

## 3차원 아바타의 실시간 얼굴표정제어를 위한 표정공간의 생성방법

김 성 호

상지대학교 컴퓨터정보공학부  
kimsh1204@sangji.ac.kr

### Generation Method of Expression Space for Realtime Facial Expression Control of 3D Avatar

Sung-Ho Kim

School of Computer, Information and Communication Engineering, Sangji Univ.

#### 요 약

본 논문은 애니메이션으로 하여금 얼굴 표정들의 공간으로부터 일련의 표정을 선택하게 함으로써 3차원 아바타의 표정을 실시간적으로 제어할 수 있도록 하기 위한 표정공간의 생성방법에 관하여 기술한다. 본 시스템에서는 약 2400여개의 얼굴 표정 프레임을 이용하여 표정공간을 구성하였다. 본 기법에서는 한 표정을 표시하는 상태표현으로 얼굴특징 점들 간의 상호거리를 표시하는 거리행렬을 사용한다. 이 거리행렬의 집합을 표정공간으로 한다. 그러나 이 표정공간은 한 표정이 다른 표정까지 이동할 때 두 표정간의 직선경로를 통해 이동할 수 있는 그런 공간이 아니다. 본 기법에서는 한 표정에서 다른 표정까지 거쳐 갈 수 있는 경로를 캡처된 표정 데이터로부터 근사적으로 유추한다. 우선, 각 표정상태를 표현하는 거리행렬간의 거리가 일정 값 이하인 경우 두 표정을 인접해 있다고 간주한다. 임의의 두 표정 상태가 일련의 인접표정들의 집합으로 연결되어 있으면 두 표정 간에 경로가 존재한다고 간주한다. 한 표정에서 다른 표정으로 변화할 때 두 표정간의 최단경로를 통해 이동한다고 가정한다. 두 표정간의 최단거리를 구하기 위해 다이내믹 프로그래밍 기법을 이용한다. 이 거리행렬의 집합인 표정공간은 다차원 공간이다. 3차원 아바타의 얼굴 표정 제어는 애니메이션이 표정공간을 항해할 때 실시간적으로 수행된다. 이를 도와주기 위해 표정공간을 차원 스케일링 기법을 이용하여 2차원 공간으로 가시화하였고, 애니메이션으로 하여금 본 시스템을 사용하여 실시간 표정 제어를 수행하게 했는데, 본 논문은 그 결과를 평가한다.

#### 1. 서 론

인간은 자신의 감정을 말보다는 얼굴 표정을 통해서 더 잘 표현하며, 상대방의 얼굴 표정을 보고 상대방의 현재 감정 상태를 파악한다. 그런 연유로 지금까지 3차원 컴퓨터 그래픽스 기법을 통해서 인간의 얼굴 표정을 표현할 수 있는 많은 방법이 연구(1, 2, 3)되었다. 최근에는 모션 캡처를 사용한 캐릭터 애니메이션이 컴퓨터 애니메이션 분야에서 각광받으면서 배우의 얼굴 모션을 캡처하여 3차원 캐릭터 애니메이션에 적용하고 있다. 캡처한 데이터를 사용하는 방법으로 크게 두 가지가 제시되었다. 첫째는 배우의 모션 데이터를 새로운 모델에 재적용하는 모션 리타겟팅 기법(3,4,5)이다. 모션 리타겟팅 기법은 먼저 몸동작에 대한 것이 제시[4,5]되었고, 얼굴 표정에 대한 것도 제시(3)되었다. 둘째는 배우의 동작을 가능한 한 많이 캡처하여 이 동작 속에 들어있는 자세들을 획득, 자세 데이터베이스를 만든 후, 애니메이션이 특정 자세들을 선택, 연결하여 새로운 동작을 생성하는 기법이다. 이 기법도 몸동작에 대한 것이 먼저 제시(6)되었는데, 아직 얼굴 표정에 대한 것은 제시된 것이 없다. 그러므로 본 논문에서는 다량의 얼굴 모션 캡처 데이터를 직관적인 공간에 분포시키고, 애니메이션이 원하는 얼굴 표정들을 선택하여, 실시간적으로 얼굴 표정을 제어하기 위한 표정공간의 생성 방법을 기술한다.

먼저 얼굴 표정을 전문적으로 연출하는 배우의 도움을 받아 그림 1과 같이 광학식 모션 캡처 시스템을 사용하여 얼굴 표정을 캡처한다. 표정을 캡처할 때, 배우는 얼굴 주 근육 부분에 작은 반사 마커 100개를 부착한다. 그런 다음 배우로 하여금 서로 다른 10개의 얼굴 모션을 연출하게 하고 초당 60 프레임으로 캡처한다. 모든 얼굴 모션은 처음 무표정 상태에서 출발하여 특정 표정까지 진행한 다음, 다시 무표정 상태로 되돌아오는 방법으로 진행한다. 한 개의 마커는 3개의 좌표 값으로 표현되므로 하나의 표정은 300차원의 데이터이다.



그림 1 : 액터의 얼굴에 부착된 마커 100개 부착위치 및 광학식 모션캡처 시스템을 사용한 얼굴 모션캡처 장면

얼굴 표정 데이터에서 임의의 두 표정간의 최단경로를 결정함으로써 표정공간을 생성하고, 이 공간을 가시적으로 확인하기 위해 다차원 스케일링(Multidimensional Scaling, MDS)(7,8) 기법으로 2차원으로 투영시킨다. 실시간 얼굴표정제어를 위해서는 애니메이션으로 하여금 투영된 2차원 공간을 항해하게 하고, 항해경로의 각 점에 해당되는 3차원 얼굴 모델을 실시간으로 디스플레이 한다.

#### 2. 얼굴 표정상태 표현법

표정공간을 생성하기 위해서는 각각의 얼굴 표정상태를 수치적으로 표현해야 한다. 표정상태는 얼굴에 부착된 마커위치들

에 의해 결정된다. 표정상태의 표현은 표정들 간의 상대적인 거리 관계를 잘 표현하는 것이어야 한다. 표정상태를 표현하는 가장 간단한 방법은 마커 위치들로 이루어진 상태벡터를 이용하는 것이다. 본 논문에서는 100개의 마커를 사용하고 한 마커는 3개의 좌표를 가지기 때문에, 표정상태벡터는 300차원이 된다. 이런 식으로 표정상태를 표현하는 방법을 '위치벡터'라고 하자. 그러나 본 논문에서는 표정상태를 표현할 때, 위치벡터방식을 사용하지 않는다. 대신 임의의 두 마커간의 상호거리를 표현하는 '거리행렬' 방식을 이용하여 표정상태를 표현한다. 왜냐하면 거리행렬이 위치벡터보다 얼굴 마커들의 분포상태에 대한 정보를 더 많이 표시하고 있고, 따라서 두 표정간의 거리를 보다 더 정확하게 표현할 수 있기 때문이다. 두 위치벡터를 비교할 때, 각 마커의 위치들 간의 거리를 구함으로써 두 표정간의 거리를 표현한다. 이때 각 마커의 표정간의 거리는 다른 마커들과 관계없이 계산한다. 이에 비해, 거리행렬로 표정상태를 표현하면, 두 표정간의 거리를 구할 때 마커들 간의 상호관계가 자연스럽게 고려된다. 물론 표정의 거리행렬은 표정의 위치벡터로부터 구할 수 있지만, 위치벡터를 사용하는 경우, 이 위치벡터에 내포되어 있는 정보를 상대간의 거리를 구하거나 할 때 사용하지 않기 때문에, 유용한 정보를 내포하고 있다는 것이 도움이 되지 않는다. 따라서 이 정보를 명시적으로 표현하는 거리행렬 방식이 더 좋은 상태표현방식이다.

### 3. 표정공간의 생성

표정상태를 거리행렬로 표현하면 표정간의 직선거리는 두 거리행렬간의 직선거리로 표현된다. 거리행렬로 표현된 표정상태들의 공간은 임의의 두 거리행렬간의 거리를 결정함으로써 결정된다. 본 논문에서는 거리행렬을 하나의 벡터로 보고, 이 벡터간의 직선거리를 거리행렬간의 직선거리로 사용한다. 즉 임의의 두 마커간의 거리가 서로 비슷한 두 표정은 서로 인접한 표정으로 간주한다. 표정공간은 임의의 두 표정간의 거리를 두 표정거리 행렬간의 직선거리로 정의할 수 있는 벡터공간이 아니다. 한 표정에서 다른 표정으로 옮겨가는 과정은 얼굴의 여러 가지 제약조건으로 말미암아, 복잡한 경로를 거치게 되기 때문이다. 표정공간은 구면과 같은 다양체(Manifold) 공간인 것이다. 다양체 공간상에서의 거리는 두 점간의 거리를 한 점에서 이 공간을 벗어나지 않으면서 다른 점까지 도달하는 최단경로의 길이로 정의(9)한다. 본 논문에서는 이 다양체 공간을 근사적으로 표현한다. 이를 위해 먼저, 두 거리행렬간의 직선거리가 일정 값 이하인 경우 이 직선거리가 두 표정간의 최단거리에 대한 근사치라고 간주한다. 이 조건을 만족하는 두 표정을 "인접표정"이라고 하는데, 임의의 표정에 대한 인접표정들은 그림 2에서 보는 것처럼 결정한다. 인접표정이 주어지면 한 표정에서 다른 표정까지 바로 이동할 수 있다고 본다. 두 표정이 인접해 있지 않은 경우에 한 표정에서 다른 표정으로 바로 이동할 수 없고, 그 사이에 있는 인접한 표정들을 통해서만 이동할 수 있다고 가정한다. 그러나 인접표정을 결정하는 한계거리를 미리 알기는 쉽지 않다. 따라서 이는 실험에 의해 최적의 값을 결정해야 한다. 다만, 인접거리 임계값은 임의의 표정에서 다른 임의의 표정으로 이동하는 데 필요한 충분한 수의 인접표정들이 나오도록 설정되어야 한다. 인접표정들이 결정되면, 인접하지 않은 두 표정상태간의 거리는 그 사이에 있는 인접 표정들 간의 거리들을 합하여 구한다. 이를 위해, 최단거리를 구하는 알고리즘인 플로이드(Floyd) 알고리즘(다이나믹 프로그래밍 기법) [10]을 이용한다. 이렇게 임의의 두 표정간의 최단거리가 구해지면 해당 다양체 공간이 결정된다. 본 논문에서는 모두 2400여 개의 얼굴표정을 사용하여 다양체 공간을 형성하였다. 그러나 이 공간을 애니메이션이 행해지면서 원하는 표정을 선택할 수는 없다. 그러므로 원래 표정공간의 구조를 근

사적으로 표현하는 2차원 또는 3차원으로 공간을 구하여 이 공간을 항해하는 방법을 사용한다.

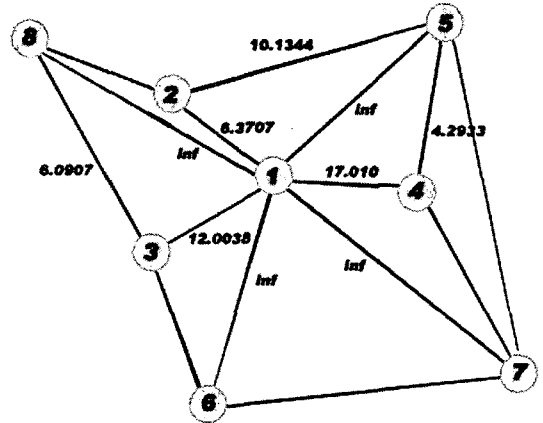


그림 2 : 플로이드(Floyd) 알고리즘을 위한 그래프 생성방법 : 인접표정간의 거리가 주어진 8개의 상태를 가정, 1번 상태를 기준으로 2, 3, 4번 상태는 1번 상태와 인접, 나머지는 그렇지 않다. 인접표정이 아닌 두 상태간의 거리는 무한대(inf)로 주어져 있는데, 이것은 한 상태에서 다른 상태까지 바로는 이동할 수 없다는 것을 의미한다. 원의 반지름은 임의의 인접거리 한계 값을 나타낸다.

본 논문에서는 2차원 공간을 생성하고 이를 위해 MDS[7, 8]를 사용한다. MDS는 고차원 데이터들 사이의 거리가 주어지면, 이 거리들의 분포를 대변하는 좌표들의 집합을 구하는 방법이다. 이때 이 좌표들의 차원은 필요에 따라 미리 정하는데, 본 연구에서는 구한 좌표들을 시각적으로 표현해야 되기 때문에, 2차원 좌표를 사용했다.



그림 3 : 사용자 인터페이스(좌: 항해공간, 우: 3차원 얼굴 모델). 항해공간에 분포된 각 얼굴 표정상태를 대표하는 작은 폴리곤을 마우스로 선택하면, 선택된 점에 해당되는 3차원 얼굴 모델이 실시간적으로 보여 줌으로써 실시간 표정제어가 가능해진다.

### 4. 실험 및 검증

2400여개의 표정으로 구성된 표정공간을 생성한 후에 이를 MDS를 통해 2차원 공간으로 투영하고, 애니메이션으로 하여금 이 공간을 항해하면서 3차원 아바타의 얼굴표정을 실시간적으

로 제어하였다. 이를 위한 사용자 인터페이스는 그림 3과 같다. 사용자 인터페이스에서 항해공간의 점들의 분포는 무표정 상태인 한 점을 중심으로 방사형 분포를 이루고 있다. 이것은 10개의 서로 다른 모션 데이터들이 동일한 무표정 상태에서 출발하여 특정 표정으로 진행해갈수록 서로 다른 모션 데이터에 속한 표정들 사이의 거리가 멀어지기 때문이다.

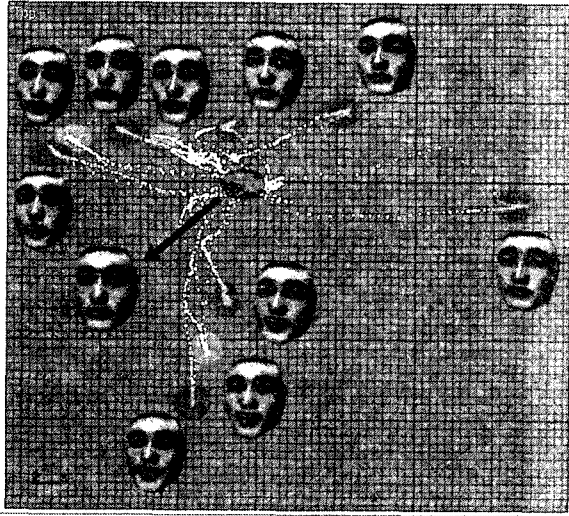


그림 4 : 애니메이터가 모든 2차원 표정공간을 항해하면서 마우스로 선택한 경로를 상에 있는 대표적인 얼굴표정들을 표시함. (원래 얼굴모델은 별도의 창에 표시되지만, 설명의 편의를 위해 항해공간에 표시함)

거리행렬 방식에 의해 표현된 표정상태를 사용하여 최적의 표정공간을 생성하기 위한 실험에서는 인접표정을 결정하는 인접거리 임계값을 실험에 의해 최적의 값을 결정해야 한다. 본 논문에서는 임계값을 정할 때, 다음과 같은 두 가지 기준을 적용하였다. 첫째, 인접거리 임계값에 따라 300차원 다양체 공간에서의 표정 최단거리 분포와 2차원 공간에서의 표정 최단거리 분포 사이의 상관도가 달라진다. 이때 이왕이면 두 분포 사이의 상관도가 높게 나오는 인접거리 임계값을 쓰는 것이 좋다. 왜냐하면 임의의 두 표정사이에 인접표정들을 거쳐서 가는 최단 경로를 구할 때, 최단경로가 존재하지 않는 경우가 많아 생기기 때문이다. 둘째, 상관도를 높게 만드는 임계값들 중에서 보다 안정적인 임계값을 사용하는 것이 좋다.

첫째 기준을 적용하기 위하여 다차원 최단거리 분포와 2차원 최단거리 분포사이의 상관계수는 피어슨(Pearson)의 상관계수,  $r[11]$ 을 이용하여 구하였다. 실험 결과 가장 높은 상관관계는  $r=0.96470$ 이며, 이때의 인접거리 임계값은 230mm이다. 둘째 기준을 적용하기 위하여 두 공간간의 가장 높은 상관계수를 가지는 인접거리 임계값 230mm 주변 임계값(220mm에서 240mm까지)을 조금씩 변경하면서 2차원 최단 거리 분포를 생성하고, 차례대로 디스플레이하면서 거리분포의 변화를 확인하였다. 그 결과 인접거리 임계값이 220mm에서 230mm까지 증가할 때 최단거리 분포에서 인접 표정들 사이의 거리가 거의 유사하거나 미세한 차이로 점점 가까워짐을 확인할 수 있었다. 또한 인접거리 임계값이 230mm에서 240mm까지 증가할 때, 인접표정들 사이의 거리가 거의 유사하거나 매우 미세하게 멀어지는 것을 확인하였다. 이것은 인접거리 임계값 230mm를 전후하여 2차원 거리 분포도가 크게 변하지 않는다는 것이다.

따라서 안정적인 임계값들 중에서 두 공간간의 상관도를 가장 높게 만드는 임계값 230mm를 사용하기로 하였으며, 이를 사용하여 3차원 아바타의 얼굴표정을 실시간적으로 제어하기 위한 최적의 표정공간을 생성하였다. 그림 4는 이 인접거리를 이용하여 구성한 2차원 최단 거리 분포이다.

## 5. 결론

본 논문에서는 다량의 얼굴 모션 데이터들을 적당한 공간에 분포시키고, 애니메이터가 이 공간을 항해하면서 원하는 얼굴 표정들을 선택하여 3차원 아바타의 얼굴 표정을 실시간적으로 제어하기 위한 표정공간의 생성방법에 대하여 기술하였다. 2400여 개의 얼굴 표정 프레임을 사용하여 항해공간을 구성하였으며, 이를 구성하기 위해서 임의의 두 얼굴 표정간의 최단 거리인 다양체 거리를 계산하였다. 3차원 아바타의 얼굴 표정을 실시간으로 제어하기 위해서 개발한 사용자 인터페이스는 최적의 다양체 거리 계산방법 즉, 표정공간의 생성방법을 실험적으로 확인하는데 매우 유용하게 사용되었다. 또한 애니메이터로 하여금 본 논문에서 개발한 사용자 인터페이스를 사용하여 3차원 아바타의 얼굴 표정을 실시간으로 제어하도록 하게 해봄으로써 표정공간이 얼마나 잘 생성되었는지 확인할 수도 있었다.

## 참고문헌

- [1] Frederic I. Parke, Keith Waters. "Computer facial animation," A K Peters, 1996
- [2] Won-Sook Lee, Nadia Magnenat-Thalmann, "Fast head modeling for animation," Journal Image and Vision Computing, Volume 18, Number 4, pp.355-364, Elsevier, 1 March, 2000
- [3] Cyriaque Kouadio, Pierre Poulin, and Pierre Lachapelle, "Real-time facial animation based upon a bank of 3D facial expressions," Proc. Computer Animation 98, June 1998
- [4] Michael Gleicher, "Retargetting motion to new characters," Proceedings of SIGGRAPH 98. In Computer Graphics Annual Conference Series, 1998
- [5] Kwangjin Choi and Hyeongseok Ko. "On-line motion retargetting," In Proceedings of the International Pacific Graphics '99, 1999
- [6] Jehée Lee, Jinxiang Chai, Paul Reitsma, Jessica Hodgins, and Nancy Pollard, Interactive Control of Avatars Animated with Human Motion Data, ACM Transactions on Graphics (SIGGRAPH 2002), volume 21, number 3, 491-500, July 2002
- [7] W. S. Torgerson. Multidimensional scaling: I. theory and method. Psychometrica., 17:401-419, 1952
- [8] T. Cox and M. Cox. "Multidimensional Scaling". Chapman & Hall, London, 1994
- [9] J. Tenenbaum. "Mapping a manifold of perceptual observations". In Advances in Neural Information Processing Systems, volume 10, pages 682--688. MIT Press, 1998
- [10] R. W. Floyd, "Algorithm 97 : Shortest Path," CACM Vol.5, pp. 345, 1962
- [11] Uprendra Shardanand, "Social information filtering for music recommendation", Master's thesis, MIT, 1994