Animating Wrinkles on Clothes

Sunil Hadap   Endre Bangerter   Pascal Volino   Nadia Magnenat-Thalmann

MIRALab, CUI
University of Geneva, Switzerland
email: {sunil, endre, pascal, thalmann}@cui.unige.ch

Abstract

This paper describes a method to simulate realistic wrinkles on clothes without fine mesh and large computational overheads. Cloth has very little in-plane deformations, as most of the deformations come from buckling. This can be looked at as area conservation property of cloth. The area conservation formulation of the method modulates the user defined wrinkle pattern, based on deformation of individual triangle. The methodology facilitates use of small in-plane deformation stiffnesses and a coarse mesh for the numerical simulation, this makes cloth simulation fast and robust. Moreover, the ability to design wrinkles (even on generalized deformable models) makes this method versatile for synthetic image generation. The method inspired from cloth wrinkling problem, being geometric in nature, can be extended to other wrinkling phenomena.

CR Categories: I.3.3 [Computer Graphics]: Picture/Image Generation—Picture/Image Generation; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation;

Keywords: cloth modeling, wrinkle modeling, deformable models

1 INTRODUCTION

Wrinkles add life to garments in fashion. Figure 1 illustrates how wrinkles are important for visual realism. In this section we outline the problem and the motivation behind the work.

In order to capture realistic wrinkles on a real-life garment, from a mere geometric point of view, the number of triangles required can be easily upto a hundred thousand. Such a large number of triangles put cloth simulation off from interactive speeds, even with adaptive time steps, introduced recently [1]. Apart from simulation time, the large triangle count increases the rendering time and the cost significantly. In order to avoid these, one can increase fineness of triangles only in the potential regions where wrinkling might occur. This is very well possible due to advances in the triangulation and interactive systems developed [2, 7, 8, 13]. Even then, a significant problem remains: how to estimate the regions and the orientations of wrinkles.

Cloth has very large in-plane deformation stiffnesses compared to its ability to bend and shear. This gives rise to very stiff equations of motion. The problem of solving stiff equations is successfully dealt with by the use of an implicit method for numerical integration by Baraff *et al*[1]. Here, though the problem of stiff equations has been tackled, it has been the strong motivation for the authors behind developing the methodology specifically for wrinkles.

Even if one wishes to have a fine triangular mesh, using robust and fast numerical solvers and having patience for long computations, it is not guaranteed that the wrinkles will be satisfactory. Accurate and fast collision detection methods[12], constraint methods[5, 6] and good deformable models[6, 9, 10] have proved to give quality cloth animation. However, real-life wrinkling is a complex phenomenon. It is characterized by frictional forces (especially between body and cloth) which are difficult to model. Collision response methods and friction models developed so far have been rather simple for such a complex problem and robust numerics too.

Figure 1: Wrinkles in fashion

We take a geometric and texture based approach to wrinkling. As it is difficult to physically simulate real life wrinkles, the user designs them interactively as a bump map on a coarse mesh cloth/garment. It is then animated by modulating it as per cloth deformation. The key theme is conservation of cloth area. The work is continuation of earlier work [11]. Other attempts to model wrinkles include those by Gotoda *et al* [3, 4] and Wu *et al* [14].

The following section dwells on issues of animating cloth using a coarse mesh and describes how a geometric wrinkling method can be developed based on area conservation. Section 3 introduces the algorithm, leaving mathematical details of the formulation to Appendix A. Section 4 describes implementation issues of the algorithm and extends it for practical use. Section 5 gives a case study of a typical animation using the algorithm. Finally a conclusion and future work are given.

2 ANIMATING CLOTH USING COARSE MESH

We would like to animate the cloth using coarse triangular mesh (typically a few thousand triangles per garment), for the reasons mentioned in the Introduction. However, Figure 2 depicts how the

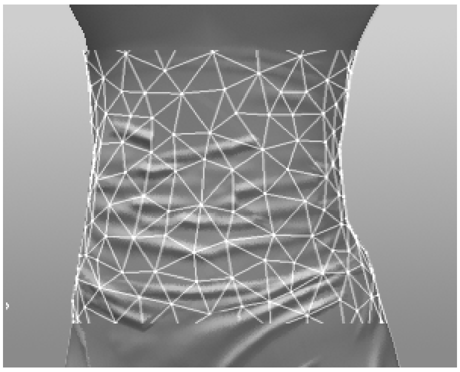

Figure 2: Wrinkles and coarse mesh

geometry of coarse mesh would be unable to capture fine wrinkles on cloth.

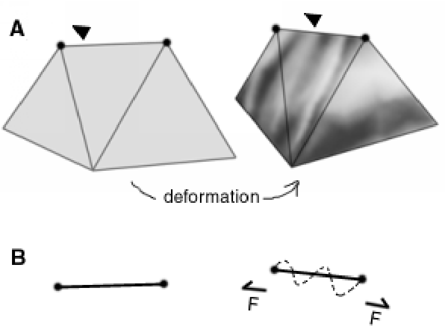

Figure 3: Large triangle deformations due to buckling

Real cloth has very little in-plane deformation as most of the deformations come from buckling. For the coarse mesh, setting high metric (in-plane deformation) stiffnesses will not work properly. For the vivid picture of the situation, refer to the triangles undergoing deformations in Figure 3A. Real cloth would wrinkle to this deformation (see typical wrinkles in Figure 3A). Consider an edge of a triangle, as shown in Figure 3B. In reality, the compression forces will buckle the edge as shown by dotted line. As the bending stiffness of the cloth is small, the buckled edge exerts small forces on the vertices. However, in the coarse mesh situation, the buckled edge is approximated by a straight line between the vertices. Consequently, the real life buckling is attributed to the compression of the edge. If we assume a high metric stiffness associated to this compression, the corresponding forces on the vertices will be high. This is in contrast with real cloth situation. Thus, to animate the cloth realistically with a coarse mesh, we need to set small metric stiffnesses. This allows otherwise possible wrinkling/buckling which is embedded in the deformation of triangle.

Very little in-plane deformations can be looked at as area conservation property of cloth. Note the change in area of the triangles due to the deformations in Figure 3 in case of large triangle deformation. We propose to capture gross cloth movements and deformations using a coarse mesh and the fine deformations (wrinkles) using a bump map (or a displacement map). Let us assume the wrinkle pattern is given by the user. We will try to modulate the amplitude of the wrinkle pattern such that, though there is a change in the area of a triangle (with the displacement map), it is invariant after applying the modulated displacement map. Compare the edge shown in Figure 3 to that in Figure 4 after the application of the proposed algorithm. In the next section, we derive the area conservation formulation. The method is inspired by the area conservation property, even though Section 3.3 points out that the empiricism introduced later does not actually conserve the area.

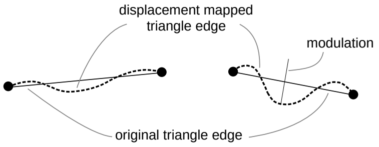

Figure 4: Edge with modulated wrinkle pattern

3 GEOMETRIC WRINKLES

3.1 Overview

In this subsection, we would like to give a heuristic overview of our *geometric wrinkling* algorithm and the context in which it is embeded. We also introduce the key terms that will be used throughout the paper, whereas their accurate definitions will be given in section 3.2. Please refer to Figure 5, which illustrates this overview.

First, let us state what serves as an input to the algorithm. We start with a user defined *wrinkle pattern*, which is given in the form of a texture and an initial undeformed triangular mesh in 3D space. This mesh may represent a garment or another deformable model. The wrinkle pattern is bump or displacement mapped onto the initial mesh by the user. Thus, we obtain a static wrinkled garment. Note that the texture mapping coordinates do not change throughout the computations described below.

By introducing a fixed scale for the displacement or bump map on a mesh triangle, we obtain a function which, we call the *wrinkle function*. Using these inputs in step 1 (refer to Figure 5), the algorithm computes a set of four parameters termed *wrinkling coefficients* for each triangle of the mesh.

The initial mesh serves as an input to a simulation engine, which in the context of cloth simulation would be the physical model with a numerical solver. The mesh that is the output by the simulation engine will be the deformed mesh.

This deformed mesh is then further processed by the proposed algorithm. For each triangle, we compute the *deformation transformation* that relates the corresponding triangle of the initial and the deformed mesh. Using this deformation transformation and the already computed wrinkling coefficients, we compute the *modulation factor*s.

These modulation factors are used to compute a *modulation map* which modulates the wrinkle pattern. The modulated wrinkle pattern, which reflects the response of the wrinkled surface to the deformation of the underlying coarse triangular mesh, is used for the rendering.

In the course of animation, as the simulation engine recalculates the deformed mesh, the procedure described above is iterated. However, note that the wrinkling coefficients need not to be recalculated during the animation. They are constant with respect to the animation process. They depend only on the initial wrinkle pattern, the initial mesh and the mapping coordinates.

3.2 Derivation of the Wrinkling Algorithm

This subsection introduces the mathematical formulation of our geometric wrinkling algorithm.

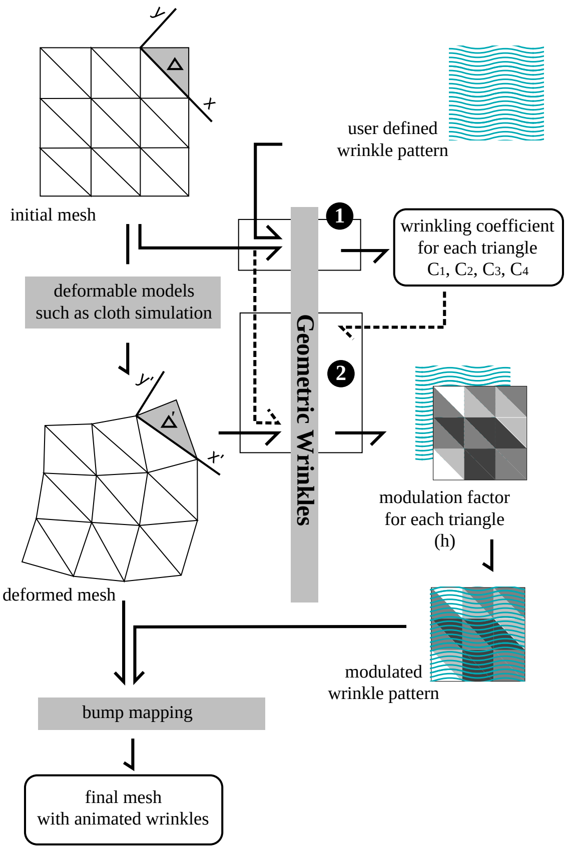

Figure 5: Wrinkles and coarse mesh

Refer to Figure 5 and consider the grayed triangle $\triangle$ undergoing a deformation ending up as the deformed triangle $\triangle'$. The deformation of the triangle can be described by a general 4D homogeneous coordinate transformation. However, the rotational and translational parts of the transformation are irrelevant to the derivation of the algorithm. We introduce a local rectangular right handed two dimensional coordinate system, which is defined by choosing any edge of the triangle as the x axis (refer to Figure 5). Hence, x, y denote the local coordinates of the initial triangle $\triangle$ and x', y' that of the deformed triangle $\triangle'$.

They are related by the *deformation transformation*

$$\begin{aligned} T : \begin{bmatrix} x' \\ y' \end{bmatrix} &= \begin{bmatrix} a & b \\ 0 & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \\ &\equiv \begin{bmatrix} T_1(x,y) \\ T_2(x,y) \end{bmatrix} \end{aligned} \qquad (1)$$

The matrix elements a and d represent scaling in the x and y direction respectively, whereas b describes a shear. Let us also introduce the inverse deformation transformation and denote it by

$$\begin{aligned} T^{-1} : \begin{bmatrix} x \\ y \end{bmatrix} &= \begin{bmatrix} a' & b' \\ 0 & d' \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix} \\ &\equiv \begin{bmatrix} T_1^{-1}(x',y') \\ T_2^{-1}(x',y') \end{bmatrix} \end{aligned} \qquad (2)$$

We define the *wrinkle function* $f(x, y)$ as the function in the coordinate system xy that results from mapping the wrinkle pattern onto the initial triangular mesh. Formally, we write

$$f(x,y) : \triangle \in R^2 \rightarrow R \qquad (3)$$

Further we require $f(x, y)$ to be continuous and that its first partial derivatives exist and also be continuous.

The wrinkle function $f'(x', y')$ on the triangle $\triangle'$ is related to $f(x, y)$ by

$$\begin{aligned} f'(x',y') \quad &: \quad \triangle' \in R^2 \mapsto R \\ &= \quad f\left(T_1^{-1}(x',y'), T_2^{-1}(x',y')\right) \end{aligned}$$

where

$$\triangle' = T(\triangle)$$

Now let us introduce the area $A(\triangle, f(x, y))$ of the wrinkle function $f(x, y)$ over the triangle $\triangle$ and in complete analogy the area of the deformed wrinkle function $A'(\triangle', f'(x', y'))$ over the deformed triangle $\triangle'$.

Generally, these two areas will be unequal

$$A(\triangle, f(x,y)) \neq A'(\triangle', f'(x',y'))$$

and therefore the overall wrinkled surface area over the whole mesh will not be conserved during the animation

However as pointed out in Section 2, real life examples such as in cloth wrinkling, the surface area is conserved. Note that this is a *purely geometric* requirement. One might think of several different approaches to meet this requirement.

Our approach realizes overall area conservation by achieving area conservation on a per triangle basis of the mesh. We parameterize the area of the deformed triangle by h, that scales $f'(x', y')$ on each triangle of the mesh.

Formally, this parameterized surface area is given by

$$A'(h, \triangle', f'(x',y')) \equiv A'(\triangle', hf'(x',y'))$$

Using this notation the requirement of area conservation on a per triangle basis is expressed by the following equation

$$A(\triangle, f(x,y)) = A'(h, \triangle', f'(x',y')) \qquad (4)$$

This is an equation for h, which we call *modulation factor*. In Appendix A, we present an approximate solution to this equation that yields

$$h = 1 - \frac{1}{C_4}(C_1(a'-1) + C_2 b' + C_3(d'-1)) \qquad (5)$$

The constants C_1, C_2, C_3, C_4 are here after referred as *wrinkling coefficients*. The following section gives significance of the formulation.

3.3 Explanations

Following rather abstract and mathematical derivation in the previous subsection, we would like to point out and discuss some properties of the algorithm we have presented so far:

- The algorithm is computationally inexpensive during animation, as the only quantities that have to be determined for each triangle are the elements a', b', d' of the inverse deformation matrix and the resulting modulation factor by using the formula given by equation 5. The wrinkling coefficients on the other hand, are computationally expensive. However, as one can see from equation 11 in Appendix A, they depend only on quantities that are known prior to entering the animation loop and can therefore be calculated once at the beginning.

- As we have pointed out in the discussion so far, we have derived an algorithm that is based on the area conservation property. Here we explain the role played by the area conservation property in our work. From a mathematical point of view, it is clear that we have presented a solution within the approximation of small deformations. There are several possibilities to deal with this restriction. One could decide to restrict the simulations to small deformations where the approximation is valid and/or take into account the higher order terms in the series expansion to extend the range of validity of the approximation.

  Instead, we propose a pragmatic approach. We *declare* that equation 5 describes the overall wrinkling behavior *only*. This frees us conceptually from the "burden of mathematical correctness". This is because, we are more concerned with the visual results of the animation, rather than precise area conservation. Moreover, the deformations during cloth simulation are moderate in general. Hence, higher order terms in the expansion may become significant but not predominant.

  The modulation factor h is a function of the deformation of triangle and has value around one. If the triangle is net compressed, h will be greater than one. For the elongation, it will be less than one. One can scale, translate and clip it to introduce a finer control required for the animation. *Note that this transformation of the modulation factor no longer satisfies the area conservation property. Also, Section 4 introduces a modulation map, which further violates area conservation.*

- Another very important property of our algorithm is that it is *local*. By local, we mean that wrinkling effects caused by deformations are confined to the deformed areas. This is crucial to obtain realistic wrinkling. For example, a garment wrinkles around the shoulder of an animated character as she lifts her arm, while it is stretched on the corresponding side. Locality is introduced in our algorithm by working on a per triangle basis. The size of the mesh triangles actually governs the extension of local wrinkling effects.

- Wrinkling coefficients are sensitive to the wrinkle function and therefore to the wrinkle patterns. Wrinkling coefficients for two different patterns on the same triangle will generally differ. Therefore, the same deformation applied to a triangle will yield two different modulation factors (one for each pattern).

  Each pattern, for instance, features a "principal wrinkling direction". Assume that the wrinkling patterns are orthogonal to each other. Then, a deformation in the orthogonal direction of one pattern will result in a smaller modulation factor as compared to a modulation factor for the other pattern. In other words, the direction of the deformation "favors one pattern over the other". This property can be used for developing multi-fold wrinkling techniques. Section 4.1 describes a multi-fold wrinkling implementation.

4 IMPLEMENTATION OF GEOMETRIC WRINKLES

The numerical computation of the formulation is trivial. For the numerical integrals of the wrinkling coefficients, we use adaptive sampling in the triangular domain to give a fixed user defined error. The following issues are worth mentioning about the implementation.

1. The wrinkle function and the wrinkle pattern, though referred to as the same entity, they differ in implementation. The wrinkle pattern is gray scale texture image defining the bump map. The user defines an overall normalization factor for the texture to map wrinkle pattern to wrinkle function. The normalization factor is important as the formulation assumes real distances for the bump map (or more precisely the displacement map). The factor should be some fraction of the overall dimensions of the average triangle of the mesh. The wrinkle function is a continuous real valued function, which is a spline approximation of the normalized texture as described in next item.

2. The wrinkling coefficient computation involves partial derivatives of wrinkle function $f(x, y)$ with respect to (x, y). For the reasonable numerical accuracy and stability, the wrinkling pattern needs to be smooth. We fit a spline function to the pattern to smooth out any discontinuities in the input. In addition to this, the user is advised to blur the pattern.

3. As stated in the formulation (Appendix A), solution to equation 4 exists if the input pattern is not constant. As the pattern is user defined, one needs to watch for the invalidity of the solution (constant C_4 in equation 5 turn out to be zero) and therefore eliminates it. In this case, we define the modulation factor to be one.

4. The modulation factor varies significantly across triangles. If we treat a constant modulation factor for a triangle (see Figure 6), wrinkles appear patchy and one can distinctly see the triangular granules. To avoid this, the modulation factors are linearly interpolated across triangles to give smooth *Modulation Map* (Figure 6).

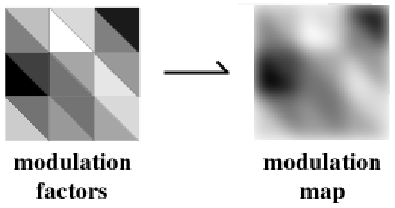

Figure 6: Smoothing modulation factors gives modulation map

5. The user is given additional control for the animation by transforming the modulation map by a scale factor, clip, and bias. The final bump/displacement map is the product of the modulation map and the wrinkling pattern.

The implementation of multi-fold wrinkling is discussed in detail in the next section.

4.1 Multi-fold Wrinkling

Animating a single wrinkle pattern is not satisfactory (particularly for cloth). In real-life, the wrinkles are not mere modulations of a fixed wrinkle pattern. Rather, the pattern changes according to the

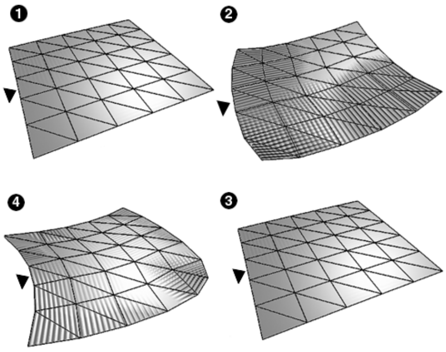

Figure 7: Simple two-fold wrinkling

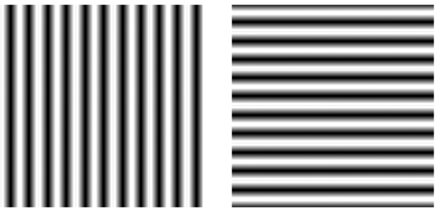

Figure 8: Orthogonal patterns for two-fold wrinkling

deformation. Hence, we would like to apply the technique using multiple wrinkle patterns.

As stated in Section 3.3, two different wrinkle patterns give different wrinkling coefficients for the same triangle geometry. Hence, for the same deformation of the triangle, corresponding to each pattern, the modulation factors will be different. It all depends on how the wrinkle pattern is oriented with respect to the deformation direction. If a pattern is orthogonal to the deformation direction (as compared to the other), corresponding modulation factor will be small. In other words, the direction of the deformation favors one pattern over the other.

To illustrate this, let us consider simple cloth animation as shown in the Figure 7. In Stage 1 cloth is undeformed. It is then stretched to the bottom left corner (Stage 2). Comes back to the neutral position (Stage 3) and finally in Stage 4, stretches to the bottom right corner.

We would like to apply multiple wrinkle patterns for this animation. For simplicity of the discussion, we consider only two wrinkle patterns, though the methodology is developed for multiple patterns. The wrinkle patterns chosen are orthogonal to each other as shown in Figure 8.

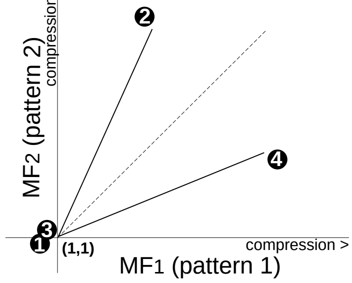

Figure 9: Movement of wrinkling coefficients

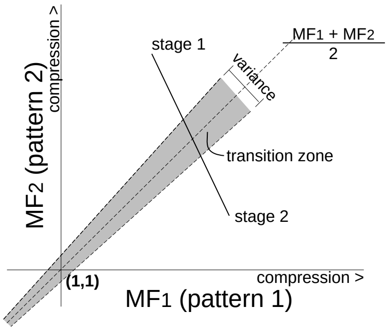

Figure 10: Transition of the patterns

As the marked triangle undergoes a series of deformations (Figure 7, Stages 1-4), it may compute different values for the modulation factor for each of the wrinkle patterns (Figure 8). These two modulation factors are then plotted against each other in Figure 9. For Stages 1 & 3 both the modulation factors are 1 as cloth is undeformed. However, for Stages 2 & 4 the modulation factors differ significantly, depending upon the direction of the deformation. The relatively small modulation factor (say MF_1 is smaller for Stage 2) indicates that the corresponding wrinkle pattern is well oriented towards the direction of the deformation. We choose this pattern for wrinkling for the deformation.

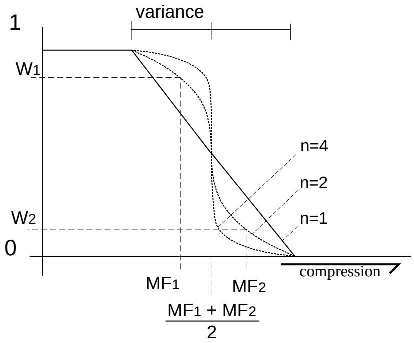

Figure 11: Smooth switch between two coefficients

This selective application of the wrinkle pattern (along with its modulation) will give a change of one pattern to the other as the deformation direction changes. However, in the animation a sudden switch of the pattern is not temporally coherent and is visually quite disturbing. Figure 10 shows typical the movement of the wrinkle functions in the situation (the movement does not refer to the animation in Figure 7). The dotted line (when two modulation factors are equal) indicates the boundary between the patterns. As the modulation factors cross the dotted line, there is a sudden switch in the pattern.

To avoid this sudden switch of pattern, we introduce a user definable variance around the mean value of the wrinkling coefficients, which defines a transition zone. This is indicated by a gray area around dotted line in Figure 10. There will be a smooth transition between wrinkling patterns in this zone. We employ a wrinkling pattern weight function as shown in Figure 11 to achieve the smooth transition. This is in fact a simple power function with an appropriate scaling and clipping.

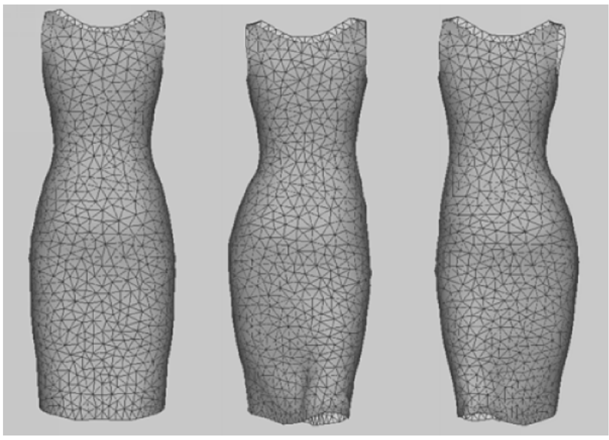

Figure 12: Animated garment without wrinkles

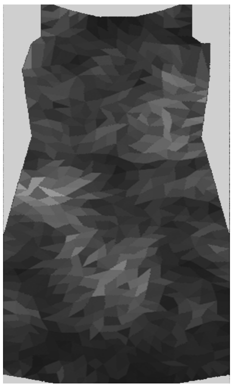

Figure 13: Visualizing strains in the garment

$$pattern_{modulated} = (MF_1 W_1 pattern_1 + MF_2 W_2 pattern_2)$$

If MF_1 is much smaller than MF_2 (stage 1 in Figure 7), MF_1 will be smaller than $(1 - variance)(MF_1 + MF_2)/2$ and MF_2 will be bigger than $(1 + variance)(MF_1 + MF_2)/2$. This gives maximum weight ($W_1 = 1$,$W_2 = 0$) to pattern 1. In the transition zone, when MF_1 and MF_2 are comparable, the two patterns will be blended smoothly. The user definable power n is representative of the tightness of the transition and $n = \infty$ is a sudden switch of pattern.

Figure 7 shows the two patterns animated according to deformation. Note that for lower left triangles in Stage 2 of the animation, both wrinkle patterns get blended. On the other hand, for lower right triangle in Stage 2, the deformation direction favors one pattern clearly.

5 ANIMATING GARMENT WRINKLES

Until now, the *Geometric Wrinkle* formulation is developed keeping in mind a general deformable model. In this section, we apply the method specifically for cloth animation. Figure 12 shows representative frames of a typical garment animation without *Geometric Wrinkle*. The garment is animated using a coarse mesh and low metric stiffnesses for the reasons explained in section 2.

Though the user can design the wrinkles according to her wish, it is worthwhile to study the strain patterns in the garment. This is because, inappropriately placed wrinkles in the region where there is no deformation will not animate satisfactorily. In addition, the pattern should be orthogonal to the deformations in general, as explained in section 4.1. Figure 13 is a strain pattern for the second frame in the Figure 12. Dark triangles are triangles with compression and depict the regions where wrinkles might appear.

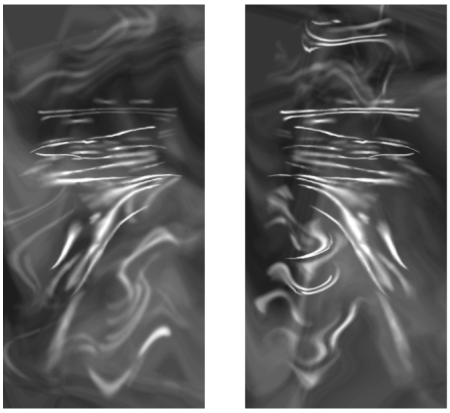

Figure 14: Wrinkling patterns for the garment

Based on such strain patterns (corresponding to two distinct frames of the garment animation in Figure 12), two wrinkling patterns are designed as shown in the Figure 14. The patterns have distinct wrinkles and additional irregularities to smooth out the sharp appearance of wrinkles. Each pattern represents a distinct direction of deformation. Note that they are considerably orthogonal.

Figure 17 (color plate) shows the result after applying the *Geometric Wrinkles*. It is interesting to see the smooth switch of the wrinkling patterns in the animation because of multi-fold wrinkling.

Figure 15 and Figure 16 (color plates) use a single wrinkle pattern (different from Figure 14). The frames on the left side correspond to the animation without *Geometric Wrinkles*. Note that, there are very few wrinkles in the second figure as there is very little deformation of the mesh. The first figure shows the modulation of the wrinkles as per the deformation.

As the calculations of the wrinkling coefficients are done on a per triangle basis, the computational time is linear with respect to number of triangles. Typically, it takes 5 minutes per thousand triangles on a MIPS R10000 200 MHz processor. Once the wrinkling coefficients are computed, the time spent on modulating wrinkle pattern is negligible compared to rendering time. In fact, for small meshes (upto a hundred polygons) the modulation of wrinkle pattern can be real time (20 fps).

6 CONCLUSION AND FUTURE WORK

We have developed a fast and versatile method for animating realistic wrinkles, which is geometric in nature. Hence, it can be applied to general deformable models such as cloth. We would like to extend the method by automatically creating wrinkle patterns from the strain pattern, which is currently a time consuming task.

7 ACKNOWLEDGMENT

This work is supported by the Swiss National Research Foundation. The reviewers comments were very helpful and encouraging. Many thanks to Chris Joslin for proof reading this document.

References

[1] David Baraff and Andrew Witkin. Large Steps in Cloth Simulation. In Michael Cohen, editor, *SIGGRAPH 98 Conference Proceedings*, Annual Conference Series, pages 43–54. ACM SIGGRAPH, Addison Wesley, July 1998. ISBN 0-89791-999-8.

[2] Martin Courshesnes, Pascal Volino, and Nadia Magnenat Thalmann. Versatile and Efficient Techniques for Simulating Cloth and other Deformable Objects. In Robert Cook, editor, *SIGGRAPH 95 Conference Proceedings*, Annual Conference Series, pages 137–144. ACM SIGGRAPH, Addison Wesley, August 1995. held in Los Angeles, California, 06-11 August 1995.

[3] Tosiyasu L. Kunii and Hironobu Gotoda. Modeling and Animation of Garment Wrinkle Formation Processes. In N. Magnenat-Thalmann and D. Thalmann, editors, *Computer Animation '90 (Second workshop on Computer Animation)*, pages 131–147. Springer-Verlag, April 1990.

[4] Tosiyasu L. Kunii and Hironobu Gotoda. Singularity Theoretical Modeling and Animation of Garment Wrinkle Formation Processes. *The Visual Computer*, 6(6):326–336, December 1990.

[5] Jen-Duo Liu, Ming-Tat Ko, and Ruei-Chuan Chang. Collision Avoidance in Cloth Animation. *The Visual Computer*, 12(5):234–243, 1996. ISSN 0178-2789.

[6] John C. Platt and Alan H. Barr. Constraint Methods for Flexible Models. In John Dill, editor, *Computer Graphics (SIGGRAPH '88 Proceedings)*, volume 22, pages 279–288, August 1988.

[7] Jim Ruppert. A Delaunay Refinement Algorithm for Quality 2-Dimensional Mesh Generation. *Journal of Algorithms*, 1995.

[8] Jonathan Richard Shewchuk. Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator. In Dinesh Manocha Ming C. Lin, editor, *Applied Computational Geometry: Towards Geometric Engineering*, volume 1148, pages 203–222. The First ACM Workshop on Applied Computational Geometry, Springer-Verlag, May 1996.

[9] Demetri Terzopoulos and Kurt Fleischer. Modeling Inelastic Deformation: Viscoelasticity, Plasticity, Fracture. *Computer Graphics (SIGGRAPH '88 Proceedings)*, 22(4):269–278, August 1988. Held in Atlanta, Georgia.

[10] Demetri Terzopoulos and Kurt Fleischer. Deformable Models. *The Visual Computer*, 4(6):306–331, December 1988.

[11] Pascal Volino and Nadia Magnenat-Thalmann. Fast Geometric Wrinkles on Animated Surfaces. In *WSCG'99*, Plzen, Feb 1999.

[12] Pascal Volino and Nadia Magnenat Thalmann. Efficient Self-Collision Detection on Smoothly Discretized Surface Animations using Geometrical Shape Regularity. *Computer Graphics Forum*, 13(3):155–166, 1994. Eurographics '94 Conference issue.

[13] Pascal Volino, Nadia Magnenat Thalmann, Shen Jianhua, and Daniel Thalmann. An Evolving System for Simulating Clothes on Virtual Actors. *IEEE Computer Graphics and Applications*, 16(5):42–51, September 1996 1996. ISSN 0272-1716.

[14] Yin Wu, Prem Kalra, Laurent Moccozet, and Nadia Magnenat-Thalmann. Simulating Wrinkles and Skin Aging. *The Visual Computer*, 15(4):183–198, 1999. ISSN 0178-2789.

A FORMULATION DETAILS

In order to present a solution to equation 4 we first give the explicit expressions for $A(\triangle, f(x,y))$ and $A'(h, \triangle', f'(x',y'))$ respectively

$$A(\triangle, f(x,y)) = \iint\limits_{\triangle} \sqrt{1 + \left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}\, dx\, dy \qquad (6)$$

and

$$A'(h, \triangle', f'(x',y')) = \iint\limits_{\triangle'} \sqrt{1 + h^2 \left(\frac{\partial f'}{\partial x'}\right)^2 + h^2 \left(\frac{\partial f'}{\partial y'}\right)^2}\, dx' dy' \qquad (7)$$

To evaluate the integral in equation 7, we perform a change of integration variables as follows

$$\begin{bmatrix} x \\ y \end{bmatrix} = T^{-1} \begin{bmatrix} x' \\ y' \end{bmatrix} \qquad (8)$$

By applying this change of variables to equation 7 (refer to equation 2 for a', b', d'), we obtain

$$\begin{aligned} & A'(h, \triangle', f'(x',y')) \\ & \quad = \iint\limits_{\triangle} \left(1 + h^2 \left(a' \frac{\partial}{\partial x} f\right)^2 \right. \\ & \quad \left. + h^2 \left(\left(b' \frac{\partial}{\partial x} + d' \frac{\partial}{\partial y}\right) f\right)^2 \right)^{1/2} \frac{1}{a'd'}\, dx\, dy \\ & \quad \equiv \iint\limits_{\triangle} I(h, a', b', d')\, dx\, dy \end{aligned} \qquad (9)$$

Now let us perform a series expansion of equation 9 in the transformation parameters and the modulation factor. As far as the transformation parameters are concerned, we choose as expansion point the identity transformation and for the modulation factor we expand around $h = 1$

$$\begin{aligned} A'(h, \triangle', f'(x',y')) = \iint\limits_{\triangle} & (I(h{=}1, a' = 1, b'{=}0, d'{=}1) \\ & + \frac{\partial I}{\partial a'}(a'{-}1) + \frac{\partial I}{\partial b'} b' \\ & + \frac{\partial I}{\partial d'}(d'{-}1) + \frac{\partial I}{\partial h}(h - 1) \\ & + \ldots)\, dx\, dy \end{aligned} \qquad (10)$$

In the following, we discard higher order terms in the expansion above. For small deformations around the identity transformation and $h = 1$, a first order expansion represents a good approximation for the value of the surface area over a deformed triangle.

Retaining terms up to first order in the series expansion and by plugging expression equation 10 into equation 4 we obtain

$$\begin{aligned} A(\triangle, f(x,y)) = C_0 &+ C_1(a'–1) + C_2 b' \\ &+ C_3(d'–1) + C_4(h–1) \\ where\ C_0 &= \iint\limits_{\triangle} I(1,1,0,1)\, dx\, dy \\ C_1 &= \iint\limits_{\triangle} \frac{\partial I}{\partial a'}\, dx\, dy \\ C_2 &= \iint\limits_{\triangle} \frac{\partial I}{\partial b'}\, dx\, dy \\ C_3 &= \iint\limits_{\triangle} \frac{\partial I}{\partial d'}\, dx\, dy \\ C_4 &= \iint\limits_{\triangle} \frac{\partial I}{\partial h}\, dx\, dy \end{aligned} \tag{11}$$

We call these expansion coefficients *wrinkling coefficients*. C_1, C_2, C_3, C_4 relate changes in the parameters a', b', d', h to changes of the area of the wrinkle function on the triangle.

By observing that $C_0 = A(\triangle, f(x,y))$ and solving equation 11 for the modulation factor we obtain

$$h = 1–\frac{1}{C_4}(C_1(a'–1) + C_2 b' + C_3(d'–1)) \tag{12}$$

Equation 12 represents the solution to equation 4 in the first order approximation to $A'(h, \triangle', f'(x', y'))$, if $f'(x', y')$ is not constant.

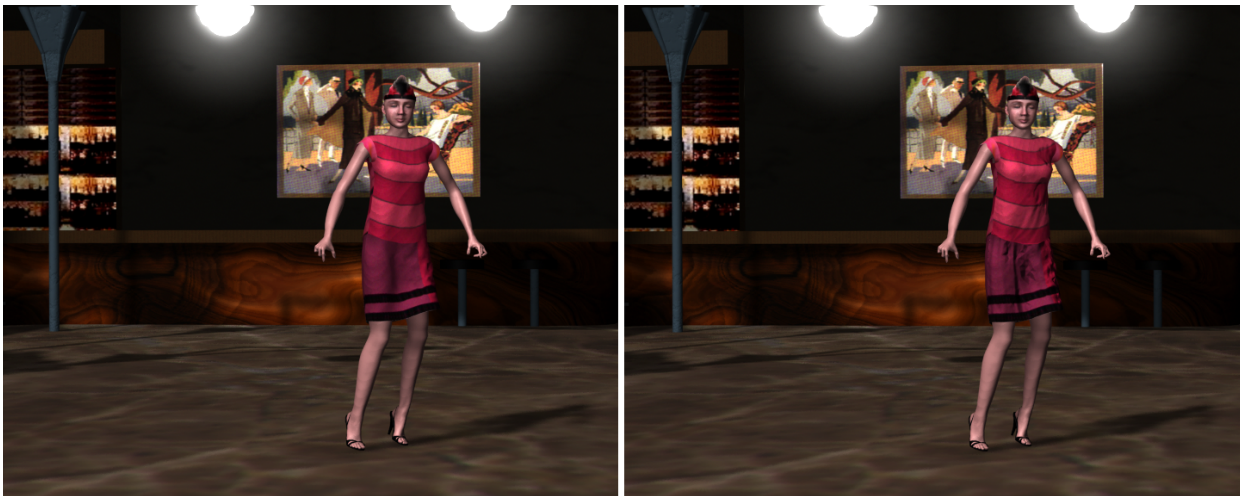

Figure 15: Garment animation without (left) and with (right) Geometric Wrinkles, frame 1, blouse - 686 triangles, skirt - 1274 triangles

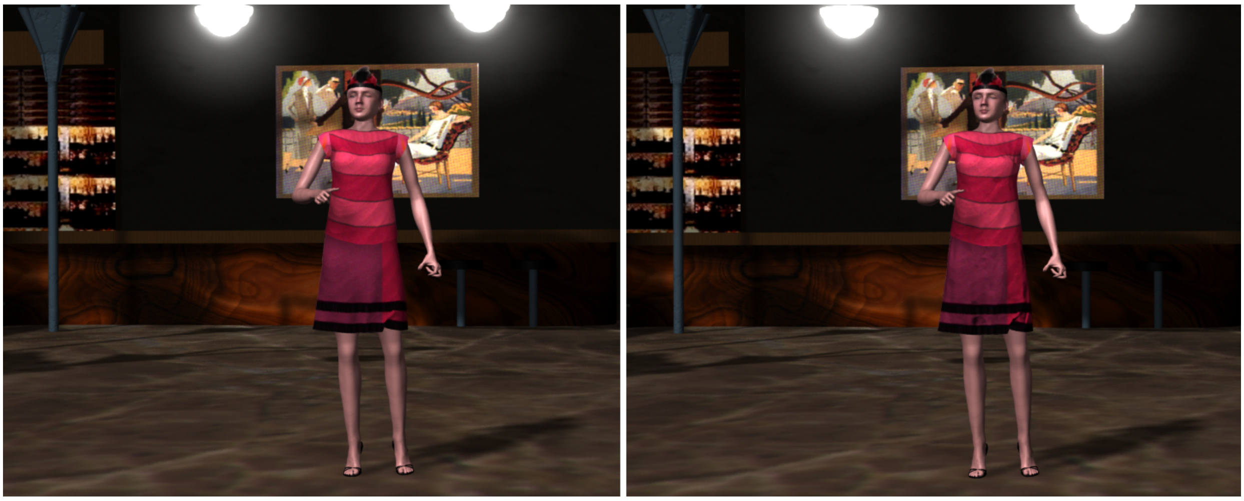

Figure 16: Garment animation without (left) and with (right) Geometric Wrinkles, frame 128. Observe very few Geometric Wrinkles because of low mesh deformations.

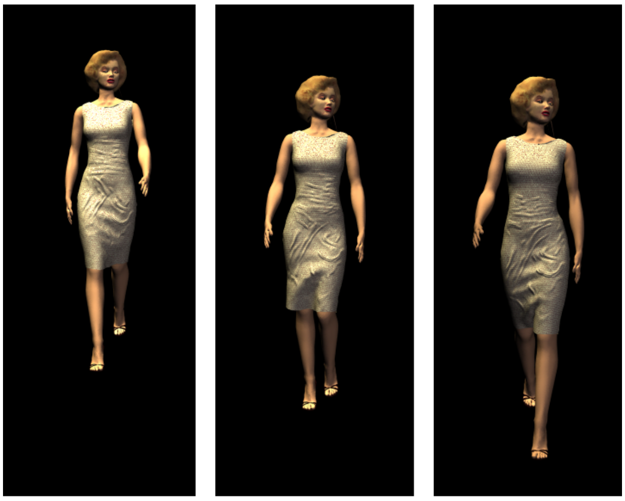

Figure 17: Multi-fold Wrinkling, frames [53, 67, 81], 25fps. Observe the pattern switch with blending.