

실시간 환경에서의 사실적인 옷감 애니메이션

강영민^o 조환규

부산대학교 전자계산학과 그래픽스응용연구실
{ymkang^o, hgcho}@pearl.cs.pusan.ac.kr

Realistic Cloth Animation in Real-time Environments

Young-Min Kang^o Hwan-Gue Cho

Graphics Application Lab, Dept. of Computer Science, Pusan National University

요약

본 논문은 실시간 옷감 애니메이션을 생성하기 위한 기법을 설명한다. 이 논문에서는 질량 스프링 모델에 기반한 물체를 가상현실 환경에서 실시간에 움직이기 위한 수치적인 기술을 다루고 있다. 많은 연구자들이 옷감과 같이 객체의 움직임과 외형을 표현하기 위한 다양한 기법들을 제안했지만, 옷감 객체를 가상 현실 환경에서 실시간으로 움직이게 하는 일은 여전히 어려운 문제이다. 옷감 애니메이션의 가장 큰 문제는 옷감의 움직임을 표현하는 운동 방정식을 수치적분하는 것이 안정적으로 수행되지 않는다는 것이다. 1998년 Baraff의 논문에서 제시된 것과 같이 암시적 적분법이 이러한 안정성 문제를 해결해 주기는 하지만, 암시적 적분법은 수치적분 문제를 선형시스템 풀이 문제로 바꾸기 때문에 사실적인 외형을 가진 복잡한 모델의 애니메이션을 실시간 혹은 상호작용적으로 생성하는 문제에는 그대로 적용할 수가 없다[1]. 암시적 적분법의 이러한 문제점들을 해결하고 실시간 혹은 상호작용적으로 생성하는 문제에는 그대로 적용할 수 있는 근사 기법들이 제안되었다. 하지만, 이러한 근사 기법들은 지나친 근사에 기초하고 있기 때문에 실제 옷감의 움직임과 같은 사실적인 동작을 생성하지 못하였다. 이 논문에서 소개할 기법은 사실적인 옷감 주름을 생성할 수 있을 정도로 복잡한 옷감 모델을 다루면서도, 이전의 근사 기법들이 생성할 수 없었던 사실적인 움직임을 얻을 수 있다. 따라서, 본 논문의 기법은 가상 현실 환경에서 시스템 전체의 상호작용성을 해치지 않으면서도 매우 사실적인 옷감 모델을 포함할 수 있도록 해 준다. 제안된 기법은 수치적분의 안정성을 위해 암시적 적분법에 기반하고 있으면서도, 선형시스템의 해를 효과적이면서도 사실성을 해치지 않도록 근사하여 실시간 옷감 애니메이션을 생성한다.

1. 서론

많은 연구자들이 옷감의 정확한 움직임을 재현하려는 다양한 기법들을 제안하였지만 사실적인 실시간 옷감 애니메이션은 여전히 어려운 문제이다. 실시간 애니메이션의 어려움을 해결하기 위한 연구들 가운데 가장 중요한 최근의 성과는 암시적 적분법을 이용하여 안정성을 보장하고 큰 시간 간격을 사용할 수 있게 한 것이다. 그러나, 암시적 적분법은 선형시스템의 풀이를 요구하는데, 이 선형시스템에 포함된 행렬의 크기는 옷감 모델을 구성하는 전체 입자 수의 제곱에 비례한다 [1]. 따라서, 암시적 적분법을 이용해도 복잡한 옷감 모델을 실시간에 움직이는 것은 현재의 하드웨어적 환경에서는 여전히 불가능하다.

몇몇 연구자들이 암시적 적분법의 안정성을 이용하면서도 선형시스템 풀이 과정의 계산 부담을 줄일 수 있는 효율적인 기법들을 제안하여 실시간 옷감 애니메이션에 이용하려고 하였지만 [5,7], 이러한 기법들은 실시간 성능을 얻기 위해 정확성을 희생하였다. Desbrun 등은 선형시스템에 포함된 행렬을 상수 행렬로 근사한 뒤 그 역행렬을 구해 매 시간간격마다 이 역행렬을 주어진 힘 벡터에 곱해서 곱하는 방법을 사용하였는데, 모델을 구성하는 질점 수의 제곱에 비례하는 크기의 행렬을 가지고, 사용되는 역행렬은 최소행렬이 아니기 때문에, 복잡한 모델에는 여전히 적합하지 않다는 한계를 가진다. 이전 연구를 통해 우리는 암시적 적분법에 기반하여 안정성을 보장하면서도, 옷감 애니메이션 시에 나타나는 선형시스템의 해를 직접 갱신식을 통해 효과적으로 근사하는 기법을 제안하였으나, 이 기법은 안정성과 계산의 효율성을 위해 지나치게 옷감 모델의 동작을 느리게 만드는 결점을 가져와서 질점의 수가 많고 질량 스프링 모델의 경직도가 높은 경우에 실제 옷감에 비해 매우 느린 동작을 생성하게 된다 [7]. Oshita와 Makinouchi는 적은 수의 질점으로 이루어진 옷감 모델을 이용하여 실시간 옷감 애니메이션을 생성하고 주름은 기하적 기법을 이용하여 생성하는 방법을 제안하였으나 사실적인 주름을 생성하지 못하였다 [9]. 최근에는 Cordier와 Magnat-Thalmann에 의해 옷감을 완전히 갖춰 입은 캐릭터 모델의 실시간 애니메이션 기법이 제안되었다. 이 기법의 기본적인 아이디어는 옷감을 여러 부분으로 나눈 뒤, 각 부분의 특성에 맞는 애니메이션 기법을 사용한다는 것이다. 그러나, 치마와 같이 반드시 물리적 시뮬레이션을 요구하는 부분에서는 이전의 전통적인 시뮬레이션 기법을 사용했기 때문에, 이러한 부분이 전체 의상에서 많은 부분을 차지하는 경우에는 실시간 애니메이션이 힘들다 [4].

2. 문제의 형식화

실시간 옷감 애니메이션의 가장 중요한 문제는 시스템의 안정성이다. 이러한 불안정성 문제는 암시적 적분법에 의해 해결될 수 있다. 그러나, 암시적 적분법은 결국 선형시스템 풀이 문제로 바뀌는데, 이 선형시스템은 옷감 모델을 구성하는 질점의 수가 n 일 때, $O(n^2)$ 의 크기를 가진다. 본 논문에서는 질량 스프링 모델에 기반한 옷감 모델을 고려한다. 질량 스프링 모델은 옷감 객체가 질점과 스프링 에지로 이루어졌다고 가정하고, 각 질점에 가해지는 힘을 스프링 에지의 변형에 따라 구하여 이를 적분하여 속도, 위치를 얻는 모델이다. 따라서, 옷감 애니메이션은 힘을 수치적분하여 속도를 얻고, 속도를 수치적분하여 새로운 위치를 얻는 것이다. 그러나, 안정성 문제는 이러한 간단한 수치적분 문제가 실시간에 이루어지는 것을 막는다. 수치적분의 안정성은 암시적 적분법을 통해 보장될 수 있으므로, 우리는 질량 스프링 모델에 기반한 옷감 모델의 동작을 다음과 같은 역 오일러(backward Euler) 기법을 통해 안정적으로 생성할 수 있다 [8]:

$$\begin{pmatrix} v^{t+h} \\ x^{t+h} \end{pmatrix} = \begin{pmatrix} v^{t+h} M^{-1} f^{t+h} \\ x^{t+h} v^{t+h} \end{pmatrix} \quad (1)$$

식 1에서 h 는 시간 간격을 의미하며, v 는 질점들의 속도를 원소로 하는 벡터이며, f 와 x 는 각각 질점들의 힘과 위치를 원소로 하는 벡터이다. 그리고, 행렬 M 은 질량 행렬이다. 위치차 t 와 $t+h$ 는 시간을 나타내는데, 의미적으로 현재 시간 상태와 다음 상태를 의미한다. f_i , x_i , v_i 와 m_i 는 각각 질점 i 의 힘, 위치, 속도, 질량이라고 할 때, 식 1에 나타난 벡터와 행렬들의 원소를 좀 더 이해하기 쉽게 나타내면 다음과 같다:

$$f = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{pmatrix}, \quad x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \quad v = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix},$$

$$M = \begin{pmatrix} M_1 & 0 & \cdots & 0 \\ 0 & M_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & M_n \end{pmatrix}, \quad M = \begin{pmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_i \end{pmatrix} \quad (2)$$

이렇게 표현된 수치적분의 계산 목표는 다음 상태의 위치, x^{t+h} 를 찾는 것이다. 다음 상태의 위치 x^{t+h} 는 사실 다음 상태의 속도인 v^{t+h} 를 얻으면 간단히 구할 수 있다. 또한 $v^{t+h} - v^t$ 를 의미하는 속도 변화 Δv^{t+h} 를 알면 계산할 수 있다. 따라서, 옷감 애니메이션은 결국 다음과 같은 속도 변화를 얻는 문제이다:

$$\Delta v^{t+h} = h M^{-1} f^{t+h} \quad (3)$$

질량 스프링 기반의 애니메이션이 식 3과 같이 간단한 식으로 표현될 수 있지만, 실시간 애니메이션은 여전히 쉽지 않은 문제이다. 훅(Hooke)의 법칙을 적용해 얻을 수 있는 스프링 힘은 오로지 현재의 스프링 변형에 의한 현재 상태의 힘인 f^t 뿐이기 때문에, f^{t+h} 는 정확한 값이 아니라 다음과 같이 현재 상태의 힘과 그 힘의 미분치를 이용한 근사로만 알 수 있기 때문이다. 본 논문에서는 여러 가지 근사 방법 가운데, 다음과 같이 간단한 1차 미분에 의한 근사를 고려한다:

$$f^{t+h} = f^t + \frac{\partial f}{\partial x} \Delta x^{t+h} = f^t + J \Delta x^{t+h} \quad (4)$$

이때 J 는 힘 벡터의 위치벡터에 대한 자코비안(Jacobian) 행렬이며, Δx^{t+h} 는 시간 t 에서 $t+h$ 사이에 일어나는 각 질점의 위치 변화를 원소로 하는 벡터이다. 행렬 J 의 크기는 $n \times n$ 이며, 각 원소는 3×3 의 크기를 가지는 부행렬이다. 위치 변화 Δx^{t+h} 는 $v^{t+h} \Delta t$ 로 나타낼 수 있기 때문에, 질량 스프링 모델의 애니메이션은 결국 다음과 같은 선형 시스템을 푸는 문제가 된다.

$$\Delta v^{t+h} = h M^{-1} f^t + h^2 M^{-1} J v^t + h^2 M^{-1} J \Delta v^{t+h} \quad (5)$$

식 5를 적절히 재배치하여 다음과 같은 $Ax = b$ 형태의 선형 시스템으로 표현할 수 있다:

$$(M - h^2 J) \Delta v^{t+h} = h f^t + h^2 J v^t \quad (6)$$

질량 스프링 모델 기반 옷감 모델의 안정적인 애니메이션은 식 6에 나타난 선형 시스템의 해를 구하는 문제가 된다. 그러나, 안정성이 보장된다 하더라도, 식 6의 선형시스템을 제한된 시간에 풀어나갈 수 없다면 실시간 애니메이션을 얻을 수 없는데, 일반적으로 식 6에 포함된 행렬 $M - h^2 J$ 는 매우 큰 행렬이다. 다행히 이 행렬이 희소 행렬이며 대칭 행렬이기 때문에 효율적으로 풀 수 있기는 하지만, 복잡한 모델이라면 선형 시스템의 크기가 지나치게 커져서 실시간 애니메이션이 어렵다. 사실적인 가상 환경을 보장하기 위해서는 반드시 충분히 많은 질점이 사용되어야 하므로 본 논문에서 다루는 사실적인 옷감 모델의 실시간 애니메이션은 식 6으로 표현되는 선형 시스템의 크기가 매우 큰 경우를 다루게 된다.

3. 옷감 모델의 실시간 애니메이션

행렬 $M - h^2 J$ 의 크기가 질점 수에 따라 급격하게 증가하지만, 일반적으로 희소행렬이기 때문에 반복 기법을 통해 빠르게 해를 근사할 수 있다. 비록 희소행렬이라는 특성만으로 실시간 애니메이션을 보장할 수는 없지만, 이 특성은 실시간 애니메이션 기법을 구현하는데 매우 중요한 특성이 된다. 설명을 단순히 하기 위해 $M - h^2 J$ 를 간단히 W 로 표현하자. 행렬 W 는 매우 중요한 여러 특성을 가지며, 이러한 특성들을 적절히 이용함으로써 실시간 애니메이션이 가능하다.

3.1 선형 시스템 해의 반복적 근사 기법

식 6의 우변은 $h(f^t + h J v^t)$ 로 표현될 수 있으며, 추가적인 항 $h J v^t$ 는 정성적이다 [5]. 자코비안 행렬이 희소행렬이라는 특성 때문에, 이 정성적을 간단히 계산할 수 있으며, 원래의 스프링 힘과 이 정성적 항을 전체 내부힘 \tilde{f} 로 표현하면, 내부힘 벡터의 각 원소는 아래와 같이 연결된 질점만을 고려함으로써 쉽게 계산할 수 있다:

$$\begin{aligned} \tilde{f}_i^t &= f_i^t + h \sum_{(j) \in E} J_{ij} v_j^t + h J_{ii} v_i^t \\ &= f_i^t + h \sum_{(j) \in E} J_{ij} (v_j^t - v_i^t) \end{aligned} \quad (7)$$

따라서, 식 6의 선형 시스템은 다음과 같이 간단히 표현될 수 있다:

$$W \Delta v^{t+h} = h \tilde{f}^t \quad (8)$$

행렬 W 의 특성에 따라, 식 8의 선형 시스템은 다음과 같은 n 개의 방정식으로 표현 가능하다:

$$\begin{aligned} W_{11} \Delta v_1^{t+h} &= h \tilde{f}_1^t + h^2 \sum_{(j) \in E} J_{1j} \Delta v_j^{t+h} \\ W_{22} \Delta v_2^{t+h} &= h \tilde{f}_2^t + h^2 \sum_{(j) \in E} J_{2j} \Delta v_j^{t+h} \\ &\vdots \\ W_{nn} \Delta v_n^{t+h} &= h \tilde{f}_n^t + h^2 \sum_{(j) \in E} J_{nj} \Delta v_j^{t+h} \end{aligned} \quad (9)$$

식 9의 i 번째 방정식을 질점 i 의 속도 변화를 중심으로 다시 정리하면 다음과 같이 표현될 수 있다:

$$\Delta v_i^{t+h} = W_{ii}^{-1} h \tilde{f}_i^t + h^2 \sum_{(j) \in E} J_{ij} \Delta v_j^{t+h} \quad (10)$$

식 10에 포함된 행렬 M_{ij} , J_{ij} , 그리고 J_{ii} 는 모두 3×3 의 대칭 행렬이며, 벡터 $\sum_{(j) \in E} J_{ij} \Delta v_j^{t+h}$ 는 연결된 질점들의 다음 단계에서의 속도 변화 Δv_j^{t+h} 를 알고 있다면 적은 양의 계산을 통해서 구할 수 있다. 그러나, 이런 성능은 각 질점의 속도 변화가 식 10에 의해 독립적으로 계산될 수 있을 때만 가능한 성능인데 각 방정식은 다른 방정식들에 의존적으로 연결되어 있다는 것이 문제가 된다.

본 논문에서 제안하는 방법은 식 10에 나타난 속도 변화 갱신식에 따라 반복적으로 갱신함으로써 선형 시스템의 해를 근사하는 것이다. 사실 새로운 점자 k 를 도입하여 반복 회수를 나타낸다면, 식 10은 자코비(Jacobi) 반복법과 동일하다. 즉 $\Delta v_i^{t+h(k)}$ 가 k 번의 반복에 얻어진 근사해라고 하고, 초기값을 간단히 $\Delta v_i^{t+h(0)} = W_{ii}^{-1} h \tilde{f}_i^t$ 라고 하면, 반복적 갱신 방법은 다음과 같이 얻어진다:

$$\begin{aligned} \Delta v_i^{t+h(0)} &= \\ \Delta v_i^{t+h(k+1)} &= W_{ii}^{-1} (h \tilde{f}_i^t + h^2 \sum_{(j) \in E} J_{ij} \Delta v_j^{t+h(k)}) \end{aligned} \quad (11)$$

식 11을 이용하여 적은 수의 반복을 수행하면, 옷감 모델의 움직임을 얻기 위한 선형 시스템의 근사해를 실시간에 얻을 수 있다. 본 논문의 반복 기법이 암시적 적분법에 기반해 있기 때문에, 애니메이션 결과는 실시간 애니메이션 시스템에 적용하기에 충분할 정도로 안정적이다. 속도 변화를 반복적으로 갱신하는 과정에서, 행렬 W_{ii} 와 벡터 $h \tilde{f}_i^t$ 는 상수 행렬, 상수 벡터로 유지된다. 따라서 이들 행렬과 벡터들은 하나의 시간 간격 내에서 일어나는 반복적 갱신 과정에서 새로이 계산될 필요가 없다. 이들 상수 행렬을 W_i , 그리고 상수 벡터를 p_i 라고 다음과 같이 정의하자:

$$W_i = M - h^2 J_{ii} \in R^{3 \times 3} \quad (12)$$

$$p_i = h W_i^{-1} \tilde{f}_i^t \in R^3 \quad (13)$$

실험을 통해 단 한 번의 반복을 통해 얻은 결과도 안정적이며 사실적인 동작을 보여주었다. 따라서, 우리는 실시간 애니메이션을 위해 다음과 같은 속도 변화의 근사해를 사용하였다:

$$\Delta v_i^{t+h} \approx \Delta v_i^{t+h(1)} = p_i + h^2 W_i^{-1} \left(\sum_{(j) \in E} J_{ij} p_j \right) \quad (14)$$

3.2 시스템 안정성을 위한 힘의 미분치 근사

이 절에서는 힘 벡터의 자코비안 행렬을 어떻게 구할 것인지를 설명한다. 질량 스프링 모델을 사용할 경우, 질점 i 와 j 를 연결하는 스프링에 의해 질점 i 에 가해지는 스프링 힘 f_{ij}^t 는 다음과 같이 훅(Hooke)의 법칙에 의해 간단히 계산될 수 있다:

$$f_{ij}^t = \kappa_{ij} (|x_j - x_i| - l_{ij}^0) \frac{x_j - x_i}{|x_j - x_i|} \quad (15)$$

이때, κ_{ij} 는 질점 i 와 j 를 연결하는 스프링 에지의 스프링 상수이며,

l_{ij}^0 는 이 스프링의 휴지상태 길이이다. 질점 i 에 가해지는 전체 힘은 질점 i 에 연결된 모든 스프링에 의해 가해지는 힘의 합이다. 질점 i 에 가해지는 힘의 질점 j 의 위치에 대한 미분치는 다음과 같다:

$$\frac{\partial f_i}{\partial x_j} = \kappa_{ij} \frac{|x_j - x_i| - l_{ij}^0}{|x_j - x_i|} I_3 + \kappa_{ij} \frac{l_{ij}^0}{|x_j - x_i|} \frac{(x_j - x_i)(x_j - x_i)^T}{|x_j - x_i|^2} \quad (16)$$

이 3×3 행렬이 전체 힘의 자코비안 행렬 J 의 i 행 j 열 원소가 된다. 그런데, 식 16은 중요한 문제를 가지고 있다. 두 질점 i 와 j 가 서로 가까워지면, 이 미분치 행렬의 원소값들은 무한히 커질 수 있다는 것이다. 즉, 스프링의 길이가 줄어든 상태에서 이 값들을 그대로 적용하면 지나치게 큰 값으로 인해 시스템의 안정성이 깨어져 시뮬레이션에 실패할 수 있다는 것이다 [3]. 이러한 상황을 피하기 위해서는 이 값들이 항상 시스템을 안정적으로 유지할 수 있는 안전한 값을 갖도록 보장에 주어야 한다. 식 16은 스프링이 늘어남 상태에서는 아무런 문제도 발생시키지 않는다. 따라서, 스프링의 길이가 $|x_j - x_i|$ 가 스프링의 원래 휴지길이 l_{ij}^0 보다 큰 경우에는 식 16을 그대로 자코비안 행렬에 사용한다. 다만, 언제나 안전한 값을 가질 수 있도록 본 논문의 기법은 스프링 수축시에 스프링의 휴지 길이가 언제나 현재 스프링 길이와 동일하다고 가정한다. 즉, 스프링 수축시에는 $l_{ij}^0 = |x_j - x_i|$ 로 가정하는 것이다. 따라서, 질점 i 에 가해지는 힘의 질점 j 위치에 대한 미분은 다음과 같이 다시 표현할 수 있다:

$$\text{if } (l_{ij}^0 \geq |x_j - x_i|) \quad \frac{\partial f_i}{\partial x_j} = \kappa_{ij} \frac{(x_j - x_i)(x_j - x_i)^T}{|x_j - x_i|^2} \quad (17)$$

식 17을 이용함으로써 본 논문에서 제안한 기법은 언제나 안전한 대응 상태 힘을 이용할 수 있었고, 실험 결과 식 17을 통해 근사한 힘의 미분치는 옷감 애니메이션의 사실성을 크게 떨어뜨리지 않았다.

4. 실험결과

본 논문의 기법을 구현한 결과가 그림 1에 나타나 있다. 실험은 일반적인 사용자들이 사용하고 있는 수준의 PC환경을 고려하여 1GHz 미만의 PENTIUM CPU를 가진 PC환경에서 테스트되었다. 실험 결과 본 논문의 기법은 그림 1의 (a)에 나타난 것과 같은 사실적인 옷감 모델을 실시간에 생성할 수 있었다. 그림 1의 (a)에 나타난 모델은 2048개의 삼각형으로 이루어진 메쉬에 텍스처를 입힌 것이다. 그림 1의 (b)는 본 논문의 기법을 이용하여 생성한 깃발을 실제 게임 프로그램에 통합한 결과이다. 본 논문에서 제안한 기법은 게임의 상호작용성을 전혀 해치지 않으면서 그림 1의 (b)에 나타난 것과 같은 깃발 모델을 생성할 수 있었다. 그림에 나타난 깃발 모델은 사용자의 동작에 따라 이동하는 차량에 부착되어 있는 것으로 다양한 사용자의 조작에 부합하는 자연스러운 깃발 동작을 생성할 수 있었다.

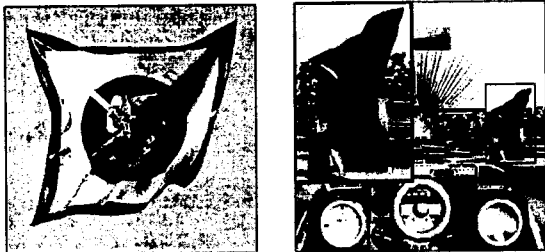


그림 1. 본 논문의 기법을 이용하여 생성한 결과: (a) 복잡한 모델의 실시간 애니메이션 결과 (b) 게임 환경에 적용된 결과

5. 결론

본 논문에서는 옷감 객체의 움직임을 가상현실 환경에서 실시간에 생성할 수 있는 효율적인 기법을 제안하였다. 이전에 제안된 여러 가지 실시간 기법들은 힘의 미분치를 지나치게 근사하여 안정성은 보장하는 반면 동작의 사실성이 크게 떨어지는 단점을 가지고 있었다. 대표적인 실시간 옷감 애니메이션 기법인 Desbrun의 미리 계산된 필터 기법은 힘의 미분치를 지나치게 근사할 뿐만 아니라, 옷감 애니메이션을 암시적 적분법으로 다룰 때 생기는 선형 시스템의 행렬이 최소행렬이라는 중요한 특성도 활용하지 못하고 있다. 우리의 이전 연구에서 개발된 근사 기법은 이러한 최소행렬 특성을 이용하기는 하였지만, Desbrun 방식과 마찬가지로 지나친 근사를 사용하기 때문에 결국 사실적인 동작을 생성할 수 없었다. 이러한 동작의 사실성 한계를 극복하기 위해 본 논문에서 제안된 기법은 옷감 애니메이션의 사실성과 시간 효율성을 높이기 위해 어떤 기하적 기법을 사용하지 않으면서도 수치적분의 해를 효과적으로 근사하고 있다. 따라서, 본 논문에서 제안된 기법은 특별한 메쉬 조작이나 기술적으로 어려운 사전 처리를 거치지 않고도 일반적인 옷감 메쉬 모델의 실시간 애니메이션을 쉽게 생성할 수 있게 해 준다. 또한, 본 연구에서 제안된 기법이 사용하는 힘의 미분치는 이전의 근사 기법과 달리 가능한 정확한 값을 갖도록 하여 움직임의 사실성을 더욱 높였으며, 정확한 힘 미분치를 사용할 경우 스프링 수축시 발생하는 바람직하지 못한 상황을 피하기 위해 수축시 미분치를 계산할 때 언제나 현재의 스프링 길이가 스프링의 휴지 상태 길이가 되도록 하는 방법을 채택하여 자연스러운 동작을 유지하면서도 시스템의 실패를 회피할 수 있도록 하였다. 본 논문에서 제안된 기법을 이용하면, 사실적인 옷감 주름을 포함할 수 있는 복잡한 메쉬의 자연스러운 움직임을 실시간 환경에 손쉽게 통합할 수 있을 것이다.

참고문헌

- [1] David Baraff and Andrew Witkin. Large steps in cloth simulation. *Proceedings of SIGGRAPH 98*, pp. 43-54, 1998.
- [2] A. Vlachos, J. Peters, C. Boyd, and J. Mitchell. Curved PN triangles. *Symposium on Interactive 3D Graphics 2001*, pp. 159-166, 2001.
- [3] Kwang-Jin Choi and Hyeong-Seok Ko. Stable but responsive cloth. *ACM Transactions on Graphics: Proceedings of SIGGRAPH 2002*, pp. 604-611, 2002.
- [4] Frederic Cordier and Nadia Magnenat-Thalmann. Realtime animation of dressed virtual humans. *Proceedings of Eurographics 2002*, 2002.
- [5] Mathieu Desbrun, Peter Schroder, and Alan Barr. Interactive animation of structured deformable objects. *Graphics Interface '99*, pp. 1-8, 1999.
- [6] Bernhard Eberhardt, Andreas Weber, and Wolfgang Strasser. A fast, flexible particle-system model for cloth draping. *IEEE Computer Graphics & Applications*, 16(5):52-59, September 1996.
- [7] Young-Min Kang, Jeong-Hyeon Choi, Hwan-Gue Cho, and Chan-Jong Park. An efficient animation of wrinkled cloth with approximate implicit integration. *The Visual Computer Journal*, 17(3):147-157, 2001.
- [8] M. Kass. An introduction to continuum dynamics for computer graphics. In *SIGGRAPH Course Note: Physically-based Modelling*. ACM SIGGRAPH, 1995.
- [9] Masaki Oshita and Akifumi Makinouchi. Real-time cloth simulation with sparse particles and curved faces. *Proc. of Computer Animation 2001*, pp. 220-227, November 2001.
- [10] Demetri Terzopoulos, John Platt, Alan Barr, and Kurt Fleischer. Elastically deformable models. *Computer Graphics (Proceedings of SIGGRAPH 87)*, 21(4):205-214, July 1987.
- [11] Pascal Volino, Martin Courshesnes, and Nadia Magnenat Thalmann. Versatile and efficient techniques for simulating cloth and other deformable objects. *Proceedings of SIGGRAPH 95*, pp. 137-144, August 1995.