Towards Automatic Character Skeletonization and Interactive Skin Deformation

T. Di Giacomo[1] & L. Moccozet[1] & N. Magnenat-Thalmann[1] & R. Boulic[2] & D. Thalmann[2]

[1]MIRALab, University of Geneva, Switzerland
[2]VRLab, EPFL, Switzerland

Abstract

Characters are of paramount importance in computer graphics since they enrich 3D worlds with immersion and life, and extend the range of possible applications. They are also very complex objects to manage due to their articulated structure, to their motion-dependent deformations, and to the familiarity we have with them.

With the wide variety of issues to be considered when modeling, rendering, and animating characters, lots of work has focused on character representation and animation. In this STAR we present core and recent techniques to automatically generate a skeleton from a character representation, in order to animate it with bone-based animation techniques. We then present core and recent methods for computing the deformation of the skin according to body movements.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computational Geometry and Object Modeling]: Hierarchy and geometric transformations I.3.7 [Three Dimensional Graphics and Realism]: Animation

1. Introduction

With respect to other objects encountered in computer graphics, characters are very specific deformable articulated objects. Their anatomical complexity is high and their animation has to match as closely as possible to the anatomical and physiological behavior of the bodies, especially because of the important ability for humans to perceive these kinds of motions.

For more than twenty years, different approaches and methods have been proposed to allow for the visual reproduction of anatomical behavior of character bodies in motion. The two main aspects that are taken into account are the simulation of the character body movements with an articulated skeleton and its control, as well as the skin deformation that should be coherent with respect to the skeleton motions and the functional anatomy of characters. From videogames, movies, entertainment and leisure applications to virtual tourist guides, presenters and training simulations for instance, recently more and more virtual environments require the simultaneous integration of a high number of characters. It therefore raises new interests and motivations to research and develop optimization methods for the generation, rendering and animation of characters.

This STAR studies the evolution and achievements of methods focusing on characters. It is worth noting that we do not cover here work on motion retargeting and motion synthesis to automate the generation of animation data. The research topics considered in this STAR are methods for the animation of characters, including skeleton skinning approaches, methods for the optimization of character generation with automatic extraction and generation of skeleton.

2. Automatic Generation of Skeleton

In this Section, we review existing methods to generate optimized representations of characters. A virtual character is composed of two main features which are its animation structure and its shape. Mainly, the animation structure of a character is based on the articulated model with skeletons. First the different approaches to automatically extract or generate skeletons are detailed. We then present optimization methods for the appearance of characters. Finally, the optimization issues of character animation are introduced, which should consider both animation structures and shapes.

2.1. Automatic Generation of Skeleton

Most of the time, the articulated character is composed of a geometric structure and a skeleton that is bound to the geometric model typically by manual interactions defining a correspondence between the primitives of each structure. Some additional bindings must be performed to couple skin surface motions to those of the skeleton; this can be done for example by generating spring networks or spatial deformation fields. These two processes are particularly tedious, especially when the model to be articulated is given only as a boundary representation. Some attempts have been proposed to automatically generate the articulated skeleton from a 3D surface or shape. The main expected advantage is to have an immediate mapping of the skin to the skeleton, in addition to the generation of the skeleton itself. The main difficulty of the automatic generation of skeleton is to map the articulated skeleton to the extracted geometric skeleton, particularly if the topology of the articulated skeleton is predefined such as for humans for instance. There is no guarantee that the topology of the extracted geometric skeleton will match the predefined topology of the corresponding articulated one. Moreover, the geometric skeleton only defines the location of the joints, but does not contain the rotation axis.

2.1.1. Medial Axis-Based Methods

Some of the existing approaches are based on the generation of the medial axis, which is further simplified in order to catch the appropriate articulated structure of the 3D shape. Bloomenthal [Blo99] proposes a method to derive a geometric skeleton from the medial axis of a static object. An articulated skeleton is further attached to the geometric and is used to control and alter the shape of the object for animation. The shape of the object is reconstructed from the updated geometric skeleton once it has been updated by the articulated one. Teichmann *et al.* [TT98] create an articulated control skeleton and bind it to the surface by first computing an approximate medial axis of the 3D mesh using 3D Voronoi diagram. The medial axis is then simplified resulting in a tree structure made of chains of edges and nodes. Selected nodes are interpreted as joints of a skeleton, and the chains connecting them as its limbs, as illustrated by Fig.1. A spring network is then produced to bind the skeleton to the boundary, so that skeletal motions will update the surface boundary as specified by the animator.

Wade *et al.* [WP02a] also describe an algorithm decomposed in automatic steps for generating a control skeleton. The main process consists in discretizing the 3D shape by voxelization, computing the corresponding Discrete Medial Surface (DMS), and then using the DMS both to create the skeletal structure and to attach the vertices of the model to that structure. Unlike previous methods, the algorithm is fully automated, requiring very little user input. Lazarus *et al.* [LV99] provide an approach that is expected to overcome the limitations of medial axis for boundary-based representations. The proposed paradigm constructs one dimensional

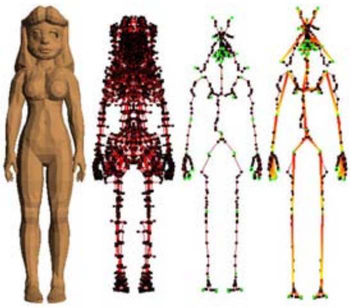

Figure 1: *Generation of control skeleton from approximate medial axis of the 3D mesh, [TT98].*

axial structures associated with a polyhedral surface. These structures are called the Level Set Diagrams (LSD). They are associated with scalar functions defined over the set of vertices of a polyhedron. They catch the overall shape and the topology of an object and can be used for deforming or animating an object. The skeletons for a human, a dolphin and a horse are tested in their work, while Fig.2 illustrates the method for a cow. However, the skeleton proposed for the human shape is not accurate for animation as the location of joints and segments do not correspond to anatomical joints and limbs with enough accuracy. This is particularly obvious for the hand. A similar diagram based on the Reeb graph has been investigated by Kanongchaiyos *et al.* [KS00]. An alternate method, based on the computa-

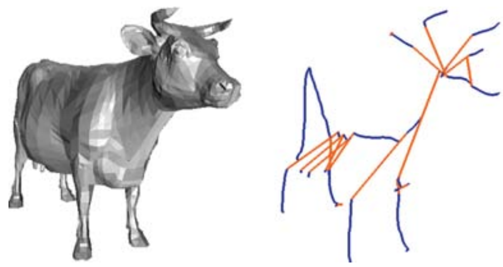

Figure 2: *Skeleton extraction using Level Set Diagrams, [LV99].*

tion of the repulsive force field with ray-casting, has been suggested by Liu *et al.* [LWM*03]. Local minimal points are selected as joint candidates, and a modified thinning algorithm is used to identify the final joints. Skin vertices are then bound to the resulting skeleton for animation using SSD. Du *et al.* [DQ04] propose to use diffusion equations to approximate medial axis of arbitrary 3D objects represented by arbitrary topological polygonal for skeleton-based shape manipulation. The skeletonization method allows user interactions in order to build user-controlled skeleton. Yoshizawa *et al.* [YBS03] propose an automatic method where a control animation structure and the associated skinning information are automatically obtained from a 3D surface. A skeletal mesh is first extracted from a given 3D surface using a Voronoi-based medial axis approximation, see Fig.3(a). This skeletal mesh is then deformed with an interactive FFD scheme, detailed in [YBS02]. During this skeletal deformation process, control points are first associated to the skeletal mesh and FFDs are applied to the skeletal mesh so that it follows the control points' displacements, see Fig.3(b). The 3D surface is further reconstructed according to the skeletal mesh deformation. A 2-steps post-process is finally applied in order to remove unwanted artifacts such as folds, protrusions and global and local self-intersections, see Fig.3(c). Gagvani *et al.* [GS01] propose a voxel skeleton based method to animate a character model made of volume data, and they compute the skeleton of the Visible Human data set. In a first step, the volume data set is skeletonized then a skeleton tree is defined by connecting voxels, prior to animate it using motion capture data. Finally, the volume data set is regenerated to match to the skeleton tree and the volume animation is produced.

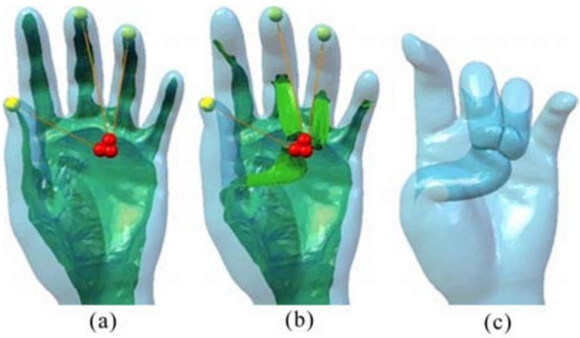

Figure 3: *Two steps (a)(b) to extract information and a sample resulting deformation (c), [YBS03].*

Most of the medial axis-based approaches are sensitive to noise and deformation, which is particularly critical for extracting articulated skeletons from human scan data sets. Moreover, there is no guarantee on the topology of the extracted skeleton. This may not be important for all characters but is mandatory for skeletons with a fixed and predefined topology.

2.1.2. Template-Based Methods

To improve the quality of animation data from motion capture, Silaghi *et al.* [SPB*98] propose a method to identify a skeleton based on the optical marker trajectories compared locally to a skeleton. This idea has been extended by Herda *et al.* [HFP*01]. Carranza *et al.* [CTMS03] also fit a multi-layer skeleton to body animation, using also videos. Seo *et al.* [SCMT03] present a template-based method to generate automatically human body, including the body shape and the control skeleton, from body measurements. The main idea consists in fitting a pre-defined template model to match a set of constraints corresponding to body measurements. The skeleton of the template model is adjusted to match the body measurements. The reconstruction process maintains the coherence between the resulting skin and skeleton. A previous template-based approach has also been proposed to reconstruct body shape from two photographs by Lee *et al.* [LGMT00]. Starck *et al.* [SCS*02] aim at animating high-resolution human surface data captured from commercially available 3D active sensing technology. They apply a model-based approach, by matching a generic control model to the acquired surface data. The generic model is registered with the surface data using a set of interactively defined feature points, referred as landmarks, and joint locations to recover the model posture. The generic model is further automatically fitted to the surface data as a shape constrained deformable surface. A similar approach is proposed by Moccozet *et al.* [MDMT*04] with a full reconstruction pipeline producing a close to animatable approximation of character scanned data, as illustrated by Fig.4. It is based on fitting a human template model as defined by Seo *et al.* [SMT03], and includes both the skin surface and the animation control information of the scanned data. An initial location of landmarks is automatically defined from a multi-scale morphological analysis of the 3D data surface. Reveret *et al.* [RFDC05] propose a method based on principal component analysis for the automatic generation of control skeletons of four legged animals. They use a set of skeletons built by a skilled animator as the learning database. The resulting morphable skeleton model can be adjusted to any 3D quadruped model by taking three measurements on the mesh projected as a side view. Animation is further controlled with smooth skinning for geometry binding.

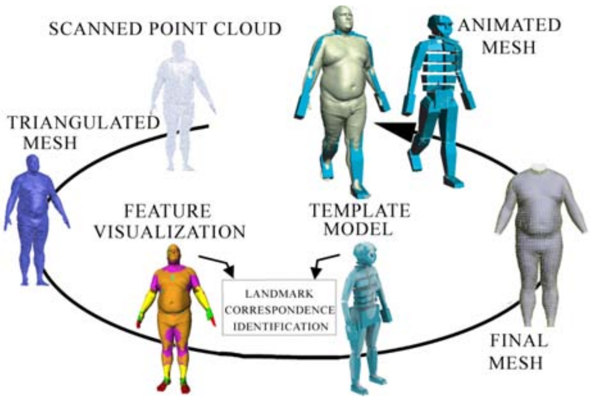

Figure 4: *Template-based reconstruction pipeline, [MDMT*04].*

Template-based methods rely on prior knowledge of the articulated structure of the articulated 3D shape to build. A drawback of this approach is that it is not general and requires producing a template for each family of articulated shape to handle. In addition, these methods usually require features extraction for template matching, *e.g.* landmarks, which is difficult to control in a fully automatic way.

2.1.3. Mesh Decomposition-Based Methods

Although in [JT05] an underlying skeleton is implicitly extracted by clustering triangles with similar rotation sequences, indicating the near-rigid structure of the mesh animation, work with a specific focus on mesh decomposition to extract skeleton have been proposed. Katz *et al.* [KT03] apply a hierarchical decomposition algorithm to define a method for generating and attaching an articulated control skeleton to a given polyhedral surface. Once the main components of the objects have been segmented and identified by the decomposition algorithm, joints are hierarchically positioned between them and attached to the skin surface as shown by Fig.5. A similar technique is proposed by Lien *et al.* [LA05] where the authors use an approximate convex decomposition that partitions a model into nearly convex components. A skeleton of the model is then extracted from the convex hulls of the nearly convex components. This process is iterated until the quality of the skeleton becomes satisfactory. They also demonstrate that this can be used to generate natural skeletal deformations. They compare their results with [KT03] and [WML*03], and show that the skeleton extraction remains stable and robust under perturbation and deformation. The method proposed by Anguelov *et al.* [AKP*04] starts from a set of 3D meshes corresponding to different configurations of an articulated object. The algorithm automatically recovers a decomposition of the object into approximately rigid segments, the location of the segments in the different object instances, and the articulated object skeleton corresponding to the segments, see Fig.6. The algorithm registers the input meshes with a correlated correspondence algorithm. It then iteratively evaluates the segment assignment for each point and the rigid transformation of each segment. Finally, the joints are estimated with articulation constraints.

Figure 5: *Polyhedral surface and associated reconstructed skeleton, [KT03].*

Similarly to the previous family of approaches, segmentation methods do not guarantee the topology of the extracted skeleton, although some of them show that they are quite robust and stable under noise, perturbation and deformation. More precise and stable results can be obtained with methods that use a set of data scans from the same subject in different postures.

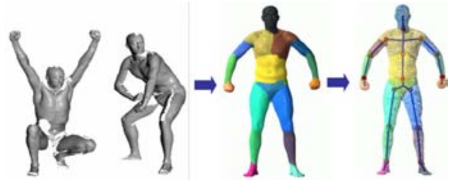

Figure 6: *Automatic decomposition of articulated object into rigid parts, [AKP*04].*

2.1.4. Discussion on Skeleton for Characters

Based on this survey and on Section 3, the following taxonomy regarding articulated skeleton generation for characters can be considered.

1. Direct binding approaches
   A skin shape and a pre-defined skeleton are first produced separately and the skin is then bound to the skeleton. Traditional approaches consist in interactively or semi-automatically placing the skeleton inside the skin and in associating skin vertices to skeleton limbs in order to bind the skin deformations to the skeleton motions. These methods are usually time-consuming and greatly rely on the skills of designers. The accuracy of the skin deformations is closely related to the correct placement of the skeleton with respect to the skin shape. It usually requires a lot of interactive tuning and refinements in order to achieve a correct and appropriate binding.
2. Pseudo-anatomical approaches
   In pseudo-anatomical approaches, intermediate layers are attached to the pre-defined skeleton. They usually mimic the behavior of a muscular layer and reproduce the real, *i.e.* active muscles, or visual behaviour. The geometric primitives associated to the layers can be deformed according to the posture of the skeleton. Once these geometric primitives have been adjusted to match the current skeleton posture, a geometric skin is produced to reflect the posture. The geometric skin can be either generated from the muscle layers at each frame or deformed to match to the shapes of the muscle geometric primitives. The main advantage of these approaches is that the articulated character is iteratively built from the skeleton layer to the skin layer, which creates a close binding between the skeleton motions and the skin deformations. In addition, these approaches allow modeling a wide range of articulated characters and are not limited to a specific type. The main drawback is related to the building of the character, which is usually tedious and time consuming. Additionally, existing methods are mainly focused towards the deformations during motion and do not address the issue of modeling new characters, which usually requires to start from scratch and create all the intermediate layers from the skeleton to the skin. Some methods provide example-based features and propose to adjust some pa-

rameters of the multi-layer structure in order to reflect possible modification of anatomy if the underlying skeleton structure is the same.

3. Skeleton extraction-based approaches
In these approaches, the articulated skeleton is directly extracted from the skin shape. The main advantages are that the resulting skeleton catches the structure of the shape and that the binding between the skeleton and the skin is immediately available from the extraction process. The drawback is that extracted skeletons are usually too complex to reflect the skeleton structure and that it is not immediately usable. Although it does not provide joint axis of the articulated structure, this family of methods provides continuously improving results.
4. Example-based approaches
In example-based approaches, a pre-defined articulated template model is fitted to match some data extracted from the instances to model. The main limitation is that these methods are based on pre-existing know-how and are therefore limited to a given range of articulated models: a new template has to be built for each kind of character. On the other hand, the binding between the skin and the skeleton is immediate and depending on the methods used to fit the template to the input data, it is possible to get a multi-modal reconstruction scheme to handle low and high levels data input, including sparse and incomplete data, for the reconstruction.

New approaches are currently investigated that do not limit capturing techniques to catch the static shape or the skeleton motion but instead consider at the same time dynamic shapes and skeleton motions in order to automate the building and deformation of articulated character. Sand *et al.* [SMP03] propose a method for the acquisition of deformable human geometry from silhouettes. The SCAPE method (Shape Completion and Animation for PEople) presented by Anguelov *et al.* [ASK*05] is a data-driven method for building a character shape model based on a database of 3D scan data from different people and poses. In [PH06], Park *et al.* present a technique for capturing and animating human body motions using a commercial motion capture system with approximately 350 markers.

In the two first approaches, the skeleton is pre-defined and the skin shape is later either attached and connected to the skeleton or fleshed and skinned with pseudo anatomical shapes. The critical issue is then to properly attach of flesh the skeleton so that the resulting shape deformations accurately follow the skeleton motions. We will survey in the next section the different related approaches.

In the third approach, the skeleton is explicitly extracted from a single shape or from a set of shapes in different poses. Most of the time the automatic extraction defines also a parameterization of the skin shape with respect to the skeleton, which can further be used for animation. We will see in the next section that data driven skinning methods can be considered as an extension of these approaches and are used to extract an implicit or explicit skeleton together with skinning information from a set of input shapes.

In the last approach, pre-defined models are used as templates and deformed to match a set of input information to produce new instances. The skeleton and the skinning information is automatically adjusted to the new instance. As we will see in the next section, the investigation of hybrid methods combining example-based and skeleton extraction-based approaches should lead to the development of robust and accurate frameworks for (re)construction of deformable articulated characters. Data-driven techniques can be considered as the emerging combination of skeleton extraction and template-based methods as they use real shape data to define a deformable model of shapes with explicit or implicit skeletons.

3. Deformation and Skinning

Many different approaches have been proposed to connect a deformable skin to its underlying skeleton. They can be roughly subdivided into three main categories: skeleton-based deformations, physics-based approaches, and example-based approaches.

3.1. Skeleton-Based Deformations

3.1.1. Sub-Space Deformations

The skeleton-driven deformation, a classical method for basic skin deformation is probably the most widely used technique in 3D character animation. In research literature, an early version was presented by Magnenat-Thalmann *et al.* [MTLT88] who introduced the concept of joint-dependent local deformation with operators to smoothly deform the skin surface. This technique has been given various names such as sub-space deformation (SSD), linear blend skinning, or smooth skinning. This method works first by assigning a set of joints $\{1,...,n\}$ with weights to each vertex v of the character. The deformation of a vertex v_0 is then computed by a weighted combination of the transformation T_j of the influencing joint j with the following function:

$$S(v_0) = (\sum_{j=1}^{n} w_j T_j).v_0$$

The skeletal deformation makes use of an initial neutral pose. While this method provides fast results and is compact in memory, its drawbacks are the undesirable deformation artifacts in case of important variation of joint angles among the influencing joints.

In [MTG03], Mohr *et al.* addresses a crucial issue related to linear blending skinning: setting the appropriate vertex weights values. Their paper surveys existing linear blending skinning and skin authoring methods. The most current approach consists in using a 3D color painting interface to interactively paint the vertex weights directly onto the skin surface using color gradation. Although the painting interface may seem attractive for designer it is difficult to assess the resulting deformations from the painted weights values. As stated by the authors, "*all control of the deformation of a linear blend skin is achieved by varying each vertex's influences and weights. Unfortunately, setting these parameters is difficult because deformed vertex positions are only implicitly related to the vertex weights.*" Their approach is expected to release users from having to interactively assign weights by letting them adjust deformations directly. Designers may directly position deformed vertices using any vertex manipulation or mesh sculpting tool. An algorithm computes the range of possible deformations, projects these desired vertex positions onto this authorized range, and computes the weights automatically. This approach frees designers from having to manually set weights by letting them update deformations directly.

In Wang *et al.* [WP02b], the authors propose an interesting extension of SSD based approaches: the multi-weighting approach assigns each vertex one weight for each coefficient of each influencing joints transformation matrix instead of a single weight per influencing joint. It therefore defines a new deformation function with a high dimensional input space: 12 weights/vertex/bone, to provide the resulting skinning approach with an increased flexibility to control and transform each skin vertex. As it would be almost impossible to define the multi-weights values interactively, an automated process is proposed in order to automatically estimate the set of values. The authors propose to derive the multi-weights from a set of examples poses. This initial set of poses can be produced from any variety of sources. Each bone must also be assigned with an influence map, which defines its influence over the surface vertices. The influence maps are user-defined. The user must also provide a scaling parameter in order to control the smoothness of the deformation solution and avoid overfitting, when large weights deform skin badly on a new pose. The training poses are then used as examples to estimate the multi-weight vectors using a statistical analysis (Principal Component Analysis) and a modified least square optimization methods. The weights computed from the set of training poses generalize to other postures.

Several attempts have been made to overcome the limitation of geometric skin deformation by using examples of varying postures and blending them during animation. Aimed mostly at real-time applications, these example-based methods essentially seek for solutions to efficiently leverage realistic shapes that come either from captured skin shape of real people, physically based simulation results, or sculpted by skilled designers. Mohr *et al.* [MG03] have presented an extension to SDD by introducing pseudo-joints. The skeleton hierarchy is completed with extra joints inserted between existing ones to reduce the dissimilarity between two consecutive joints. These extra joints can also be used to simulate some nonlinear body deformation effects such as muscle bulges. Once all the extra joints have been defined, they use a fitting procedure to set the skinning parameters of these joints. The weights and the dress position of the vertices are defined by a linear regression so that the resulting skin surface fits to example body shape designed by artists. Having weights well defined, those examples could be discarded during runtime. Mohr *et al.* [MG03] state that the Multi-Weighting Envelop [WP02b] is most similar to their approach: both approaches propose an extension of linear blend skinning as an underlying deformation model. However, Multi-Weighting Envelop is adding more vertex weights to the model whereas Mohr *et al.* adds more joints.

Kavan *et al.* [KZ05] are also aiming at addressing the intrinsic limitations of linear blending such as collapsing joints or twisting problem. Their basic idea consists in moving to another interpolation domain. To this end, they interpolate the transformations instead of the transformed vertex positions. They use quaternions to represent transformations consisting as translation and rotation. As they move to nonlinear interpolation domain with quaternions, they have to cope with two problems for which they propose accurate solutions: estimation of the rotation center and the interpolation of multiple quaternions. The main author has proposed an extension of this initial work in [KCZO07] with dual quaternion blending. The resulting method can easily be implemented in GPU for real-time environments. An additional advantage of the proposed approach is that it does not require additional inputs with respect to the traditional linear blending skinning and existing skinning systems can be easily adapted to quaternion based skinning. The authors however state the limitations of their approach: dual quaterninon blending skinning does not help to produce realistic musculature or dynamic effects and is not suitable for models whose parts can scale or shear such as cartoon characters.

Recently, Merry *et al.* [MMG06] proposed a framework called Animation Space, for generalizing SSD by keeping it linear and with the goal to reduce initial shortcomings of SSD (especially collapses during bending).

3.1.2. Skin Mapping

An example of this approach is proposed by Thalmann *et al.* [TSC96] where ellipsoids are attached to the control skeleton to define a simplified musculature. The main parameters of the ellipsoids, the axis, are depending on joint angles. To produce the final skin for rendering, the ellipsoid muscles are combined together in metaballs. Cani *et al.* [CGD97] propose to use an automatic skinning algorithm based on swept surfaces. The skeleton is used as a path for extrusion of interpolating spline surfaces. This approach is called automatic fleshing of skeletons and it wraps a regular skin around the skeleton of an object. This skeleton coating process is based on various surface operators proposed in order to create complex shapes from simple surfaces such as welding and pinching. Skin binding is achieved by linking the skeleton to the skin with stiff springs.

Figure 7: *Layers of the elastic surface layer model, [TG98].*

Turner *et al.* [TG98] propose a more sophisticated multi-layers structure, see Fig.7, where the skin is modeled as an elastic surface, muscles as ellipsoids, and connective tissues as springs connecting the skin surface to the muscles.

Generally, it appears that in skin mapping the skin is produced separately as a geometric surface and attached to the bones. Intermediate data structures or layers can also be used in order to connect the skin to the skeleton. Two main constraints have to be considered:

- The skin has to follow the motion of the skeleton and to keep globally the same relative position;
- The skin has to mimic the real behavior of the physiology and anatomy of the character simulated. The shape of muscles is changing according to the posture and therefore local deformations have to be controlled and adjusted along the limbs and at the joints.

The first issue consists in segmenting the skin and in binding each skin segment to the bone segment(s) that is(are) expected to control it during motion. Then, it is required to define a local representation system so that each skin segment can be represented locally with respect to the influencing bone segments. Once the local representation is defined between the skin and the skeleton, skeleton driven deformation can be applied to deform the skin when the skeleton is in motion. In most of the skeleton driven deformation models, the articulated character is made of two layers: skin and skeleton. More complex models include intermediate layers in order to improve and refine the control of the skin. For example, a complex layered elastic framework for animated characters is proposed by Chadwick *et al.* [CHP89]. An intermediate muscle layer is introduced between the flexible skin and the rigid skeleton. This intermediate structure is made of Free-Form Deformation (FFD) control boxes attached to the skeleton. The control boxes shapes are parameterized with the joint angles of the skeleton. Two main deformation operators are applied to the control boxes, the first one control the bending of the mesh around the joint and the second one mimic the muscle behavior by inflating the skin along the bones to replicate muscle contraction effect. Similar extended and optimized approaches have been proposed in Moccozet *et al.* [MMT97] where the deformable skin is embedded inside a control lattice. Singh *et al.* [SK00] propose an alternate intermediate structure which consists of a low resolution mesh instead of a control lattice.

In [KG05], Kho *et al.* propose a technique for implicitly defining skeleton based deformation of unstructured polygon meshes. The implicit skeleton is defined by interactively sketching curves onto the 3D shape in the image plane. The proposed approach is quite similar to a family of Free-Form Deformations known as axial-based deformations [Cha94, LCJ93, QP97]. However the user interactions seem much more intuitive. The designer draws a pair of curves on the image plane. The second curve is interpreted as a deformation of the first one. The reference curves drawn by the user on the image plane are used to simultaneously segment the mesh, serve as a control handle for the deformation, and define the scalar field to parameterize the region to deform. The reference curve is scaled and bent to match the target curve. The vertices of the mesh are connected to the reference curve in order to be translated and rotated along with the reference curve.

In [ZHS*05] Zhou *et al.* describe a deformation system providing a close interaction scheme: the designer initially specify a curve on the mesh surface, called the original control curve. The resulting 3D curve is projected onto one or more planes to obtain 2D editable curves. Once edited, the modified 2D curves are projected back to 3D to get the deformed control curve, which is used to control the deformation. The deformation model is based on the volumetric graph Laplacian (VGL). A graph representing the volume inside the initial mesh is constructed, which encodes volumetric details as the difference between each point in the graph and the average of its neighbors. Volumetric details are preserved by minimizing a quadric energy function in a least-squares sense, distributing error uniformly over the whole deformed mesh. The authors apply the deformation scheme to deformation retargetting. However it could also be used to control articulated characters.

3.1.3. Anatomically-Based

Skeleton skinning has also been investigated to simulate anatomy and reproduce as closely as possible the anatomical behavior of characters, such as the work of Scheepers *et al.* [SPCM97] and Wilhem *et al.* [WV97], see Fig.8. Anatomy based modeling considers that using flexible surfaces at joints and along the limbs is an oversimplification and thus adopts a similar approach as artistic anatomy, assuming that the skin surface changes may be more accurately represented as the result of the behavior of the underlying anatomical structures such as muscles, fat and connective tissues. Complex volumes are used to approximate the real shape of muscles with tendons attachments to the bones. Various approaches have been investigated:

- Passive muscles, where geometric or physics-based deformations are used to adjust the shape of the muscle to the current posture of the skeleton;
- Active muscles, where biomechanical physics-based deformation models are used to simulate the dynamic behavior of muscles, such as proposed by Chen *et al.* [CZ92].

Muscles elongation and contraction, see equation below [SPCM97], are used to activate the skeleton joints and produce motions:

$$r = (1-t)r_n + ktr_n = (1-t+kt)r_n$$

where k is a tension control parameter, t a tension parameter, r the ratio of the width and height of the muscle, and r_n this ratio in a fully relaxed state.

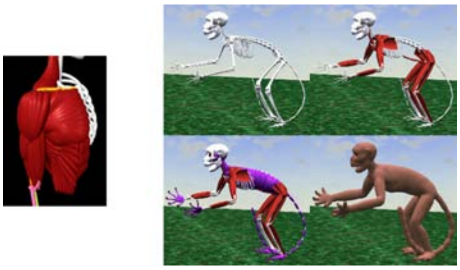

Figure 8: *Realistic musculature structure attached to the skeleton (left), [SPCM97]. Anatomy based modeling of vertebrae animals (right), [WV97].*

Aubel *et al.* [AT01] define a two-layered muscle model suitable for computer graphics applications. A muscle is decomposed into two layers: a skeleton and a surface mesh. The skeleton is simply defined by the action lines of the muscle. This idea is based on the work by Nedel *et al.* [NT98]. Each vertex of the muscle mesh is parametrized by one or two underlying action lines so that the deformations of the muscle shape are driven by the associated action lines. It therefore reduces the 3D nature of the elasticity problem to 1D for fusiform muscles (one action line) and 2D for flat muscles (several action lines). Each action line is approximated by a polyline associated to a 1D mass-spring-damper system. The muscle insertion and origin are anchored to the skeleton joints in order to estimate the action lines deformations from skeleton motions. Collisions avoidance between components is handled with repulsive force fields.

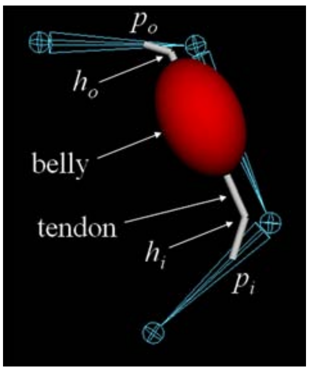

Figure 9: *Structure of a generated ellipsoidal muscle, [PCLS05].*

Pratscher *et al.* [PCLS05] propose a different approach to anatomical models. Rather than modeling internal tissues and attaching a skin to it, they generate a musculature from a skeleton and a skin mesh at a rest pose, with what they called an outside-in approach. Once the muscles are generated, the deformation model can be arbitrary, in their paper they use the procedural model of [SPCM97], and a new geometric model they propose, based on multiple poses and multiple musculatures generation.

In more details, to generate the muscles, first the mesh is partitioned into regions corresponding to bones, with surface normal test direction and Laplacian smoother for surfaces with high-frequency geometry. Then the user optionally specifies a plane for generating a muscle pair, processed by automatic fitting of two ellipsoidal muscles and connecting tendons using four points for each, see Fig.9. Finally, users can refine the layout of the muscle primitives. Concerning the proposed skin deformation model, it is based on radial deformation along the ellipsoids, using several shells (hard, sticky, drop) that the user can vary. After deforming points in local space, they are returned to world space either using the ellipsoid transformation matrix, or with the addition of an accessory copy of the ellipse to stretch the surface, or finally with a parametric curve to provide a trajectory option.

3.2. Physics-Based Approaches

3.2.1. Elasticity and Energy

Since Terzopoulos *et al.* [TF88, TW88], various approaches have been proposed to introduce more realism by introducing dynamic-based deformations such as proposed by Faloutsos *et al.* [FVT97]. Among the most recent ones, Capell *et al.* [CGC*02] introduce a framework for skeleton-driven animation of elastically deformable characters. The proposed framework is somehow similar to FFD-based animation approaches as it embeds the object in a control lattice. However, the authors use continuum elasticity and FEM to compute the dynamics of the object being deformed. Bones of the control skeleton are restricted to lie along the edges of the control lattice, so that the skeleton can be considered as a constraint in the dynamic system. The animator specifies the skeleton and the control lattice interactively, see Fig.10. A joint is created by clicking on the object and it is placed midway between the 2 mouse ray/surface intersections to ensure that joints are centrally located inside the object. By selecting a root joint and defining bones, a complete control skeleton can be defined. The control lattice is also interactively constructed by the user by adding cells incrementally and updating the control vertices as needed. Although the authors stated that existing volumetric meshing schemes should allow the automatic construction of the control lattice, they acknowledged that several hours are required even for an experienced user to create a moderately complex control lattice.

Figure 10: *Interactive construction process for skeleton-driven animation, [CGC*02].*

Recently, Huang *et al.* [HSL*06] have presented a method extending existing gradient domain name mesh deformation approaches. We include it in this STAR since, with their method, skeleton constraints are possible, see Fig.11, and thus it is usable by rigidifying bones for character animation. Actually, they handle both linear and non-linear constraints, such as the skeleton constraints among others, handled as a quasi-linear constraint, with a non-linear energy minimization problem, and solve it with iterative algorithm.

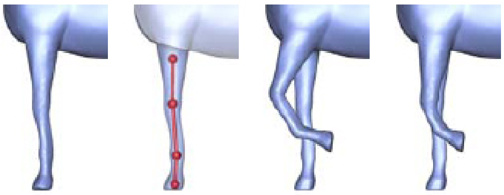

Figure 11: *Deformation with skeleton constraint, third image from the left, and without skeleton constraint, last image, [HSL*06].*

For that, a subspace technique building a coarse mesh around the original mesh is developed and then the deformation energy and the constraints are projected onto the control mesh vertices. This ensures a lower number of vertices, but also a faster and more stable convergence due to the smoothness of the mean value used for the projection. Practically, the user draws strokes corresponding to bones, and the method generates automatically the skeleton segments and associated constraints.

3.2.2. Rigging Dynamics

As stated by Capell *et al.* [CBC*05], "elastic simulation has proved to be a powerful method both for automatically creating plausible skeleton-dependent deformations and for introducing secondary motions". Elastic simulation has been recently extended by Capell *et al.* [CBC*05] to provide more control and flexibility at interactive rate by introducing force-based rigging. Force-based rigs provide more flexibility to the animator for controlling the shape, other than by only moving the skeleton, as illustrated by Fig.12. The realism of the deformations is improved by introducing a collision scheme for managing collisions near creases. Rig forces can be computed from sculpted or measured surface deformations. Moreover, rigs can be transferred between characters.

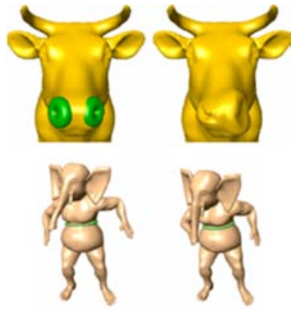

Figure 12: *Sample rig on a cow with resulting deformations, and constrained deformations on a Ganesh-like character, [CBC*05].*

Larboulette *et al.* [LCA05] present flesh elements to add secondary dynamic effects to the skinning while Guo *et al.* [GW05] propose a pseudo-anatomical skinning method. In this work, their purpose is to provide a simple intermediate layer between the skeleton and the skin, namely the chunks, to avoid the tedious design of the anatomical layers. Their hypothesis is that internal anatomical structures do not need to mimic the real ones. A chunk is a deformable structure modeled as a set of nodes connected by links in a multi-sliced structure, automatically extracted from a few patches designed on the skin by the user. All chunks are connected and constitute a continuous layer between the skeleton and the skin. When large volume of space needs a huge amount of nodes to fill it up, it is possible to reduce computational cost by only taking a thinner layer of space between the skin and a base shell attached to the skeleton. Deformations of chunks, controlled with a finite element method (FEM), are adjusted according to the skeleton posture. The authors demonstrate that their approach is able to depict most of the visual aspects of skin deformations generated by real internal anatomical structure and particularly by muscle contractions. However, the final visual results and the computational costs greatly rely on the designed chunks architecture and therefore on the designer's skills.

3.3. Example-Based Approaches

3.3.1. Pose-Space Deformation

The idea of Pose Space Deformation (PSD) [LCF00] approaches is to use artistically sculpted skin surfaces of varying posture and blending them during animation. Each vertex on the skin surface is associated with a linear combination of radial basis functions (RBF) that compute its position given the pose of the moving character, see Figure 13.

These functions are formed by using the example pairs - the poses of the character, and the vertex positions that comprise skin surface. First, each sample pose p_k is translated to the rest coordinate frame by computing displacements of sample vertex v_k and rest pose vertex, using inverse SSD as follows:

$$d_k = (\sum_{j=1}^{n_joint} w_j.T_j)^{-1} v_k - v_0$$

Then displacements of arbitrary pose can be computed by RBF or normalized RBF with arbitrary pose displacements d_a as:

$$d_a = D(p_a) = \sum_{k=1}^{n_pose} r_k(p_a) d_k$$

where n_{pose} is the number of sample poses, and $r_k(p_a)$ a weight defining the contribution of each sample pose. Recently, Kry *et al.* [KJP02] propose an extension of that technique by using principal component analysis, allowing for optimal reduction of the data and thus faster deformation. Sloan *et al.* [SRC01] have shown similar results using RBF for blending the arm models. 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 DOF, which is potentially large, thus resulting in an improved performance.

Figure 13: *Pose space deformation, [LCF00].*

Kurihara *et al.* [KM04] propose a method to construct an example-based deformable hand model from medical images, and deform it with an extension of the PSD approach, called Weighted Pose Space Deformation (WPSD), see Fig.14. The multiple medical images allow for precise bone shape and skin surface, as well as estimation of joint centers. With PSD, deformations are processed by interpolating sample shapes, which are thus determining the range of possible deformations. For arbitrary poses of the hand and its fingers, a high number of samples would be required. Therefore, taking advantage of independency of deformation with poses, the authors propose an interpolation method based on PSD by including influence of links on vertices, as a weight, in the distance γ computation for obtaining the deformed shape as follows:

$$\gamma_{i,k}(p_a, p_k) = \sqrt{\sum_{j=1}^{n_{joint}} w_{i,j}(p_{a,j} - p_{k,j})^2}$$

while PSD distance is:

$$\gamma_k(p_a, p_k) = \sqrt{\sum_{j=1}^{n_{joint}} (p_{a,j} - p_{k,j})^2}$$

Although it provides good results with less poses, it also involves more computation than the standard PSD.

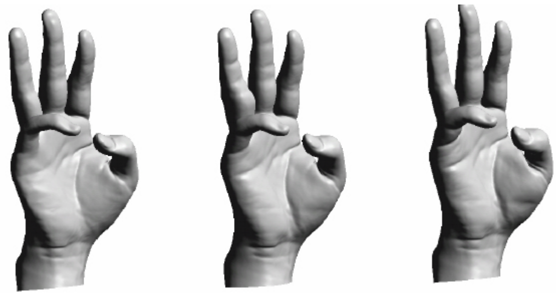

Figure 14: *Comparison of deformation methods, from left to right: SSD, PSD, and WPSD, using the same sample poses, [KM04].*

Following this conclusion, Rhee *et al.* [RLN06] present a WPSD parallel implementation performed on the GPU to achieve real-time performance, around twenty times faster than on CPU. Their algorithm also manages PSD and SSD. Compared to existing GPU algorithms for deformation and skinning, they take advantage of fragment processors and not only vertex processors. Since fragment processors can not access vertex information, texture maps are used to handle the following data: vertices, normals, joint weights, and displacements. For the weights and displacements, the authors reduce the storage required by applying respectively weight values sorting plus a limiting threshold, and PCA of pose space. The algorithm consists of three rendering passes: first, the per-vertex deformation is rendered in a texture map, then normal deformations are computed, and in the last pass, a vertex program transforms the rest pose to the final deformed position. Additionally, they propose an automatic computation of joint weights for each vertex from the sample poses using a non-negative least square method to solve weights minimizing the displacement between the SSD and PSD deformed vertex.

3.3.2. Data-Driven Methods

These methods have recently arisen with the growth of 3D acquisition devices and particularly with full body 3D scanners. These acquisition devices allow now to get reliable and accurate data from complex shapes such as the human body. The basic idea of these approaches is to use real data that cover the space of shape variation of the character to train a deformable model. The underlying hypothesis is that it is possible to generalize from a set if individuals by for example applying statistical analysis to the training set. This family of methods has initially started by modelling static shape variations (morphology) such as the human face [BV99] and has been further applied to address dynamic shape variations (poses) such as deformable human bodies. We have now reached the stage where both types of variations (morphologies and poses) are simultaneously handled.

The method proposed by Allen *et al.* [ACP02] is based on skin pose examples. Similarly to animation by target-morph, they use a set of poses as targets or examples in order to produce the skinning information, whereas traditional approaches rely on much less information, namely a skeleton and a template skin in a neutral posture. If the set of target poses correctly covers the pose space, then the estimated skinning can be applied to any motion. The resulting advantages over other approaches are to be able to introduce the targets as constraints in the skinning and to rely on real or manually designed data, such as the ones obtained from 3D scanners. Moreover, it alleviates the designer's involvement so that the results are less dependent on their skills but still remains user controllable.

The approach recently proposed by James *et al.* [JT05], named Skinning Mesh Animations (SMAs), presents an original skin example-based scheme. The main difference is that it does not require any pre-defined skeleton. The method takes as inputs a set of deformed meshes representing the pseudo-articulated deformable shape in different poses. It then automatically estimates statistically relevant bones based on the hypothesis that clustering triangles with analogous rotation sequences indicates the near-rigid structure of the mesh animation. It further determines bone transformations, bone-vertex influence sets, and vertex weight values for producing skinned animations that approximate the original deformable animation. The skinning approximation is particularly suited for shapes with a sufficient near-rigid structure and does not apply well for highly deformable shapes. SMAs supports hardware rendering and pose editing.

The Multi-Weighting Envelop approach [WP02b] described in section 3.1.1 on sub-space deformations can also be considered as a data-driven method as it requires a set of pre-defined training poses in order to estimate the vertex weights values for skinning the skeleton.

Sand *et al.* [SMP03] propose a method for the acquisition of deformable human geometry from silhouettes. The technique involves a tracking system to capture the motion of the skeleton, and estimates skin shape for each bone using constraints provided by the silhouettes from one or more cameras. The proposed algorithm provides a simple mechanism for solving the problems of view aggregation, occlusion handling, hole-filling, noise removal, and deformation modeling. The resulting model can be parameterized to generate geometry for new poses of the skeleton. The quality of this kind of approach is limited by the amount of detail captured, the accuracy of the skeleton estimation from motion-capture, and the range of motion in the training data set as usual for example-based approaches. The SCAPE method (Shape Completion and Animation for PEople) presented by Anguelov *et al.* [ASK*05] is a data-driven method for building a character shape model. It simultaneously catches the variation in both subject shape and pose. The method is based on a representation that incorporates both skeletal and body deformations. A pose deformation model is trained to derive the body surface deformation as a function of the pose of the articulated skeleton. A second model is trained to acquire the variation of the body shape. The two models are combined to produce 3D surface models with realistic body deformation for different people in different postures.

Optical motion capture is widely used in Computer Graphics to capture real motion data and apply them to articulated skeleton structures to produce realistic animations. Park *et al.* experiment another application of optical motion capture to capture the deformation of the body skin during motion.

In [PH06], they present a technique for capturing and animating human body motions using a commercial motion capture system with approximately 350 markers. The large set of markers is not placed on bony landmarks as usual for motion capture but on muscular and fleshy parts of the body as the goal is to capture not only the motion of the skeleton but also the motion of the surface of the skin. The technical issues are the processing of missing/noisy markers due to the large number of small markers and the occlusion by the different body parts and the deformation of the reference model according to the markers motions as the number of markers is only a sparse representation of the real complete shape of the body. To address the first issue, a local model is assigned to each marker with respect to its neighbors to merge the disconnected segments and fill in the missing markers. The position of occluded markers is estimated from the position of the visible neighboring markers. For the second issue, the marker motion is divided into a set of rigid motions and the local deformation is approximated as a quadratic deformation and its residual.

Another recent approach is proposed by Der *et al.* in [DSP06] extended from [SZGP05]. It transforms a set of unarticulated example shapes into a controllable, articulated model, providing an inverse-kinematics method allowing the interactive animation of detailed shapes using a space learned from a few poses obtained from any source (scanned or hand-sculpted). Their method extracts control parameters of a reduced deformable model with a set of transformation matrices that control shape deformations. The designer is further able to select and move any subset of mesh vertices to pose the complete shape. It uses a nonlinear optimization to get the pose that simultaneously best meets the interactive constraints defined by the designer and gets close to the space of examples. Finally, a linear reconstruction computes the new deformed vertex positions.

4. Conclusion

This STAR has presented existing methods to automate and optimize character representation and animation. It covers a wide variety of research, from advanced skinning and skeleton-driven deformation techniques to automatic extraction and generation of character skeleton for real-time applications.

Although efficient methods and advanced research have been proposed in these different areas, there are some possible future improvements to be investigated such as for instance the development of fast but reliable coupling of skeleton-driven deformation methods with their counterpart in skeletonization, such as physically-based skinning to generate examples for real-time accurate skinning, anatomical definition and automatic extraction of landmark for skeleton extraction on raw scanned data, *etc.*

Skeleton subspace deformations are basically simple approaches. They are appropriate for real-time interactive systems and can be easily implemented with GPUs. However the intrinsic limitations are difficult to overcome and introducing detailed deformations such as muscles buggling is not obvious. Moreover it is difficult to maintain very detailed meshes. The quality of the results is very much related to the accuracy of the vertex weight values. This tedious task and associated guesswork is most of the time delagated to the designers. Only recently, the question of accurately estimating the control parameters has been addressed particularly for estimating vertex weights for blending skinning.

Multi-layers skinning models complexify the modeling of the character and drastically increase the amount of input parameters that designers are expected to provide in order to control the deformations. Anatomy-based models are able to provide dynamic and realistic effets such as muscle buggling. However, they are difficult to build muscle by muscle. It is also difficult to estimate the accurate parameters for controlling the deformations. In [PCLS05] the authors propose a method to automatically extract a few pseudo-muscles from a skin mesh. It would be interesting to investigate and extend data-driven similar approaches in order to automatically estimate these parameters.

Data-driven methods provide detailed results. The main drawback is that they require to provide initial data sets. It requires either using expensive and complex acquisition devices or to use one of the previous skeleton-based animation method to produce the training data set as for target morphing. Moreover, the quality of the results is closely related to the quality of the training data. The intial data set has to cover as much as possible the animated shapes space in order to correctly generalize the animation to any pose. An interesting issue would be to investigate how to retarget the deformations in order to generalize a deformable models to other ones without having to produce a specific training data set for each model. A huge advantage of these methods with respect to previous ones is that designers do not need anymore to explicitly build and connect a skeleton to the skin mesh and to assign vertex weights values. However tedious tasks remain: producing accurate training data sets and control parameters.

In [ACPH06], Allen *et al.* are proposing a very elegant approach to combine skeleton-based deformation model with data-driven models. The skeleton-based deformation is trained to cover variable morphologies and poses from a training data set of 3D scan data. The algorithm is robust with respect to sparse, irregular and incomplete data. Apart from being able to jointly capture morphology dependent and poses dependent shape variation, a major interest of this approach is that by directly training the deformable body model instead of only the body shape, the articulated character can be further animated with traditional skeleton animation techniques such as Motion Capture. As stated by the authors, the main limitations remain the availability of real data, the aptitude of the training set of data to accurately cover the space of poses and morphologies, and the limitations due to the acquisition devices that restrict the possible poses that can be acquired.

Acknowledgements

This work has been partly supported by the AIM@Shape Network of Excellence, grant 506766 by the European Commission. The authors would also like to thank Fabien Dellas for his valuable help.

References

[ACP02] ALLEN B., CURLESS B., POPOVIĆ Z.: Articulated body deformation from range scan data. *ACM Transactions on Graphics 21*, 3 (2002), 612–619.

[ACPH06] ALLEN B., CURLESS B., POPOVIĆ Z., HERTZMANN A.: Learning a correlated model of identity and pose-dependent body shape variation for real-time synthesis. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2006), Eurographics Association, pp. 147–156.

[AKP*04] ANGUELOV D., KOLLER D., PANG H.-C., SRINIVASAN P., THRUN S.: Recovering articulated object models from 3d range data. In *Proc. Uncertainty in Artificial Intelligence Conference, UAI'04* (2004).

[ASK*05] ANGUELOV D., SRINIVASAN P., KOLLER D., THRUN S., RODGERS J., DAVIS J.: Scape: Shape completion and animation of people. *ACM Transactions on Graphics 24*, 3 (2005), 408–416.

[AT01] AUBEL A., THALMANN D.: Interactive modeling of the human musculature. In *Proc. Computer Animation, CA'01* (2001).

[Blo99] BLOOMENTHAL J.: Skeletal methods of shape manipulation. In *Proc. Shape Modeling and Applications, SMI'99* (1999), pp. 44–49.

[BV99] BLANZ V., VETTER T.: A morphable model for the synthesis of 3D faces. In *Siggraph 1999, Computer Graphics Proceedings* (1999), Addison Wesley Longman, pp. 187–194.

[CBC*05] CAPELL S., BURKHART M., CURLESS B., DUCHAMP T., POPOVIĆ Z.: Physically based rigging for deformable characters. In *Proc. Symposium on Computer Animation, SCA'05* (2005), pp. 301–310.

[CGC*02] CAPELL S., GREEN S., CURLESS B., DUCHAMP T., POPOVIĆ Z.: Interactive skeleton-driven dynamic deformations. In *Proc. SIGGRAPH'02* (2002), pp. 41–47.

[CGD97] CANI-GASCUEL M.-P., DESBRUN M.: Animation of deformable models using implicit surfaces. *IEEE Transactions on Visualization and Computer Graphics 3*, 1 (1997), 39–50.

[Cha94] CHANG Y.AND ROCKWOOD A. P.: A generalized de casteljau approach to 3d free-form deformation. In *Proceedings of SIGGRAPH '94, Computer Graphics* (July 1994), ACM Press, pp. 257–260.

[CHP89] CHADWICK J. E., HAUMANN D. R., PARENT R. E.: Layered construction for deformable animated characters. In *Proc. SIGGRAPH'89* (1989), pp. 243–252.

[CTMS03] CARRANZA J., THEOBALT C., MAGNOR M., SEIDEL H.: Free-viewpoint video of human actors. In *Proc. SIGGRAPH'03* (2003), pp. 569–577.

[CZ92] CHEN D. T., ZELTZER D.: Pump it up: Computer animation of a biomechanically based model of muscle using the finite element method. In *Proc. SIGGRAPH'92* (1992), pp. 89–98.

[DQ04] DU H., QIN H.: Medial axis extraction and shape manipulation of solid objects using parabolic PDEs. In *Proc. Symposium on Solid Modeling and Applications* (2004), pp. 25–35.

[DSP06] DER K. G., SUMNER R. W., POPOVIC J.: Inverse kinematics for reduced deformable models. *ACM Transactions on Graphics 25*, 3 (2006).

[FVT97] FALOUTSOS P., VANDEPANNE M., TERZOPOULOS D.: Dynamic freeform deformations for animation synthesis. *IEEE Transactions on Visualization and Computer Graphics 3*, 3 (1997), 201–214.

[GS01] GAGVANI N., SILVER D.: Animating volumetric models. *Graphical models 63*, 6 (2001), 443–458.

[GW05] GUO Z., WONG K. C.: Skinning with deformable chunks. *Computer Graphics Forum 24*, 3 (2005), 373–382.

[HFP*01] HERDA L., FUA P., PLAENKERS R., BOULIC R., THALMANN D.: Using skeleton-based tracking to increase the reliability of optical motion capture. *Human Movement Science Journal 20*, 3 (2001), 313–341.

[HSL*06] HUANG J., SHI X., LIU X., ZHOU K., WEI L. Y., TENG S. H., BAO H., GUO B., SHUM H. Y.: Subspace gradient domain mesh deformation. *ACM Transactions on Graphics 25*, 3 (2006), 1126–1134.

[JT05] JAMES D. L., TWIGG C. D.: Skinning mesh animations. *ACM Transactions on Graphics 24*, 3 (2005).

[KCZO07] KAVAN L., COLLINS S., ZARA J., O'SULLIVAN C.: Skinning with dual quaternions. In *2007 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games* (April/May 2007), ACM Press.

[KG05] KHO Y., GARLAND M.: Sketching mesh deformations. In *Proc. Symposium on Interactive 3D Graphics and Games, SI3D'05* (2005), pp. 147–154.

[KJP02] KRY P., JAMES D., PAI D.: Eigenskin: Real time large deformation character skinning in hardware. In *Proc. Symposium on Computer Animation, SCA'02* (2002), pp. 153–160.

[KM04] KURIHARA T., MIYATA N.: Modeling deformable human hands from medical images. In *Proc. Symposium on Computer Animation, SCA'04* (2004), pp. 355–363.

[KS00] KANONGCHAIYOS P., SHINAGAWA Y.: Articulated reeb graphs for interactive skeleton animation. In *Proc. Modeling Multimedia Information and System* (2000), pp. 451–467.

[KT03] KATZ S., TAL A.: Hierarchical mesh decomposition using fuzzy clustering and cuts. *ACM Transactions on Graphics 22*, 3 (2003), 954–961.

[KZ05] KAVAN L., ZARA J.: Spherical blend skinning: a real-time deformation of articulated models. In *Proc. Symposium on Interactive 3D Graphics and Games, SI3D'05* (2005), pp. 9–16.

[LA05] LIEN J.-M., AMATO N. M.: *Simultaneous Shape Decomposition and Skeletonization Using Approximate Convex Decomposition*. Tech. rep., Texas A&M University, 2005.

[LCA05] LARBOULETTE C., CANI M., ARNALDI B.: Adding real-time dynamic effects to an existing character animation. In *Proc. Spring Conference on Computer Graphics, SCCG'05* (2005).

[LCF00] LEWIS J. P., CORDNER M., FONG N.: Pose space deformations: A unified approach to shape interpolation and skeleton-driven deformation. In *Proc. SIGGRAPH'00* (2000), pp. 165–172.

[LCJ93] LAZARUS F., COQUILLART S., JANCÈNE P.: Interactive axial deformations. In *Modeling in Computer Graphics* (June 1993), Springer-Verlag.

[LGMT00] LEE W., GU J., MAGNENAT-THALMANN N.: Generating animatable 3d virtual humans from photographs. *Computer Graphics Forum 19*, 3 (2000).

[LV99] LAZARUS F., VERROUST A.: Level set diagrams of polyhedral objects. In *Proc. Solid Modeling, SM'99* (1999).

[LWM*03] LIU P., WU F., MA W., LIANG R., OUHYOUNG M.: Automatic animation skeleton construction using repulsive force field. In *Proc. Pacific Graphics, PG'03* (2003), p. 409.

[MDMT*04] MOCCOZET L., DELLAS F., MAGNENAT-THALMANN N., BIASOTTI S., MORTARA M., FALCIDIENO B., MIN P., VELTKAMP R.: Animatable human body model reconstruction from 3d scan data using templates. In *Proc. Workshop on Modeling and Motion Capture Techniques for Virtual Environments, CAPTECH'04* (2004).

[MG03] MOHR A., GLEICHER M.: Building efficient, accurate character skins from examples. *ACM Transactions on Graphics 22*, 3 (2003), 562–568.

[MMG06] MERRY B., MARAIS P., GAIN J.: Animation space: A truly linear framework for character animation. *ACM Transactions on Graphics 25*, 4 (2006), 1400–1423.

[MMT97] MOCCOZET L., MAGNENAT-THALMANN N.: Dirichlet free-form deformations and their application to hand simulation. In *Proc. Computer Animation, CA'97* (1997), pp. 93–102.

[MTG03] MOHR A., TOKHEIM L., GLEICHER M.: Direct manipulation of interactive character skins. In *Proc. Symposium on Interactive 3D Graphics, SI3D'03* (2003), pp. 27–30.

[MTLT88] MAGNENAT-THALMANN N., LAPERRIERE R., THALMANN D.: Joint-dependent local deformations for hand animation and object grasping. In *Proc. Graphics Interface, GI'88* (1988), pp. 26–33.

[NT98] NEDEL L. P., THALMANN D.: Real time muscle deformations using mass-spring systems. In *Proc. Computer Graphics International, CGI'98* (1998), pp. 156–165.

[PCLS05] PRATSCHER M., COLEMAN P., LASZLO J., SINGH K.: Outside-in anatomy based character rigging. In *Proc. Symposium on Computer Animation, SCA'05* (2005), pp. 329–338.

[PH06] PARK S., HODGINS J. K.: Capturing and animating skin deformation in human motion. *ACM Transactions on Graphics 25*, 3 (2006), 881–889.

[QP97] Q. PENG X. JIN J. F.: Arc-length-based axial deformation and length preserved animation. In *Computer Animation 1997* (1997), IEEE press, pp. 86–92.

[RFDC05] REVERET L., FAVREAU L., DEPRAZ C., CANI M.-P.: Morphable model of quadrupeds skeletons for animating 3D animals. In *Proc. Symposium on Computer Animation, SCA'05* (2005).

[RLN06] RHEE T., LEWIS J. P., NEUMANN U.: Realtime weighted pose-space deformation on the gpu. *Computer Graphics Forum 25*, 3 (2006), 439–448.

[SCMT03] SEO H., CORDIER F., MAGNENAT-THALMANN N.: Synthesizing animatable body models with parameterized shape modifications. In *Proc. Symposium on Computer Animation, SCA'03* (2003), pp. 120–125.

[SCS*02] STARCK J., COLLINS G., SMITH R., HILTON A., ILLINGWORTH J.: Animated statues. *Journal of Machine Vision Applications* (2002).

[SK00] SINGH K., KOKKEVIS E.: Skinning characters using surface oriented free-form deformations. In *Proc. Graphics Interface, GI'00* (2000), pp. 35–42.

[SMP03] SAND P., MCMILLAN L., POPOVIĆ J.: Continuous capture of skin deformation. *ACM Transactions on Graphics 22*, 3 (2003), 578–586.

[SMT03] SEO H., MAGNENAT-THALMANN N.: An automatic modeling of human bodies from sizing parameters. In *Proc. Symposium on Interactive 3D Graphics, I3D'03* (2003), pp. 19–26.

[SPB*98] SILAGHI M., PLAENKERS R., BOULIC R., FUA P., THALMANN D.: Local and global skeleton fitting techniques for optical motion capture. In *Proc. Modeling and Motion Capture Techniques for Virtual Environments, CAPTECH'98* (1998), pp. 26–40.

[SPCM97] SCHEEPERS F., PARENT R., CARLSON W., MAY S.: Anatomy-based modeling of the human musculature. In *Proc. SIGGRAPH'97* (1997), pp. 163–172.

[SRC01] SLOAN P. P., ROSE C., COHEN M.: Shape by example. In *Proc. Symposium on Interactive 3D Graphics, I3D'01* (2001), pp. 135–Ű143.

[SZGP05] SUMNER R. W., ZWICKER M., GOTSMAN C., POPOVIC J.: Mesh-based inverse kinematics. *ACM Transactions on Graphics 24*, 3 (2005), 488–495.

[TF88] TERZOPOULOS D., FLEISCHER K.: Modeling inelastic deformation: Viscoelasticity, plasticity, fracture. *Computer Graphics 22*, 4 (1988), 269–278.

[TG98] TURNER R., GOBBETTI E.: Interactive construction and animation of layered elastically deformable characters. *Computer Graphics Forum 17*, 2 (1998), 135–152.

[TSC96] THALMANN D., SHEN J., CHAUVINEAU E.: Fast realistic human body deformations for animation and vr applications. In *Proc. Computer Graphics International, CGI'96* (1996).

[TT98] TEICHMANN M., TELLER S.: Assisted articulation of closed polygonal models. In *Proc. Eurographics Workshop on Animation and Simulation, EGCAS'98* (1998).

[TW88] TERZOPOULOS D., WITKIN A.: Physically based models with rigid and deformable components. *IEEE Computer Graphics and Applications 8*, 6 (1988), 41–51.

[WML*03] WU F.-C., MA W.-C., LIOU P.-C., LAING

R.-H., OUHYOUNG M.: Skeleton extraction of 3d objects with visible repulsive force. In *Computer Graphics Workshop* (2003).

[WP02a] WADE L., PARENT R.: Automated generation of control skeletons for use in animation. *The Visual Computer 18*, 2 (2002), 97–110.

[WP02b] WANG X. C., PHILLIPS C.: Multi-weight enveloping: least-squares approximation techniques for skin animation. In *Proc. Symposium on Computer Animation, SCA'02* (2002), pp. 129–138.

[WV97] WILHELMS J., VANGELDER A.: Anatomically based modeling. In *Proc. SIGGRAPH'97* (1997), pp. 173–180.

[YBS02] YOSHIZAWA S., BELYAEV A. G., SEIDEL H.-P.: A simple approach to interactive free-form shape deformations. In *Proc. Pacific Graphics, PG'02* (2002), pp. 471–474.

[YBS03] YOSHIZAWA S., BELYAEV A. G., SEIDEL H.-P.: Free-form skeleton-driven mesh deformations. In *Proc. ACM Solid Modeling* (2003), pp. 247–253.

[ZHS*05] ZHOU K., HUANG J., SNYDER J., LIU X., BAO H., GUO B., SHUM H. Y.: Large mesh deformation using the volumetric graph laplacian. *ACM Transasctions on Graphics 24*, 3 (2005), 496–503.