
인터랙티브 헤어 스타일링 인터페이스

Interactive Hair Styling Interface

조정현 Junghyun Cho*, 고흥석 Hyeong-Seok Ko**

요약 통계적 방법을 사용하여 머리카락을 생성하고 제약 조건을 이용하여 헤어 스타일을 완성하는 방법에 대한 연구가 논문 [2]에 자세히 소개되어 있다. 논문 [2]의 방법과 함께 제시된 프로그램은 매우 다양한 헤어 스타일을 가능하게 하지만 일반 사용자가 쉽게 다루기 어려운 단점이 있다. 따라서 본 논문에서는 머리카락과 헤어 스타일의 생성에 있어서는 논문 [2]의 방법을 따르되 사용자가 쉽게 헤어 스타일을 조정할 수 있도록 맵 기반 인터랙티브 헤어 스타일링 인터페이스를 제안한다. 머리카락이 생성될 두피 부분을 공간 분할하고 이 부분에 사용자가 머리카락의 밀도, 길이, 색상 맵을 직접 그릴 수 있게 하여 직관적인 헤어 스타일 생성이 가능하게 한다. 복잡한 형태의 표면에 대해서도 효과적으로 적용될 수 있도록 공간을 분할하고 접근하는 과정에 GPU 가속화 방법을 사용한다. 또한 제약 조건을 쉽게 다루기 위해 몇 가지 템플릿을 제공하고 이의 생성과 수정을 쉽게 한다.

Abstract The statistical wisp model for hairstyle generation was introduced in [1]. It provided a program to load human models, set parameters, generate wisps and strands, and make constraints. However, the program used hard-coded human models and prescribed constraints so that it was hard to change different models and manipulate constraints. Hence we provide a simple interface by drawing maps and constraints. Also, we can increase the speed of computation by using GPU acceleration.

핵심어: *Hair modeling; Hairstyle generation; Hair Styling; Statistical wisp model; Interactive user interface; GPU acceleration*

본 논문은 Brain Korea 21 사업의 지원에 의하여 연구되었음

*주저자 : 서울대학교 계산과학전공 박사과정 e-mail: jhcho@graphics.snu.ac.kr

**교신저자 : 서울대학교 전기공학부 교수 e-mail: ko@graphics.snu.ac.kr

1. 서론

사실적인 애니메이션 제작 및 가상 캐릭터 생성에 있어서 헤어 모델링과 시뮬레이션은 필수적인 아이템이다. 따라서 헤어 모델링과 스타일링, 시뮬레이션, 그리고

렌더링에 대한 연구의 동향이 논문 [1]에 폭넓게 개관되었다. 대부분 영상 제작에 필요한 고급 기술로써 모델링과 애니메이션 제작 과정에 익숙한 전문 사용자들을 대상으로 하고 있다.

그 가운데 논문 [2]는 통계적 방법을 활용하여 비교적 간단하고 직관적인 변수들로부터 자연스러운 머리카락과 헤어 스타일을 생성하는 방법을 제시하고 있다. 특히 이 논문에서 머리카락 다발은 머리카락 변형에 관한 특정 제약 조건을 만족하도록 생성되는데, 이를 통해 다수의 머리카락이 서로 상호작용하는 사실적인 효과를 얻을 수 있다. 이 논문의 결과로써 제시된 프로그램은 몇 가지 확률 변수와, 머리카락의 밀도, 길이, 색상 맵, 그리고 제약 조건을 활용하면 매우 다양한 헤어 스타일 생성이 가능함을 보여주고 있다. 하지만 이 프로그램은 많은 사용자 조절 부분이 프로그램 내부에 구현되어 있어 일반 사용자가 사용하기 어려운 단점이 있다.

최근 들어 논문 [3]과 [4]에서 사용자 편의를 고려한 스케치 기반 헤어 스타일링 인터페이스를 제안하였다. 하지만 이 방법들은 논문 [2]에 비해 머리카락 및 헤어스타일 생성에 있어 만족스러운 결과를 보여주지 못하고 있다.

따라서 본 논문에서는 자연스러운 머리카락 및 헤어스타일 생성에 있어 좋은 결과를 보여주고 있는 논문 [2]의 방법을 사용하고, 이와 더불어 사용자의 편의를 고려한 헤어 스타일링 인터페이스를 제안한다.

↓

2. 머리카락 모델링 기법

머리카락 생성은 논문 [2]의 통계적 방법을 사용한다. 기본적으로 머리카락은 묶음 단위로 모델링 되며, 머리카락 묶음의 형태는 길이와, 폭, 흐트러짐 등을 나타내는 변수들에 의해 결정된다. 머리카락 묶음의 중심 가닥은 머리카락 건본으로부터 마코프 체인(Markov Chain) 모델을 기반으로 생성되고, 위에 소개한 변수들을 바탕으로 주변 가닥들이 생성된다.

이렇게 생성된 머리카락 묶음은 제약 조건이 주어진 필드에서 다음의 두 식에 따라 의사 물리적으로 변형된다.

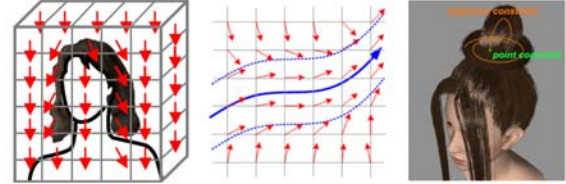
$$E = [(1 - \omega)\Phi(\mathbf{x}) + \omega\Psi(\mathbf{x})] \cdot \mathbf{q} + \kappa\mathbf{q}_0 \cdot \mathbf{q} \quad (1)$$

$$\rho(\mathbf{x}) + \rho_{\Delta}(\mathbf{x}) > \tau \quad (2)$$

여기서 식 (1)은 머리카락의 생성 방향을 결정하기 위해 우리가 최대화시켜야 하는 에너지 식이다. $\Phi(\mathbf{x})$ 는 기본적인 스타일 필드이고 $\Psi(\mathbf{x})$ 는 제약 조건이 가해진 변형 필드이고, ω 와 κ 는 가중치와 강성, \mathbf{q} 와 \mathbf{q}_0 는 계산하고자 하는 머리카락의 방향과 변형 전 방향을 각각 나타낸다. 머리카락의 생성 방향이 필드의 방향과 변형 전

머리카락의 방향과 평행이 될수록 에너지 값이 증가함을 살펴볼 수 있다.

식 (2)는 머리카락의 총돌을 판단하기 위해 사용되는 식이다. $\rho(\mathbf{x})$ 는 어느 한 지점의 변형 전 밀도 값이고, $\rho_{\Delta}(\mathbf{x})$ 는 변형 후 밀도의 변화량 값이다. 따라서 총돌이



감지되면 총돌이 일어나지 않도록 머리카락의 생성 방향을 재설정한다.

그림 1. 스타일 필드, 변형 필드, 활용 예(논문 [2])

위 두 식에서 헤어 스타일 생성에 가장 큰 영향을 미치는 것은 스타일 필드와 변형 필드이다. 그림 1의 활용 예에서 볼 수 있는 바와 같이 이 두 가지 제약 조건을 사용하면 매우 다양한 헤어 스타일 연출이 가능하다. 특히 변형 필드의 경우에는 특성에 따라 점, 경로, 방향 변형 필드로 나눌 수 있고, 우리는 이들을 조작할 수 있는 편리한 헤어 스타일링 환경을 제공할 수 있다. 스타일 필드의 경우에는 3 차원 공간 상의 벡터 필드를 다루는 문제로 본 논문에서는 다루지 않지만 매우 흥미 있는 후속 연구가 될 수 있다.

또한 헤어 스타일 생성에 영향을 주는 것으로 머리카락이 생성될 두피에 지정된 밀도, 길이, 색상 정보가 있다. 논문 [2]에서 이 값들은 기본적으로 2 차원 맵으로 주어지는데 2 차원 맵은 사용자가 다루기 어려운 단점이 있다. 따라서 우리는 이런 부분을 개선하여 보다 편리한 헤어 스타일링 환경을 제공할 수 있다.

3. 헤어 스타일링 인터페이스

3.1 맵 그리기

머리카락의 밀도는 머리를 심고자 하는 두피에 주어진 밀도 맵을 기반으로 결정된다. 밀도 맵은 외부 프로그램을 사용하여 제작하거나, 특정한 경우에 대해 미리 계산된 값을 사용할 수 있다. 하지만 이 경우 모델이 바뀌고 원하는 헤어스타일이 달라질 때마다 번거롭고 복잡한 과정을 반복해서 수행해야만 한다. 따라서 그림 2와 같이 두피 부분을 공간 분할하고 사용자가 직접 밀도 맵을 그릴 수 있는 인터페이스를 제공한다. 이를 통해 맵 제작의 번거로움을 줄이고, 보다 직관적으로 헤어스타일을 생성할 수 있도록 한다. 이는 머리카락의 밀도 맵 뿐만 아니라, 머리카락의 길이 맵과 색상 맵도 같은 방식으로 생성하고 수정할 수 있게 함으로 효율적이다.



그림 2. 공간 분할 및 맵 그리기

3.2 공간 분할 방법

공간 분할의 방법에는 여러 가지가 있으나, 그 중 가장 많이 쓰이고 효과적인 방법으로 8진 트리(Octree)를 꼽을 수 있다. 8진 트리는 필요한 데이터가 들어 있는 옥면체를 8개의 자식 노드로 나누고, 또 그 자식 노드를 원하는 수준까지 재귀적으로 나누어 데이터를 관리하는 자료 구조로 상대적으로 적은 양의 메모리 사용과 빠른 속도의 장점이 있다.

머리카락이 자라날 두피의 위치에 쉽게 접근하고 데이터(색상)를 쉽게 추출하고 변경하기 위해서 통상적으로 CPU에서 8진 트리를 구현할 수 있다. 하지만 본 논문에서는 데이터의 추출과 기록이 가시화에 주로 사용되고, 형태가 복잡한 동물의 털 생성에도 응용될 수 있도록 GPU에서 8진 트리를 구현하였다.

GPU를 사용하여 8진 트리를 구현하는 방법은 [6]을 참조하였다. [6]의 방법을 사용하면 그림 3의 왼쪽과 같은 8진 트리를 그림 3의 오른쪽과 같이 GPU 메모리 상에 만들 수 있다. 그러면 [0,1] 사이의 어떤 지점 M 의 데이터는 다음의 식을 사용하여 접근할 수 있다.

$$p = \frac{I_D + \text{frac}(M \cdot N^D)}{S} \quad (3)$$

여기서 D 는 트리의 깊이이고, I_D 는 GPU 메모리 상에서 해당 깊이의 노드 인덱스 벡터이며, N 은 8진 트리의 경우 2이고 S 는 GPU 메모리 크기 벡터이다. (나눗셈은 벡터 원소 별로 한다.) 그림 3의 경우에는 C가 단말 노드인데 이를 참조하기 위해선 우선 A와 B를 지나야 한다. 예시를 위해 4진 트리와 2차원 맵을 나타냈는데 실제 구현에는 3차원 맵이 사용되었다. 식 (3)과 같이 메모리를 열람하는 GPU 프로그램을 구현하면 저장된 단말 노드의 데이터를 가시화하기 위해서 맵을 바인딩하기만 하면 되므로 효과적이다.

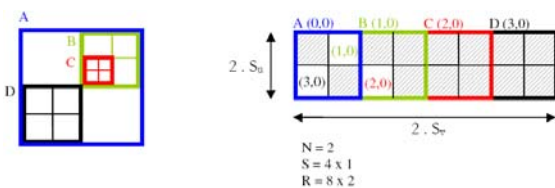


그림 3. GPU 상에서의 8진 트리 생성 예([6] 참조)



그림 4. 맵 그리기(위로부터 밀도, 길이, 색상 맵)

그림 4는 앞서 설명한 방법으로 구현한 프로그램의 간단한 실행 모습이다.

3.3 제약 조건 다루기

논문 [2]에서 헤어 스타일의 제약 조건으로 점과, 경로, 방향의 세가지 종류를 제시하였다. 일반 사용자가 3차원 상에서 이와 같은 제약 조건을 생성하고 수정하는 것은 쉽지 않은 일이다. 따라서 기본적인 제약 조건 템플릿을 제공하고 이것을 복사하고 편집함으로써 원하는 결과를 얻을 수 있도록 인터페이스를 제공한다. 이는 물체를 복사하고 이동하고 회전시킬 수 있도록 도와주는 조정자를 구현함으로써 이루어질 수 있다. 제약 조건은 논문 [2]와 동일하게 적용하였다.

4. 결론

통계적 방법에 의한 헤어 스타일 생성의 효과적인 제어 툴로서 맵 기반의 헤어 스타일링 인터페이스를 제안하였다. 단순하지만 직관적인 방법으로 헤어 스타일 생성의 중요한 부분을 조절할 수 있게 함으로써 사용자의 편의를 높이고자 하였다. 본 연구에서 한걸음 더 나아가 논문 [7]과 같이 보다 효율적인 GPU 라이브러리 구현 방법이나, 스케치 기반의 제약 조건으로부터 벡터 필드를 생성하는 방법, 또 논문 [8]과 같이 캐싱(Caching)이나 GPU를 사용하여 실시간으로 고품질의 머리카락 이미지를 얻는 방법 등을 추가로 생각해 볼 수 있다. 본 논문은 이와 같은 후속 연구의 밑바탕이 될 것이다.

참고문헌

- [1] Kelly Ward, Florence Bertails, Tae-Yong Kim, Stephen R. Marschner, Marie-Paule Cani, Ming C. Lin , " *A Survey on Hair Modeling: Styling, Simulation, and Rendering* " , IEEE Transactions on Visualization and Computer Graphics, 2007
- [2] Byoungwon Choe and Hyeong-Seok Ko., " *A Statistical Wisp Model and Pseudophysical Approaches for Interactive Hairstyle Generation* " , IEEE Transaction on Visualization and Computer Graphics, 2004.
- [3] Hongbo Fu, Yichen Wei, Chiew-Lan Tai, and Long Quan, " *Sketching Hairstyles* " , Eurographics Workshop on Sketch-Based Interfaces and Modeling, 2007.
- [4] Jamie Wither, Florence Bertails, and Marie-Paule Cani, " *Realistic Hair from a Sketch* " , 2008.
- [5] Sylvain Lefebvre, Samuel Hornus, and Fabrice Neyret, " *Octree Textures on the GPU* " , GPU Gems 2, 2005
- [7] Aaron E. Lefohn, Joe Kniss, Robert Strzodka, Shubhabrata Sengupta, and John D. Owens, " *Glift: Generic, Efficient, Random-Access GPU Data Structures* " , ACM Transactions on Graphics, 2006
- [8] Arno Zinke, Cem Yuksel, Andreas Weber, and John Keyser, " *Dual Scattering Approximation for Fast Multiple Scattering in Hair* " , ACM SIGGRAPH, 2008