Artist-Oriented Real-time Skin Deformation Using Dynamic Patterns

Masaki Oshita   Kenji Suzuki

Kyushu Institute of Technology
680-4 Kawazu, Iizuka, Fukuoka, 820-8502, Japan
oshita@ces.kyutech.ac.jp suzuki@cg.ces.kyutech.ac.jp

Abstract. In this paper, we propose an easy-to-use real-time method to simulate realistic deformation of human skin. We utilize the fact that various skin deformations such as wrinkles, bulging and protruding bones can be categorized into various deformation patterns. A deformation pattern for a local skin region is represented using a dynamic height map. Users of our system specify a region of the body model in the texture space to which each deformation pattern should be applied. Then, during animation, the skin deformation of the body model is realized by changing the height patterns based on the joint angles and applying bump mapping or displacement mapping to the body model. The pattern deformation process is efficiently executed on a GPU.

Keywords: skin deformation, deformation pattern, height map, GPU.

1   Introduction

Character animation is used in many applications such as movies and computer games. However, creating realistic human skin is still difficult to achieve in computer animations. The rendering technique for human skin has recently been greatly improved [1]. However, simulating skin deformations remains a challenge, because the skin shape deforms in a complex way. Although it is possible to simulate human skin deformations using an anatomy-based physical simulation [14][15], in order to apply such methods to a human model, anatomical information such as bones, muscles, tendons, and tissues needs to be supplied in addition to the conventional human figure model. Specifying such information is both tedious for the designer and requires in-depth anatomical knowledge. Moreover, simulating the deformation requires a great deal of computational time, and as such cannot be used in interactive applications such as computer games and web graphics. Currently the SSD (Skeletal Subspace Deformation) model [9] is used extensively in computer games. The SSD simply deforms the geometry of a human model based on the posture of the skeleton and ensures smoothness of the deformed geometries. However, skin-like deformations are not simulated and it lacks reality.

In this paper, we propose an easy-to-use real-time method to simulate realistic deformation of human skin. We utilize the fact that various skin deformations such as wrinkles, bulges and protruding bones can be categorized into various deformation
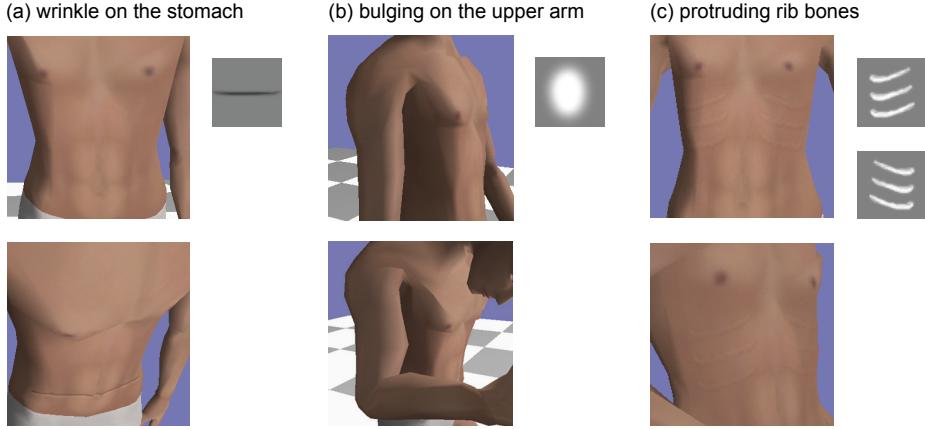
(a) wrinkle on the stomach   (b) bulging on the upper arm   (c) protruding rib bones

Fig. 1. Examples of skin deformations using dynamic height patterns. As the character moves, the height patterns are dynamically changed according to the angle of the corresponding joint.

patterns. A deformation pattern for a local skin region is represented using a dynamic height map. Users of our system specify a region on a body model in the texture space to which each deformation pattern should be applied. Then, during the animation, the skin deformation is realized by changing the height patterns based on the joint angles and applying bump mapping or displacement mapping to the body model.

We propose two deformation models for the dynamic patterns (Fig. 1). In the first model, the height patterns are scaled in response to the corresponding joint angle. This model is used to represent wrinkles and bulges (Fig. 1(a) and (b)). In the second model, the height patterns are biased in response to the corresponding joint angle. This model is used to represent protruding bones (Fig. 1(c)). The pattern deformation process is efficiently executed on a GPU.

Human bodies are considered to have common deformation patterns, because they share the same musculoskeletal structure, even if the sex, age, height, or weight thereof differs. Once we have created good patterns, these can be applied to various human models. With the pre-created patterns, users of our system merely have to place the required pattern on the texture space and specify appropriate deformation parameters. Although a user needs to have basic knowledge of skin deformation such as knowing to which region of the human body each pattern can be applied, knowledge of the anatomy is not necessary. Our method is thus easy to use and is expected to be useful.

The dynamic height map has previously been used in research studies and applications [6]. The main contribution of this research is the categorization of skin deformations into three types (wrinkles, bulging and protruding bones) and the realization thereof using two different deformation models. Moreover, by patternizing the deformations, our method allows users to apply common patterns to a new human model and to make these look believable more easily and more quickly.

The rest of this paper is organized as follows. In Section 2, we review related works. In Section 3, we analyze human skin deformations and describe how our method realizes the deformations. Section 4 details the implementation of our method. Experimental results and a discussion are presented in Section 5. Finally, Section 6 concludes the paper.

2   Related Work

Methods for skin deformation can be categorized as either model-based or data-based methods.

Anatomy based physical simulation of skin deformation has been widely researched [14][15][19]. This model-based approach realizes very human-like skin deformation by modeling bones, tendons, muscles, and tissues as a real human body and by simulating the interactions of these with one another. In practice, however, such anatomy-based simulation is not widely employed since specifying anatomical information is tedious for creators and requires in-depth anatomical knowledge. Moreover, the simulation is generally computationally expensive, although it can be computed more efficiently by using pre-computed muscle deformations [19].

There are generic methods for simulating deformable objects that employ either mass-spring models [3] or finite element methods [17][2]. These simulations can be driven by a skeleton [2]. Although bulging deformations can be simulated in this way, wrinkles and protruding bones cannot.

In addition to the methods specific to skin deformations, wrinkle generation or deformation methods for cloth simulation can also be applied to skin deformation. These methods generate wrinkles on a coarse cloth mesh based on the compression of the mesh computed using physics-based simulation or geometric deformation. Oshita et al. [12] proposed a geometric method to add wrinkles to the edges of a cloth mesh based on the length variation of the edges. Larboulette et al. [8] described the generation of wrinkles on a cloth or skin mesh along the position and orientation specified by the designer. The curves of the wrinkles are controlled by the compression of the mesh. Li et al. [10] presented a method to generate wrinkles on a face based on the deformation of the keynodes and wrinkle lines specified by the user. Loviscach [11] developed a method to generate wrinkles over a cloth mesh using sinusoidal curves based on the wrinkle magnitude and direction computed for each vertex from the compression of the cloth mesh. These methods generate wrinkles based on the compression of the geometry. However, since human skin is more elastic than cloth, wrinkles cannot simply be computed from compression. Moreover, these methods cannot simulate bulging or protruding bones.

Various data-based methods have also been developed. Using sample data acquired from real human or physics simulations, realistic skin deformation can be represented without using any anatomical model or time consuming physics simulations. Sloan et al. [16] applied a shape blending method to scanned body shapes. Kry et al. [7] proposed the EigenSkin method to compute shape blending in real-time on a GPU using linear approximation. This method was applied to hand skin using sample
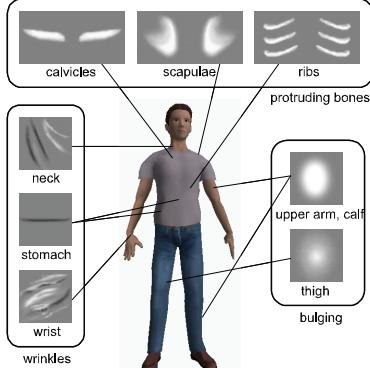
Fig. 2. Examples of the three types of deformation with height patterns for each. The patterns are dynamically altered according to the angle of the corresponding joint.
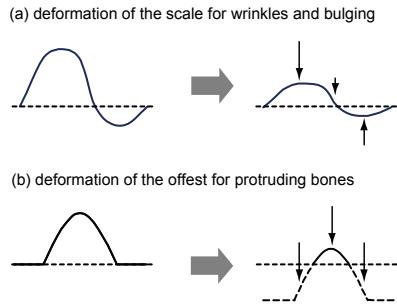
Fig. 3. Two deformation models for height patterns. (a) For wrinkles and bulging, the scale of the height pattern is controlled. (b) For protruding bones, the offset of the height pattern is controlled.

shapes computed from a physics based simulation. However, in order to employ these blending methods, high-quality sample shape data must be prepared.

Hapap et al. [6] proposed a method for generating cloth wrinkles using height patterns. The height patterns, drawn by a designer, are scaled based on the compression of the coarse cloth geometry. Cutler et al. [4] developed a wrinkle generation system for cloth animation, that uses curve-based wrinkle patterns created by a designer for each sample pose. These sample patterns are then blended according to a stress map, computed from the coarse cloth geometry, to generate the final wrinkles. Our method uses the height patterns created by a designer as input in a similar way to [6] and [4]. However, we deform the height patterns according to the joint angles, and not the compression of the geometry, because this is more suited to human skins. Moreover, our method simulates not only wrinkles, but also bulging muscles and protruding bones by changing the scales and offsets of the height patterns.

3   Skin Deformation

Based on the anatomy and observation of the human body [14][13], we categorize skin deformations into three types: wrinkles, bulging, and protruding bones (Fig. 1). Wrinkles are observed on the stomach, waist, wrist, neck, elbow, knee, etc. and are caused when the skin is compressed by bending or twisting an adjacent joint (Fig. 1(a)). Bulging occurs on the thigh, upper arm, etc. when muscles contract or tissue is compressed by bending an adjacent joint (Fig. 1(b)). Protruding bones are observed on the clavicles, ribs, scapulae, etc. and occur when bones are pressed against the skin by extending joints and the shape of the bones is seen through the skin (Fig. 1(c)). All these deformations are caused by moving a corresponding joint.

Our method simulates these deformations using dynamic height patterns. The height pattern for each deformation is created by a designer (Fig. 2). By changing the scaling or offset of each height pattern according to the angle of the corresponding joint, a height map is dynamically generated. For wrinkles or bulging (Fig. 3(a)), the scaling of the height pattern is changed to vary the height and depth of the wrinkles or bulging. In the case of protruding bones (Fig. 3(b)), the offset of the height pattern is changed to vary the area revealing bones, without changing the shape of the bones.

We use a height image for each pattern. Each pixel has a signed height value between -1.0 and 1.0. Each pattern is dynamically altered based on the angle around a rotational axis of the corresponding joint. For example, wrinkles on the stomach are caused by bending at the waist (that is, by moving the waist joint around the front-back rotational axis). Rotations around a difference axis of a particular joint cause different deformations. For example, twisting around the waist joint causes wrinkles on the sides of the waist. Since such deformations from different axes are considered to be independent, we can assign different patterns to each axis.

To date we have prepared 8 basic patterns as shown in Fig. 2. As explained in Section 1, these patterns are common to every human body, and can therefore, be applied to any human body by placing a pattern in the texture space and setting the appropriate deformation parameters. Users may also add their own custom patterns.

4   Implementation

An overview of our method is shown in Fig. 4. A conventional SSD (Skeletal Subspace Deformation) model [9] and pattern information are given to the system. The pattern information contains the positions, scales and orientations in the texture coordinates for each height pattern. It also contains deformation parameters. The pattern information is specified by the user via an easy-to-use interface.

Rendering of a human mode is done in two stages. In the first stage, the height map is generated by rendering each pattern on the specified region on the height map while changing the scale and offset of the patterns according to the joint angles of the character given by the animation system. This process is computed on a GPU using a
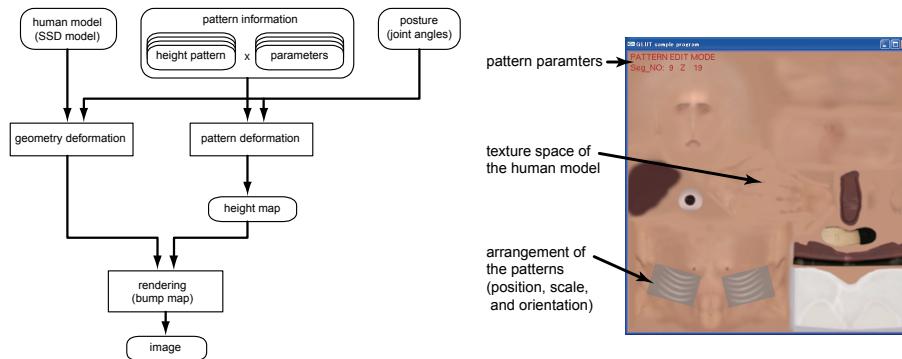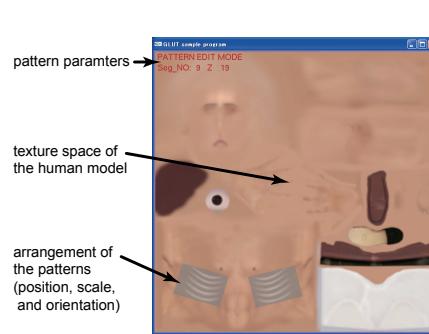

Fig. 4. System overview.

Fig. 5. User interface.

render-to-texture technique and programmable shaders. In the second stage, the SSD model is rendered with the height map using bump mapping and displacement mapping. The deformation of the SSD model is also computed efficiently on a GPU using vertex shaders [5]. Since all processes are computed on a GPU, our method executes very quickly. In addition, our system can easily be integrated with conventional rendering frameworks of existing animation systems or computer games.

For rendering the SSD model using a height map, displacement mapping is expected to generate better images than bump mapping. However, it is still a challenge to apply displacement mapping in real-time on the current graphics hardware and the shader model. Therefore, in our current implementation, we use both bump mapping and geometry deformation with two height maps. The height map for wrinkles and protruding bones is used for bump mapping, while the height map for bulging is used to move the vertices of the geometry of the human model in their normal direction. Since bulging affects the contour of the model, geometry deformation is considered necessary, whereas wrinkles and protruding bones are subtle and thus bump mapping is sufficient to represent these.

4.1 User Interface

Users of our system specify how each deformation is applied to a human figure model. We have developed a simple interface as shown in Fig. 5. Users can choose a pattern via the keyboard and change the position, scale, and orientation of the pattern in the texture space with the mouse. They can also specify the necessary deformation parameters, as explained in Section 4.2, using the keyboard. The result of the deformation can be reviewed by switching the view from the texture space to a 3D space showing the rendered model with animation. The user repeats this process for each deformation pattern. Typically, it takes between ten to twenty minutes to specify all patterns on a new human model.

Users can add their own custom patterns to the system. These patterns are created using existing digital painting software such as Adobe Photoshop or Corel Painter.

4.2 Deformation Model

As explained in Section 4, each pattern is rendered on the specified region on the height map. The output height of each pixel is computed based on the height pattern using the following equations. Recall that we use two different models for the different types of skin deformations.

For wrinkle and bulging patterns, as shown in Fig. 3(a), the height of the corresponding pixel of the pattern is scaled based on the angle of the specified joint,

$$f = \frac{\theta - \theta_{min}}{\theta_{max} - \theta_{min}} + 1, \ L_{out} = \begin{cases} 0 & if \ \theta \leq \theta_{min} \\ L \cdot h & if \ \theta \geq \theta_{max} \\ L \cdot h \cdot \log(f) & otherwise \end{cases} , \quad (1)$$

where L is the height value from the corresponding pixel of the pattern and θ is the joint angle. θ_{max} is the joint angle at which the height reaches its maximum

value (the value of the ordinal pattern), while θ_{min} is the joint angle at which the height is zero. Since wrinkles and bulging are non-linear, we use a logarithmic function. h is the parameter that controls the overall scale of the pattern and is specified by the user. This parameter is used to make the patterns more or less noticeable for individual figure models and body parts (the default value is 1.0). The deformation parameters $\theta_{max}, \theta_{min}, h$ are changed by the user for each pattern.

For protruding bones, as shown in Fig. 3(b), the height of the corresponding pixel of the pattern is biased according to the angle of the specified joint.

$$g = \frac{\theta - \theta_{max}}{\theta_{min} - \theta_{max}} + 1, \quad L_{out} = \begin{cases} 0 & \text{if } \theta \leq \theta_{min} \\ L \cdot h & \text{if } \theta \geq \theta_{max} , \\ \max\left[h(L - \log(g)), 0\right] & \text{otherwise} \end{cases} \quad (2)$$

where $\max[a,b]$ returns the higher value of a or b. As for bones, the shape of the bones does not change even if the joints are moved; what changes is how much of the bones is noticeable. Therefore, in our model, the offset of the height is biased according to the angle of the specified joint.

5   Experiments and Discussion

We have tested our method with a human model and the deformation patterns shown in Fig. 2. The human model consists of 4340 triangles and a texture image. Its skeleton has 50 bone segments. We have developed a program for the experiments using Visual Studio 2005 (C++), OpenGL, and Cg [5] for GPU programming. The program has been tested on a standard PC (Pentium 4 3.2 GHz, 1 GB Memory, GeForce 6800GT) with Windows XP Pro.

The time taken to render one human model with 15 patterns is 11 milliseconds with a frame rate of 150 fps when all processes are executed on a GPU using render-to-texture and programmable shaders. However, the speed drops to 19 milliseconds and 48 fps when the height map generation is executed on a CPU. These experiments show that our method achieves real-time performance by utilizing a GPU.

Examples of generated animations are shown in Fig. 1 and 6. Additional generated animations are available on the authors' website (*http://www.cg.ces.kyutech.ac.jp*). Included in the animations is a comparison with static patterns (standard bump map), showing that our method generates more plausible animations by dynamically changing the height patterns. Although we currently use a simple shading model, the skin shading can be improved by using a more advanced model [1]. This is beyond the scope of this paper, since the aim of our method is to improve the deformation of skin and as such shading is not dealt with at present.

As explained in Section 4.2, we use a simply approximated deformation model that does not reflect the physics. Although we have tried more complex models, the differences were not noticeable because skin deformations are very subtle. On the contrary, the differences between our model and the linear deformation model were indeed noticeable; as were the differences between the scaling deformation for
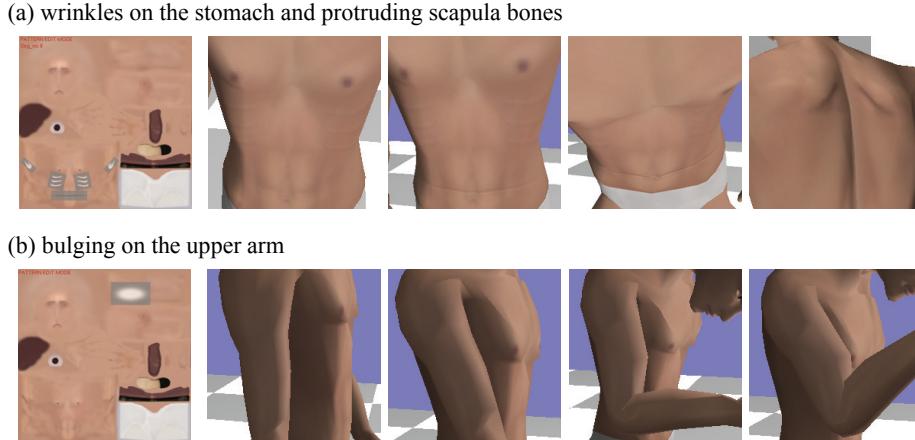
(a) wrinkles on the stomach and protruding scapula bones

(b) bulging on the upper arm

Fig. 6. Images from generated animations.

wrinkles and bulging and the offset deformation for bones. We believe that the complexity and quality of our model suits our purpose admirably.

A weakness of our model is that at present it cannot handle deformation from external forces (e.g., from contact with other characters or objects) or gravity. In order to handle such deformation, physics of some kind needs to be introduced. We believe that simplified physics such as that used in [2] can easily be integrated into our method since our method relies only on height maps and does not change the geometry which can be deformed by using a physics-based method.

In the latest generation of computer games, characters have been given their own height maps to improve the quality of the rendering with a relatively small number of polygons. Although the human model in our experiments does not have such a height map, our method can be applied to character models with height maps by adding the original height map and the dynamic height patterns on the fly.

6   Conclusion

In this paper, we proposed an artist-oriented pattern-based real-time skin deformation technique. Experiments show that our method works in real-time and produces plausible animations. Users of our system are able to make a character model look more human-like merely by specifying how predefined patterns are applied to the model. We believe that our method is very practical. Our future work includes improving the deformation model and integrating it with a simplified physics simulation. Improving the default deformation patterns is also seen as future work.

References

1. Curtis Beeson, Kevin Bjorke, Skin in the "Dawn" demo, GPU Gems, pp. 45-72, Addison-Wesley, 2004.
2. Steve Capell, Seth Green, Brian Curless, Tom Duchamp, Zoran Popovic, Interactive Skeleton-Driven Dynamic Deformations. ACM Transactions on Graphics (Proc. of SIGGRAPH 2002), Vol. 21, Issue 3, pp. 586-593, 2002.
3. John E. Chadwick, David R. Haumann, Richard E. Parent. Layered construction for deformable animated characters. Proc. of SIGGRAPH 1989. Vol.23, No. 3, 1989.
4. Lawrence D. Cutler, Reid Gershbein, Xiaohuan Corina Wang, Cassidy Curtis, Erwan Maigret, and Luca Prasso. An art-directed wrinkle system for CG character clothing. Proc. of ACM SIGGRAPH Symposium on Computer Animation, pp. 117-126, 2005.
5. Randima Fernando, Mark J. Kilgard. The Cg tutorial: the definitive guide to programmable real-time graphics. Addison-Wesley, 2003.
6. Sunil Hadap, Endre Bangerter, Pascal Volino, and Nadia Magnenat-Thalmann. Animating wrinkles on clothes. Proc. of the conference on Visualization '99, pp. 175-182, 1999.
7. Paul G. Kry, Doug L. James, and Dinesh K. Pai. EigenSkin: real time large deformation character skinning in hardware. Proc. of ACM SIGGRAPH Symposium on Computer Animation 2002, pp. 153-160, 2002.
8. Calorine Larboulette, Marie-Paule Cani. Real-time dynamic wrinkles. Proc. of Computer Graphics International 2004, pp. 522-525, 2004.
9. J. P. Lewis, Matt Cordner, Nickson Fong. Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. Proc. of SIGGRAPH 2000, pp. 165-172, 2000.
10. Ming Li, BaoCai Yin, DeHui Kong, XiaoNan Luo. Modeling expressive wrinkles of face for animation. Proc. of Fourth International Conference on Image and Graphics 2007. pp. 874-879, 2007.
11. J. Loviscach. Wrinkling coarse meshes on the GPU. Computer Graphics Forum (Proc. of EUROGRAPHICS 2006), Vol. 20, Issue 3, pp. 467-476, 2006.
12. Masaki Oshita, Akifumi Makinouchi. Real-time cloth simulation with sparse particles and curved faces. Proc. of Computer Animation 2001, pp. 220-227, 2001.
13. Akihiko Sakuragi, Miyuki Takeda. Anatomy for CG Creators (in Japanese). Bone Digital Inc., 2002.
14. F. Scheepers, R. E. Parent, W. E. Carlson, S. F. May. Anatomy based modeling of the human musculature. Proc. of SIGGRAPH 97, pp. 163-172, 1997.
15. M. Simmons, J. Wilhelms, A. V. Gelder. Model-based reconstruction for creature animation, Proc. of ACM SIGGRAPH Symposium on Computer Animation 2002, pp. 139-146, 2002.
16. Peter-Pike J. Sloan, Charles F. Rose III, Michael F. Cohen. Shape by example. Proc. of Symposium on Interactive 3D Graphics 2001, pp. 135-143, 2001.
17. Demetri Terzopoulos, John Platt, Alan Barr, Kurt Fleischer. Elastically deformable models. Proc. of SIGGRAPH 1987, pp. 205-214, 1987.
18. J. Wilhelms, A. V. Gelder. Anatomically based modeling, Proc. of SIGGRAPH 1997, pp. 173-180, 1997.
19. Xiaosong Yang, Jian Chang, and Jian J. Zhang. Animating the human muscle structure. Computing in Science & Engineering, Vol. 9, Issue 5, pp. 39-45, 2007.