Use of multiple representations for simulating cloth shapes and motions: An overview

by T. Ohta

Cloth is relatively difficult to simulate for computer graphics, primarily because of the complexity of its shapes and motions. Because of the presence of numerous folds and wrinkles of various sizes, conventional means cannot be used to measure or recreate its shapes. This overview describes a method in which a physically based simulation technique is adapted for generating cloth shapes and motions, and presents some examples of images produced by the method. Use is made of a structure composed of multiple representations or "layers," each of which is assigned a different role in calculation: typically, surface rendering, dynamics, collision detection, and stitching. Inclusion of the latter makes it possible to simulate the joining together of several cloth sections, thus making the method applicable to geometrically complicated tasks such as clothing design. Deformations of cloth objects can be calculated by solving associated equations of motion in a time-marching manner, thereby also producing serial data for animation.

Introduction

Advances in computer graphics (CG) now make it possible to render highly realistic images of rigid objects. The extent of the realism depends on the CG rendering technique used and on the detail and precision of the data that describe the objects to be rendered. There has been considerable motivation for extending that capability to objects constructed of cloth. However, because cloth is easily deformed and can assume very complicated shapes with many folds and wrinkles, that has been difficult to achieve.

Although the highly realistic images and associated data for rigid objects can be obtained by the use of computer-aided design (CAD) or applicable mathematical equations, until the past few years that has not been possible for cloth objects. Weil [1] has specified geometric constraints in order to generate images of cloth objects, and others have adapted physical simulation techniques for that purpose [2–6]. The former approach is not suitable for simulating cloth motion and tends to be limited to a specific application area, but it has the advantage of being able to generate a static shape quickly. The latter approach can generate consecutive data, and can be applied to a much wider range of application areas, but it requires a considerable amount of computational time.

0018-8646/95/$3.00 © 1995 IBM

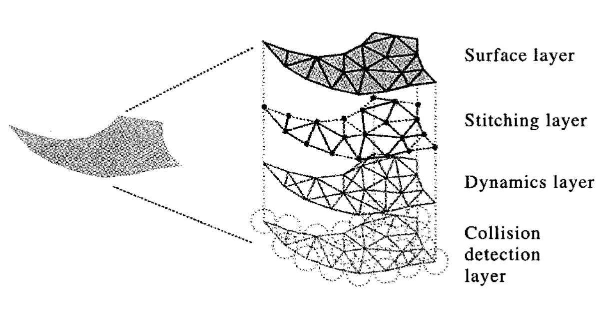


Figure 1

Schematic of typical layers used in the method described in this paper.

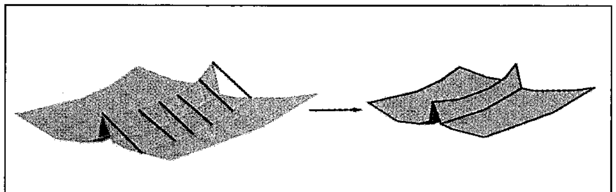


Figure 2

Example of a folded shape.

Whatever approach is used, an appropriate grid system is required for representing the cloth object. Although straightforward for a cloth object having a simple rectangular shape, the major application in the simulation of cloth—the simulation of clothing—usually involves the use of more complex shapes. The generation of grid systems for such shapes is generally difficult, especially if use is to be made of a structured grid system. Furthermore, the simulation of clothing involves the use of shapes with three-dimensional topology, thus requiring the use of three-dimensional rather than two-dimensional modelings; three-dimensional topology must be created from two-dimensional structures. This requires the incorporation of a function that performs a series of operations similar to stitching together sections of cloth.

In this paper, we provide an overview of a method for simulating cloth shapes and motions that is based on a simple connected-mass structure composed of multiple representations or "layers," each of which is assigned a different role in calculation: typically, surface rendering, dynamics, collision detection, and stitching. Inclusion of the stitching function makes it possible to simulate the joining together of several cloth sections, thus making the method applicable to geometrically complicated tasks such as clothing design. Deformations under external forces can be calculated by solving associated equations of motion in a time-marching manner. The relative flexibility in combination of functions makes the method applicable to materials other than cloth.

Our approach

Previously proposed methods for simulating a cloth object make use of a specific grid system, whose characteristics define the object rather rigidly. This sometimes makes it difficult to generate a grid, especially when the grid is structured. Instead of a single layer, the approach described here makes use of several layers, each of which is assigned to a specific function, as shown in the schematic in **Figure 1**. The surface is formed by a set of triangles, and the dynamics of the associated model are considered in terms of a structure of mass points and connections that have specific physical properties. Information on stitching is assigned to one of the layers. Another layer, for collision detection, consists of spheres placed so that they cover the entire surface. Interpenetration of spheres indicates the occurrence of collisions. Mass points can be connected arbitrarily, and consequently grid generation becomes easier. The layers are related to each other with respect to the grid structure, and thus none can be selected completely independently of the others. However, the relationships of the individual layers to each other can also be arbitrary. This flexibility reduces the burden of constructing models and provides several advantages. Each layer can have a structure suited to its own role, regardless of the other functions. Furthermore, by separating the surface, which is the part that contributes to the appearance, one can add extra properties to the model without violating the shapes to be rendered. For example, consider the folded shape shown in **Figure 2**. The connections linking the second-nearest points (red lines) help to sustain the folded shape, but do not contribute to the formation of the surface. One aspect in which cloth models have not succeeded in simulating natural cloth behavior is that wrinkles cannot be preserved when a cloth is spread out again after being folded and deformed several times. This can be achieved if a structure such as the one described above is applied to the entire region. The example shows a model with four layers, but these could be arranged arbitrarily. Special effects can be added by superimposing another layer for a particular function. In the simplest case, a model can be constructed using only two layers, representing surface and dynamics aspects, or, more generally, by combining any number of layers with specific roles.

Dynamics of the model

In simulating cloth deformation, the equation of motion for the model is solved numerically in a time-marching manner. To calculate the dynamic behavior of the cloth, we consider a structure of mass points and connections. Equations must be formulated for every mass point.

The equations are derived as a Lagrangian expression by estimating all of the required energies for the model. The choice of energies, i.e., the choice of physical system, defines the characteristics of the model.

The Lagrangian equation is of the form

$$\frac{d}{dt}\frac{\partial L}{\partial r} - \frac{\partial L}{\partial r} = F(r, t), \tag{1}$$

where L is a Lagrangian, r is a position vector, and F is an external term. A dissipation term is added to the equation for stability when the equation is discretized. Differential terms are discretized by the finite difference method, and the equations are solved numerically for all of the mass points in a time-marching manner.

The simplest substitution of connections is assumed to have the characteristics of a spring, leading to the energy equation

$$E_s = \sum_{n,m} \{k_{n,m}(|r_n - r_m| - l_{n,m})^2, \tag{2}$$

where E_s is the potential energy of the spring, $k_{n,m}$ is its spring constant, and $l_{n,m}$ is the original distance between the mass points at n and m; r_n and r_m are the position vectors of the mass points at n and m, respectively. The use of other potential functions (for example, exponentials) instead of the above function would produce a model having different characteristics.

The characteristic response to a bending force may be found by considering the following degree of curvature:

$$E_c = \sum_{n} C_n \left|\frac{\partial^2 r_n}{\partial u^2}\right|^2, \tag{3}$$

where E_c is the energy created by bending, C_n is the coefficient of hardness, and u is a coordinate in a calculating space.

Choosing different parameters for these energies can create different behaviors and different shapes, even in the static state. **Figure 3** shows examples of the differences that can occur. The shapes were calculated under the same conditions except for a difference in some parameters, assuming that the cloth was suspended at three corners.

Finally, external forces, such as gravity and wind or air friction, can be introduced into the model as external terms of Equation (1).

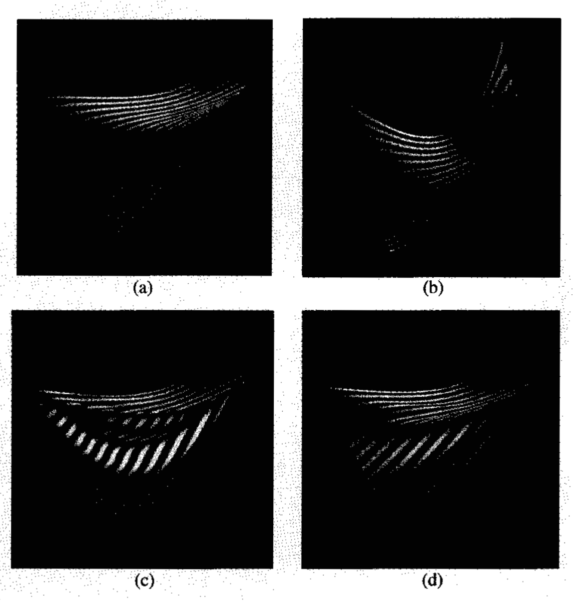


Figure 3

Examples of shapes that can be generated by using different parameters in Equations (2) and (3): (a) initial shape; (b) shape suggesting more stretchability; (c) shape suggesting more flexibility; and (d) shape suggesting more flexibility in one direction.

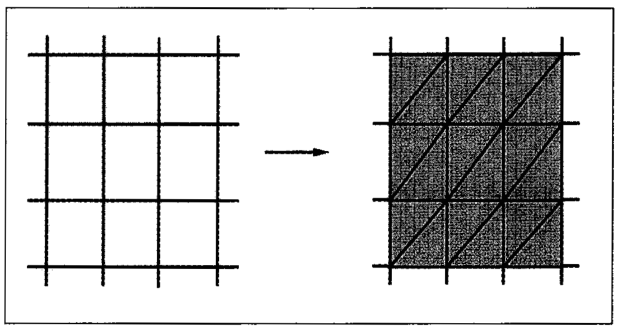


Figure 4

Cartesian and triangular representations of a section of cloth.

Surfaces

Generally, the grid selected has the same shape as the surface layer, which is rendered in order to obtain the final result. In the simplest example, which is a rectangular cloth section represented by a simple Cartesian coordinate system, the easiest way to define a surface is by using triangles, two of which occupy each grid cell, as shown in **Figure 4**. The sides of each triangle coincide with the grid

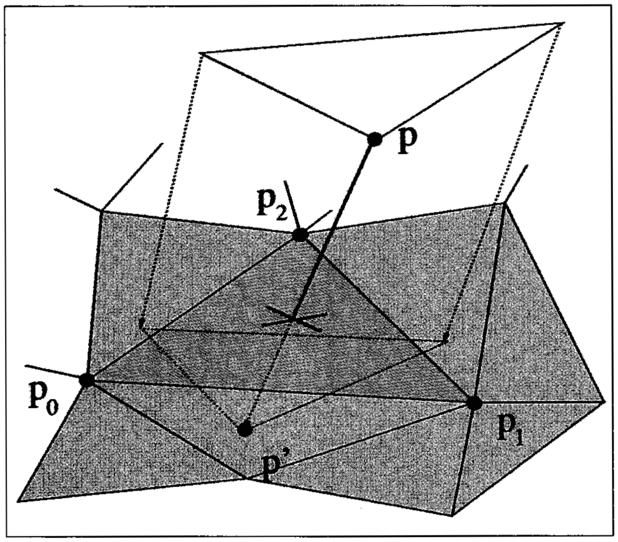


Figure 5

Illustration of a method for detecting collisions.

lines. In other cases as well, to simplify matters, a surface is replaced by a set of triangles, and the edges are taken to be identical with the connection lines, although there are no restrictions on how a triangle should be placed with respect to the connections.

Returning to the example in Figure 2, triangles are not placed on the grid for all of the connections. Thus, a layer for a surface can be placed rather freely, though not completely arbitrarily. If this concept is applied more broadly, a model that has a three-dimensional structure, with surfaces only on its outermost region, can be treated by using the same scheme. Two applications in which such an approach would be viable are a model of human skin and a model for seat design. In these cases, a finer set of connections for the outer surface could be attached to a coarser one of a three-dimensional structure for an inner material.

When initial patterns are designed two-dimensionally, they are given texture coordinates. This is useful when the model is applied to clothing.

Collision detection

Collision detection is the most time-consuming process in the method described here, assuming that it is carried out in detail. Not only must cases be considered in which a cloth penetrates other objects, but since cloth is a very soft material, the model could be greatly deformed and penetrate another part of itself. To avoid such unnatural behavior, all of the triangles forming a surface must be checked with all of the other triangles to determine whether they interpenetrate. Moreover, since all of the triangles move simultaneously as time passes, the checking process is not as simple as in the case of two static triangles. Once a collision has been detected, an appropriate action must be taken in order to create realistic motion.

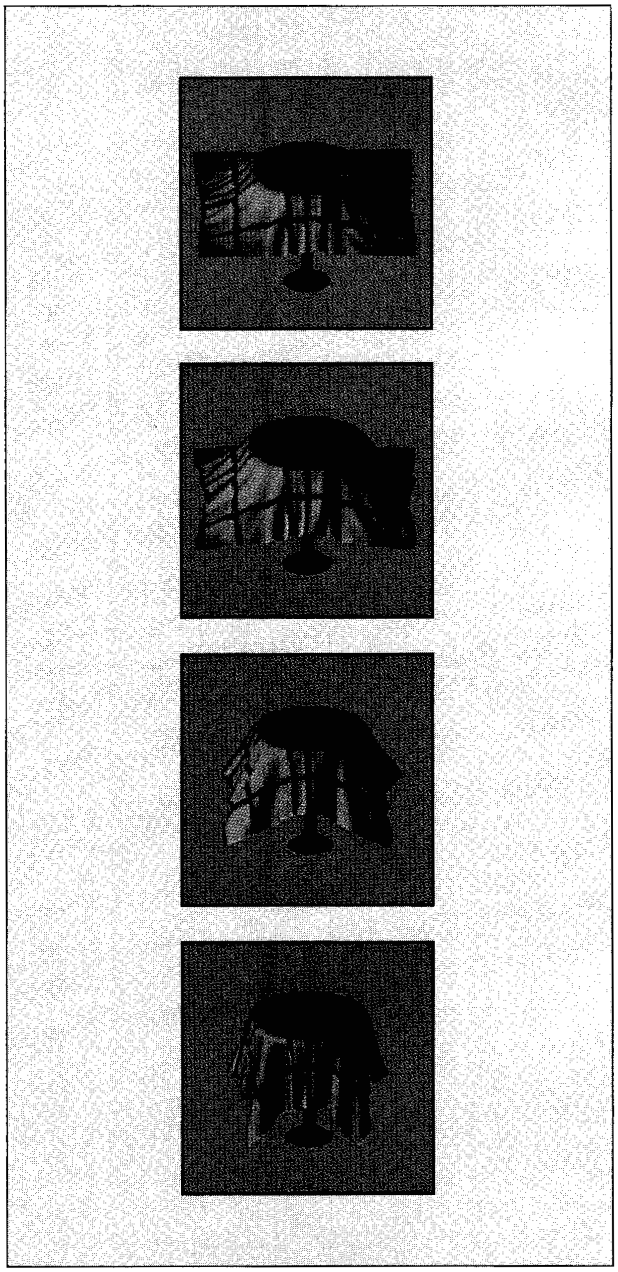


Figure 6

Examples of conformation of cloth to a table.

Collisions can be detected by checking for the penetration of triangles by points, as shown in **Figure 5**. Collisions are detected by solving the following equation [7]:

$$P + (P' - P)t = P_0 + (P_1 - P_0)u + (P_2 - P_0)v, \qquad (4)$$

where P and P' are the positions of a point at consecutive time intervals, and P_0, P_1, and P_2 are the positions of the vertices of a triangle.

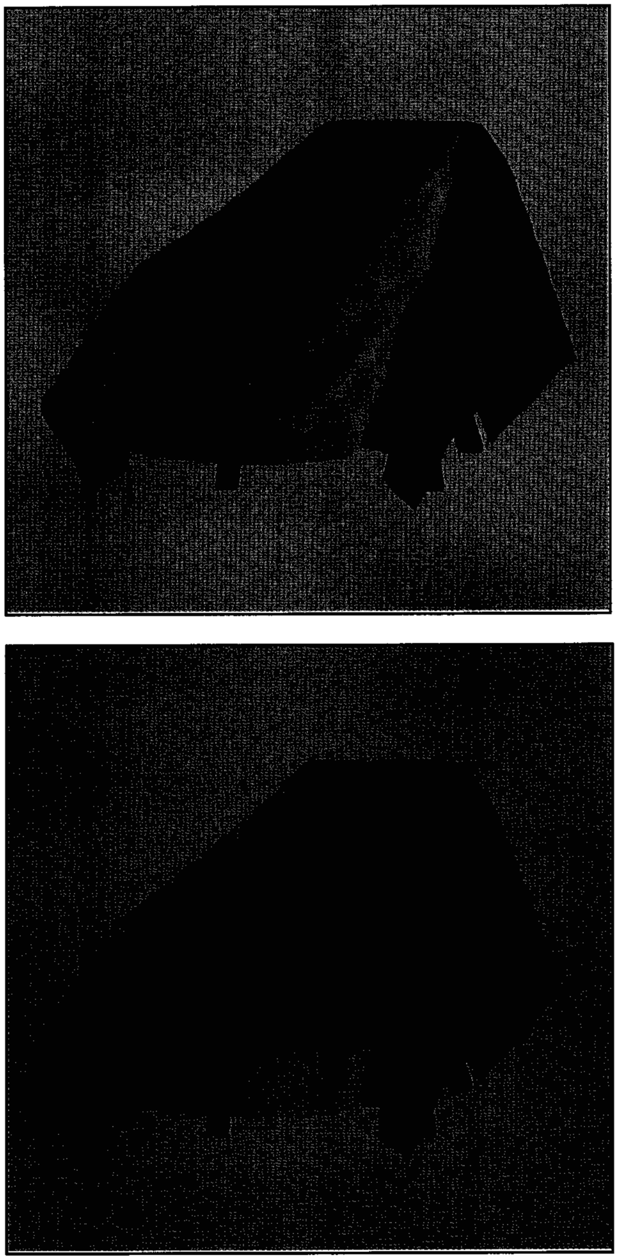


Figure 7

Examples of conformation of cloth to a chair.

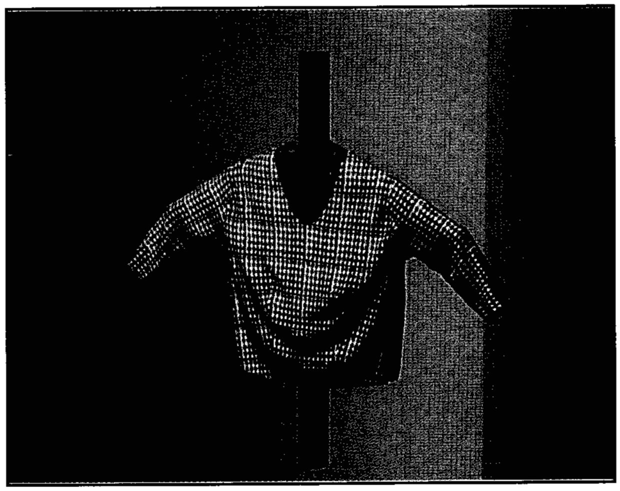


Figure 8

Example of the rendition of clothing.

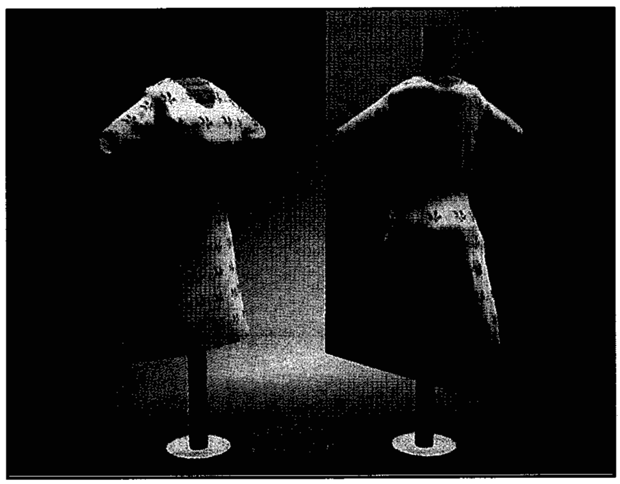


Figure 9

Additional examples of the rendition of clothing.

This method is not entirely sufficient, because the triangles are also in motion during a time interval; however, a vast amount of computation would be required if collisions were to be detected in detail.

A simpler means for detecting collisions is to make use of a collision detection layer consisting of several spheres that cover the surface, as shown in Figure 1. A collision between two spheres can be detected easily. If this method is used, a gap exists between two surfaces involved in a

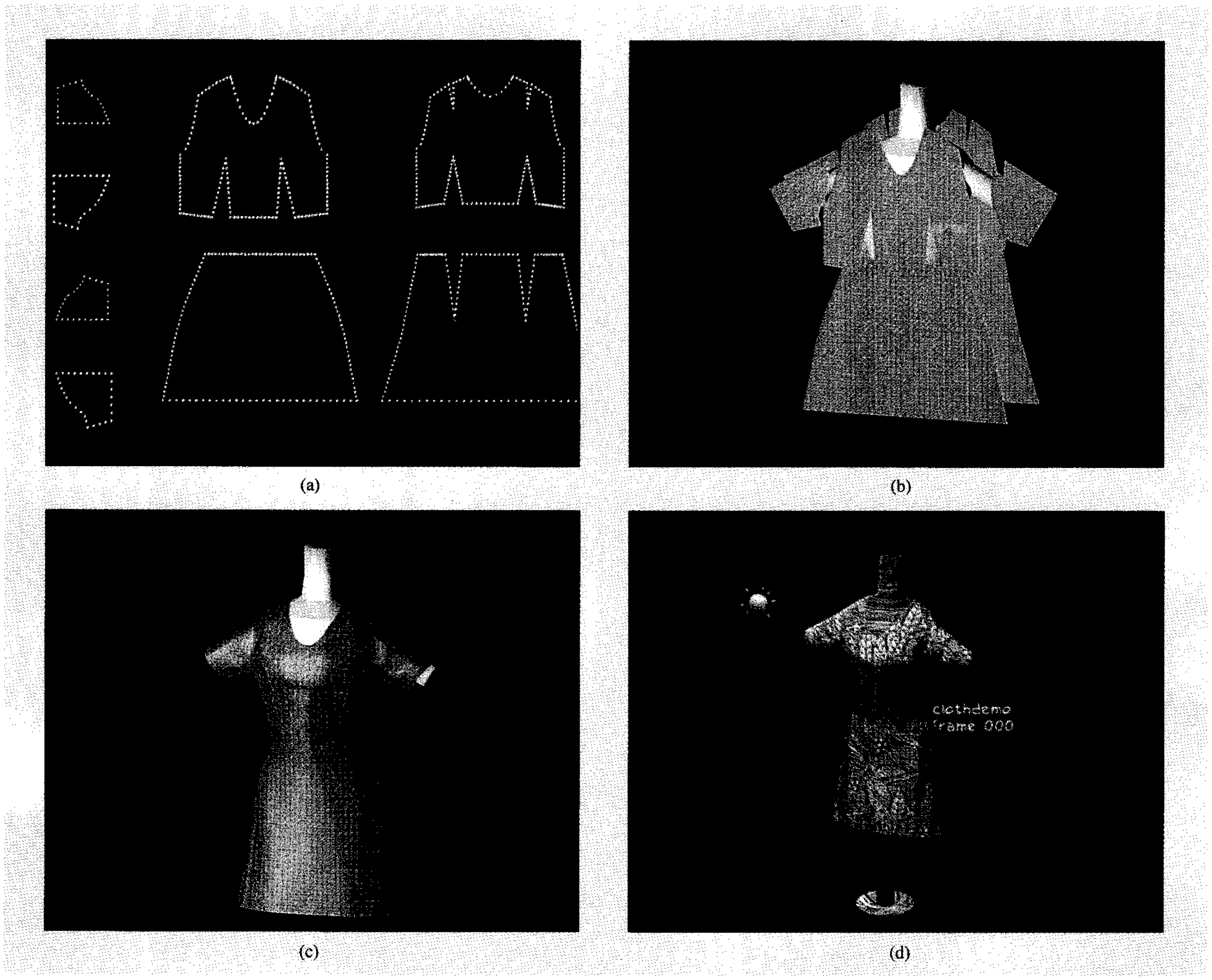


Figure 10

Sequence of steps used to obtain the renditions in Figure 9: (a) cloth sections; (b) after placement on a human-like object; (c) after stitching; and (d) after texturing.

collision, because of the radii of the spheres. However, if each triangle is small enough—for example, if the radius of each sphere is less than a hundredth of the length of one edge of the rectangular grid—the result is acceptable.

Considering that the model presented here is primarily for use in computer graphics, the simpler and faster the process, the better, even if it is not very exact. Corresponding measures in CAD systems require much more precise analysis.

Deformation process

Once the entire structure of the model has been determined, the deformation of the cloth can be calculated by taking account of the following specific conditions:

- Initial placements of the model and other objects.
- External forces acting on the model.
- Regions to be stitched together with other sections (if any).

Generally, in the type of simulation discussed here, a section of cloth is regarded as an object to be placed on some other object or objects, and is not isolated in empty space. Therefore, other objects, modeled by polygons, must be introduced as constraints. Initially, one or more sections of cloth and/or objects are placed appropriately in a three-dimensional space in such a way that the desired result can be obtained. If the cloth object is simply a tablecloth, the cloth is placed just above a table, so that it can be deformed as it covers the table. The deformation process is shown in **Figure 6.**

The simulation of clothing is more complicated. Since it is difficult to design an initial clothing model three-dimensionally, it would be convenient if several sections could be stitched together after being designed in two-dimensional coordinates. Accordingly, separate sections are placed around a model of a human body, to simulate the way in which a piece of clothing would actually be worn upon completion of the stitching process. Before the process begins, the regions to be stitched together must be defined. After the calculation has begun, each region is brought toward its paired region, until the two regions meet and are combined. A region is not restricted to being stitched together with a single other region, but may be stitched together with any number of regions.

Simulation of the stitching requires much more time than would a procedural or mathematical method if the aim were to obtain just one static shape. However, since wrinkles are created through the associated deformation, more natural cloth shapes can thus be obtained.

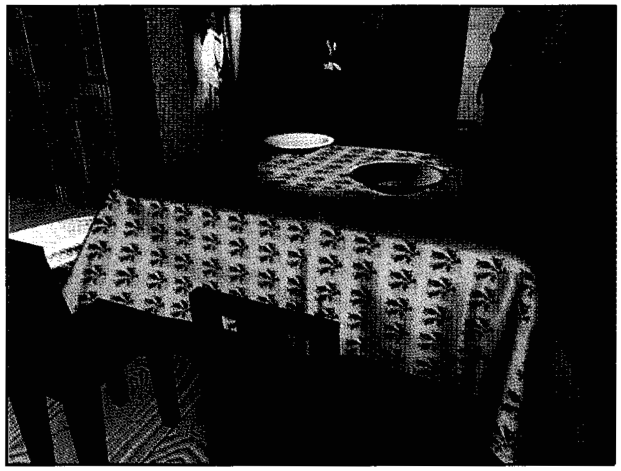
Examples

Figures 6–11 show examples obtained by using the approach discussed in the previous sections.

The examples in Figure 6 and **Figure 7** show that wrinkles are formed naturally. As we mentioned in an earlier section, these examples were obtained as the final result of a deforming process.

Figures 8 and **9** show two examples of clothing formed by stitching together six and eight sections, respectively. Some of these sections include a region that is to be stitched together not only with other sections, but also with another region of itself. A stitching process actually connects a point in a region to another point in a different region. Multiple points can be assigned to a single point in order to stitch together several cloth sections. The examples in the figures contain instances in which four points are connected. In the required calculation, the number of grid points in an entire section is about 3000, while the human-body-like object consists of 200 triangles. The time required to obtain the final result using an IBM RISC System/6000® Type 550 processor was about three hours.

Figure 10 depicts the sequence of steps used to obtain the examples in Figure 9. First, an object formed by polygons, with which a cloth is to interact, is placed in a scene. Then, a human-body-like object is placed in the calculation space. Sections are placed in front of and behind the body, respectively, facing each other. Edges assigned to be stitched together are brought toward each other until they meet and merge. After that, the cloth is deformed naturally by assuming the influences of friction with the body's surface and gravity.



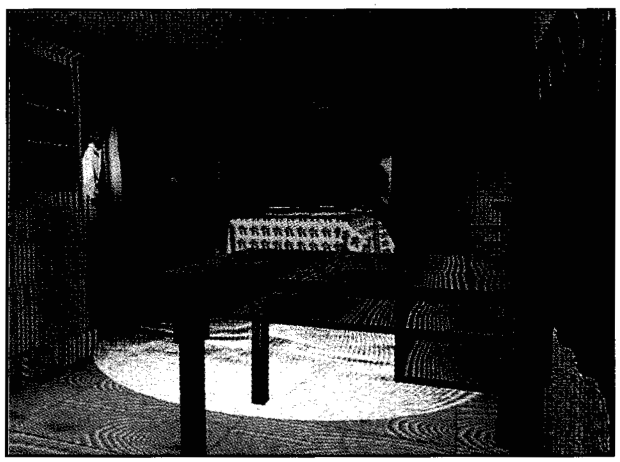


Figure 11

Examples of renditions of an interior.

The last examples shown (**Figure 11**) are renditions in which the tablecloth, curtains, and carpets were all simulated using the approach described here.

Concluding remarks

In this brief overview, we have described a multiple representation method for simulating cloth shapes and motions. The flexibility of the method permits its use under various conditions. The realism of the renditions obtained depends partly on the amount of detail in their data. However, the inclusion of more detail requires more computation, which increases the time needed to obtain a given simulation. The finer the grid, the more realistic the simulation, but the longer the time required to obtain the simulation, making it difficult for users to create trial shapes repeatedly. Moreover, in view of the ways in which the method is used, interactivity is an important factor in providing a more convenient simulation environment. The design of initial patterns and the setting of conditions must be carried out interactively, and this also requires that the computation time be minimal. In both respects, reducing the computation time is a very important requirement in this type of simulation. In practice, design or animation requires many trials for a satisfactory final result to be obtained.

In the examples discussed, use was made of an IBM RISC System/6000 Type 550 processor. Computation times were typically several minutes for the simplest case of a rectangular shape that does not interact with any objects. In the most complicated example discussed, that of the simulation of a garment on a human-like object, it took about three hours to complete the computations. Fortunately, it does not follow that every scene in an animation sequence requires several hours of computation; once a garment has been shaped, a movement that deforms it only slightly can be simulated in several minutes. Nevertheless, a much shorter time is desirable in order to facilitate repeated trials, especially when real-time interactivity is a consideration. Although model simplification reduces computation time, practical considerations may require the contrary. Parallel computation might offer a viable alternative.

RISC System/6000 is a registered trademark of International Business Machines Corporation.

References

1. J. Weil, "The Synthesis of Cloth Objects," *Computer Graph.* **20,** No. 4, 49–54 (1986).
2. M. Aono, "A Wrinkle Propagation Model for Cloth," *Proceedings of Computer Graphics International '90*, 1990, pp. 95–115.
3. T. Kunii and H. Gotoda, "Modeling and Animation of Garment Wrinkle Formation Processes," *Proceedings of Computer Animation '90*, Springer, Tokyo, 1990, pp. 131–147.
4. D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer, "Elastically Deformable Models," *Computer Graph.* **21,** No. 4, 205–214 (1987).
5. M. Carignan, Y. Yang, N. M. Thalmann, and D. Thalmann, "Dressing Animated Synthetic Actors with Complex Deformable Clothes," *Computer Graph.* **26,** No. 2, 99–104 (1992).
6. H. Okabe, H. Imaoka, T. Tomita, and H. Niwaya, "Three Dimensional Apparel CAD system," *Computer Graph.* **26,** No. 2, 105–110 (1992).
7. M. Moore and J. Wilhelms, "Collision Detection and Response for Computer Animation," *Computer Graph.* **22,** No. 4, 289–298 (1988).

Received August 16, 1994; accepted for publication February 20, 1995

Takashi Ohta *Tokyo Research Laboratory, IBM Japan, Ltd., 1623-14 Shimotsuruma, Yamato-shi, Kanagawa 242, Japan (OHTAK at TRLVM, ohtak@trl.ibm.co.jp).* Mr. Ohta is a researcher in the Media Systems Institute at the Tokyo Research Laboratory. He received B.S. and M.S. degrees in physics from Keio University, Japan, in 1985 and 1987, respectively. He also studied at the University of Tokyo and at the Institute of Space and Astronautical Science, with emphasis on computational fluid dynamics. Mr. Ohta joined IBM in 1990 and has since been working on physical simulation modeling for computer graphics.