Skin Deformation Methods for Interactive Character Animation

Nadine Abu Rumman[1](✉) and Marco Fratarcangeli[2]

[1] Department of Computer, Control, and Management Engineering,
Sapienza University of Rome, Via Ariosto 25, 00185 Rome, Italy
aburumman@dis.uniroma1.it
[2] Department of Applied Information Technology,
Chalmers University of Technology, Lindholmsplatsen 1,
41296 Gothenburg, Sweden
marcof@chalmers.se
http://www.dis.uniroma1.it/~aburumman/
http://www.fratarcangeli.net/

Abstract. Character animation is a vital component of contemporary computer games, animated feature films and virtual reality applications. The problem of creating appealing character animation can best be described by the title of the animation bible: "*The Illusion of Life*". The focus is not on completing a given motion task, but more importantly on how this motion task is performed by the character. This does not necessarily require realistic behavior, but behavior that is believable. This of course includes the skin deformations when the character is moving. In this paper, we focus on the existing research in the area of skin deformation, ranging from skeleton-based deformation and volume preserving techniques to physically based skinning methods. We also summarize the recent contributions in deformable and soft body simulations for articulated characters, and discuss various geometric and example-based approaches.

Keywords: Character animation · Deformation · Skinning · Skeleton-based animation · Volume-preserving · Physics-based animation

1 Introduction

Producing believable and compelling skin deformations for articulated characters is a multi-disciplinary problem, which can be divided into three main problems: generating high-quality skin deformations, simulating skin contact in response to collisions, and producing secondary motion effects such as flesh jiggling when a character moves. Traditionally, bone transformations describe the position and orientation of the joints, and the skin deformation is computed by linearly blending bone transformations to the skin. However, such simple and linear blending to the bone transformations cannot be expected to capture complex deformations. In contrast, by employing a physically based method into the skinning process,

© Springer International Publishing AG 2017
J. Braz et al. (Eds.): VISIGRAPP 2016, CCIS 693, pp. 153–174, 2017.
DOI: 10.1007/978-3-319-64870-5_8

marcof@chalmers.se

the believability of character motions is highly enhanced. Physics-based simulations manage to bring skeleton-based animation beyond the purely kinematic approach by simulating secondary motions like jiggling, volume preservation and contact deformation effects. Despite offering such realistic effects, physically based simulation is computationally demanding and complex, thus it is usually avoided in interactive applications.

The key challenge of producing believable deformations is to satisfy the conflicting requirements of real-time interactivity and believability. Believability requires achieving sufficient deformation detail, which means capturing the full range of desired effects. Producing these deformations demand at least an order of magnitude more computation time than current interactive systems. The aim of this paper is to provide a comprehensive survey on the existing skinning techniques in literature. These techniques can be classified into three categories: skeleton-based deformation (geometry and example-based skinning), volume preserving skinning and physics-based skinning methods. We discuss the existing skinning methods and how they address the above-mentioned problems, and we also highlight the advantages and disadvantages of each method.

2 Skeleton-Based Deformations

The most common approach for deforming articulated character's skin is to define the surface geometry as a function of an underlying skeletal structure. Due to the simplicity, intuitive manipulation, and the ability to quickly solve inverse-kinematics on a small subspace (i.e., the skeleton), skeleton-based methods are very popular and widely used in the animation industry. When modeling a skeleton-based deformation, the challenge is to obtain high-quality skin deformations. The current skeleton-based deformation techniques can be divided into two sections: geometry-based methods (also called smooth skinning, Sect. 2.1) and example-based methods (Sect. 2.2).

2.1 Geometric Skinning Techniques

In geometric skinning techniques, skeleton-to-skin binding is defined in a direct, geometrical way. Geometric approaches to deform articulated characters have shown reasonable results at interactive rates. We start by discussing the standard real-time method "*skeletal subspace deformation*", also known as linear blend skinning (LBS). This method has been widely adopted in real-time applications such as games, for its computational efficiency and straightforward GPU implementation.

Linear Skinning Methods. The fundamental technique to drive the deformation of a character skin is via an underlying skeleton. Among the many proposed skeleton-based deformation techniques, linear blend skinning (LBS) is the most popular technique due to its effectiveness, simplicity, and efficiency

marcof@chalmers.se

[57]. It has been given many different names over the years, including "*skeletal subspace deformation*", matrix palette skinning, enveloping, vertex blending, smooth skinning (Autodesk Maya), bones skinning (Autodesk 3D Studio Max), or linear blend skinning (the open-source Blender). What follows is an explanation of both the concepts and mathematics necessary to understand linear blend skinning. We will also review the algorithm and its limitations. In linear blend skinning, the basic operation is to deform the skin according to a given list of bone transformations (Fig. 1). The formulation of the LBS model requires the following input data:

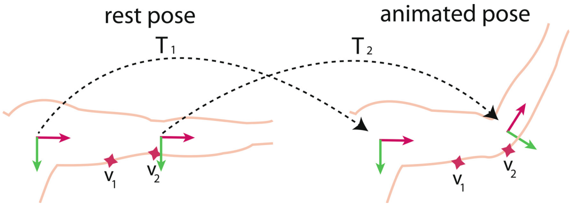

Fig. 1. An example illustrates the main concept of LBS. There are two transformations $\mathbf{T}_1$ and $\mathbf{T}_2$, corresponding to the transformations of shoulder and elbow joints from the rest pose to an animated posture.

- **Surface mesh**, a 3D model represented as a polygon mesh, where only vertex positions will change during deformations. We denote the rest pose vertices as $\mathbf{v}_i, \ldots, \mathbf{v}_n$, and $\mathbf{v}_i \in \mathbb{R}^4$ is given in homogeneous coordinates.
- **Bone transformations**, representing the current deformation using a list of matrices $\mathbf{T}_1, \ldots, \mathbf{T}_m \in \mathbb{R}^{4\times4}$. The matrices can be conveniently defined using an animated skeleton. $\mathbf{T}_j$ is the transformation matrix associated with bone j in its current (animated) pose, $\mathbf{R}_j^{-1}$ is the inverse transformation of the same bone in the rest pose $\mathbf{R}$. Additionally, the matrix $\mathbf{R}$ remains constant, so its inverse is also constant and can safely be pre-computed. Each bone transformation influences part of the mesh.
- **Skinning weights**, For each vertex $\mathbf{v}_i$, we have weights $w_{i,1} + \ldots + w_{i,m} \in \mathbb{R}$, in which each $w_{i,j}$ describes the amount of influence of bone j on vertex $\mathbf{v}_i$. The binding weights are normally assumed to be convex, so $w_{i,1} + \ldots + w_{i,m} = 1$ and $w_{i,j} \geq 0$.

The surface mesh is driven by a set of bones. Every vertex is associated with the bones via a bone-vertex bind weight, which quantifies the influence of each bone to the vertices. Therefore, the basic idea behind LBS is to linearly blend the transformation matrices. The skin is deformed by transforming each vertex through a weighted combination of bone transformations from the rest pose. Thus, the final transformed vertex position $\mathbf{v}_i'$ is a weighted average of its initial position transformed by each of the attached bones. The whole process can be expressed with the following equation:

marcof@chalmers.se

$$\mathbf{v}_i' = \sum_{j=1}^{m} w_{i,j}\mathbf{T}_j\mathbf{R}_j^{-1}\mathbf{v}_i \qquad (1)$$

In practice, it is rarely makes sense to attach a vertex to more than four joints. However, some old games systems used a variant dubbed rigid binding, which corresponds to allowing only one influencing bone per vertex. *Rigid binding* leads to unrealistic non-smooth deformations near joints and suffers from self-intersections. With increasing polygon budgets, linear blend skinning quickly replaced rigid binding because it allowed for smooth transitions between individual transformations (see Fig. 2). Linear blend skinning begins by assigning weights for every vertex on the skin mesh to the underlying bones. The binding weights can be constructed automatically based on the distance between the skin vertices and line segments representing the bones, but it is hard to reliably create good weights automatically. Usually the artists must paint the weights on the mesh directly, using their knowledge of anatomy. Technically, the implementation of a linear blend skinning algorithm is straightforward.

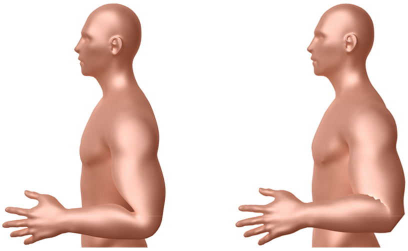

Fig. 2. *Left.* Linear blend skinning. Note the loss of volume at the elbow joint. *Right.* Rigid binding. Note the self-intersections and unnatural deformations in the areas around the elbow joint.

Linear blend skinning works very well when the blended transformations $\mathbf{T}_j$ are not very different. Issues arise if we need to blend transformations, which differ significantly in their rotation components. Despite its fast and straightforward implementation, linear blend skinning suffers from some visible artefacts when we rotate joint more than 90°. In a rotating joint, we expect the skin to rotate around the joint too, maintaining the volume. But the linear blend model instead interpolates skin vertex positions linearly between where the bones expect them, which shrinks the volume of the skin. Therefore, this linearly blending rotation leads to the well-known "*candy-wrapper*" artefact (as we can see in Fig. 3).

Figure 4 illustrates the problems of LBS, which are loss of volume when bending and the "*candy-wrapper*" artefact when twisting. The limitations of LBS have been extensively studied, where many techniques have been proposed to avoid its artefacts. One possibility is to enrich the space of skinning weights, leading to methods which are still linear but offer a wider range of deformations. These methods are called multi-linear skinning techniques [62,86], in which the

marcof@chalmers.se

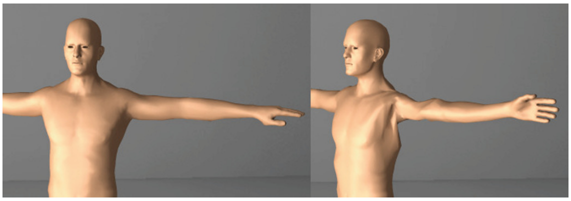

Fig. 3. The well-known "*candy-wrapper*" artefact of linear blend skinning. *Left.* The character model in its rest pose. *Right.* The model deformed with linear blend skinning, where the areas around the shoulder joint suffer from the "*candy-wrapper*" artefact and volume loss when twisting.

extra weights are learned from input examples and regularization is used to prevent overfitting. Merry et al. propose a multi-linear skinning model called *Animation Space* [59], which uses 4 weights per vertex-bone pair. However, this increase in the number of weights carries an additional cost in time and space, as well as parameter passing. While linear skinning techniques are popular due to their efficient implementations, they cannot entirely remove the "*candy-wrapper*" artefact, which is in all cases noticeable under large joint rotations. For a comprehensive survey on linear skinning techniques, we refer the interested reader to [34]. Selecting good skinning weights is critical to avoid the artefacts and generate more natural deformations. Recently, an automatic computation of skinning weights was presented in [16]. In their method, the influence weights are determined using geodesic distances from each bone, which makes the inverse-distance weights shape-aware and can even work with production meshes that may contain non-manifold geometry. Despite that associating skinning weights with the mesh vertices can be done automatically, this method tends to either increase or decrease the volume around joints.

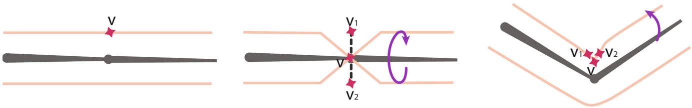

Fig. 4. *Left to right:* The skin in its rest pose. Rigid transformations (express rotation and translation). While twisting, the weighted combination of vertices $\mathbf{v}_1$ and $\mathbf{v}_2$ is not guaranteed to be a rigid transformation, which result in the "*candy-wrapper*" artefact. When bending, the linear interpolation of LBS between the vertices $\mathbf{v}_1$ and $\mathbf{v}_2$ produces $\mathbf{v}$ at an inadequate location, which result in a loss of volume.

Nonlinear Skinning Methods. By replacing linear blending with nonlinear blending [29,42,45], the "*candy-wrapper*" artefact can be completely avoided. Nonlinear skinning methods convert the affine rigid transformation matrices

marcof@chalmers.se

to <*quaternion*, *translation*> pairs, which are then easier to blend than their matrix equivalents. In [29] some constraints are imposed on the character's rigging, whereas *spherical skinning* [45] uses a computationally expensive Singular Value Decomposition (SVD) scheme. The practical impact of these two methods is limited because of their dealing with the translational component of the skinning transformations. In contrast, dual quaternion skinning (DQS [42]) uses an approximate blending technique based on dual quaternions (essentially, two regular quaternions). DQS delivers a rigid transformation in all cases and it is almost as fast as LBS. While the underlying mathematics may not be trivial, an actual implementation of dual quaternion skinning is quite straightforward. Instead of using matrices to express the rigid transformations of the bones, dual quaternion skinning uses the geometric algebra of quaternions. The formulation of DQS is based on dual quaternions [13], which is a generalization of regular quaternions that can be used to express both the translation and the rotation. We are blending rigid transformations, however, the blended matrices $\mathbf{M}$ are no longer rigid transformations under rotations, but general affine transformations (potentially containing scale and shear factors). Therefore, instead of blending matrices $\mathbf{M} = \sum_{j=1}^{m} w_{i,j}\mathbf{T}_j\mathbf{R}_j^{-1}$. In DQS, we blend dual quaternions $\hat{\mathbf{q}}_{\mathbf{j}}$. The figure below illustrates the intuition of why DQS is better than LBS.

To compute the deformed position of a vertex with DQS. First, the transformation matrices are converted to unit dual quaternions. Then, these unit dual quaternions are blended linearly, similarly to linear blend skinning. In other words, we blend dual quaternion transformation $\hat{\mathbf{q}}_{\mathbf{j}}$, weighted by $w_{i,j}$. The result is normalized with $\| \sum_{j=1}^{m} w_{i,j}\hat{\mathbf{q}}_{\mathbf{j}} \|$ to produce the final dual quaternion used to transform a vertex from its rest pose to the deformed position. This guarantees to represent only a rotation and translation. While linearity is lost (due to the projection on unit dual quaternions), the resulting algorithm can be implemented very efficiently. Dual quaternion skinning computes deformed vertex positions according to the following formula (a more detailed explanation on dual quaternion skinning can be found in [43]):

$$\mathbf{v}_i' = \hat{\mathbf{q}}_{\mathbf{j}}\mathbf{v}_i\hat{\mathbf{q}}_{\mathbf{j}}^{*} \tag{2}$$

where $\hat{\mathbf{q}}_{\mathbf{j}} = \frac{\sum_{j=1}^{m} w_{i,j}\hat{\mathbf{q}}_{\mathbf{j}}}{\| \sum_{j=1}^{m} w_{i,j}\hat{\mathbf{q}}_{\mathbf{j}} \|}$ is a unit dual quaternion and $\hat{\mathbf{q}}_{\mathbf{j}}^{*}$ is the conjugate of $\hat{\mathbf{q}}_{\mathbf{j}}$. Unlike matrices, by using dual quaternions there will not be any scale factor that shrinks the mesh around the joints. Dual quaternion skinning thus successfully eliminates the "*candy-wrapper*" effect, but it reveals its own artefact, called joint-bulging artefact (as we can see in Fig. 5). Moreover, dual quaternion skinning has a number of limitations especially in a production environment, as dual quaternions are unable to represent non-rigid transformations, such as shearing. The undesired *joint-bulging* artefact of DQS requires artistic manual work to be fixed. Because fixing these artefacts manually is a tedious process, automatic skinning techniques are becoming increasingly popular [5,12,35].

An interesting extension of linear blend skinning called *spline-skinning* comes from [18,19], which often produces better skinning deformations and suppresses (but not completely eliminates) the "*candy-wrapper*" artefact. Instead of using

marcof@chalmers.se

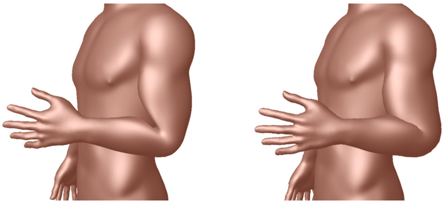

Fig. 5. A demonstration of the artefacts of linear blend skinning (LBS) and Dual quaternion skinning (DQS). *Left.* LBS suffers from volume loss while bending. *Right.* DQS successfully eliminates the "*candy-wrapper*" effect and preserve the volume of the skin, but produces the joint-bulging artefact while bending.

conventional matrix rotation, spline-skinning represents each bone of the skeleton by a spline. Furthermore, an appealing extension of DQS that is successfully applied in a production setting (Disney's *Frozen*), can be seen in [53]. For a extensive discussion on nonlinear skinning methods, we refer the reader to [41]. Whilst all the above-mentioned methods fully define the surface positions based on skeletal configuration, they cannot capture secondary motion effects and skin contact behavior in response to collision. Recently, more advanced geometric skinning methods were introduced to limit the artefacts of LBS, while keeping their simplicity. Kavan and Sorkine [44] developed a new skinning method based on the concept of joint-based deformers, which avoids the artefacts of linear blend skinning as well as the bulging artefact of dual quaternion skinning. More recently technique proposed by Jacobson and Sorkine [36], where they expanded skinning to support bending, stretching and twisting by using a slight variation on the standard skinning equations. Another impressive skinning results can be obtained using the technique presented in [83], which generates visually plausible skin deformations in real-time (see Fig. 6). Their method automatically captures contact surfaces between skin parts, without requiring any collision detection step. Moreover, they extended their framework to handle local skin contacts and produce the effect of skin elasticity (sliding effect) [84]. More recently, [48] proposed a post-processing method for dual quaternion skinning, which eliminates the joint-bulging artefacts and its suitable for real-time character animation.

In spite of improvements, skinning using geometric skinning techniques remains purely kinematic, lacking of secondary motions effects like passive jiggling motion of the fatty tissues or muscle bulging. In the next section, we present the most influential example-based skinning methods, which are able to alleviate the limitations of geometric skinning method, and add dynamic effects to the skin.

2.2 Example-Based Skinning Methods

In contrast to geometric approaches, example-based skinning methods (also known as data-driven techniques) have permitted more complex skinning effects such as muscle bulges and wrinkles, while also addressing the artefacts of linear

marcof@chalmers.se

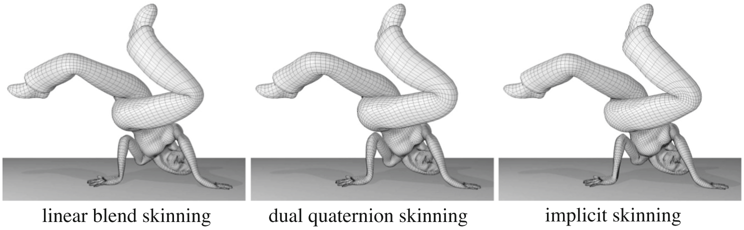

Fig. 6. Dana model in a break-dance pose. From *left* to *right*, the model is deformed with linear blend skinning, dual quaternion skinning and implicit skinning. Note the visible loss of volume produced by LBS (*left*). Implicit skinning (*right*), however, generates visually plausible skin deformations, which avoids the artefacts of linear blend skinning, as well as the bulging artefacts of dual quaternion skinning [83].

skinning techniques. These methods take as input a series of sculpted example poses and interpolate them to obtain the desired deformation. One of the first example-based methods is pose space deformation (PSD [54]), which uses a radial basis function to interpolate correction vectors among the example poses. In pose space deformation method, *pose space* is a set of degrees of freedom of a character's model, which vary between the example poses. A particular *pose* is a set of particular values of these degrees of freedom. Pose space deformation comprises one family of approaches, in which example poses (or *local frame corrections*) are interpolated as a function of a character pose. A more sophisticated extension of PSD was presented in [79]. Their method interpolates an articulated character using example poses scattered in an abstract space. This abstract space consists of dimensions describing global properties of the 3D character, such as age and gender, in addition to dimensions that are used to describe the configuration, such as the amount of bend at the elbow joint. Moreover, PSD was generalized to support weight (weighted pose space deformation WPSD [50,70]), which largely reduces the number of required example poses. Although WPSD can handle large-scale deformations well, it cannot provide detailed deformation and it requires more computation than the original pose space deformation (PSD). In these methods, the amount of memory grows with the number of training examples, thus they are more popular in animated feature film (DreamWorks Animation's Shrek 2) than in real-time application. To tackle this problem, [49] proposed a method similar in spirit to PSD called EigenSkin. Instead of using all the displacements for example poses, they used precomputed principal components of deformation influences on individual joints. The resulting algorithm leads to considerable memory savings and enables to transfer the computations to the GPU. Despite the fact that pose space deformation methods are simple to implement, they require tremendous effort from artists, as they have to create different poses by hand for a wide variety of examples. Another class of example-based methods, which is a direct generalization of LBS, but does not require data interpolation, is formed by methods such as single-weight enveloping (SWE [62]) and multi weight enveloping (MWE [86]). Single-weight enveloping

marcof@chalmers.se

estimated single weight per vertex with rigid character bones, with provisions made for adding additional bones. Multi-weight enveloping, however, is based on a linear framework supporting multiple weights per vertex-bone, where it provides better approximations than SWE, but at the cost of 12 weights per vertex-bone, instead of 1 weight per vertex-bone in SWE. This class of methods allows a smaller number of poses to be used to generate a larger number of deformations, while introducing more weight parameters. Thus, these numerous parameters come at a cost of complicated computation of the weights.

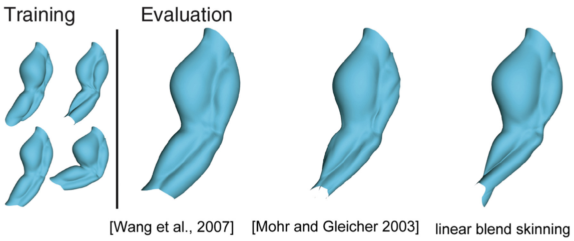

Fig. 7. A set of example poses from an anatomically motivated arm model with both bending and twisting at the elbow. The twisting and muscle bulges are enough to prevent LBS from approximating the examples well. The technique of [62] does better, but still differ from the given example poses. The model from [85] well-approximate the examples poses. (Image taken from [85].)

As an alternative to using sculpted example poses; several example-based approaches use scanned or photographed data. Early work that uses 3D scanned poses of a human body in character skinning has been presented in [61]. Additionally, the method in [2] creates a high quality posable upper body model from range scan data and markers. In their method, to learn the skinning model, they obtain deformations corresponding to different poses by matching a subdivision surface template to the range data. Recently, more advanced example-based techniques have been effectively integrated with mesh deformation algorithms to further improve the quality of skinning [32,77]. A rotational regression model was proposed in [85], which captures common skinning deformation such as muscle bulging (as we can see in Fig. 7) and twisting, specifically in challenging regions such as the shoulders. Park and Hodgins also introduced an interesting technique that captures and synthesizes detailed skin deformations such as bulging and jiggling [66,67], when a character performs dynamic activities. They use a very dense and large set of markers to capture the dynamic motions. Then, they employ a second-order skinning scheme followed by a radial basis function of the residual errors to provide detailed skin deformations. While high-quality skin deformations can be captured accurately using scanned data, marker-based motion capture systems typically have a time-consuming calibration process and high hardware cost.

Example-based skinning methods are attractive since they can provide rich details from physical measurements and add realistic secondary deformation to

marcof@chalmers.se

the skeleton-based animations. Shi et al. presented an appealing method that is able to provide the jiggling of the fatty tissues in real-time by taking a surface mesh and a few sample sequences of its physical behavior [77]. Moreover, the method proposed in [32] is capable of synthesizing high resolution hand mesh deformation with rich and varying details, from only 14 examples poses. However, these approaches do not capture detailed soft-tissue deformations on a wide variety of body shapes. This limitation has been addressed by (Dyna [68]), which learns a model of soft-tissue deformations from examples using a high-resolution 4D capture system. Dyna captures surface deformations of the body at high spatial and temporal resolutions, and constructs a mathematical model for relating these deformations to the motion and body shapes of novel characters. The major drawback of example-based methods is the need for example poses. Besides the fact that when the example poses cannot be captured on a real actor, creating these poses requires either tremendous effort from an artist, or a complex physical simulation on a volumetric version of the skin mesh. In both cases, the mesh and its associated skeleton at rest are not sufficient, and further human intervention is required. An interesting discussion on example-based deformation methods, can be found in [17]. In the next section, we discuss several volume preservation methods for skinned characters, which have been proposed to tackle the loss of volume artefact of linear skinning techniques.

3 Volume Preserving Skinning Methods

Volume preservation is an important aspect in the context of skin deformation that has been addressed in a variety of research papers over the last years. Volume preservation methods allow artists to correct the volume changes through the generation of extra bulges and/or wrinkles. The method that has been proposed in [15] is one of the first methods to introduce volume preserving deformation, where they use *local volume controllers* to guarantee volume conservation of implicitly described soft substances. Moreover, *multi-resolution methods* [7] can preserve surface details by decomposing a mesh into several frequency bands. Furthermore, Funck et al. presented an appealing approach that deforms the mesh vertices based on vector field integration [20,21]. However, these two methods are either computationally expensive or do not fit into the standard animation pipeline. Angelidis and Singh developed a skinning algorithm based on a powerful embedding into the volumetric space, which enables to preserve volume locally and globally [3]. In their method, a degree of freedom is left to the artist to control the final shape, although its combination with skinning weights variation along the mesh makes this control somewhat indirect. Recently, [71] presented an automatic volume correction method to model the constant volume behavior of soft tissues. It corrects the resultant deformations of LBS using a set of local deformations. In their work, they used an automatic way to segment an organic shape into a set of regions corresponding to the main muscle and fatty tissue areas, in which volume is computed and locally corrected. Huang et al. employed a nonlinear version of the volumetric graph Laplacian, which

marcof@chalmers.se

features nonlinear volume preservation constraints [33]. Lipman et al. introduce a shape and volume preserving mesh editing technique [55], where meshes are represented by moving frames. These frames are scaled during deformation such that the volumetric shape properties are preserved.

Several impressive works that create an inner scaffolding of spring, which resist compression to maintain volume are described in [30,88]. The method in [88] provides an excellent introduction to these *interior lattice methods*. Lattice-Based freeform deformation (FFD) are widely-used in commercial software (such as Autodesk 3D Studio Max and Maya) for providing smooth deformations and preserving the volume of the skin [60]. For example, Autodesk Maya 2007 supports the notion of *flexors*. This lattice flexor uses a local FFD lattice, which can then be driven by joint transformations. However, the flexors do not support skinning transfer and the use of flexors can require significant setup and tweaking because of the multitude of lattice points. FFD was first formally proposed in [76] both as a representation for free-form solids and as a method for sculpturing solid models. Using FFD, a complex character can be deformed by positioning the control vertices of the coarse control grid. A more general extension of FFD (EFFD) was later presented by [14]. Moreover, Hsu et al. provided a method that directly manipulates the FFDs [31] and the method in [4] uses an independent deformation function to provide a more flexible FFD. Although lattice-based methods give the artist the flexibility of creating the desired deformation, they require additional setup work and the deformation is sometimes difficult to predict.

On the other hand, cage-based skinning techniques consider an appealing way to control the deformation of an enclosed fine-detailed mesh and help to preserve the volume of skin deformations [39,75]. Cage-based techniques can be considered as a generalization of the lattice-based freeform deformation. Instead of a regular control lattice, a cage is defined by a fixed-topology control lattice that is fitted to the character skin. The cage can be seen as a low-resolution abstraction of the character, which enables the user to deform a character using a simpler mesh. Most cage-based deformation methods are special case of linear blend skinning, where the handle (cage vertex) transformations are restricted to be translations and the focus is on choosing the weights. The method presented in [40] uses cage-based deformations to implement skinning templates, which offer a flexible design space within which to develop reusable skinning behavior. In their method, the skeleton drives the motion of the cage vertices using an example-based skinning technique, where the cage smoothly deforms the character model (see Fig. 8). Joshi et al. proposed a powerful cage-based deformation method based on harmonic coordinates [38] for use in high-end character articulation. Their technique guarantees that the influence of each cage vertex is non-negative and falls off with distance as measured within the cage. It generates a pleasing deformation, but computing the harmonic coordinates is not easy. In spite of that, cage-based techniques allow smooth deformation of skin geometry. Posing the cage requires manual manipulation of the cage vertices. For an overview discussion on volume-preserving deformation methods, we refer the

marcof@chalmers.se

reader to [22,65]. Another promising way to preserve the volume of the skin and to achieve realistic deformation is by applying physics-based simulation into the skin layer around the skeleton. The following section describes the vast literature on physics-based methods.

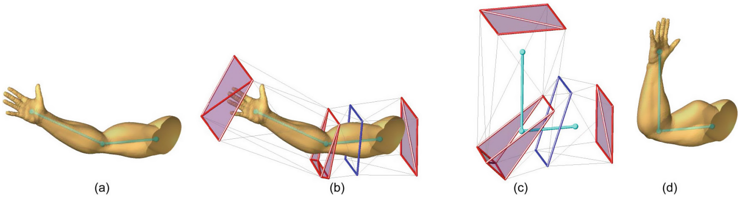

Fig. 8. Skinning with cage: (a) Input geometry with skeleton. (b) An initial cage constructed from four templates, which are associated with the hand joint, elbow joint, upper arm bone, and the shoulder joint. (c) The skeleton deforms the mesh templates. (d) The geometry is deformed by the cage, yielding a non-pinching elbow and muscle bulging. (Image taken from [40].)

4 Physics-Based Methods

In order to model dynamic phenomena, such as the vibration of fatty tissues, muscles bulging and skin contact deformations due to collisions, the animator must configure the deformation for each keyframe. While manually posing a character for each animation keyframe allows artists to create such realistic effects, this process is tedious. Therefore, an alternative method is to employ physics into the skinning process, which highly enhances the believability and realism of character motions. Accordingly, physics-based simulation manages to bring skeleton-driven animation beyond the purely kinematic approach by simulating secondary motions, such as jiggling of soft tissues when the character is moving. Those secondary motions enrich the visual experience of the animation and are essential for creating appealing characters for movie productions and virtual reality applications. After the pioneering work of Terzopoulos et al. [81] and concurrently Lasseter's animation principle "*squash and stretch*" [52]. Physical simulation has taken an important role in the animated feature film industry and computer games, where many physically based methods encouraged to simulate soft bodies or add dynamic effects to the skin. In the following subsections, we first discuss soft body simulations in (Sect. 4.1), and then physically based skinning methods in (Sect. 4.2).

4.1 Deformable and Soft Bodies Simulations

Simulating soft bodies can be achieved in different ways, and the design choice often has to balance the required accuracy and performance. The most popular techniques for simulating soft bodies in computer animation are force-based methods. In particular, most of the techniques used to simulate dynamics rely on

marcof@chalmers.se

mass spring systems, because of the simplicity and efficiency. The general idea is to represent the vertices of the mesh as mass points, governed by Newton's second law of motion, and the edges as elastic massless links (spring). Hence, the mesh is deformed when the lengths of the elastic links change. This happens when the relative position of the mass points changes due to external forces. Mass-spring systems are based on a local description of the material, in which the physics of such systems is straightforward and the simulator is easy to implement. However, to simulate a particular material, it is important to select carefully the parameters of the springs, such as the stiffness and damping. Despite that these systems are fairly easy to implement, they suffer from instability and overshooting problems under large time steps. Moreover, mass-spring systems are often not accurate, since they are strongly topology dependent and are not built based on elasticity theory. On the other hand, finite element methods (FEM) allows to model elastic materials, in which both the masses and the internal and external forces are lumped to the vertices. The vertices in the mesh are treated like mass points in a mass spring system while each element acts like a generalized spring connecting all adjacent mass points. The methods presented in [11,81] are the first to demonstrate the effectiveness of comparatively simple mass-spring based approaches. In their methods, they applied the Lagrangian equations of motion using a finite difference scheme to simulate elastic objects with regular parametrizations. Here, the physical material properties can be described using only few parameters that are used to model soft bodies in an accurate manner. Unlike mass-spring systems, finite element methods are easy to simulate for any particular material. This makes things easier for artists in charge of modelling different types of soft bodies. Unfortunately, finite element methods are avoided in real-time applications, because they are computationally expensive and complex to implement. Various methods have been proposed to address the drawbacks of mass-spring systems and finite element methods [26,51,69]. A comprehensive survey of Nealen et al. [64] provides the details about these techniques. The concept of employing dynamic simulations into skinning for the purpose of character animation was introduced over two decades ago [25], where many techniques were proposed to reduce the accuracy of the simulation to help improve performance and interactivity. Capell et al. [10] used a volumetric finite element mesh to represent the deformation of skin, driven by the underlying skeleton motion. They extended their method to include rigging forces, which guide the deformation to a desired shape [9]. In their method, they effectively handled the effect of skin movement by using skeletal constraints, but by using forces that can violate the conservation of momentum makes their simulation unstable under large time steps. Shinar et al. [78] presented a framework of a two-way coupling between rigid and deformable bodies, in which they use a time integration scheme for solving dynamic elastic deformations in soft bodies interacting with rigid bodies. However, their method does not facilitate the development of an interactive animation system, because of the massive computation required for the finite elements representing the deformable body. In contrast, a possible way to accelerate the simulation of soft bodies is to focus

marcof@chalmers.se

on the surface rather than the volume [8]. In particular, Galoppo et al. [23] presented a fast method to compute the skin deformation on the surface of a soft body with rigid core. Their formulation only considers the elastic energy from skin-layer deformation, and does not include the deformation inside the volume. This may lead to inaccuracies when capturing pose-dependent deformations. All the above methods are only valid for small deformations and are unsuitable for an articulated character's large deformations. On the other hand, Müller et al. [63] achieved a good real-time performances for large rotational deformations, by using a pre-computed linear stiffness matrix to generate the deformations; their method is simple and rotationally invariant. Recently, Kim and Pollard [46] proposed an approach relying on finite element method to simulate the skin deformation, able to handle both one-way and two-way simulations. Their method generates compelling dynamic effects and the deformations are obtained at near interactive rate. Jain and Liu presented a robust approach that realistically simulate characters with soft tissue at the site of contact, where they used two-way coupling between articulated rigid bodies and deformable objects [37]. Liu et al. developed a framework that simulates and controls skeleton driven soft body characters [56]. Their method couples the skeleton dynamics and the soft body dynamics to enable two way interactions between the skeleton, the skin geometry, and the environment at interactive rates (as we can see in Fig. 9).

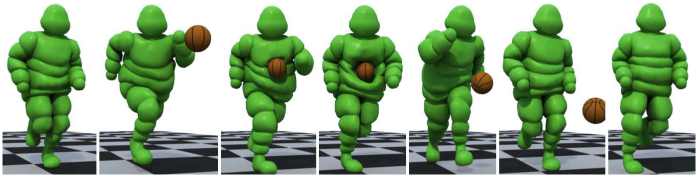

Fig. 9. Employing dynamic simulation into skinning process allows two-way interactions between the skeleton, the skin geometry, and the environment at interactive rates. (Image taken from [56].)

4.2 Physically Based Skinning

Physics-based methods are the natural choice for creating secondary motion effects such as flesh jiggling when a character is moving [24,82]. Turner and Thalmann model the elasticity of skin for character animation and simulate the fat layer by Hookian spring forces [82]. However, they treat muscles as purely controlled elements. Thus, they do not model muscles with deformable methods. Wilhelms [87] presented an approach for animated animals by simulating individual bones, muscles, soft tissues and skin. The use of muscles, soft tissues and flesh elements makes it hard to fit this approach into the skinning framework. Moreover, Hahn et al. [27,28] generated secondary skin dynamics based on the rig degrees of freedom. Their methods simulate the deformation of a character's fat and muscles in the nonlinear subspace induced by the character rig.

marcof@chalmers.se

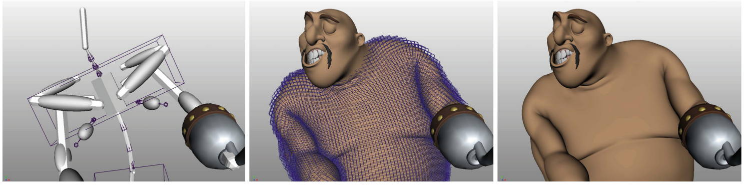

Fig. 10. The Method of [58] takes a skeleton and a surface mesh as input. Based on a hexahedral lattice with 106,567 cells (*center*), their method simulates the deformed surface (*right*) obeying self-collision and volumetric elasticity at 5.5 s per frame. (Image taken from [58].)

In the other direction, Kim and James [47] proposed a domain-decomposition method to simulate articulated deformable characters entirely within a subspace framework, where they combined locally rotated nonlinear subspace models to simulate the detailed deformations of the models. In order to simulate the musculotendons of the human hand and forearm, [80] add anatomic detail using the tendons and bones. While physics-based skinning methods can automatically generate secondary motion with high visual quality, they entail a significant computational burden that slows production and prohibits its use in interactive environments. McAdams et al. [58] presented a robust method using a uniform hexahedral lattice, which provides convincing deformations of the skin with contact handling. In addition, they introduce a one-point quadrature scheme and a multi-grid solver in order to improve the performance and stabilize the simulation. Although their method can capture appealing skin deformations and guarantee pinch-free geometry, it works at best at near interactive performance (as we can see in Fig. 10). Recently, Bender et al. [6] introduced a multi-layer character skinning based on shape matching with oriented particles, used to simulate the elastic behavior of a closed triangular mesh as a representation of a skin model. They make a use of position-based constraints for coupling the skeleton with the skin and handling self-collisions. In [1,72–74], a two-layered approach, the skin is first deformed with a classic linear blend skinning and then the vertex positions are adjusted using Position-based Dynamics. This allows to robustly mimic the behavior of the skin, and achieve and tune effects like volume preservation and jiggling at interactive rate.

5 Conclusion

In this paper, we have reviewed the common techniques for modelling deformations, especially those for character animation purposes. A greater attention was paid to skeleton-based methods, and physics-based methods. In skeleton-based skinning such as linear blend skinning, dual quaternion skinning and pose space deformation, surface deformation is restricted to the skeletal pose that fully defines the surface deformation. While skeleton-based methods can produce good results, the believability of the deformation using these methods is

marcof@chalmers.se

limited. Because they cannot capture secondary motions effects and skin contact deformations. Skeleton-based deformation alone is not sufficient for capturing believable skin deformations, such as skin stretching, secondary motion effects and skin contact due to collisions. In contrast, physics-based simulations bring skeleton-based animation beyond the purely kinematic approach by simulating secondary motions such as jiggling of soft tissues when the character is moving, as well as capturing skin contact deformation. Simulating such flesh-like deformations is difficult due to the coupling between the skeleton and soft skin. Moreover, the resultant deformation has a high number of independent degrees of freedom, in which it does not respect any manipulation done by the artist. Therefore, once the deformation parameters are specified, it is difficult to control the actual resulting shape of the character in every animation frame. Furthermore, physics-based methods are computationally expensive and usually avoided in interactive applications. In the future skinning research, the computational process for obtaining believable skin motion must trade-off between these requirements: it must (1) be fast enough to achieve interactive rate (i.e., >30 fps), (2) produce believable animation to minimize manual post-processing time, and (3) be controllable and stable.

Acknowledgements. The authors would like to express their deepest thanks to Dr. Bart de Keijzer for the careful proof-reading of this paper.

References

1. Abu Rumman, N., Fratarcangeli, M.: Position-based skinning for soft articulated characters. Comput. Graph. Forum **34**(6), 240–250 (2015). doi:10.1111/cgf.12533
2. Allen, B., Curless, B., Popović, Z.: Articulated body deformation from range scan data. In: Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2002, NY, USA, pp. 612–619 (2002). http://doi.acm.org/10.1145/566570.566626
3. Angelidis, A., Singh, K.: Kinodynamic skinning using volume-preserving deformations. In: Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA 2007, San Diego, California, USA, 2–4 August 2007, pp. 129–140 (2007). http://doi.acm.org/10.1145/1272690.1272709
4. Aubert, F., Bechmann, D.: Volume-preserving space deformation. Comput. Graph. **21**(5), 625–639, September 1997. http://dx.doi.org/10.1016/S0097-8493(97)00040-X
5. Baran, I., Popović, J.: Automatic rigging and animation of 3D characters. In: ACM SIGGRAPH 2007 Papers, SIGGRAPH 2007, NY, USA (2007). http://doi.acm.org/10.1145/1275808.1276467
6. Bender, J., Müller, M., Otaduy, M.A., Teschner, M.: Position-based methods for the simulation of solid objects in computer graphics. In: EUROGRAPHICS 2013 State of the Art Reports. Eurographics Association (2013)
7. Botsch, M., Kobbelt, L.: Multiresolution surface representation based on displacement volumes. Comput. Graph. Forum **22**(3), 483–491 (2003). doi:10.1111/1467-8659.00696

marcof@chalmers.se

8. Bro-nielsen, M., Cotin, S.: Real-time volumetric deformable models for surgery simulation using finite elements and condensation. In: Computer Graphics Forum, pp. 57–66 (1996)
9. Capell, S., Burkhart, M., Curless, B., Duchamp, T., Popović, Z.: Physically based rigging for deformable characters. In: Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA 2005, NY, USA, pp. 301–310 (2005). http://doi.acm.org/10.1145/1073368.1073412
10. Capell, S., Green, S., Curless, B., Duchamp, T., Popović, Z.: Interactive skeleton-driven dynamic deformations. In: Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2002, NY, USA, pp. 586–593 (2002). http://doi.acm.org/10.1145/566570.566622
11. Chadwick, J.E., Haumann, D.R., Parent, R.E.: Layered construction for deformable animated characters. SIGGRAPH Comput. Graph. **23**(3), 243–252 (1989). http://doi.acm.org/10.1145/74334.74358
12. Chen, C.H., Lin, I.C., Tsai, M.H., Lu, P.H.: Lattice-based skinning and deformation for real-time skeleton-driven animation. In: Proceedings of the 2011 12th International Conference on Computer-Aided Design and Computer Graphics, CAD-GRAPHICS 2011, pp. 306–312. IEEE Computer Society, Washington, DC (2011). http://dx.doi.org/10.1109/CAD/Graphics.2011.41
13. Clifford, W.: Mathematical papers. Technical report, June 1882
14. Coquillart, S.: Extended free-form deformation: a sculpturing tool for 3D geometric modeling. In: Proceedings of the 17th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1990, NY, USA, pp. 187–196 (1990). http://doi.acm.org/10.1145/97879.97900
15. Desbrun, M., Gascuel, M.P.: Animating soft substances with implicit surfaces. In: Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1995, NY, USA, pp. 287–290 (1995). http://doi.acm.org/10.1145/218380.218456
16. Dionne, O., de Lasa, M.: Geodesic voxel binding for production character meshes. In: Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA 2013, NY, USA, pp. 173–180 (2013). http://doi.acm.org/10.1145/2485895.2485919
17. Feng, W.W., Kim, B.U., Yu, Y.: Real-time data driven deformation using kernel canonical correlation analysis. In: ACM SIGGRAPH 2008 Papers, SIGGRAPH 2008, NY, USA, pp. 91:1–91:9 (2008). http://doi.acm.org/10.1145/1399504.1360690
18. Forstmann, S., Ohya, J.: Fast skeletal animation by skinned arc-spline based deformation. EG 2006 Short Papers, pp. 1–4 (2006)
19. Forstmann, S., Ohya, J., Krohn-Grimberghe, A., McDougall, R.: Deformation styles for spline-based skeletal animation. In: Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA 2007, Eurographics Association, Aire-la-Ville, Switzerland, pp. 141–150 (2007). http://dl.acm.org/citation.cfm?id=1272690.1272710
20. von Funck, W., Theisel, H., Seidel, H.P.: Vector field based shape deformations. ACM Trans. Graph. **25**(3), 1118–1125 (2006). http://doi.acm.org/10.1145/1141911.1142002
21. von Funck, W., Theisel, H., Seidel, H.P.: Volume-preserving mesh skinning. In: Proceedings of the Vision, Modeling, and Visualization Conference 2008, VMV 2008, Konstanz, Germany, 8–10 October 2008, pp. 409–414 (2008)

marcof@chalmers.se

22. Gain, J., Bechmann, D.: A survey of spatial deformation from a user-centered perspective. ACM Trans. Graph. **27**(4), 107:1–107:21, November 2008. http://doi.acm.org/10.1145/1409625.1409629
23. Galoppo, N., Otaduy, M.A., Tekin, S., Gross, M.H., Lin, M.C.: Soft articulated characters with fast contact handling. Comput. Graph. Forum **26**(3), 243–253 (2007)
24. Gilles, B., Bousquet, G., Faure, F., Pai, D.K.: Frame-based elastic models. ACM Trans. Graph. **30**(2), 15:1–15:12 (2011). http://doi.acm.org/10.1145/1944846.1944855
25. Girard, M., Maciejewski, A.A.: Computational modeling for the computer animation of legged figures. In: Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1985, NY, USA, pp. 263–270 (1985). http://doi.acm.org/10.1145/325334.325244
26. Gourret, J.P., Thalmann, N.M., Thalmann, D.: Simulation of object and human skin formations in a grasping task. In: Proceedings of the 16th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1989, NY, USA, pp. 21–30 (1989). http://doi.acm.org/10.1145/74333.74335
27. Hahn, F., Martin, S., Thomaszewski, B., Sumner, R., Coros, S., Gross, M.: Rig-space physics. ACM Trans. Graph. **31**(4), 72:1–72:8 (2012). http://doi.acm.org/10.1145/2185520.2185568
28. Hahn, F., Thomaszewski, B., Coros, S., Sumner, R., Gross, M.: Efficient simulation of secondary motion in rig-space. In: Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA 2013 (2013)
29. Hejl, J.: Hardware skinning with quaternions. In: Kirmse, A. (ed.) Game Programming Gems, vol. 4, pp. 487–495. Charles River Media, Newton Centre (2004)
30. Hong, M., Jung, S., Choi, M.H., Welch, S.: Fast volume preservation for a mass-spring system. Comput. Graph. Appl. IEEE **26**(5), 83–91 (2006)
31. Hsu, W.M., Hughes, J.F., Kaufman, H.: Direct manipulation of free-form deformations. In: Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1992, NY, USA, pp. 177–184 (1992). http://doi.acm.org/10.1145/133994.134036
32. Huang, H., Zhao, L., Yin, K., Qi, Y., Yu, Y., Tong, X.: Controllable hand deformation from sparse examples with rich details. In: Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA 2011, NY, USA, pp. 73–82 (2011). http://doi.acm.org/10.1145/2019406.2019416
33. 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 Trans. Graph. **25**(3), 1126–1134 (2006). http://doi.acm.org/10.1145/1141911.1142003
34. Jacka, D., Reid, A., Merry, B., Gain, J.: A comparison of linear skinning techniques for character animation. In: Proceedings of the 5th International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa, AFRIGRAPH 2007, NY, USA, pp. 177–186 (2007). http://doi.acm.org/10.1145/1294685.1294715
35. Jacobson, A., Panozzo, D., Glauser, O., Pradalier, C., Hilliges, O., Sorkine-Hornung, O.: Tangible and modular input device for character articulation. ACM Trans. Graph. **33**(4), 82:1–82:12 (2014). http://doi.acm.org/10.1145/2601097.2601112
36. Jacobson, A., Sorkine, O.: Stretchable and twistable bones for skeletal shape deformation. In: Proceedings of the 2011 SIGGRAPH Asia Conference, SA 2011, NY, USA, pp. 165:1–165:8 (2011). http://doi.acm.org/10.1145/2024156.2024199

marcof@chalmers.se

37. Jain, S., Liu, C.K.: Controlling physics-based characters using soft contacts. ACM Trans. Graph. (SIGGRAPH Asia) **30**, 163:1–163:10 (2011). http://doi.acm.org/10.1145/2070781.2024197
38. Joshi, P., Meyer, M., DeRose, T., Green, B., Sanocki, T.: Harmonic coordinates for character articulation. ACM Trans. Graph. **26**(3) (2007). http://doi.acm.org/10.1145/1276377.1276466
39. Ju, T., Schaefer, S., Warren, J.: Mean value coordinates for closed triangular meshes. ACM Trans. Graph. **24**(3), 561–566 (2005). http://doi.acm.org/10.1145/1073204.1073229
40. Ju, T., Zhou, Q.Y., van de Panne, M., Cohen-Or, D., Neumann, U.: Reusable skinning templates using cage-based deformations. ACM Trans. Graph. **27**(5), 122:1–122:10 (2008). http://doi.acm.org/10.1145/1409060.1409075
41. Kavan, L., Collins, S., O'Sullivan, C.: Automatic linearization of nonlinear skinning. In: Proceedings of the 2009 Symposium on Interactive 3D Graphics and Games, I3D 2009, NY, USA, pp. 49–56 (2009). http://doi.acm.org/10.1145/1507149.1507157
42. Kavan, L., Collins, S., Zára, J., O'Sullivan, C.: Skinning with dual quaternions. In: Proceedings of the 2007 Symposium on Interactive 3D Graphics and Games, I3D 2007, NY, USA, pp. 39–46. ACM, New York (2007)
43. Kavan, L., Collins, S., Zára, J., O'Sullivan, C.: Geometric skinning with approximate dual quaternion blending. ACM Trans. Graph. **27**(4), 105:1–105:23 (2008). http://doi.acm.org/10.1145/1409625.1409627
44. Kavan, L., Sorkine, O.: Elasticity-inspired deformers for character articulation. ACM Trans. Graph. **31**(6), 196:1–196:8 (2012). http://doi.acm.org/10.1145/2366145.2366215
45. Kavan, L., Zára, J.: Spherical blend skinning: a real-time deformation of articulated models. In: Proceedings of the 2005 Symposium on Interactive 3D Graphics and Games, I3D 2005, NY, USA, pp. 9–16 (2005). http://doi.acm.org/10.1145/1053427.1053429
46. Kim, J., Pollard, N.S.: Fast simulation of skeleton-driven deformable body characters. ACM Trans. Graph. **30**(5), 121:1–121:19 (2011). http://doi.acm.org/10.1145/2019627.2019640
47. Kim, T., James, D.L.: Physics-based character skinning using multi-domain subspace deformations. In: Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA 2011, NY, USA, pp. 63–72 (2011). http://doi.acm.org/10.1145/2019406.2019415
48. Kim, Y., Han, J.: Bulging-free dual quaternion skinning. J. Vis. Comput. Anim. **25**(3–4), 323–331 (2014). doi:10.1002/cav.1604
49. Kry, P.G., James, D.L., Pai, D.K.: Eigenskin: Real time large deformation character skinning in hardware. In: Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA 2002, NY, USA, pp. 153–159 (2002). http://doi.acm.org/10.1145/545261.545286
50. Kurihara, T., Miyata, N.: Modeling deformable human hands from medical images. In: Proceedings of the 2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA 2004, Eurographics Association, Aire-la-Ville, Switzerland, pp. 355–363 (2004). http://dx.doi.org/10.1145/1028523.1028571
51. Larboulette, C., Cani, M.P., Arnaldi, B.: Dynamic skinning: adding real-time dynamic effects to an existing character animation. In: SCCG, pp. 87–93 (2005)
52. Lasseter, J.: Principles of traditional animation applied to 3D computer animation. SIGGRAPH Comput. Graph. **21**(4), 35–44 (1987). http://doi.acm.org/10.1145/37402.37407

marcof@chalmers.se

53. Lee, G.S., Lin, A., Schiller, M., Peters, S., McLaughlin, M., Hanner, F.: Enhanced dual quaternion skinning for production use. In: ACM SIGGRAPH 2013 Talks, SIGGRAPH 2013, NY, USA, p. 9:1 (2013). http://doi.acm.org/10.1145/2504459.2504470
54. Lewis, J.P., Cordner, M., Fong, N.: Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2000, pp. 165–172. ACM Press/Addison-Wesley Publishing Co., New York (2000). http://dx.doi.org/10.1145/344779.344862
55. Lipman, Y., Cohen-Or, D., Gal, R., Levin, D.: Volume and shape preservation via moving frame manipulation. ACM Trans. Graph. **26**(1) (2007). http://doi.acm.org/10.1145/1189762.1189767
56. Liu, L., Yin, K., Wang, B., Guo, B.: Simulation and control of skeleton-driven soft body characters. ACM Trans. Graph. **32**(6), 215:1–215:8 (2013). http://doi.acm.org/10.1145/2508363.2508427
57. Magnenat-Thalmann, N., Laperrière, R., Thalmann, D.: Joint-dependent local deformations for hand animation and object grasping. In: Proceedings on Graphics Interface 1988, Canadian Information Processing Society, Toronto, Ontario, Canada, pp. 26–33 (1988). http://dl.acm.org/citation.cfm?id=102313.102317
58. McAdams, A., Zhu, Y., Selle, A., Empey, M., Tamstorf, R., Teran, J., Sifakis, E.: Efficient elasticity for character skinning with contact and collisions. ACM Trans. Graph. **30**(4), 37:1–37:12 (2011). http://doi.acm.org/10.1145/2010324.1964932
59. Merry, B., Marais, P., Gain, J.: Animation space: a truly linear framework for character animation. ACM Trans. Graph. **25**(4), 1400–1423 (2006). http://doi.acm.org/10.1145/1183287.1183294
60. Milliron, T., Jensen, R.J., Barzel, R., Finkelstein, A.: A framework for geometric warps and deformations. ACM Trans. Graph. **21**(1), 20–51 (2002). http://doi.acm.org/10.1145/504789.504791
61. Min, K.H., Baek, S.M., Lee, G.A., Choi, H., Park, C.M.: Anatomically-based modeling and animation of human upper limbs. In: Proceedings of International Conference on Human Modeling and Animation (2000)
62. Mohr, A., Gleicher, M.: Building efficient, accurate character skins from examples. In: ACM SIGGRAPH 2003 Papers, SIGGRAPH 2003, NY, USA, pp. 562–568 (2003). http://doi.acm.org/10.1145/1201775.882308
63. Müller, M., Dorsey, J., McMillan, L., Jagnow, R., Cutler, B.: Stable real-time deformations. In: Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, pp. 49–54. ACM, New York (2002)
64. Nealen, A., Mueller, M., Keiser, R., Boxerman, E., Carlson, M.: Physically based deformable models in computer graphics. Comput. Graph. Forum **25**(4), 809–836 (2006). doi:10.1111/j.1467-8659.2006.01000.x
65. Nieto, J., Susin, A.: Cage based deformations: a survey. In: González Hidalgo, M., Mir Torres, A., Varona Gómez, J. (eds.) Deformation Models. LNCVB, vol. 7, pp. 75–99. Springer, Netherlands (2013). http://dx.doi.org/10.1007/978-94-007-5446-1_3
66. Park, S.I., Hodgins, J.K.: Capturing and animating skin deformation in human motion. In: ACM SIGGRAPH 2006 Papers, SIGGRAPH 2006, NY, USA, pp. 881–889 (2006). http://doi.acm.org/10.1145/1179352.1141970
67. Park, S.I., Hodgins, J.K.: Data-driven modeling of skin and muscle deformation. In: ACM SIGGRAPH 2008 Papers, SIGGRAPH 2008, NY, USA, pp. 96:1–96:6 (2008). http://doi.acm.org/10.1145/1399504.1360695

marcof@chalmers.se

68. Pons-Moll, G., Romero, J., Mahmood, N., Black, M.J.: Dyna: a model of dynamic human shape in motion. ACM Trans. Graph. **34**(4), 120:1–120:14 (2015). http://doi.acm.org/10.1145/2766993
69. Popović, J., Seitz, S.M., Erdmann, M.: Motion sketching for control of rigid-body simulations. ACM Trans. Graph. **22**(4), 1034–1054 (2003). http://doi.acm.org/10.1145/944020.944025
70. Rhee, T., Lewis, J.P., Neumann, U.: Real-time weighted pose-space deformation on the GPU. Comput. Graph. Forum **25**(3), 439–448 (2006). doi:10.1111/j.1467-8659.2006.00963.x
71. Rohmer, D., Hahmann, S., Cani, M.P.: Exact volume preserving skinning with shape control. In: Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA 2009, NY, USA, pp. 83–92 (2009). http://doi.acm.org/10.1145/1599470.1599481
72. Rumman, N.A., Fratarcangeli, M.: Position based skinning of skeleton-driven deformable characters. In: Proceedings of the 30th Spring Conference on Computer Graphics, SCCG 2014, NY, USA, pp. 83–90 (2014). http://doi.acm.org/10.1145/2643188.2643194
73. Rumman, N.A., Fratarcangeli, M.: State of the art in skinning techniques for articulated deformable characters. In: Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications, pp. 198–210 (2016)
74. Rumman, N.A., Schaerf, M., Bechmann, D.: Collision detection for articulated deformable characters. In: Proceedings of the 8th ACM SIGGRAPH Conference on Motion in Games, MIG 2015, NY, USA, pp. 215–220 (2015). http://doi.acm.org/10.1145/2822013.2822034
75. Savoye, Y., Franco, J.S.: Cageik: dual-Laplacian cage-based inverse kinematics. In: Proceedings of the 6th International Conference on Articulated Motion and Deformable Objects, AMDO 2010, pp. 280–289 (2010). http://dl.acm.org/citation.cfm?id=1875984.1876011
76. Sederberg, T.W., Parry, S.R.: Free-form deformation of solid geometric models. In: Proceedings of the 13th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1986, NY, USA, pp. 151–160 (1986). http://doi.acm.org/10.1145/15922.15903
77. Shi, X., Zhou, K., Tong, Y., Desbrun, M., Bao, H., Guo, B.: Example-based dynamic skinning in real time. ACM Trans. Graph. **27**(3), 29:1–29:8 (2008). http://doi.acm.org/10.1145/1360612.1360628
78. Shinar, T., Schroeder, C., Fedkiw, R.: Two-way coupling of rigid and deformable bodies. In: Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA 2008, Eurographics Association, Aire-la-Ville, Switzerland, pp. 95–103 (2008). http://dl.acm.org/citation.cfm?id=1632592.1632607
79. Sloan, P.P.J., Rose III., C.F., Cohen, M.F.: Shape by example. In: Proceedings of the 2001 Symposium on Interactive 3D Graphics, I3D 2001, NY, USA, pp. 135–143 (2001). http://doi.acm.org/10.1145/364338.364382
80. Sueda, S., Kaufman, A., Pai, D.K.: Musculotendon simulation for hand animation. ACM Trans. Graph. **27**(3), 83:1–83:8 (2008). http://doi.acm.org/10.1145/1360612.1360682
81. Terzopoulos, D., Platt, J., Barr, A., Fleischer, K.: Elastically deformable models. SIGGRAPH Comput. Graph. **21**(4), 205–214 (1987). http://doi.acm.org/10.1145/37402.37427

marcof@chalmers.se

82. Turner, R., Thalmann, D.: The elastic surface layer model for animated character construction. In: Thalmann, N.M., Thalmann, D. (eds.) Proceedings of Computer Graphics International 1993. CGS CG International Series, pp. 399–412. Springer, Heidelberg (1993). doi:10.1007/978-4-431-68456-5_32
83. Vaillant, R., Barthe, L., Guennebaud, G., Cani, M.P., Rohmer, D., Wyvill, B., Gourmel, O., Paulin, M.: Implicit skinning: real-time skin deformation with contact modeling. ACM Trans. Graph. **32**(4), 125:1–125:12 (2013). http://doi.acm.org/10.1145/2461912.2461960
84. Vaillant, R., Guennebaud, G., Barthe, L., Wyvill, B., Cani, M.P.: Robust isosurface tracking for interactive character skinning. ACM Trans. Graph. **33**(6), 189:1–189:11 (2014). http://doi.acm.org/10.1145/2661229.2661264
85. Wang, R.Y., Pulli, K., Popović, J.: Real-time enveloping with rotational regression. ACM Trans. Graph. **26**(3) (2007). http://doi.acm.org/10.1145/1276377.1276468
86. Wang, X.C., Phillips, C.: Multi-weight enveloping: least-squares approximation techniques for skin animation. In: Proceedings of the 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA 2002, NY, USA, pp. 129–138 (2002). http://doi.acm.org/10.1145/545261.545283
87. Wilhelms, J.: Modeling animals with bones, muscles, and skin. Technical report, University of California (1994)
88. Zhou, K., Huang, J., Snyder, J., Liu, X., Bao, H., Guo, B., Shum, H.Y.: Large mesh deformation using the volumetric graph Laplacian. In: ACM SIGGRAPH 2005 Papers, SIGGRAPH 2005, NY, USA, pp. 496–503 (2005). http://doi.acm.org/10.1145/1186822.1073219

marcof@chalmers.se