourselin. Mr image segmentation using phase information and a novel multiscale scheme. *Proc. of MICCAI'06*, 2:920–927, 2006.
- [BH91] L. Blankevoort and R. Huiskes. Ligamentbone interaction in a threedimensional model of the knee. *J Biomech Eng*, 113:263–269, 1991.
- [BHB<sup>+</sup>99] C.F. Beaulieu, D. K. Hodge, A.G. Bergman, K. Butts, B.L. Daniel, C.L. Napper, R.D. Darrow, C.L. Dumoulin, and R.J. Herfkens. Glenohumeral relationships during physiologic shoulder motion and stress testing: initial experience with open mr imaging and active imaging-plane registration. *Radiology*, 212(3):699–705, 1999.
- [BHJ<sup>+</sup>04] T.W. Beck, T.J. Housh, G.O. Johnson, J.P. Weir, J.T. Cramer, J.W. Coburn, and M.M. Malek. Mechanomyographic amplitude and mean power frequency versus torque relationships during isokinetic and isometric muscle actions of the biceps brachii. *J. of Electromyography and Kinesiology*, 14:555–564, 2004.
- [BKM<sup>+</sup>04] C. Baten, M. Kleinhorsman, D. Magermans, W. De Vries, B. Koopman, F. Van Der Helm, and P. Veltink. A system for threedimensional interactive simulation of hand biomechanics. *Proc. of the international symposium on 3D human motion analysis*, 2004.

- [BKNC88] B.H. Brown, T. Karatzas, R. Nakielny, and R.G. Clark. Determination of upper arm muscle and fat areas using electrical impedance measurements. *Clin. Phys. Physiol. Meas.*, 9:47–55, 1988.
- [BLLSW05] C. Bonifasi-Lista, S.P. Lake, M.S. Small, and J.A. Weiss. Viscoelastic properties of the human medial collateral ligament under longitudinal, transverse and shear loading. *J Orthop Res*, 23:67–76, 2005.
- [Blu64] H. Blum. A transformation for extracting new descriptors of shape. In *Symposium on Models for the Perception of Speech and Visual Form*, M.I.T. Press, pages 139–146, 1964.
- [BM92] P.J. Besl and N.D. McKay. A method for registration of 3-d shapes. *IEEE Trans. PAMI*, 14(2):239–256, 1992.
- [BMM<sup>+</sup>88] G. Bilbro, R. Mann, T. Miller, W. Snyder, D. Van den Bout, and M. White. Optimization by mean field annealing. *Avances in Neural Information Processing Systems*, 1:91–98, 1988.
- [BMS<sup>+</sup>93] J. Brossmann, C. Muhle, C. Schroder, U.H. Metchert, C.C. Bull, R.P. Spielmann, and M. Heller. Patellar tracking patterns during active and passive knee extension: evaluation with motion-triggered cine mr imaging. *Radiology*, 187(1):205–212, 1993.
- [BMT97] R. Boulic, R. Mas, and D. Thalmann. Complex character positioning based on a compatible flow model of multiple supports. *IEEE Transactions on Visualization and Computer Graphics*, 3(3):245–261, 1997.
- [BN96] M. Bro-Nielsen. Medical image registration and surgery simulation. *Ph.D. Thesis, Department of Mathematical Modelling, Technical University of Denmark*, 1996.
- [BNC96] M. Bro-Nielsen and S. Cotin. Real-time volumetric deformable models for surgery simulation using finite elements and condensation. *Computer Graphics Forum (Proc. of Eurographics 96)*, pages C57–C66, 1996.
- [Boo89] F.L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6):567–585, 1989.
- [Bor84] G. Borgefors. Distance transformation in arbitrary dimensions. *Computer Vision, Graphics and Image Processing*, 27:321–345, 1984.
- [BPB90] A.L. Bell, D.R. Petersen, and R.A. Brand. A comparison of the accuracy of several hip center location prediction methods. *Journal of Biomechanics*, 23:617–621, 1990.
- [BPS72] W.A.M. Brekelmans, H.W. Poort, and T.J.J.H. Slooff. A new method to analyse the mechanical behaviour of skeletal parts. *Acta. orthop. Scand.*, 43:301–317, 1972.
- [BPTY04] P. Beillas, G. Papaioannou, S. Tashman, and K.H. Yanga. A new method to investigate in vivo knee behavior using a finite element model of the lower limb. *J. of Biomechanics*, 37:1019–1030, 2004.
- [Bro69] C.G. Broyden. A new double-rank minimization algorithm. *Notices of the American Mathematical Society*, 16:670, 1969.
- [Bro92] L.G. Brown. A survey of image registration techniques. *ACM Computing Surveys*, 24(4):325–376, December 1992.
- [BS79] N. Badler and S. Smoliar. Digital representations of human movement. *ACM Computer Survey*, 11, 1979.

- [BSSH04] G. Bianchi, B. Solenthaler, G. Szekely, and M. Harders. Simultaneous topology and stiffness identification for mass-spring models based on fem reference deformations. *Proc. of MICCAI'04*, pages 293–301, 2004.
- [BT87] W.L. Buford and D. E. Thompson. A system for threedimensional interactive simulation of hand biomechanics. *IEEE Trans. Biomed. Eng.*, 34:444–453, 1987.
- [BW90] J. Bloomenthal and B. Wyvill. Interactive techniques for implicit modeling. *Proc. SIGGRAPH Symposium on Interactive 3D Graphics, Computer Graphics*, 24(2):109–116, 1990.
- [BW98] D. Baraff and A. Witkin. Large steps in cloth simulation. *Proc. of SIGGRAPH98, Computer Graphics*, 32:106–117, 1998.
- [Cac02] P. Cachier. Recalage non rigide d’images médicales volumiques. contributions aux approches iconiques et géométriques. *Phd Thesis, ENS Cachan, France*, 2002.
- [Cap84] A. Cappozzo. Gait analysis methodology. *Human Movement Science*, 3:27–54, 1984.
- [CB81] R.D. Crowninshield and R.A. Brand. A physiologically based criterion of muscle force prediction in locomotion. *J. of Biomechanics*, 14:793–801, 1981.
- [CC93] L. Cohen and I. Cohen. Finite element methods for active contour models and balloons for 2-d and 3-d images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1131–1147, 1993.
- [CDA99a] S. Cotin, H. Delingette, and N. Ayache. A hybrid elastic model allowing real-time cutting, deformations and force-feedback for surgery training and simulation. *Proc. of Computer Animation 99*, 1999.
- [CDA99b] S. Cotin, H. Delingette, and N. Ayache. Real-time elastic deformations of soft tissues for surgery simulation. *IEEE Transactions on Visualization and Computer Graphics*, 5(1):62–73, 1999.
- [CDM04] B. Cutler, J. Dorsey, and L. McMillan. Simplification and improvement of tetrahedral models for simulation. *Proc. of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*, pages 95–104, 2004.
- [CGC<sup>+</sup>02] S. Capell, S. Green, B. Curless, T. Duchamp, and Z. Popovic. Interactive skeleton-driven dynamic deformations. *Proc. of SIGGRAPH'02*, pages 586–593, 2002.
- [CGD97] M.P. Cani-Gascuel and M. Desbrun. Animation of deformable models using implicit surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 3(1), 1997.
- [Cha03] E.Y.S. Chao. Graphic-based musculoskeletal model for biomechanical analyses and animation. *Medical Engineering and Physics*, 25:201–212, 2003.
- [CHP89] J.E. Chadwick, D.R. Haumann, and R.E. Parent. Layered construction for deformable animated characters. *Proc. of SIGGRAPH'89, Computer Graphics*, 23(3):243–252, 1989.
- [Chr94] G.E. Christensen. Deformable shape models for anatomy. *Washington University Ph.D. Thesis*, 1994.
- [CKP00] M. Chen, T. Kanade, and D. Pomerleau. Bootstrap a statistical brain atlas. *IEEE Workshop on Applications of Computer Vision*, pages 114–119, 2000.
- [CMD<sup>+</sup>95] A. Collignon, F. Maes, D. Delaere, D. Vandermeulen, P. Seutens, and G. Marchal. Automated multimodality image registration using information theory. *in Information Processing in Medical Imaging*, pages 263–274, 1995.

- [CN89] C.A. Carlstedt and M. Nordin. Biomechanics of tendons and ligaments. *in Basic Biomechanics of the Musculoskeletal System, Lea and Febiger*, 1989.
- [Coh91] L. Cohen. On active contour models and balloons. *Computer Vision, Graphics, and Image Processing: Image Understanding*, 53(2):211–218, 1991.
- [Cow01] S.C. Cowin. Bone mechanics handbook, second ed. *Cambridge University Press*, 2001.
- [CP00] P. Cachier and X. Pennec. 3d non-rigid registration by gradient descent on a gaussian weighted similarity measure using convolutions. *Proc. of MMBIA'00*, pages 182–189, 2000.
- [CPCZ03] J. Crouch, S.M. Pizer, E.L. Chaney, and M. Zaider. Medially based meshing with finite element analysis of prostate deformation. *Proc. of MICCAI'03*, pages 108–115, 2003.
- [CRM96] G.E. Christensen, R.D. Rabbitt, and M.I. Miller. Deformable templates using large deformation kinematics. *IEEE Transactions on Image Processing*, 5(10):1435–1447, 1996.
- [CSM<sup>+</sup>02] M. Carotti, F. Salaffi, P. Manganelli, D. Salera, B. Simonetti, and W. Grassi. Power doppler sonography in the assessment of synovial tissue of the knee joint in rheumatoid arthritis: a preliminary experience. *Ann. Rheum. Dis.*, 61:877–882, 2002.
- [CT01] T.F. Cootes and C.J. Taylor. Statistical models of appearance for medical image analysis and computer vision, in medical imaging. *Proc. of SPIE*, 4322:238–248, 2001.
- [Cut93] A. Cutts. Muscle physiology and electromyography. *Mechanics of Human Joints: Physiology, Pathophysiology, and Treatment*, ed. by V. Wright, E.L. Radin, 1993.
- [CYB06] P. Coupé, P. Yger, and C. Barillot. Fast non local means denoising for 3d mr images. *Proc. of MICCAI'06*, 4191:33–40, 2006.
- [CZ92] D. Chen and D. Zeltzer. Pump it up: Computer animation of a biomechanically based model of muscle using the finite element method. *Computer Graphics (Proc. of SIGGRAPH 92)*, pages 89–98, 1992.
- [Dan73] D.A. Danielson. Human skin as an elastic membrane. *J. of Biomechanics*, 6:539–546, 1973.
- [DBN<sup>+</sup>96] A. Van Doorn, P.H. Bovendeerd, K. Nicolay, M.R. Drost, and J.D. Janssen. Determination of muscle fibre orientation using diffusion-weighted mri. *Eur. J. Morphol.*, 34:5–10, 1996.
- [dC76] M.P. do Carmo. Differential geometry of curves and surfaces. *Prentice-Hall, Englewood Cliffs, NJ*, 1976.
- [DDBC99] G. DeBunne, M. Desbrun, A. Barr, and M-P. Cani. Interactive multiresolution animation of deformable models. *Computer Animation and Simulation 99*, 1999.
- [DDCB01] G. DeBunne, M. Desbrun, M-P. Cani, and A. Barr. Dynamic real-time deformations using space and time adaptive sampling. *Computer Graphics (Proc. of SIGGRAPH 01)*, 2001.
- [Del94a] H. Delingette. Modélisation, déformation et reconnaissance d’objets tridimensionnels à l’aide de maillages simplexes. *Phd thesis, Ecole Centrale de Paris*, 1994.
- [Del94b] H. Delingette. Simplex meshes: a general representation for 3d shape reconstruction. *Proc. of CVPR'94*, pages 856–857, 1994.
- [DeL97] C.J. DeLuca. the use of surface electromyography in biomechanics. *J. App. Biomech.*, 13:135–163, 1997.
- [Del99] H. Delingette. General object reconstruction based on simplex meshes. *International Journal of Computer Vision*, 32(2):111–146, 1999.

- [DGB03] F. Dupont, B. Gilles, and A. Baskurt. Progressive transmission of 3d object based on balls and cones union from medial axis transformation. *ICIP'03 IEEE International Conference on Image Processing*, 2003.
- [DGS01] L. Ding, A. Goshtasby, and M. Satter. Volume image registration by template matching. *Image and Vision Computing*, 19(12):821–832, 2001.
- [DH95] M. Dalstra and R. Huiskes. Load transfer across the pelvic bone. *J. of Biomechanics*, 28(6):715–724, 1995.
- [DHW<sup>+</sup>04] K. Denis, T. Huysmans, T. De Wilde, C. Forausberger, W. Rapp, B. Haex, J. Vander Sloten, R. Van Audekercke, G. Van der Perre, K.R. Heitmann, and H. Diers. A 4d-optical measuring system for the dynamic acquisition of anatomical structures. *Proc. of MICCAI'04*, 2:259–266, 2004.
- [DJGMT03] T. DiGiacomo, C. Joslin, S. Garchery, and N. Magnenat-Thalmann. Adaptation of facial and body animation for mpeg-based architectures. *Proc. of IEEE International Conference on CyberWorld (CW'03)*, pages 221–229, 2003.
- [DKA95] R. Deriche, P. Kornprobst, and G. Aubert. Optical flow estimation while preserving its discontinuities: A variational approach. *Proc. of the 2nd Asian Conference on Computer Vision*, pages 71–80, 1995.
- [DKT98] T. DeRose, M. Kass, and T. Truong. Subdivision surfaces in character animation. *Computer Graphics (Proc. of SIGGRAPH 98)*, pages 85–94, 1998.
- [DL95] S.L. Delp and J.P. Loan. A graphics-based software system to develop and analyze models of musculoskeletal structures. *Comput. Biol. Med.*, 25(1):21–34, 1995.
- [DL00] S.L. Delp and J.P. Loan. A computational framework for simulating and analyzing human and animal movement. *IEEE Computing in Science and Engineering*, 2(5):46–55, 2000.
- [DLH<sup>+</sup>90] S.L. Delp, J.P. Loan, M.G. Hoy, F.E. Zajac, E.L. Topp, and J.M. Rosen. An interactive graphics-based model of the lower extremity to study orthopaedic surgical procedures. *IEEE Transactions on Biomedical Engineering*, 37(8):757–767, 1990.
- [DMSB00] M. Desbrun, M. Meyer, P. Schröder, and A. Barr. Discrete differential-geometry operators in nd. <http://www.multires.caltech.edu/pubs/>, 2000.
- [DP94] J.E. Drace and N.J. Pelc. Measurement of skeletal muscle motion in vivo with phase-contrast mr imaging. *J. of Magnetic Resonance Imaging*, 4:157–163, 1994.
- [DP02] K. DeRoy and L. Peeraer. Visualising the pressure distribution on the plantar surface of the foot during stance, walking and running using a plantar pressure measurement system. *the encyclopedia of imaging science and technology, chapter 8*, 2002.
- [DRR<sup>+</sup>01] M. Dresner, G. Rose, P. Rossman, R. Muthupillai, A. Manduca, and R. Ehman. Magnetic resonance elastography of skeletal muscle. *J. Magn. Reson. Imaging*, 13:269–276, 2001.
- [DS92] W. Dahmen and C.A. Micchelli and H.P. Seidel. Blossoming begets b-spline built better by bpatches. *Mathematics of Computation*, 59:97–115, 1992.
- [DSB99] M. Desbrun, P. Schröder, and A. Barr. Interactive animation of structured deformable objects. *Proc. of Graphics Interface*, pages 1–8, 1999.
- [DSC95] S.L. Delp, K. Statler, and N. C. Carroll. Preserving plantar flexion strength after surgical treatment for contracture of the triceps surae: a computer simulation study. *J. Orthop. Res.*, 13:96–104, 1995.

- [DSMK01] C. Davatzikos, D. Shen, A. Mohamed, and S.K. Kyriacou. A framework for predictive modeling of anatomical deformations. *IEEE Trans. Med. Imaging*, 20(8):836–843, 2001.
- [DW03] Q. Du and D. Wang. Tetrahedral mesh generation and optimization based on centroidal voronoi tessellations. *International Journal on Numerical Methods in Engineering*, 56(9):1355–1373, 2003.
- [EEH00] B. Eberhardt, O. Etmuss, and M. Hauth. Implicit-explicit schemes for fast animation with particles systems. *Eurographics workshop on Computer Animation and Simulation*, 2000.
- [EF83] B.R. Epstein and K.R. Foster. Anisotropy in the dielectric properties of skeletal muscle. *Med. Biol. Eng. Comput.*, 21:51–55, 1983.
- [ETK<sup>+</sup>03] M. Endo, T. Tsunoo, S. Kandatsu, S. Tanada, H. Aradate, and Y. Saito. Four-dimensional computed tomography (4d ct) concepts and preliminary development. *Radiation Medicine*, 21(1):17–22, 2003.
- [EW58] N. Eshkol and A. Wachmann. Movement notation. *Weldenfeld and Nicolson*, 1958.
- [FDC01] A. Fenster, D.B. Downey, and H.N. Cardinal. Three-dimensional ultrasound imaging. *Phys. Med. Biol.*, 46:67–99, 2001.
- [Fer00] S.J. Ferguson. Biomechanics of the acetabular labrum. *PhD thesis, Queen’s University, Kingston, Ontario, Canada, Department of Mechanical Engineering*, 2000.
- [FHG<sup>+</sup>99] N.J. Fuller, C.R. Hardingham, M. Graves, N. Sreaton, A.K. Dixon, L.C. Ward, and M. Elia. Predicting composition of leg sections with anthropometry and bioelectrical impedance analysis, using magnetic resonance imaging as reference. *Clinical Science*, 96:647–657, 1999.
- [FHM00] J.M. Fitzpatrick, D.L.G. Hill, and C.R. Maurer. Image registration, chapter 8. *Handbook of Medical Imaging - Volume 2, Medical Image Processing and Analysis*, pages 447–513, 2000.
- [FL93] R.W. Fright and A.D. Linney. Registration of 3-d head surfaces using multiple landmarks. *IEEE Transactions on medical imaging*, 12(3):515–520, 1993.
- [FLMM97] J. Fuller, L.J. Liu, M.C. Murphy, and R.W. Mann. A comparison of lower-extremity skeletal kinematics measured using skin- and pinmounted markers. *Human Mov. Sci.*, 16:219–242, 1997.
- [FMT<sup>+</sup>03] J. Fernandez, F. Mithraratne, S. Thrupp, M. Tawhai, and P. Hunter. Anatomically based geometric modeling of the musculo-skeletal system and other organs. *Biomech Mod. Mechanobio*, 3:139–155, 2003.
- [Fun81] Y.C. Fung. Biomechanics: Mechanical properties of living tissues. *Springer-Verlag*, 1981.
- [FVMO04] L. Flórez-Valencia, J. Montagnat, and M. Orkisz. 3d graphical models for vascular-stent pose simulation. *Machine Graphics and Vision*, 13(3):235–248, 2004.
- [FWG<sup>+</sup>99] M. Ferrant, S. Warfield, C. Guttman, F. Jolesz, and R. Kikinis. 3d image matching using a finite element based elastic deformation model. *Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 1679:202–209, 1999.
- [GBT04] P. Glardon, R. Boulic, and D. Thalmann. A coherent locomotion engine extrapolating beyond experimental data. *Proc. CASA*, pages 73–84, 2004.
- [GdJB84] F.L.H. Gielen, W. Wallinga de Jonge, and K.L. Boon. Electrical conductivity of skeletal muscle tissue: experimental results from different muscles in vivo. *Med. Biol. Eng. Comput.*, 22:569–577, 1984.

- [GFSF95] N.R. Grubb, A. Fleming, G.R. Sutherland, and K.A.A. Fox. Skeletal muscle contraction in healthy volunteers: Assessment with doppler tissue imaging. *Radiology*, 194:837–842, 1995.
- [GMMT06] B. Gilles, L. Moccozet, and N. Magnenat-Thalmann. Anatomical modelling of the musculoskeletal system from mri. *Proc. of MICCAI'06*, 4190:289–296, 2006.
- [GMTT89] J.P. Gourret, N. Magnenat-Thalmann, and D. Thalmann. Simulation of object and human skin deformations in a grasping task. *Proc. of SIGGRAPH'89, Computer Graphics*, 23:21–30, 1989.
- [GP00] B. Garner and M. Pandy. The obstacle-set method for representing muscle paths in musculoskeletal models. *Comput. Methods Biomech. Biomed. Eng.*, 3:1–30, 2000.
- [GP01] B. Garner and M. Pandy. Musculoskeletal model of the upper limb based on the visible human male dataset. *Comput. Meth. Biomech. Biomed. Eng.*, 4:93–126, 2001.
- [GPMTV04] B. Gilles, R. Perrin, N. Magnenat-Thalmann, and J-P. Vallée. Bones motion analysis from dynamic mri: acquisition and tracking. *Proc. of MICCAI'04*, 2:942–949, 2004.
- [GPV+04] B. Gilles, R. Perrin, J-P. Vallée, L. Moccozet, N. Magnenat-Thalmann, and F. Terrier. A methodology for anatomical 3d modelling of patients bones from mri. *Proc. of ISMRM'04*, 2004.
- [Gra00] H. Gray. Anatomy of the human body. *20th edition. Philadelphia: Lea and Febiger*, 2, 2000.
- [GS83] E.S. Grood and W.J. Suntay. A joint coordinate system for the clinical description of three-dimensional motions: application to the knee. *Journal of Biomechanical Engineering*, 105:136–144, 1983.
- [GSZW98] A. Ghanei, H. Soltanian-Zadeh, and J.P. Windham. A 3d deformable surface model for segmentation of objects from volumetric data in medical images. *Computers in Biology and Medicine*, 28:239–253, 1998.
- [GT95] R. Grzeszczuk and D. Terzopoulos. Automated learning of muscle-actuated locomotion through control abstraction. *Proc. of SIGGRAPH'95*, pages 63–70, 1995.
- [Gue93] A. Guezic. Large deformable splines, crest lines and matching. *Proc. of ICCV*, pages 650–657, 1993.
- [HA] H-Anim. Iso/iec fcd 19774:200x, humanoid animation, <http://h-anim.org>.
- [HAGM04] G. Hamarneh, R. Abu-Gharbieh, and T. McInerney. Medial profiles for modeling deformation and statistical analysis of shape and their use in medical image segmentation. *International Journal of Shape Modeling*, 10(2):187–209, 2004.
- [Hah98] J.K. Hahn. Realistic animation of rigid bodies. *Computer Graphics*, 22(4):299–308, 1998.
- [Har79] R.M. Haralick. Statistical and structural approaches to texture. *Proc. of the IEEE*, 67(5):786–804, 1979.
- [Hau04] M. Hauth. Visual simulation of deformable models. *Phd, Tübingen university*, 2004.
- [Haw02] D. Hawkins. Biomechanics of musculoskeletal tissues. *UC-Davis*, 2002.
- [HE01] M. Hauth and O. Etmuss. A high performance solver for the animation of deformable objects using advanced numerical methods. *Proc. of Eurographics*, 2001.



- [HFDKR00] H.J. Hermens, B. Freriks, C. Disselhorst-Klug, and G. Rau. Development of recommendations for semg sensors and sensor placement procedures. *J Electromyography and Kinesiology*, 10:361–374, 2000.
- [HFS<sup>+</sup>01] G. Hirota, S. Fisher, A. State, C. Lee, and H. Fuchs. An implicit finite element method for elastic solids in contact. *Proc. of Computer Animation*, 2001.
- [HG04] S. Ho and G. Gerig. Profile scale-spaces for multiscale image match. *MICCAI'04*, 1:176–183, 2004.
- [HGGV02] J.D. Hewitt, R.R. Glisson, F. Guilak, and T.P. Vail. The mechanical properties of the human hip capsule ligaments. *J. of Arthroplasty*, 17(1):82–89, 2002.
- [HHD<sup>+</sup>00] M. Holden, D.L. Hill, E.R. Denton, J.M. Jarosz, T.C. Cox, T. Rohlfing, J. Goodey, and D.J. Hawkes. Voxel similarity measures for 3-d serial mr brain image registration. *IEEE Transactions on Medical Imaging*, 19(2):94–102, 2000.
- [Hil38] A.V. Hill. The heat of shortening and the dynamic constants of muscle. *Proc. R. Soc.*, 126:136–195, 1938.
- [HJD<sup>+</sup>03] G. Heers, T. Jenkyn, M.A. Dresner, M.O. Klein, J.R. Basford, K.R. Kaufman, R.L. Ehman, and K.N. An. Measurement of muscle activity with magnetic resonance elastography. *Clinical Biomechanics*, 18:537–542, 2003.
- [HL81] T.J.R. Hughes and W.K. Liu. Nonlinear finite element analysis of shells-1. three-dimensional shells. *Comput Meth Appl Mech Eng*, 26:331–362, 1981.
- [HM06] E. Haber and J. Modersitzki. Intensity gradient based registration and fusion of multi-modal images. *Proc. of MICCAI'06*, (2):726–733, 2006.
- [HP78] S.L. Horowitz and T. Pavlidis. A graph-theoretic approach to picture processing. *CGIP*, 7(2):282–291, 1978.
- [HPP<sup>+</sup>03] K.H. Höhne, B. Pflessner, A. Pommert, K. Priesmeyer, M. Riemer, T. Schiemann, R. Schubert, U. Tiede, H. Frederking, S. Gehrman, S. Noster, and U. Schumacher. Voxel-man 3d navigator: Inner organs. regional, systemic and radiological anatomy. *Springer-Verlag Electronic Media, Heidelberg, ISBN 3-540-40069-9*, 2003.
- [HRS<sup>+</sup>99] A. Hagemann, K. Rohr, H.S. Stiehl, U. Spetzger, and J.M. Gilsbach. A biomechanical model of the human head for elastic registration of mr images. *Bildverarbeitung fur die Medizin*, pages 44–48, 1999.
- [HS81] B.K.P. Horn and B.G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.
- [HSV<sup>+</sup>05] A.M. Heemskerk, G.J. Strijkers, A. Vilanova, M.R. Drost, and K. Nicolay. Determination of mouse skeletal muscle architecture using three-dimensional diffusion tensor imaging. *Magnetic Resonance in Medicine*, 53:1333–1340, 2005.
- [HYC<sup>+</sup>05] D.E. Hyun, S.H. Yoon, J.W. Chang, J.K. Seong, M.S. Kim, and B. Jttler. Sweep-based human deformation. *Proc. of Pacific Graphics '05*, 2005.
- [IMU<sup>+</sup>00] C. Imielinska, D. Metaxas, J. Udupa, Y. Jin, and T. Chen. Hybrid segmentation of the visible human data. *Proc. of the Visible Human Project Conference*, 2000.
- [JD75] R.H. Jensen and D.T. Davy. An investigation of muscle lines of action about the hip: a centroid line approach vs. the straight line approach. *Journal of Biomechanics*, 8:103–110, 1975.

- [JMB00] T. Johansson, P. Meier, and R. Blickhan. A finite-element model for the mechanical analysis of skeletal muscles. *J. Theor. Biol.*, 206(1):131–149, 2000.
- [Jol86] T. Jolliffe. Principle component analysis. *SpringerVerlag, New York*, 1986.
- [JP99] D. James and D. Pai. Artdefo: Accurate real time deformable objects. *Proc. of SIGGRAPH 99, Computer Graphics*, 1999.
- [JT05] D.L. James and C.D. Twigg. Skinning mesh animations. *ACM Transactions on Graphics (SIGGRAPH'05)*, 24(3), 2005.
- [Jul04] P. Julkunen. Implementation, optimization and application of fibril reinforced poroviscoelastic model for articular cartilage. *Master of Science Thesis, Tampere university of technology*, 2004.
- [Kar87] H. Kardestuncer. Finite element handbook. *McGraw-Hill, ISBN-0-07-033305-X*, 1987.
- [KDG91] K. Klaue, C.W. Durnin, and R. Ganz. The acetabular rim syndrome. *J. Bone Joint Surg.*, 73B:423–429, 1991.
- [KdH04] B.L. Kaptein and F.C.T. Van der Helm. Estimating muscle attachment contours by transforming geometrical bone models. *Journal of Biomechanics*, 37:263–273, 2004.
- [KG96] R.M. Koch and M.H. Gross. Simulating facial surgery using finite element models. *Proc. Siggraph '96, Computer Graphics*, 30:421–428, 1996.
- [KGB78] J. Kastelic, A. Galeski, and E. Baer. The multicomposite ultrastructure of tendon. *Connective Tissue Research*, 6:11–23, 1978.
- [KGD64] R.M. Kenedi, T. Gibson, and C.H. Daly. Bioengineering studies of the human skin. *Biomechanics and Related Bioengineering Topics*, ed. by R.M. Kenedi, in . *Oxford: Pergamon Press*, 1964.
- [KGG96] E. Keeve, S. Girod, and B. Girod. Craniofacial surgery simulation. *Proc. of VBC'96*, pages 541–546, 1996.
- [KGSMT03] S. Kshirsagar, S. Garchery, G. Sannier, and N. Magnenat-Thalmann. Synthetic faces: Analysis and applications. *International Journal of Imaging Systems and Technology*, 13(1):65–73, 2003.
- [KHC05] I. Kardos, L. Hajder, and D. Chetverikov. Bone surface reconstruction from ct/mr images using fast marching and level set methods. *Proc. Joint Hungarian-Austrian Conference on Image Processing and Pattern Recognition*, pages 41–48, 2005.
- [KHM<sup>+</sup>98] J.T. Klosowski, M. Held, J.S.B. Mitchell, H. Sowrizal, and K. Zikan. Efficient collision detection using bounding volume hierarchies of k-dops. *IEEE Transactions on Visualization and Computer Graphics*, 4(1):21–36, 1998.
- [KHS01] K. Kahler, J. Haber, and H.P. Seidel. Geometry-based muscle modeling for facial animation. *In Proc. Graphics Interface*, pages 37–46, 2001.
- [KHYS02] K. Kahler, J. Haber, H. Yamauchi, and H.P. Seidel. Head shop: Generating animated head models with anatomical structure. *In Proc. ACM SIGGRAPH Symposium on Computer Animation*, pages 55–64, 2002.
- [KJP02] P. Kry, D. James, and D. Pai. Eigenskin: Real time large deformation character skinning in hardware. *Proc. of SCA '02*, pages 153–159, 2002.

- [KKCO89] L. Kaufman, D.M. Kramer, L.E. Crooks, and D.A. Ortendahl. Measuring signal-to-noise ratios in mr imaging. *Radiology*, 173:265–267, 1989.
- [KL99] S.M. Klisch and J.C. Lotz. Application of a fiber-reinforced continuum theory to multiple deformations of the annulus fibrous. *J. of Biomechanics*, 32:1027–1036, 1999.
- [KM04] T. Kurihara and N. Miyata. Modeling deformable human hands from medical images. *Proc. of SCA'04*, pages 355–363, 2004.
- [KMTM<sup>+</sup>98] P. Kalra, N. Magnenat-Thalmann, L. Moccozet, G. Sannier, A. Aubel, and D. Thalmann. Real-time animation of realistic virtual humans. *Computer Graphics and Applications*, 18(5):42–56, 1998.
- [KN04] T. Komura and A. Nagano. Evaluation of the influence of muscle deactivation on other muscles and joints during gait motion. *J. of Biomechanics*, 37:425–436, 2004.
- [Koz98] J.R. Koza. Genetic programming. *Encyclopedia of Computer Science and Technology*, 39:29–43, 1998.
- [KRG<sup>+</sup>02] M. Koch, S.H.M. Roth, M.H. Gross, A.P. Zimmermann, and H.F. Sailer. A framework for facial surgery simulation. *Proc. of ACM SCCG*, 2002.
- [KSMMT03] M.J. Kang, H. Sadri, L. Moccozet, and N. Magnenat-Thalmann. Hip joint modeling for the control of the joint center and the range of motions. *Proc. of the IFAC symposium on modelling and control in biomedical systems*, pages 23–27, 2003.
- [KWT88] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1:321–331, 1988.
- [KZ05] L. Kavan and J. Zara. Spherical blend skinning: A real-time deformation of articulated models. *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pages 9–16, 2005.
- [LAM99] R.N. Leekam, A.M. Agur, and N.H. McKee. Using sonography to diagnose injury of plantaris muscles and tendons. *American Journal of Roentgenology*, 172:185–189, 1999.
- [Lan79] Y. Lanir. The rheological behavior of the skin: experimental results and a structural model. *Biorheology*, 16:191–202, 1979.
- [Lan87] Y. Lanir. Skin mechanics. in *Handbook of Bioengineering*, ed. by R. Skalak, S. Chien. New York: McGraw-Hill, 1987.
- [Lar86] W.F. Larrabee. A finite element model of skin deformation. i biomechanics of skin and soft tissue: A review. *Laryngoscope*, 96:399–405, 1986.
- [LC87] W. Lorenzen and H. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *Proc. of SIGGRAPH'87, Computer Graphics*, 21(39):163–169, 1987.
- [LCF00] J.P. Lewis, M. Cordner, and N. Fong. Pose space deformation: A unified approach to shape interpolation and skeleton-driven deformation. *Proc. of SIGGRAPH'00*, 2000.
- [LCSK92] M. Lafortune, P. Cavanagh, H.J Sommer, and A. Kalenak. Threedimensional kinematics of the human knee during walking. *Journal of Biomechanics*, 25:347–357, 1992.
- [LEHW01] R. Lemos, M. Epstein, W. Herzog, and B. Wyvill. Realistic skeletal muscle deformation using finite element analysis. *Proc. of SIBGRAPI 2001*, 2001.

- [LHH97] J.A. Little, D.L.G. Hill, and D.J. Hawkes. Deformations incorporating rigid structures. *Computer Vision and Image Understanding*, 6(2):223–232, 1997.
- [LM99] J.O. Lachaud and A. Montanvert. Deformable meshes with automated topology changes for coarse-to-fine three-dimensional surface extraction. *Med. Image Anal.*, 3:1–21, 1999.
- [LM01] J. Lötjönen and T. Mäkelä. Segmentation of mr images using deformable models: Application to cardiac images. *International Journal of Bioelectromagnetism*, 3(2):37–45, 2001.
- [LMT06] J. Liang, T. McInerney, and D. Terzopoulos. United snakes. *Medical Image Analysis*, 10:215–233, 2006.
- [LO99] T.W. Lu and J.J. O’Connor. Bone position estimation from skin marker co-ordinates using global optimisation with joint constraints. *J. of Biomechanics*, 32:129–134, 1999.
- [LRMK99] J. Lötjönen, P.J. Reissman, I.E. Magnin, and T. Katila. Model extraction from magnetic resonance volume data using the deformable pyramid. *Medical Image Analysis*, 3(4):387–406, 1999.
- [LSAB03] L.P. Li, A. Shirazi-Adl, and M.D. Buschmann. Investigation of mechanical behavior of articular cartilage by fibril reinforced poroelastic models. *Biorheology*, 40:227–233, 2003.
- [LT06] S.H. Lee and D. Terzopoulos. Heads up! biomechanical modeling and neuromuscular control of the neck. *Proc. of SIGGRAPH’06*, 2006.
- [LTW95] Y. Lee, D. Terzopoulos, and K. Walters. Realistic modeling for facial animation. *Proc. of SIGGRAPH’95, Computer Graphics*, 29:55–62, 1995.
- [MA92] E. McVeigh and E. Atalar. Cardiac tagging with breath-hold cine-mri. *Magnetic Resonance in Medicine*, 28:318–327, 1992.
- [Mac05] A. Maciel. A biomechanics-based articulation model for medical applications. *Thesis, EPFL no 3360*, 2005.
- [MBT99] T. Molet, R. Boulic, and D. Thalmann. Human motion capture driven by orientation measurements. *Presence*, 8(2):187–203, 1999.
- [MC94] B. Mirtich and J. Canny. Impulse-based dynamic simulation. *The Algorithmic Foundations of Robotics*, 1994.
- [McC82] C.W. McCutchen. Cartilage is poroelastic, not viscoelastic (including an exact theorem about strain energy and viscous loss, and an order of magnitude relation for equilibrium time). *J. of Biomechanics*, 15:325–327, 1982.
- [McM87] T.A. McMahon. Muscle mechanics. *Handbook of Bioengineering*, 8(2):187–203, 1987.
- [MCM<sup>+</sup>05] M. Monziols, G. Collewet, F. Mariette, M. Kouba, and A. Davenel. Muscle and fat quantification in mri gradient echo images using a partial volume detection method. application to the characterization of pig belly tissue. *Magnetic Resonance Imaging*, 23:745–755, 2005.
- [MCV<sup>+</sup>97] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens. Multimodality image registration by maximization of mutual information. *IEEE Transactions on Medical Imaging*, 16(2):187–198, 1997.
- [MD00] J. Montagnat and H. Delingette. Space and time shape constrained deformable surfaces for 4d medical image segmentation. *Proc. of MICCAI ’00*, 1935:196–205, 2000.

- [MD01] J. Montagnat and H. Delingette. A review of deformable surfaces: topology, geometry and deformation. *Image and Vision Computing*, 19(14):1023–1040, 2001.
- [MD05] J. Montagnat and H. Delingette. 4d deformable models with temporal constraints: application to 4d cardiac image segmentation. *Medical Image Analysis*, 9(1):87–100, 2005.
- [Men97] F. Menschik. The hip joint as a conchoid shape. *Journal of Biomechanics*, 30(9):971–973, 1997.
- [Men00] A. Menache. Understanding motion capture for computer animation and video games. *Academic Press, San Diego*, 2000.
- [MET<sup>+</sup>04] S. Mori, M. Endo, T. Tsunoo, S. Kandatsu, S. Tanada, H. Aradate, Y. Saito, H. Miyazaki, K. Satoh, S. Matsushita, and M. Kusakabe. Physical performance evaluation of a 256-slice ct-scanner for four-dimensional imaging. *Medical Physics*, 31(6):1348–1356, 2004.
- [MG03] A. Mohr and M. Gleicher. Building efficient, accurate character skins from examples. *ACM Transactions on Graphics*, 22(3):562–568, 2003.
- [MG04] M. Müller and M. Gross. Interactive virtual materials. *Proc. of Graphics Interface'04*, pages 239–246, 2004.
- [MH06] C. McIntosh and G. Hamarneh. Genetic algorithm driven statistically deformed models for medical image segmentation. *Proc. of MedGEC and GECCO*, 2006.
- [MHHR06] M. Müller, B. Heidelberger, M. Hennix, and J. Ratcliff. Position based dynamics. *Proc. of Virtual Reality Interactions and Physical Simulations (VRIPhys)*, pages 6–7, 2006.
- [MHTG05] M. Müller, B. Heidelberger, M. Teschner, and M. Gross. Meshless deformations based on shape matching. *Proc. of SIGGRAPH'05*, pages 471–478, 2005.
- [Mil98] K. Miller. Modelling soft tissue using biphasic theory a word of caution. *Computer Methods in Biomechanics and Biomedical Engineering*, 1:261–263, 1998.
- [MKAE00] D. MacDonald, N. Kabani, D. Avis, and A.C. Evans. Automated 3-d extraction of inner and outer surfaces of cerebral cortex from mri. *NeuroImage*, 12:340–356, 2000.
- [MKE03] J. Mezger, S. Kimmerle, and O. Etmuss. Hierarchical techniques in collision detection for cloth animation. *WSCG*, 11(2):322–329, 2003.
- [MKLA80] V.C. Mow, S.C. Kuei, W.M. Lai, and C.G. Armstrong. Biphasic creep and stress relaxation of articular cartilage: Theory and experiments. *ASME J Biomech Eng*, 102:73–84, 1980.
- [Mon99] J. Montagnat. Modèles déformables pour la segmentation et la modélisation d'images médicales 3d et 4d. *Phd thesis, Nice-Sophia Antipolis University*, 1999.
- [Moo40] M. Mooney. A theory of large elastic deformation. *J. Appl. Phys.*, 11:582–592, 1940.
- [Mr05] J. Mosegaard and T.S. Sørensen. Gpu accelerated surgical simulators for complex morphology. *Proc. of IEEE Virtual Reality '05*, pages 147–153, 2005.
- [MRL<sup>+</sup>96] R. Muthupillai, P. Rossman, D. Lomas, J. Greenleaf, S. Riederer, and R. Ehman. Magnetic resonance imaging of transverse acoustic strain waves. *Magn. Reson. Med*, 36:266–274, 1996.
- [MRP04] S. Majumder, A. Roychowdhury, and S. Pal. Variations of stress in pelvic bone during normal walking, considering all active muscles. *Trends Biomater. Artif. Organs.*, 17(2):48–53, 2004.

- [MSV95] R. Malladi, J. Sethian, and B. Vemuri. Shape modeling with front propagation : A level set approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2):158–174, 1995.
- [MT95] T. McInerney and D. Terzopoulos. A dynamic finite element surface model for segmentation and tracking in multidimensional medical images with application to cardiac 4d image analysis. *Comp. Med. Imag. Graph.*, 19(1):69–83, 1995.
- [MT96] T. McInerney and D. Terzopoulos. Deformable models in medical image analysis: a survey. *Medical Image Analysis*, 1(2):91–108, 1996.
- [MT97] L. Moccozet and N. Magnenat Thalmann. Dirichlet free-form deformations and their application to hand simulation. *Computer Animation*, pages 93–102, 1997.
- [MT99] T. McInerney and D. Terzopoulos. Topology adaptive deformable surfaces for medical image volume segmentation. *IEEE Transactions on Medical Imaging*, 18(10):840–850, 1999.
- [MT00] W. Maurel and D. Thalmann. Human shoulder modeling including scapulo-thoracic constraint and joint sinus cones. *Computers and Graphics*, 24(2):203–218, 2000.
- [MTLT88] N. Magnenat-Thalmann, R. Laperriere, and D. Thalmann. Joint-dependent local deformations for hand animation and object grasping. *Proc. Graphics Interface*, pages 26–33, 1988.
- [MTT90] N. Magnenat-Thalmann and D. Thalmann. Computer animation: Theory and practice - second revised edition. *Computer Science Workbench, Springer-Verlag*, 1990.
- [MV98] J.B.A. Maintz and M.A. Viergever. A survey of medical registration. *Medical image analysis*, 2(1):1–36, 1998.
- [MWTT98] W. Maurel, Y. Wu, N. Magnenat Thalmann, and D. Thalmann. Biomechanical models for soft tissue simulation. *Springer-Verlag Berlin/Heidelberg*, page 173, 1998.
- [MZ02] S. Mori and P.C. Van Zijl. Fiber tracking: principles and strategies a technical review. *NMR Biomed*, 15:468–480, 2002.
- [NA96] C. Nastar and N. Ayache. Frequency-based nonrigid motion analysis: Application to four dimensional medical images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(11):1067–1079, 1996.
- [ND98] L. Nedel and D. Thalmann. Real-time muscle deformations using mass-spring systems. *Proc. CGI 98, IEEE Computer Society Press*, 1998.
- [Neb01] J.C. Nebel. Soft tissue modeling from 3d scanned data. *Deformable Avatars*, pages 85–97, 2001.
- [NH01] M.P. Nash and P.J. Hunter. Computational mechanics of the heart. *J. Elast*, 61:113–141, 2001.
- [NLM99] The Visible Human Project NLM. National library of medicine, [http://www.nlm.nih.gov/research/visible/visible\\_human.html](http://www.nlm.nih.gov/research/visible/visible_human.html). 1999.
- [NM65] J.A. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, 7:308–313, 1965.
- [NMK+03] A. Nava, E. Mazza, F. Kleinermann, N. J. Avis, and J. Mc-Clure. Determination of the mechanical properties of soft human tissues through aspiration experiments. *Proc. of MIC-CAI'03*, 1:222–229, 2003.

- [NMK<sup>+</sup>05] A. Nealen, M. Miller, R. Keiser, E. Boxerman, and M. Carlson. Physically based deformable models in computer graphics. *In Eurographics State-of-the-Art Report (EG-STAR) 3*, pages 71–94, 2005.
- [NTH00] V. Ng-Thow-Hing. Anatomically-based models for physical and geometric reconstruction of humans and other animals. *Ph.D. Thesis, Department of Computer Science, University of Toronto*, 2000.
- [OH99] J. O'Brien and J. Hodgins. Graphical models and animation of brittle fracture. *Proc. of Siggraph99, Computer Graphics*, pages 137–146, 1999.
- [OLL<sup>+</sup>96] C. Orizio, D. Liberati, C. Locatelli, D. De Grandis, and A. Veicsteinas. Surface mechanomyogram reflects muscle fibres twitches summation. *J. of Biomechanics*, 29:475–481, 1996.
- [OR03] M. O'Donnell and K.W. Rigby. Real-time aberration correction for medical ultrasound. *Proc. of WCU'03*, pages 841–846, 2003.
- [OS88] S. Osher and J. Sethian. Fronts propagating with curvature dependent speed: Algorithms based on hamilton-jacobi formulation. *Journal of Computational Physics*, 79:12–49, 1988.
- [Pai02] D.K. Pai. Strands: Interactive simulation of thin solids using cosserat models. *Proc. of Eurographics'02*, 2002.
- [Pan01] M.G. Pandy. Computer modeling and simulation of human movement. *Annu. Rev. Biomed. Eng.*, 3:245–273, 2001.
- [Par82] F.I. Parke. Parametric model for facial animation. *IEEE Computer Graphics and Applications*, 2(9):61–68, 1982.
- [PBHS05] S. Papazoglou, J. Braun, U. Hamhaber, and I. Sack. Two-dimensional waveform analysis in mr elastography of skeletal muscles. *Phys. Med. Biol.*, 50:1313–1325, 2005.
- [PCS<sup>+</sup>89] C.A. Pelizzari, G.T.Y. Chen, D.R. Spelbring, R.R. Weichselbaum, and C.T. Chen. Accurate three-dimensional registration of ct, pet and/or mr images of the brain. *J. of Computer Assisted Tomography*, 13:20–26, 1989.
- [PDA03] G. Picinbono, H. Delingette, and N. Ayache. Non-linear anisotropic elasticity for real-time surgery simulation. *Graph. Models*, 65(5):305–321, 2003.
- [PDTNA] A. Pitiot, H. Delingette, P.M. Thompson, and JOURNAL =NeuroImage, Special Issue: Mathematics in Brain Imaging VOLUME=23 PAGES=85–96 YEAR =2004 N. Ayache, TITLE = Expert Knowledge Guided Segmentation System for Brain MRI.
- [Pen96] X. Pennec. L'incertitude dans les problèmes de reconnaissance et de recalage – applications en imagerie médicale et biologie moléculaire. *PhD thesis, Ecole Polytechnique*, 1996.
- [PFJ<sup>+</sup>03] S.M. Pizer, P.T. Fletcher, S. Joshi, A. Thall, J.Z. Chen, Y. Fridman, D.S. Fritsch, A.G. Gash, J.M. Glotzer, M.R. Jiroutek, C. Lu, K.E. Müller, G. Tracton, P. Yushkevich, and E.L. Chaney. Deformable m-reps for 3d medical image segmentation. *International Journal of Computer Vision*, 55(2):85–106, 2003.
- [PHBH02] J.M. Penrose, G.M. Holt, M. Beaugonin, and D.R. Hose. Development of an accurate three-dimensional finite element knee model. *Comput Meth Biomech Biomed Eng.*, 5:291–300, 2002.
- [PHSE91] N.J. Pelc, R.J. Herfkens, A. Shimakawa, and D.R. Enzmann. Phase contrast cine magnetic resonance imaging. *Magnetic Resonance Quarterly*, 7:229–254, 1991.

- [PIKV04] R. Perrin, M. Ivancevic, S. Kozerke, and J.P. Vallée. Comparative study of fast gradient echo mri sequences: phantom study. *J Magn Reson Imaging*, 20(6):1030–1038, 2004.
- [Pir05] C. Dan Pirnog. Articular cartilage segmentation and tracking in sequential mr images of the knee. *Thesis 15914, ETH Zurich*, 2005.
- [PL05] J.E. Pierce and G. Li. Muscle forces predicted using optimization methods are coordinate system dependent. *J. of Biomechanics*, 38:695–702, 2005.
- [PLCJ02] P.P. Provenzano, R.S. Lakes, D.T. Corr, and R. Vanderby Jr. Application of nonlinear viscoelastic models to describe ligament behavior. *Biomechan Model Mechanobiol*, 1:45–57, 2002.
- [PLP+02] B. Pflessner, R. Leuwer, A. Petersik, U. Tiede, and K.H. Höhne. A computer-based simulation for petrous bone surgery with haptic feedback. *Comput. Aided Surg.*, 7(2):117, 2002.
- [PM90] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE trans. on Pattern Analysis and Machine Intelligence*, 12(7):629–639, 1990.
- [PMD+06] C.W.A. Pfirmann, B. Mengiardi, C. Dora, F. Kalberer, M. Zanetti, and J. Hodler. Cam and pincer femoroacetabular impingement: Characteristic mr arthrographic findings in 50 patients. *Radiology*, 240:778–785, 2006.
- [PMTK01] J.Y. Park, T. McInerney, D. Terzopoulos, and M.H. Kim. A non-self-intersecting adaptive deformable surface for complex boundary extraction from volumetric images. *Computers and Graphics*, 25(3):421–440, 2001.
- [PN98] L. Porcher-Nedel. Anatomic modeling of human bodies using physically-based muscle simulation. *Ph.D. Thesis, EPFL*, 1831, 1998.
- [Pow87] M.J.D. Powell. Radial basis functions for multivariate interpolation: a review. *Algorithms for Approximation, Oxford University Press*, pages 143–167, 1987.
- [Pro97] X. Provot. Collision and self-collision handling in cloth model dedicated to design garments. *Proc. of Graphics Interface*, pages 177–189, 1997.
- [PSW05] D.K. Pai, D. Sueda, and Q. Wei. Fast physically based musculoskeletal simulation. *Proc. of Sketches and Applications of SIGGRAPH'05*, 2005.
- [PTVF92] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. Numerical recipes in c (2nd ed.). *Cambridge University Press*, 1992.
- [PVC+01] Q.C. Pham, F. Vincent, P. Clarysse, P. Croisille, and I.E. Magnin. A fem-based deformable model for the 3d segmentation and tracking of the heart in cardiac mri. *Proc. of In ISPA '01*, pages 250–254, 2001.
- [PW89] A. Pentland and J. Williams. Good vibrations: Modal dynamics for graphics and animation. *Proc. of SIGGRAPH'89*, 23(3):215–222, 1989.
- [PZ91] M.G. Pandy and F.E. Zajac. Optimal muscular coordination strategies for jumping. *J. of Biomechanics*, 24(1):1–10, 1991.
- [QLH+02] H.H. Quick, M.E. Ladd, M. Hoevel, S. Bosk, J.F. Debatin, G. Laub, and T. Schroeder. Real-time mri of joint movement with truefisp. *Journal of Magnetic Resonance Imaging*, 15(6):710–715, 2002.
- [RAD03] G.K. Rohde, A. Aldroubi, and B.M. Dawant. The adaptive bases algorithm for intensity-based nonrigid image registration. *IEEE Trans. Med. Imag.*, 22:1470–1479, 2003.



- [RDS<sup>+</sup>03] J. Rasmussen, M. Damsgaard, E. Surma, S.T. Christensen, M. de Zee, and V. Vondrak. Anybody - a software system for ergonomic optimization. *Fifth World Congress on Structural and Multidisciplinary Optimization*, 2003.
- [RFS99] K. Rohr, M. Fornefett, and H.S. Stiehl. Approximating thin-plate splines for elastic registration: Integration of landmark errors and orientation attributes. *Lecture Notes in Computer Science*, 1613:252–265, 1999.
- [RGST79] E.F. Rybicki, W.A. Glaeser, J.S. Strenkowski, and M.A. Tamm. Effects on cartilage stiffness and viscosity on a nonporous compliant bearing lubrication models for living joints. *J. of Biomechanics*, 12:403–409, 1979.
- [RGTC98] S.H.M. Roth, M.H. Gross, S. Turello, and F.R. Carls. A bernstein-bézier based approach to soft tissue simulation. *Computer Graphics Forum*, 17(3):285–294, 1998.
- [RJ92] O.M. Rutherford and D.A. Jones. Measurement of fibre pennation using ultrasound in the human quadriceps in vivo. *Eur J Appl Physiol Occup*, 65:433–437, 1992.
- [RKW<sup>+</sup>98] R.C. Rhoad, J.J. Klimkiewicz, G.R. Williams, S.B. Kesmodel, J.K. Udupa, J.B. Kneeland, and J.P. Iannotti. A new in vivo technique for 3d shoulder kinematics analysis. *Skeletal Radiol.*, 27:92–97, 1998.
- [RLK99] D. Reynolds, J. Lucas, and K. Klaue. Retroversion of the acetabulum. a cause of hip pain. *J. of None Joint Surgery*, 81(2):281–288, 1999.
- [RM77] V. Roth and V.C. Mow. Finite element analysis of contact problems for indentation of articular cartilage. *Advances in bioengineering, American society of mechanical engineers*, pages 47–48, 1977.
- [RMA00] A. Roche, G. Malandain, and N. Ayache. Unifying maximum likelihood approaches in medical image registration. *International Journal of Imaging Systems and Technology: Special Issue on 3D Imaging*, 11(1):71–80, 2000.
- [RMPA98] A. Roche, G. Malandain, X. Pennec, and N. Ayache. The correlation ratio as a new similarity measure for multimodal image registration. *proc. of MICCAI'98*, 1496:1115–1124, 1998.
- [RS03] A.J. Rebmann and F.T. Sheehan. Precise 3d skeletal kinematics using fast phase contrast magnetic resonance imaging. *Journal of Magnetic Resonance Imaging*, 17(2):206–213, 2003.
- [RSH<sup>+</sup>99] D. Rueckert, L.I. Sonoda, C. Hayes, D.L.G. Hill, M.O. Leach, and D.J. Hawkes. Nonrigid registration using free-form deformations: Application to breast mr images. *IEEE Trans. Med. Imag.*, 18:712–721, 1999.
- [RSS<sup>+</sup>96] K. Rohr, H.S. Stiehl, R. Sprengel, W. Beil, T. M. Buzug, J. Weese, and M.H. Kuhn. Point-based elastic registration of medical image data using approximating thin-plate splines. *Visualization in Biomedical Computing*, pages 297–306, 1996.
- [RST<sup>+</sup>06] K. Rajamani, M. Styner, H. Talib, L.P. Nolte, and M.A. Gonzalez-Ballester. Statistical deformable bone models for robust 3d surface extrapolation from sparse data. *Medical Image Analysis*, 2006.
- [SACD99] D.J. Schmidt, A.S. Arnold, N.C. Carroll, and S.L. Delp. Length changes of the hamstrings and adductors resulting from derotational osteotomies of the femur. *J. Orthop. Res.*, 17:279–285, 1999.
- [SAG03] V. Surazhsky, P. Alliez, and C. Gotsman. Isotropic remeshing of surfaces: a local parameterization approach. *Proc. of the 12th International Meshing Roundtable*, pages 215–224, 2003.

- [Sal98] K. Saladin. *Anatomy and physiology: The unity of form and function*. McGraw-Hill Company, 1998.
- [SB87] T. Sussman and K.J. Bathe. A finite element formulation for non-linear incompressible elastic and inelastic analysis. *J. Computer and Structures*, 26:357–409, 1987.
- [SBRP99] A. Samani, J. Bishop, E. Ramsay, and D. Plewes. A 3-d contact problem finite element model for breast shape deformation derived from mri data. *23rd Annual Meeting of the American Society of Biomechanics*, 1999.
- [SC04] F.G. Shellock and J.V. Crues. Mr procedures: Biologic effects, safety, and patient care. *Radiology*, 232:635–652, 2004.
- [SCW01] C. Studholme, V.A. Cardenas, and M.W. Weiner. Multiscale image and multiscale deformation of brain anatomy for building average brain atlases. *Proc. SPIE (Medical Imaging 2001: Image Processing)*, 4322:557–568, 2001.
- [SD92] L.H. Staib and J.S. Duncan. Deformable fourier models for surface finding in 3d images. *Proc. of Visualization in Biomedical Computing (VBC)*, pages 90–104, 1992.
- [SD00] D. Shen and C. Davatzikos. Adaptive-focus statistical shape model for segmentation of 3d MR structures. *MICCAI'00*, pages 206–215, 2000.
- [SD06] R.A. Siston and S.L. Delp. Evaluation of a new algorithm to determine the hip joint center. *Journal of Biomechanics*, 39:125–130, 2006.
- [SdDA<sup>+</sup>98] J. Sijbers, A.J. den Dekker, J. Van Audekerke, M. Verhoye, and D. Van Dyck. Estimation of the noise in magnitude mr images. *Magnetic Resonance Imaging*, 16(1):87–90, 1998.
- [SE03] P.J. Schneider and D.H. Eberly. *Geometric tools for computer graphics*. The Morgan Kaufmann Series in Computer Graphics and Geometric Modeling, 2003.
- [SET02] D. Stahl, N. Ezquerro, and G. Turk. Bag-of-particles as a deformable model. *IEEE Computer Society: TCGVG*, 2002.
- [SGT98] A. Singh, D. Goldgof, and D. Terzopoulos. Deformable models in medical image analysis. *IEEE Computer Society*, 1998.
- [She94] J.R. Shewchuk. An introduction to the conjugate gradient method without the agonizing pain. *Technical report, Carnegie Mellon University*, 1994.
- [Sho85] K. Shoemake. Animating rotation with quaternion curves. *Proc. of SIGGRAPH'85, Computer Graphics*, 19:245–254, 1985.
- [SK00] K. Singh and E. Kokkevis. Skinning characters using surface oriented free-form deformations. *Graphics Interface*, pages 35–42, 2000.
- [SKBG95] G. Székely, A. Kelemen, C. Brechbüler, and G. Gerig. Segmentation of 3d objects from mri volume data using constrained elastic deformations of flexible fourier surface models. *Proc. of CVRMed*, pages 495–505, 1995.
- [SL96] R. Szeliski and S. Lavallée. Matching 3-d anatomical surfaces with non-rigid deformations using octree-splines. *J. Comp. Vision*, 18:171–186, 1996.
- [SLP06] D. Skerl, B. Likar, and F. Pernus. A protocol for evaluation of similarity measures for rigid registration. *IEEE Transactions on Medical Imaging*, 25(6):779–791, 2006.

- [SLS<sup>+</sup>92] W.E. Snyder, A. Logenthiran, P. Santago, K. Link, G.L. Bilbro, and S. Rajala. Segmentation of magnetic resonance images using mean field annealing. *Image Vision Comput.*, 10(6), 1992.
- [SMEH99] A.J. Stoddart, P. Mrazek, D. Ewins, and D. Hynd. Marker based motion capture in biomedical applications. *Proc. of the IEE colloquium on motion analysis and tracking*, 1999.
- [SMT03] H. Seo and N. Magnenat-Thalmann. An automatic modeling of human bodies from sizing parameters. *ACM SIGGRAPH 2003 Symposium on Interactive 3D Graphics*, pages 19–26, 2003.
- [SNF05] E. Sifakis, I. Neverov, and R. Fedkiw. Automatic determination of facial muscle activations from sparse motion capture marker data. *Proc. of SIGGRAPH'05, TOG*, 24, 2005.
- [SP86] T. Sederberg and S. Parry. Free form deformations of solid geometric models. *Proc. of SIGGRAPH'86, Computer Graphics*, 20(4):150–161, 1986.
- [SPCM97] F. Scheepers, R.E. Parent, W.E. Carlson, and S.F. May. Anatomy-based modeling of the human musculature. *Proc. of Siggraph97, Computer Graphics*, pages 163–172, 1997.
- [SPMC96] F. Scheepers, R. Parent, S. F. May, and W. Carlson. A procedural approach to modeling and animating the skeletal support of the upper limb. *Technical Report, OSU-ACCAD-1/96/TR1, The Ohio State University*, 1996.
- [SS99] D.E.T Shepherd and B.B Seedhom. Thickness of human articular cartilage in joints of the lower limb. *Ann Rheum Dis*, 58, 1999.
- [SSLF86] P.A. Shoemaker, D. Scheider, M.C. Lee, and Y.C. Fung. A constitutive model for two-dimensional soft tissues and its application to experimental data. *J. Biomechanics*, 19(6):695–702, 1986.
- [SSLS<sup>+</sup>01] D.W. Shattuck, S.R. Sandor-Leahy, K.A. Schaper, A. Rottenberg, and R.M. Leahy. Magnetic resonance image tissue classification using a partial volume. *Neuroimage*, 13:856–876, 2001.
- [ST94] T. Saito and J.I. Toriwaki. New algorithms for euclidean distance transformations of an n-dimensional digitised picture with applications. *Pattern Recognition*, 27(11), 1994.
- [Sto94] G. Storvik. A bayesian approach to dynamic contours through stochastic sampling and simulated annealing. *IEEE Trans. PAMI*, 16:976–986, 1994.
- [SZD97] F.T Sheehan, F.E Zajac, and J.E. Drace. Using cine phase contrast magnetic resonance imaging to non-invasively study in vivo knee dynamics. *Journal of Biomechanics*, 31:21–26, 1997.
- [SZM93] L.A. Setton, W. Zhu, and V.C. Mow. The biphasic poroviscoelastic behavior of articular cartilage: Role of the surface zone in governing the compressive behavior. *J. of Biomechanics*, 26:581–592, 1993.
- [TA75] G.J. Torotra and N.P. Anagnostakos. Principles of anatomy and physiology. *Canfield Press, New York*, 1975.
- [TA03] S. Tashman and W. Anderst. In vivo measurement of dynamic joint motion using high speed biplane radiography and ct: application to canine acl deficiency. *Biomechanical Engineering*, 125:238–245, 2003.
- [TA06] D.G. Thelen and F.C. Anderson. Using computed muscle control to generate forward dynamic simulations of human walking from experimental data. *J. of Biomechanics*, 39(6):1107–1115, 2006.

- [TBHF03] J. Teran, S. Blemker, V. Ng Thow Hing, and R. Fedkiw. Finite volume methods for the simulation of skeletal muscle. *Proc. of Eurographics/SIGGRAPH Symposium on Computer Animation*, 2003.
- [TBP03] J. Tsao, P. Boesiger, and K.P. Pruessmann. k-t blast and k-t sense: dynamic mri with high frame rate exploiting spatiotemporal correlations. *Magnetic Resonance in Medicine*, 50(5):1031–1042, 2003.
- [TDG<sup>+</sup>95] S. Tashman, K. DuPre, H. Goitz, T.L. Kolowich, and M. Flynn. A digital radiographic system for determining 3-d joint kinematics during movement. *Proc. 19th Annu. Meeting Amer. Soc. Biomech.*, pages 249–250, 1995.
- [TF88] D. Terzopoulos and K.W. Fleischer. Deformable models. *The Visual Computer*, 4(6):306–331, 1988.
- [TG70] S.P. Timoshenko and J.N. Goodier. Theory of elasticity. *McGraw-Hill*, 1970.
- [Thi95] J.P. Thirion. Fast non-rigid matching of 3d medical images. *INRIA Technical report 2547*, 1995.
- [THMG04] M. Teschner, B. Heidelberger, M. Müller, and M. Gross. A versatile and robust model for geometrically complex deformable solids. *Proc. of Computer Graphics International CGI'04*, pages 312–319, 2004.
- [Tho29] A. Thomson. A handbook of anatomy for art students. *Dover Publications, Fifth Edition, New York*, 1929.
- [TKH<sup>+</sup>05] M. Teschner, S. Kimmerle, B. Heidelberger, G. Zachmann, L. Raghupathi, A. Fuhrmann, M-P. Cani, F. Faure, N. Magnetat-Thalmann, W. Strasser, and P. Volino. Collision detection for deformable objects. *Computer Graphics Forum*, 24(1):61–81, 2005.
- [TM91] D. Terzopoulos and D. Metaxas. Dynamic 3d models with local and global deformations: Deformable superquadrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence archive*, 13(7):703–714, 1991.
- [TPBF87] D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer. Elastically deformable models. *Computer Graphics*, 21(4):205–214, 1987.
- [TPGB99] G.M. Treece, R.W. Prager, A.H. Gee, and L. Berman. Fast surface and volume estimation from non-parallel cross-sections, for freehand three-dimensional ultrasound. *Med Image Analysis*, 3(2):141–173, 1999.
- [TS92] D. Terzopoulos and R. Szeliski. Active vision, chapter tracking with kalman snakes. *MIT Press*, 1992.
- [TSB<sup>+</sup>05] J. Teran, E. Sifakis, S. Blemker, V. Ng Thow Hing, C. Lau, and R. Fedkiw. Creating and simulating skeletal muscle from the visible human data set. *IEEE TVCG*, 11:317–328, 2005.
- [TSC96] D. Thalmann, J. Shen, and E. Chauvineau. Fast realistic human body deformations for animation and vr applications. *Computer Graphics International*, 1996.
- [TSF05] W. Tsang, K. Singh, and E. Fiume. Helping hand: An anatomically accurate inverse dynamics solution for unconstrained hand motion. *Proc. of SCA'05*, pages 1–10, 2005.
- [TSIF05] J. Teran, E. Sifakis, G. Irving, and R. Fedkiw. Robust quasistatic finite elements and flesh simulation. *Proc. of the ACM/Eurographics Symposium on Computer Animation (SCA)*, pages 181–190, 2005.

- [TT93] R. Turner and D. Thalmann. The elastic surface layer model for animated character construction. *Proc. ComputerGraphics International*, pages 399–412, 1993.
- [TVG95] R.J. Thielke, R. Vanderby, and E.S. Grood. Volumetric changes in ligaments under tension. *Proc. of ASME Summer Bioengineering Conference*, 1995.
- [TW88] D. Terzopoulos and A. Witkin. Physically based models with rigid and deformable components. *IEEE Computer Graphics and Applications*, 8(6):41–51, 1988.
- [TWK87] D. Terzopoulos, A. Witkin, and M. Kass. Symmetry-seeking models and 3d object reconstruction. *International Journal of Computer Vision*, 1:211–221, 1987.
- [Ver67] L. Verlet. Computer experiments on classical fluids. ii. equilibrium correlation functions. *Physical Review*, 165:201–204, 1967.
- [VGW06] A.I. Veress, G.T. Gullberg, and J.A. Weiss. Measurement of strain in the left ventricle during diastole with cine-mri and deformable image registration. *ASME Journal of Biomechanical Engineering*, 2006.
- [VHS<sup>+</sup>97] B.C. Vemuri, S. Huang, S. Sahni, C.M. Leonard, C. Mohr, T. Lucas, R. Gilmore, and J. Fitzsimmons. A robust and efficient algorithm for image registration. *IPMI*, pages 465–470, 1997.
- [Vio95] P.A. Viola. Alignment by maximization of mutual information. *Ph.D. thesis, Massachusetts Inst. Technol.*, 1995.
- [VM04] D. Valtorta and E. Mazza. Dynamic measurements of soft tissue viscoelastic properties with a torsional resonator device. *Proc. of MICCAI'04*, 2:284–292, 2004.
- [VMT00a] P. Volino and N. Magnenat-Thalmann. Accurate collision response on polygonal meshes. *Proc. of Computer Animation*, pages 154–163, 2000.
- [VMT00b] P. Volino and N. Magnenat-Thalmann. Implementing fast cloth simulation with collision response. *Proc. of the International Conference on Computer Graphics*, pages 257–268, 2000.
- [VMT01] P. Volino and N. Magnenat-Thalmann. Comparing efficiency of integration methods for cloth animation. *Proc. of Computer Graphics International (CGI)*, pages 265–274, 2001.
- [VMT05] P. Volino and N. Magnenat-Thalmann. Implicit midpoint integration and adaptive damping for efficient cloth simulation: Collision detection and deformable objects. *Computer Animation and Virtual Worlds*, 16:163–175, 2005.
- [VW70] D.R. Veronda and R.A. Westmann. Mechanical characterization of skin-finite deformations. *J. of Biomechanics*, 3:111–124, 1970.
- [VYCL03] B.C. Vemuri, J. Ye, Y. Chen, and C.M. Leonard. Image registration via level-set motion: applications to atlas-based segmentation. *Med Image Anal*, 7(1):1–20, 2003.
- [WC03] J.P. Wilssens and A. De Cock. Barefoot, insole and ground pressure measurement comparison during 5m/s running. *Proc. of the 6th Symposium on footwear biomechanics*, 2003.
- [WCM92] R.P. Woods, S.R. Cherry, and J.C. Mazziotta. Rapid automated algorithm for aligning and reslicing pet images. *J. Comp. Assisted Tomogr.*, 16(4):620–633, 1992.
- [WDO<sup>+</sup>04] R.S. Witte, D.E. Dow, R. Olafsson, Y. Shi, and M. O'Donnell. High resolution ultrasound imaging of skeletal muscle dynamics and effects of fatigue. *Proc. of IEEE Ultrasonics Symposium*, pages 764–767, 2004.

- [WG97] J. Wilhelms and A. Van Gelder. Anatomically based modeling. *Proc. of Siggraph'97, Computer Graphics*, 31:173–180, 1997.
- [WG01] J.A. Weiss and J.C. Gardiner. Computational modeling of ligament mechanics. *Critical Reviews in Biomedical Engineering*, 29(4):1–70, 2001.
- [WGL<sup>+</sup>05] J.A. Weiss, J.C. Gardiner, T.J. Lujan, B.J. Ellis, and N. Phatak. Three-dimensional finite element modeling of ligaments: Technical aspects. *Medical Engineering and Physics*, 27:845–861, 2005.
- [WGW90] A. Witkin, M. Gleicher, and W. Welch. Interactive dynamics. *Computer Graphics*, 24(2):11–22, 1990.
- [WHQ05] K. Wang, Y. He, and H. Qin. Incorporating rigid structures in non-rigid registration using triangular b-splines. *VLSM*, pages 235–246, 2005.
- [Wit70] C. Withrow. A dynamic model for computer-aided choreography. *UTEC CSc-70-103, Dept Computer Science, Univ. Utah*, 1970.
- [WKMT99] Y. Wu, P. Kalra, and N. Magnenat-Thalmann. Simulation of static and dynamic wrinkles of skin. *The visual computer*, 15:183–198, 1999.
- [WMG96] J.A. Weiss, B.N. Maker, and S. Govindjee. Finite element implementation of incompressible, transversely isotropic hyperelasticity. *Computer Methods in Applied Mechanics and Engineering*, 135:107–128, 1996.
- [WS00] Y. Wang and L.H. Staib. Physical model-based non-rigid registration incorporating statistical shape information. *Medical Image Analysis*, 4:7–20, 2000.
- [WSA<sup>+</sup>02] G. Wu, S. Siegler, P. Allard, C. Kirtley, A. Leardini, D. Rosenbaum, M. Whittle, D.D. D’Lima, L. Cristofolini, H. Witte, O. Schmid, and I. Stokes. Isb recommendation on definitions of joint coordinate system of various joints for the reporting of human joint motion-part i: ankle, hip, and spine. *Journal of biomechanics*, 35:543–548, 2002.
- [WTGW03] T. Williams, C. Taylor, Z. Gao, and J. Waterton. Corresponding articular cartilage thickness measurements in the knee joint by modelling the underlying bone. *Proc. of MICCAI'03*, 2:480–487, 2003.
- [WVA<sup>+</sup>96] W.M. Wells, P. Viola, H. Atsumi, S. Nakajima, and R. Kikinis. Multi-modal volume registration by maximization of mutual information. *Medical Image Analysis*, 1(1):35–51, 1996.
- [WZG<sup>+</sup>04] G. Wang, G. Zheng, P.A. Grützner, J. Von Recum, and L.P. Nolte. A ct-free intraoperative planning and navigation system for high tibial dome osteotomy. *Proc. of MICCAI'04*, 2:610–620, 2004.
- [XP98] C. Xu and J.L. Prince. Snakes, shapes, and gradient vector flow. *IEEE trans. on image processing*, 7(3):359–369, 1998.
- [YCGMMT04] L. Yahia-Cherif, B. Gilles, T. Molet, and N. Magnenat-Thalmann. Motion capture and visualization of the hip joint with dynamic mri and optical systems. *Computer Animation and Virtual Worlds.*, 15:377–385, 2004.
- [YCMT06] L. Yahia-Cherif and N. Magnenat-Thalmann. Quantification of skin movements artefacts using mri. *Proc. of the 9th International Symposium on 3D analysis of Human Movement*, 2006.
- [Yos92] S. Yoshimoto. Ballerinas generated by a personal computer. *Journal of Visualization and Computer Animation*, 3:85–90, 1992.

- [YSAT01] B. You, P. Siy, W. Anderst, and S. Tashman. In vivo measurement of 3-d skeletal kinematics from sequences of biplane radiographs: Application to knee kinematics. *IEEE Transactions on medical imaging*, 20(6):514–524, 2001.
- [YSD04] J. Yang, L.H. Staib, and J.S. Duncan. Neighbor-constrained segmentation with level set based 3-d deformable models. *IEEE Transactions on Medical Imaging*, 23(8):940–948, 2004.
- [Zaj89] F.E. Zajac. Muscle and tendon: properties, models, scaling and application to biomechanics and motor control. In *CRC Critical Reviews in Biomedical Engineering*, 17(4):359–411, 1989.
- [ZCK98] Q-H Zhu, Y. Chen, and A. Kaufman. Real-time biomechanically-based muscle volume deformation using fem. *Proc. of Eurographics98, Computer Graphics Forum*, pages 275–284, 1998.
- [ZGNW01] L. Zöllei, E. Grimson, A. Norbash, and W. Wells. 2d-3d rigid registration of x-ray fluoroscopy and ct images using mutual information and sparsely sampled histogram estimators. *IEEE CVPR*, 2001.
- [ZH06] H. Zhou, H. Hu, N.D. Harris, and J. Hammerton. Applications of wearable inertial sensors in estimation of upper limb movements. *Biomedical Signal Processing and Control*, 1:22–32, 2006.
- [Zie77] O. Zienkiewicz. The finite element method. *McGraw-Hill, London, 3rd edition*, 1977.
- [ZLW03] L. Zuo, J.T. Li, and Z.Q. Wang. Anatomical human musculature modeling for real-time deformation. *Proc. of WSCG 03*, 2003.
- [ZPR<sup>+</sup>88] E. Zerhouni, D. Parish, W. Rogers, A. Yang, and E. Shapiro. Human heart: Tagging with mr imaging method for noninvasive assessment of myocardial motion. *Radiology*, 169:59–63, 1988.