UNIVERSITÉ DE GENÈVE

FACULTÉ DES SCIENCES
Département d'informatique — Professeur J. Rolim

FACULTÉ DES SCIENCES
ÉCONOMIEQUES ET SOCIALES
Département de systèmes d'information — Professeur N. Magnenat-Thalmann

---

Parameterized Human Body Modeling

THÈSE

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

par

Hyewon SEO
de
Taejone (Corée du Sud)

Thèse N° 3500

GENÈVE
Atelier de reproduction de la Section de physique
2004

To my parents and family members

Acknowledgements

I wish to express sincere appreciation to my thesis supervisor Professor Nadia Magnenat-Thalmann for her guidance and encouragement during this work. As well, the great personal energy and enthusiasm she brings to all her pursuits have not gone unnoticed. I would also like to address my thanks to the jury members of the thesis Prof. José Rolim (University of Geneva), Prof. KwangYun Wohn (Korea Advanced Institute of Science and Technology) and Prof. Thomas Vetter (University of Basel) for the final review.

My research work at MIRALab benefited from the multidisciplinary nature of the group, as each discipline contributed to crucial element in this thesis. In particular, I want to thank Nedjma Cadi for providing her expertise on skinning and for her kind assistance with the user manual, Marlène Arevalo for providing generic models and decors, Grégoire L'Hoste for preprocessing the scan data and for helping me with video editings, and Christiane Luible for dressing the body models. Thanks go out to Pascal Volino for his collision detection library, Alexandro Foni for helping me with rendering images, and Tom Molet for the motion capturing. The French resume in this dissertation could be completed only with the translation and proof-reading of Lydia Yahia-Cherif and Etienne Lyard. My special thanks to Frederic Cordier, not only for his skeleton driven deformation library, but also for many exciting discussions and enjoyable collaboration during and outside my thesis work. I extend my thanks to other – former and present – members of MIRALab, really too many to list here, who have influenced me through their friendship, enthusiasm, and discussions.

I also wish to thank my Korean friends in Geneva and in Lausanne, for making my stay at Switzerland very pleasurable.

My avenue of research can be traced back to the time at KAIST, where I finished my undergraduate studies and the master degree. Professor Kwang-Yun Wohn and Professor Sung-Yong Shin were the first persons to introduce me virtual reality and computer graphics. It was with them I learned to formulate problems through, and to be willing to inform myself with mathematical tools to solve them.

Not to forget my 'old buddies' I have known since my undergraduate studies at KAIST – you know who you are. I was extremely lucky to be in friendship with these amazing people. In particular, there have been continuous discussions with (now Dr.) Jun-Hwan Kim, who has helped indirectly to make this work more acceptable and understandable.

Finally, I would also like to thank my family members, who showed confidence in me throughout this entire process. I am indebted for your endless love and support.

This research was funded in part by the FNRS (Le Fonds National Suisse de la Recherche Scientifique or the Swiss National Research Foundation) and in part by OFES (Office Fédéral de l'Éducation et de la Science or the Federal Office for Education and Science) in the context of European projects.

Abstract

Highly realistic virtual human models are rapidly becoming commonplace in computer graphics. These models, often represented by complex shape and requiring labor-intensive process, challenge the problem of automatic modeling. While there has been a variety of body modeling research work, previous methods are not able to simultaneously satisfy the controllability and the quality shape that are efficiently obtained from shape capture, in the domain of whole-body modeling. By controllability we refer the modification of existing models (posture, size, and other body-related properties) by the user to meet new needs. This thesis addresses the problem of time-saving generation of realistic, controllable human whole-body models, primarily for real-time applications. The specific focus of this work lies in the appropriate combination of statistics and geometry for estimating the most plausible physique from a few user-supplied parameters.

In order to gain insight into the variety (and commonality) of physical appearance, a database consisting of approximately 100 captured 3D body geometry of real people has been used. To establish the geometric correspondence among these data, an optimization-based fitting method is proposed, which finds the error and energy minimizing transformation of a template model onto each scan geometry in the database. The PCA (Principal Component Analysis) of the body geometry set consisting of skeleton configuration and the skin mesh is carried out. This has resulted into a compact, redundancy-free representation of the body geometry.

Given the prepared ‘example’ models, the mapping function that we termed as modeling synthesizer is constructed, which smoothly transforms the control parameter space onto the body geometry space. The measured sizes and shapes of real people in the database are used as valuable resources to reason about the most plausible body geometry given the control parameter set. Subsequently, the modeler enables the user to automatically and effectively generate new desired models that best match the input parameter sets, almost instantly.

Finally, the system is further extended by identifying the geometry components that are parameter-variant and those that are inherent to characteristics of individual body, through a mathematical tool of weighted regression. By using what we call modifier synthesizer, various manipulations of an individual are demonstrated according to more complex parameters such as fat percentage.

Résumé

De nos jours, les modèles d'humains virtuels très réalistes se banalisent rapidement en infographie. Ces dernières années ont vu l'émergence (tant pour l'aspect que pour l'animation) de personnages humanoïdes crédibles dans les films, les jeux, l'animation 3D, et les émissions de télévision. Ces modèles, souvent représentés à l'aide de formes complexes et exigeant d'être traités manuellement, posent le problème de l'automatisation de ce processus. Les techniques de modélisation actuelles, qui exigent des jours ou même des mois de travail par des équipes entières d'infographistes talentueux, pourraient être remplacées par des techniques plus efficaces, rendant le processus presque automatique ou même instantané.

Cette thèse traite l'une des tâches les plus difficiles rencontrées par les infographistes – modéliser le corps humain. Plus spécifiquement, nous nous concentrons sur les techniques de paramétrisation pour modéliser la forme statique de corps (la variété de formes du corps humain). L'objectif premier de ce travail est de produire une génération de modèles complets d'humanoïdes, économique en temps de travail, réaliste et contrôlable. En même temps, nous recherchons une manière systématique d'obtenir la diversité et l'individualité de l'aspect physique des corps qui sont propres aux vraies personnes.

Modéliser et animer le corps humain est l'une des tâches les plus difficiles que les infographistes rencontrent. Plus particulièrement, modéliser le corps humain de manière réaliste exige une surface géométrique précise. Les outils de nos jour disponibles permettent la conception interactive des corps humains à partir de rien ou en modifiant un modèle existant. Ils exigent cependant de considérable intervention des utilisateurs et nécessitent ainsi un temps relativement long de production. Ces outils n'offrent pas de méthodes de haut niveau pour pouvoir modéliser le corps humain. D'ailleurs, respecter certaines contraintes géométriques sur le modèle est une tâche difficile et méticuleuse, même avec les logiciels les plus sophistiqués.

D'un autre côté, la modélisation assistée par ordinateur a à sa dispose d’une variété de méthodes de scanner 3D qui ont été spécifiquement développées pour le corps humain, et avec lesquelles on peut capturer et digitaliser les formes et les tailles de vrais corps humains. En particulier, les scanners 3D deviennent de plus en plus disponibles et permettent ainsi la capture de la forme du corps humain avec un échantillonnage dense et précis de façon rapide et efficace. Malheureusement, ils sont par essence limités au recodage des formes qu'ils observent - il n'y a aucun moyen simple de modifier automatiquement les formes une fois qu'elles ont été créées. En outre, les données ainsi obtenues doivent être simplifiées et le bruit de mesure doit être réduit avant qu'elles ne puissent être converties en modèles utilisables pour la visualisation ou la modélisation. Pire encore, pour les animer, il faut d'abord trouver le centre des articulations du squelette et établire une relation appropriée en les os et la peau.

Il serait intéressant de pouvoir contrôler la forme d'un modèle géométrique 3D à l'aide de paramètres de haut niveau, par exemple sa taille, qualité de forme, complexité, matériaux, etc. Cette thèse traite le problème de la manipulation de la taille et des contraintes de forme du corps humain. Etre capable de modifier automatiquement les formes pour satisfaire les nouveaux besoins est souhaitable pour au moins deux raisons. Premièrement, c'est souvent pour satisfaire ces nouveaux besoins que nous voulons modifier les formes. Dans une application textile, par exemple, nous pourrions vouloir créer un modèle de corps 3D de

manière à ce qu'il satisfasse un certain nombre de contraintes sur ses mensurations avec un minimum d'intervention de la part de l'utilisateur, et ce dans un contexte d'application temps réel et interactif. Deuxièmement, la modification automatique permet d'éviter les répétitions dans une foule. Sans devoir créer ou reconstruire chacun individu, on peut par exemple obtenir des foules en mélangeant des modèles existants.

Les techniques basées sur l'exemple sont de bonnes alternatives pour surmonter de telles limitations, puisqu'elles permettent d'influer sur la qualité des exemples. En particulier, ces méthodes qui extraient des statistiques utiles à partir des exemples sont prometteuses car elles peuvent guider l'évaluation d'information incertaine et renforcer ainsi la robustesse du modeleur. En effet, la modélisation de l'humain semble être un secteur où les statistiques se révèlent très efficaces (Blanz et Vetter, 1999) (DeCarlo et autres, de 1998 utile). Cependant l'utilisation des exemples fut limitée à la modélisation du visage ou seulement à des sous parties du corps, sans doute en raison de la complexité inhérente à nos corps. Ces méthodes seront discutées plus en détail au Chapitre 2.

En résumé, en dépit de nombreux travaux de recherche sur la modélisation de corps, les méthodes présentées n'ont pas réussi à offrir simultanément la contrôlabilité et la qualité qui émanent des formes capturées, dans le domaine de la modélisation de corps entiers. Par la contrôlabilité nous entendons la modification de certaines propriétés du modèle (maintien, taille, et d'autres attributs relatifs au corps) par l'utilisateur pour répondre à de nouvelles exigences. La recherche décrite ici examine le problème de la génération rapide des modèles de corps humains entiers réalistes et contrôlables, principalement pour des applications en temps réel. La base de ce travail est la combinaison appropriée des statistiques et de la géométrie pour estimer le physique le plus plausible à partir de quelques paramètres donnés par l'utilisateur. En comparaison avec les techniques précédentes qui sont principalement basées sur une étude statistique d'une base de données anthropométriques, notre approche combine directement les données anthropométriques avec la forme géométrique des corps humains. Une telle combinaison permet d'étudier de façon précise les corrélations entre la forme des corps et les données anthropométriques correspondantes.

A. Conception de modeleur (le Chapitre 3)

Une méthode efficace qui a été largement utilisée dans la biologie et la morphométrie pour analyser la corrélation et la variation entre les formes d'une même classe est de considérer les transformations nécessaires pour déformer une forme en une autre. De cette façon, il est possible de créer une famille de formes continues avec les caractéristiques désirées. Si un ensemble de formes ayant des caractéristiques formelles distinctes est défini, nous pouvons produire d'une forme qui combine ces caractéristiques dans n'importe quelle proportion en utilisant l'interpolation. De telles transformations de forme peuvent être employées de diverses manières avec comme intérêt la modélisation ou bien l'évaluation et sont au centre de cette thèse. En créant des modèles de déformations paramètrés, nous montrons comment un modeleur peut utiliser à bon escient des données et des connaissances sur la forme et les dimensions du corps.

Afin de gagner en perspicacité dans la variété (et la vulgarisation) de l'aspect physique, nous fondons notre modeleur sur l'analyse statistique de la géométrie 3D des corps capturés sur de réelles personnes. Cette information fournit la meilleure source disponible pour

modéliser et estimer la corrélation entre les paramètres et la forme. Pour établir la correspondance géométrique entre ces exemples, nous proposons une méthode d'ajustement basée sur l'optimisation (Chapitre 4). Ceci constitue la première partie de ce travail de thèse. En cherchant les transformations géométriques qui permettent d'ajuster le modèle générique à chaque exemple de corps scanné, nous obtenons un ensemble de corps humains partageant la même topologie et le même squelette.

Etant donné les modèles "exemples", nous construisons la fonction de correspondance que nous avons appelé "générateur de modèles", et qui transforme de façon continue l'espace des paramètres de contrôle vers celui de la géométrie des corps entiers (Chapitre 5). La base de données des tailles et des formes mesurées sur de vraies personnes est employée en tant qu'élément de "réflexion" valable sur la géométrie de corps la plus probable étant donné un ensemble de paramètres de contrôles. Puisque ce problème peut être ramené à un problème d'interpolation, divers algorithmes tels que la fonction à base radiale (RBF) peut être utilisée. Cependant, les techniques standard de RBF exigeraient trop de fonctions d'interpolation, dues à la dimension élevée du descripteur de forme. Notre approche à ce problème est d'adopter une représentation compacte et efficace des formes en calculant la base de L'ACP. L'ACP (analyse en composantes principales) de l'ensemble des caractéristiques géométriques du corps, composé de la configuration du squelette et du déplacement du maillage de la peau, est calculée. Ceci a produit une représentation compacte et sans redondance de la géométrie du corps. Plus tard, durant l'exécution du programme, le modeleur permettra à l'utilisateur de produire automatiquement et efficacement de nouveaux modèles qui correspondent aux paramètres d'entrée, presque instantanément. Pour la re-animation des modèles résultants, nous recalculons l'attachement de peau grâce aux données de peau initiales et le déplacement nouvellement évalué de la peau.

Finalement, nous nous intéressons à l’extension du system en identifiant les composantes géométriques qui dépendent des paramètres et celles qui sont inhérentes aux caractéristiques de chaque corps, à travers un outil mathématique de régression linéaire (Chapitre 6). En utilisant ce que nous appelons le générateurs de modifications, diverses modifications d’un individu sont effectuées selon des paramètres plus complexes comme le pourcentage de graisse.

L’une des contributions les plus importante de ce travail de thèse est un environnement de travail qui combine la puissance de diverses techniques réunies au sein de ce modeleur robuste: des formes de qualités et réalistes obtenues par des techniques de capture, les méthodes statistiques qui capturent les différences et les points communs des modèles 3D, la technique d’apprentissage automatique par l’exemple, et la capacité de contrôler la posture et la forme d’un modèle de corps entier en utilisant des paramètres de haut niveau. Les contributions spécifiques apportées à chaque composante de cet environnement de travail sont discutées dans les chapitres associés.

Récemment, d’autres travaux (Blanz et Vetter, 1999)(Sloan et al, 2001)(Allen et al, 2002) ont traitée le problème de la modélisation et l’animation paramétrique d’humains en utilisant des surfaces de peau scannées et en les combinent durant la manipulation, il y a cependant plusieurs différences importantes qui distinguent notre approche des autres: tout d’abord, nous couvrons une variété de problèmes plus large en modélisant le corps tout entier. Ensuite, nous utilisons une représentation unique et compacte de la géométrie du corps. Enfin, nous nous

intéressons principalement à générer diverses apparences, bien qu'il y ait principalement des déformations dynamiques lors de l'animation de corps.

Le système est composé principalement de trois modules – le traitement de données du scanner 3D, la construction des générateurs paramétriques, l'évaluateur temps réel qui génère de nouvelles géométries de corps à partir des paramètres spécifies par l'utilisateur. Comme nous l'avons dit précédemment, le système repose sur des systèmes de modélisation extérieurs ou bien des techniques de capture de géométrie pour créer les échantillons de formes initiaux. Etant donné les échantillons exemples, le *module de préparation des modèles d'exemple* assure la correspondance entre eux en collant un modèle générique sur chaque exemple et en utilisant une méthode d'ajustage basée sur l'optimisation. De plus, les attributs des modèles d'exemple sont mesurés et annotés. A partir des échantillons préparés et des attributs qui leur correspondent, le *module de construction du générateur* détermine la fonction qui transforme l'espace des paramètres (attributs) vers l'espace des corps tout entiers. A partir du générateur construit durant cette phase, le module temps réel évalue les déformations nécessaires du modèle générique pour obtenir le modèle désiré à la volée.

Les avantages d'un tel system sont triples. Premièrement, l'utilisateur peut aisément contrôler le modèle obtenu en changeant les paramètres. Deuxièmement, on produit de la géométrie plutôt réaliste. Finalement, les modèles générés sont immédiatement utilisables dans un environnement de visualisation ou de modélisation, qui utilise les données d'attachement de la peau initialement affectées au modèle générique.

B. Préparation des modèles d'exemple: Chapitre 4 et (Seo and Magnenat-Thalmann, 2003)

Comme il est décrit ci-dessus, nos modèles d'exemple sont des modèles existants principalement obtenus à partir de scanners 3D. Ces modèles, chacun ayant une topologie et une posture différentes, ne sont pas utilisables directement pour la transformation de formes. Comme nous le verrons plus en détails au cours du Chapitre 5 et du Chapitre 6, nos modeleurs sont basés sur la transformation de formes, dans laquelle on se sert des exemples mis en correspondance pour créer une portion continue de l'espace des paramètres. Ainsi, un des pré-requis cruciaux de telles transformations est l'identification de points de correspondance sur chaque exemple scanné. De plus, on doit estimer le centre des articulations de chaque modèle scanné car on désire que le modèle ainsi obtenu soit animable à n'importe quel instant de la transformation de ses formes. Cela implique que tous les exemples partagent la même topologie de maillage (même nombre de sommets et connectivite identique) ainsi que la hiérarchie des joints. Remarquez que cela nous permet de représenter la géométrie des corps avec un vecteur de taille fixe.

Durant ce travail, nous obtenons la correspondance en conformant le modèle générique sur chaque maillage scanné. Evidement, il y a deux hypothèses sont faites: (1) la topologie du modèle (du modèle générique) est connue à priori, et (2) toute géométrie relative à un corps peu être obtenue en déformant le modèle générique. Dans le Chapitre 4, nous présentons notre algorithme d'optimisation rapide ajusté de manière à trouver une estimation du squelette ainsi qu'une déformation de la peau générique pour qu'elle s'ajuste sur le maillage scanné. L'idée est d'utiliser un ensemble de repères prédéfinis ou points caractéristiques pour mesurer la précision de l'ajustement et guider la déformation, de manière à assurer la correspondance

entre les points caractéristiques tout en minimisant l'énergie de déformation. Ces points caractéristiques pourraient soi être automatiquement détectés, soit désignés manuellement.

Tout au long de ce travail, la géométrie relative aux corps est censée être formée de deux entités distinctes: les composantes rigides et élastiques de la déformation. La déformation rigide est représentée par les paramètres des joints correspondants, qui, une fois appliquées au squelette générique détermineront l'approximation linéaire de la physique. La déformation élastique est essentiellement des déplacements de sommets, qui, une fois ajoutée à la surface générique après la déformation rigide, dépeignent les détails de la forme du corps.

L'ajustement du modèle générique sur le maillage scanné est composé de trois étapes successives:

- Etape 1: identification des points caractéristiques: un certain nombre de points caractéristiques ou repères, environ 30, sont étiquetés de manière semi-automatique.

- Etape 2. Ajustement du squelette: cette étape calcule l'approximation linéaire (posture et proportion) du modèle scanne en déformant le modèle générique vers les données scannées grâce à des déformations guidées par le squelette. En se basant sur les points caractéristiques, on trouve la configuration des articulations qui correspond le mieux aux position des points caractéristiques. On ramène ce problème à un problème de minimisation multidimensionnelle. Les variables de l'optimisation sont les degrés de liberté de chaque joint dans la hiérarchie du squelette. On trouve les valeurs de ces variables qui minimisent la distance des points caractéristiques correspondants.

- Etape 3. Ajustement précis de la peau: partant du modèle approximativement ajusté de l'étape précédente, cette étape déforme le maillage générique en calculant des vecteurs de déplacement pour chaque sommet. Le processus améliore itérativement la précision de l'ajustement en minimisant la différence de forme entre les modèles générique et scanne. La différence de forme est sauvegardée dans une carte des déplacements.

Les modèles d'exemple sont alors organisés de manière à servir d'entrée pour l'interpolateur de forme. En fait, chaque modèle d'exemple devrait être mesuré à divers endroits intéressants et être annotés.

C. Modeleur: Chapitre 5 et (Seo et Magnenat-Thalmann, 2003)

Une fois que l'on a fourni des exemples de modèles au système, on est en mesure de générer continûment toute une variété de formes. En bref, la sortie de ce module est le générateur qui est basé sur le mélange des exemples de corps.

Ayant à faire à des données de géométrie n-dimensionnelle et à un nombre d'exemple relativement petit, notre problème se ramène à un problème d'interpolation à partir d'un petit nombre d'échantillons. Le problème d'interpolation entre plusieurs variables se formule comme suit: étant donne N points distincts $\boldsymbol{p}_i$ dans $\boldsymbol{R}_n$ et N valeurs $\boldsymbol{v}_i$ dans $\boldsymbol{R}$, trouver une fonction f: $\boldsymbol{R}_n \rightarrow \boldsymbol{R}$ telle que pour tout i, $f(\boldsymbol{p}_i) = \boldsymbol{v}_i$ et telle que f n'oscille pas de façon parasite entre les valeurs.

Etant donné l'ensemble d'exemples obtenus des étapes précédentes, nous considérons que les paramètres de taille définissent un espace des paramètres dans lequel chaque mesure

définit un axe distinct. Chaque exemple sera donc représenté par un point dans l'espace des paramètres Le but est ici de produire pour chaque point $\boldsymbol{p}$ de l'espace des paramètres une nouvelle forme $\boldsymbol{X}(\boldsymbol{p})$ obtenue par l'interpolation entre les exemples. Quand $\boldsymbol{p}$ est confondu avec $\boldsymbol{p}_i$ pour un exemple de modèle i, alors $\boldsymbol{X}(\boldsymbol{p})$ doit être égale à $\boldsymbol{X}_i$. Entre les valeurs connues, une interpolation lisse et intuitive devrait être réalisée. On peut obtenir une interpolation lisse à travers l'espace des paramètres en résolvant un système linéaire pour certains paramètres Il y a deux types de générateurs: le générateur de joints s'occupe de chaque degré de liberté des joints, le générateur de déplacement s'occupe de trouver les déplacement de la peau pour des paramètres donnés.

L'utilisation des fonctions à base radiale de Gauss (GRBFs) est une approche courante au problème de l'interpolation de données éparpillées et peut également être utilisé ici. Considérons une correspondance entre un espace d-dimensionnel $\mathbf{x}$ vers un espace uni-dimensionnel. La sortie s($\mathbf{x}$) de la correspondance utilise une combinaison linéaire des fonctions de base:

$$s(\mathbf{x}) = \sum_{i}^{N} w_i \cdot \exp(-\beta \left\| \mathbf{x} - \mathbf{x}^i \right\|^2), \mathbf{x} \in R^d, w_i \in R,$$

Avec l'ensemble des exemples les N vecteurs en entrée x^i, β est le paramètre qui contrôle le lissage de la fonction d'interpolation, et $\|\cdot\|$ est la norme euclidienne. Comme on le verra dans la Section 4.2, le poids w_i peut être obtenu en résolvant le système linéaire associé.

Cependant, les techniques standard de RBF demanderaient trop de fonctions d'interpolation, à cause du grand nombre de dimensions du descripteur de forme. Nos modèles génériques sont constitués de 33 joints avec chacun 6 degrés de liberté (DoF : degree of freedom), et environ 900 sommets avec 3 DoF. Notre approche à ce problème est d'adopter une représentation des formes compacte et efficace en utilisant l'analyse en composantes principales ou PCA (Press et al, 1988), une des techniques courantes pour réduire le nombre de dimensions de données. En trouvant la base orthogonale appelée base propre, le vecteur de données original $\boldsymbol{x}$ de dimension n peut être représenté par sa propre projection sur les M (<< n) premiers vecteurs propres qui correspondent aux M plus grandes valeurs propres. Dans notre cas, vingt-cinq bases ont été utilisées à la fois pour les joints et pour les interpolateurs de déplacements.

D. Une évolution d'un individu: Chapitre 6 et (Seo et al, 2003)

Une des limitations principales du générateur de modèles est qu'un seul et unique modèle est produit étant donne un ensemble de paramètres Souvent, on veut commencer avec un individu particulier et appliquer certaines modifications tout en conservant d'autres caractéristiques de son physique. Ainsi nous devons considérer le cas ou l'on veut commencer avec un individu particulier et appliquer des changements selon certains attributs tout en conservant des propriétés identifiables. Un exemple typique des variations de la forme paramètré pourrait être: à quoi ressemblera la personne si elle perdait du poids ?

Nous traitons le problème comme décrit dans la Section 6.3. Nous construisons un modèle de régression, en utilisant les paramètres de forme comme évaluateurs et chaque composante du vecteur du corps comme des variables en sortie. Comme il est décrit dans la Section 6.1, nous avons choisi ce que nous appelons paramètres de contrôle de la forme (rapport entre le

diamètre de la taille et des hanches, pourcentage de graisse…) pour décrire et simuler les changements d'un individu particulier.

En général, il est difficile d'identifier qu'est ce qui est invariant dans le corps d'une personne à partir de changements provenant de divers facteurs (sport, régime, vieillissement, etc.) De plus, les exemples 3D de corps d'individus ayant subit divers changements dans leur apparence sont rares (bien que l'on puisse relativement aisément obtenir de tels exemples quand le but est de construire l'espace des paramètres de postures, comme dans (Allen et al, 2002). Ainsi, le problème de l'identification (a) des éléments géométriques qui caractérisent un individu et (b) les éléments contrôlables issus des exemples donnés doivent être résolus.

Pour quantifier la corrélation entre les paramètres de forme et la forme du corps, nous avons utilisé le modèle de régression suivant pour chaque élément du vecteur de corps (voir Section 3.2).

$$Y_i = \beta_i X + c_i, (i = 1,2,..30),$$

avec

W_m : la matrice d'échantillonnage du poids définie dans la Section 5.1

X : le paramètre de forme,

Y_i : le *i*-ieme élément du vecteur de corps, b_i ,

β_i : le coefficient de la fonction de régression Y_i ,

c_i : l'intersection de la fonction de régression Y_i .

La variable significative doit être suffisamment grande pour $p < 0.05$ et les modèles qui n'ont pas une corrélation significative ($p >= 0.05$) sont rejetés par le générateur

En partant maintenant d'un modèle pour lequel la i-ième valeur du vecteur de corps est $b_{i,src}$, et le paramètre de forme désiré est x_{trg} , nous pouvons dériver une nouvelle valeur du vecteur de corps $b_{i,trg}$ comme suit. Pour le paramètre de forme initiale x_{src} du modèle donné, la fonction de régression $E_i(x)$ donne la valeur moyenne $E_i(x_{src}) = \hat{b}_{i,src}$. La différence $e_{i,src} = b_{i,src} - \hat{b}_{i,src}$ est appelé résidus de la régression. Nous considérons ce résidu comme l'élément distinctif du corps, c'est à dire la déviation des composantes du vecteur de corps par rapport à sa valeur moyenne. De la même manière que (Kahler et al, 2002), nous faisons l'hypothèse que les composantes du corps conserve leur variance par rapport à la moyenne statistique tout au long des changements: De larges épaules resteront larges. Il est intéressant de noter que nous pouvons calculer le corps moyen pour un paramètre de forme donne x en calculant la fonction de régression $E_i(x)$ pour tous les b_i .

Ainsi, pour un vecteur de corps donné pour lequel nous connaissons le paramètre de forme, nous sommes en mesure de calculer la valeur résiduelle de chaque composante du vecteur de corps. Le nouveau vecteur de corps est alors obtenu en additionnant ces résidus à chacun des vecteurs des corps moyens pour le paramètre de forme désiré. Formellement, étant donné les

paramètres de forme actuels et désirés, x_{src} et x_{trg}, la nouvelle valeur du i-ième vecteur de corps $b_{i,trg}$ est donnée par:

$$b_{i,trg} = E_i(x_{trg}) + \left(b_{i,src} - E_i(x_{src})\right).$$

Nos exemples de modèles sont, cependant, relativement peu nombreux (n=57) et mal repartis. Travailler avec ces données qui ne sont pas distribuées équitablement pourrait influencer les coefficients du modèle de régression utilisé pour modifier la forme. Idéalement, la distribution du pourcentage de graisse ne devrait pas être corrélée avec la taille. C'est pourquoi nous effectuons la calibration des échantillons avant de créer le modèle de régression, comme décrit dans la Section 6.2.

E. Implémentation et résultats: Chapitrer 7

Nous avons utilisé le générateur proposé pour produire des modèles de corps humains de diverses dimensions. A partir de nouvelles mesures arbitraires qui ont été choisies par l'application, le vecteur entré au modeleur est déterminé, après normalisation. Avec ce vecteur entré, le modeleur explore rapidement l'espace des paramètres construits précédemment et évalue les formes interpolées correspondantes. Le modèle demeure animable à tout instant, durant la manipulation, à travers le recalcule de l'attachement de la peau.

En permettant la modification automatique à partir d'un ensemble de paramètres, notre générateur de modèles pourrait aboutir finalement à la génération automatique de population virtuelle diversifiée.

F. Conclusion

Le travail présenté dans cette thèse s'est concentré sur la modélisation de corps humains réalistes à partir de mesures. L'utilisation de maillages scannés de personnes réelles est indiscutablement une des meilleures ressources pour l'analyse statistique et la modélisation de formes de corps humains.

Notre approche est basée sur le fait que la taille et la forme du corps humain sont souvent corrélées. En utilisant une base de données de corps obtenus par scanner, notre modeleur permet d'estimer la forme et la taille du corps de façon réaliste, même dans le cas ou les données fournies au modeleur sont restreintes. En conséquence, l'utilisateur peut générer ou modifier un corps humain en entrant un faible nombre de paramètres.

En résumé, la thèse apporte les contributions suivantes:

- L'étape de préparation des modèles permet de convertir des maillages scannés en modèles qui peuvent être immédiatement animés. L'ajustement d'un modèle générique aux maillages scannés se fait en deux étapes. Dans la première étape, nous proposons un algorithme qui permet d'estimer automatiquement les proportions du corps par l'ajustement des dimensions des joints du squelette du modèle générique au maillage du corps scanné. L'ajustement du modèle générique est ensuite affiné avec une approche qui calcule le déplacement des points du maillage générique. Cette étape est basée sur un calcul de minimisation de l'erreur d'ajustement entre les deux maillages, celui du modèle générique et celui scanné.

- Cette thèse introduit aussi une représentation compacte de la géométrie des corps humains. Cette représentation permet une manipulation plus rapide et plus efficace de la géométrie des corps.

- Une méthode qui permet la manipulation automatique de la géométrie complète du corps à partir de données sur la taille a été décrite. Cette méthode utilise des exemples de corps scannés en 3D afin d'exploiter la corrélation entre la forme des corps et leur taille. Les deux interpolateurs, un pour les joints du squelette et l'autre pour le déplacement des points, sont construits en utilisant une méthode d'interpolation de données éparpillées sur les exemples préparés précédemment.

- Un des moyens naturels de manipulation de corps humains est la modification d'un corps existant. Ce type de manipulation a été rendu possible en analysant statistiquement les variations de la forme des corps scannés. En conséquence, notre modeleur permet changer les attributs d'un corps existant à l'aide de paramètres de contrôle.

- Notre travail de recherche est basé sur des paramètres anthropométriques qui sont cohérents avec l'étude de populations humaines. Il est donc facile d'étendre les fonctionnalités de notre modeleur pour générer de façon automatique des populations d'humains.

- Finalement, nous avons intégré notre contribution dans un cadre général de création et de modification de corps complets qui peut être utilisé pour diverses applications. Nos résultats indiquent qu'une combinaison adéquate de méthodes statistiques et géométriques apporte une vraie solution pour le développement d'un modeleur robuste et efficace qui peut être utilisé pour des objets de même classe que le corps humain.

Bien que les même techniques aient été utilisé pour des objets graphiques différents, en particulier les mouvements du corps, appliquer ces méthodes sur de nouvelles classes d'objets est un travail de recherche. Par exemple, même si la même technique peut être utilisée pour les objets graphiques différents, données de mouvements, par exemple, travailler avec d’autres types d’objet reste un travail non exploré. On peut appliquer les techniques basées sur des exemples aux données de mouvements du corps pour que la façon de marcher du model change aussi quand la taille des jambes ou le pourcentage de gras est modifié.

Nous pensons que notre travail peut être utile pour les études en biologie, anthropologie et en psychologie. En utilisant l'approche basée sur des exemples, présentée dans cette dissertation, il serait possible d'analyser comment le corps évolue en fonction de divers facteurs. A part l'application sur la modélisation des habits qui a été décrite dans cette thèse, d'autres applications sont possibles, par exemple l'étude de la croissance des enfants, le vieillissement ou les effets de l'alimentation et de la pratique du sport sur le corps humain.

Bien que la taille de la base de données de corps scannés soit suffisante pour nos analyses statistiques (La plupart des auteurs recommandent de travailler sur des échantillons qui sont de taille 10 à 20 fois plus grande que le nombre de variables approximés par régression linéaire), cette base est peut être encore plus étendu par l'ajout de corps scannés. Il est clair qu'une base de données plus grande permet une plus grande variété de corps générés et une plus grande flexibilité. Au fur et à mesure que la base de données est étendue, il se posera le problème du temps de calcule pour construire le modeleur. Bien que nous ayons calculé une

solution analytique pour construire l'interpolateur, d'autres méthodes comme les réseaux croissants de RBF auraient pu être utilisée (Esposito, 2000). Il est connu que cette méthode permet de travailler sur des bases de données très grandes parce qu'elle résout les poids et variances de chaque nœud du réseau de façon itérative au lieu de résoudre le system d'un bloc.

Notre méthode réutilise les données d'attachement pour la déformation de la peau du modèle générique. En conséquence, la déformation de la peau est contrôlée par les données d'attachement définies sur le modèle générique. Une amélioration possible pour un futur travail de recherche consisterait à créer un modèle qui contrôle en même temps les déformations statiques (apparence) et dynamiques (dépendantes de la posture du corps).

Les vecteurs représentant la géométrie du corps et leur distribution dépendent aussi des données initiales d'attachement de la peau au squelette. Quand la peau est attachée avec un poids fort aux avant-bras, une faible variation de la taille du joint donne une modification immédiate de la forme finale de la peau. Si le poids d'attachement est faible, l'obtention de la même forme finale requière une variation beaucoup plus important de la taille du joint. Un affinement automatique des données d'attachement de la peau afin d'obtenir une déformation de la peau optimale pourrait être un sujet de recherche. Par exemple, cet affinement pourrait calcule de façon à minimiser la variation des vecteurs représentant les corps.

Bien que nous aillions principalement travaillé sur les modifications de la taille et des paramètres de forme, il existe d'autres critères qui, nous pensons, valent la peine d'être expérimenté. Ces critères peuvent être par exemple, la pratique du sport, l'age et l'appartenance ethnique. La combinaison de ces paramètres avec les paramètres de la taille est certainement une suite possible de notre travail.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

"Men and women drifted as in a trance, with no control over the direction they take. This contradicted the notions of science and knowledge prevailing among scholars at the time, and was felt to be a grave affront; it was the University that had commissioned these early works from Klimt"

Gilles Néret

1.1 Importance and Background

Nowadays, highly realistic virtual human models are rapidly becoming commonplace in computer graphics. Recent years have witnessed believable (both in the appearance and the movement) human characters in games, 3D animation films, and television presentations. These models, often represented by complex shape and requiring labor-intensive process, challenge the problem of automatic modeling. The current modeling schemes, which demands many days or even months of interactive design work by skilled CG artists, could be replaced by more efficient ones, making the process almost automatically or even instantly.

This thesis addresses one of the most difficult tasks encountered by CG artists – human body modeling. More specifically, we focus on the parameterization techniques for modeling static body shape (the variety of human body shapes). A time-saving generation of realistic, controllable whole body model is the primary motivation behind this work. At the same time, we are looking for a systematic way of obtaining the diversity and individuality of physical appearances of bodies that are exhibited in real people.

In general, the modeling of body has received less attention than the modeling of face, and the issue of static (morphological) shaping than the issue of dynamic (skeleton-driven) shaping, in the area of the body modeling. There are several factors that make the modeling of realistic bodies so elusive. Firstly, the human body is extremely complex geometric form (see Figure 1-1). For example, there are more than 50 degrees of freedom in the underlying skeleton and minimum of two thousand polygons are common in real-time applications. Secondly, the bodies are articulated and highly mobile objects. Finally, our eyes are sensitive to the object we know so well – detail shape and movement of human bodies. We can easily identify unrealistic body shapes and/or movements, through the subconscious cognitive evaluation that we perform automatically. Yet, we are not always able to explain what is actually missing or wrong.

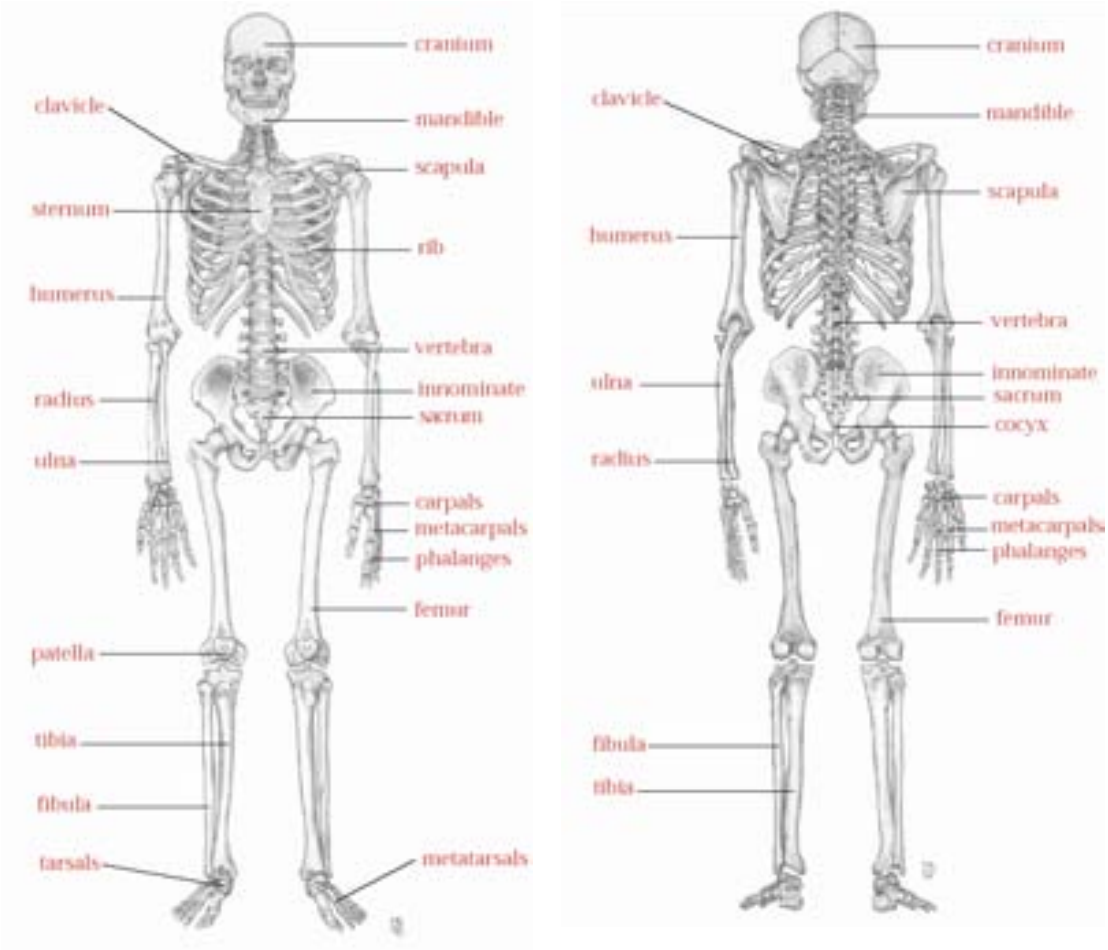


Figure 1-1. The human skeleton structure (http::Brown).

The interest of human body research is very broad. Notably, human body has been a long observed subject for various purposes; Anthropometrists, medical doctors and artists, have committed measurement, classification and comparative studies on diversity of sizes and shapes of bodies that are exhibited across the population (See Figure 1-2). Naturally, the topic of human body modeling exposes numerous opportunities for combining various research areas like biology, statistics, robotics, artificial intelligence, computational geometry, image processing, and psychology. In this thesis work, we primarily investigate appropriate combinations of statistics and geometry.

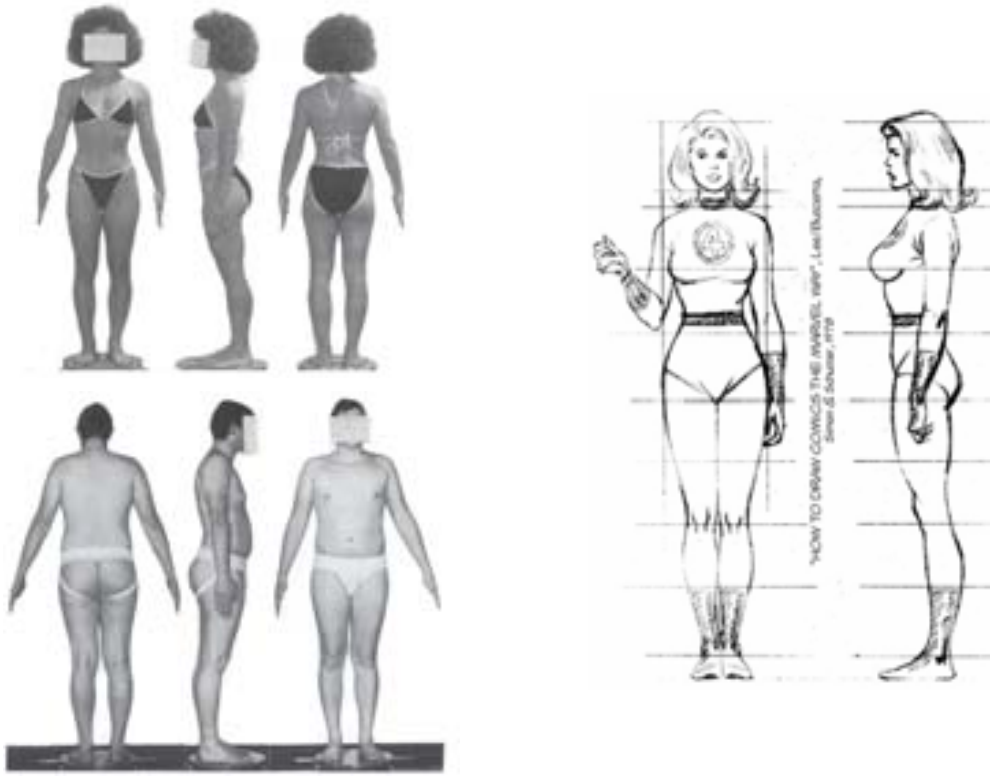


Figure 1-2. Observation of human body for various purposes. On the left, lightly clothed female and male are photographed for the somatotyping analysis (Heath and Carter, 1988). On the right, an ideal proportion of a female cartoon character is illustrated.

1.2 Problem Statement

Human body modeling and animation have been one of the most difficult tasks encountered by animators. In particular, realistic human body modeling requires an accurate geometric surface. Design tools available today allow the interactive design of human bodies either from scratch or by modifying existing model. They however require considerable user intervention and thus suffer from a relatively slow production time and a lack of efficient control facilities. Moreover, ensuring certain geometric constraints on the model is a difficult and painstaking task, even with the most sophisticated softwares.

On the other side of the computer-aided modeling, a variety of 3D scanning methodologies that are specially developed for human body are available, with which one can capture and digitize shapes and sizes of human bodies that exist in the real world. In particular, 3D range scanners are increasingly becoming available that allows dense, accurate measurement of human body data quickly and effectively. Unfortunately, they are restricted to recoding shapes that can be observed – there is no simple means of automatically modifying the shapes

once they have been created. Also, they require significant simplification and noise reduction before they can be converted into useful models for visualization or modeling. Worse, it is rather time consuming to animate them by finding accurate joint centers and establishing an appropriate bone-to-skin relationship.

There are various properties of a 3D geometry model that one may be interested in controlling or varying, including its size, shape quality, complexity, materials, etc. This thesis addresses the problem of manipulating size and shape constraints in the human body model. Automatically modifying shapes to meet new needs is desirable for at least two reasons. Firstly, it is often the case that we want to modify shapes to meet new needs or requirements. In a garment application, for example, we might want to create a 3D body model in a way that it satisfies a number of measurement constraints with minimum user intervention, and in an interactive runtime setting. Secondly, automatic modification makes it easy to avoid redundancy in a crowd. Without having to create or reconstruct each individual, one can obtain crowds by blending of existing models, for instance.

Example based techniques are good alternatives to overcome such limitations, since they allow to leverage the quality that exist in example models. In particular, those methods that extract useful statistics from the examples are promising because they can guide the estimation of uncertain information and thus add robustness to the modeler. Indeed, human modeling seems to be an area where statistics finds to be most useful (Blanz and Vetter, 1999) (DeCarlo et al, 1998). The use of examples however has been limited to facial models or only subparts of the body, perhaps primarily due to the complexity that is inherent to our bodies. These methods will be discussed in greater detail in Chapter 2.

To summarize, despite a variety of body modeling research work, previous methods have not succeeded to simultaneously offer the controllability and the quality shape that are pertaining to captured shapes, in the domain of whole-body modeling. By controllability we refer the modification of certain properties of the model (posture, size, and other body-related attributes) by the user to meet new requirements. The research outlined here examines the problem of time-saving generation of realistic, controllable human whole-body models, primarily for real-time applications. The specific focus of this work lies in the appropriate combination of statistics and geometry for estimating the most plausible physique from a few user-supplied parameters. Unlike previous anthropometric modelers, for instance, which typically involve intensive use of statistic information, we directly make use of 3D shape capture dataset from real people, rather than statistically analyzed form of anthropometric data that had been previously compiled through sizing survey. Such an approach allows to regulate subtle correlations between control parameters and the whole body geometry through manipulation.

1.3 Motivating Application

The necessity to model human bodies appears in a wide range of applications in computer animation and simulation. Graphical modeling of human bodies obviously plays an important role in the entertainment industry for character animation and simulations involving people. Other applications encompass several different domains – representative or autonomous agents in virtual environments, human factors analysis in ergonomics, training, education, and simulation-based design.

As have seen in previous section, our specific interest lies in the synthesis of realistic human body models that allows parameterized control. Given user-supplied parameters, the modeler generates the corresponding body almost automatically. Therefore, we limit our discussion to modeling with high level parameters.

1.3.1 Garment prototyping and retail

Figure 1-3. A garment designer works on real materials.

Consumers that purchase garments online today base their purchase and size-selection decision mostly on 2D photos of garments and sizing charts. This method is not precise enough and not interactive enough to provide right sizing as well as right choice to the consumer. If 3D body creation techniques are to be meaningful in this context, they must be able to produce correctly sized body models and be efficient enough to be used in an interactive runtime setting with minimum user intervention. Computer aided modeling of human body that satisfies such requirements is a difficult task, even with the most sophisticated software available. There is no simple means of automatically modifying existing models to adapt to specific measurement constraints once they have been created. One of the most challenging research areas in this context is in developing a robust methodology on automatically building realistic 3D body models that satisfy given size or measurement constraints in real-time performance.

The commence of the European project E-Tailor (http://www.atc.gr/e-tailor/) in the beginning of year 2000 laid the foundation of this work. The broad objective of that project has been to develop advanced infrastructures for virtual retailing services of customized clothing. To addresses problems in the garment retail, the fitting problem, the complexity of the selection process, and the personal data management, following tasks have been established:

- A European Sizing Infrastructure, to solve the sizing problem (non-uniformity of size designations)
- An advanced Customised Clothing Infrastructure enabling the production and distribution of custom-made garments at reasonable prices, in short time and with a close to perfect fit (intelligent pattern alteration software, order custering and CAD standards)
- An innovative Virtual Shopping Infrastructure enabling customers to visualise realistic models of themselves wearing clothes on offer at e-kiosks and Internet shops (Virtual Try-on)
- Generic body representations and standards (European Anthropometric Database)
- Smartcard applications, securing data privacy

- Integration of current state-of-the-art systems (3D scanners, MTM CAD, Virtual Shopping applications) in the form of across the value-added chain demonstrators.

The consortium is composed of 6 academic and 10 industry partners, among them are 3D range scanner companies Techmath and Telmat whose tasks are to provide the size data of the customers to the consortium.

Figure 1-4. Different persons wearing the same size 'L' garments. (Courtesy by Clive Arrowsmith)

Having worked on 3D clothes for more than ten years, MIRALab's (http://www.miralab.unige.ch) involvement in E-Tailor has been related to the development of advanced clothing simulation software and a web-site based virtual fitting room. Specifically, the new research objective of MIRALab has been the investigation of time-saving and quality-assuring solutions for the virtual fitting room. To successfully integrate the Virtual Shopping Service with 3D scanned individual body data, smart card media has been proposed as the SPDS (Secure Personal Data Storage) at an early stage of the project. Due to the limited size (23K) and data transfer speed constraints, it has been decided that only the representative measurements of the scanned data is stored. This implies that the Virtual Shopping Service should be able to regenerate full 3D geometry models from the given anthropometric measurement data, with seamless integration of other modules such as 3D garment simulation and character animation. In this thesis work, focus has been on the exploitation of scanned data for the measurement-driven, time-critical generation of realistic body models.

1.3.2 Populating virtual environments

The range of variation in their body size and shape heavily contributes to the diversity and individuality of the people we encounter in daily life. Simulating virtual environments populated with virtual but realistic crowds requires hundreds of different face and body geometries, maybe even a more distinct one for each person, than in real life. It is of monumental challenge to achieve such spectrum with modelling techniques. Often, it is desirable to obtain the specific population almost automatically, rather than by relying on CG artists for interactive, manual assistance.

1.3.3 Anthropometric modelling and ergonomics

For product designers to test their designs, models that are driven from the anthropometry play an important role. Timely detection of ergonomic deficiencies results in higher quality and prevents additional costs invoked by available corrections.

1.4 Objectives

1.4.1 Realistic models – Size vs. Shape

Arguably, anthropometric measurements allow the most complete control over the shape of the body but it would be almost impractical to provide all the measurements required to detail the model. Therefore it is not uncommon to use a subset of measurements. (Later in this document, we use 8 primary measurements to make 3D mannequin for the clothing application.) In such cases, the problem of estimating other measurements and determining appropriate shape of the full geometry need to be solved. To take an example, there exist infinite numbers of ways to determine the shape of a closed contour given its length constraint. Until now, the shape has been dealt with as a passive constitute – the final body shape is a result of a series of deformation such that it satisfies the given measurement. At times, they have tried to keep as much as possible the shape of the template model. In Section 4.3, we show how we exploit existing models to automatically derive realistic body shapes on the fly.

1.4.2 Fast generation

In various applications of anthropometric body modeling technique, one can not assume the potential users are 3D artist who are familiar with graphics packages. If 3D body creation techniques are to be meaningful in this context, they must be able to produce correctly sized body models and be efficient enough to be used in an interactive runtime setting with minimum user intervention. Thus, one key challenge is how to hide the complexities of the human body and to efficiently transform the information to the manipulation of body geometry.

1.4.3 Animatable models

A static body has limited sense. Many engineering application requires appropriately sized bodies to be at different postures or even animate. In order to obtain such animatable body models, the skin surface and the underlying skeleton structure often form an integrated part of the body modeling. This means the skeleton structure should remain at correct locations inside the skin at any time during the size-driven deformation.

1.5 Organization

This document proceeds as follows. The next chapter, Chapter 2, reviews previous three broad categories of research work on human body modeling. Also included is a summary of anthropometry and its application in computer graphics.

Our method is based on examples and we start in Chapter 3 with a description of the database of 3D body scans that our synthesizers are built on. Because we want to represent the body geometry as a vector in the body space, the topology of all the body meshes in the

example database should be identical. The procedures for preparing the example models are outlined in the same section.

There exist two different types of synthesizers: modeling synthesizers that create new body geometries using body measurements as input parameters, and modifier synthesizer that manipulate existing models. In Chapter 4, we introduce our modeling synthesizers and explain how we derive them by making use of the database. Chapter 5 gives a description of the modifier synthesizer. The main purpose of the modifier synthesizer is to obtain variation of the body geometry according to certain body attributes whilst keeping the distinctiveness of the individual as much as possible.

Once the synthesizers are constructed, new shapes are created simply by controlling the parameters, which derives the desired deformation on the reference model by evaluating the constructed synthesizers. After demonstrating the robustness of the approach in Chapter 6, we conclude the document in Chapter 7.

1.6 Summary

In this chapter we have addressed the importance, problem definition, motivation, and challenging issues. After a brief summary of the contribution of the work, the organization of the thesis is outlined.

Chapter 2

Previous work

"All Gaul is divided into three parts, one of which the Belgae inhabit, the Aquitani another, those who in their own language are called Celts, in our Gauls, the third. All these differ from each other in language, customs and laws. The river Garonne separates the Gauls from the Aquitani; the Marne and the Seine separate them from the Belgae."

Julius Caesar in Gallic War

The problem of building robust and accurate models to represent human bodies is certainly not new. We begin with a discussion of the relevant research work on human body modelling techniques, which broadly divide into interactive design methods, reconstructive approaches, and example based approaches. The next section briefly describes the first category. Another popular approach for body modelling is based on 3D scanning and reconstruction (Section 2.2). For the applications here, we use example models to exploit correlations. This is often called example-based techniques, and is the subject of Section 2.3.

These modelling techniques say little about the actual variability seen in human bodies and body models. This has been systematically studied in the field of anthropometry – the biological science of human body measurement. Section 2.4 provides a brief review of anthropometry and its application in the graphical modelling of human body.

2.1 Interactive design methods

A variety of human body models are available to represent and deform the human body shape, which broadly divide into surface model, multi-layer model, and anatomically based model. Magnenat-Thalmann and Thalmann (1987) used plaster modeling to create a polygonal surface model to animate Marilyn Monroe, also introduced the concept of Joint-dependent Local Deformation (JLD) operators (Magnenat-Thalmann and Thalmann, 1991) to deform the skin surface. Those operators were specific local deformations that depend on the nature of joints. Forsey (1991) extended the hierarchical B-spline technique to 3D character animation. A hierarchical surface was attached to an underlying skeleton in such a way that the figure designer had control over the location and scope of the surface deformation.

The multi-layered models contained the skeleton layer, intermediate layers to simulate the body volume such as muscles, fat, bones and so on. Chadwick et al (1989) proposed a layered technique based on Free-Form Deformation to apply muscle efforts to a skeleton. Shen and Thalmann (1995) proposed an effective multi-layered approach based on metaball for constructing and animating realistic human bodies.

More recently, multi-layered techniques were applied to anatomically-based models of humans and animals. They observe that the models should mimic actual components of the body and their models consist of multi-layers for simulating individual muscles, bones and tissues. Wilhelms (1997) proposed a multi-layered anatomically based model to simulate animals. She applied anatomical and physiological principles to model and animate animals.

In her model, an animal was defined as a structure of individual bones, muscles, and other generic tissue covered by a flexible skin. Nedel and Thalmann (1998a) proposed a method to simulate human beings based on anatomy concepts believing that the closer the model is to reality, the better will be the results. A mass-spring system was used to simulate the muscle deformation (Nedel and Thalmann, 1998b).

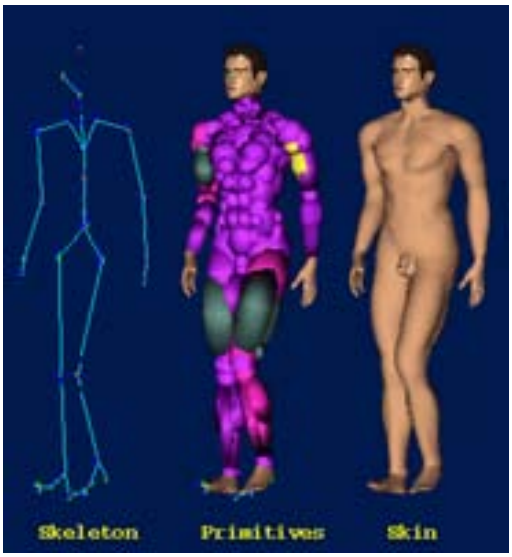


Figure 2-1. Multi-layered approach to generate a body model (Shen and Thalmann, 1995).

While allowing for an interactive design of human bodies either from scratch or by modifying existing model, most of them however require considerable user intervention and thus suffer from a relatively slow production time and a lack of efficient control facilities.

2.2 Reconstructive approaches

Lately, much work has been devoted to the reconstructive approach. One of the most important applications of surface reconstruction is 3D scanning – the measurement and modeling of shape and other visual properties.

2.2.1 3D scanning hardware

There are numerous methods for acquiring shape information. For instance, in computer vision, registration of landmarks in multiple views is used to infer object shape. In the manufacturing industries, mechanical touch probes mounted on coordinate measuring machines are used to record points on surfaces such as car bodies and airplane wings. The resulting measurements are very accurate, but the technique is extremely slow and limited to materials that can withstand mechanical contact.

3D body scanners divide roughly into active and passive. Passive scanners are based on photogrammetry and reconstruct the surface from a single or multiple(stereo) images or from a video recording of the object in relative motions. The 3D information of a point on the surface can be calculated by calculating the binocular disparities between corresponding points from the images captured by several pairs of cameras whose orientation and intrinsic parameters such as focus length and distortion parameters are calibrated. Most existing stereoscopic systems are not truly passive, as a texture must be projected on the object to facilitate the determination of the corresponding points.

The second category is laser range scanners. They illuminate the object with a laser beam, and measure distance using either triangulation, interference, or time-of-flight principles. An extensive survey of range imaging sensors can be found in (Besl, 1989). Range scanning systems typically produce range images – rectangular grids of distances from the sensor to the object being scanned. If the sensor and object are fixed, only objects that are "point viewable" can be fully digitized. More sophisticated systems, such as those produced by Cyberware Laboratory, Inc. (http::Cyberware) are capable of digitizing cylindrical objects by rotating either the sensor or the object. Laser range scanners are promising because they can provide dense, accurate range data at high bandwidths.

In Table 2-1, a list of current body scanning systems with comparisons between them has been compiled by Daanen (1998) and Carrere et al (2000).

Table 2-1. Comparison of Commercial 3D Body Scan Systems (Carrere et al, 2000)

Products	Operating System	Software
WB4 (Cyberware)	SGI/PC compatible	Cyscan
WBX (Cyberware)	SGI/PC compatible	Cyscan, CydirWB, DigiSize
BL Scanner (Hamamatsu)	Win32 PC/NT (98 95 2000)	BL Manager
Voxelan (Hamano)	Windows NT 4.0/Windows95	VOXELAN
3T6 ([TC]2)	Windows NT	BodyMeasurement System
2T4 ([TC]2)	Windows NT	
SYMCAD Turbo Flash/3D (Telmat Industrie)		SYMCAD, SYMCAD Body card
TriForm BodyScanner (Wicks & Wilson)	Windows NT	BodyScanner
RAMSIS (TecMath)	Pentium PC/minimum 100MHz processor	RAMSIS
Contour (TecMath)	Windows 95/Standard PC	Contour
Vitus Smart (Vitronic)	Intel PC	Vitronic
Vitus (Vitronic)		
FASTSCAN (Polhemus)	Windows NT/PC or Workstation	Included
Micro Scribe 3DLX (Immersion)	PC Windows/Mac/SGI	3D Digitizing
ModelMaker (3D scanner in London)	Windows NT 4.0	
LASS (Loughborough)	Windows NT	Real time shadow-scanning

2.2.2 Surface reconstruction from unorganized points

To fully realize the potential of 3D scanning, it is essential to develop general, automatic, efficient, and robust surface reconstruction algorithms for converting the data points that 3D scanners produce into useful models. Perhaps the most well-known algorithm for converting unorganized point sets to B-Splines is the one developed by Hoppe (1994). Unlikely to others (Section 2.2.3), which have typically been designed to exploit additional knowledge in specific problem instances, Hoppe poses a unifying general problem from unorganized points. It is a very elegant technique that requires no assumptions about the objects' topology and can

also provide optimal results, in terms of minimization of the error between data and reconstructed surface.

Another active area is so-called surface decimation or simplification. In order to achieve computational efficiency as well as satisfactory visualization of human body models, it is important to consider working with multiple levels of detail. Schroeder (1992) has presented one of the first efficient algorithms for reducing the complexity of triangle meshes. This has been improved by Hoppe (1993) and further optimized by Heckbert et al. (1994), Cohen (1996) and Hoppe (1996). A recent algorithms have also been proposed by Seo et al. (2000) and Oliveira et al. (2001), which have been developed specifically with human face and body models in mind respectively. The level of detail of the model, when it is applied to human body model, can be truly optimal by considering the geometry in conjunction with the animation. For instance, a human torso requires less detail than the head/face area. Also other attributes of the model, such as the complexity of joint hierarchy, will depend on the levels of detail. It is therefore necessary to determine, with respect to particular application domains, what kind of detail can be considered sufficient.

Figure 2-2. Surface reconstruction from unorganized points (Hoppe, 1994).

2.2.3 Algorithms assuming or exploiting additional knowledge

The problem of surface reconstruction can be made easier if assumptions can be made or additional knowledge can be exploited on the shape of the object being represented or the structure in the data. A common restriction of surface reconstruction method is that they assume that the topological type of the surface is known a priori. Function reconstruction techniques, those who calculate the best approximating function to fit the surface fall into this category. The goal of function reconstruction can be stated as follows: Given a surface ***D***, a set $\{x_i \in D\}$ and a set $\{y_i \in R\}$, determine a function $\{f : D \to R\}$, such that $\{f(x_i) \approx y_i\}$. Radial basis function approach (Powell, 1987) introduces a set of N basis functions and the function is taken to be a linear combination of the basis functions

$$f(x) = \sum \omega_i \cdot \Phi(\|x - x_i\|).$$

A rich set of algorithms were devoted to the efficient calculation of and basis function optimization. An in-depth discussion and applications of it in neural network field can be found in (Bishop, 1995).

In computer graphics, RBF (Radial Basis Function) has initially applied to image warping by Ruprecht and Muller (1995) and Arad et al (1994), and has recently also been applied to 3D geometry (Turk and O'Brien, 1999)(Carr et al, 1997) and motion interpolation (Rose et al, 1998). Carr et al (1997) have demonstrated the use of RBF to solve the ubiquitous problem of interpolating incomplete meshes (hole-filling) from dense point-clouds derived from 3D range scanners. More recently, they have shown in (Carr et al, 2001) the use of implicit function with RBF for the representation of object surfaces as a unified framework for the problem of

interpolating incomplete meshes and smoothing/remeshing noisy surfaces. They also propose a method for fast fitting and evaluation methods to make it feasible to use RBF for a large data sets and complicated objects.

Figure 2-3. Surface fitting by function reconstruction (Carr et al, 2001).

Another approach in surface reconstruction is to exploit structure in the data. For instance, algorithms solving the surface from contours data make heavy use of the fact that the data points are organized into contours, and that the contours lie in parallel planes. In many medical studies it is common to slice biological specimens into thin layers with a microtome. Similarly, algorithms to reconstruct surfaces from multiple range images typically exploit the adjacency relationship of the data within each range image. Merriam (1992) suggests two methods for merging range images: a virtual milling technique that intersects polyhedra constructed from the different range images, and a pruning technique that first constructs the 3D Delaunay Triangulation of the points and then prunes away tetrahedral “exposed” in the various range images. Turk and Levoy (1994) describe a mesh zipping approach, in which overlapping surfaces are “stitched” together.

It is only recently that the development of technologies especially for human body modeling has become a popular area. To recover the degrees of freedom associated with the shape and motion of a moving human body, most of the exiting approaches introduce simplifications by using a model-based approach. Kakadiaris et al, in (Kakadiaris et al, 1995), use 2D images from three mutually orthogonal views to fit a deformable model to approximate the different body size of subjects. The model then can be segmented to different body parts as the subject moves. Plaenkers et al (1999) also use video cameras with stereo pair for the model acquisition of body part. A person’s movements such as walking or raising arms are recorded to several video sequences and the program automatically extracts range information and tracks outline of body. The problem to be solved is twofold: First robustly extract silhouette information from the images; second fit the reference models to the extracted information. The data was used to instantiated the models and the models, augmented by our knowledge about human body and its possible range of motions, are in turn used to constrain the feature extraction. They focus however more on the tracking of movement and the extraction of a subject’s model is considered as the initial part of a tracking process.

Recently, more sophisticated models were introduced with and their aims limited to the construction of realistic human model. Recent work of Hilton et al (1999) involves the extraction of body silhouettes from a number of 2D views (front, side and back) and the subsequent deformation of a 3D template to fit the silhouettes. The 3D views are then mapped as texture onto the deformed model to enhance realism. Similarly, Lee et al (2000) proposed a feature based approach where silhouette information from three orthogonal images is used to deform a generic model to produce personalized animatable model.

Figure 2-4. Human model reconstruction from images from orthogonal views. (Hilton et al, 1999)

Figure 2-5. Human model reconstruction from images of orthogonal views. (Lee et al, 2000)

Based on adding details or features to an existing generic model, these approaches concern mainly the individualized shape and visual realism using a high quality textures. While they are effective and visually convincing in the cloning aspect, these approaches hardly give any control to the user; i.e., it is very difficult to modify these meshes to a different shape as the user intends. These approaches have the drawback that they must deal with special cases using ad hoc techniques.

2.3 Example based approaches

A considerable amount of work has been undertaken with respect to editing existing animations and blending between more than two examples. Interpolation provides a way to

leverage existing models to generate new ones. In particular, the radial basis function (Section 4.2) has been used as one of the primary tools for scattered data interpolation in computer graphics and CAD. Rose et al.'s paper "Verbs and Adverbs"(Rose et al, 1998) has shown results in this area using radial basis functions. Each example of motion is manually annotated with a set of adverb values, such as happy, angry, tired, etc. After normalization, each annotated motion is used to form an adverb and verb spaces using scattered data interpolation. Once the continuous range spaces are formulated, at any point in the adverb space, a corresponding motion is derived through RBF interpolation of example motions.

Sloan et al (2000, 2001) have shown the application of similar technique in face model generation. Using a number of example face models obtained from image based capture, they have shown interactive blending results with control parameters such as gender and age. Their contribution lies in that they make use of equivalent of cardinal basis function. The blending functions are obtained by solving the linear system per example rather than per degree of freedom, which potentially is of a large number, thus resulting in an improved performance.

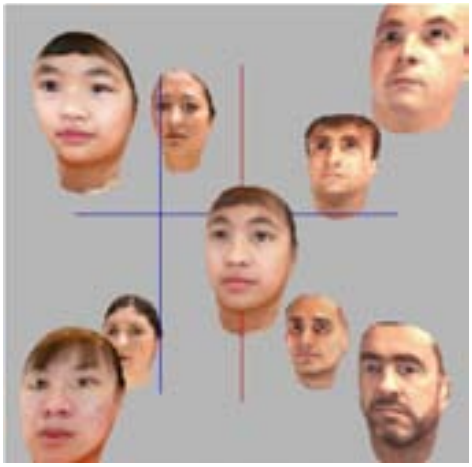


Figure 2-6. Example based skin deformation by Sloan et al.

More recently, novel interpolation methods that start with range scan data and use data interpolation to generate controllable diversity of appearance in human face and body models have been introduced. In works with similar goals but applied to face models, Blanz and Vetter (1999) have introduced a 'morphable model' for manipulating an existing model according to changes in certain facial attributes such as age, gender, hooked versus concave noses. New faces are modeled by forming linear combinations of the prototypes that have been collected from 200 scanned face models. They first compute the correspondence between individual models using optic flow. Manual assign of attributes are then used to define shape and texture vectors that, when added to or subtracted from a face, will manipulate a specific attribute. By making use of principal component analysis, they project the vector representing a face model into an eigenspace to reduce the dimension of the data. PCA describes a face shape as a weighted sum of an orthogonal basis of 3D shapes (called principal components). This basis is constructed from a large bank of examples that have been placed in mutual correspondence. They also have shown that is it possible to deform the 3D template of a human face in order to reconstruct a 3D face from a single 2D photo. Statistical estimation is used to recover missing information from the input image.

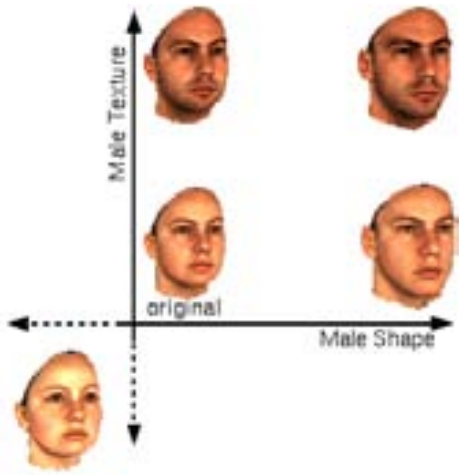


Figure 2-7. Interpolated shapes (Blanz and Vetter, 1999)

For the application here, we use multidimensional interpolation to obtain shape information from measurement, which extracts information from given set of samples. Improvements to this are to simply exploit the size and shape correlation expected in most situations. Of course, this is based on an assumption that there exist relations among measurements and between measurements and shapes.

2.4 Anthropometric human models

2.4.1 Anthropometry

Anthropometry means, literally, the measurement of people (Norgan, 1994). It has come however to be used in a more restrictive sense to mean the comparative study of sizes and proportions of the human body. In this dissertation, our focus is more on sizes, say height and/or hip girth, than on physical performance or composition of the body.

Anthropometry, the biological science of human body measurement, systematically studies human variability in faces and bodies. The procedures for measurement in anthropometry are precisely specified, allowing data between individuals to be successfully compared, and for useful statistics of population groups to be derived. When applied in graphics, it allows for not only the automatic generation of varied human body geometries but also more likely occurring individuals. Later in this document, our approaches rely on this large body of existing data that describes the statistical information on measurement of people's bodies.

The description of the human form by proportions goes back to da Vinci, a Renaissance artist, who created the drawing of the Viturvian Man based on the "ideal proportions". Anthropometrists have found that proportions give useful information about the correlations between features and can serve as more reliable indicators of group membership than can simple measurements.

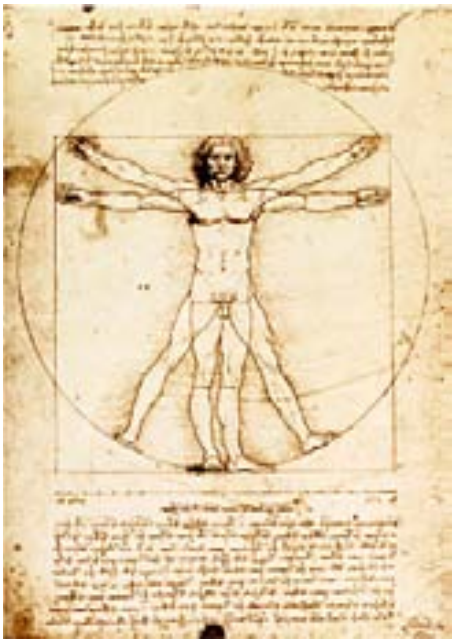
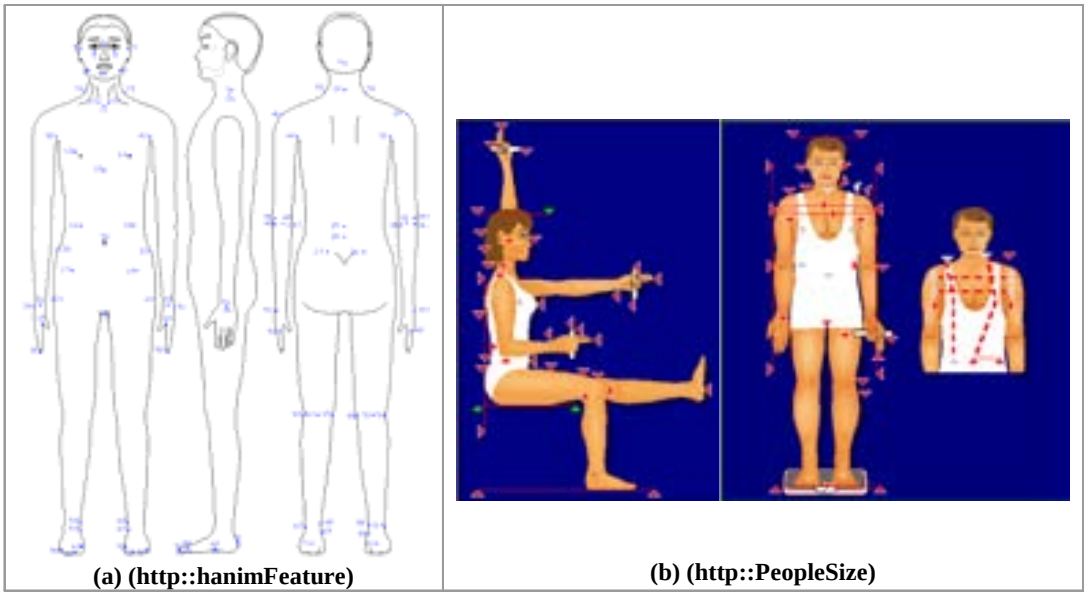


Figure 2-8. Leonardo da Vinci's Vitruvian[♣] Man (DaVinci, 1972)

2.4.1.1 Measurement definition and data collection



(a) (http::hanimFeature) **(b) (http::PeopleSize)**

Figure 2-9. (a) Anthropometric landmarks in H-Anim specification and (b) some measurements definition in commercial software.

[♣] "Vitruvian" refers to Vitruvius, the Roman Architect from whom Da Vinci derived his drawing from.

Anthropometric evaluation begins with the identification of particular locations on a subject, called landmark points. Anatomical landmarks are features (usually points or curves) on the surface of a human body that indicate the location of a body organ or component. These features do not necessarily have a robust definition in terms of differential geometry. However, their definitions are more intuitive and, since they are anatomically meaningful, they can easily be located by a human expert. Such landmarks are typically used in the clothing industry, but also in anatomy and anthropology. Typical examples include: Nape of neck (7th cervical vertebra), top of head, iliac crests, underbust and mid-clavicle point.

A series of measurements between these landmarks is then taken using carefully specified procedures and measuring instruments such as calipers and measuring tape. Typically the measurements are taken in scientific surveys, and involve careful positioning of people before measurements are taken. This is to enable precise definition of what was measured. Figure 2-9 on the left illustrates the anthropometric landmarks used in CAESAR project (Burnsides et al, 2001), which then was adapted to 'feature points' in H-Anim (http://hanimFeature) standard. Important landmarks and points defined by Carrere et al (2000) are summarized in Figure 2-10.

2.4.1.2 Computer graphics for modern anthropometry

Much of the anthropometric data currently in general use is derived from data acquired by manual measurement of a sample of the population nearly 50 years ago. Improvements in diet and healthcare in recent years have resulted in an increase in the average size of the population, thus making the 50-year-old data redundant. Recent surveys conducted by the British and US Army and others have endeavored to remedy this situation by utilizing 3D scanners for acquiring dense, accurate 3-dimensional human body data quickly and effectively. Manual techniques, which involve the use of callipers and measuring tapes, are extremely slow and costly, produce sparse data and are almost impossible to automate. Since the advent of 3D image capture technology, there has been a great deal of interest in the application of this technology to the measurement of the human body and the goal is now to develop a 3D image capture system capable of routinely providing dense, accurate anthropometric data (Daanen and van de Water, 1998).

Landmark	Definition
Abdominal Extension (Front High-Hip)	Viewed from the side, it is the measure of the greatest protrusion from one imaginary side seam to the other imaginary side seam usually taken at the high hip level (ASTM, 1999); taken approximately 3 inches below the waist, parallel to the floor (ASTM, 1995)
Acromion (Shoulder Point)	The most prominent point on the upper edge of the acromial process of the shoulder blade (scapula)[T] as determined by palpitation (feeling) (Jones, 1929; McConville, 1979).
Ankle (Malleolus)	The joint between the foot and lower leg; the projection of the end of the major bones of the lower leg, fibula and tibia, that is prominent, taken at the minimum circumference (McConville, 1979; O'Brien & Sheldon, 1941; ASTM, 1999).
Armpit (Axilla)	Points at the lower (inferior) edge determined by placing a straight edge horizontally and as high as possible into the armpit without compressing the skin and marking the front and rear points or the hollow part under the arm at the shoulder (McConville, 1979; ASTM, 1999). *See Scye.
Bicep Point	Point of maximum protrusion of the bicep muscle, the brachii, as viewed when elbow is flexed 90 degrees, fist clenched and bicep strongly contracted (Gordon, Churchhill, Clauser, Bradtmiller, McConville, Tebbetts, & Walker, 1989; ASTM, 1999).
Bust Point	Most prominent protrusion of the bra cup (Gordon, et.al, 1989, McConville, 1979; O'Brien & Sheldon, 1941); apex of the breast (ASTM, 1999).
Cervicale (Vertebra Prominous)	At the base of the neck [R] portion of the spine and located at the tip of the spinous process of the 7th cervical vertebra determined by palpitation, often found by bending the neck or head forward (McConville, 1979; Jones, 1929; Gordon, et.al, 1989; O'Brien & Sheldon, 1941; ASTM, 1999).
Elbow (Olecranon)	When arm is bent, the farthermost (lateral) point of the olecranon which is the projection of the end of the inner most bone in the lower arm (ulna) (O'Brien & Sheldon, 1941); the joint between the upper and lower arm (ASTM, 1999).
Gluteal Furrow Point	The crease formed at the juncture of the thigh and buttock (McConville, 1979; Gordon, et. Al, 1989).
Hip Bone (Greater Trochanter)	Outer bony prominence of the upper end of the thigh bone (femur) (ASTM, 1999; O'Brien & Sheldon, 1941).
Iliocristale	Highest palpable point of the iliac crest of the pelvis, ½ the distance between the front (anterior) and back (posterior) upper (superior) iliac spine (Gordon, et.al, 1989; Jones, 1929).
Kneecap	Upper and lower borders of the kneecap (patella) located by palpitation (Gordon, et.al, 1989; McConville, 1979); joint between the upper and lower leg (ASTM, 1999).
Neck	Front (anterior) and side (lateral) points at the base of the neck; points on each cervical and upper borders of neck ends of right and left clavicles [J] (O'Brien & Sheldon, 1941; Gordon, et.al, 1989).
Infrathyroid (Adam's Apple)	The bottom (inferior), most prominent point in the middle of the thyroid cartilage found in the center front of the neck (Gordon, et.al, 1989).
Shoulder Blade (Scapula)	Large, triangular, flat bones situated in the back part of the chest (thorax) between the 2nd and 7th ribs (Tolbra, 1986; Bryan, Davies, & Middlemiss, 1996).
Scye	Points at the folds of the juncture of the upperarm and torso associated with a set-in sleeve of a garment (Gordon, et.al, 1989; McConville, 1979; O'Brien & Sheldon,1941). *See Armpit.
Top of the Breastbone (Suprasternal)	Bottom most (inferior) point of the jugular notch of the breastbone (sternum) (Gordon, et. al, 1989; Jones, 1929).
Tenth Rib	Lower edge point of the lowest rib at the bottom of the rib cage (Gordon, et. al, 1989; O'Brien & Sheldon, 1941).
7th Thoracic Vertebra	The 7th vertebra of 12 of the thoracic type which covers from neck to lower back (Tolbra, 1986).

Figure 2-10. Important landmark terms and definitions (Carrere et al, 2000).

Accordingly, recent algorithms include automatic processing of scan data to extract features or measurements. Nurre (Nurre, 1997) presents an incremental approach that progressively refines the identification of data points. The first phase of identification is to orient and segment the human body data points. Algorithms for these tasks are presented,

including the ‘discrete point cusp detector’, with a description of their use. In order to show the robustness of the software, it has been tested on twenty different body scan data sets. Works by Dekker (Dekker, 2000) on the identification indicates that there are robust ways to find about 100 anatomical landmarks on a human body, at least for specific categories. A partially automatic landmark detection method was used in CAESAR (Civilian American and European Surface Anthropometry Resource) project (Burnsides et al, 2001).

Figure 2-11. Digitized and skinned scan subject, with anthropometric landmarks (Dekker et al, 1999)

Gu et al. (Gu et al, 1998) concentrate on human body modeling for the purpose of measurement of human body for the clothes industry. They use a hexagonal supporting framework to form a closed space for imaging with 12 cameras for upper and lower body parts in six views. Additionally a slide projector with a grid pattern is used to catch the chest area using the stereo pair of the intersections of horizontal and vertical lines.

Meunier and Yin (Meunier and Yin, 2000) proposes two-dimensional, image-based anthropometric measurement systems that offer an interesting alternative to traditional and three-dimensional methods in applications such as clothing sizing. Their attractiveness lies in their low cost and the speed with which they can measure size and determine the best-fitting garment.

In the market, there are now also available some systems that are optimized either for extracting accurate measurements from parts of the body, or for realistic visualization for use in games, virtual environments and, lately, e-commerce applications. Cyberware Inc.’s DigiSize was ‘partially developed in a joint government project to improve and automate fitting and issuing of military clothing’. They claim they offer complete and state-of-the-art solutions to age-old tape measurements and trial-and-error fitting problems (http::Cyberware).

Figure 2-12. Automatic measurement extractions from the scan data (http::Cyberware)

Several approaches are under active development to endow semantic structure to the scan data. Dekker et al. (Dekker et al, 1999) have used a series of meaningful anatomical assumptions in order to optimize, clean and segment data from a Hamamatsu whole body range scanner in order to generate quadmesh representations of human bodies and build applications for the clothing industry. Ju and others (Ju et al, 2000) (Ju and Siebert, 2001) introduce methods to automatically segment the scan model to conform it to an animatable model.

Figure 2-13. Automatic segmentation and animation of a scan (Ju and Siebert, 2001)

2.4.1.3 Statistical issues

Systematic collection of anthropometric measurements has made possible a variety of statistical investigations of groups of subjects. Subjects are often grouped on the basis of gender, race, age, “attractiveness” or the presence of a physical syndrome. Means and variances for the measurements within a group, as can be found in (http::PeopleSize), effectively provide a set of measurements, which captures virtually all of the variation that can occur in the group.

Most of the anthropometric data available for us was with percentiles, even though more desirable statistical methods such as regression analysis, multivariate analysis were addressed by some of the recent studies (http::Cardlab). A percentile is a very simple statistic. It shows a relative ranking of a given individual for a single measurement and is expressed in terms of the percentage of people who are smaller than that individual for that measure. For example, the distribution for the body dimension, Stature, in the 1967 anthropometric survey of USAF personnel is shown in Figure 2-14. The fifth percentile value indicated on the figure is 65.8 inches. This means simply that five percent of this population is shorter than 65.8 inches, and 95 percent of the same population is taller.

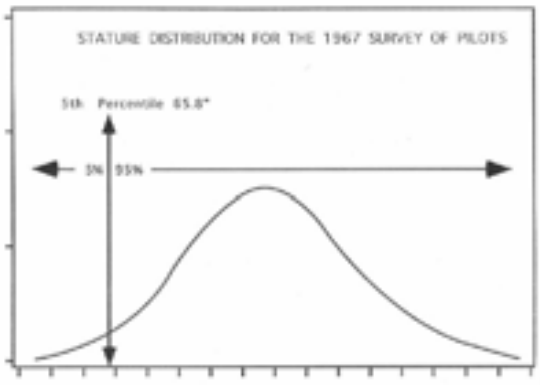


Figure 2-14. Measurement of stature represented in percentile

Limitations of the percentile approach are immediately apparent - First, percentiles are only relevant for one dimension (univariate), and second, they are specific to the population for which they are calculated.

2.4.1.4 Application

Anthropometric data is used in many areas of manufacture to provide information for the design of products such as clothing, footwear, safety equipment, furniture, vehicles and any other objects with which people interact. In the armed services this data is of particular importance as the survival of the serviceman may depend upon it. For example, in the accurate shape of cloth fitting protection equipment such as body armour or combat helmets, or in the optimum ergonomic design of combat vehicles.

2.4.2 Anthropometric human models in CG

2.4.2.1 Rigid scaling

Since anthropometry was first introduced in Computer Graphics by Dooley (1982), a number of researchers, including Azuola et al (1994) and Grosso et al (1989), have investigated the application of anthropometric data in creation of virtual humans. Spreadsheet Anthropometry Scaling System (SASS) presented by Azula et al (1994), enables the user to generate a variety of accurately scaled human models that can be manipulated in their animation system 'Jack'. The system creates a standardized human model based on a given statistically processed population data or alternatively, a given person's dimension can be directly used in the creation of a virtual human model. In the former case, it generates dimensions of each segment of a human figure based upon population data supplied as input. Their initial virtual human was composed of thirty-one segments, of which twenty-four had a geometrical representation. For each segment or body structure with geometrical representation, three measurements were considered, namely the segment length, width, and depth or thickness. Measurements were compiled from the NASA Man-Systems Integration Manual (NASA, 1987) and the Anthropometry Source Book (NASA, 1978).

Apart from these size measurements, they also explored kinematical properties of human body such as center of mass, joint limits, and strength. They allow the user to select the desired anthropometric group to browse or modify. Despite for their innovative efforts, however, the system was limited in several points. First, they are based on percentile. As it has been reviewed earlier in Section 2.4.1, a percentile is a very simple statistic and has undesirable properties in many cases. Secondly, although the system may be grounded on

accurate size information, it could not capture the true variety that is exhibited in detail shape of the body, due to the simplistic geometric representation adopted – the desired size was implemented by applying scale transformation on each segment and physical characteristics, although they later on showed the extension of the system equipped with partially deformable models (Azuola et al, 1994).

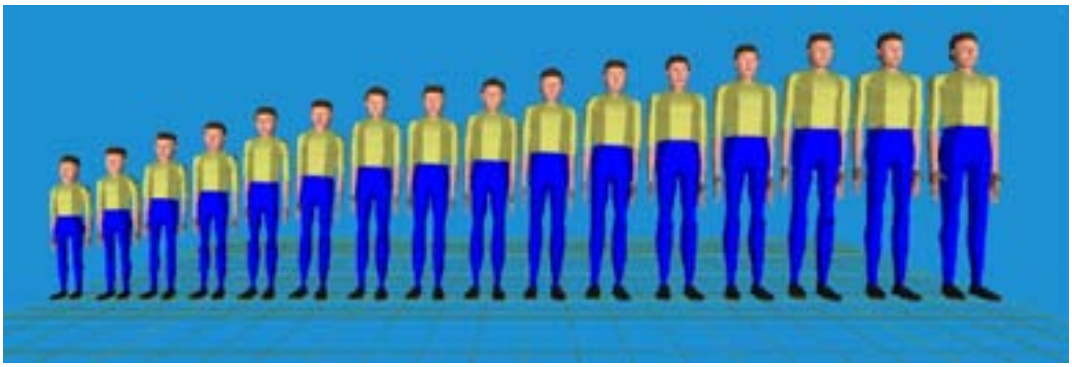


Figure 2-15. Anthropometric models for 'Jack' (Azuola et al, 1994).

2.4.2.2 Variational modeling

DeCarlo et al (1998) use a limited set of measurements and proportions between a set of facial landmarks. Given base measurements, they produce a plausible set of constraints on the geometry using anthropometric statistics. The idea behind is to generate a shape that shares the important properties of a typical face as far as possible and yet still respecting a given set of anthropometric measurement. They cast the problem as a constrained optimization one: Anthropometric measurements are treated as constrains, and the remainder of the face is determined by optimizing a surface objective function. A variety of faces are then generated for a population through random generation of face measurements according to anthropometric statistics.

This is an interesting approach that, unfortunately, is slow in creation time (approximately one minute per each face) owing to the nature of variation modeling. Also, the shape remains as a passive constitute as the prototype shape is conformed to satisfy the measurement constraints while 'fairness', i.e. smoothness of the shape is being maximized. Therefore, every desirable facial feature has to be explicitly specified as a constraint in order to obtain realistic shape in the resulting model that are observable in real faces, such as hooked nose or double chin. Later in this document, we adopt active shapening instead, wherein shape information is determined in relation with the given size and fed to the system to deform the mesh.

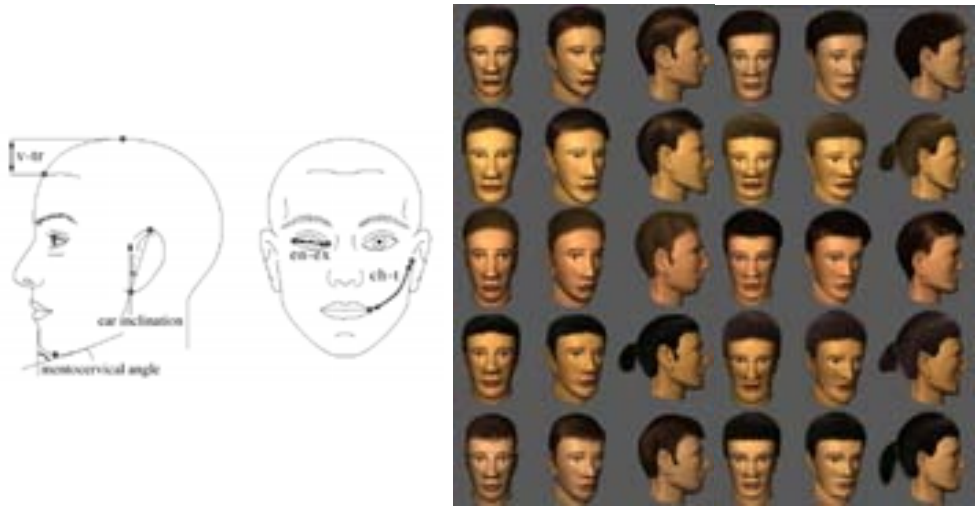


Figure 2-16. Facial landmarks and automatically generated face models (DeCarlo et al, 1998)

2.4.2.3 Geometric methods

The problem of correctly sized body models can be implemented by geometric deformation methods. These methods often come up with user interfaces with which the user can interactively manipulate the predefined parts of the given template models until s/he obtains the desired size. The modeling therefore consists of a series of "modify and re-measure" process and it is up to the user to appropriately manipulate the given mesh to obtain a good shape (Seo et al, 2000). Such an approach can often be found in commercial softwares such as "C-Me" from Browzwear (http::Browzwear).

Figure 2-17. The differently sized body models using geometric methods (http::Browzwear).

While it is relatively simple and straightforward in implementation, there remain however a number of drawbacks with this approach. When the modification shapes the desired model based on interpolation of differently sized body models, the result shape highly relies on the CG artist's skill – As is the case with many commercial softwares, interpolant models are mostly created from artists' interactive work. Or, in case it is based on geometric deformation

as in Seo et al (2000), it requires careful internal settings such as control point location/displacement to acquire acceptable shape. Often, they are obtained in a trial-and-error basis.

2.5 Conclusion

In this chapter we have reviewed the state of the art in body modeling. We considered modelers that rely on interactive design, reconstructive approaches, example based approaches, and the anthropometric human modeling methods. Table 2-2 summarizes each category along with its controllability, generation time and the level of realism.

Table 2-2. Comparison of various body modelers

	Interactive design	Reconstructive approaches	Example-based approaches	Anthropometric modeling methods
Unit of control (Controllability)	geometric primitives	N/A	high level parameters	size parameters
Generation time (Runtime)	hours ~ days	minutes	seconds	seconds ~ minutes
Realism	depends on the artist's skill	high	depends on example models	low ~ mid

Chapter 3

Modeler Design

"Those who are skilled in combat do not become angered, those who are skilled at winning do not become afraid. Thus the wise win before they fight, while the ignorant fight to win."

Zhuge Liang, 3rd century C.E.

As seen in the previous chapter, existing body modelers have typically been designed either to capture existing shape or to offer the user with sculpting tools. The use of examples had been limited to facial models or only subparts of the body, perhaps primarily due to the complexity that is inherent to our bodies. Previous anthropometric body modelers do not seem to have fully exploited both geometric and statistic methods; mere geometric consistency of measurements does not guarantee a realistic appearance of the resulting model, although the provided measurements may be statistically consistent.

In contrast, our approach attempts to simultaneously offer the controllability and the quality shape that are pertaining to captured shapes. In addition, we aim to go beyond the time-saving generation of whole-body models and consider time-critical generation, with real-time applications in mind.

To be able to meet these requirements, we consider a major challenge lies in the appropriate combination of statistics and geometry for estimating the most plausible physique from a few user-supplied parameters. Specifically, we have focused on parameterized human body modeling scheme through which the whole static body space is efficiently explored in statistically meaningful directions. Our approach is perhaps most comparable to previous anthropometric modelers. While they typically involve intensive use of anthropometric data in statistically analyzed form that has been previously compiled through sizing survey, we instead directly make use of 3D shape capture dataset from real people. We strongly believe that such an approach allows to regulate subtle correlations between control parameters and the whole body geometry under manipulation.

3.1 Overview of the method

One powerful method that has been widely accepted in biology and morphometry to analyze the correlation and variation between shapes of the same class is to consider the transformations required to deform one shape into the other. In this way, it is possible to create a continuous family of shapes with desired features. If a set of shapes with distinct formal characteristics is defined, we can generate a shape that blends these characteristics in any proportion using interpolation. Such shape transformations can be used in various ways for modeling and estimation purposes and is at the center of this thesis. Through creating parameterized deformation models, we show how a modeler can exploit existing data and knowledge about body shape and size.

In order to gain insight into the variety (and commonality) of physical appearance, we base our modeler on the statistic analysis of the captured 3D body geometry of real people, which arguably provides the best available resource to model and estimate correlations between parameters and the shape. To establish the geometric correspondence among these example geometries, an optimization-based fitting method is proposed (Chapter 4), which forms the first component of this thesis work. By finding the appropriate (error and energy minimizing) transformation for a template model to fit onto each example, each model is considered to share the same number of vertices and connectivity in the mesh as well as the same joint hierarchy.

Given the prepared 'example' models, the mapping function that we termed as modeling synthesizer is constructed, which smoothly transforms the control parameter space onto the whole body geometry space (Chapter 5). The measured sizes and shapes of real people in the database are used as valuable resources to reason about the most plausible body geometry given the control parameter set. Since our framework can be casted as an interpolation problem, various algorithms such as radial basis function can be employed. However, the standard RBF techniques would require too many interpolation functions, due to the high dimension of the shape descriptor. Our approach to this problem is to adopt a compact and efficient shape representation by finding the basis. The PCA (Principal Component Analysis) of the body geometry set consisting of skeleton configuration and the skin mesh displacement is carried out. This has resulted into a compact, redundancy-free representation of the body geometry. Subsequently, the modeler during run-time enables the user to automatically and effectively generate new desired models that best match the input parameter sets, almost instantly. For the re-animation of resulting models, we recalculate the skin attachment based on the initial skinning data and newly evaluated skin displacement.

Finally, we investigate further extension of the system by identifying the geometry components that are parameter-variant and those that are inherent to characteristics of individual body, through a mathematical tool of weighted regression (Chapter 6). By using what we call modifier synthesizer, various manipulations of an individual are demonstrated according to more complex parameters such as fat percentage.

One of the main contributions of this thesis work is a framework that combines the strengths of various techniques adopted for a robust modeling task: realistic, quality shape from capture techniques, the statistical methods that captures the variety and commonality of the 3D models, the automated learning technique of example-based methods, and the capacity to control the posture and the shape of whole body models using high level parameters. Specific contributions of each component are discussed in the related chapters.

Recently, some of the other works (Blanz and Vetter, 1999) (Sloan et al, 2001) (Allen et al, 2002) approach the problem of parameterized modeling and animation of human model by using scanned skin surface and blending them during manipulation, there are several key differences that distinguish our framework from others: First, it takes broader scope by dealing with the whole body model. Second, it adopts unique and compact shape description of the body geometry. Finally, we focus on generating diversity of appearance, although theirs is primarily on dynamic shape deformation during animation, in the domain of body modeling.

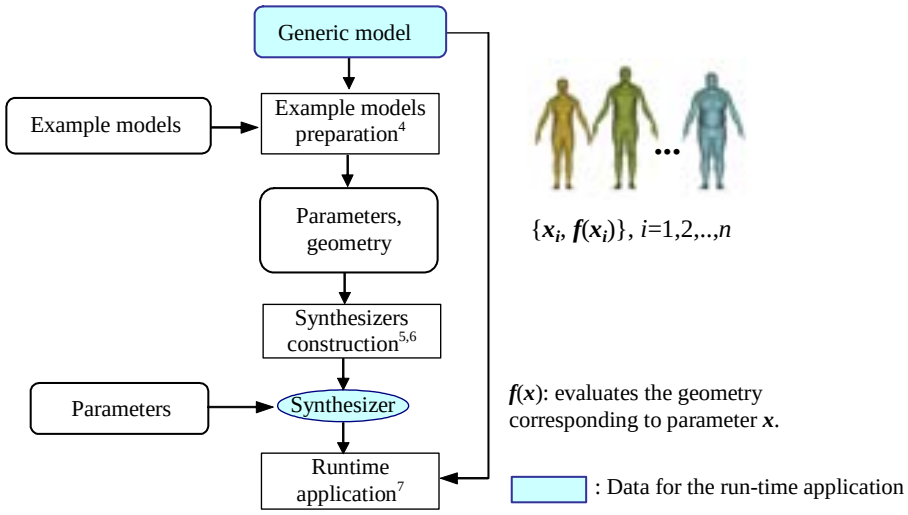


Figure 3-1. Workflow of the method. The numbers in the diagram indicate the corresponding chapter numbers in this document.

The overall framework is illustrated in Figure 3-1. The system largely consists of three major modules – the preprocessing of 3D scanned data, the construction of the parameterized synthesizers, and the runtime evaluator of the synthesizers that generates new body geometries from the user input parameters. As mentioned earlier, the system relies on external modeling systems or geometry capture technology to create initial sample shapes. Given the sample models, the *example model preparation module* ensures the correspondence among them by conforming a template model onto each example using the optimization-based fitting method. In addition, the attributes of the example model is measured and annotated. Based on the prepared samples and the corresponding attributes, the *synthesizers construction module* formulates the transformation function that maps the parameter (attribute) space onto the whole body space. According to the synthesizers constructed from this stage, the *runtime evaluation module* evaluates necessary deformations of the template model in order to obtain desired model on the fly.

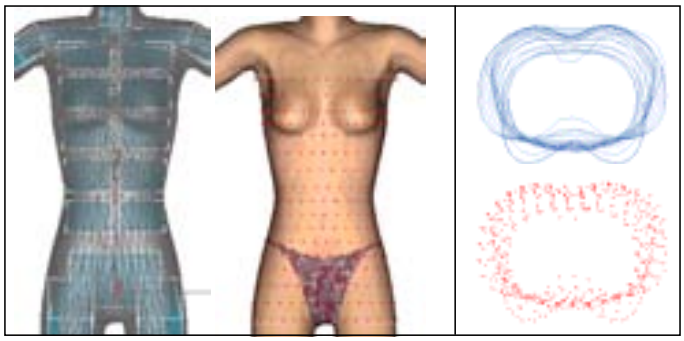
3.2 Earlier attempts

Earlier, as detailed in (Cordier et al, 2003), we have used a set of contours to represent the detail shape of the body geometry, particularly the torso part. The correspondence among examples has been found by fitting the template contour onto each example model. Note that many of the sizing parameters are associated with contours around the torso. The deviation of the contour vertex through the fitting process from its initial position is used to represent the elastic component of the deformation (see Figure 3-2).

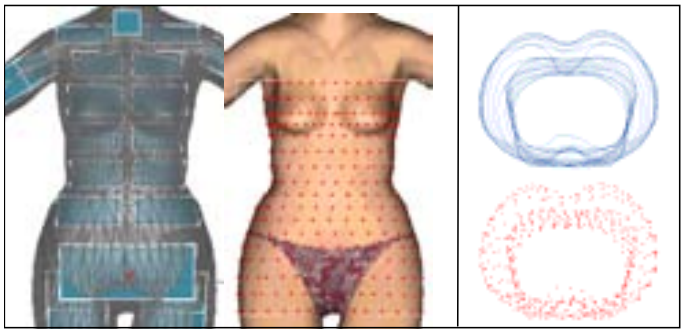
Several limitations of that approach can be listed: First, it assumes the intersection between the body geometry and a horizontal plane always forms a single contour, (i.e., the body geometry can be represented by the cylindrical projection) and thus cannot not handle general shapes. Obviously such an assumption cannot be made, for instance, on ptotic (droopy) breasts or belly. Second, the points-driven deformation of the mesh using the radial basis function does not handle some parts well, unless the width parameter(s) (the Gaussian function chosen) and the vertical alignment of the contours are carefully chosen. Typically, the deformation would perform poorly where the points are dense, such as armpits. Finally, the linear system associated with the point specifications severely limits the number of contours and thus that of point specifications (less than approximately 350 on an 800 MHz

Windows PC), in order to maintain interactive time (<1 second) performance for each new generation.

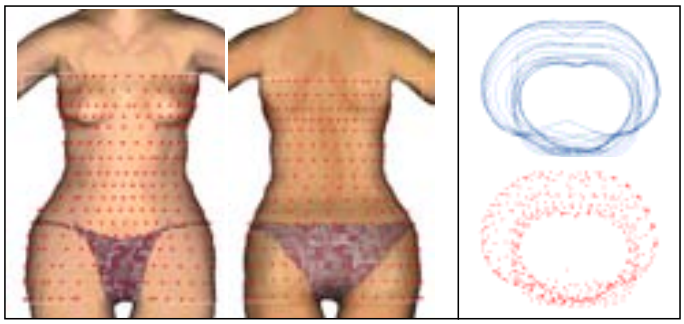
Later in this document, we examine more general representation that makes few assumptions about the shape and posture of the example body geometry. Since we want to capture as much variety as possible that are captured from real people, we attempt to minimize assumptions on the use of examples. Instead, we will consider preconditions on the template model. Also, we will evaluate the advantage of the general representation by demonstrating that it allows for the whole-body deformation at an interactive time.



(a) Initial reference model



(b) After the skeletal deformation via joint interpolation



(c) After adding detailed shape via shape interpolation

Figure 3-2. The early version of this work used contours.

Chapter 4

Model Preparation

"The most praiseworthy form of painting is the one that most resembles what it imitates"

Leonardo DaVinci

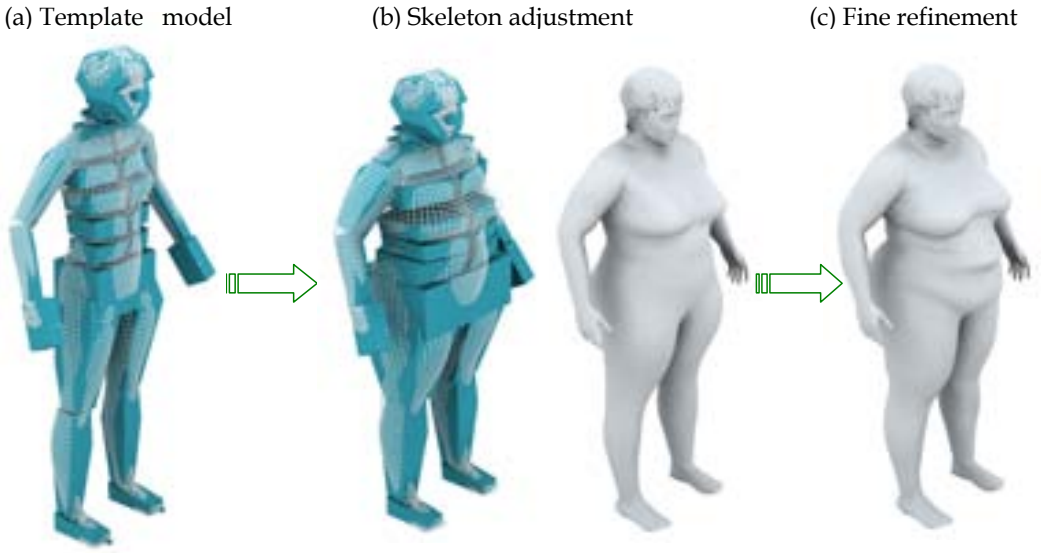


Figure 4-1. The two-phases of the deformation

In this chapter, we describe the acquisition and preprocessing of template model and the scanned data, which comprises the example database of our modelers.

As noted earlier, our example models or examples rely on existing models mostly obtained from 3D range scanners. These models, each of which having different topology and posture, are not directly usable for the shape transformation. As will be discussed in detail in Chapter 5 and Chapter 6, our modelers are based on shape transformation, where examples in correspondence are used to form a continuous range of the parameter space. Thus, one crucial prerequisite of such formulation is to identify corresponding points on every example scans. Additionally, joint centers of each scan model also must be estimated, because we want the resulting models remain animatable at any time during the shape transformation. This means that each example shares the identical mesh topology (same number of vertices and connectivity) as well as the joint hierarchy. Note that this allows us to represent the body geometry as a vector of fixed size.

A closely related problem, which is also known as the registration, has been extensively studied in medical research and 2D/3D warping (Besl and McKay, 1992)(Blanz and Vetter, 1999)(Lee and Lee, 2002). However, our problem differs from theirs in that our aim is not only to achieve the correspondence among different geometrical objects of the same class, but also the estimation of the skeletal configuration inside the skin surface.

In this work, we obtain the correspondence by conforming a template model onto each scanned mesh. Evidently, there are two assumptions made: (1) the topology of the model (the template model) is known in a priori, and (2) any human body geometry can be obtained by deforming the template model. In Section 4.3 and Section 4.4, we present our fast optimization algorithm tailored to find the skeletal estimation and the conformation of the template skin, to fit to the scan mesh. The basic idea is to use a set of pre-selected landmarks or feature points to measure the fitting accuracy and guide the conformation, in order to assure feature correspondence while minimizing the deformation energy. These feature points could either be automatically detected or manually assigned.

Having a predetermined topology has both theoretical and practical merits: First, it allows an efficient way to describe the shape of the objects (vector representation) we are dealing with at a desired level of detail. Second, we can reuse the initial skinning data so that all resulting models are made to be immediately animatable. Finally, different postures of examples are easily resolved by applying appropriate transformations to the template skeleton that derives the linear deformation of the template mesh.

Throughout this work, body geometry is considered to have two distinct entities, namely rigid and elastic component of the deformation. The rigid deformation is represented by the corresponding joint parameters, which, when applied to the template skeleton, will determine the linear approximation of the physique. The elastic deformation is essentially vertex displacements, which, when added to the template surface after the rigid deformation, depicts the detail shape of the body. Consequently, the fitting of the template model onto a scanned mesh is composed of two major steps~ skeletal fitting and fine refinement. Figure 4-1 illustrates this two-step preprocessing phase.

Since our preprocessing phase converts the scanned body models into animatable ones as a partial result, approaches for obtaining semantic structure on the scan data are also comparable to ours. Dekker et al (1999) have used a series of anatomical assumptions in order to optimize, clean and segment data from a Hamamatsu whole body range scanner in order to generate quadpatch representations of human bodies and build applications for the clothing industry. Ju and Siebert (2000) have introduced methods to automatically segment the scan model to conform it to an animatable model. Similarly to our approach, they employ a two-phase algorithm that consists of global and local mapping. While their local mapping is based on, closest points in Euclidean distance, however, our fine skin refinement undergoes iterations of relaxation and re-mapping phases so that the initial regularity on the template mesh is kept as much as possible.

To summarize, our preprocessing consists of three successive phases:

Phase 1. Feature point identification: A number of feature points or landmarks, around 30 in numbers, are labeled in a semi-automatic manner.

Phase 2. Initial skeleton fitting: This phase finds the linear approximation (posture and proportion) of the scanned model by conforming the template model to the scanned data through skeleton-driven deformation. Based on the feature points, the most likely joint configuration is found that best fits the corresponding feature points location. We cast this problem as multidimensional minimization problem. The variables in the optimization are the degree of freedom of each joint in the skeleton hierarchy. An optimizing joint variables are determined that minimize the distance of corresponding feature points location.

Phase 3. Fine skin fitting: Starting from the roughly fitted model from the previous phase, this phase further deforms the template mesh by computing the displacement vector for each vertex. It iteratively improves the fitting accuracy by minimizing the shape difference

between the template and the scan model. The found shape difference is saved into the displacement map.

The example models should then be organized to serve as input into the shape interpolator. More concretely, each example model should be measured at various parts of interest and be annotated.

4.1 Data Acquisition

4.1.1 3D Scanned Data

Example body models used in this work were obtained from 120 scanned bodies (60 for each) of European adults, acquired mostly from the Tecmath range scanner (Tecmath, AG)[♠]. The scanner aims at a low cost, automatic, and rapid – almost instant once the range scan is completed – whole body measurement extraction. Faces had intentionally been removed due to the privacy policy (scanned subjects were mostly customers of a clothing company). As is the case with many other scanner data, they are noisy, overly complex, and have holes. Thus we have made additional processing on the initial polygonal models that are obtained from the scanner, using commercial graphics packages (http::3ds max); holes were filled, the number of triangles was reduced up to 75,000, and all triangles were connected to form one single mesh. As a result, each scan data is composed of one single mesh with no holes and no open edges. Some of the example data after the additional processing are shown in Figure 4-2. Throughout this work, we assume that each example data is a polygonal mesh with moderate complexity. Texture data was not available and therefore is not within the scope of this work.

As is evident from Figure 4-2, all subjects were in an erect posture with arms and legs slightly apart, and are lightly clothed, which permitted to carry out ‘touchless’ measurements. Each of the 3D scanned data was measured at various parts – mainly around the torso – and was annotated with its sizing parameters. The sizing parameters used in this document are summarized in Figure 5-2, in Section 5.1.

[♠] The data was made available by courtesy of Athens Technology Center (ATC), in the framework of European Research Project E-Tailor, IST-1999-10549.

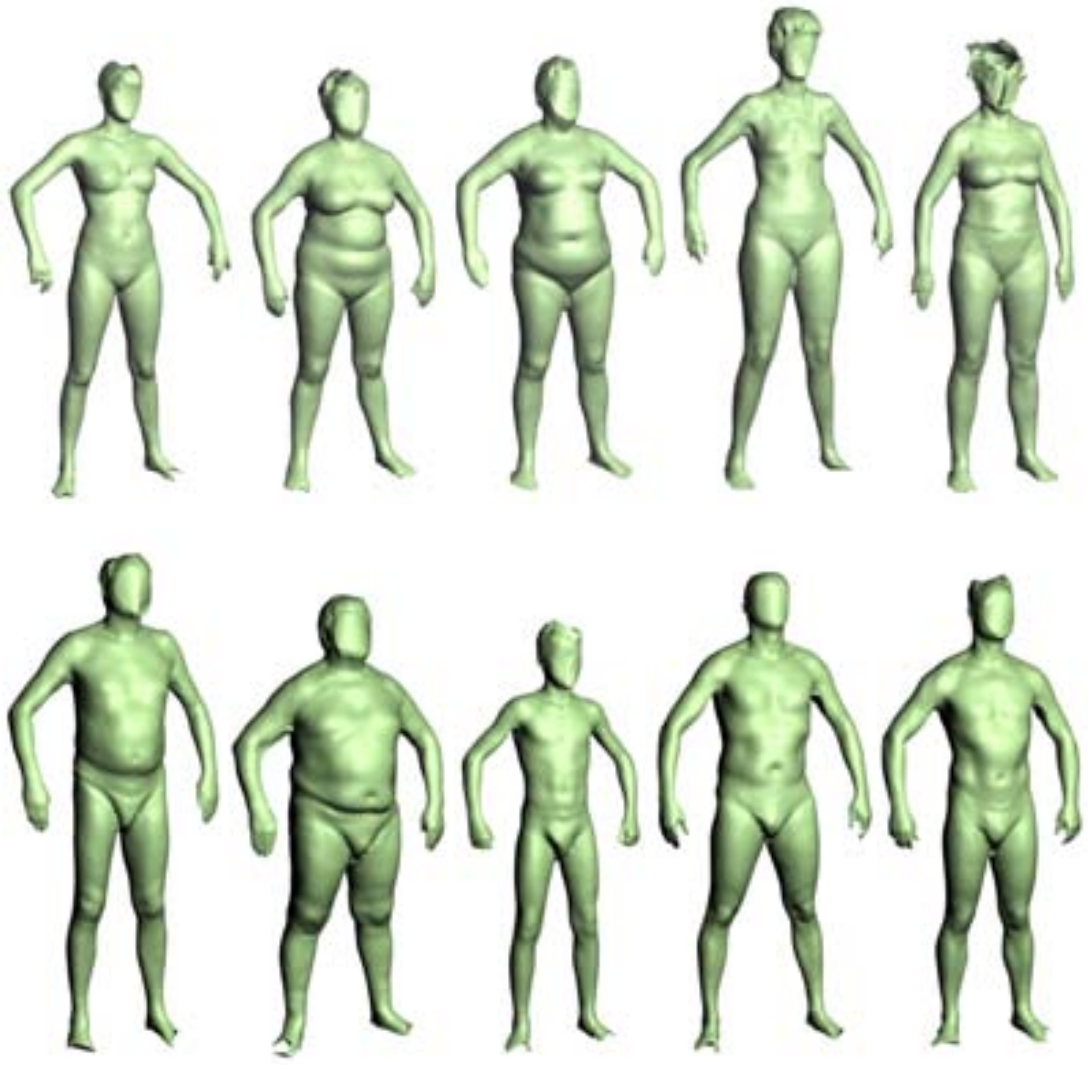


Figure 4-2. Some of the scan canned subjects used in our example database.

4.1.2 Template model

As shown in Figure 4-3, we use a two-layer model for the template – skeleton and skin mesh without intermediate layers representing the fat tissue and/or muscle. The preparation of the template models consists of four successive procedures: (1) A skeleton and (2) a template mesh are created and their relative positions are adjusted to each other. (3) The skin attachment information is then calculated. (4) Finally, feature points and feature contours are defined on the template mesh. The whole process is made in the 3D Studio MaxTM (http::3ds max) environment. For the skin-to-bone attachment, a commercially developed plug-in (http::BonesPro) was used. The feature point and contour definition was made by using our in-house plug-in.

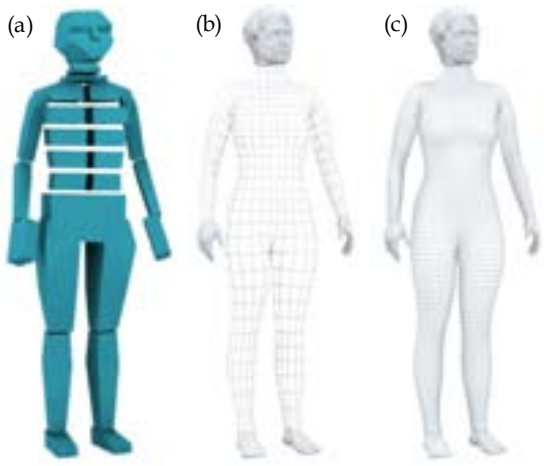


Figure 4-3. The skeleton and the mesh of the female template model at different levels of details: (a) skeleton hierarchy; (b) The skin mesh at a lower resolution is made of 861 vertices and 1,676 patches; (c) The one at a higher resolution has 3,401 vertices and 6,704 patches.

4.1.2.1 The skeleton

We have chosen an HAnim Level of Articulation (LoA) 2 skeleton (http://HAnim). Composed of 33 joints, excluding those for hands and feet, LoA2 offers skeleton that is a good approximation of true human kinematics and yet not too complicated to solve for or animate. Typically those joints along the spine are simplified. All bones are hierarchically linked, meaning that the transformation (scale, rotation, translation) is inherited from the parent to the children. (In our work, 'skullbase' is treated exceptionally by disabling the scale inheritance from its parent.)

We note that at the time of skin attachment, all bones have the scale transformation of value 1.f, although, typically, the bones are fitted to the skin mesh through scale, translate and rotation of each bone of the template skeleton.

4.1.2.2 The template mesh

Our initial template mesh as shown in Figure 4-3(b) has been generated using one of the scan models. The template mesh has a grid structure: it is composed of vertical lines and horizontal contours that form a set of Bezier quad-patches. Such a grid structure enables us to take all supported measurements – girths and lengths – immediately, at any time during the process. That is to say, measuring the model is simply computing lengths of a number of predefined contours. Quadpatch grid also offers an easy way to refine the mesh by using Bezier patch subdivision.

4.1.2.3 The Skeleton-Driven Deformation

Upon completion of the mesh model and the skeleton, the template skeleton was adjusted to fit correctly to the mesh before we perform a process called skinning or skin attachment. Skinning is essentially to establish a skin-to-bone relation so that a smooth skin deformation can be obtained whilst the joints are transformed.

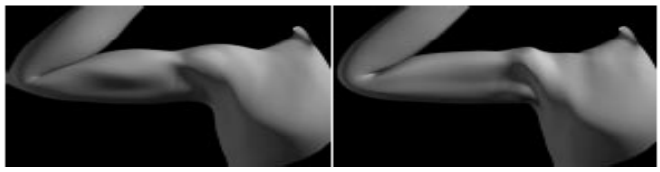


Figure 4-4. Pose space deformation (Lewis et al 2002, left) result compared to the skeleton driven deformation (right).

The skeleton-driven deformation (Weber, 2000), a classical method for the basic skin deformation is perhaps the most widely used technique in 3D character skin deformation. Despite significant works on visualizing realistic, dynamic skin shape (Lewis et al, 2000)(Sloan et al, 2001)(Allen et al, 2002), most character animation in interactive applications is based on this geometric skeletal deformation technique, which is also employed here. A prerequisite of a successful skeletal-driven deformation is an appropriate attachment of the skin to the underlying skeleton. The attachment is considered as assigning for each vertex of the mesh its affecting bones and corresponding weights. To say that a vertex is "weighted" with respect to a bone means that the vertex will move as the bone is transformed in order to stay aligned with it. In our current implementation, this is done using an external application (http::BonesPro).

Later in this dissertation, the skeletal deformation is used not only for the skin deformation but also to modify the size of particular segments and the posture by applying rigid transformation on the joints (Section 4.3). The shape of scan data is first approached by skeletal deformation on the generic mesh. Transformation matrix of every joint is calculated in a way that the generic shape obtained by skeletal deformation is as close as possible to the scan mesh. These generated matrices are the skeleton configuration and they correspond to the global proportion of the scan bodies.

Figure 4-5. Skeletal deformation examples occurred by rotation and scale of selected joints.

4.1.2.4 Feature points and feature contours

Parallel to acquiring the skin attachment, feature points are identified to guide the skeleton fitting. Despite enormous amount of works in various fields of computer graphics, the problem of automatic feature detection is not yet completely solved (Lee and Lee, 2002). In our case, some of the body features such as the breast nipple or the navel are particularly difficult to identify in a fully automatic way. The texture is missing and the precision of scan

data is not good enough to identify their location. In our approach, user input is utilized to identify those feature points.

Feature contours are also defined at this stage. They are used not only in measuring the template model at any time during the process but also in the fitting process, as detailed in Section 4.4.2.

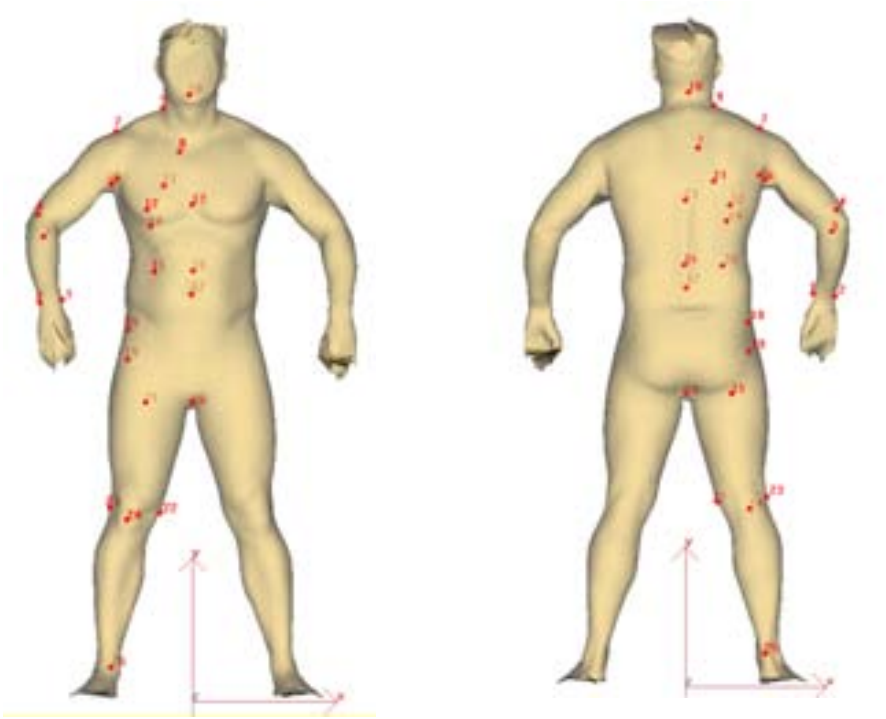


Figure 4-6. Feature points set used in our approach.

4.2 Representation

4.2.1 Skeleton configuration and displacement map

We assume that any body geometry can be obtained by deforming the template model. The deformation has two distinct entities, namely the skeleton and displacement components of the deformation. The skeleton component is the linear approximation of the physique, which is determined by the joint transformations (scale and translation; rotation remains constant throughout the manipulation as we assume that the pose is determined by the front-end application afterwards.) through the skinning (See Section 4.1.2). The displacement component is essentially vertex displacements, which, when added to the skin surface resulting from the skeletal deformation, depicts the detailed shape of the body.

We denote the skeleton component as

$$J = (t_x^1, t_y^1, t_z^1, s_x^1, s_y^1, s_z^1, t_x^2, ..., t_y^m, t_z^m, s_x^m, s_y^m, s_z^m)^T \in R^{6m},$$

where t_x^j and s_x^j are the translation and scale of joint j ($j = 1,2,...,m$) along the x-axis, and the displacement component as

$$D = (d_x^1, d_y^1, d_z^1, d_x^2, ..., d_y^n, d_z^n)^T \in R^{3n},$$

where d_x^1 is the displacement of vertex v ($v = 1,2,..,n$) along x-axis on the skin mesh. We therefore represent the geometry by a body vector $B = (J, D)$.

4.2.2 Principal Component Analysis (PCA)

Although the transformations computed independently for each bone may be used as above, significant redundancy exists in the bone transformations, exhibiting similar transformations among neighboring bones. For instance, bodies with larger hip girths than the average (large s_x and s_y of the 'sacroilliac' joints) will most of the time also have larger thighs (large 'hip' joints). A similar redundancy exists as regards the skin displacement. Thus we seek simplifications that allow the synthesizer to operate on compact vector space. In both cases, we adopt PCA (Press et al, 1988), one of the common techniques to reduce data dimensionality. Upon finding the orthogonal basis called eigen vectors, the original data vector $\boldsymbol{x}$ of dimension n can be represented by the projection of itself onto the first M ($<<n$) eigen vectors that corresponds to the M largest eigen values. In our case, the first 25 bases were enough to describe 99% of variations among the data so a 25-dimensional space was formed. Thus, the final representation of the body vector $B_N = (J_N, D_N)$ is composed of two sets of 25 coefficients that are obtained by projecting the initial body vector onto each set of eigenvectors, where

$$J_N = (\iota^1, \iota^2, ..., \iota^{25})^T \in R^{25}, D_N = (\delta^1, \delta^2, ..., \delta^{25})^T \in R^{25}.$$

4.3 Skeleton fitting

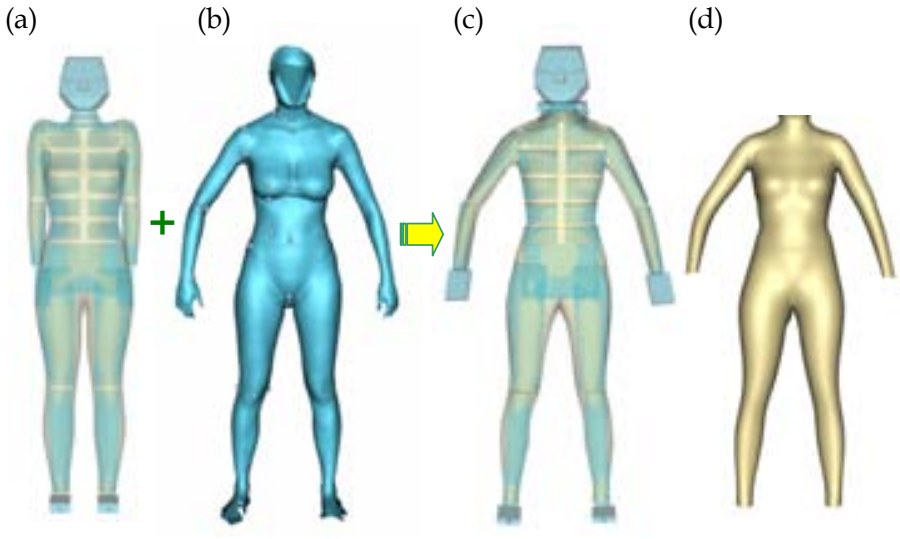


Figure 4-7. Skeleton fitting. The template model (a) is fitted to the scan data (b), resulting in the linear approximation of the scan model as shown (c) (with the skeleton) and (d) (mesh only).

The first phase of the algorithm finds the relative position, proportion and the posture of scanned model by appropriately fitting the reference model to it, as illustrated in Figure 4-7. We refer to the transformations applied to each joint of the reference skeleton as joint parameters. When applying different joint parameters during fitting, we use skeleton-driven deformation to calculate the new positions of the skin surface according to the transformation of the underlying skeleton.

The most likely joint parameters are the ones that drive the template skin such that it best matches the feature location by minimizing the error given by

$$E_F(X) = \sum_{i=1}^{m} \left\| P_i - P_i'(X) \right\| \qquad (4.1)$$

where X is the joint parameters, m is the number of feature points that are influenced by X , P_i and $P_i'(X)$ denote the location of feature point i in the scanned mesh and the calculated location of the corresponding feature point in template model that is driven by the skeletal deformation, respectively.

We recall that a bone affects only a few feature points and therefore the error distance is summed on the influenced feature points only. Using the skin attachment information, we can easily identify the feature points that are influenced by a bone. When there are no feature points that are attached to the current bone, feature points from its child or descendants from the highest depth are used. Using only a subset of the feature points provides sufficient accuracy for our purposes. At times, feature points that are weakly attached to a bone may result in distortion of the bone, finding an extreme scale transformation along one or two directions. Weighting the distance with the vertex-to-bone weight can prevent this, but it will result in loose fitting for some points. To overcome this problem, we add an internal energy term to the objective function that penalizes the joint configuration that violates the scale ratio uniformity. Thus, the energy given in Equation (4.1) is modified to:

$$E(X) = \alpha \cdot E_F + \beta \cdot E_D$$

with

$$E_D(X) = \frac{s_x}{s_y} + \frac{s_y}{s_z} + \frac{s_z}{s_x},$$

Optimization methods based on analytic function derivation as in (Allen et al, 2002) could be employed to find a solution. However, due to the complexity of the objects we are dealing with, it would be too time consuming. In our work, we employ a direction set method (Press et al, 1988). Half of the bones on which the feature points are attached are searched. End bones (the head, hand and feet) are excluded, as there are no closing markers to derive a meaningful search.

We find the joint parameters one bone at a time. For each new bone that is encountered in the depth first search (DFS), we form a search space that is defined by its DoF. The root joint has full DoF (translation, scale and rotation) while other joints excluding spinal joints have only scale (sx, sy, sz) and rotation (rx, ry, rz). Spinal joints have limited rotation, to avoid undesirable torsion and/or abduction of the spine joints due to the limited specification of feature points. We restrict the range of the rotation search space to (–pi/2 ~ pi/2) and the scale to larger than 0.0.

An obvious problem with this approach is that the search may terminate with different parameters depending on the order of the degree of freedom in the search space. We have experimented with translation, rotation, and scaling (TRS) and scale, rotation and translation (SRT) and concluded changing the order of transformation does not affect the resulting skeleton configuration significantly. In our work, TRS was used and the X_j becomes:

$$X_j = (t_{x_j}, t_{y_j}, t_{z_j}, \theta_{x_j}, \theta_{y_j}, \theta_{z_j}, s_{x_j}, s_{y_j}, s_{z_j})$$

where t_{x_j} and s_{x_j} are the translation and scale along the x-axis respectively, and θ_{x_j} represents the rotation around the *x*-axis of the joint j.

Due to the hierarchy, each bone inherits the transformations from its parent bone. We found the inherited transformation serves as a good initial value. For the 'HumanoidRoot' bone, the default value from the template model was used as the initial transformation. In some instances, the shared influence of the bones on a feature point (For instance, markers around the elbow are influenced both by the 'upper arm' and the 'forearm' bones.) can lead to a subsequent search with a child bone that results in an increase of the distance of the feature point whose minimal solution is found earlier by its parent bone. Nevertheless, we found sequentially visiting bones in a depth first search manner (DFS) and finding locally optimizing transformations results in a good approximation of the subjects.

While this generally produces acceptable linear approximation, the user has the option to selectively define the transformation of each bone after the automatic fitting. Repeated application of the optimization procedure resulted in a fairly good skeletal configuration. Figure 4-7 shows the skeleton and the template skin control mesh that has been fitted accordingly through the skeleton-driven deformation. The running time for one complete fitting is approximately 5 seconds on a 1.0 GHz Pentium 3.

```
proc skeleton_fitting ()
{
   P'← feature points on the scanned data;
   for each bone b encountered in the DFS order          // One bone at a time
   {
      if (b is an end-bone) continue;                    // End-bones are excluded
      X ← DoF of b;                                      // Use the inherited DoF as initial value
      P ← feature_points_under_influence (b);

      r ←b;   l ← 1;
      while (P =∅)              // When there's no feature point attached to the current bone,
                                // use the fp's from its child or the closest descendants.
     {
         for each c, a child in the level l  in a tree with r as a root
            P ← add (P, feature_points_under_influence (c));
         l ← l+1;
      }
      direction_set_method (X, P, P');
   }
```

Figure 4-8. Pseudo code of the skeleton fitting procedure.

4.4 Fine refinement

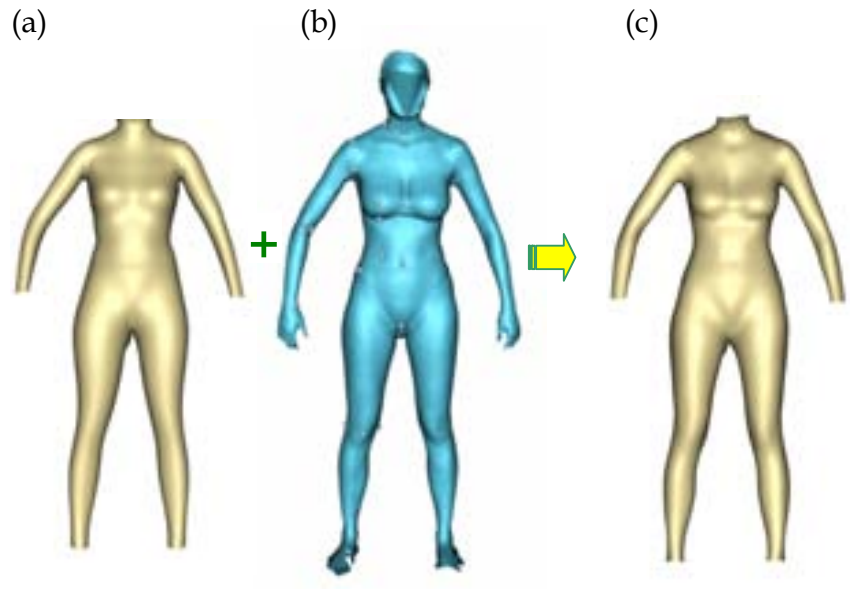


Figure 4-9. Fine fitting. (a) We start with a roughly fitted model from the previous phase. (b) The scanned model (c) After the fine refinement.

Starting with a roughly fitted template model that has been found from the skeleton fitting phase, this phase iteratively improves the fitting accuracy by minimizing the shape difference between the template and the scan model (see Figure 4-9 above). The found shape difference is saved into the displacement map. The fine refinement operates in the following procedure:

1) Initial mapping: First, initial mapping of the template mesh onto the scan mesh is found. Each vertex on the template mesh is projected onto the closest vertex, edge or triangle on the scan mesh through collision detection. Barycentric coordinates are generated for each template vertex.

2) Relaxation: Then, the vertex positions of the template mesh are updated by minimizing the energy function that (4.1) minimizes the amount of mesh deformation and (2) constraints vertices that belong to the same feature contour to be co-planar. A displacement vector between the previous and the new position is calculated.

3) Mapping: New vertices are projected on the surface of the scan mesh. Barycentric coordinates that are initially calculated from step1 are updated with optimized collision detection using the displacement vectors.

4) If the new shape is satisfactory, the algorithm terminates, otherwise step2~step3 is repeated.

5) Refitting of the skeleton: Now that we have found the displacement map, we can refine the skeleton fitting from Section 0 so that the thickness of the displacement map is minimized. The idea is to compute the transformation of each joint that minimizes the distance between the two segments – one the result of the skeletal deformation only and the other with the displacement map added. The cost function in this minimization computes the summation of the vertex distances between the source and the target. The direction set method is used to do that.

4.4.1 Mapping

Each vertex on the template mesh is moved to the closest vertex, edge or triangle on the scan mesh using collision detection. The collision detection used here is based on the hierarchical scheme described in (Volino and Magnenat-Thalmann, 1994). That algorithm first constructs a hierarchy called surface region tree on the polygonal mesh as a preprocessing. A surface region is essentially a segment of the surface that is composed of a group of adjacent polygons. The hierarchical structure permits to efficiently limit the possible range of collisions, and subsequently to gain calculation time. Collision detection is performed by testing collisions between the bounding volumes of the surface regions of the hierarchy.

In order to avoid re-computation of collision detection after each relaxation step, we check before collision detection if the newly calculated displacement of a vertex has changed since the previous mapping. Only those vertices whose displacement has changed are newly calculated. Feature points are directly mapped onto the corresponding feature point locations, as shown in Figure 4-10.

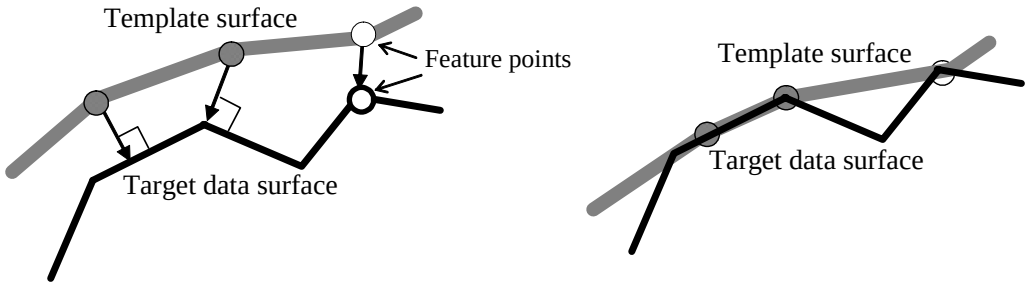


Figure 4-10. Surface mapping via collision detection.

4.4.2 Relaxation

Attempting to fit surface through collision often produces the irregular, unwieldy result mesh, which is unacceptable. In this section, we consider the problem of improving the regularity of the conformed mesh by finding appropriate vertex relocation.

Recalling that mapping started from the fair template mesh, repositioning the vertex so that it restores its initial state can guide to a good fit and can be formulated as an energy minimization problem. We define two energy terms for each vertex. The first term measures the regularity of the deformation relative to the initial mesh. As shown in Figure 4-11, each edge (AB, BC, CD, DA) of a quad-patch is considered as a spring whose rest length is defined from the initial template mesh. Plus, two additional springs (AC and BD) are superimposed along the diagonal.

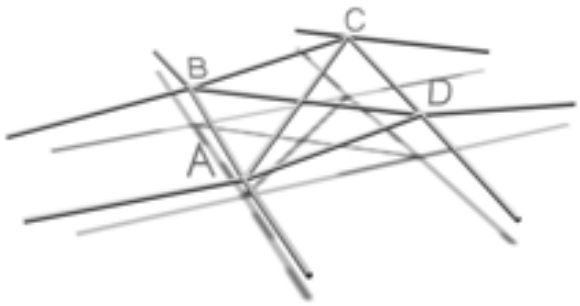


Figure 4-11. There are six springs over a quad-patch (AB, BC, CD, DA, AC, BD).

The energy function of the spring mesh thus can be written as:

$$E = \sum_i \frac{1}{2} k \left| e_i - e_i' \right|^2 ,$$

where k is the spring coefficient and e_i and e_i' are the i-th edge lengths of the initial and the current mesh, respectively. If the vertex is displaced, its neighboring edges will develop restoring energy such that it tends to move towards the most stabilizing location.

The second term is to constrain vertices that belong to the same feature contour to be co-planar. For every contour, an average plane is calculated. Vertices are forced to move to their projected position on the plane with zero-length springs.

The relaxation is performed by an iterative gradient descent method (Press et al, 1998) to minimize the total energy function. It is an attractive optimization method in that it is conceptually straightforward and often converges quickly. For example, we find that for female template model that is composed of 861 vertices, as few as 5~10 iterations are sufficient to get a result with acceptable precision.

4.4.3 Refitting of the skeleton

Now that we have found the displacement map, we can refine the skeleton fitting from Section 4.3 so that the thickness of the displacement map is minimized.

First, the skin mesh is segmented into different pieces whose the most influencing bone is identical. For each segment, there exist two shapes; the source shape is the result of the skeletal deformation only and the target shape with the displacement map added. The fitting process can then be formulated as the following problem: compute the transformation of each joint that minimizes the distance between the two segments. The cost function in this minimization computes the summation of the vertex distances between the source and the target. We use the direction set method (Press et al, 1988) to do that. As a result, a new skeleton configuration is generated.

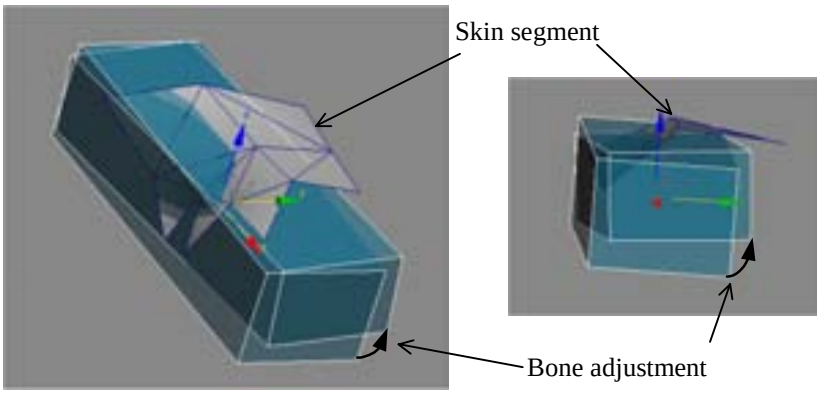


Figure 4-12. The bone refinement is driven by the attached skin segment.

4.4.4 Multi-resolution subdivision

It is evident that the detail of the captured shape through the conformation is limited by the number of geometric elements (vertices, edges, facets,..) in the template model. Earlier in Chapter 1, we pointed out that we are primarily aiming at a real-time application, possibly with other complex objects such as garments. Although this naturally resulted in the

avoidance of the dense representation in the template mesh, we also want to be able to increase its level of detail when the visual realism is a priority.

To elevate the density of the mesh representation, we apply subdivisions on the template mesh; each quadrilateral face is splitted into n^2 sub quadpatches, where n (=2,3,…) is the desired level of subdivision. We first convert the mesh into a set of Bezier patches. Each vertex on the mesh is considered as an end control point ($\mathbf{b_0}$ and/or $\mathbf{b_3}$) of a cubic Bezier patch. As illustrated in Figure 4-13, other control points $\mathbf{b_1}$ and $\mathbf{b_2}$ can be found by the continuity condition that neighboring patches are assumed to be joined continuously. We then interpret each quadrilateral face as a Bezier patch and sample the patch at ($u = 1/n, v = 1/n$) using DeCasteljau's algorithm (Farin, 2002). Each newly generated vertex can be mapped onto the scan data and thus the fitting accuracy is refined.

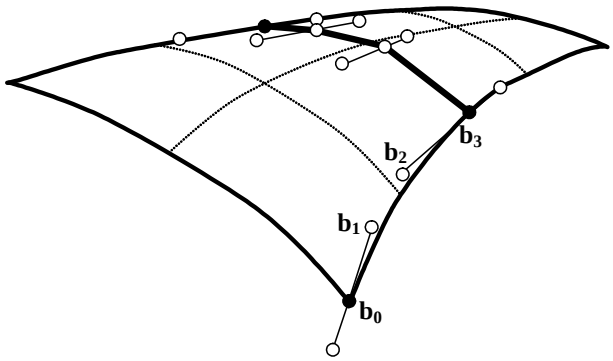


Figure 4-13. After intermediate control points b_1 and b_2 are found using the continuity condition of adjacent patches, the patch is subdivided using DeCasteljau's algorithm.

4.5 Results

In Figure 4-14, Figure 4-15 and Figure 4-16, the template model shown in Figure 4-3 has been fitted to the scan models. In Figure 4-14, the left columns show the scan data; the right columns show the template model after the skeleton fitting. Figure 4-15 shows the results of fine skin refinement after the initial mapping (left columns) and 10 iterations of mapping and relaxation (right columns). Figure 4-16 shows the final displacement maps. Blue colored points have smaller magnitudes than red points.

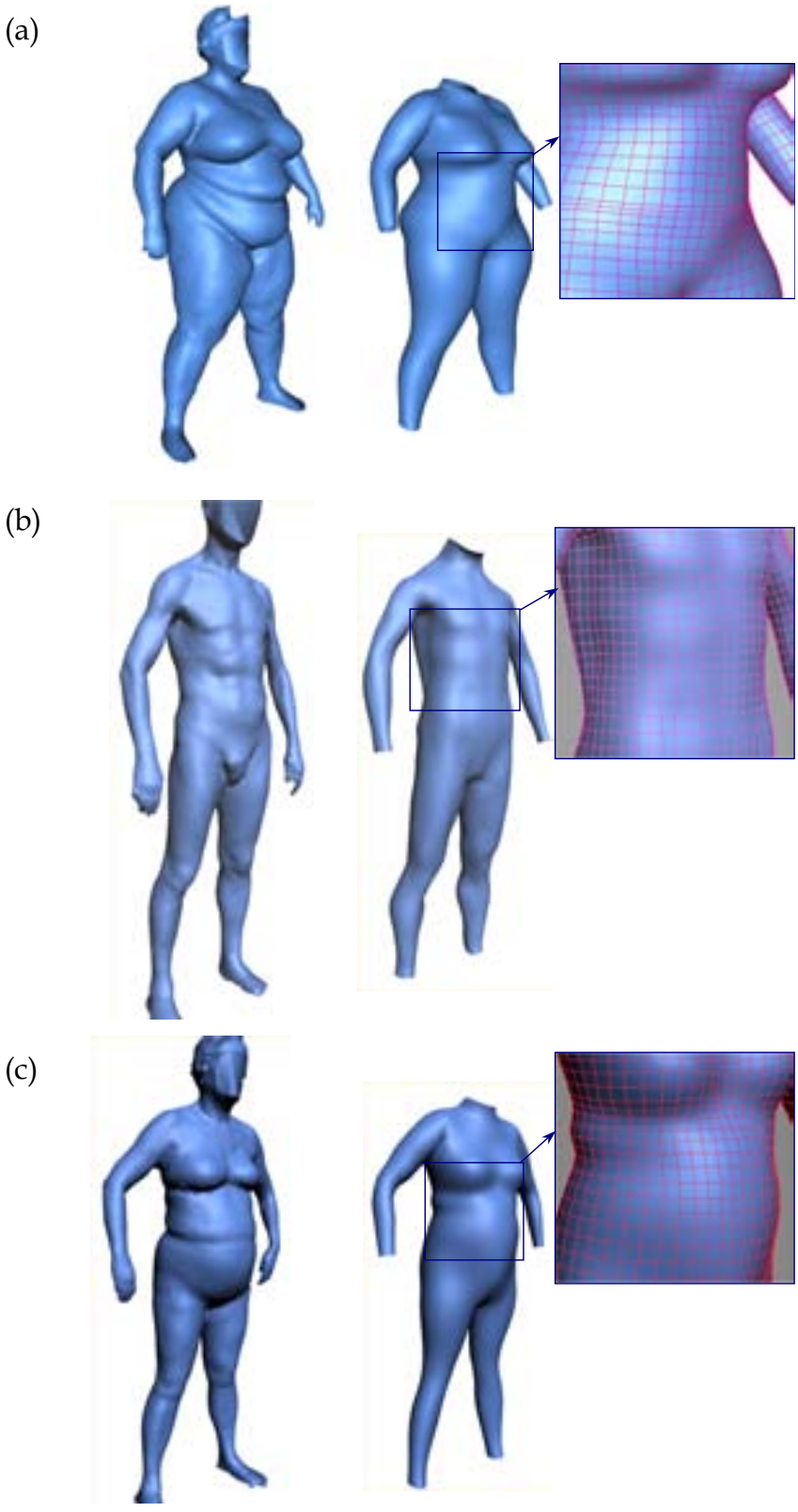


Figure 4-14. Three target scanned data (left column) and the template model after the skeleton fitting (right column).

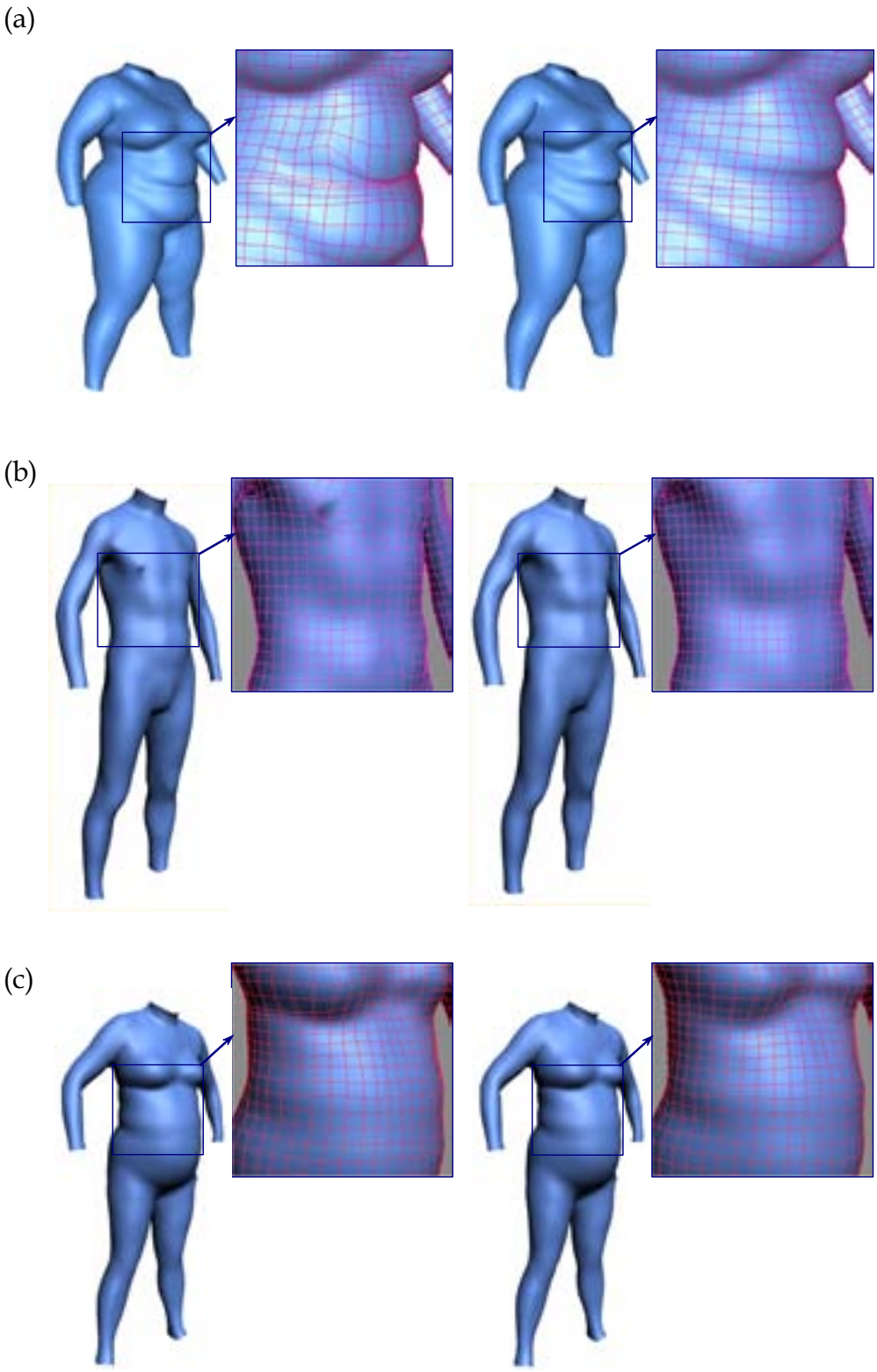
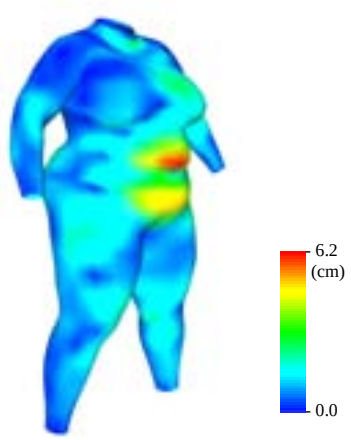


Figure 4-15. Initial mapping and the (left column) and after 10 iteration of relaxation and mapping (right column).

(a)



(b)

(c)

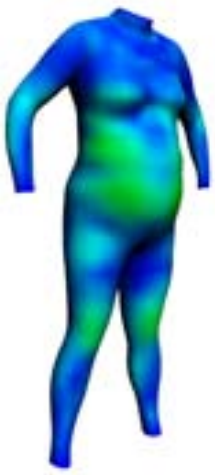


Figure 4-16. The final displacement map.

Chapter 5

Modeling Synthesizer

"You know the funny thing about morphin? You don't appreciate it till you can't do it any more!"

– David Yost (as Billy) in Mighty Morphine Power Rangers: The Movie (1995)

Figure 5-1. Various different bodies generated by our modeling synthesizer

The preprocessing phase, described in the previous chapter, finds the transformation required to deform the template model into each example data. In this chapter, we make use of these transformations to build what we call the modeling synthesizer. Once constructed, this synthesizer allows for the runtime evaluation of the body shape from the given input parameters.

As mentioned earlier, we model the body geometry by controlling size parameters. One might think this can be done by directly applying appropriate transformation of the corresponding bones from the sizing parameters. Such an approach has a number of limitations: first, one must estimate other transformations that are not directly determined from the size parameters. The second problem, which is essentially the same as the first one, is that it hardly guarantees realistic shape in the resulting geometry.

One powerful method to model and estimate the shapes of the same class is to leverage the quality that exists in example shapes. If a set of shapes with distinct formal characteristics is defined, we can generate a shape that blends these characteristics in any proportion using interpolation. Such shape transformations can be used in various ways for modeling and estimation purposes and is at the heart of this thesis.

Similarly to many of example-based methods (Allen et al, 2002)(Blanz and Vetter, 1999)(Sloan et al, 2001), we are looking for a smooth interpolation that transforms the parameter space onto the body geometry space by using examples as interpolants. Given a set of examples obtained from the previous stage, we consider that the measurements define a dimension space where each measurement represents a separate axis in the dimension space. Each example will then be assigned to a location in the dimension space. The goal is to produce, at any point p in the dimension space, a new deformation $X(p)$ of the template

model derived through the interpolation of the example shapes. When p is equal to the position p_i for a particular example model i, then $X(p)$ should equal to X_i, which is the shape of example i. In between the examples, smooth intuitive changes should take place.

Dealing with high dimensional geometry data and relatively small number of examples, our problem falls into a scattered interpolation problem. The standard problem of multivariable interpolation is as follows: given N distinct points p_i in R_n and N values v_i in R, find a function $f : R_n \to R$ such that for all i, $f(p_i) = v_i$ and such that f does not oscillate badly between values.

Given a set of examples obtained from the previous stage, we consider that the measurements define a dimension space where each measurement represents a separate axis in the dimension space. Each example will be assign to a location in the dimension space. The goal is to produce at any point p in the dimension space a newly deformed shape $X(p)$ derived through interpolation of the example shapes. When p is equal to the position pi for a particular example model i, then $X(p)$ should equal to X_i. In between the examples, smooth intuitive shape transformation should take place. A smoothly varying interpolation can be obtained across the dimension space by solving for coefficients of a linear system. There are two types of synthesizers: joint synthesizer handles each DoF of the joints. Displacement synthesizer is for finding the appropriate displacements on the skin from the input parameters.

The use of Gaussian Radial Basis Functions (GRBFs) is a common approach to the multidimensional scattered data interpolation problem and can also be used here. Consider a mapping from a d-dimensional input space **x** to a one-dimensional space. The output s(**x**) of the mapping takes a linear combination of the base functions:

$$s(\mathbf{x}) = \sum_{i}^{N} w_i \cdot \exp(-\beta \left\| \mathbf{x} - \mathbf{x}^i \right\|^2), \mathbf{x} \in R^d, w_i \in R,$$

where the N input vectors x^i consist of the example set, β is a width parameter that controls the smoothness of the interpolating function, and $\|\cdot\|$ denotes the Euclidean norm. As will be detailed in Section 5.2, the weight w_i can be determined by solving the associated linear system.

However, the standard RBF techniques would require too many interpolation functions, due to the high dimension of the shape descriptor. We recall that our template models are made of 33 joints with each having 6 DoF, and 841 vertices with 3 DoF. Our approach to this problem is to adopt a compact and efficient shape representation by using Principal Component Analysis, or PCA from hereafter (Press et al, 1988), one of the common techniques to reduce the data dimensionality. Upon finding the orthogonal basis called eigen vectors, the original data vector x of dimension n can be represented by the projection of itself onto the first M (<< n) eigen vectors that corresponds to the M largest eigen values. In our case, 30 bases for each DoF were used both for the joint and the displacements interpolators. We remind that our template models are made of 33 joints with each having 6 DoF, and 841 vertices with 3 DoF.

As described in Section 4.2.1 we consider the body geometry to have two distinct entities, namely the rigid and the elastic component of the shape deformation. The rigid component is represented by joint parameters of a person, which will determine the global proportion of the physique. Then, shape parameters are used to express the elastic component, which, when added, depicts the detail shape of the body. Subsequently, we split the modeling synthesizers into two kinds: joint synthesizers handle each DoF of the joints, while displacement synthesizers are used to find the appropriate displacements on the template skin, from the control parameters supplied as input (See Figure 5-2).

At runtime, these synthesizers evaluate the interpolation function to obtain necessary deformation that produces an interpolated shape. By allowing automatic modification from a set of parameters, our modeling synthesizer may lead eventually to the automatic generation of a variety of population models.

This chapter proceeds as follows: we first introduce the control parameters that are used in our modeler in Section 5.1. A background on RBF is detailed in Section 5.2. After discussing various elements that constitute the modeling synthesizer in Section 5.3, we describe the skin attachment recalculation method in Section 5.4. Finally, validation and results are presented in Section 5.5.

5.1 Control parameters

Before we discuss the synthesizers construction in detail, we explain the choice of control parameters. Arguably, the sizing parameters or anthropometric measurements allow the most complete control over the body geometry but providing tens of measurements required to detail the model would be almost impractical. Supporting dozens of measurements is beyond the scope of this work. Instead, we have chosen eight anthropometric measurements (5 girths and 3 lengths), which have been defined as the primary body measurements for product independent size assignment (Hohenstein, 1999). Descriptions of these sizing parameters along with the pictorial representation are shown in Figure 5-2. Using such a small measurement set not only provides compact parameters for the body geometry representation but also allows to be obtained easily by anyone, enabling applications such as an online clothing store, where a user is asked to enter his/her measurements for customized apparel design. Throughout this document, we assume that these key measurements are provided as input.

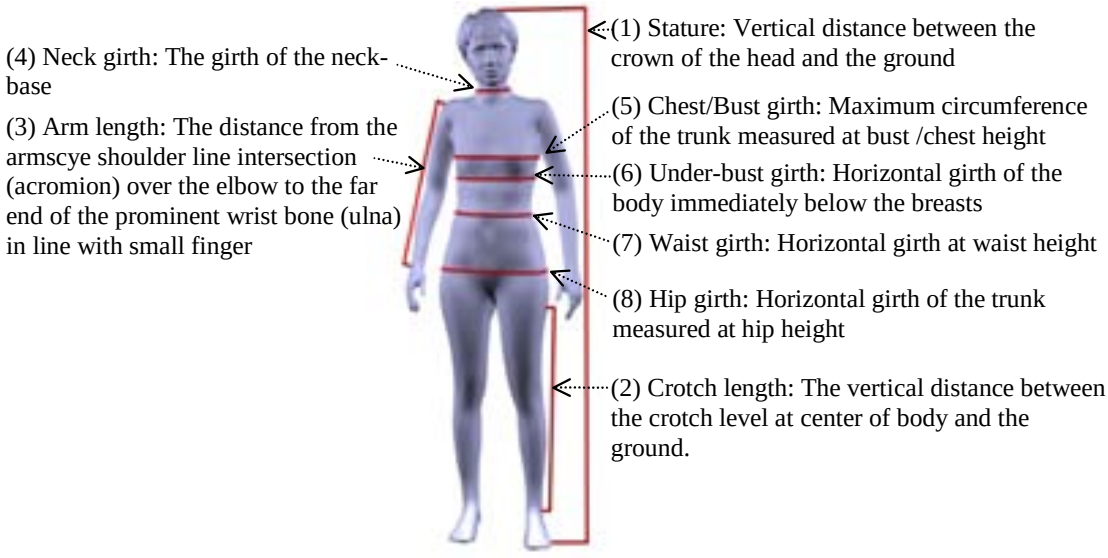


Figure 5-2. Anthropometric measurements used in our modeling synthesizer.

5.2 RBF as a scattered data interpolation

We begin by introducing radial basis function approximation for performing exact interpolation of a set of data points in a multi-dimensional space (Powell, 1987). Consider $f : R^d \rightarrow R$ a real valued function of *d* variables that is to be approximated by $s : R^d \rightarrow R$,

given the values $\{f(x_i): i = 1,2,...,n\}$, where $\{x_i : i = 1,2,...,n\}$ is a set of distinct points in R^d called the nodes of interpolation. We will consider approximations of the form

$$s(x) = p_m(x) + \sum_i^N w_i \cdot \Phi(\|x - x\|), x \in R^d, w_i \in R, \quad x \in R^d, \omega_i \in R \tag{5.1}$$

where p_m is a low degree polynomial, or is not present, $\|\cdot\|$ denotes the Euclidean norm and Φ is a fixed function from R^d to R. d and N is the dimension of the parameter space and the number of examples, respectively. Thus the radial basis function s is a linear combination of translates of the single radially symmetric function $\Phi(\|\cdot\|)$, plus a low degree polynomial. Then the coefficients, ω_i, of the approximation s are determined by requiring that s satisfy the interpolation conditions

$$s(x_j) = f(x_j), j = 1,2,...,n. \tag{5.2}$$

and orthogonality or side conditions

$$\sum_{i=1}^N w_i q(x_i) = 0, \text{for all polynomials } q \text{ of degree at most } m.$$

These side conditions along with the interpolation conditions of Equation (5.1) lead to a linear system to solve for the coefficients that specify the RBF. Some examples of popular choice of Φ includes

$\Phi(r) = r$ (linear)

$\Phi(r) = r^2 \log r$ (thin-plate)

$\Phi(r) = e^{-ar^2}$ (Gaussian)

$\Phi(r) = (r^2 + \sigma^2)^\beta, 0 < \beta < 1,$ (multiquadratic when $\beta = 1/2$)

where $r \geq 0$ and a and c are positive constants.

In our interpolation we are particularly concerned with deformation of the template model that is driven by the specified parameters: By specifying n examples $p_1, p_2, ..., p_n$ of body attributes and the value of the deforming transformation $W(p_i)$ at each of the parameter points, a Gaussian function interpolant is constructed which will be used to compute all points in the parameter space. Note that the body geometry is represented by finding the orthogonal basis and re-projecting the original vector (the rigid and elastic component of the deformation) onto its first m basis, as described in Section 4.2.2. Then Equation (5.2) implies that the coefficients of the radial basis function and the polynomial can be found by solving a linear system of m equations:

$$s_m(p_i) = \sum_{i=1}^n \omega_i^m \cdot \Phi(\|p - p_i\|), m = 1,2,...$$

where s_i interpolates the i-th eigen component of the body geometry, which is defined by $W = (s_1, s_2, ...s_m)$. $W(p) = (p_1', p_2', ..., p_m')$ is then inverse-transformed from the orthogonal space back to the original coordinate system, to obtain the actual deformation that constitute the result body geometry.

Through this document, we use a Gaussian function $\Phi(r) = e^{-\frac{r^2}{\sigma^2}}$ as the basis function for the multi-way blending. Gaussian Radial Basis Functions (GRBF) are reputed to behave well in a very mild condition. As well as being localized, they have a number of useful analytic properties.

The variance or width parameter σ controls the degree of locality or smoothness properties of the interpolating function. When too small value is chosen for σ, the resulting function is insufficiently smooth and gives a highly oscillatory function especially when the data is noisy. When too large, the function is over-smoothed, and gives again poor representation. Although in general the width parameter can be chosen to be roughly twice the average spacing between the basis functions, various techniques for setting and fine-tuning the basis function parameters, including σ, have been developed. We recommend readers to refer to (Bishop, 1995) for an extensive discussion on these techniques. In this work, instead of having a common width parameter σ, each basis function is given its own width σ_j whose value is roughly twice the spacing of the nearest neighbor interpolant.

The algorithm complexity of this technique is $O(n^3)$ for computing the coefficients, since values for the weights are found using matrix inversion, and $O(nN)$ for evaluating the interpolation functions to warp the object, where N is the number of geometric element (number of surface points, number of joints,…) of the graphical object in question.

5.3 Synthesizer construction

At the time of synthesizer construction, we define a parameter space where each measurement represents a separate axis. Each example models after the fitting is measured in various parts to find its corresponding location in the parameter space. A smoothly varying interpolation of a particular component of the body geometry can then be obtained across the measurement space by using these examples as interpolation nodes.

5.3.1 Joint interpolation

Although the transformations computed independently for each bone may be used, significant redundancy exists in the bone, e.g. neighboring bones often undergo similar transformations. For instance, when the bone ‘sacroilliac’, whose main influence covers the hips part, has a large scaling transformation, the neighboring bone ‘l_hip’ also tends to undergo a similar transformation, depicting a large thigh. This also results partially due to the fact that the size parameters themselves have correlations, when they are from a real individual. As mentioned earlier, PCA of bone transformations serves to find an optimal orthogonal basis. At first, the eigenskeletons for each DoF are computed. We truncate the eigensksleton basis and still have a good basis for approximating the observed bone transformations.

Each interpolator is responsible for one element of the joint parameters. As mentioned earlier, there are $30N$ (N: DoF, 6 in our case) interpolators, which are responsible for the scale and translation transformation of the joints in x, y and z direction (See Figure 5-3). Note that the joint interpolators do not contain rotation transformation, as they are considered to be independent of the body size.

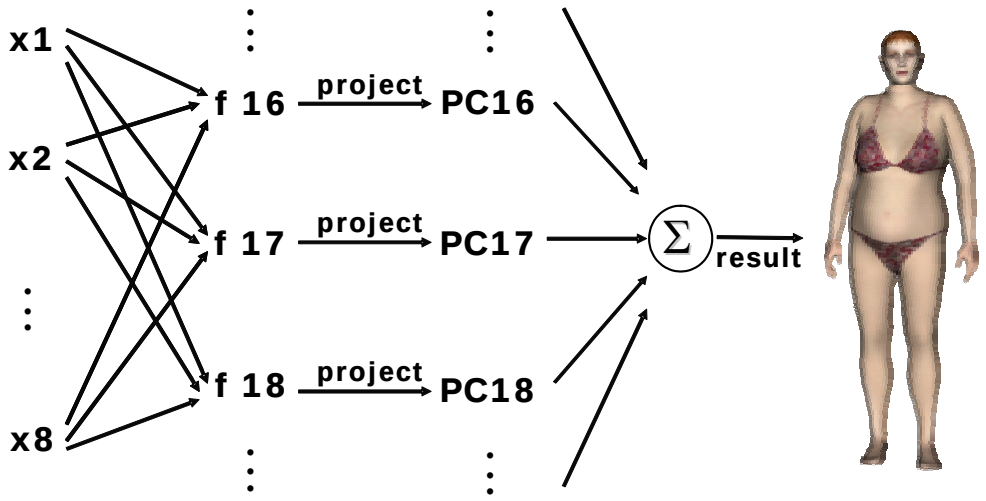


Figure 5-3. The modeling synthesizers, f1 to f30, evaluates the element of the body vector from the given control parameters (x1 to x8), each of which is projected onto the corresponding principal component. When summed up into a body vector, it results in the desired deformation on the template model.

It should be noted that the human body is a hierarchical structure and there are two ways of describing the transformation (scale for instance) of a joint: relative or absolute. Relative transformation specifies the transformation of each segment relative to its parent whereas absolute transformation specifies the transformation of each segment independently relative to the origin. Since the interpolator is ignorant of the hierarchical nature of the joint, we chose an absolute description of the joint parameters, each independent of its parent nodes.

5.3.2 Displacements interpolation

The displacements are essentially the vertex displacements of the template mesh as a result of the fine fitting, computed in a global coordinate system, and thus can have different values depending on the posture of the subject at the time of computation. In order to remove such posture dependent characteristics from the displacements map, the displacements are recorded from the same posture (in our case the posture shown in Figure 4-9(c)).

As described earlier, we use point vectors to represent the residual component of the body geometry. As is the case of the joint parameters, there is a significant spatial coherence in the skin displacement. Thus, we start by finding eigen-displacements through PCA. In our application here, the first 30 principal components were sufficient to reconstruct the original body shape. Consequently, three sets of interpolators of size 30 were constructed for x, y and z directional displacements.

Analogous to the joint interpolators, displacements interpolators exploit the size to shape correlations that exist in the example bodies, to regulate the estimation of realistic shapes given sizing parameters as input. Unlike the joint interpolators, however, they contain no linear term pm in the approximation function (see Equation (5.1)). This is due to the fact that the fat, muscles, are highly nonlinear and can be better modeled by purely elastic deformation.

5.3.3 Parameter space normalization

Radial basis functions have values that are determined solely by one parameter, the distance from a center point in the multi-dimensional space. In our applications, the measurements define such space where each measurement represents a separate axis in the dimension space. However, since each measurement has different average and variation, and since measurements are often correlated to each other, it is prohibitive to define the space by directly using the measurements unit. To take an example, since the height has the largest variation among all measurements, it will contribute largely to the distance and the result shape of the body will be sensitive to the height change. Another point, since the neck and the hips are positively correlated, the distance between the two models that have different neck and the hips will tend to be multiplied. Again, the result shape will become more sensitive to the neck or the hips than, say, the leg length. Instead, we use a normalized the parameter space by using PCA. In order to maximize the accuracy of the normalized space, the dimension of the original parameter space is kept, that is no truncation took place in the orthogonal basis.

5.3.4 Population synthesizer

Now that we have the individual synthesizer as described in the previous section, the problem of automatically generating a population is reduced to the problem of generating the desired number of plausible sets of parameters.

When automatically generating a set of anthropometric measurements, one must consider the dependencies or correlations among different measurements, as in (DeCarlo et al, 1998). In our case, as described earlier, we have removed the dependencies and formulated a normalized parameter space that is defined by orthogonal bases. Thus, we can generate each parameter independently to automatically produce the desired number of parameters that are needed to generate new individuals. We adopted a Gaussian random number generator (Press et al, 1988). The variance amongst the population is adjustable by the user, although default variance is set to the variance of the database.

5.3.5 Exaggeration

In our framework, it is possible to implement caricature and anti-caricature in three stages. (a) Starting with target geometry, a corresponding vector is determined in the normalized parameter space. (b) The parameter vector is fed to the synthesizer to generate a body vector in the normalized body space. (c) The body vector is multiplied by a scalar factor (negative in case of the “anti-exaggeration”) that is set by the level of exaggeration. (d) Exaggerated body model is created by combining the body vector with the eigen joints and eigen displacements to generate the skeleton and the displacement map. Figure 5-4 shows an exaggeration example using our modeler.

(a) Anti-exaggeration (b) Original model (c) Exaggeration

Figure 5-4. Exaggeration of a body is obtained by making use of the example database and the average body

We note that using our representation, the direction of the body (vector) from the origin in the normalized space represents the distinctive characteristic of the body from the average. By increasing the distance from the origin in that direction, it is clear that we can exaggerate the characteristics of the corresponding body. Similar approach has been demonstrated in the morphable face models by (Blanz and Vetter, 2000), although in our case there is a separation of the skeleton and the displacement components of the object.

5.3.6 Defining the limits of input measurements

Very often it is desirable to prohibit the user from inputting invalid measurements to the modeling synthesizer. The underbust girth should not exceed the bust girth; the inside leg length should be assigned with an appropriate measurement in relation with the height, and vice versa. Unlike geometric modeling, the interpolation we have adopted here is rather sensitive to the erroneous measurement input, especially when the number of examples is relatively small.

There are at least two possibilities to automatically filter out invalid measurements set and thus avoid odd and unrealistic body shapes from the modeling synthesizer. The first is to use the "if-then-else" scheme by using a number of min-max values that is defined from the correlation coefficients and conditional probability of anthropometric measurements. Table 5-1 illustrates such a scheme using the statistics collected from the German males of age 18~64, that has been compiled from PeopleSize Pro™ (PeopleSize, 1993-2001). This method works well when the measurements are provided in a certain order, like dependent measurements (inside leg lengths, bust girth) following the independent measurements (height, underbust girth).

The latter is to base the validity test on normalized measurement sets of examples. The correlation or redundancy among different measurement is first removed through normalization. Then the valid range of the measurement set can be defined by the min-max values of example measurements along each axis.

We use the second approach, which is the use of normalization (See Figure 5-5). The example measurements are normalized through PCA, with the record of min and the max values along each axis of orthogonal measurement space. We call the volume that is formed by these min and max values as valid measurement volume. Upon receiving the input measurements, we project them onto the orthogonal space and check if the measurement point is within the valid measurement volume. If the point is outside the volume, we can either automatically correct the measurement by repositioning the point onto the closest location in the valid measurement volume or just retrieve the most recent measurements that are valid.

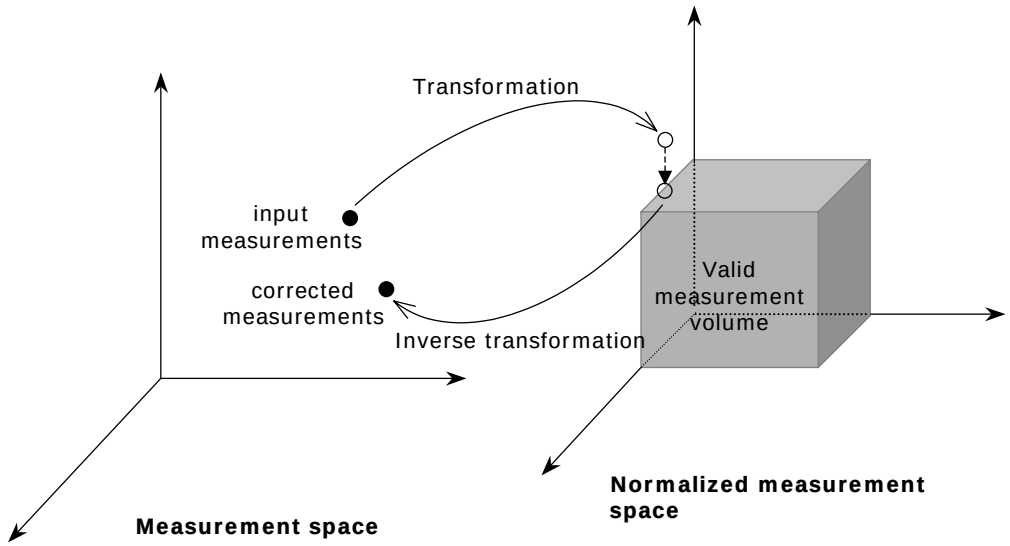


Figure 5-5. Determining the valid range of the measurement input in the normalized space.

Table 5-1. Anthropometric data collected from a population could be used for defining the valid range of measurement input.

Grouping

Group1	Group2	Group3
A. Stature B. Crotch height C. Chest girth	D. Arm length E. Waist girth F. Hip girth G. Neck girth	H. Arm girth I. Leg girth

Percentile of measurements (unit: mm)

A (Stature)

0.01	1%	20%	40%	50%	60%	80%	99%	99.9
1515	1519	1703	1742	1758	1774	1813	1908	2001

B (Crotch height, standing)

0.01	1%	20%	40%	50%	60%	80%	99%	99.9
667	730	797	823	834	846	872	939	1001

C♣ (Chest circumference at the

0.01	1%	20%	40%	50%	60%	80%	99%	99.9
708	807	912	954	972	990	1032	1137	1236

D (Arm length, Shoulder (Acromion) to Wrist, surface)

0.01	1%	20%	40%	50%	60%	80%	99%	99.9
529	572	618	637	645	654	673	726	774

E (Waist circumference)

0.01	1%	20%	40%	50%	60%	80%	99%	99.9
666	761	872	921	942	966	1023	1210	1368

F (Hip circumference, maximum)

0.01	1%	20%	40%	50%	60%	80%	99%	99.9
851	925	1012	1050	1066	1084	1128	1267	1386

G (Neck circumference, below the Adam's apple)

0.01	1%	20%	40%	50%	60%	80%	99%	99.9
328	354	384	397	402	408	423	467	506

H (Upper arm circumference, at the midpoint)

0.01	1%	20%	40%	50%	60%	80%	99%	99.9
240	271	306	322	329	336	354	412	461

I (Thigh circumference just below Buttock, standing)

0.01	1%	20%	40%	50%	60%	80%	99%	99.9

	456 506 565 591 602 614 644 742 825
Correlations among measurements (unit: mm)	A-to-D (height-armlen) 0.01% 20% 40% 60% 80% 99.9% min ma min max mi max mi max min max mi max 509 625 571 687 58 700 59 711 608 724 67 786 C-to-E (chest – waist) 0.01% 20% 40% 60% 80% 99.9% mi max mi max mi max mi max min max mi max 61 855 78 1024 82 106 85 110 907 115 11 1446 C–to-F (chest-hip) 0.01% 20% 40% 60% 80% 99.9% mi max min max mi max mi max min max min max 81 102 934 114 96 117 99 120 102 123 123 144 C-to-G (chest-neck): N/A E-to-H (waist-to-armgirth) 0.01% 20% 40% 60% 80% 99.9% min ma min ma min ma min ma min ma min ma 223 310 274 361 286 373 297 384 312 398 397 485 E-to-I (waist-to-thighgirth) 0.01% 20% 40% 60% 80% 99.9% min ma min ma min ma min ma min ma min ma 442 648 497 701 510 714 522 726 536 741 626 833

5.4 Skin attachment recalculation

Once the body shape has been modified through the displacement, the skin attachment data needs to be adapted accordingly so that the model retains smooth skin deformation capability. Generally, the deformed vertex location p is computed as

$$p = \sum_{i=1}^{n} w_i M_i D_i^{-1} p_d$$

where M_i and w_i are the transformation matrix and influence weight of the i-th influencing bone, D_i is the transformation matrix of i-th influencing bone at the time of skin attachment (in most cases D_i's are chosen to be so called dress-pose, with open arms and moderately open legs) and pd is the location of p at the dress-pose, described in global coordinate system.

Recomputing the skin attachment data involves updating the location p_d for each of its influencing bone. Note that the model has to be back into the dress-pose when the recalculation takes place.

5.5 Results and validation

We have used the proposed synthesizers to produce variously sized human body models. The system runs on a PC with Windows NT environment as a 3ds max™ (http::3ds max) plug-in. Given a new arbitrary set of measurements, an input size vector is determined from the normalization. With this input, the joint parameters as well as the displacements to be applied on the reference model are evaluated from the joint and displacement synthesizers, respectively. Figure 5-6 and Figure 5-7 show these models with corresponding input measurements listed in Table 5-2 and Table 5-3. Notice that some of the measurements used go beyond the range of parameter space that we captured from examples. All our models are animatable using motion sequence, through the update of the vertex-to-bone weight (see Section 5.4) that is initially assigned to the template model.

Table 5-2. Measurements used to generate male bodies shown in Figure 5-6.

Subject	Neck girth	Bust girth	Underbust	Waist girth	Hip girth	Height	Crotch length	Arm length
(a)	38	105	100	93	98	177	72	59
(b)	42	105	101	107	109	179	68	67
(c)	36	87	84	79	94	184	73	70
(d)	35	102	95	84	92	180	76	64
(e)	44	114	108	102	101	177	70	63
(f)	40	102	100	101	98	179	73	65
(g)	46	120	117	114	115	171	66	60
(h)	36	99	95	88	94	180	74	67
(i)	38	103	96	92	97	174	68	60
(j)	41	105	101	98	101	182	75	66
(k)	42	113	108	106	108	196	77	66
(l)	35	89	83	77	92	195	83	70
(m)	41	103	99	99	101	184	73	66
(n)	44	115	109	98	100	182	74	67

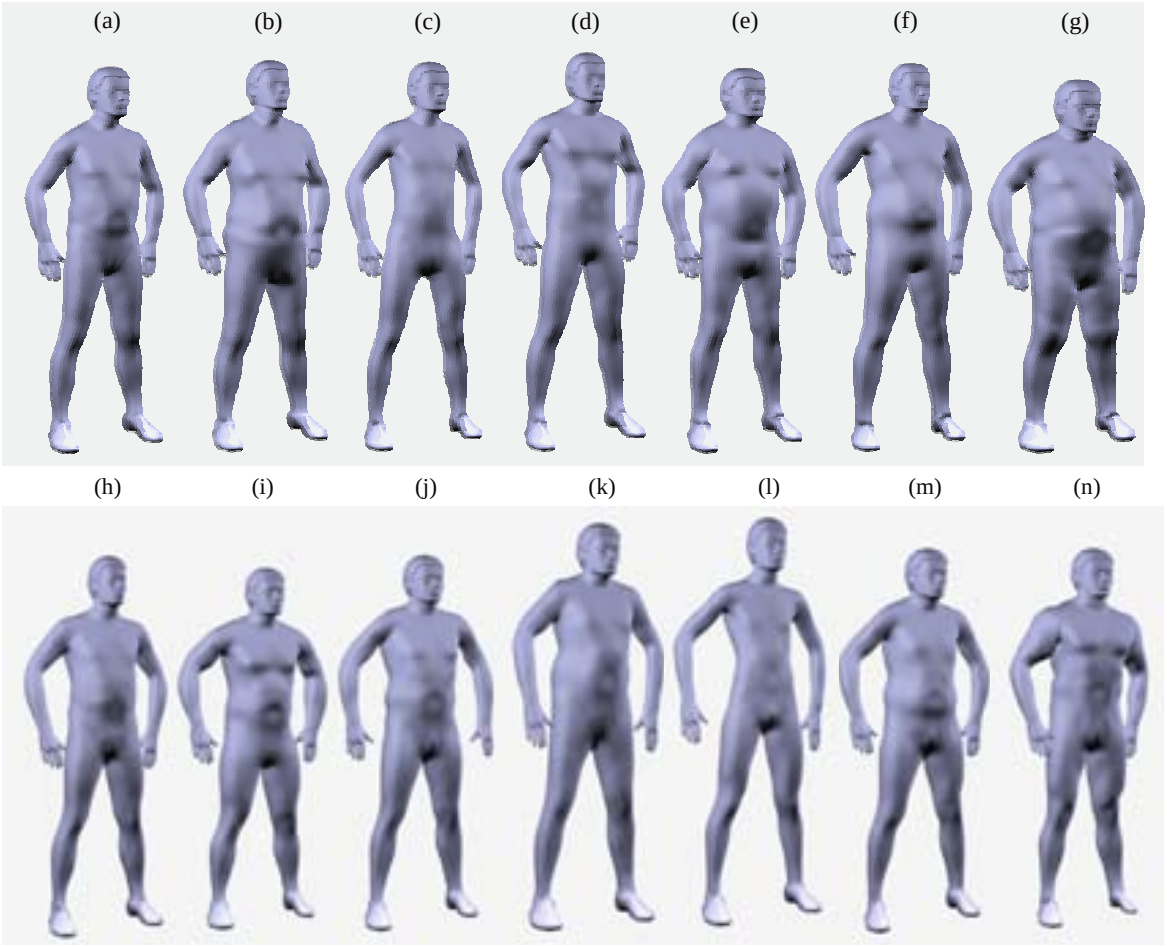


Figure 5-6. Male models generated from our modeling synthesizer using the input measurements listed in Table 5-2.

Table 5-3. Measurements used to generate female bodies shown in Figure 5-7.

Subject	Neck girth	Bust girth	Underbust	Waist girth	Hip girth	Height	Crotch length	Arm length
(a)	34	103	84	89	107	167	72	56
(b)	34	92	76	75	112	184	82	57
(c)	32	87	75	68	94	170	74	58
(d)	32	85	71	73	93	154	66	49
(e)	33	91	74	63	93	174	75	53
(f)	41	124	109	116	122	177	74	58
(g)	40	114	103	113	122	164	70	55
(h)	32	88	73	63	89	168	69	53
(i)	34	98	80	72	103	175	78	58
(j)	37	98	89	87	103	154	67	52
(k)	38	116	102	104	116	170	73	54
(l)	33	89	75	70	99	175	77	60
(m)	32	85	74	71	98	169	76	60
(n)	36	101	82	87	99	160	71	50
(o)	37	91	81	81	98	162	71	56
(p)	35	102	86	79	107	173	73	59

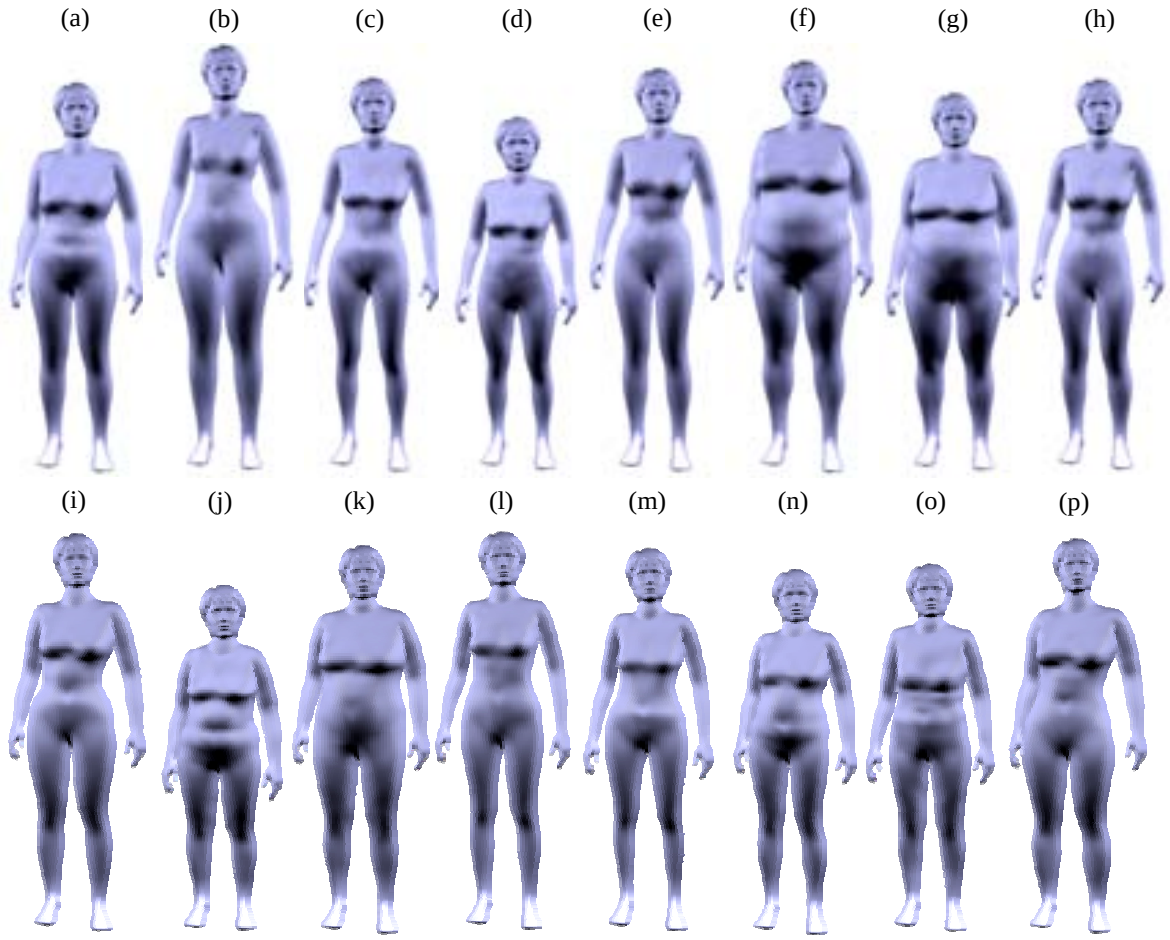


Figure 5-7. Female models generated from our modeling synthesizer using the input measurements listed in Table 5-3.

5.5.1 Computation time

When solving for the coefficients of the associated linear system of size n x n, with n the number of examples, we use LU decomposition (Press et al, 1988), yielding to the n^3 complexity.

In order to reduce the computation time as much as possible at runtime, optimization is made in two perspectives. Firstly, the exponential function, which is known to be expensive to evaluate, is implemented using a pre-calculated lookup table. It is essentially a three dimensional array, that is filled up by exponential function evaluation of the discretized variable – the radii. Each dimension of the array corresponds to the integer, the first and the second digit of the radius, respectively. Secondly, although the Gaussian is a localized basis function with the property that $\Phi \to \infty$ as $r \to \infty$, it is costly to evaluate and has infinite support – every interpolant point influences the whole transformation. If the influence of each interpolant is limited, the run time of applying the interpolation function can be reduced. Thus we have limited the range of influence and skip the computation of the interpolants whose distances are more than the maximum radius (which, in our case is set to 50).

The evaluation of 270 synthesizers (30×6 for the joints and 30×3 for the displacements) upon receiving the user input parameters takes approximately one second on a 1.0 GHz Pentium 3. This includes the time for the update of the vertex-to-bone weight that is initially assigned to the template model, so the synthesized models are immediately controllable using motion sequence. In Figure 5-8, a captured, key frame based motion data sequence is used for the animation of our models.

Figure 5-8. Animation of the resulting models.

5.5.2 Cross validation

Our modeling synthesizer not only results in visually pleasing body shapes but also faithfully reproduces models that are consistent with the input parameters. In order to quantify this performance, we report the results of the modeling synthesizers by cross-validation using all examples in the database. Each example has been excluded from the example database in training the synthesizer and was used as a test input for it. The output models have been measured and compared with the input measurements that are used to synthesize the model. For the five girth-related measurements, the average correlation coefficient was 0.980 and the average mean difference 0.623 (cm). The correlation coefficient and the mean difference are given in Table 1 for each measurement.

Table 5-4. Cross validation results of the modeling synthesizer.

	Bust girth	Underbust girth	Waist girth	Hips girth	Neck girth
Corr. Coeff.	0.973	0.984	0.989	0.982	0.974
Mean diff (cm)	0.654	0.585	0.692	0.618	0.567
Mean (cm)	92.765	82.970	67.776	93.056	32.091

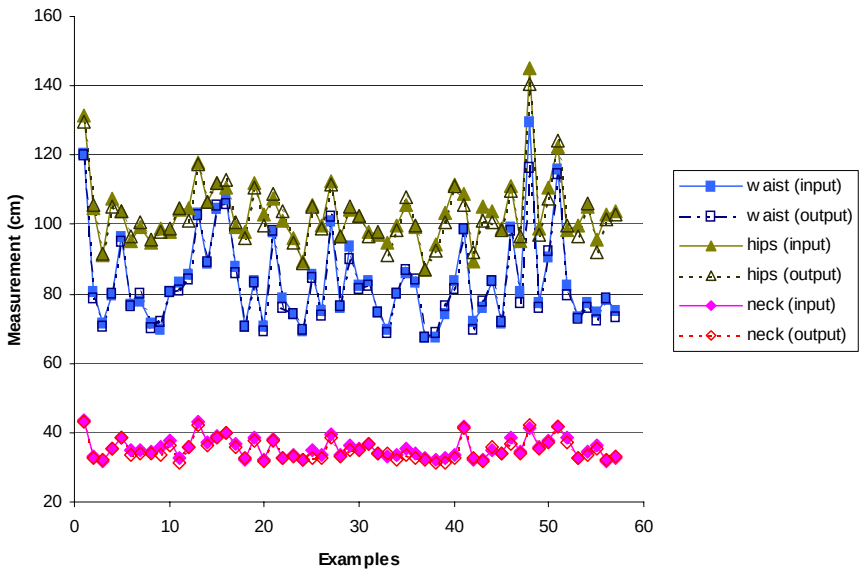


Figure 5-9. Cross validation results of the modeling synthesizer.

Chapter 6

An Evolution of an individual

"Our body is a machine for living. It is organized for that, it is its nature. Let life go on in it unhindered and let it defend itself, it will do more than if you paralyze it by encumbering it with remedies."

- Leo Tolstoy

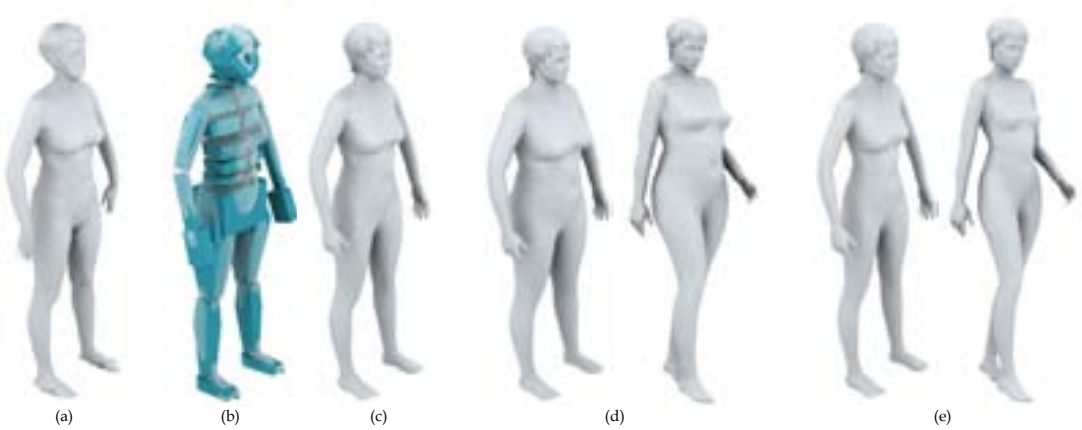


Figure 6-1. Modification of an individual body obtained by our modifier synthesizer. (a) scan data; (b) template model with animation structure is fitted to the scan data; (c) the fitted template mesh (d) modification of the physique (fat percent 38%); (e) modification of the physique (fat percent 22%).

In the previous chapter we looked for methods for synthesizing a new individual or a population model from sizing parameters. One main limitation of the modeling synthesizer is that a unique, identical model is produced given the same set of parameters. Often, we want to start with a particular individual to apply modifications according to certain attributes while keeping identifiable characteristics of the physique. Thus we must consider the case where we want to start with a particular individual to apply changes to according to certain attributes while keeping identifiable properties. A typical example of the parameterized shape variation could be: how a person looks like when she loses weight.

This challenging problem has recently been addressed in computer graphics, specifically in the domain of facial modelling. Blanz and Vetter (1999) use example database models from scanners and a linear function that maps facial attributes (gender, weight, and expression) onto the 3D model. Kähler et al (2002) make use of anthropometric data to calculate the landmark-based facial deformation according to changes in growth and age. Our goal here is to introduce body shape control for obtaining variation of the body geometry according to certain body attributes, whilst keeping the distinctiveness of the individual as much as possible.

We approach the problem as described in Section 6.3 by building regression models, using shape parameters as estimators and each component of the body vector as response variables. Because the changes in an individual body are generally global, it would be undesirable to use

size parameters (as used for the individual synthesizer) directly to control the change of a particular individual. Instead, we seek for intuitive parameters that give overall description of the physique and still can be represented as numerals. As explained in Section 6.1, we have chosen what we call control shape parameters (Waist-Hip Ratio, fat percentage…) to describe and simulate the change of a particular individual.

Our example models are however, relatively small (n=57) and skewed. Working with this data that contains unequal distribution may influence the coefficients of the regression models used for modifying the shape. Ideally, the distribution of the fat percentage should not be correlated to the height. We therefore perform the sample calibration prior to the regression modeling, as detailed in Section 6.2.

6.1 Shape parameters

Body attributes such as hourglass, pear/apple shape are typically those that provide a global description and dramatically reduce the number of parameters. Unfortunately, it is difficult to find an objective metric to precisely describe these attributes numerically. More importantly, they cannot directly be used to describe a population. Thus, we choose parameters whose statistical information is available from literature such as anthropometry and somatotyping.

Somatotype (Carter and Heath, 1990) as opposed to anthropometry presents a metric to quantify the shape and composition of the body. A three-dimensional system is used to describe the size-dissociated properties of the body, namely endomorphy, mesomorphy, and ectomorphy. Endomorphy, the first component, is a rating on a continuum of relative fatness of a physique. Mesomorphy, the second, is a rating on the relative predominance of muscle, bone and connective tissue. Ectomorphy, the third component, means relative linearity or ‘stretched-outness’ of a physique. In Figure 6-2, seven example bodies of drastically different sizes are illustrated along with their somatotype ratings.

There are two major problems in adopting the somatotype parameters directly. First, it requires inspection by a highly skilled somatotyper in order to obtain a consistent rating, since the main part of the rating procedure is ‘photoscopic’ that is based on comparative observation of photographs. Second, it additionally requires some measurement information that is not available from the scanned data. For example, in the ‘anthropometric somatotyping’, skinfolds of four selected parts of the body are required to give an endomorphic rating.

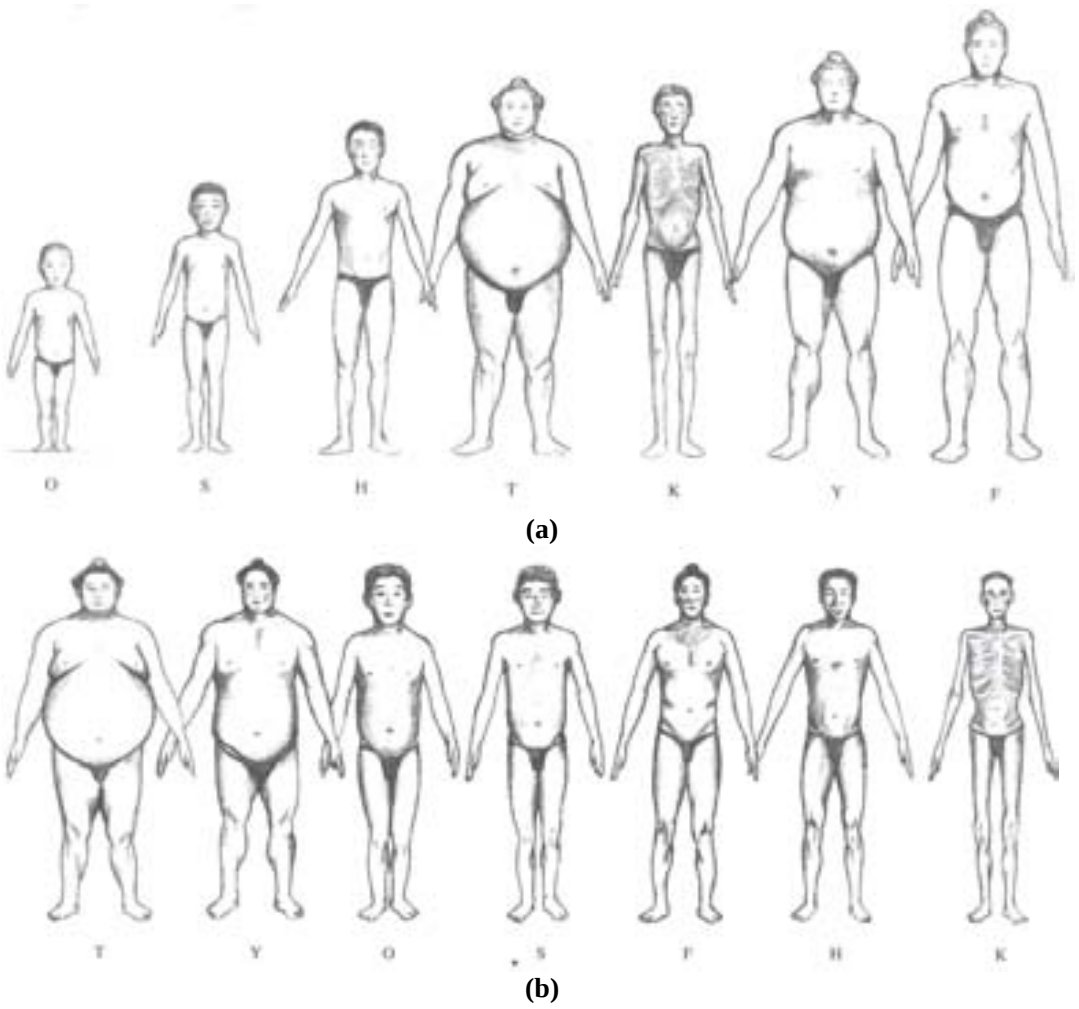


Figure 6-2. Seven Japanese males of vastly different size drawn (a) to absolute size, and (b) redrawn to the same height (Heath and Carter, 1988).

Table 6-1. Height, weight, Height-Weight ratio, and somatotype of seven Japanese males of different absolute sizes for subjects shown in Figure 6-2(b) (Heath and Carter, 1988).

Subjects	T	Y	O	S	F	H	K
Height (cm)	174.0	182.0	101.3	119.8	207.0	156.0	174.0
Weight (kg)	166.0	139.0	18.3	25.3	127.0	52.0	42.0
HWR	31.67	35.14	38.44	40.81	41.19	41.80	50.06
somatotypes	10-7-0.5	6.5-7.5-0.5	5-5-1	4.5-4.5-1.5	3-6.5-2	3.5-4.5-2	1-1-7

Fortunately, there are a number of empirical results that allow us to transform the somatotyping system into more accurate measurements. First, mesomorphy and ectomorphy are closely related to the HWR (height-weight ratio) (Carter and Heath, 1990). Second, endomorphy is essentially the fat percentage, which can be estimated through a number of anthropometric measurements (Hodgdon and Friedl, 1999). Therefore, we adopt the following parameters to control the body shape:

- Fat percentage (%): According to Hodgdon and Friedl (1999), it is possible to estimate the fat percentage of the body from a number of anthropometric measurements. The regression equation used to estimate the fat percentage of a Caucasian man and a woman are:

$$86.010 \times \log_{10}(abdomen - neck) - 70.041 \times \log_{10}(height) + 36.76$$

and

$$163.205 \times \log_{10}(abdomen + hip - neck) - 97.684 \times \log_{10}(height) - 78.387,$$

respectively.

- Height: Since the weight measure of subject data is not available, we cannot derive the HWR (Height divided by the cube root of weight) for subjects in the database. Instead, we simply use the height, the control of which is useful in practice.

- WHR (Waist-Hip ratio): Waist girth divided by the hip girth, gives a good deal of information on the torso shape and is perhaps the best quantification of the colloquial body attributes such as hourglass, pear/apple shape.

6.2 Weighting adjustment for the calibration of the sample

We observed that tall-slim and short-overweight bodies are overrepresented in our database, exhibiting a high correlation between the height and the fat percentage (r =-2.6155, p <0.0001). Directly using such data that contains unequal distribution may result in false estimation by the modifier synthesizer. We use sample calibration (DeVille and Särndal, 1992) to avoid such erroneous estimation.

Sample calibration (DeVille and Särndal, 1992) has become a popular and powerful framework in survey estimation. This technique improves the representativeness of the sample in terms of the size, distribution and characteristics of the population by giving a weight to each element of the sample. In our case, we wish to compute the weights for each sample so that the conditional distribution of the body fat percentage given the height of the sample is the same as those of the population. That is, we want the linear function that maps the height to the fat percentage has the slope 0.

Consider a sample consisting of n elements. Associated with each element k are a target variable y_k and a vector $\mathbf{x}_k$ of p auxiliary variables. Consider also that $\mathbf{x}_k$'s are correlated with y_k by the regression $Y = XB$, where $(X)_{kj} = x_k^j$ denotes the jth variable of element k, Y denotes the vector of n target variables, and the correlation vector $B = [b_1, b_2, .., b_n]^T$ is known for the population.

The calibration method aims to compute a weight w_k for each element so that the sample distribution of the variables X agrees with the population distribution.

The calibration problem can then be formulated as follows:

Problem 6-1: Minimize the distance

$$\sum_{k=1}^{n} G(w_k) \quad (6.1)$$

subject to a calibration constraint defined by the weighted least square equation:

$$X^T W_m X B = X^T W_m Y \quad (6.2)$$

where

$$X = \begin{bmatrix} x_1^1 & \cdots & x_1^p \\ \vdots & \ddots & \vdots \\ x_n^1 & \cdots & x_n^p \end{bmatrix}, \qquad Y = \begin{bmatrix} y_1 \\ \vdots \\ y_p \end{bmatrix},$$

and $W_m = \text{diag}(w_1, \ldots, w_n)$.

The so-called distance function ***G*** measures the difference between the original weight values (uniform weighting of 1.0 in our case) and the new weights wk. The objective is to derive w_k 's that are as close as possible to the original weights.

In our case, the regression model is linear (p =1), x_i^j and y_i are the fat percentages raised to the power of j - 1 and the height of i -th example, respectively. The coefficient vector B is $[b_1, 0]^T$. Taking these values, the development of the Equation (6.2) yields:

$$b_1 \sum_{k=1}^{n} (x_k^1 w_k) = \sum_{k=1}^{n} (w_k x_k^1 y_k)$$

$$\vdots \qquad\qquad \vdots$$

$$b_1 \sum_{k=1}^{n} (x_k^p w_k) = \sum_{k=1}^{n} (w_k x_k^p y_k)$$

By rewriting the equation above using matrix convention, we have:

$$L \cdot W_v = \begin{bmatrix} x_1^1(b_1 - y_1) & \cdots & x_n^1(b_1 - y_n) \\ \vdots & \ddots & \vdots \\ x_1^p(b_1 - y_1) & \cdots & x_n^p(b_1 - y_n) \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix} = 0 \qquad (6.3)$$

The linear system in (6.3) is an underdetermined, with the condition of $n > p$. These weights have to be computed in a way that the equation (6.1) is minimized. To find a solution to (6.3) with (6.1) minimized, we replace w_k with $w'_k + 1$ and rewrite (6.3) as:

$$L \cdot W'_v = -\begin{bmatrix} \sum_{k=1}^{n} x_k^1(b_1 - y_k) \\ \vdots \\ \sum_{k=1}^{n} x_k^p(b_1 - y_k) \end{bmatrix}, \qquad (6.4)$$

where

$$W'_v = \left[w'_{1,\ldots,} w'_n\right]^T .$$

In addition, we add one more equation

$$\sum_{k=1}^{n} w_k' = 0$$

to (6.4), since we want the average of the weights w_k to be equal to 1. The distance function $\boldsymbol{G}$ used in our approach is the quadratic function $G(w_k) = \frac{1}{2}(w_k - 1)^2$ (DeVille and Särndal 1992). One advantage of such function is that minimizing this function remains as minimizing the norm of the solution vector $\|W_v'\|$. The underdetermined linear system can then be solved using the least norm solution, which involves Cholesky decomposition (Press et al, 1988) of the composite matrix. The least norm solution to the linear system $Ax = b, A \in R^{m \times n}, m \le n$ is

$$x = A^T (AA^T)^{-1} b,$$

and can be found by the following procedure:

Form $C = AA^T$ ($m^2 n$ flops)

Cholesky factorization $C = LL^T$ ((1/3) m^3 flops)

Forward substitution: solve $Lw = b$ (m^2 flops)

Backward substitution: solve $L^T z = w$ (m^2 flops)

Multiply with A^T: $x = A^T z$ ($2mn$ flops)

Upon calculation of w_k' using the procedure described above, we finally obtain the weights w_k from $w_k = w_k' + 1$. Throughout the modifier construction phase which is the topic of the next section, all our examples are assumed to be weighted using these.

6.3 Parameterized variation of an individual

Similarly with the problem of the parameter-driven modeling synthesizer, as described in previous chapter, we wish to obtain the modification of the body geometry as a function evaluation of the chosen parameter, in our case fat percentage and HWR. Clearly, we can make use of the examples to derive such function. Unlike the modeling synthesizer, however, we would like some of the attributes, i.e. those which characterize a particular individual model to remain unchanged as much as possible during the modification, whilst other attributes are expected to be changed according to the control parameter.

In general, it is difficult to identify what is invariant in an individual body through changes from various factors (sport, diet, aging, etc.). In addition, 3D body examples of an individual under various changes in his/her appearance are rare. (Although one can relatively easily obtain such examples when the aim is to build pose parameter space, as used in (Allen et al, 2002). Therefore, the problem of identifying (a) geometric elements that characterize an individual and (b) controllable elements from the given examples needs to be solved.

To quantify the correlation between the shape parameter and the body shape, we ran the following regression model for each element of the body vector (see Section 4.2).

$$Y_i = \beta_i X + c_i, (i = 1,2,..30)$$

The formulation of the regression model based on the weighted least square method is:

$$W_m Y_i = \beta_i W_m X + c_i,$$

where

W_m : the sampling weight matrix defined in Section 6.1,

X : the shape parameter,

Y_i : the *i*-th element of the body vector, b_i ,

β_i : the coefficients of the regression function Y_i ,

c_i : the intercept of the regression function Y_i .

We used the regression data analysis tool in Mircosoft Excel to estimate the regression parameters. The explanary variable had to be significant at $p < 0.05$ and those models that do not have significant correlation ($p \geq 0.05$) are discarded by the synthesizer.

Table 6-2 shows regression tests for the first 6 principal components of the 'scaleX' joint parameter with the fat percentage as the explanary variable. Note that each example data (the shape parameter and the body vector pair) is multiplied by the weight value that has been calculated from the sample calibration. In Figure 6-3, the first 6 principal components of the 'scaleX' joint parameter in relation with the fat percentage variable are shown.

Table 6-2. Regression models of 'scaleX' components of the body vector with the fat percentage as the explanary variable. Significant regression model is marked with '*' symbol.

Analysis	regression	standard	*F*-ratio	*p*-value
PC1 of scaleX	-0.00973	0.00846	1.32119	0.255444
*PC2 of scaleX	-0.02849	0.003272	75.82538	7.23e-12
PC3 of scaleX	-0.00354	0.00219	2.62024	0.11134
PC4 of scaleX	-0.00047	0.00197	0.05686	0.81244
*PC5 of scaleX	0.003328	0.001485	5.024998	0.029115
PC6 of scaleX	3.20e-05	0.00136	0.00056	0.98127
*PC7 of scaleX	-0.00261	0.001122	5.420798	0.023673
PC8 of scaleX	0.00080	0.00098	0.66361	0.41887
*PC9 of scaleX	-0.00184	0.000738	6.186507	0.015994
PC10 of scaleX	0.00024	0.00069	0.11998	0.73040

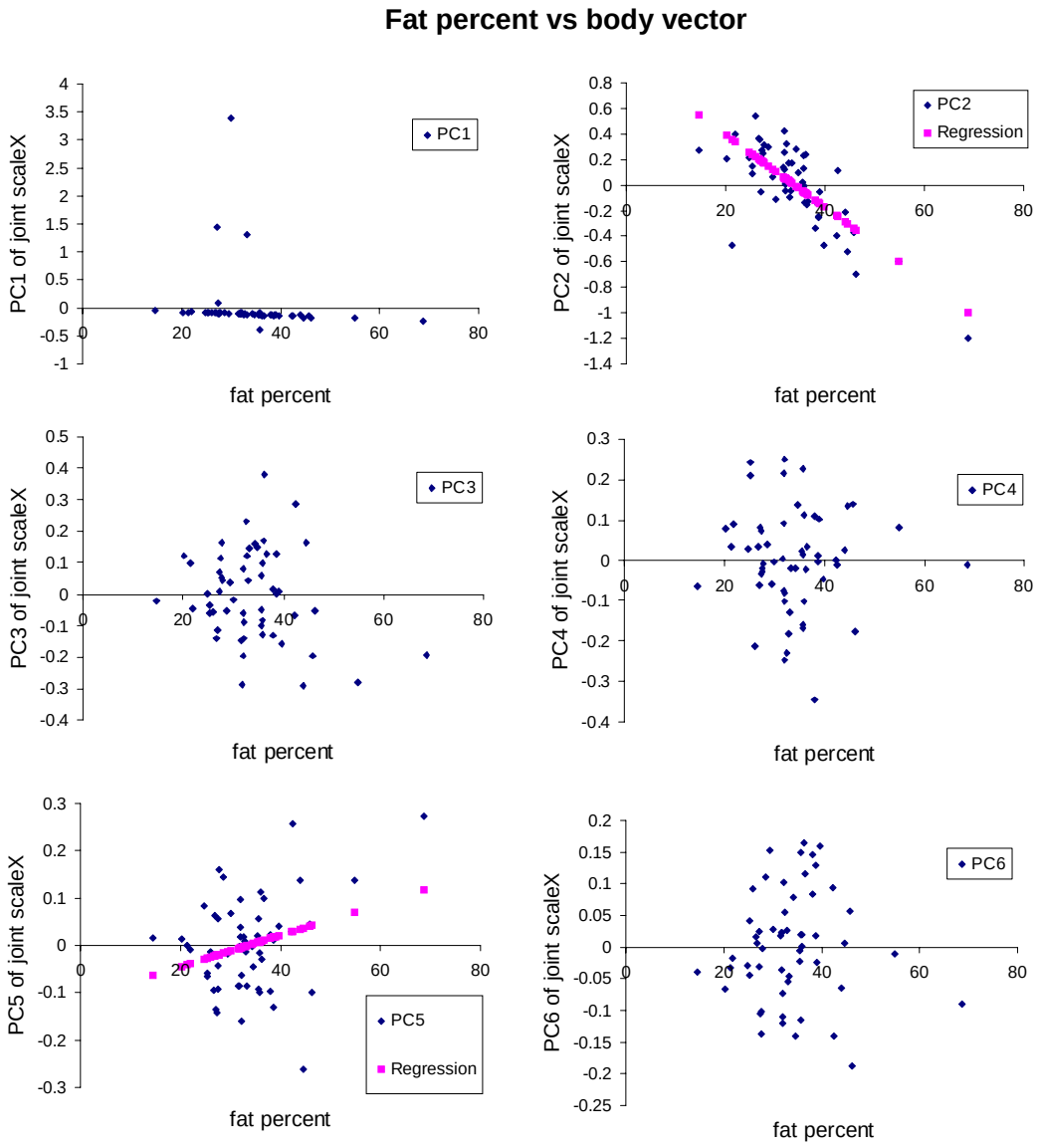


Figure 6-3. Regression analysis of the 'scaleX' body vector and the fat percentage, for the female subjects.

Now starting from a model whose i-th body vector value is $b_{i,src}$ and the desired shape parameter x_{trg}, we can derive a new body vector value $b_{i,trg}$ as follows. For the initial shape parameter x_{src} of the given model, the regression function $E_i(x)$ gives the average value $E_i(x_{src}) = b_{i,src}$ (see Figure 6-4). The difference $e_{i,src} = b_{i,src} - b_{i,src}$ is called the residual of the regression. We consider this residual as the distinctiveness of the body, i.e. the deviation of the body vector component from its average value. Similarly to Kähler et al (2002), we assume that the body component keeps its variance from the statistical mean over the changes: a shoulder that is relatively large will remain large. It is worth noting that by computing the regression functions $E_i(x)$ for every b_i, we can compute the average body for the given shape parameter value x.

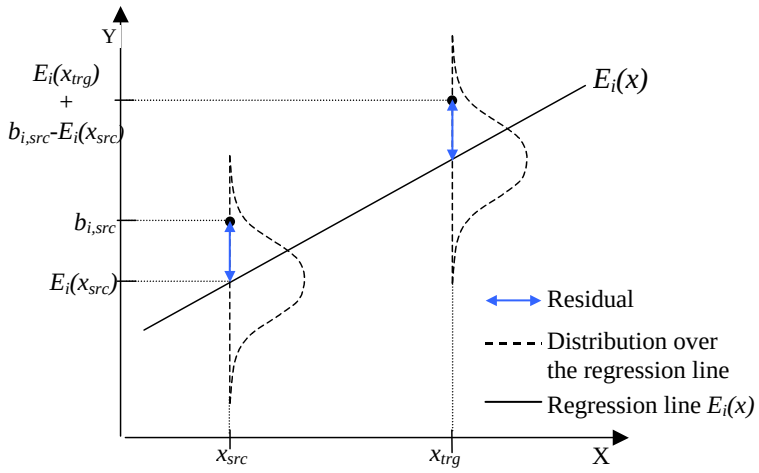


Figure 6-4: Shape variation with the regression line and the residual.

Therefore, for a given body vector for which we know the shape parameter, we can compute the residual value of each the components of the body vector. The new body vector is then obtained by adding those residuals to each of the average body vector for the desired shape parameter. Formally, given current and desired shape parameters, x_{src} and x_{trg}, the new value of i-th body vector $b_{i,trg}$ is given by:

$$b_{i,trg} = E_i(x_{trg}) + \left(b_{i,src} - E_i(x_{src})\right).$$

6.4 Results

Table 6-3 and Table 6-4 summarize the regression models used in our modifier synthesizer. While there exists a good deal of consistency, our regression models on the body vector should be differentiated from those for anthropometric body measurements. The regression model of 'scaleX' of the bone, for instance, shows that the physical fat of subjects is partly captured by, and interpreted as, the linearity in the model.

Figure 6-5 shows the two female subjects that have been deformed by our modifier synthesizer using various shape control parameters. It is clear that the bodies remain identifiable during the modification. Again, all our models remain animatable during the manipulation, through the update of the vertex-to-bone weight that is initially assigned to the template model.

Body vectors and their distribution also depend on the initial skinning. When there is a strong weight on the upper arm, a small-scale value will be sufficient to result in a large arm while a large scale is required to have the same effect when the attachment is loose. An automatic refinement of the skin attachment to obtain an optimal skinning could be found, so that the variation of body vectors for all examples is minimized, for instance.

Table 6-3. Regression models of joint components of the body vector with the fat percentage as the explanary variable

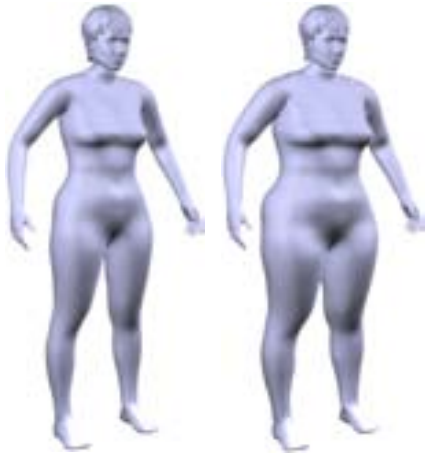
Explanary variable	regression	standard	*F*-ratio	*p*-value
PC2 of scaleX	-0.02849	0.003272	75.82538	7.23e-12
PC5 of scaleX	0.003328	0.001485	5.024998	0.029115
PC7 of scaleX	-0.00261	0.001122	5.420798	0.023673
PC9 of scaleX	-0.00184	0.000738	6.186507	0.015994
PC1 of scaleY	0.043125	0.004694	84.39235	1.26e-12
PC1 of scaleZ	-0.00975	0.003704	6.926509	0.011048
PC2 of transX	0.025454	0.012929	3.876025	0.054122
PC3 of transX	0.02357	0.01203	3.838827	0.05525
PC6 of transX	0.022441	0.007838	8.196972	0.005961
PC3 of transY	-0.06049	0.018584	10.59697	0.001958
PC5 of transZ	0.07172	0.025202	8.098249	0.00625
PC8 of transZ	0.03009	0.014531	4.287934	0.043178

Table 6-4. Regression models of displacement components of the body vector with the fat percentage as the explanary variable

Explanary variable	regression	standard	*F*-ratio	*p*-value
PC1 of displacementX	-2.9363	0.039295	8.621829	0.004872
PC5 of displacementX	-0.06207	0.029464	4.438548	0.039797
PC6 of displacementX	0.121447	0.024977	23.64189	1.04e-05
PC1 of displacementY	0.214258	0.04131	26.90133	3.3e-06
PC4 of displacementY	-0.07232	0.03268	4.897688	0.031139
PC5 of displacementY	0.107049	0.033195	10.39955	0.002141
PC2 pf displacementZ	-0.08119	0.028622	8.046944	0.006405
PC3 of displacementZ	-0.06428	0.025047	6.58599	0.013085
PC10 of displacementZ	-0.064	0.012807	24.97014	6.48e-06

(a) Original model

Fat % : 30%, 45%



Waist- Hip Ratio : 0.801, 0.953

Height : 160cm, 178cm

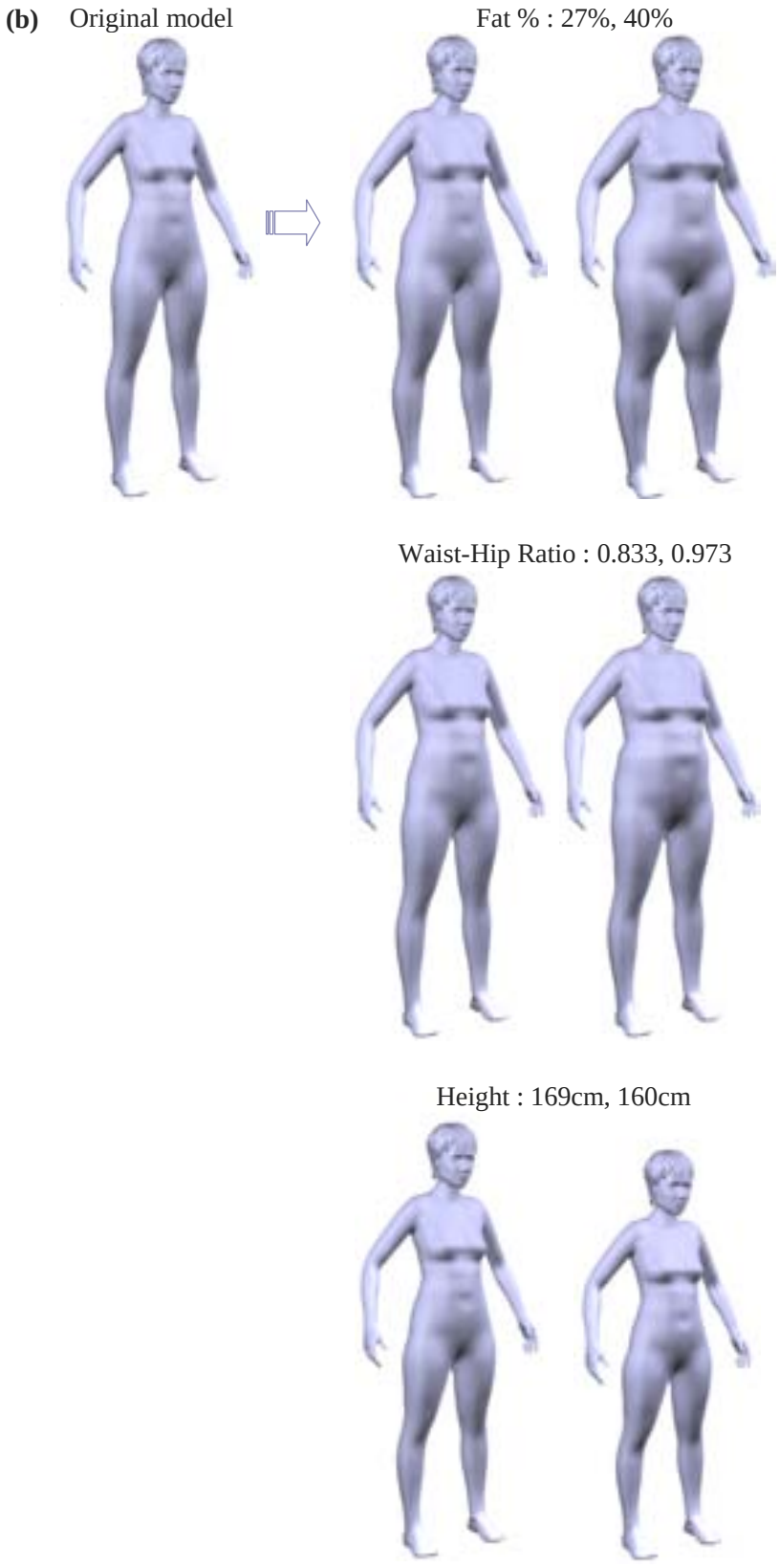


Figure 6-5. Modification of two individuals, (a) and (b), controlled by fat percent, waist-hip ratio and height.

Chapter 7

Implementation and Results

"Beware of the bugs in the above code, I have only proved it correct, not tried it."

Donal Knuth

So far, in Chapters 4, 5, and 6, we have looked for theoretical framework for preprocessing and manipulating body models using size and shape parameters. In this chapter we describe software implementation of the proposed methodologies, which largely divides into preprocessing and the runtime application. In Section 7.1 we give the implementation detail of the preprocessing that involves the 3ds max plugin development. 3ds max, a commercial graphics modeling and animation software, holds large market both in industries and educational organizations. Also outlined in the same section is the workflow of the process for the plugin users. In Section 7.2 the implementation of the proposed synthesizers is described in the frame of runtime or end-user application. It takes VHD++ (Magnenat-Thalmann et al, 2002) as an integration platform, MIRALab's in-house system for real-time scene loading, navigation and virtual human manipulation. The real-time graphics visualization and scene graph management of VHD++ is based on Cosmo3D and OpenGL Optimizer.

In both cases, we have used C and C++ as the programming language and MFC (Microsoft Foundation Class) library as the user interface development tool.

7.1 Preprocessing

7.1.1 3DS Max plug-in structure

3ds max offers a mature set of SDK (Software Development Kit), object-oriented programming libraries that enable developers to extend its functionality by creating plug-in applications. Using the SDK one can combine and extend to create seamlessly integrated plug-in applications. In fact, much of 3ds max itself is written as plug-in applications.

There are three kinds of plug-ins – creation, modifiers and utility plug-ins. Located at a specific directory, plug-ins are loaded in the 3ds max system when the application launches, or can be dynamically loaded and unloaded using the Plug-in Manager.

3ds max also has a built-in scripting language called MaxScript, which can be used by non-programming users to implement plug-ins using. It is also useful to repetitive tasks, which is also used in our case for the automation of the final exportation of the data files.

A pipeline is the system used by 3ds max that allows a node in the scene to be altered, perhaps repeatedly, through the application of modifiers. For the system to evaluate the state of the object at the end of the pipeline, it must apply each modification along the way, from beginning to end. Naturally, a stack of modifiers added on a graphical object represents a

series of modification applied to the object. Each modifier takes the outcome of the previous modifier result and provides the income object to the next modifier in the stack. Note that the order of modifiers in the stack affects the intermediate and final object.

7.1.2 System overview

Table 7-1 lists seven plugins pertaining to the preprocessing phase. The details are as follows.

Table 7-1. Various plug-ins related to Preprocessing.

Plug-in name	Plug-in type	Description
FastSkin	modifier	Performs skeleton driven deformation using the imported attachment data.
Feature Points	modifier	Identifies feature points and contours on a geometric mesh object of EditMesh type.
Scan Fitter	utility	Based on the feature points defined by the FeaturePoints modifier, and on the skeleton driven deformation by the FastSkin modifier, finds the global proportion and posture of the template model.
OffsetDeformer	modifier	When used after the ScanFitter, OffsetDeformer captures the fine detail of the target scan model and save it as a displacement map. Also contains a number of relaxation functionalities.
QuadPatch	modifier	Reorganizes the vertex coordinate of geometric object of EditMesh type such that the quadratic elements precede triangular ones.
BodyManager	utility	Contains various tools to convert, import and export standardized (H-Anim) body models and motion data.
Spline Surface	modifier	Performs Bezier subdivision on the target geometric object of QuatPatch type.

7.1.2.1 FastSkin modifier

The MIRALab FastSkin modifier plugin is a modified version of Skin modifier in 3D Studio MAX with enhancements on advanced parameters and import/export facilities. The Skin modifier is a skeletal deformation tool which allows the user to deform one object with another one. For example, the mesh, patch objects can be deformed by Bones, splines or other objects. After applying the Skin modifier and assigning bones, each bone has a capsule-shaped "envelope." (See Figure 7-1) Vertices within these envelopes move with the bones. Where envelopes overlap, vertex motion is a blend between the envelopes.

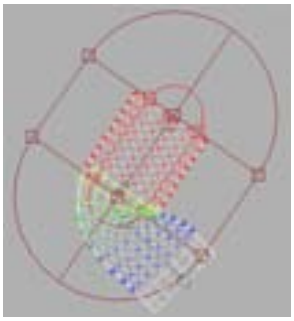


Figure 7-1. The skeletal driven deformation – the ‘envelope’ represents the influence region of the bone. Different colors are used to illustrate the vertex weight.

The MIRALab FastSkin modifier has added two important functions to the Skin modifier: Import/Export of skin attachment data and Segmentation. Import/Export enables the user to read and write the attachment data, allowing for the flexible reuse of the once-made work. The supported skin attachment data file format is ‘.atc’.

The segmentation takes place prior to the H-Anim exportation of the prepared model. Basically it segments the skin mesh by grouping those vertices that share the same bone as its most influencing bone. It then locates each segment into the skeleton hierarchy as a proper child node of the corresponding joint. The segmentation algorithm is developed by Frederic Cordier and is described in (Seo et al, 2000). Figure 7-2 illustrates the user interface panel of the FastSkin modifier.

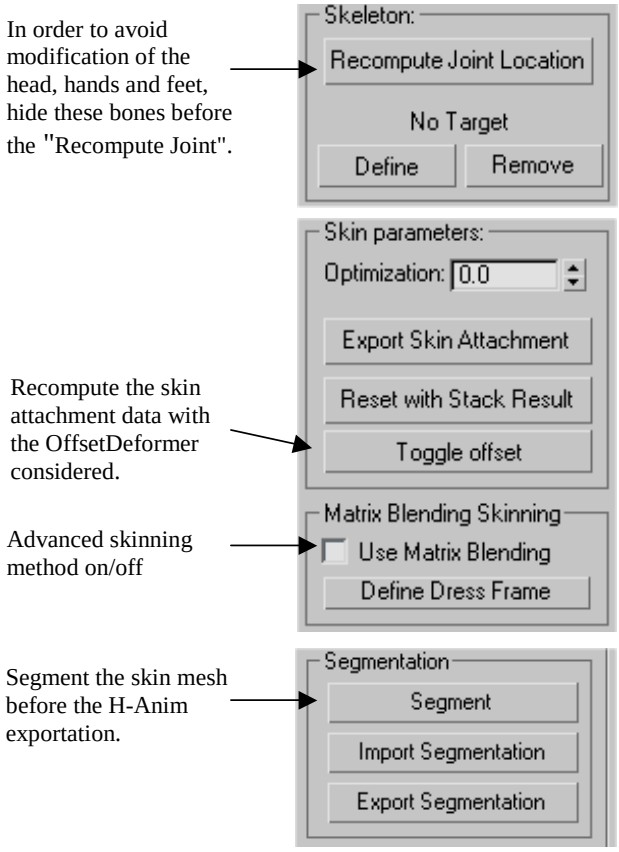


Figure 7-2. The MIRALab FastSkin modifier.

7.1.2.2 FeaturePoints modifier

Figure 7-3 shows FeaturePoints modifier plug-in in action. It is mainly for identifying feature points and counters on the target geometric object of type EditMesh.

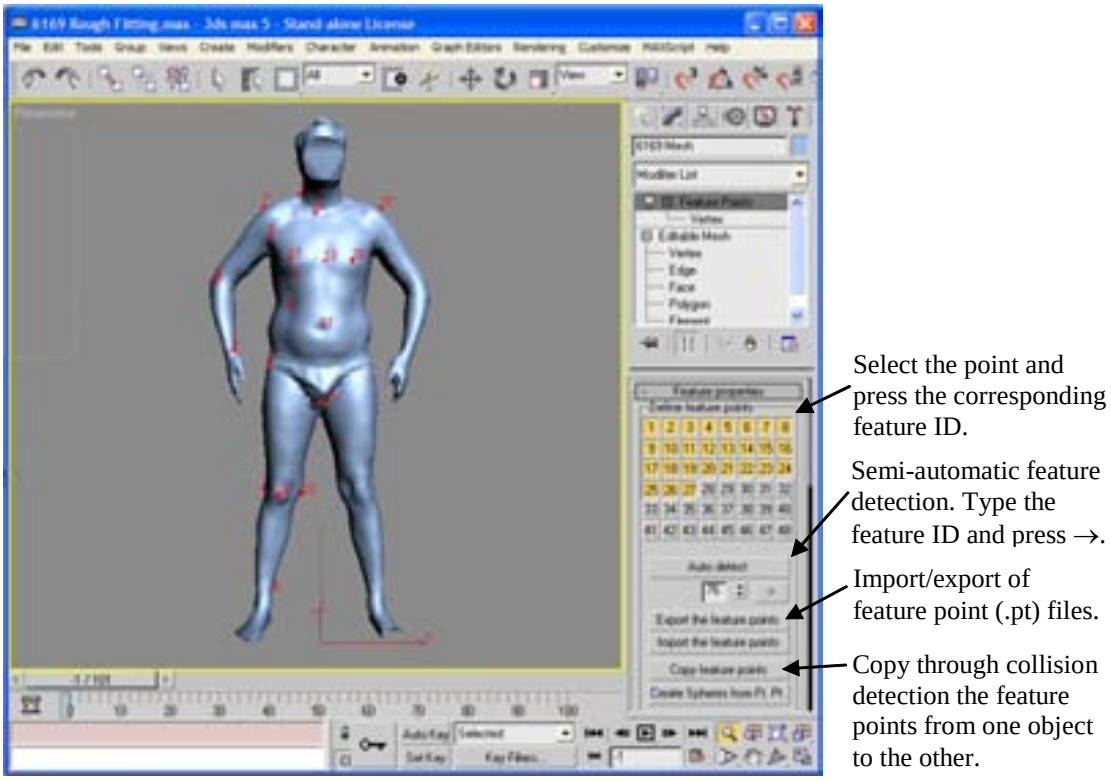


Figure 7-3. FeaturePoints modifier is for defining the feature points on the EditMesh object.

7.1.2.3 Scan Fitter utility

Is essentially the implementation of the skeleton fitting process described in Section 4.3. It calculates the skeleton configuration of the template model that best approximates the target scan data, by making use of the feature points. Needless to say, both the scan model and the template model are assumed to have FeaturePoints modifiers that are mutually consistent.

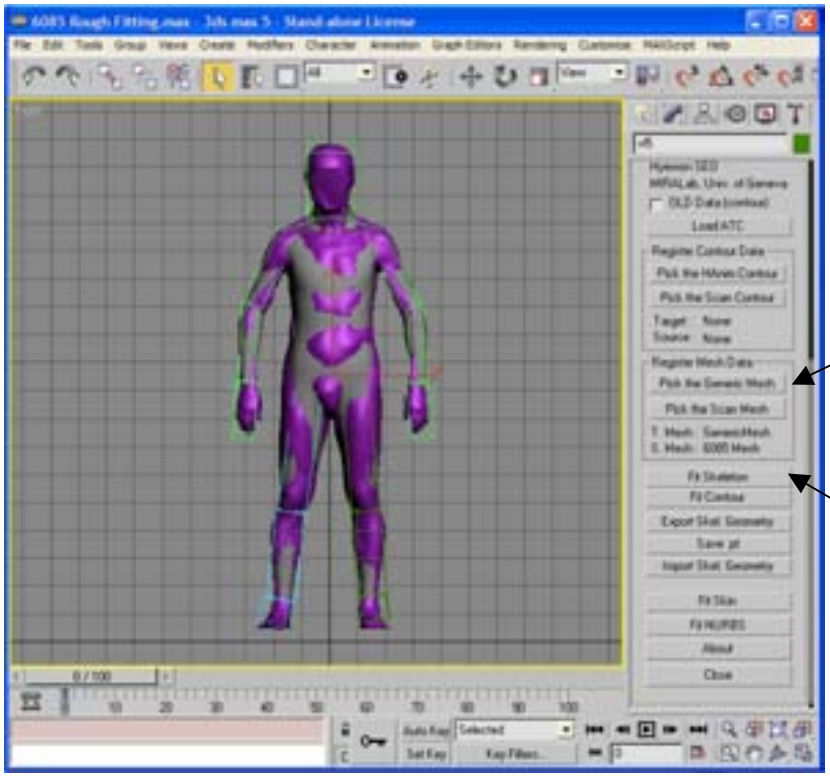


Figure 7-4. The ScanFitter utility plugin automatically adjusts the template skeleton by using feature points that have been previously defined.

7.1.2.4 OffsetDeformer modifier

Implementation of the mapping and the relaxation methods as described in Section 4.4. By the time of application, the template model is assumed to be in sufficient proximity to the target scan mesh through the skeleton adjustment, although the approximate level of the surface distance can be informed to the modifier by adjusting the maximum possible 'collision distance' in the panel. When there is the FeaturePoints modifier that has been applied to the template model prior to this modifier, those feature points are used for exact mapping, i.e. they are anchored to their corresponding feature locations. Typically, those points that one want to make sure of exact mapping are the breast and the navel. Again, the feature points on these two meshes must be consistent. Contours in the FeaturePoints modifier are used also to ensure the horizontal correspondence between the target and the source. For the data we have demonstrated in this dissertation, we have used the waist and the breast girths for the mapping purpose. The interface panel of the OffsetDeformer is shown in Figure 7-5 and Figure 7-6.

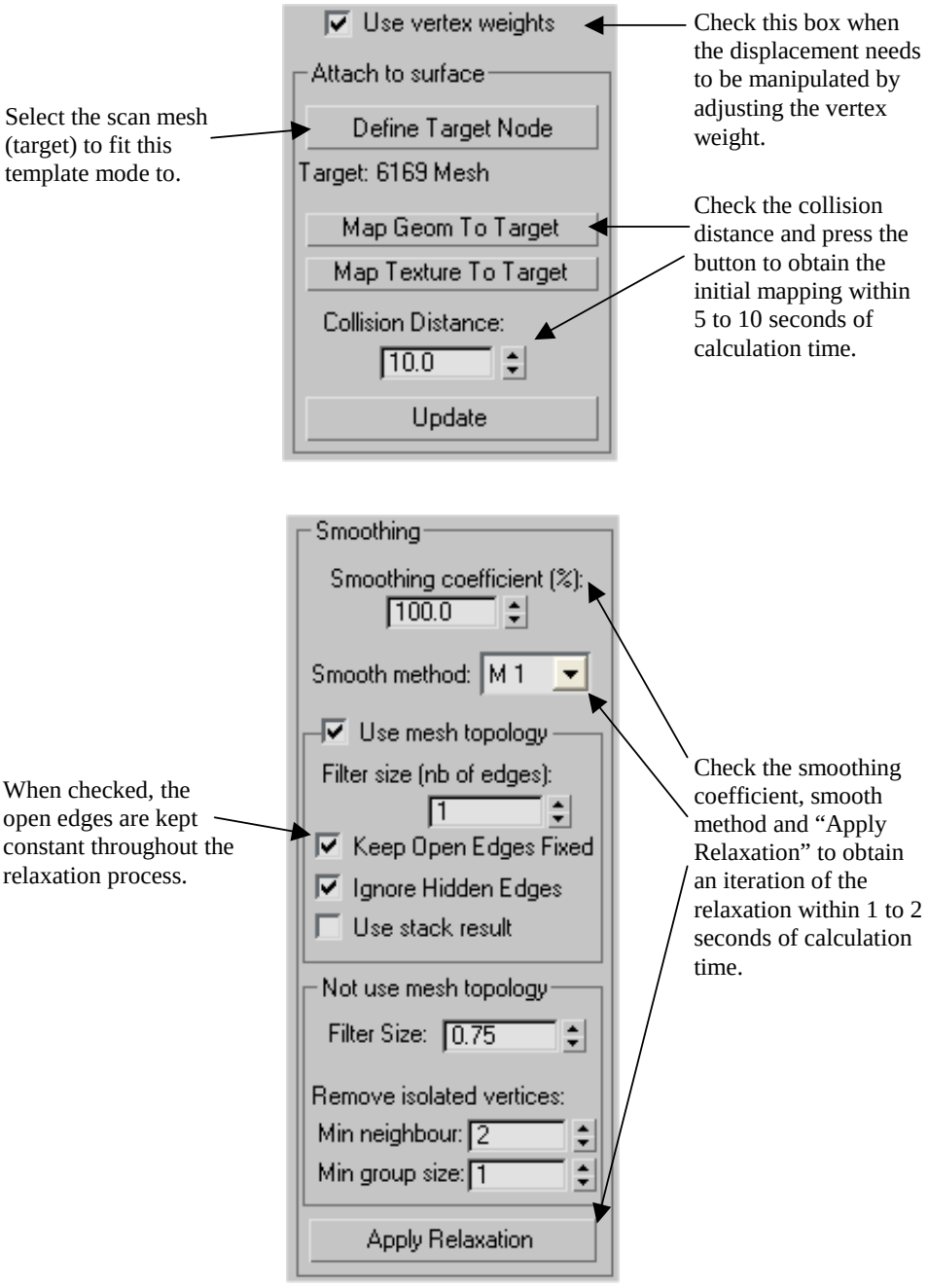


Figure 7-5. The plugin planel of the OffsetDeformer modifier concerning the mapping and the relaxation of the template mesh.

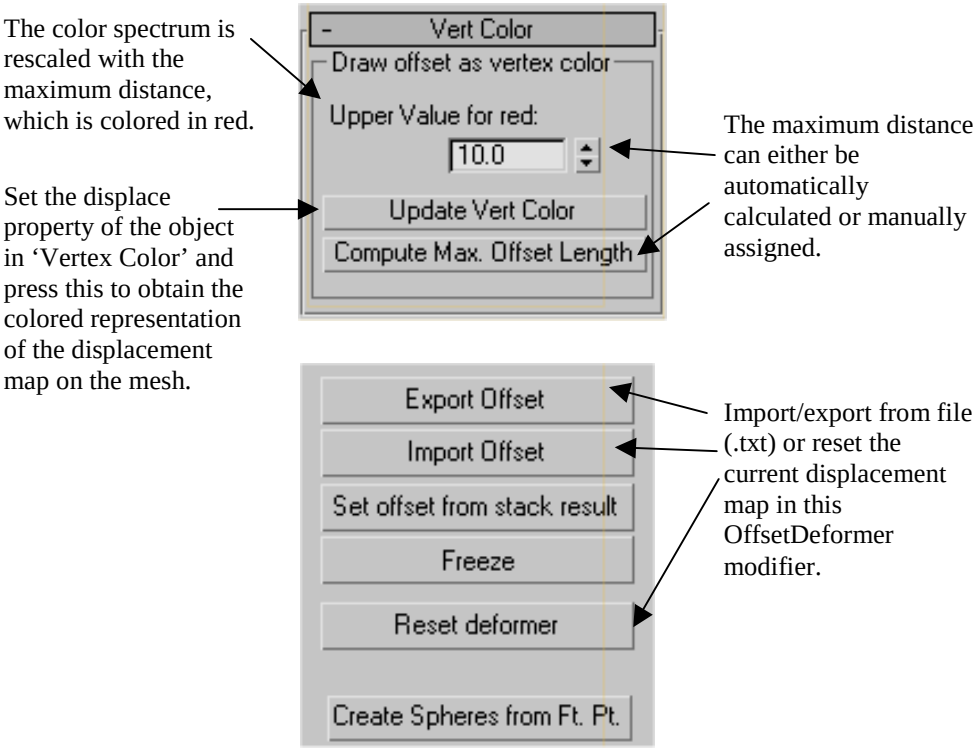


Figure 7-6. The plugin planel of the OffsetDeformer modifier concerning the vertex coloring and the file I/O of the displacement map.

7.1.2.5 QuadPatch modifier

Reorganizes the vertex coordinate of geometric object of EditMesh type such that the quadratic elements precede triangular ones. This permits an easy reuse of the manipulation by independently treating the head, hands and the feet from the body part. For instance, the head model can be modified or replaced by another one of different topology without having to repeat what had been made on the body part of the template mesh. Figure 7-7 shows the application of QuadPatch modifier to our male template model.

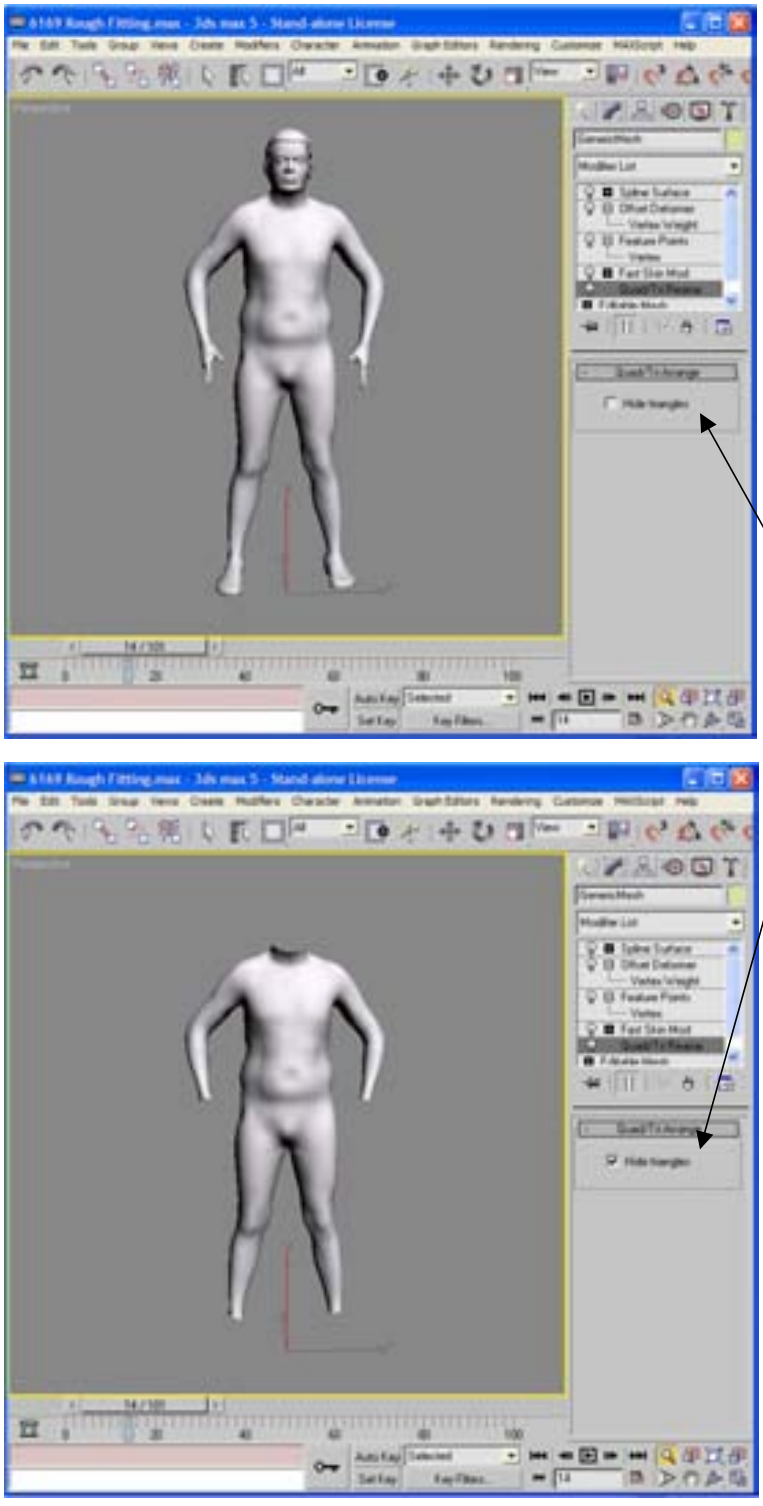


Figure 7-7. The QuadPatch modifier hides and/or unhides the head, hands and feet by rearranging the vertex coordinate of the mesh according to the element type and hiding/unhiding only the triangular meshes.

7.1.2.6 BodyManager utility

Developed mostly by Cordier (2004) at MIRALab, it contains various functions to support the compatibility between the standard/commercial plugins (standard: VRML 2.0, H-Anim, commercial: BonesPro, Character Studio Biped) and MIRALab softwares. Figure 7-8 and Figure 7-9 shows the interface panel of the BodyManager.

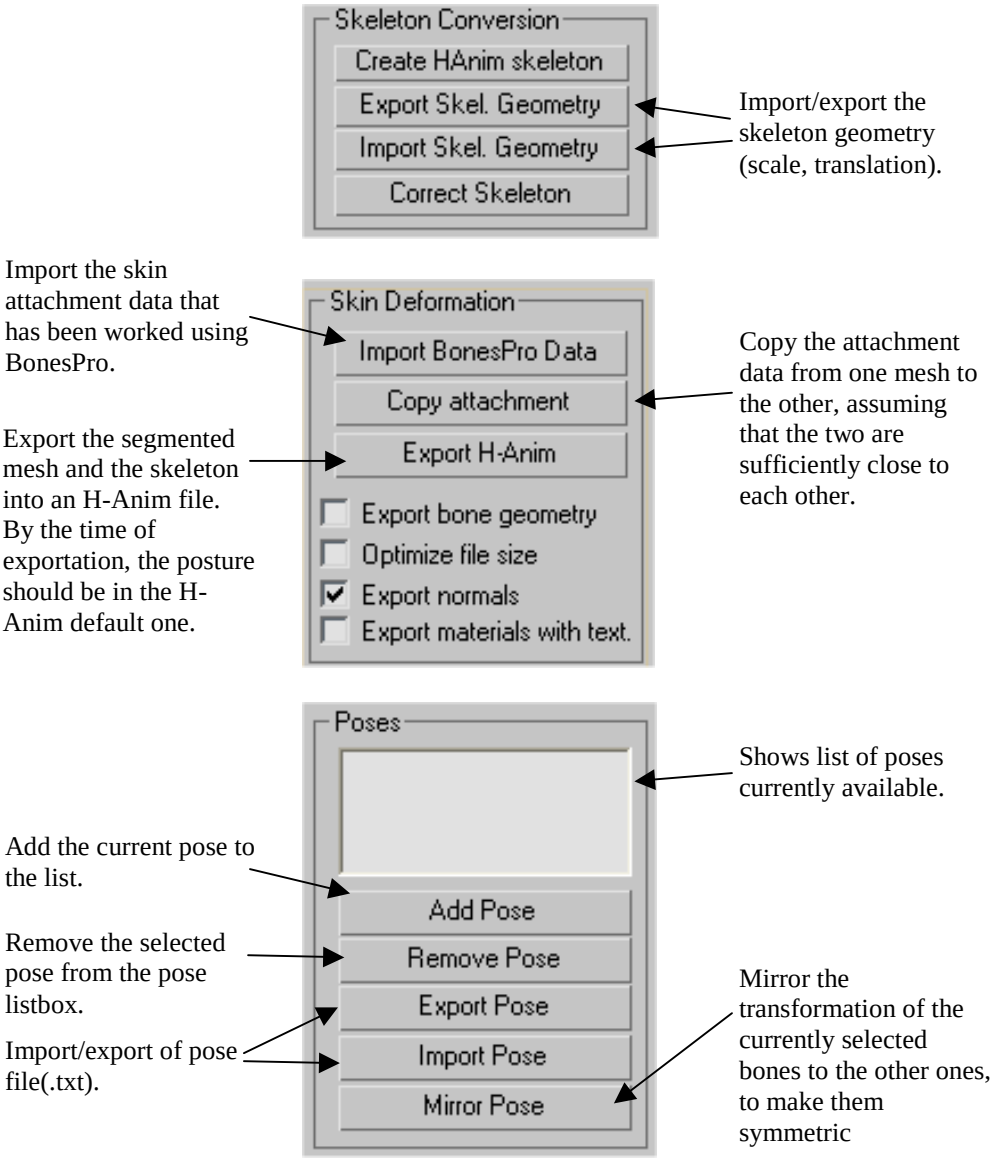


Figure 7-8. The BodyManager plugin is a set of utilities concerning the skin and animation control, conversion, and import/export.

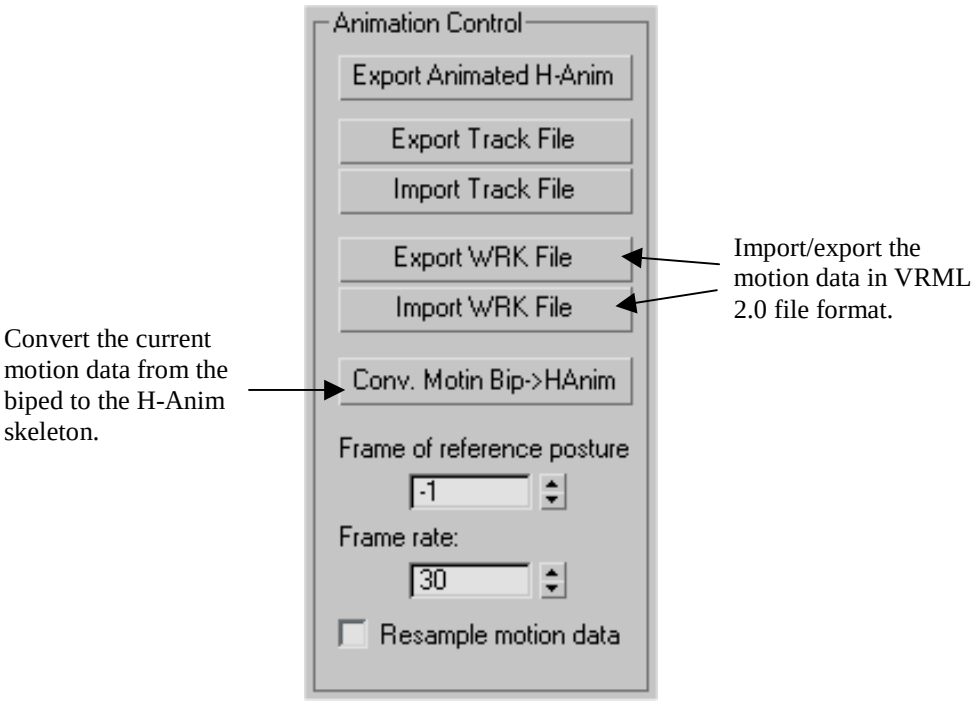


Figure 7-9. Animation control panel of the BodyManager plugin.

7.1.2.7 Spline Surface modifier

Performs subdivision of the template surface in order to elevate the density of the surface mesh, as described in Section 4.4.4. Figure 7-10 shows how the mesh increases its level of detail through the Spline Surface modifier.

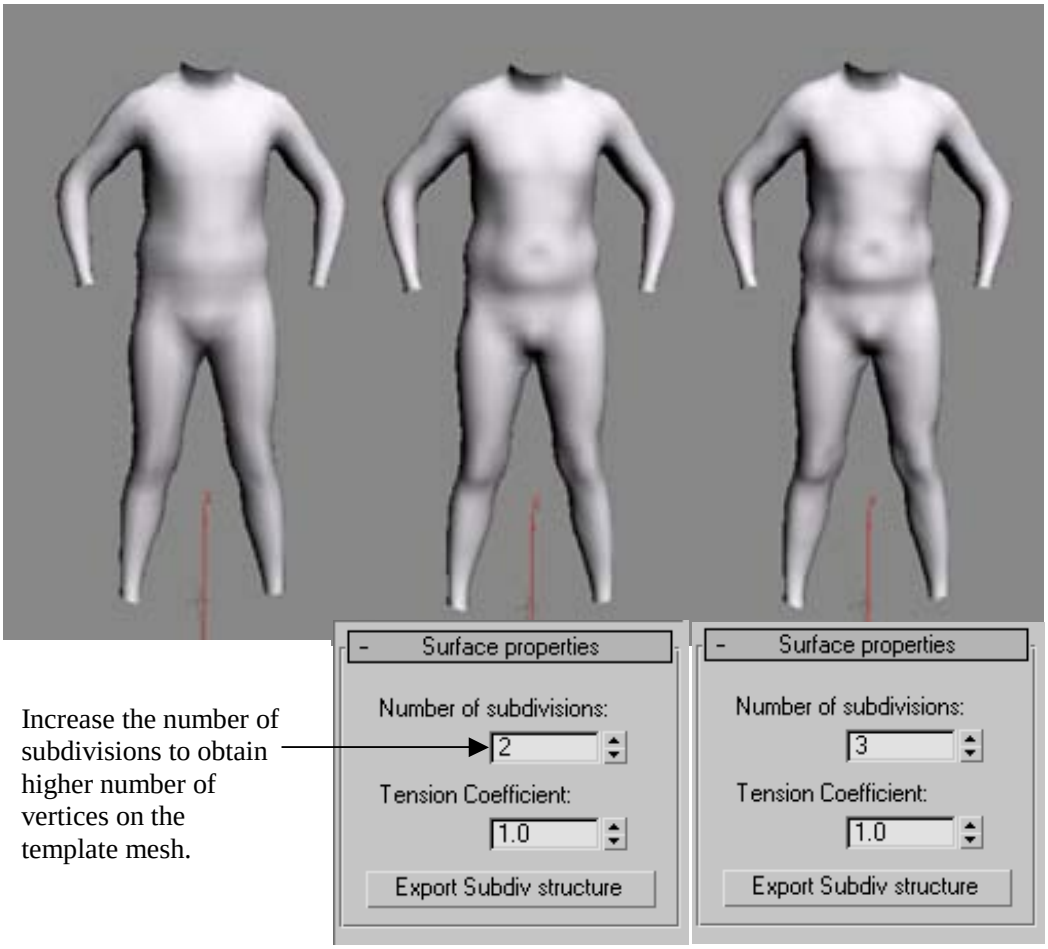


Figure 7-10. The spline surface modifier elevates the number of vertices and subsequently the levels of detail of the template mesh.

7.1.3 Workflow

Figure 7-11 illustrates the series of modification applied on the template mesh model while it is conformed onto a scanned data. Firstly the scan model and the target scan data are loaded and feature points are defined on both of them. Then the user loads the ScanFitter utility to automatically adjust the pose and the proportion of the template skeleton. What follows is the mesh relaxation by OffsetDeformer, which repositions vertices to capture the fine detail and to complete the accurate, smooth fitting. Finally, SplineSurface modifier is added to obtain the finer model. If necessary, "Mirror Pose" from the BodyManager is used to assure the symmetry of the resulting model. Prior to the exportation (skeleton, displacement and contour), the posture of the template model needs to be adjusted, in order to ensure that the displacements are stored at the same posture for every model.

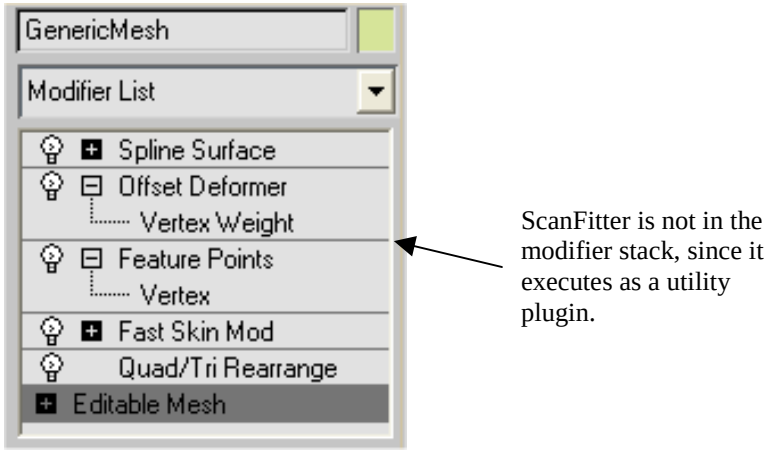


Figure 7-11. The modifier stack of the template mesh through the preprocessing phase. The plugin at a higher location is added later than the one at a lower location. Thus, the order of plugin applied through the preprocessing are: EditMesh → FastSkin → QuadPath → FeaturePoints (→ ScanFitter) → OffsetDeformer (→ BodyManager) → SplineSurface.

7.2 Runtime application

The synthesizers have been implemented as a C++ library ('BodySizer' hereafter) and integrated into MIRALab's VHD++ system. The overview of the system is shown in Figure 7-12. It makes frequent calls to the real-time skin deformation module developed by Cordier (2004) at MIRALab, for the real-time skeleton driven deformation. (Bournisien, 2003) has implemented the user interfaces shown in Figure 7-13 and Figure 7-14 during the pursuit of the master course at MIRALab.

As shown in Figure 7-12, BodySizer makes use of a number of submodules, which have been modularized either as classes or libraries.

- PCA class (PCALib): Performs the principal component analysis of the given data set. Returns the found eigen basis along with the projection of the input data onto the orthogonal basis.

- RBF class (RBFLib): Builds interpolation and/or extrapolation function of the given interpolant nodes. There exist two different implementations; one based on the Neural Network and the other on an analytic linear system solver.

- ShapeManager class: Reads in the contour file and offset file database and builds the displacement interpolator. Upon receiving new desired set of measurements, it generates the corresponding displacement map to be applied to the template model. It is also responsible for continuous measuring of the current template model before or after the resizing deformation.

- JointManager class: Reads in the skeleton file database and builds the joint interpolator/extrapolator. Upon receiving new desired set of measurements, it generates the corresponding skeleton transformation to be applied to the template model.

- Regression class: Builds the regression model to derive the modifier synthesizer, as described in Chapter 6.

- Subdivision class: Is responsible for subdividing the mesh surface to increase the detail.

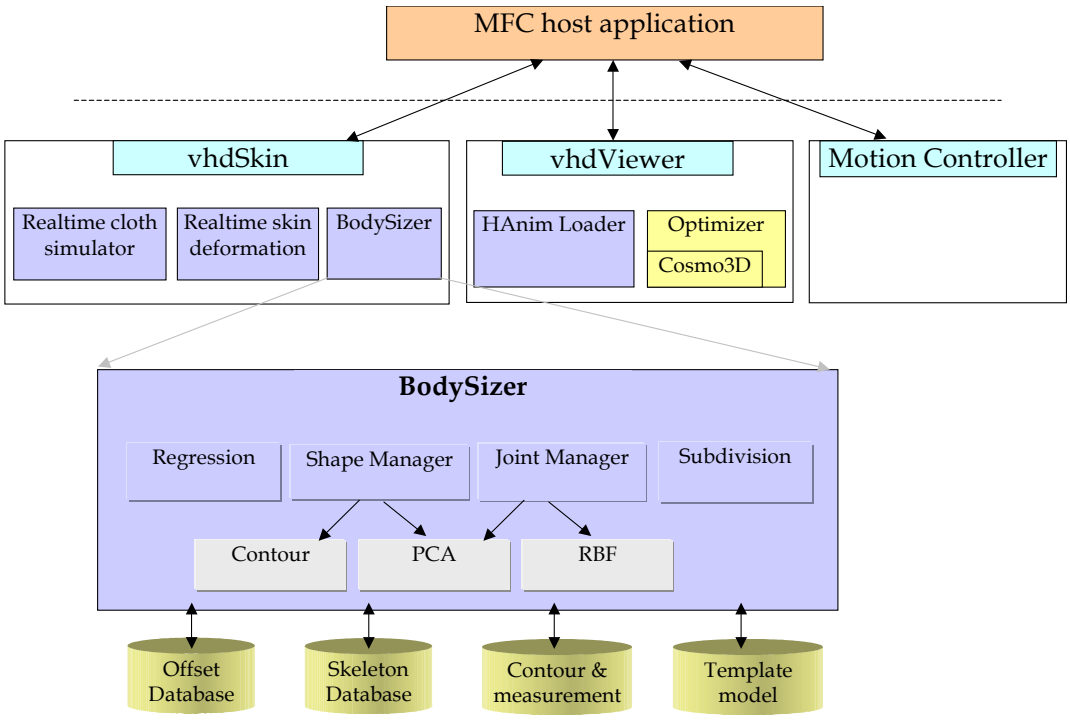


Figure 7-12. The system architecture of the BodySizer and the VirtualTryOn application.

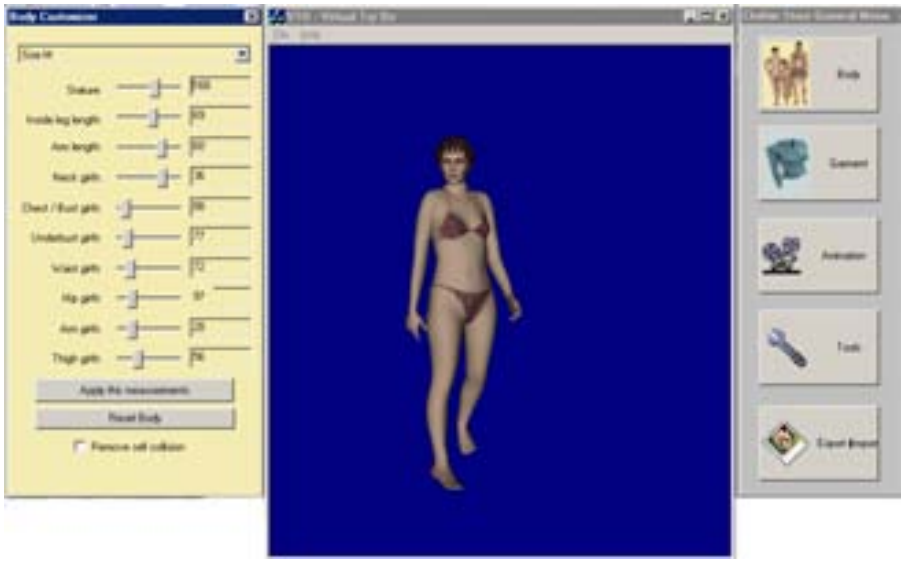


Figure 7-13. The VirtualTryOn software is for the real-time animation and visualization of resizable dressed body models.

7.3 Results

We have already given the research results of the proposed methodology in Chapters 4, 5, and 6. In this section we report end-user results, with an emphasis on the runtime performance.

The walk motion is motion captured using Vicon 8 optical motion tracking system (http::Vicon). The animators used 3ds max Character Studio for importing and refining the motion data. BodyManager plugin, MIRALab's in-house was then used for converting the motion data from the biped to the H-Anim skeleton structure and for exporting the H-Anim

file. The skin -to-bone attachment was set up by using the BonesPro plugin and has been reimported from the BodyManager, prior to the H-Anim exportation.

The exported template models along with the database files (contour, skeleton and offset files of each scan example) from the preprocessing phase are loaded inside the Virtual-Try On application, a real-time animation and visualization system that has been developed in the framework of the European IST project ETailor. The garments have been designed inside Fashionizer (Volino et al, 2003), MIRALab's brand-new virtual cloth design and simulation system. As the body models are resized and animated by the user interaction, the garments worn on the body is simulated and properly resized by using and recalculating the initial attachment information (Cordier et al, 2003). Resizing the body is fairly easy and almost instant. Figure 7-14 depicts some of the differently sized 3D mannequins that have been generated by BodySizer.

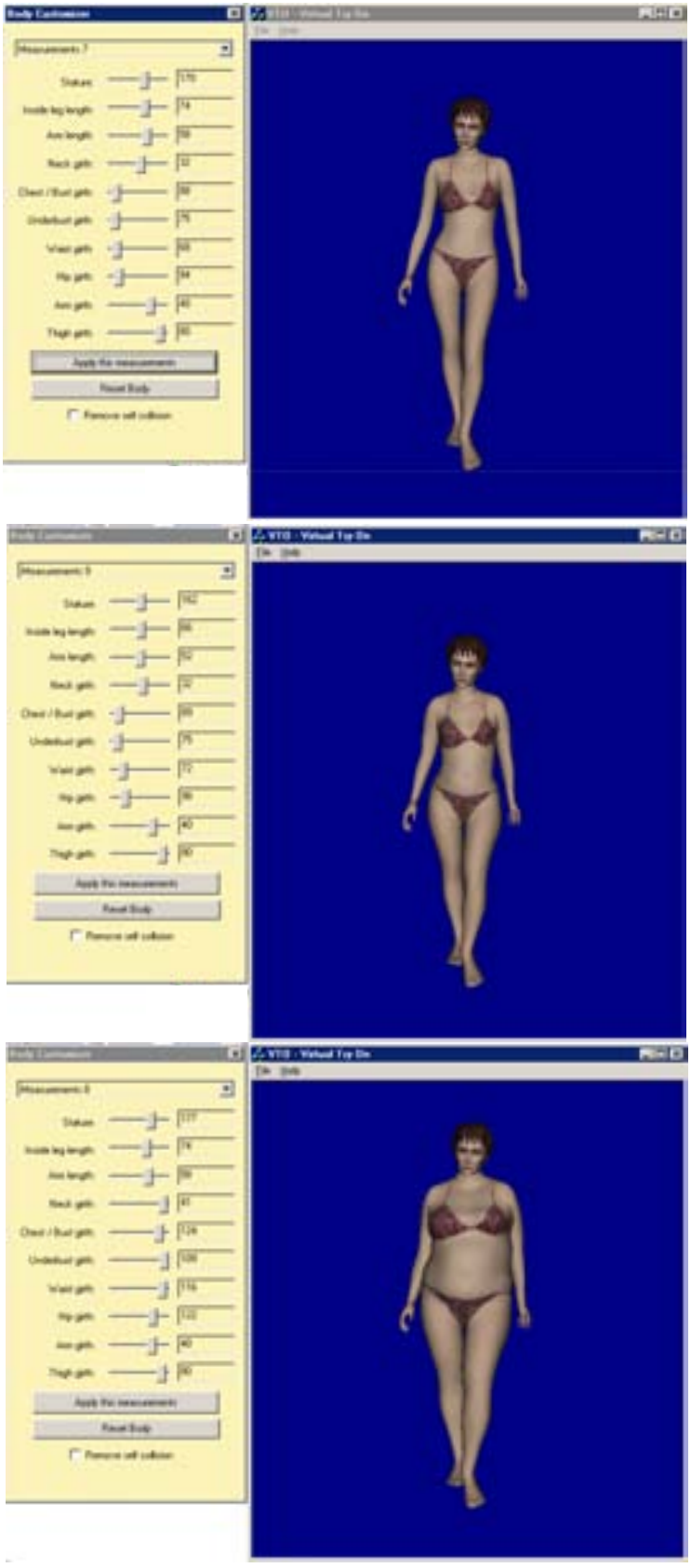


Figure 7-14. The modeling synthesizer in action in MIRALab's Virtual Try On application

Chapter 8

Conclusion

"...Never do anything that bores you. My experience in science is that someone is always telling you to do things that leave you flat. Bad idea. I'm not good enough to do well something I dislike. In fact, I find it hard enough to do well something that I like."

James Watson, "Succeeding in Science: Some Rules of Thumb", Science, 261, 24 (September 1993): 1812.

8.1 Contributions

This thesis has primarily been concerned with the synthesis of realistic 3D human body models from various control parameters. Arguably, captured 3D body geometry of real people provides the best available resource for modeling and statistical estimations. For this reason we have chosen to work with the range scanned data.

Our approach is inspired by the fact that the size and shape of human body are often correlated. Based on the example database obtained from range scanners, our synthesizers allow a reasonable estimation of the shape and size of the whole body even when only a little information is available. Consequently, the user can generate a new model or modify an existing one simply by inputting a few parameters to the system.

To summarize, this dissertation contains the following contributions:

- The preprocessing converts the scanned body models into animatable ones by estimating the semantic structure. A model-based approach is taken and two major steps are employed to conform a template model onto each subject data. A simple but efficient algorithm is proposed to automatically find the linear approximation of the scanned data through skeletal deformation of template model. Then, a combination of geometric and energy-based process is performed to calculate vertex displacements.

- Also presented is a new compact and efficient representation for the body geometry, which results in an effective, time saving manipulation of the model.

- A framework for automatically manipulating the full body geometry from sizing parameters is presented. The method takes as examples 3D scanned human body models in order to exploit the parameters-to-geometry correlations that are presented in the real bodies of individuals. The joint and shape interpolators are constructed by the scattered data interpolation based on the prepared examples.

- One natural manipulation of the body geometry is modification of existing models. Our paradigm for the parameterized shape modification is based on the statistical analysis of the examples. Consequently, our modifier synthesizer permits the user to estimate various changes on the current model that is indicated by the control parameters.

- Since we have chosen quantitative parameters that are also relevant to existing population studies, it is fairly easy to enhance our synthesizers to the generation of population models.

- Finally, we have integrated these contributions into a complete framework to generate and modify whole body models according to various needs. Our results indicate that a suitable combination of statistics and geometric methods is one true solution to an efficient, robust modeler that concerns the same class of objects. Combined together, our modeler produces realistic models that are readily animatable.

Although almost the same technique could be used for the different graphical objects, specifically motion data, dealing with other types of objects remains an uncovered work. For example, a more plausible or even stylish walk might be attainable if the motion data had also changed when the size of the limbs was changed or the model had undergone the fat percentage modification. Similarly, we could perform experiments to face models.

We believe our work can be of use in the study of biology, medicine, anthropology, and psychology. By using the example-based approach taken in this dissertation, it might be possible to estimate how the body shape evolves according to various factors. Apart from the clothing application that has been reported in this dissertation, other possible applications include growth studies, aging, and the effects of nutrition and sports on body shape.

8.2 Limitation and future work

Although the scanned database used here is sufficiently large to derive some statistical results, (Most authors recommend that one should have at least 10 to 20 times as many observations and one has variables in order to have stable estimates of the regression line.) it can become much larger as more scan data becomes available. Of course, it is clear that a larger database of our examples would result in more variety and flexibility in the appearance of body models. As the size of the database grows, the ability to rapidly construct the synthesizers will become more and more critical. Although here we have used the analytic solver to build the interpolators, other solutions like the growing RBF network (Esposito et al, 2000) could be considered. It is known to scale well to larger database, because it solves the weight and variance of its nodes in an iterative manner instead of solving the whole system with at once. Whenever there’s a new example scan to be added into the database,

The potentially large database also challenges the problem of automatic preprocessing. Though here the two-phase fitting has been made primarily by comparing the degree of feature-to-feature matching and by exploiting the topology of the template mesh, statistic analysis of the previously processed examples could also be used, when considering a large dataset. A recent example of such an approach is the markerless matching algorithm presented by Allen et al (2003), where the matching of the template model to a new scan data is found by formulating the energy term in the PCA space that has been constructed from the previously obtained body geometry.

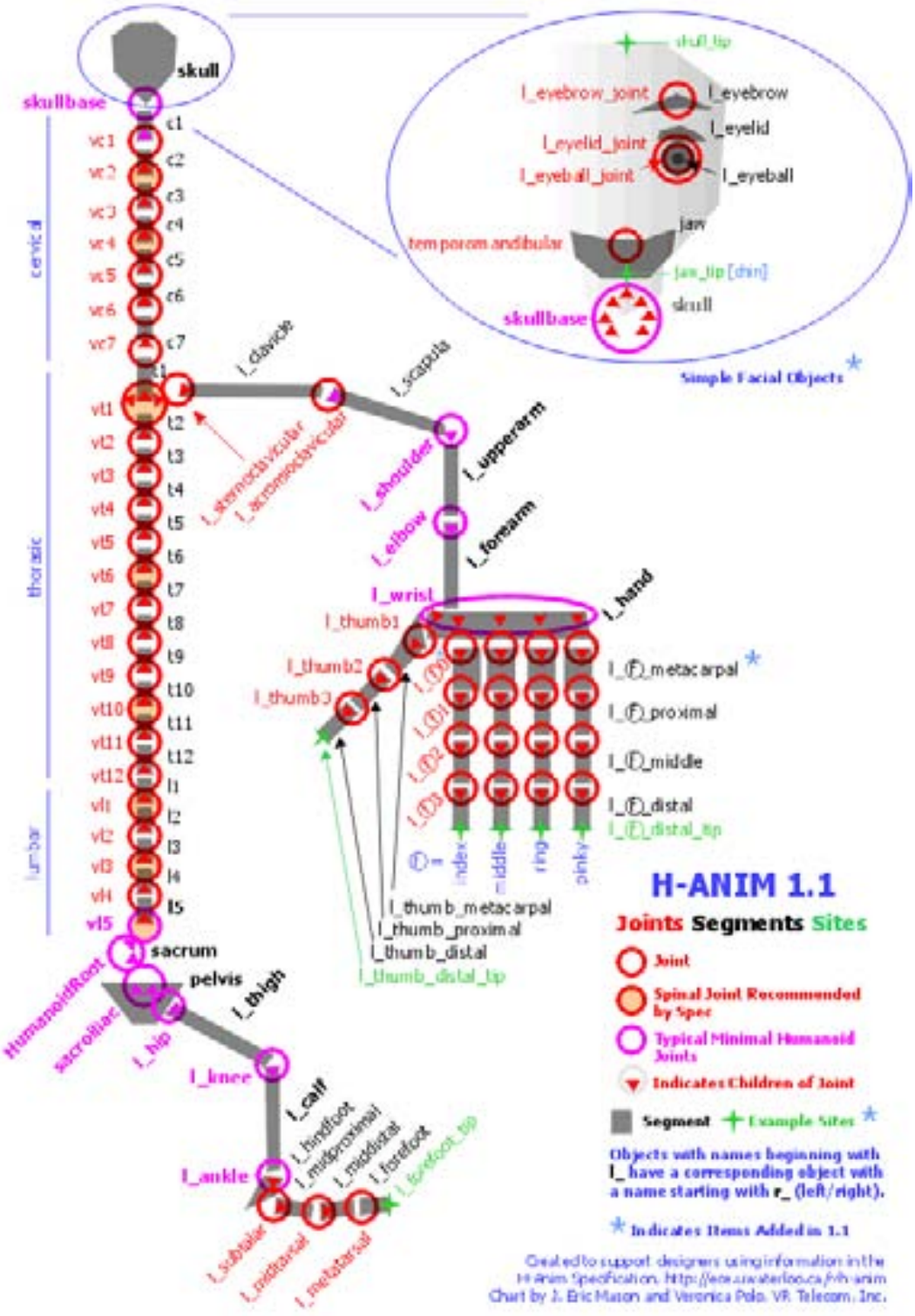
Currently, we rely on initial skinning and the attachment recalculation on non-zero displacements, for the skin deformation during joint-driven animation. As a result, pose-dependent shape changes are predicted only by the initial skinning of the template model, which in fact has required many changes in order to obtain acceptable skin deformation as the body moves. We believe however there should be a unifying framework for controlling both the dynamic (pose-dependent) and static (morphological) shape deformation. For instance, a multivariate regression model on the example data can be used to model the combinatory deformation. One important problem to address will be extraction of the shape transformation that is explained by the pose and by the morphological characteristics, respectively.

Body vectors and their distribution also depend on the initial skinning. When there is a strong weight on the upper arm, a small-scale value will be sufficient to result in a large arm while a large scale is required to have the same effect when the attachment is loose. An automatic refinement of the skin attachment to obtain an optimal skinning could be found, so that the variation of body vectors for all examples is minimized, for instance.

Although in this work we have experimented mainly with size and shape parameters, there are other types of criteria that we believe are worth exploring. Some examples are sports, aging, and ethnicity. Combining these parameters with the sizing parameters is certainly one possibility to extend our work. As well, when combined with physically-based, anatomical modelers, our modeler can be used not only in enriching the variety of anatomical models but also in fine-tuning the physical parameters.

Finally, we are also looking at investigating parameterized texture synthesis technique, in order to benefit from images in building realistic human models. In that case, it will be more desirable to work with the scan data that come along with textures.

Appendix A. H-Anim Skeleton hierarchy



Appendix B. H-Anim Levels of Articulation 2 (http::HAnim)

Joint	Center
HumanoidRoot	0.0000 0.8240 0.0277
sacroiliac	0.0000 0.9149 0.0016
l_hip	0.0961 0.9124 -0.0001
l_knee	0.1040 0.4867 0.0308
l_ankle	0.1101 0.0656 -0.0736
l_subtalar	0.1086 0.0001 -0.0368
l_midtarsal	0.1086 0.0001 0.0368
l_metatarsal	0.1086 0.0000 0.0762
r_hip	-0.0950 0.9171 0.0029
r_knee	-0.0867 0.4913 0.0318
r_ankle	-0.0801 0.0712 -0.0766
r_subtalar	-0.0801 0.0000 -0.0368
r_midtarsal	-0.0801 0.0000 0.0368
r_metatarsal	-0.0801 0.0039 0.0732
vl5	0.0028 1.0568 -0.0776
vl3	0.0041 1.1276 -0.0796
vl1	0.0048 1.1912 -0.0805
vt10	0.0056 1.2848 -0.0822
vt6	0.0059 1.3866 -0.0800
vt1	0.0065 1.4951 -0.0387
vc4	0.0066 1.5662 -0.0084
vc2	0.0066 1.5928 -0.0103
skullbase	0.0044 1.6209 0.0236
l_sternoclavicular	0.0820 1.4488 -0.0353
l_acromioclavicular	0.0962 1.4269 -0.0424
l_shoulder	0.2029 1.4376 -0.0387
l_elbow	0.2014 1.1357 -0.0682

l_wrist	0.1984 0.8663 -0.0583
l_thumb1	0.1924 0.8472 -0.0534
l_thumb2	0.1951 0.8226 0.0246
l_thumb3	0.1955 0.8159 0.0464
l_index0	0.1983 0.8024 -0.0280
l_index1	0.1983 0.7815 -0.0280
l_index2	0.2017 0.7363 -0.0248
l_index3	0.2028 0.7139 -0.0236
l_middle0	0.1987 0.8029 -0.0530
l_middle1	0.1987 0.7818 -0.0530
l_middle2	0.2013 0.7273 -0.0503
l_middle3	0.2026 0.7011 -0.0494
l_ring0	0.1956 0.8019 -0.0794
l_ring1	0.1956 0.7815 -0.0794
l_ring2	0.1973 0.7287 -0.0777
l_ring3	0.1983 0.7045 -0.0767
l_pinky0	0.1925 0.8066 -0.1036
l_pinky1	0.1925 0.7866 -0.1036
l_pinky2	0.1938 0.7452 -0.1024
l_pinky3	0.1948 0.7277 -0.1017
r_sternoclavicular	-0.0694 1.4600 -0.0330
r_acromioclavicular	-0.0836 1.4281 -0.0401
r_shoulder	-0.1907 1.4407 -0.0325
r_elbow	-0.1949 1.1388 -0.0620
r_wrist	-0.1959 0.8694 -0.0521
r_thumb1	-0.1899 0.8502 -0.0473
r_thumb2	-0.1874 0.8256 0.0306
r_thumb3	-0.1864 0.8190 0.0506
r_index0	-0.1961 0.8055 -0.0218
r_index1	-0.1961 0.7846 -0.0218
r_index2	-0.1954 0.7393 -0.0185
r_index3	-0.1945 0.7169 -0.0173

r_middle0	-0.1972 0.8060 -0.0468
r_middle1	-0.1972 0.7849 -0.0468
r_middle2	-0.1950 0.7304 -0.0441
r_middle3	-0.1939 0.7042 -0.0432
r_ring0	-0.1951 0.8049 -0.0732
r_ring1	-0.1951 0.7845 -0.0732
r_ring2	-0.1920 0.7318 -0.0716
r_ring3	-0.1908 0.7077 -0.0706
r_pinky0	-0.1926 0.8096 -0.0975
r_pinky1	-0.1926 0.7896 -0.0975
r_pinky2	-0.1902 0.7483 -0.0963
r_pinky3	-0.1908 0.7540 -0.0960

Appendix C. Software User Manual

Introduction

About BodyManager©

BodyManager is a full-featured human body model manipulator that has been continuously developed at MIRALab - University of Geneva, in the framework of EU projects SoNG, MELIES, LifePlus and E-Tailor. Implemented as a set of plug-ins to *3ds max*, it offers various functionalities to manipulate human body models, as listed in Table C-1.

Table C-1. List of features of *BodyManager*©.

Id	Features	Plug-ins
1	Compatibility with standards (*H-Anim, VRML*) and other commercial softwares including *BonesPro*© and *Character Studio*'s *Biped*	*BodyManager*
2	Preprocessing of dressed human model for the real-time cloth simulation	*ClothPreprocessing1, ClothPreprocessing2*
3	SDD(Skeleton driven deformation) related functions	*FastSkin, BodyManager*
4	Conformation of a template body model onto another body mesh model	*Feature Points, SkanFitter, OffsetDeformer, Quad/Tri rearrange, SplineSurface*
5	Body sizing, shape modification, population generation	*ShapeInterpolator*
6	Motion data conversion	*BodyManager*

Authors

BodyManager is primarily written by Frederic Cordier and Hyewon Seo at MIRALab - University of Geneva. Other contributors include Tom Molet for the support of *track* (.trk) file format (*BodyManager* utility) and MinTuan Bui for the automatic landmark detection (*ScanFitter* utility).

How to install it

Copy **BodyManager.dlm** in the *3ds max* plug-in directory, (e.g. C:/ProgramFiles/3dsmax5/plugins). In case you are using a licensed version of the plugin, you should copy the Flexlm dll file (e.g. lmgr327b.dll) to the directory where the 3ds max execution file is located (e.g. C:/ProgramFiles/3dsmax5).

About this document

This document provides a brief tutorial designed to help you master the usage of the *BodyManager* plug-ins. It primarily focuses on the features that are described in this dissertation (Feature item number 4 and number 5 in Table C-1) and description on other features of the plug-in can be found at (Cordier, 2003).

List of plugins

Table C-2 lists the plugins that are detailed in this document.

Table C-2. List of plug-ins described in this document.

Id	Plug-in name	Plug-in type	Description
1	*FastSkin*	modifier	Performs skeleton-driven deformation using the imported attachment data from *BonesPro*©.
2	*Feature Points*	modifier	Assists users to identify feature points and contours on a geometric mesh object of *EditMesh* type.
3	*ScanFitter*	utility	Based on the feature points defined by the *FeaturePoints* modifier, and on the skeleton-driven deformation by the *FastSkin* modifier, *ScanFitter* finds the global proportion and posture of the template model that best fits the target (scan) body model.
4	*OffsetDeformer*	modifier	When used after the *ScanFitter*, *OffsetDeformer* captures the fine detail of the target scan model and save it as a displacement map.
5	*Quad/Tri rearrange*	modifier	Reorganizes the vertex coordinate of geometric object of *EditMesh* type such that the quadratic elements precede triangular ones. It is primarily designed to separate and/or modify the head, hands and the feet parts from the body.
6	*BodyManager*	utility	Offers various functionalities to convert and export body models into *H-Anim* standards. Also offered are the motion data import, export and conversion tools.

7	*Spline Surface*	modifier	Performs Bezier subdivision on the target geometric object of *QuatPatch* type.
8	*Shape Interpolator*	utility	Implements the *Modeling Synthesizer* and *Modifier Synthesizer* described in this dissertation.

Preparing a template model

The template model is composed of a skeleton and a mesh, which is well aligned and attached to the skeleton. See Figure C-1 for the female template model used in this dissertation work.

1. **The skeleton hierarchy** should strictly follow the *H-Anim LoA2* standard, including the bone names.

   √ Note: Disable the scale inheritance of each bone. Select the bone, go to *Hierarchy* panel followed by the *Link Info* sub-panel, and uncheck all scale inheritances.

2. **The mesh** should be regular grid. This means the mesh is composed of a set of horizontal contours and vertical lines connecting these contours together. Wherever possible, **regularize the edge lengths of the quad-patches** over the mesh. Increased sampling around the joint may cause problems later, especially during the relaxation phase of the conformation process.

   √ Note: The mesh should not have any scale other than 100%.

3. The mesh should be properly attached to the skeleton hierarchy through the ***FastSkin*** modifier on the mesh. In case you generate the skinning information from the ***BonesPro***©, save the ***structure*** **(.txt) file** which is imported from the *FastSkin* modifier. It is also recommended to save the ***BonesPro***© **(.bpm) file** as well, for the potential modification of the attachment data.

   √ Note: The mesh should not be attached to the *HAnimRoot* joint.

   √ Note: It is recommended to maximize the locality in the bone-to-skin influence. For instance, with the *skullbase* bone which has influence on the neck will result in an undesirable deformation during the manipulation.

4. Add *Quad/Tri rearrange* modifier.

5. **Feature points and the contours** should be defined in the *FeaturePoints* modifier.

   a. To define a contour, select all vertices and press *Find next available* button followed by *Insert vertex* and *Reorder vertices* buttons.

   b. To export contours into a file, press *Export contour* button.

   c. To define a feature point, select one vertex and press the feature point number. The feature number appears in red near the defined vertex, as shown in .

   d. To export feature points into a file, press *Export Feature Points* button.

√ Note: Once you have completed the contour set and the feature points, save them into files.

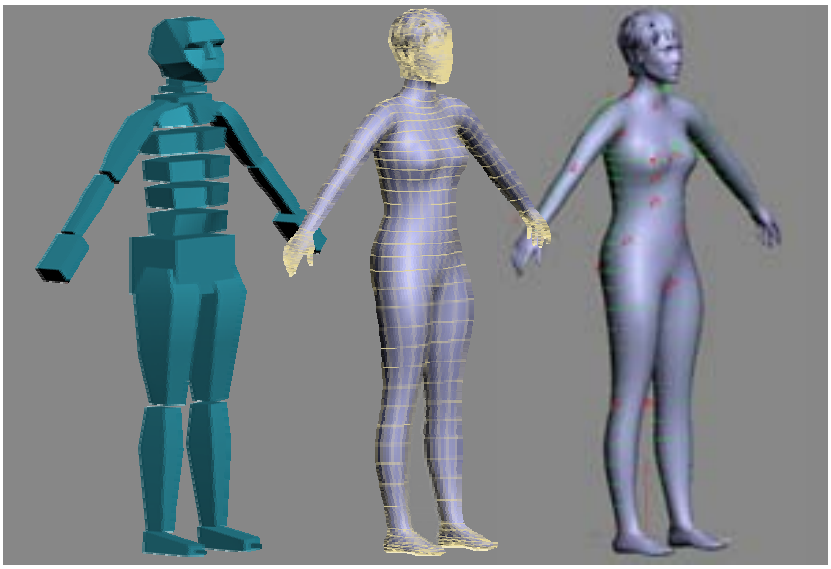


Figure C-1. The female template model.

6. To define a posture, move to the keyframe where you want to define posture and **press *Auto key* button** at the bottom right. Define the posture either by rotating selected bones or importing a posture from file through *Import Pose* function in the *BodyManager* utility. As an analogy, you can save the pose in the pose list or into a file through *Export Pose* function.

7. Save the file and name it as 'Generic.max'.

Exporting the template model to *VirtualTryOn*©

VirtualTryOn© requires the bodies and clothes in *H-Anim* and *VRML* format, respectively. Once ready, the template mesh is segmented and exported into standard files from *BodyManager* plug-in.

Usually the pre-processing is made with a lower resolution model, but it is strongly recommended to **export higher resolution model** as follows. This allows visualizing the higher quality model while maintaining the pre-processing data obtained at a lower resolution.

1. Make new attachment data for the high resolution mesh as follows:

   a. Open the max file containing the template model.

   b. Apply the '**open legs and arms**' posture on the template model, either by moving to the appropriate keyframe or by importing the posture from file. You can also interactively define the posture by applying rotation transformation on selected bones.

   c. **Clone the template mesh**. We will name it as 'GenericHigh'.

   d. *Collapse all* modifiers on the 'GenericHigh' mesh and apply *Spline Surface* modifier. Adjust the subdivision level properly. **Subdivision by 2** is acceptable in

most cases.

e. Press *Export Subdiv structure* in the *Spline Surface* modifier and **accept the default** file name '*QuadPatchStruct.txt*'. **Copy this file later into the *H-Anim* subdirectory** ('Female_parts' or 'Male_parts' directory).

f. Select the two meshes (the original mesh 'Generic' and the new mesh 'GenericHigh') and press ***Copy attachment*** in the *BodyManager* utility. Depending on the complexity of the meshes, it takes 5~8 seconds.

g. Check if the 'GenericHigh' mesh has been successfully attached to the skeleton and remove the original mesh.

2. Export to *H-Aanim* file with reserved names 'female.wrl' and 'male.wrl'.

a. Go to the *FastSkin* modifier of the mesh and *Segment* it.

b. Delete (press *delete* key) the selected mesh to keep only segmented patches and the skeleton in the scene.

c. Move to the *H-Anim default* posture by selecting the *HumanoidRoot* joint and double clicking the first pose from the pose list in the *BodyManager* utility.

d. *Export HAnim* from the BodyManager.

Changing the head/hands/feet of the template model

The *BodyManager* allows the user to easily change heads, hands and feet.

1. Open the template model you want to change the head from.
2. You can remove the current head, hands and feet from the current template model in several different ways. One recommendable way is to use the *Quad/Tri rearrange* modifiers as follows:

a. Remove all modifiers except the *Quad/Tri rearrange*.

b. Check *Hide triangles* as shown in Figure C-2.

c. *Collapse all* modifier on the template mesh.

d. Load the *BodyManager* utility and "*Import BonePro data*". This will add a *FastSkin* modifier automatically to the template model.

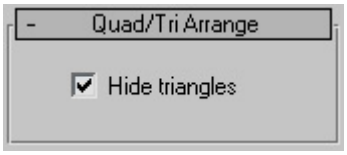


Figure C-2. Use *Quad/Tri rearrange* modifier to hide triangles from the currently selected object, so that only quadpatches are visible.

3. Transform the new head model so that it is well aligned to the neck.
4. Clear up the modifier stack so that you work at the *EditMesh* modifier level.
5. At the *EditMesh* modifier level, seamlessly connect the head to the body by welding vertices in the *EditMesh* modifier. The topology of the body part should not be modified.

6. Generate new skin attachment data, using *BonesPro* modifier.

   a. Import the *BonesPro* file (.bpm) of the current template model, if applicable.

   b. Work on the skull and neck bones to attach the new head properly. Work similarly on the hands and feet.

   c. Save the new *BonesPro* data and export the *structure* file (.txt).

   d. Load the *BodyManager* utility and *Import BonePro data*. This will add a *FastSkin* modifier automatically to the modified template model.

7. Every modifier that has been added after the *Quad/Tri rearrange* modifier on the current template model should now be reusable.

   a. Import the exported posture from the *BodyManager* utility.

      √ Note: Do not forget to set *AutoKey* before importing the posture so that other frames are not affected.

   b. To reuse/restore the scale and translation transformation, import the SkelGeom file from the *BodyManager* utility.

   c. To reuse/restore the offset, import the Offset file from the *OffsetDeformer* modifier.

   d. To redefine the feature points, you can copy feature points from the previous model using the *FeaturePoints* modifier.

      See also: Reusing feature points

Preparing a scanned model

1. Open the max file containing the scan mesh. Apply global transformation (scale, translate and rotate) at the *EditMesh* level so that the body is straight and the feet are aligned with the center.

   √ Note: The unit of the 3ds max should be in **meter**. You can check and set up the current unit from the *Customize* Menu followed by the *Units Setup*.

   √ Note: The scan mesh is assumed to have no holes and no open edges. If applicable, apply *Cap Holes* modifier.

   √ Note: The scan mesh is assumed to be at a moderate complexity (less than 70,000 triangles). If applicable, apply *PolygonCruncher*© modifier to simplify the mesh.

2. Add *Feature points* modifier and define the feature points that are consistent with the template model. The use of the *Feature points* modifier is illustrated in Figure C-3.

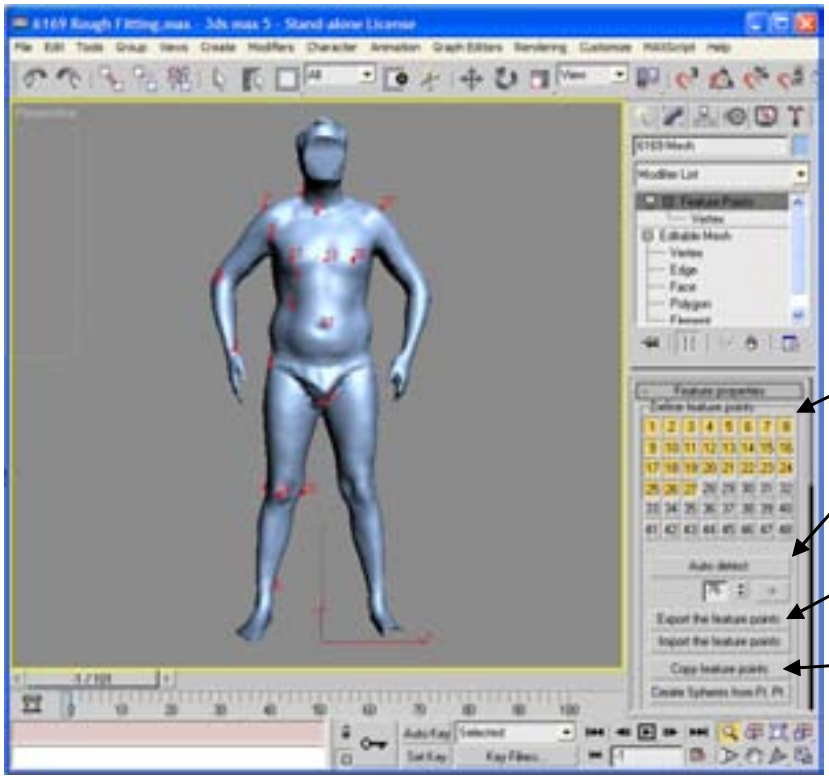


Figure C-3. The *Feature points* modifier.

Conforming the template model to a scan data

1. Load the scan model.
2. Merge the template model to conform to the scan model.
3. Load *ScanFitter* utility.
   a. *Pick the generic mesh* and *the scan mesh* from the plugin panel.
   b. Press *Fit Skeleton*. Depending on the number of joints, it will take 5~10 seconds.
4. Check the ending parts of the body, like wrist, ankle and neck. Check also the "vl3" bone. It is recommended to **use local coordinate system** when applying transformation.
5. It is often useful to *Mirror Pose* when feature points are defined only on the half of the body, and if the target scan model is symmetric from left to right.
   a. Select the bone or sets of bones which you want to **copy the transformation from**.
      √ Note: Some bones ('acromioclavicular', for instance) may not have volumes and thus are not possible to be selected from the viewpoint. Such dummy bones can be better seen using the *Select by Name*.
   b. Load the *BodyManager* utility and press *Mirror Pose*. (Do it at least twice)
6. Hide head/hands/feet using the *Hide triangles* from the *Quad/Tri rearrange* modifier.

7. Add the *OffsetDeformer* modifier on the template mesh.

   a. Press *Define Target Node* and select the scan mesh.

   b. Press the *Map Geom To Target* (collision distance: 10.0 by default) and *Update*.

      √ Note: The collision distance indicates the maximum surface distance (in centimeters) between the source mesh and the target. When the two meshes are close enough, you can save several seconds by reducing the distance. As an analogy, increase the distance when the maximum distance between the two meshes seems to be larger. In most cases, however, it is recommended to accept the default value

   c. Set the *smoothing parameters* with 100% *M1* and press *Apply Relaxation* for 2~3 times.

8. Add a new *Feature Points* modifier.

   a. *Import Feature Points* (e.g. Feature points BreastNavel.txt).

   b. *Import Contours* (e.g. Contour LowRes Torso.txt).

9. Go to *Feature Points* modifier of the scan model and define two additional feature points, say 26 (left breast) and 27 (left shoulder).

10. (Hide all except the generic mesh.) Move the newly added *Feature Points* below the *OffsetDeformer* and disable the previous one.

11. Go to *Vertex Weight* sub-selection level of the *OffsetDeformer* and adjust the vertex weights (from 0% to 100%) by **increasing or decreasing** the actual displacement relatively to the computed one. Press *Increase the looseness* button to increase the vertex weight (closer to the calculated displacement) and *Reduce the looseness* to decrease it (closer to 0). *Clear looseness* sets the vertex weight to 0%, leaving the displacement map to virtually null. Figure C-4 illustrates the use of the *OffsetDeformer* at the *Vertex Weight* subselection level.

    √ Note: **Check "Use Vertex weights" from the *OffsetDeformer* to actually activate the vertex weighting function.**

    √ Note: You can also import and export the vertex weight in order to avoid repetitive process. In this dissertation work, most of the weight adjustment had been on the armpit and crotch, where the scan data tends to be the most noisy due to the self-occlusion. Depending on the level of noise, different weight set ("Offset Weight LoRes Strong.txt" or "Offset Weight LoRes Weak.txt") have been loaded and re-adjusted.

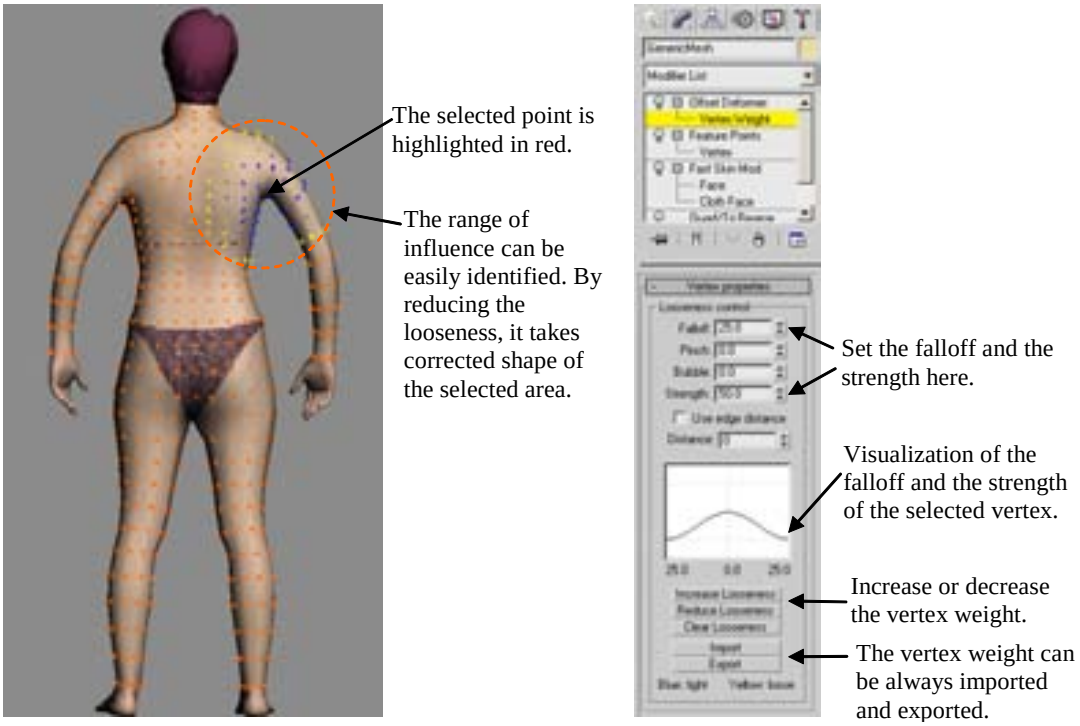


Figure C-4. The OffsetDeformer allows the offset adjustment through the vertex weight.

12. Move back to the *OffsetDeformer* modifier. Set the *smoothing parameters* with 20% *M2* and ***Apply Relaxation*** 3~5 times.
13. You can manually displace certain points using *EditMesh* and transfer the displacement to the *OffsetDeformer*.

    a. Add *EditMesh* modifier on top of the modifier stack and make necessary modification.

    b. Come back to the *OffsetDeformer* and *Set offset from stack result*. This accumulates the vertex displacement in the modifier stack inside the displacement map of the *OffsetDeformer*.

    c. Remove the *EditMesh* modifier.

See Figure C-5 for the visual illustration of the above procedures.

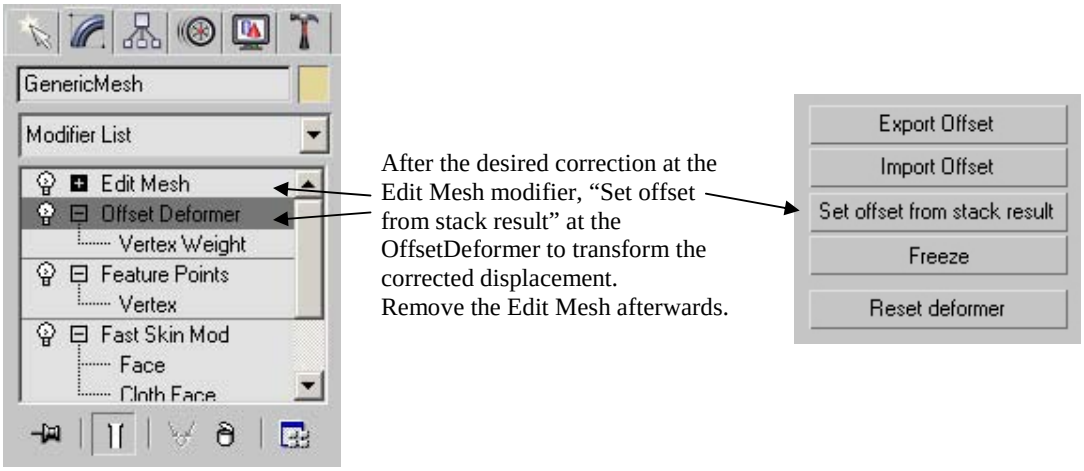


Figure C-5. The modification at the *EditMesh* level can be transferred to the *OffsetDeformer* by using *Set offset from stack result*.

14. Make sure to display all bones, i.e. unhide all bones.
15. Go to *FastSkin* modifier of the template skin. After checking that "Show end result" is on, press *Recompute Joint*.

    √ Note: In order to avoid modification of the head, hands and feet, hide these bones before the *Recompute Joint.*
16. Add *SplineSurface* modifier and adjust the number of subdivision to obtain the desired resolution. However, accepting the default subdivision number (2) is recommended.
17. In case a finer detail is desired, you can repeat the step 7 at a higher resolution model, as follows. However, our experience shows that adding the *SplineSurface* modifier (step (a) below) already gives satisfactory results for real-time applications.

    a. Add the second *OffsetDeformer* on top of the *SplineSurface* modifier.

    b. Make exact fitting through applying the *Map Geom To Target* with a lower collision distance, say 2~3.

    c. Apply *M1* relaxation as many times as necessary, although 2~3 times are sufficient in most cases.
18. Unhide the head, hands and feet by unchecking the *Hide triangles*.

Reusing feature points

You can copy and paste the *FeaturePoints* modifier as any other ones in *3ds max*, but it has restrictive sense. For instance, the user has to correct the copied feature points if the mesh object has different topology from the other one. If these two mesh objects are in similar shape, however, the *Copy feature points*" function is useful. One such example is shown in Figure C-6.

1. Select the mesh to define the feature points on and add a *FeaturePoints* modifier.
2. Load the mesh to copy feature points from and make sure the two meshes are well aligned.

3. Press *Copy feature points* button and select the source mesh.

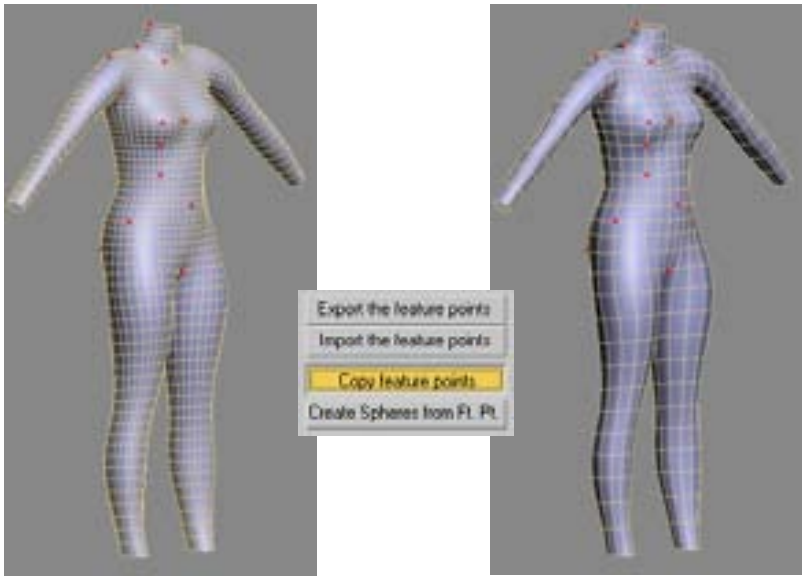


Figure C-6. *Copy feature points* in the *Feature Points* modifier uses the feature points defined on one model and calculate them for another, on geometric distance basis.

Exporting the preprocessed model into files

1. Define a good posture prior to file exportations.

   a. Frame 0 is for the captured posture of the scanned subject.

   b. Frame -1 is reserved for the *H-Anim* posture. Go to frame -1, click *Auto key* and apply *H-Anim default* posture from the *BodyManager*'s pose listbox.

   c. Frame 1 is for the displacement exportation from the *OffsetDeformer*.

      √ Note: Since every model should be exported at the same posture, using predefined posture through files is useful. In case you are importing the posture from file, go to frame 1, click *Auto key* and import the file containing the posture (e.g. "Posture for Offset.txt").

2. Go to frame 1.

3. ***Export offset*** from the *OffsetDeformer* modifier. Keep constant name convention for every offset files, such as 'Offset.txt'.

4. Make sure that all skeleton are displayed from the viewpoint (*Unhide All* if necessary.) and with the *HumanoidRoot* node selected, press ***Export skel. geometry*** in the *BodyManager* utility.

5. Export the contour file at frame 1.

   a. **Add a new *FeaturePoint*** modifier on the template mesh.

   b. **Import the contour file** that contains the complete contour set defined on the template mesh. (In this dissertation work, "Contour LoRes.txt" contains the full

contour set.)

c. Make sure the head of the template model is shown. **Uncheck *Hide triangles*** from the *Quad/Tri rearrange* modifier.

d. Press *Export Contour* in the *FeaturePoint* modifier.

Integrating a new data in the existing database

1. **Copy the three exported files** (Contour.txt, Offset.txt and SkelConf.txt) in the database directory.
2. Check the latest sample number and **rename the extension of the three files** above using the next sequential integer number. If the current largest sample number is 29, for instance, rename Contour**.txt** to Contour**.30**, Offset**.txt** to Offset**.30**, and SkelConf**.txt** to SkelConf**.30**.

   √ Note: The sample number **starts from 0**.
3. Remove all intermediate files.

   See also: Organizing the database directory

Organizing the database directory

1. The name convention used in this thesis work for the database directories is shown in Table C-3.

Table C-3. Name convention for the database directory.

	ShapeInterpolator utility	*VirtualTryOn*©
female	Database_F	statistics_f
male	Database_M	statistics_m

√ Note: The directory structure for the VirtualTryOn application is shown in Figure C-7.

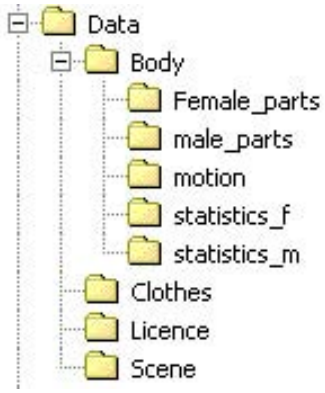


Figure C-7. The data directory organization of the *VirtualTryOn*©

2. The database directory contains the contour, offset and skeleton files of all examples, sequentially numbered from 0,1,... For instance, the data files from the example 1 are ‘Contours.1’, ‘Offset.1’, and ‘SkelConf.1’.
3. The name convention used in this thesis work for the template model is shown in Table C-4.

Table C-4. Name convention for the template model.

	ShapeInterpolator utility	*VirtualTryOn*©
female	$(DATABASE_PATH)\Generic.max	$(DATABASE_PATH)\..\female.wrl
male	$(DATABASE_PATH)\Generic.max	$(DATABASE_PATH)\..\male.wrl

4. ‘projection.*’, ‘pca.result.*’, ‘jprojection.*’, ‘jpca.result.*’ files in the database directory are intermediate files produced by the application. If you start the application without these intermediate files, the initialization of the application will newly generate these files, thus taking longer time. Once generated, however, the application skips the intermediate file generation and simply reads those files already existing before starting.

   √ Note: **Whenever there’s a new example files** that has been newly inserted into the database, **remove all intermediate** files before launching the application.

Using scripts to construct the database with many new data

By using scripted functions, the user can avoid repetitive process of inserting one data after another.

1. Organize the model directory using the following name conventions described as follows:

   a. Locate each model in different directories and name the completed conformation model into “$(MODEL_NAME) **Ready4DB LowRes.max**”.

   b. **Create “Database LowRes” subdirectory** for each model directory.

2. From the *MAXScript* utility, load the max script file “Export Database LowRes.ms”.

   a. Press *Locate root dir* and choose the directory which contains all example model directories.

   b. ***Export Contours*** from the *Export Body DB* script. Upon request, select the full contour file that has been defined on the template model. ***Export Offset*** and ***Export SkelConf***. This will save Contours.txt, Offset.txt and SkelConf.txt in the “Database LowRes” subdirectory of each model.

3. Launch the *MakeDatabase* script by double-clicking MakeDatabase.html.

   a. Browse the *Source* directory where the model files are located.

   b. Browse the *Target* directory where you want to compile the database.

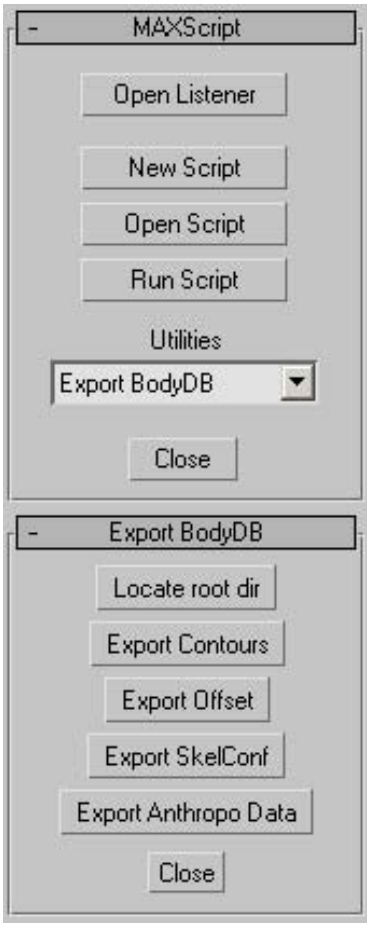


Figure C-8. The *Export Database* script is used to insert many new example data automatically.

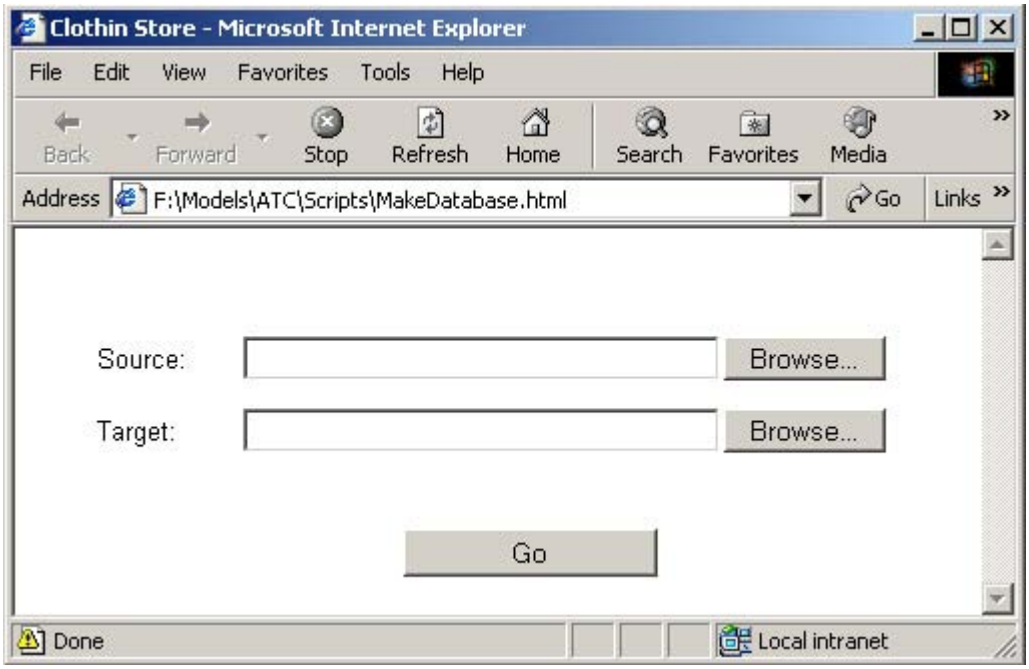


Figure C-9. The *MakeDatabase* script automatically copies each example files in the database directory and renames them with sequential numbers.

The Synthesizers

The *Bodysizer* is available in two platforms, one as a *3ds max* plug-in (***ShapeInterpolator*** utility) and the other in the *VirtualTryOn/Fashionizer*©. In any cases, the resizing of the model is fairly simple – set the measurements from the user interface, either by typing numbers or moving slider bars.

BodySizing in the *ShapeInterpolator* utility

1. Make sure that the female database (e.g. 'Database_F') and male database ('Database_M') directories are properly located.
2. Load the *Shape Interpolator* utility.

   a. Set the gender selection ('F' or 'M') and locate the database files by pressing ***Locate Database***.

   b. Check the template model that has been automatically loaded. If the loaded template model appears abnormal, try to reload the generic model through the *3ds max File* menu.

   c. You can visualize each example model (model ID: 1,2,…) by using *Show Example*.

   √ Note: You can make exaggerating models by setting the exaggeration level in the slider bar. Positive integers indicate the **exaggeration** whereas negative ones are used for the **anti-exaggeration**. Setting it with '1' will deactivate the exaggeration.

   d. Set the desired measurements using the slider bars.

   √ Note: It is strongly recommended to **keep the measurements consistent** at any

time during the manipulation. For instance, it does not make sense to have the underbust girth larger than the bust girth.

3. If applicable, the **model can be measured at various parts, using the contour structure** defined on the template model. Set the contour number of interest and press '→' to obtain the contour length.
4. To generate **population models**, set the desired number of population and press ***Random Gen***.

Shape modification in the *ShapeInterpolator* utility

1. Make a model whose modification is of interest, using one of the *ShapeInterpolator* functionalities described above.
2. Set the desired attribute in number using the slider bars in '*Shape variation with ..*' group.

BodySizing in the VirtualTryOn©

1. A number of predefined measurement sets are made available for a quick visualization of different individuals. Simply choose the name of the measurement set from the listbox and the corresponding modification on the template model will occur shortly.
2. Set the desired measurements using the slider bars, as an analogy to the *ShapeInterpolator* utility.

Bibliography

3DS MaxTM, http://www.discreet.com/products/3dsmax/

Arad N., Dyn N., Reisfeld D., and Yeshurun Y., “Image Warping by Radial Basis Functions: Application to Facial Expressions”, CVGIP: Graphical Models and Image Processing, Vol. 56, No. 2, pp.161-172, 1994, Academic Press.

Allen B., Curless B., Popovic Z., “Articulated body deformation from range scan data”, Proc. SIGGRAPH 2002, pp.612-619, Addison-Wesley.

Azuola F., Badler N.I., Ho P., Kakadiaris I., Metaxas D. and Ting B., “Building anthropometry-based virtual human models”, Proc. IMAGE VII Conf 1994 (June), Tuscon, AZ.

Besl P., “Chapter1: Active optical range imaging sensors”, Advances in Machine Vision, pp.1-63, Springer-Verlag, 1989.

Besl P.J., McKay N.D., “A Method for Registration of 3-D Shapes”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 14(2): pp.239-256, February 1992.

Bishop C. M., “Chapter5: Radial basis functions”, Neural Networks for Pattern Recognition, Oxford University Press, 1995.

Blanz B. and Vetter T., “A morphable model for the synthesis of 3D faces”, Computer Graphics Proc. SIGGRAPH ’99, Addison-Wesley, pp.187-194, 1999.

Bournisien F. “Développement d'une interface graphique OpenGL pour l’animation d'habits en temps-réel ”, Rapport de stage DESS, Université Claude Bernard Lyon I, 2003.

BonesPro, a plug-in to 3D Studio Max™, http://www.digimation.com/

Burnsides D., Boehmer M., Robinette K., “Landmark Detection and Identification in the CAESAR Project”, Proc. 3DIM (3rd International Conference on 3D Digital Image and Modeling), pp.393-398, 2001.

Carr J. C., Fright W. R. and Beatson K., “Surface interpolation with radial basis functions for medical imaging”, IEEE Transactions on Medical Imaging, , IEEE Computer Society, Vol. 16, No 1, pp.96-107, 1997.

Carr J. C., Beatson R. K., Cherrie J. B., Mitchell T. J., Fright W. R., McCallum B. C. and Evans T. R., “Reconstruction and representation of 3D objects with radial basis functions”, Computer Graphics Proc. SIGGRAPH ’01, Addison-Wesley, pp.67-76, 2001.

Carrere C., Istook C., Little T., Hong H. and Plumlee T., “Automated Garment Development from Body Scan Data”, Annual report, NTC Project I00-S15.

Carter J. E. I. and B. H. Heath, “Somatotyping – Development and applications”, Cambridge: Cambridge University Press, 1990.

Chadwick J. E., Haumann D. R. and Parent R. E., “Construction for Deformable Animated Characters”, Proc. SIGGRAPH ’89, Addison-Wesley, pp.243-252. 1989.

Cohen J., Varshney A., Manocha D., Turk G., Weber H., Agarwal P., Brooks F. and Wright W., “Simplification envelopes”, Proc. SIGGRAPH '96, Addison-Wesley, pp.119-128, 1996.

Cordier F., “Real-Time Animation of Dressed Virtual Humans”, PhD thesis, Départment d’informatique, University of Geneva, 2004.

Curless B. and Levoy M., “A volumetric method for building complex models from range images”, Proc. SIGGRAPH '96, Addison-Wesley, pp.303-312, 1996.

Daanen H. A. M. and van de Water G. J., “Whole Body Scanners”, Elsevier Displays 19, pp.111-120, 1998.

Da Vinci L., Paul Richter J. (Editor), “The Notebooks of Leonardo da Vinci, Vol. 1 (of a 2 vol. set in paperback) pp.182-183, Dover, 1972.

DeCarlo D., Metaxas D., and Stone M., “An anthropometric face model using variational techniques”, Proc. SIGGRAPH ’98, Addison-Wesley, pp.67-74, 1998.

Dekker L., Douros I., Buxton B. F. and Treleaven P., “Building Symbolic Information for 3D Human Body Modeling from Range Data”, Proc. of the Second International Conference on 3-D Digital Imaging and Modeling, IEEE Computer Society, pp.388-397, 1999.

Dekker L., “3D Human Body Modeling from Range Data”, PhD thesis, University College London, 2000.

Deville J. C., and Särndal C. E., “Calibration Estimators in Survey Sampling”, Journal of the American Statistical Association, 87: pp. 376-382, 1992.

Dooley M., “Anthropometric Modeling Programs – A Survey”, IEEE Computer Graphics and Applications, IEEE Computer Society, Vol.2, No.9, pp. 17-25, 1982.

Eck M., DeRose T., Duchamp T., Hoppe H., Lounsbery M.and Stuetzle W., “Multiresolution analysis of arbitrary meshes”, Proc. SIGGRAPH '95, Addison-Wesley, pp. 173-82, 1995.

Esposito A., Marinaro M., Oricchio D., Scarpetta S., “Approximation of continuous and discontinuous mappings by a growing neural RBF-based algorithm”, pp. 651-665, Neural Networks 13, Elsevier Science, 2000.

Farin G., “Curves and Surfaces for CAGD”, Morgan-Kaufmann Press, 2002, ISBN: 0-12-249054-1.

Forsey D. R., “A Surface Model for Skeleton-Based Character Animation”, Proc. of the Second Eurographics Workshop on Animation and Simulation, pp.55-73, 1991, Eurographics Association.

Gravesen J., “The length of Bezier Curves”, In Andrew Glassner, editor, Graphics Gems V, pp.199-205. Academic Press, 1995.

Grosso M., Quach R., Otani E., Zhao J., Wei S.. Ho P., Lu J. and Badler N. I., “Anthropometry for Computer Graphics Human Figures”, Technical Report MS-CIS-89-71, Department of Computer and Information Science, University of Pennsylvania, 1989.

Gu J., Chang T., Mak I., Gopalsamy S., Shen H. C. and Yuen M. M. F., “A 3D Reconstruction System for Human Body Modeling”, Proc. First International Workshop on Modelling and Motion Capture Techniques for Virtual Environments(CAPTECH ’98), pp. 229-241, 1998.

Heckbert P. S. and Garland M., “Multiresolution Modeling for Fast Rendering”, Proc. Graphics Interface '94, Canadian Inf. Proc. Soc., Banff, pp. 43-50, 1994.

Hilton A., Beresford D., Gentils T., Smith R. and Sun W., “Virtual People: Capturing human models to populate virtual worlds”, Proc. Computer Animation, IEEE Press, pp. 174-185, 1999.

Hodgdon J. A., Friedl K., “Development of the DoD Body Composition Estimation Equations”, Technical Document No. 99-2B, Naval Health Research Center, 1999.

Hohenstein, Uniform Body Representation, Workpackage Input, E-Tailor project, IST-1999-10549.

Hoppe H., DeRose T., Duchamp T., McDonald J. and Stuetzle W., “Surface reconstruction from unorganized points”, Proc. SIGGRAPH '92, Addison-Wesley, pp.71-78, 1992.

Hoppe H., DeRose T., Duchamp T., McDonald J. and Stuetzle W., “Mesh optimization”, Proc. SIGGRAPH '93, Addison-Wesley, pp.19-26, 1993.

Hoppe H., “Surface reconstruction from unorganized points”, PhD Thesis, Department of Computer Science and Engineering, University of Washington, June 1994.

Hoppe H., “Progressive meshes”, Proc. SIGGRAPH '96, Addison-Wesley, pp. 99-108, 1996.

(http::AdvancedClothing) http://www.advancedclothing.org/

(http::Brown) Brown P., Australian & Asian Palaeoanthropology, teaching resources, http://www-personal.une.edu.au/~pbrown3/oskel.html, 1992-2003.

(http::Browzwear) http://www.browzwear.com, Browzwear.

(http::Cardlab) http://www.hec.afrl.af.mil/cardlab/applications/multivariate.html, Computerized Anthropometric Research and Design Laboratory.

(http::Cyberware) http://www.cyberware.com/

(http::HAnimFeature) http://h-anim.org/Specifications/H-Anim1.1/appendices.html#appendixc.

(http::Vis) http://www.vis.ca/ VisImage System Inc.

(http::Vicon) Vicon Motion Systems, http://www.vicon.com.

Ju X., Werghi N. and Siebert J. P., “Automatic segmentation of 3D human body scans”, Proc. IASTED Int. Conf. on Computer Graphics and Imaging 2000 (CGIM 2000), pp.239-244, Las Vegas, USA, 2000, ACTA Press.

Ju X.and Siebert J. P., “Conforming generic animatable models to 3D scanned data”, Proc. 6th Numerisation 3D – Scanning 2001 Congress, Paris, pp.487-488, France, 2001.

K. Kähler, J. Haber, H. Yamauchi, H.-P. Seidel, “Head shop: Generating animated head models with anatomical structure”, pp.55-64, Proc. ACM SIGGRAPH Symposium on Computer Animation, 2002.

Kakadiaris I. A. and Metaxas D., “3D Human body model acquisition from multiple views”, In Proc. of the Fifth International Conference on Computer Vision, pp.618-623, Boston, MA, June 1995.

Kry P. G., James D. L., and Pai D. K., “EigenSkin: Real Time Large Deformation Character Skinning in Graphics Hardware”, Proc. ACM SIGGRAPH Symposium on Computer Animation, pp.153-159, 2002.

Lee W., Gu J. and Magnenat-Thalmann N., “Generating animatable 3D virtual humans from photographs”, Computer Graphics Forum, Vol. 19, No. 3, Proc. Eurographics '2000 Interlaken, Switzerland, August, pp.1-10, 2000.

Lee Y. and Lee S., “Geometric Snakes for Triangular Meshes”, Computer Graphics Forum (Eurographics 2002), Vol. 21, No. 3, 2002.

Lewis J.P., Cordner M., Fong N., “Pose Space Deformations: A Unified Approach to Shape Interpolation and Skeleton-Driven Deformation”, Proc. SIGGRAPH 2000, Addison-Wesley, pp.165-172, 2000.

Magnenat-Thalmann N., Papagiannakis G., Ponder M., Molet T., Kshirsagar S., Cordier F., Thalmann D., “LIFEPLUS: Revival of life in ancient Pompeii”, Proc. VSMM (Virtual Systems and Multimedia) 2002, October 2002.

Magnenat-Thalmann N., Thalmann D., “The Direction of Synthetic Actors in the film Rendez-vous à Montreal”, Computer Graphics and Applications, IEEE Computer Society Press, Vol. 7, No.12, pp.9-19, 1987.

Magnenat-Thalmann N., Thalmann D., “Human Body Deformations Using Joint-dependent Local Operators and Finite-Element Theory”, Making Them Move, N.Badler, B.A.Barsky, D.Zeltzer, eds, Morgan Kaufmann, San Mateo, California, pp.243-262, 1991.

Merriam M. L., “Experience with the cyberware 3D digitizer”, Proc. NCGA, pp.125-133, March 1992.

Meunier P.and Yin S., “Performance of a 2D image-based anthropometric measurement and clothing sizing system”, Applied Ergonomics, Vol. 31, No. 5, pp.445-451, 2000.

NASA Reference Publication 1024, “The Anthropometry Source Book”, Volumes I and II, NASA Scientific and Technical Information Office, 1978.

NASA Man-Systems Integration Manual (NASA-STD-3000), 1987.

Nedel L. and Thalmann D., “Modelling and Deformation of Human Body using an Anatomy-Based Approach”, Proc. Computer Animation ’98, Philadelphia, IEEE Computer Society Press, pp.34-40, 1998a.

Nedel L. and Thalmann D., “Real Time Muscle Deformation Using Deformations Using Mass-Spring System”, Proc. CGI ’98, IEEE Computer Society Press. 1998b.

Nurre J., “Locating landmarks on human body scan data”, International Conference Recent Advances in 3-D Digital Imaging and Modeling, pp.289-295, IEEE Computer Society Press, 1997.

Oliveira J., Buxton B., “Light-weight virtual humans”, Proc. of EUROGRAPHICS UK ‘2001.

PeopleSize Professional, Open Ergonomics Ltd., 1993-2001.

Plaenkers R., Fua P. and D’Apuzzo N., “Automated Body Modeling from Video Sequences”, Proc. mPeople Workshop, IEEE-ICCV99, pp.45-52, Corfu, Greece, 1999.

Powell M. J. D., “Radial basis functions for multivariate interpolation: A review”, In Algorithms for Approximation, eds. Mason, J.M. and Cox, M.G., pp.143-167, Oxford University Press, 1987.

Press W. H., Flannery B. P., Teukolsky S. A., and Vetterling W. T., Numerical Recipes in C, The art of scientific computing, Cambridge University Press, 1988.

Rose C., Cohen M., and Bodenheimer B., “Verbs and Adverbs : Multidimensional Motion Interpolation Using RBF”, IEEE Computer Graphics and Applications, IEEE Computer Society Press, Vol.18, No.5, pp.32-40, 1998.

Ruprecht D. and Muller H., “Image Warping with Scattered Data Interpolation”, IEEE Computer Graphics and Applications, IEEE Computer Society Press, Vol. 3, pp.37-43. 1995.

Schneider P.J., “An algorithm for automatically fitting digitized curves”, In Andrew Glassner, editor, Graphics Gems, pp.612-626, Academic Press, San Diego, CA, 1990.

Schroeder W. J., Zarge J. A., Lorensen W. E., “Decimation of triangle meshes”, Proc. SIGGRAPH '92, Chicago, IL, USA, pp.65-70, 1992.

Seo H., Cordier F., Philippon L., Magnenat-Thalmann N., "Interactive Modelling of MPEG-4 Deformable Human Body Models", pp.120-131, Deform'2000; Geneva, Switzerland, Nov. 29-20, 2000, Workshop on Virtual Humans by IFIP Working Group 5.10 (Computer Graphics and Virtual Worlds), Kluwer Academic Publishers.

Seo H. and Magnenat-Thalmann N., “LoD Management on Animating Face Models”, In Proc. IEEE Virtual Reality '2000, IEEE Computer Society Press, pp.161-168, 2000.

Shen J., Thalmann D., “Interactive Shape Design Using Metaballs and Splines”, In Proc. of Implicit Surfaces `95, Grenoble, France, pp.187-196.

Sloan P. P., Rose C. and Cohen M., “Shape and Animation by Example”, MSR-TR-2000-79, July, 2000.

Sloan P. P., Rose C. and Cohen M., “Shape by Example”, Symposium on Interactive 3D Graphics, March, 2001.

Tecmath AG, http://www.hs.tecmath.com/

Turk G. and Levoy M., “Zippered polygon meshes from range images”, Proc. SIGGRAPH '94, Orlando, FL, USA, Addison-Wesley, pp.311-318. 1994.

Turk G. and O'Brien J., “Shape Transformation Using Variational Implicit Functions”, Proc. SIGGRAPH '99, Addison-Wesley, pp.335-342, 1999.

Volino P. and Magnenat-Thalmann N., “Efficient Self-Collision Detection on Smoothly Discretized Surface Animations using Geometrical Shape Regularity”, Proc. Eurographics '94, pp.155-166, Computer Graphics Forum, Vol. 13, No. 3, Blackwell Publishers, 1994.

Volino P. and Magnenat-Thalmann N., “Accurate Collision response on polygonal Meshes”. In Computer Animation 2000, Annual Conference Series, IEEE Computer Society Press, pp.154-163, 2000.

Volino P., Cordier F. and Magnenat-Thalmann N., “From Early Virtual Garment Simulation to Interactive Fashion Design”, journal of CAD/CAM, 2003, to appear.

Weber J., “Run-Time Skin Deformation”, In Proc of Game Developers Conference 2000.

Werghi N., Fisher R. B., Ashbrook A. and Robertson C., “Faithful recovering of quadric surfaces from 3D range data”, Proc. of 3DIM’99, IEEE Computer Society Press, pp.289-289.

Wilhelm J., “Animals with Anatomy”, IEEE Computer Graphics and Applications, IEEE Computer Society Press, pp.22-30, 1997.

Publications arising from this thesis

Cordier F., Seo H. and Magnenat-Thalmann N., “Made-to-Measure Technologies for Online Clothing Store”, IEEE Computer Graphics and Applications, pp.38-48, January/February 2003.

Seo H. and Magnenat-Thalmann N., “An Automatic Modeling of Human Bodies from Sizing Parameters”, Proc. ACM SIGGRAPH Symposium on Interactive 3D Graphics 2003 (April), pp.19-26, pp.234 (color plate), Monterey, USA.

Seo H., Cordier F. and Magnenat-Thalmann N., “Synthesizing Animatable Body Models with Parameterized Shape Modifications”, Proc. ACM SIGGRAPH/Eurographics Symposium on Computer Animation 2003 (July), pp.120-125, San Diego, USA.

Magnenat-Thalmann N., Seo H., and Cordier F. “Automatic Modeling of Animatable Virtual Humans”, Proc. 3D Digital Imaging and Modeling 2003 (October), pp. 2-10, Alberta, Canada, IEEE Computer Society Press.

Seo H. and Magnenat-Thalmann N., “An Example-Based Approach to Human Body Manipulation”, Graphical Models, Vol. 66, No. 1, pp. 1~23, 2004, Academic Press.

Seo H, “Anthropometric Body Modeling”, Chapter 4, “Virtual Humans” Book, Magnenat-Thalmann N. and Thalmann D., Eds, in preparation, John Wiley & Sons.

Other publications

Joslin C., Seo H., Lefevre C., Lee W., Magnenat-Thalmann N., Jovovic M., Rougeot S., Esmerado J., Thalmann D., “Distance Communication using Networked Virtual Collaborative Environments”, Proc. of the Closing Conference of Swiss Priority Programme, pp.59-62, February 2000.

Pandzic I., Babski C., Capin T., Lee W., Magnenat-Thalmann N., Musse S. R., Moccozet L., Seo H. and Thalmann D., “Simulating Virtual Humans in Networked Virtual Environments”, Presence, Vol. 10, No. 6, pp.632-646, 2001, MIT Press.

Seo H., Magnenat-Thalmann N., Lee W., Kshirsagar S. and Goto T., “Facial Communication in Virtual Environments”, Tutorial, IEEE Virtual Reality 2000 (March), New Brunswick, USA.

Seo H. and Magnenat-Thalmann N., “LoD Management on Animating Face Models”, Proc. IEEE Virtual Reality 2000 (March), pp.161-168, New Brunswick, USA.

Seo H., Joslin C., Berner U., Magnenat-Thalmann N., Jovovic M., Esmerado J., Thalmann D. and Palmer I., “VPARK – A Windows NT Software platform for a Virtual Networked Amusement Park”, Proc. Computer Graphics International 2000 (June), pp.309-315, Geneva, Switzerland, IEEE Computer Society Press.

Seo H., Cordier F., Philippon L., Magnenat-Thalmann N., "Interactive Modelling of MPEG-4 Deformable Human Body Models", Proc. Deform 2000 (November), pp.120-131, Geneva, Switzerland, Workshop on Virtual Humans by IFIP Working Group 5.10 (Computer Graphics and Virtual Worlds), Kluwer Academic Publishers.

Seo H., Yahia-Cherif L., Goto T., and Magnenat-Thalmann N., “GENESIS : Generation of E-Population Based on Statistical Information”, Proc. Computer Animation 2002 (June), pp.81-88, Geneva, Switzerland, IEEE Computer Society Press.