

가상의복 디자인을 위한 인터랙티브한 패턴조작

Automatic Pattern Manipulation for Virtual Garment Design

정휘룡 Hwiryong Jung, 원광연 Kwangyun Wohn

한국과학기술원 문화기술대학원

요약 본 논문에서는 기존 패턴캐드시스템 및 드레이핑시스템을 이용한 의복제작과정의 한계점과 개선가능성에 대해 알아보고, 이를 기반으로 새로 디자인한 의복제작시스템을 소개한다. 이 시스템은 원피스, 바지, 셔츠, 재킷 등 옷의 큰 스타일인 의복템플릿으로부터 의복디자인을 시작 한다. 소매, 카라, 포켓, 넥라인과 같은 템플릿 디테일들은 디자이너가 마음대로 다른 세부 템플릿들과 교체할 수 있으며, 세부 템플릿으로 커버할 수 없는 디테일한 수정은 3D 드레이핑 시스템과 2D패턴에디터에서의 직관적인 오퍼레이션을 통해 이루어진다.

핵심어: VR, Clothes Simulation

1. 서론

1980년대 중반부터 연구되기 시작한 Cloth Simulation 기술은 현재 패션산업에서 뿐만 아니라 게임, 애니메이션, 영화특수효과에 이르기까지 다양한 분야에서 가상의복을 디자인하고 시각화하는데 사용되고 있다. 적용되는 영역에 따라, 현실 세계에서 실제 의복을 제작하는 기초로 가상의복을 사용하기도 하고, 현실 세계에서의 활용과는 상관없이 오로지 가상 세계의 아바타를 위해 가상의복을 사용하기도 한다. 하지만 한 가지 공통되는 점은 어느 분야에서 사용되는 유사한 가상의복 디자인 프로세스를 거치고, 또 유사한 시뮬레이션과정을 거쳐서 가상의복을 시각화 한다는 것이다. 게임이나 애니메이션, 영화 특수효과에서 사용하는 가상의복은 Maya나 3dsMax에서 제작하고, 패션 분야에서 사용하는 가상의복은 전용 패턴캐드나 드레이핑 시스템 등을 사용하여 제작하지만 이러한 구분은 단지 겉으로 보이는 차이점일 뿐 내부적인 디자인 프로세스를 뜯어보면 여러 영역에서 사용하는 각기 다른 가상의복 제작 툴에서의 의복 디자인과정과 시뮬레이션 과정이 유사함을 알 수 있다.

이러한 디자인툴에서의 가상의복 제작방법은 전통적인 의복 디자인 방법에 기초하고 있다. 때문에 철저하게 패션디

자이너와 패턴사와 같은 패션업계 종사자가 예전에 사용하던 기존 의복 제작방법대로 시스템이 구성되어 있다. 이는 패션 도메인에서 사용하는 툴뿐만 아니라 Maya나 3dsMax에서도 마찬가지이다. 가상의복을 디자인하는 사람은 어떤 툴을 쓰던 패턴을 만들고, 박음질 선을 설정하고, 아바타의 몸에 패턴을 배치시키는 과정을 반복해야한다. 이러한 현실 세계의 의복 디자인 과정에 기반한 가상의복디자인 프로세스는 여러 측면에서 불편한 점과 개선할 점을 내포하고 있다. 우선 현재 많은 시스템들이 패턴을 디자인하는 과정과 의복을 시뮬레이션 하는 과정이 나누어져 있다. 그러기 때문에 패턴 수정과 결과 확인이 동시에 이루어지지 않는다. 특히 패턴 에디터와 드레이핑 시스템이 나누어져 있는 패션 도메인에서 사용하는 툴의 경우 패턴을 수정할 때마다 re-Triangulation하고 다시 패턴을 아바타 주위에 배치시키는 과정을 필요로 하기 때문에 디자인 프로세스가 비효율적일 수밖에 없다. 그리고 의복의 디자인 과정이 철저하게 패턴중심으로 짜여 있기 때문에 직관적으로 의복을 디자인할 수 없는 제약점이 존재한다.

이러한 가상의복 제작과정의 비효율성과 비직관성은 1차적으로는 컴퓨터 기반으로 의복 디자인 환경을 옮겨가고 있는 패션디자이너들에게 마이너스로 작용하고, 2차적으로는

3D 애니메이터와 같이 다른 도메인에서 가상의를 제작하는 디자이너들에게 마이너스로 작용한다. 그래서 이 논문에서는 기존의 가상의복 디자인 프로세스의 문제점과 개선가능성을 살펴보고, 가상 의복 디자이너가 보다 쉽고 편리하게 가상의를 디자인할 수 있는 프로세스를 제안한다.

2. 가상의를 위한 인터랙티브한 패턴조작

2장에서는 본격적으로 이 연구에서 제안하는 가상의를 위한 인터랙티브한 패턴조작 프로세스에 대해 알아본다. 먼저 2.1장에서 전반적인 시스템의 구조와 디자인 프로세스에 대해 알아보고, 이어지는 장들에서 각각의 단계에 대해 상세히 이야기 하도록 한다.



2.1. 전체 가상의복 제작 프로세스



그림 1. 시스템 구성

프로토타입으로 제작된 시스템은 위와 같은 구조로 구성되어 있다. 오른쪽은 2D 패턴에디터 윈도우이다. 디자이너는 이 윈도우에서 패턴을 디자인한다. 왼쪽은 드레이핑 윈도우로서 패턴으로부터 만들어질 의복을 확인 할 수 있다.

이 논문에서 제안하는 디자인 프로세스의 핵심은 크게 3가지로 나누어 볼 수 있다.

- 2D 패턴에디터에서의 인터랙티브한 의복 수정
- 3D 드레이핑 윈도우에서의 직관적인 의복 수정
- 템플릿 선택과 디테일 교체에 기반한 의복 디자인

첫 번째 항목인 2D 패턴 에디터에서의 인터랙티브한 의복수정은 패턴을 수정했을 때 그 결과를 실시간으로 2D 패턴 에디터와 3D 드레이핑 윈도우에서 확인할 수 있게 하자는 것이다. 여기에는 추가적으로 Seam Line Constraint를 자동으로 처리하여 패턴을 수정하는 부분이 포함된다.

두 번째 항목인 3D 드레이핑 윈도우에서의 직관적인 의복 수정은 2D 패턴 에디터가 아닌 3D 드레이핑 윈도우에서 직접 의복을 수정할 수 있도록 하는 기능이다. 가봉 수준의 전반적인 폭과 길이를 수정하는 기능뿐만 아니라 원하는 부분이면 어떤 디자인 디테일이라도 3D 드레이핑 윈도우에서 디자인을 변경 할 수 있다.

세 번째 항목인 템플릿 선택과 디테일 교체에 기반한 의복 디자인은 옷의 스타일별로 2D 패턴과 3D 의복 메쉬의 템플릿을 데이터베이스화 해 제공함으로써 보다 편리하게 디자이너가 의복을 제작할 수 있도록 하는 기능이다. 선택된 템플릿에서 세부적인 소매, 넥라인 같은 디테일을 교체하는 기능 역시 제공된다.

본 논문에서 제안하는 디자인 프로세스를 활용한 가상 의복 제작 순서를 살펴보도록 하자.

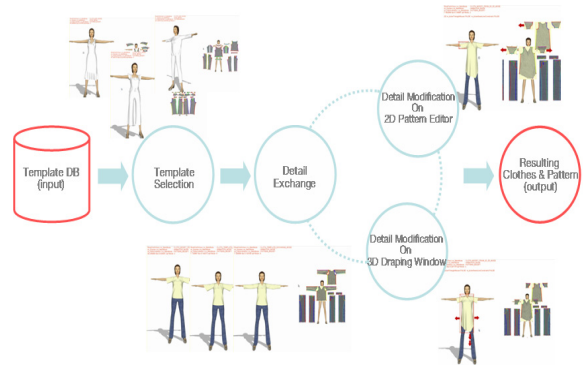


그림 2. 전체 디자인 프로세스

전반적인 디자인 순서는 위에서 보는 바와 같다. 가장 왼쪽의 Template DB는 미리 만들어서 데이터베이스화 해놓은 가상의를 템플릿이다. 이 데이터베이스에는 옷의 기본 스타일 별로 2D 패턴과 3D 의복 메쉬가 저장되어있다. 먼저 Template Selection 단계에서 가상의를 디자이너는 자신이 원하는 의복의 스타일에 맞추어 의복 템플릿을 선택한다. 이 때 데이터베이스에서 2D 패턴과 3D 의복 메쉬가 동시에 로드되어 패턴에디터와 드레이핑 윈도우에 배치된다. Template Selection 옆으로 보이는 Detail Exchange와 Detail Modification on 2D Pattern Editor, 그리고 Detail Modification On 3D Draping Window는 일정한 순서를 가지고 처리되는 과정이 아니라 디자이너의 선택에 따라 적절하게 그 순서를 달리하여 반복적으로 수행될 수 있는 과정이다. 먼저 Detail Exchange에 대해 설명을 하면, 이 부분은 선택된 템플릿에서 부분적인 디테일을 교체하는 부분이다. 첫 번째 단계인 Template Selection단계에서 옷의 큰 스타일을 결정했다고 할지라도 소매나 넥 라인 등 부분적인 디테일에 따라서 여러 스타일의 옷이 나올 수 있으므로 이런 세부적인 디테일을 교체하는 기능이 필요하다. 이러한 디테일은 다른 옷의 동일한 디테일을 가져와서 교체하거나 아

니면 디테일만 따로 저장되어있는 디테일 템플릿 데이터베이스에서 선택하여 교체할 수 있다. 세부적인 디테일이 교체되었다고 하더라도 모든 넥라인 스타일과 소매 스타일 등을 디테일 교체만으로 커버 할 수는 없다. 이렇게 디테일 교체로 처리할 수 없는 부분은 Detail Modification On 2D Pattern Editor와 Detail Modification On 3D Draping Window에서 처리한다. Detail Modification On 2D Pattern Editor는 2D 패턴에디터에서의 의복 수정인데 기존 프로세스와의 차이점이라면 패턴 수정결과를 3D 드레이핑 윈도우와 2D 패턴 에디터 상에서 실시간으로 확인이 가능하다는 것이다. 그리고 패턴의 엣지가 수정되었을 때 Seam Line Constraint에 맞게 대응되는 엣지를 적절하게 수정해주는 기능을 제공한다. Detail Modification On 3D Draping Window는 3D 드레이핑 윈도우에서 직접 의복 수정을 가능하게 하는 부분이다. 전반적인 길이, 폭 수정뿐만 아니라 3D 의복 메쉬의 어느 부분이라도 수정이 가능하다.

지금까지 우리는 이 논문에서 제안하는 가상의복 제작 프로세스에 대해 전반적으로 알아보았다. 2.2장부터는 각각의 디자인 단계에 대해 하나하나 상세하게 알아보도록 한다.

2.2. 의복 템플릿 선택



그림 3. 의복 템플릿 선택

가상의복 디자인 프로세스의 첫 번째 단계는 템플릿을 선택해 옷의 전반적인 스타일을 결정하는 것이다. 이 때 원피스나 투피스, 바지, 티셔츠 등 전반적인 옷의 형태가 결정된다. 템플릿은 옷의 스타일별로 디자이너가 미리 제작해 놓은 2D 패턴과 3D 의복 메쉬의 쌍으로 구성된다. 원하는 스타일에 맞추어 템플릿을 선택하면 2D패턴과 3D 의복 메쉬가 동시에 로드되어 드레이핑 윈도우와 패턴에디터에 보여진다. 사용자는 여러 템플릿을 바꾸어가며 자신이 원하는 스타일의 결정하게 된다.

3.3. 템플릿 디테일 교체



그림 4. 템플릿 디테일 교체

템플릿 디테일 교체 단계에서는 소매, 넥라인, 카라 등 세부적인 스타일을 교체하게 된다. 위의 그림은 전체적인 스타일로 바지와 티셔츠를 선택한 후에 소매의 디테일을 교체하는 그림이다. 이때 교체될 디테일들은 다른 의복의 동일한 디테일을 이용하거나, 디테일만 따로 저장해놓은 디테일 템플릿에서 가져오게 된다. 이와 같이 여러 샘플들로부터 부분을 취하여 새로운 모델을 만드는 방법은 컴퓨터 그래픽스 분야에서 많이 연구되어왔다.



그림 5. Modeling by Example[14]

Thomas Funkhouser [9] 등은 데이터베이스화 해놓은 모델 샘플들로부터 부분 부분을 취하여 새로운 모델을 생성해 내는 방법을 제안하였다. 위의 그림은 데이터베이스의 서로 다른 5개의 의자 모델로부터 부분 부분을 가져와서 새로운 의자를 생성하는 예제이다.

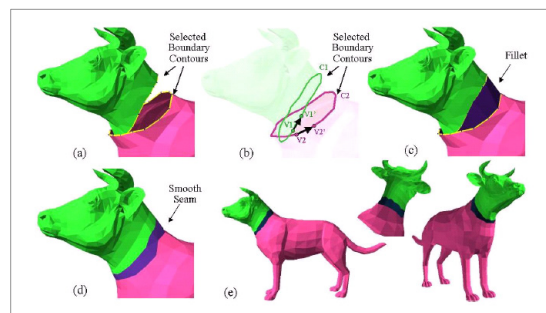


그림 6. Modeling by Example

이와 같은 예제 기반의 모델링 기법들은 공통적으로 3D surface를 자르고 합치는 과정을 필요로 한다. 위의 그림은 소와 개 모델에서 각각 머리와 몸통을 잘라내어 새로운 모델을 생성하는 그림이다. 여기서 역시 3D surface를 자르고 합치는 과정을 확인 할 수 있다.

본 연구에서는 유사하게 예제기반으로 템플릿의 디테일을 교체하는 기능을 제공하고 있지만, 기존의 연구와 다른

접근 방법을 취하고 있다. 드레이핑 윈도우의 3D 의복 메쉬는 패턴에디터의 2D 패턴과 1대 1 매치를 이루고 있다. 이 연구에서는 2D 패턴이 변화될 때, 자동적으로 3D 의복 메쉬를 변화시키는 기능을 제공하고 있다. 이에 대한 자세한 내용은 2.4절에서 상세하게 이야기하도록 한다. 이와 같은 2D 패턴 변화에 따른 자동적인 3D 의복 메쉬 변화 기능을 이용하여, 예제 모델에서 3D Surface를 잘라서 붙이는 노력 없이, 단순히 원본 2D 패턴을 타겟 디테일의 패턴 모양으로 모방(imitation)하는 과정만으로 현재 디자인하고 있는 의복의 디테일을 다른 의복의 디테일로 교체할 수 있다.

가상의복의 부분 패턴을 다른 의복의 부분 패턴으로 모방을 통해 교체하는 과정은 다음 그림과 같다.

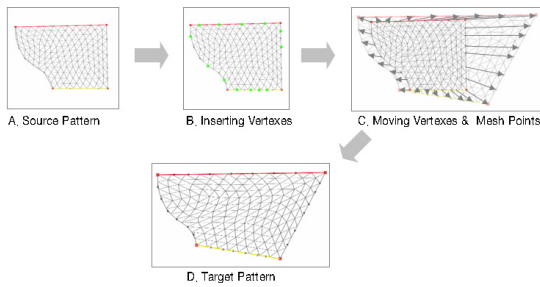


그림 7. Pattern Imitation

그림 4. “템플릿 디테일 교체” 그림에서 첫 번째 반팔 소매를 두 번째 소매로 변화시키는 과정의 예를 들어 설명하도록 한다. 우리는 위의 그림 A에 보이는 원본 패턴을 D에 보이는 타겟 패턴으로 변화시켜야한다. 첫 번째 단계는 외곽선 버텍스의 수를 맞추어 주는 것이다. 타겟 패턴과 원본 패턴의 외곽선 버텍스 수의 차이만큼 원본 패턴에 버텍스를 추가하거나 빼주어 버텍스의 개수를 맞추어준다. 다음 단계는 원본 패턴의 버텍스와 메쉬포인트를 타겟 패턴의 버텍스와 메쉬포인트 위치로 이동시키는 과정이다. 이 과정을 마치면 원본 패턴은 정확하게 타겟 패턴과 동일한 모양을 가지게 된다. 이와 같이 패턴이 변형되면 2.4절에서 설명할 패턴 수정에 따른 3D 메쉬 수정 모듈에 의해 3D 메쉬 역시 변형되고, 결과적으로 원본 가상의복의 2D 패턴과 3D 메쉬가 선택한 디테일로 교체되는 효과를 얻을 수 있다.

2.4. 2D 패턴 에디터에서의 패턴수정

이번 장에서는 2D 패턴에디터에서의 가상의복 수정에 관해 이야기한다. 먼저 2.4.1. 절에서는 이 시스템에서 가장 핵심적인 역할을 수행하는 패턴 수정에 따라 실시간으로 3D 의복 메쉬를 변형하는 부분에 대해 알아보고 이어지는 2.4.2. 절에서는 십라인 제약 조건 처리부분에 대해 알아본다.

2.4.1. 패턴 수정에 따른 실시간 3D 의복 메쉬 수정

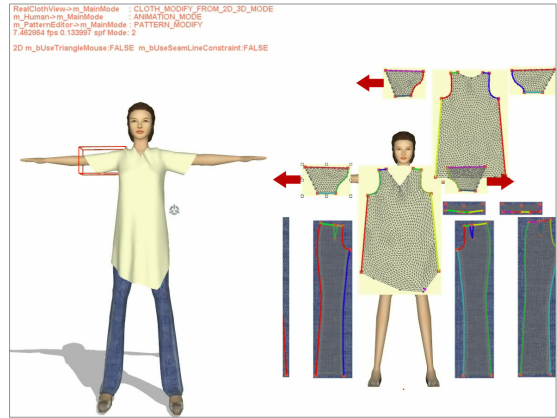


그림 8. 2D 패턴에디터에서의 의복 수정

본 논문이 제안하는 시스템에서는 패턴에디터에서 2D 패턴을 수정했을 때, 드레이핑 윈도우에서 3D 의복 메쉬의 변화를 실시간에 확인할 수 있다. 이와 같이 실시간으로 3D 의복 메쉬의 변화를 확인할 수 있는 이유에 대해 알아보자.

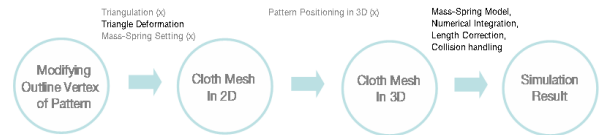


그림 9. 개선된 3D 의복 메쉬 생성과정

기존 프로세스에서는 패턴이 수정되었을 때 외곽선에 맞추어 2D 메쉬를 변형하기 위해 re-Triangulation하고 Triangle Deformation하고 Mass-Spring Setting하는 과정을 반복했어야 했다. 그리고 그 계산량에 따른 시간지연 때문에 패턴을 수정했을 때 실시간으로 드레이핑 윈도우에서 3D 의복 메쉬의 변화를 확인할 수 없었다. 그리고 re-Triangulation될 경우 삼각형, 엣지 그리고 버텍스의 ID 값이 바뀌게 되어 다시 패턴을 3D 드레이핑 윈도우상에서 아바타의 주변에 배치하는 과정이 필요한데, 이 역시 많은 시간을 필요로 하기 때문에 실시간에 3D 의복 메쉬 변형 결과를 확인할 수 없게 하는 원인이 된다.

우리는 패턴의 외곽선이 변형되었을 때 re-Triangulation하는 방법 대신, 기존의 삼각형을 유지시키면서 삼각형의 형태를 적절하게 변화시켜 바뀐 외곽선에 맞게 메쉬를 변형시키는 방법을 사용한다. 결과적으로 위의 그림에서 보는 바와 같이, 패턴을 변형 시켰을 때 Triangulation, Triangle Deformation, Mass-Spring Setting, Pattern Positioning in 3D 과정을 거쳐야 했던 것을, Triangle Deformation 하는 비용만으로 처리하기 때문에 그만큼 빠른 시간 내에 패턴변형을 처리할 수 있고, 드레이핑 윈도우

에서 실시간으로 의복 변화를 확인할 수 있게 된다.

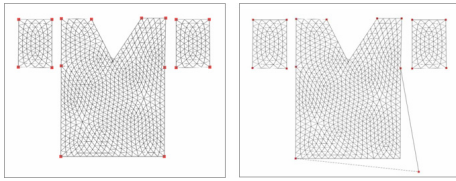


그림 10. 2D 패턴 에디터에서의 패턴 외곽선 수정

이해를 돕기 위해 패턴 외곽선의 변형을 기존 시스템과 이 논문에서 제안하는 시스템이 어떻게 다르게 처리하는지 비교해 보도록 하겠다. 위의 그림 왼쪽에 보이는 패턴은 마우스 오퍼레이션에 의해 오른쪽 그림과 같이 외곽선이 변형된다. 기존 시스템은 이러한 외곽선 변화가 일어나면 re-Triangulation 과정을 거쳐서 아래 왼쪽 그림에서와 같이 다시 2D 메쉬 구조를 생성한다. 이 논문에서 제안하는 시스템은 다시 Triangulation하는 과정을 거치지 않고 기존 삼각형을 유지하면서 그 모양만 변화시켜 변화된 외곽선에 맞게 2D 메쉬를 변형한다. 아래 오른쪽 그림에서 Triangle Deformation만으로 2D 메쉬를 변형시킨 결과를 확인할 수 있다.

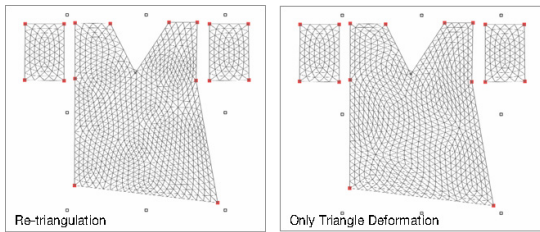


그림 11. Re-Triangulation한 경우와 only Triangle Deformation한 경우 비교

Triangle Deformation으로 2D 패턴 메쉬를 변형시켰으면, 변형된 2D 패턴 메쉬를 반영해 3D 의복 메쉬를 변화시키는 과정이 필요하다. 이 과정은 변형된 2D 패턴 메쉬의 각 엣지의 길이를 3D 의복 메쉬의 Mass-Spring model, Rest Length로 설정해 주면 간단하게 처리된다.

이제까지 2D 패턴이 변화 되었을 때 어떻게 3D 의복 메쉬가 실시간에 변화되는지 살펴보았다. 이 과정은 2D 패턴 에디터에서의 의복 수정뿐만 아니라 템플릿 디테일 교체와 3D 드래이핑 윈도우에서의 의복수정에 공통으로 이용되는 부분이다.

3.4.2. 심라인 제약 조건 처리

디자이너는 패턴 외곽선을 디자인한 후, 서로 박음질되어야 하는 엣지들을 연결하여 심라인을 설정한다. 이 심라인으로 설정된 두 엣지는 2D 패턴으로부터 3D 의복 메쉬를 생

성하는 단계에 하나로 합쳐져서 하나의 메쉬를 생성하게 된다. 심라인으로 연결된 두 엣지는 기본적으로 그 길이가 같아야하고 한쪽이 변화되었을 때 반대쪽이 적절하게 변화되어야 하는 제약조건을 가진다. 한쪽 엣지가 변화되었을 때 반대쪽 엣지를 어떻게 변화시키는지에 따라 심라인 처리방법을 두 가지 정도로 나눌 수 있다.

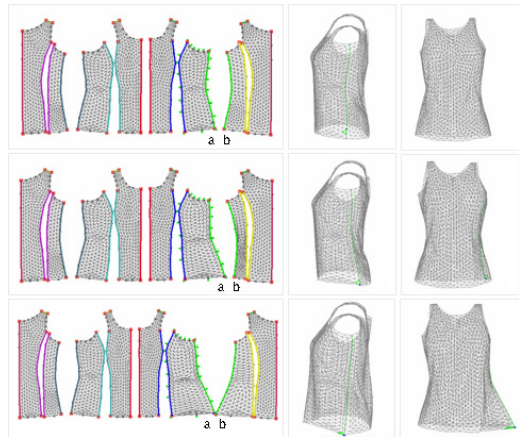


그림 12. 심라인 제약조건 처리

그림 12에서 a, b로 표현된 세로방향의 두 엣지는 서로 심라인으로 연결되어 있다. a를 오른쪽으로 움직였을 때 b를 어느 쪽으로 움직이느냐에 따라 서로 다른 의복 수정 효과를 얻을 수 있다. 먼저 두 번째 줄에서 보는 것과 같이 a가 움직이는 방향과 같은 방향으로 b를 움직여주게 되면, 아랫단의 전체 길이가 유지되기 때문에 두 번째 줄 세 번째 그림에서 보는 것과 같이 앞에서 바라본 옷의 전체 실루엣은 변화가 없게 된다. 하지만 두 번째 줄 두 번째 그림에서 보듯이 심라인의 위치는 바로 위의 그림과 비교해 봤을 때, 이동된 것을 확인할 수 있다. 세 번째 줄과 같이 a를 움직였을 때, x축 기준으로 반대 방향으로 b를 움직이게 되면 밑단의 폭이 늘어나게 되는 것이므로, 세 번째 줄 세 번째 그림에서 보는 것과 같이 티셔츠의 밑단 폭이 늘어나는 것을 확인할 수 있다. 반면 이 경우는 양쪽이 동일하게 늘어나서 심라인의 위치 변화는 거의 없다. 세 번째 줄 두 번째 그림에서 심라인 위치가 거의 유사함을 확인할 수 있다. 약간의 기울어짐이 발생하는 이유는 심라인의 위치가 가운데에서 이동했기 보다는 옷의 폭이 늘어났기 때문에 시뮬레이션하는 과정에서 넓어진 티셔츠 폭에 의해서 옷이 처지게 되어서 발생하는 현상이다. 정리해보면 두 번째 줄과 같은 심라인 제약 조건 처리는 폭수정이 아닌 심라인의 위치조정을 위한 의복 수정일 때에 유용하게 사용될 수 있고, 세 번째 줄과 같은 심라인 제약 조건 처리는 전반적인 의복의 폭을 늘이거나 줄일 때 사용할 수 있다.

이러한 심라인 제약 조건 처리가 무조건적으로 이루어져야 하는 것은 아니다. 왜냐하면 경우에 따라서 심라인을 구성하는 두 엣지의 길이가 서로 달라야 하는 경우가 있기 때문

이다.

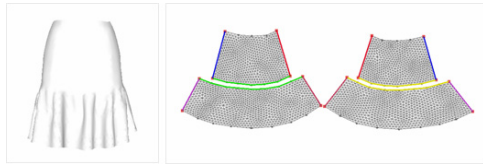


그림 13. Flair Skirt

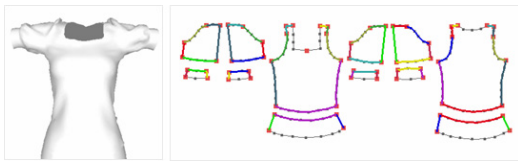


그림 14. Puff Sleeve

그림13과 그림14의 Flair Skirt와 Puff Sleeve 같은 경우 의도적으로 심라인을 구성하는 두 엷지의 길이를 다르게 해서 주름과 볼륨감을 표현한다. 따라서 이와 같은 디자인을 허용하기 위해서는 선택적으로 심라인 제약조건을 처리할 수 있어야한다. 본 시스템에서는 디자이너가 선택적으로 심라인 제약 조건을 적용할 수 있도록 옵션을 제공하고 있다.

2.5. 3D 드레이핑 윈도우에서의 패턴수정

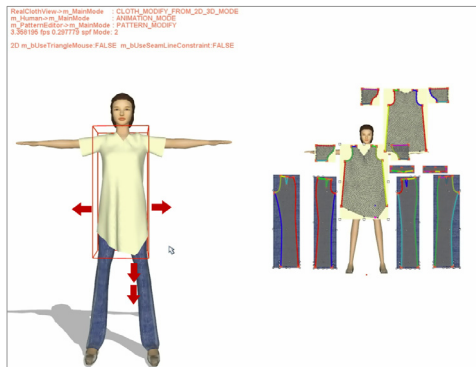


그림 15. 드레이핑 윈도우에서의 가상 의복 수정

본 시스템에서는 드레이핑 윈도우에서 직접 마우스 오퍼레이션을 통해 3D 의복 메쉬와 2D 패턴을 수정할 수 있다. 티셔츠의 밑단을 잡고 당기면 티셔츠 밑단의 실루엣이 변한다. 넥라인을 잡고 드래그해서 V넥을 U넥으로, 혹은 사각형의 넥라인으로 바꿀 수 있다. 좌우 엷지를 잡고 당겨서 원하는 부분이면 어느 곳이나 폭 수정을 할 수 있다. 3D에서의 의복 수정은 길이나 폭 수정, 넥라인 모양 변경에 한정되지

않는다. 원하는 부분이면 어느 곳이나 마우스 오퍼레이션을 통해 디자인을 변경할 수 있다.



그림 16. 사용자 입장에서의 동작원리

3D 드레이핑 윈도우에서의 의복 수정은 실제 걸음으로 보이는 것과 내부적인 구현이 조금 다르게 구성되어있다. 사용자의 입장에서는 위의 그림과 같이 3D 드레이핑 윈도우 상에서 마우스 조작을 하고, 그 결과로 3D 의복 메쉬가 변형되어서 2D 패턴이 따라서 변형되는 것으로 보인다. 하지만 실제 내부적인 구현은 정반대이다.

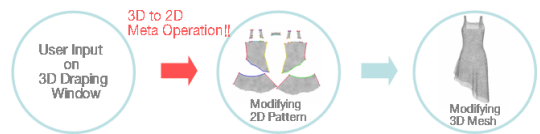


그림 17. 내부적인 구현

실제 내부적인 구현은 위의 그림과 같다. 우선 사용자의 3D 드레이핑 윈도우 상에서의 마우스 조작은 메타 오퍼레이션을 통해 2D 패턴에디터로 전달된다. 전달된 마우스 오퍼레이션에 따라 2D 패턴이 수정되고, 그에 따라 3D 의복 메쉬가 변형되는 것이다.

이와 같이 3D 드레이핑 윈도우에서의 마우스 조작을 굳이 2D 패턴 에디터로 넘겨서 처리하는 이유는 보다 수월하게 의복을 수정할 수 있기 때문이다. 3D 드레이핑 윈도우에서 직접 의복을 수정할 경우, 3차원 벡터스 정보를 직접 다루어야 하기 때문에 2차원에서 벡터를 다루어 의복을 수정하는 것보다 처리가 어렵다. 그리고 의복 수정은 Mass-Spring Model에서 스프링의 Rest Length를 기준으로 하게 되는데, 드레이핑 윈도우의 의복 메쉬는 물리 시뮬레이션 되고 있는 상태이기 때문에 중력과 신체와의 접촉, 그리고 바람 같은 외부적인 힘의 영향을 받아 의복 메쉬가 변형되어 있는 상태여서 적합하지 않다.

이제 3D 드레이핑 윈도우에서의 의복수정에서 가장 핵심

적인 부분인 3D 마우스 오퍼레이션을 2D 패턴에디터로 넘겨주는 부분에 대해 설명하도록 하겠다. 이 과정은 3D-to-2D 메타 오퍼레이션을 통해 이루어지는데 가장 핵심적인 아이디어는 미리 계산되어져 있는 매칭 테이블을 이용하는 것이다. 2D 패턴 메쉬로부터 3D 의복 메쉬가 생성되었기 때문에 2D 패턴 메쉬의 어떤 삼각형이 3D 의복 메쉬의 어떤 삼각형과 매칭되는가 하는 정보가 존재한다. 우리는 이 정보를 이용해 3D 드레이핑 윈도우에서 상에서 클릭된 삼각형과 서로 매칭되는 2D 패턴에디터 상의 삼각형을 구할 수 있다. 마우스에 의해 클릭된 삼각형 내부의 정확한 위치는 선택된 삼각형의 엣지를 기준으로 한 상대 좌표를 이용해 구할 수 있다. 이와 같이 상대 좌표를 이용하는 이유는 드레이핑 윈도우의 3D 의복 메쉬가 시물레이션 되고 있는 상태이기 때문에 같은 삼각형이라고 할지라도 2D 패턴에디터에서 대응되는 삼각형과 그 크기와 형태가 다르기 때문이다.

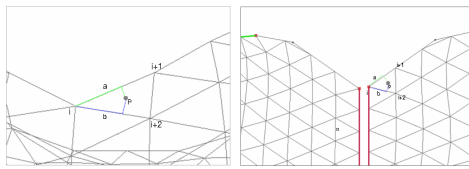


그림 18. 2D-to-3D 메타 오퍼레이션

위의 그림에서 왼쪽은 3D 드레이핑 윈도우이고 오른쪽은 2D 패턴에디터이다. 각각 V백 티셔츠의 백라인 가장 아래쪽을 구성하는 삼각형을 보여주고 있는데, 같은 삼각형임에도 불구하고 선택된 삼각형이 서로 다른 모양을 하고 있음을 확인할 수 있다.

마우스에 의해 클릭된 삼각형 기준의 상대 좌표는 아래와 같은 식에 의해 구해진다.

$$\alpha = a/L_{i,i+1}$$

$L_{i,i+1}$: Edge Length between i ~ $i+1$ th Vertex

$$\beta = b/L_{i,i+2}$$

$L_{i,i+2}$: Edge Length between i ~ $i+2$ th Vertex

그림18에서 P는 마우스에 의해 클릭된 점이고, a는 벡터 i 에서부터 P의 i ~ $i+1$ 엣지 위로의 정사영 위치를 갖는 선분의 길이이다. b는 벡터 i 에서부터 P의 i ~ $i+2$ 엣지 위로의 정사영 위치를 갖는 선분의 길이이다.

4. 결론

이제까지 우리는 기존 가상의복 제작시스템을 개선한 가상의복제작을 위한 인터랙티브한 패턴조각 프로세스에 대해서 알아보았다.

본 시스템에서 개선하고자한 기존 프로세스의 문제점은 우선 패턴에디터에서 패턴을 수정했을 때 바로 그 결과를 드레이핑 윈도우에서 확인 할 수 없었다는 점이었다. 이 문제점은 패턴을 다시 Re-Triangulation하지 않고 적절하게 삼각형의 모양을 바꾸어 변형된 외곽선에 맞는 2D 의복 메쉬를 생성함으로써 해결하였다. 기존 프로세스에서 사용자의 마우스 오퍼레이션이 요구되어 많은 시간을 필요로 했던 3D 상에서의 패턴 배치 역시 Re-Triangulation과정이 없어지면서 다시 반복할 필요가 없게 되었다.

기존 프로세스의 두 번째 문제점은 3D 드레이핑 윈도우로부터의 피드백이 없다는 것이었는데, 이 부분은 3D 드레이핑 윈도우에서 직접 의복수정을 가능하게 하면서 해결하였다. 3D 드레이핑 윈도우에서의 의복수정은 내부적으로 3D-to-2D 메타오퍼레이션을 통해서 구현되는데, 그 핵심 아이디어는 미리 계산된 삼각형 매칭테이블을 이용하는 것이었다.

디자인 프로세스 개선을 위한 첫 번째 아이디어는 템플릿을 사용하는 것이었다. 처음부터 하나하나 패턴을 그려서 의복을 제작하던 기존 프로세스를 스타일별로 분류되어 저장된 가상의복 템플릿을 이용하여 개선하였다. 그리고 전체적인 스타일뿐만 아니라 디테일들까지 템플릿이나 다른 예제에서 불러와 교체할 수 있게 함으로써 보다 세세한 디테일까지 쉽게 디자인을 바꿀 수 있게 하였다.

두 번째 개선 아이디어는 십라인 제약조건을 자동으로 처리해주는 것이었다. 기존 패턴 캐드 시스템에서 벡터의 높이를 기준으로 외곽선을 처리해주는 방법과 달리 한쪽 엣지가 변형되었을 때 사용자의 선택에 따라 십라인으로 연결되어 있는 반대 엣지를 적절하게 수정해줌으로써 보다 디자인 프로세스를 쉽게 개선하였다.



참고문헌

[1] 오승우, 원광연, “시간 임계적 의상 시물레이션을 위한 에너지 보존 기반 다단계 기법”, 한국과학기술원 박사학위논문, 2007

[2] 오승우, 원광연, “적은 감쇠 효과를 가지는 의복 시물레이션을 위한 새로운 암시적 적분법”, 한국 컴퓨터 그래픽스 학회 학술대회 논문집, 2004

[3] 장미정, 원광연, “온라인 의류 쇼핑몰을 위한 가상 가봉 시스템”, 한국과학기술원 석사학위논문 Call No. MCS

04038, 2004

- [4] 오승우, 원광연, “인터랙티브 의상 애니메이션”, 한국과학기술원 석사학위논문, 2002
- [5] Frederic Cordier, Heywon Seo and Nadia Magnenat-Thalmann, “Made-to-Measure Technologies for an Online Clothing Store”, IEEE Computer Graphics and Applications, Jan-Feb, Vol.23, Issue 1, pp.38-48, 2003
- [6] Takeo Igarashi, John.F.Hughes, “Clothing Manipulation”, ACM UIST, 2002
- [7] Emmanuel Turquin, Marie-Paule Cani and John F.Hughes, “Sketching garments for virtual characters”, EUROGRAPHICS workshop on Sketch-based Interface and Model, 2004
- [8] Philippe Decaudin, Dan Julius, Jamie Wither, Laurence Boissieux, Alla Sheffer, Marie-Paule Cani, “Virtual Garments: A Fully Geometric Approach for Clothing Design”, EUROGRAPHICS, 2006
- [9] Thomas Funkhouser, Michael Kazhdan, Philip Shilane, “Modeling by Example”, SIGGRAPH, 2004
- [10] Donald H.House and David E. Breen, “Cloth Modeling and Animation AK”, Peters, 2000
- [11] X. Provot, “Deformation constraints in a mass-spring model to describe rigid cloth behavior”, Proceedings of Graphics Interface, Canadian Computer-Human Communications Society, pp.147-154, 1995
- [12] B. Eberhardt, O. Eitzmub and M. Hauth, “Implicit-explicit schemes for fast animation with particle systems”, Proceedings of the Eurographics Workshop on Computer Animation and Simulation,

Springer-Verlag, 2000

- [13] M. Desbrun, P. Schröder, and A. Barr, “Interactive animation of structured deformable objects”, Proceedings of Graphics Interface, Canadian Computer-Human Communications Society, pp.1-8, 1999
- [14] K.-J. Choi and H.-S. Ko, “Stable but responsive cloth”, ACM Transactions on Graphics (ACM SIGGRAPH 2002) 21, no.3, 604-611, 2002
- [15] R. Bridson, R. Fedkiw, and J. Anderson, “Robust treatment of collisions”, contact and friction for cloth animation. ACM Transactions on Graphics (ACM SIGGRAPH 2002) 21, no.3, 594-603, 2002
- [16] 서승욱, 이재민, “FX in 3D”, 성안당, 2006
- [17] Maya, www.autodesk.com
- [18] 3dsMax, www.autodesk.com
- [19] OptiTex, www.optitex.com
- [20] TukaTech, www.tukatech.com
- [21] Lectra, www.lectra.com
- [22] Gerber Technology, www.gerberetechnology.com
- [23] Pad System, www.padsystem.com
- [24] StyleZone, www.stylezone.com
- [25] Qualoth, www.qualoth.com
- [26] Digital fashion ltd, www.dressingsim.com