

# 모션 데이터의 PCA 투영에 의한 3차원 아바타의 실시간 표정 제어

김 성 호<sup>†</sup>

## 요 약

본 논문은 사용자로 하여금 얼굴표정들의 공간으로부터 일련의 표정을 실시간적으로 선택하게 함으로써 3차원 아바타의 실시간적 얼굴 표정을 제어하는 기법을 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. 3차원 아바타의 얼굴 표정은 사용자가 표정공간을 향해하면서 실시간적으로 제어한다. 이를 도와주기 위해 표정공간을 PCA 투영 기법을 이용하여 2차원 공간으로 가시화했다. 본 시스템이 어떤 효과가 있는지를 알기 위해 사용자들로 하여금 본 시스템을 사용하여 3차원 아바타의 얼굴 표정을 제어하게 했는데, 본 논문은 그 결과를 평가한다.

## Realtime Facial Expression Control of 3D Avatar by PCA Projection of Motion Data

Sung-Ho Kim<sup>†</sup>

## ABSTRACT

This paper presents a method that controls facial expression in realtime of 3D avatar by having the user select a sequence of facial expressions in the space of facial expressions. The space of expression is created from about 2400 frames of facial expressions. To represent the state of each expression, we use the distance matrix that represents the distances between pairs of feature points on the face. The set of distance matrices is used as the space of expressions. Facial expression of 3D avatar is controled in real time as the user navigates the space. To help this process, we visualized the space of expressions in 2D space by using the Principal Component Analysis(PCA) projection. To see how effective this system is, we had users control facial expressions of 3D avatar by using the system. This paper evaluates the results.

**Key words:** Facial Motion Capture(얼굴 모션 캡처), Realtime Facial Expression Control(실시간 얼굴 표정 제어), Facial State Vector(표정상태벡터), Distance Matrix(거리행렬), PCA Projection (PCA 투영)

## 1. 서 론

인간은 자신의 감정을 말보다는 얼굴 표정을 통해

※ 교신저자(Corresponding Author): 김성호, 주소: 서울시 중구 예장동 8-3(100-751), 전화: (02)872-1322, E-mail: kimsh1204@hotmail.com

접수일: 2004년 1월 5일, 완료일: 2004년 5월 3일

<sup>†</sup> 정회원, 송의여대 정보통신계열 멀티미디어콘텐츠전공 겸임교수

서 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[1-6]되었다. 특히 인터넷의 보급이 활발해지고 멀티미디어의 기능이 고급화됨으로써 인터넷과 같은 가상공간에서 사용자 자신을 대신하는 3차원 아바타가 등장하게 되었으며, 이제는 많은 멀티미디어 분야에서

사용되고 있다. 또한 가상공간에서 3차원 아바타를 표현할 때 가장 관심을 가지는 것은 역시 3차원 아바타의 얼굴 표정이다. 왜냐하면 인간은 표정의 변화를 보고 가장 빠르게 상대방의 감정을 파악하고, 아울러 자신의 감정을 실시간으로 상대방에게 보여주고 싶어 하기 때문이다. 이에 가상공간에서 3차원 아바타의 표정을 실시간적으로 사용자가 제어할 수 있는 기법이 필요하다. [6]에서는 3차원 아바타의 표정을 제어하는 기법을 제안하였는데, 서로 다른 6개의 표정을 3차원 얼굴 모델을 사용하여 표현하고 표정 합성을 위한 파라미터 공간을 형성한다. 파라미터 공간은 각 표정들을 키 프레임으로 설정하고, 각 키 프레임들을 보간하여 다양한 표정들을 생성할 수 있도록 구성하였다. 그리고 사용자가 미리 준비된 Cubic 스플라인(Spline) 곡선을 파라미터 공간에서 임의로 변경하고 재생하면 보간법에 의하여 새로운 표정이 생성되게 된다. 그러나 이 기법에서는 사용자가 파라미터 공간에서 스플라인 곡선을 생성할 때 실시간적으로 표정이 보여지지 않는다. 그런 이유로 본 논문의 목적인 가상공간에서 3차원 아바타의 표정을 실시간으로 제어하기 위한 조건을 만족하지 못한다.

그러므로 3차원 아바타의 표정을 실시간적으로 제어하기 위한 새로운 방법이 필요하며, 본 논문에서는 실시간적으로 자연스러운 표정을 제어하는 방법으로 모션 캡처 기법을 제안한다. 모션 캡처 데이터를 사용한 표정 제어는 배우의 표정을 가능한 많이 캡처하여 표정 데이터베이스를 만든 후, 사용자가 특정 표정들을 실시간적으로 선택하여 보여주는 것이다. 그러므로 본 논문에서는 다량의 얼굴 모션 캡처 데이터를 직관적인 공간에 분포시키고, 사용자가 적당한 공간을 향해하면서 원하는 얼굴 표정들을 실시간으로 선택하여, 3차원 아바타의 얼굴 표정을 실시간으로 제어하는 방법을 기술한다.

얼굴 표정 데이터를 사용한 3차원 아바타의 표정 제어를 위해서는 다음과 같은 사전 작업이 필요하다. 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 수개의 얼굴 모션을 연출하게 하고, 초당 60 프레임으로 캡처한다. 한 개의 마커는 3차원 좌표 값으로 표현되므로 100개의 마커 위치로 표현되는 한, 표정은 300차원의 데이터

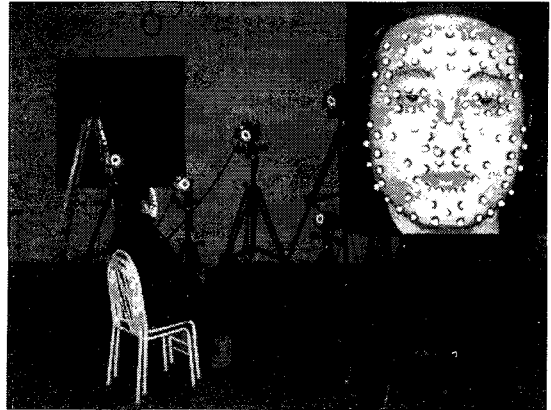


그림 1. 마커 100개 부착위치 및 광학식 모션 캡처 시스템에 사용한 얼굴 모션 캡처 장면

이다. 얼굴 표정 데이터는 3차원 공간상의 위치 값으로만 구성되어 있기 때문에, 얼굴 표정들 사이의 유사성을 수치적으로 표현하기가 어렵다. 얼굴 표정들 사이의 유사성은 300차원의 데이터를 2차원이나 3차원과 같은 저차원 평면상에 분포시킬 경우, 얼굴 표정들 사이의 거리 값으로 구분하기 때문에 필요하다. 그러므로 다양한 얼굴 표정들을 잘 구분할 수 있도록 하기 위해서 마커들 사이의 거리를 이용한 벡터 행렬로 변경하고, 이를 Principal Component Analysis(이하 'PCA') 투영 기법[7-9]으로 2차원에 투영시킨다. 실시간 표정 제어를 위해서는 사용자로 하여금 투영된 2차원 공간을 향해하게 하고, 향해경로의 각 점에 해당되는 얼굴 표정을 3차원 얼굴 모델을 사용하여 디스플레이 한다.

## 2. 얼굴 표정상태 표현법

표정공간을 생성하기 위해서는 각각의 얼굴 표정상태를 수치적으로 표현해야 한다. 표정상태는 얼굴에 부착된 마커들의 위치에 의해서 결정된다. 표정상태의 표현은 표정들 간의 상대적인 거리 관계를 잘 표현하는 것이어야 하고, 이로써 얼굴 표정을 구분할 수 있어야 한다. 표정의 상태를 표현하는 가장 간단한 방법은 마커들의 위치들로 이루어진 상태벡터를 이용하는 것이다. 본 논문에서 사용한 얼굴 모션 프레임 데이터는 100개의 마커를 사용하고, 한 마커는 3차원 좌표를 가지기 때문에, 표정상태벡터는 300차원이 된다. 이런 식으로 표현된 표정상태벡터를 '위치벡터'라고 하자. 본 논문에서는 표정상태를 표현할

때, 위치벡터를 사용하지 않는다. 대신 위치벡터의 임의의 두 마커간의 거리를 표현하는 “거리행렬”을 이용하여 표정상태를 표현한다. 왜냐하면 거리행렬이 위치벡터보다 얼굴 마커들의 분포상태에 대한 정보를 더 많이 표시하고 있기 때문이다. 또한 두 표정이 서로 다르다는 것을 시각적으로 정확하게 비교할 수 있기 때문이다. 따라서 이 정보를 명시적으로 표현하는 거리행렬 방식이 더 좋은 표정상태 표현방식이다.

거리행렬은 위치벡터를 기반으로 다음과 같이 구한다. 즉, 하나의 위치벡터에서 마커와 마커들 사이의 거리를 식 (1)과 같이 직선거리 방법으로 계산한다.

$$D_{ii,c} = \sum_{i=1}^n \left( \sum_{j=1}^{99} \sum_{k=j+1}^{100} \sqrt{(M_{i,j:l2} - M_{i,k:l2})^2} \right) \quad (1)$$

여기서  $M$ 은 위치벡터들의 집합,  $D$ 는 거리행렬들의 집합,  $i$ 는 2400여개의 위치벡터의 수,  $ii$ 는 2400여개의 거리행렬의 수를 의미한다.  $j_1, j_2$  및  $k_1, k_2$ 는 모션 프레임 데이터에서 프레임별 각 마커들의 3차원 좌표 값으로서 마커와 마커 사이의 거리를 구하기 위해서 사용된다.  $i$ 는 위치벡터별 100개 마커에서 마커와 마커 사이의 거리를 계산하기 위한 기준 마커를 의미한다.  $k$ 는 위치벡터별 100개 마커  $j$ 에 의해서 비교되어지는 마커를 의미한다.  $c$ 는 마커 100개에 대한 마커들 사이의 거리 계산 회수로서  $100 \times (100-1)/2=4950$ 을 의미한다.

### 3. PCA 투영

앞에서 생성한 거리행렬 방식의 표정상태 공간은 300차원 공간이다. 따라서 이 공간을 사용자가 행해 하면서 원하는 표정을 선택할 수는 없다. 그러므로 원래 표정공간의 구조를 근사적으로 표현하는 2차원 또는 3차원으로 공간을 구하여 이 공간을 향해하는 방법을 사용한다. 본 논문에서는 2차원 공간을 생성한다. 그리고 이를 위해 PCA 투영기법을 사용한다. PCA는 Unsupervised 알고리즘으로서, 입력 데이터의 모든 선형 변환으로부터 직교하는 특징 정보를 선택하는 방법을 사용한다. 데이터는 이러한 특징 벡터들 상의 고차원 데이터들의 투영으로 나타난다. 이때 투영된 좌표들의 차원은 필요에 따라 미리 정하는데, 본 연구에서는 구한 좌표들을 시각적으로 표현해

야 되기 때문에, 2차원 좌표를 사용한다.

표정상태를 나타내는  $n$ 개의 다차원 공간상의 점들을  $x_i, i=1, \dots, n$ 라 하고, 이를 집합  $\{x_i\}$ 로 표시한다. 그리고 다차원 표정공간을 근사적으로 나타내는 2차원 평면상의 점들을  $y_i, i=1, \dots, n$ 라 하고, 집합  $\{y_i\}$ 로 표시한다. 다차원 공간상의 점들의 집합  $\{x_i\}$ 에 PCA 투영기법을 적용하면, 집합  $\{x_i\}$ 가 2차원 평면에 투영되면서 집합  $\{x_i\}$ 와 가장 근사한 분포를 가지도록 하는 2차원 평면상의 점들의 집합  $\{y_i\}$ 를 구할 수 있다. 집합  $\{x_i\}$ 와 집합  $\{y_i\}$  사이의 거리의 Sum squared error를 최소화하는 PCA 투영의 최소화 함수는 식 (2)와 같다. PCA는 집합  $\{x_i\}$ 가 입력으로 주어졌을 때, 식 (2)를 최소화하는 벡터  $V$ 을 찾는다.

$$PCA(V) = \frac{1}{2} \sum_{i=1}^n |x_i - \mu|^2 \quad (2)$$

여기서  $\mu = \frac{1}{n} \sum_{i=1}^n x_i$ 이다. 식 (2)를 최소화하는 벡터  $V$ 을 구하고, 이를 이용하여 원하는 집합  $\{y_i\}$ 를 구하는 것은 잘 개발된 이론에 근거하고 있다. 이 이론은 Hotelling[13]과 Pearson[14]에 의해서 개발되었으며, 다음과 같이 세 부분으로 이루어져 있다. 첫째는 집합  $\{x_i\}$ 을 이용하여 공분산을 구하는 것이고, 둘째는 공분산 결과로부터 고유치(Eigenvalue)와 고유벡터(Eigenvector)를 찾는다. 그리고 셋째는 이로부터 집합  $\{y_i\}$ 를 찾는 방법이다.

집합  $\{x_i\}$ 을 입력으로 이용한 공분산 행렬  $C$ 는 다음 식 (3)과 같이 계산한다.

$$C = \frac{1}{n-1} \sum_{i=1}^n (x_i - \mu)(x_i - \mu)^T \quad (3)$$

그리고 공분산 행렬  $C$ 의 고유분석(Eigen-analysis)을 통하여,  $C$ 의 고유치와 고유벡터를 구한다. 이는 다음 식 (4)를 해결함으로써 구한다.

$$(C - \lambda I)v_i = 0, i=1, \dots, d \quad (4)$$

여기서  $I$ 는 항등행렬,  $\lambda$ 는 고유치,  $v_i$ 는 고유벡터이다. 그리고 구해진 고유치로부터  $d$ 개의 가장 큰 고유치  $\lambda_1, \dots, \lambda_d$ 을 선택한다. 또한  $d$ 개의 고유치에 대응하는 고유벡터를 모아서 행렬  $V^* = [v_1, \dots, v_d]$ 으로 표시한다. 마지막으로 집합  $\{x_i\}$

에 행렬  $V^*$ 의 열벡터를 다음 식 (5)와 같이 적용하여  $m$ 차원 평면에 투영한 집합  $\{y_i\}$ 를 구한다.

$$y_i = V^{*T} x_i \quad (5)$$

그림 2는 거리벡터인 표정상태벡터들의 집합을 PCA 투영에 적용하여 2차원 평면에 투영한 결과이다.

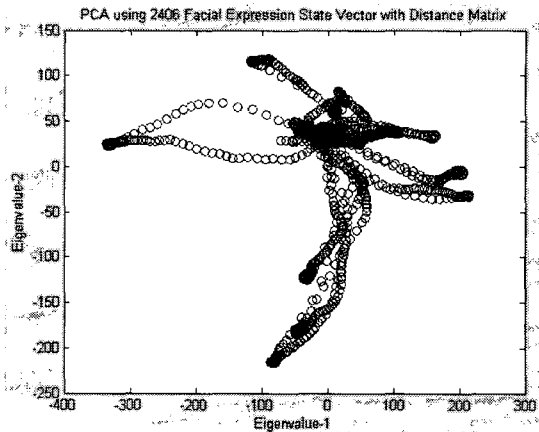


그림 2. 약 2400여개의 거리벡터인 표정상태벡터를 PCA 투영기법에 적용하여 2차원 평면에 투영한 결과

그림 2의 가로축은 공분산 행렬  $C$ 로부터 계산된 가장 큰 고유치  $v_1$ 에 의해서 구해진  $y_{i,1}$ 의 값이고, 세로축은 두 번째로 큰 고유치  $v_2$ 에 의해서 구해진  $y_{i,2}$ 의 값이다. 본 논문에서는 그림 2의 투영된 결과를 실시간 얼굴 표정 제어에 각각 사용하여 실험하고 확인한다.

본 논문에서는 2차원 평면상의 점들의 집합  $\{y_i\}$ 를 구하기 위해 Matlab V6.5를 사용하였다. Matlab에서 PCA 투영을 위한 함수는 princomp 혹은 pcaproj 등이 있고, 집합  $\{x_i\}$  및  $d=2$ 을 입력으로 사용하여 2차원 평면상의 점들의 집합  $\{y_i\}$ 를 구하였다.

#### 4. 사용자 인터페이스 및 실험

약 2400여개의 표정으로 구성된 표정공간을 생성한 후에 이를 PCA를 통해 2차원 공간으로 투영하고, 사용자로 하여금 이 공간을 항해하면서 3차원 아바

타의 얼굴 표정을 실시간으로 제어하였다. 이를 위한 사용자 인터페이스는 그림 3과 같고, 3D Studio MAX R5.1의 MAX Script로 구현하였다.

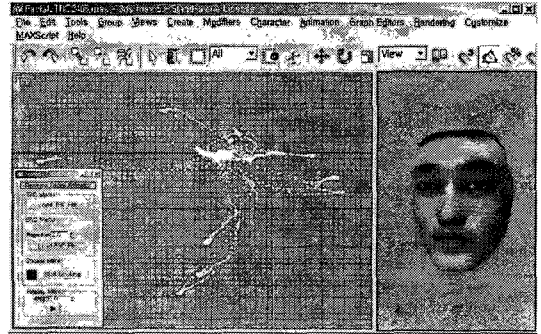


그림 3. 사용자 인터페이스(좌: 항해 공간과 스크립트 메뉴, 우: 3차원 얼굴 모델), 사용자가 2차원 항해공간에 분포된 각 얼굴 표정상태를 대표하는 작은 점을 마우스로 선택하면서 항해를 하면, 선택된 점에 해당되는 표정이 3차원 얼굴 모델에 표시된다.

그림 3의 왼쪽은 약 2400여개의 표정상태벡터를 2차원 평면에 분포시킨 항해 공간과 실시간 표정 제어에 사용될 스크립트 메뉴이고, 오른쪽의 3차원 얼굴 모델은 사용자가 항해 공간을 항해할 때 실시간으로 얼굴표정을 보여주게 한다. 그림 4는 PCA 투영에 의해서 생성된 약 2400여개의 2차원 표정공간을 사용자가 항해한 경로를 표시한 것이다.

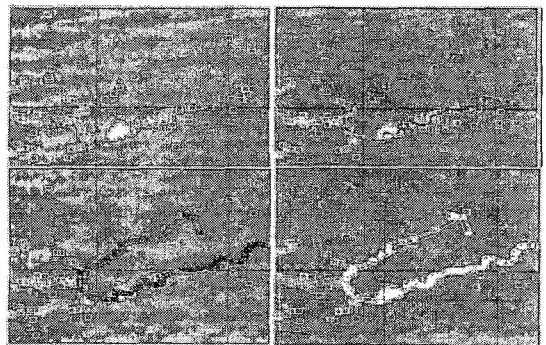


그림 4. 좌상: 2차원 표정공간의 점들, 우상: 항해과정. 빨간색 점들이 항해 경로, 좌하: 항해 경로를 항해 순서대로 연결, 우하: 적실자는 반복 재생 시 재생중인 현재 프레임

사용자는 마우스를 사용하여 그림 4의 우상과 같이 항해 공간을 항해하고 동시에 3차원 얼굴 모델에 적용된 얼굴 표정을 보게 되는데, 이때에는 표정 하

나하나를 향해 과정과 향해 속도에 따라 확인하면서 보게 된다.

사용자의 향해 과정이 끝나면 사용자의 향해 궤도에 해당되는 얼굴 표정들을 연속적으로 보고 표정의 변화를 확인할 필요가 있다. 이때에는 사용자의 향해 궤도를 처음부터 끝까지 자동으로 반복 향해하여 연속된 얼굴 표정의 변화를 3차원 얼굴 모델이 보여지게 된다. 그림 4 우하의 적십자는 이때의 향해 경로를 따라가는 현재의 경로 위치를 가리켜주기 위한 지시자이다. 본 사용자 인터페이스는 사용자가 그림 4와 같은 향해경로를 다수 개 생성하여 서로 다른 색상으로 표현할 수 있으며, 사용자가 향해한 경로들 중 임의의 경로 하나를 선택하여 재생하고 확인하여 수정할 수 있다.

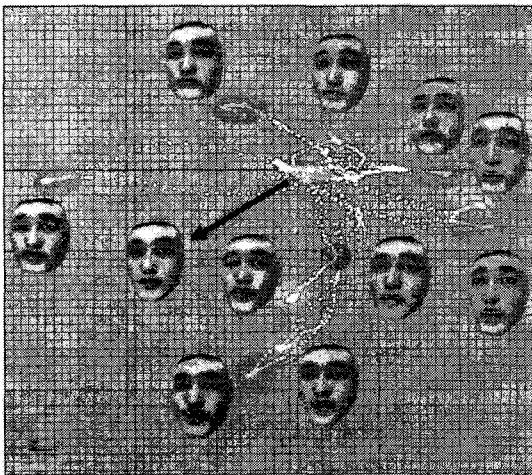


그림 5. 사용자가 PCA 투영에 의한 모든 2차원 표정공간을 향해하면서 마우스로 선택한 경로들 상에 있는 대표적인 얼굴표정들을 표시(원래 얼굴모델은 그림 5의 오른쪽 창에 표시되지만, 설명의 편리를 위해 향해공간에 표시)

그림 5는 사용자가 모든 2차원 표정공간을 향해하면서 마우스로 선택한 경로들 상에 있는 대표적인 얼굴표정들을 표시한 것으로서, 원래 얼굴표정은 그림 3에서와 같이 사용자 인터페이스의 오른쪽 창(얼굴 모델)에 표시되지만, 설명의 편리를 위해 향해공간에 표시하였다. 그리고 그림 5는 거리행렬 방식에 의해 표현된 표정상태를 사용한 결과로서, 임의의 두 표정 사이의 경로가 존재하기 때문에 3차원 아바타의 얼굴 표정을 실시간으로 제어하기에 효율적으로 분포되어져 있었다.

그러나 위치벡터방식으로 표정상태를 표현하는 경우, 위에서 언급한 결과들이 나오지 않고, 부분적으로 매우 인접해 있는 표정상태임에도 불구하고 서로 다른 표정 분포를 보이고 있다. 이는 임의의 두 표정간의 경로가 존재하지 않는 경우가 있다는 것으로서 본 논문의 목적에 맞지 않아 위치벡터방식으로 표정상태를 표현하는 경우는 사용하지 않는다.

그림 5에 분포된 표정공간은 무표정 상태를 중심으로 방사형을 구성하고 있는데, 사용자가 한두 번 정도의 향해를 해보면 수개의 얼굴 표정이 어떻게 분포되어져 있는지 쉽게 파악할 수 있다. 그러므로 실시간 얼굴 표정 제어를 위해 본 사용자 인터페이스를 사용하는 데에는 큰 문제가 없었다.

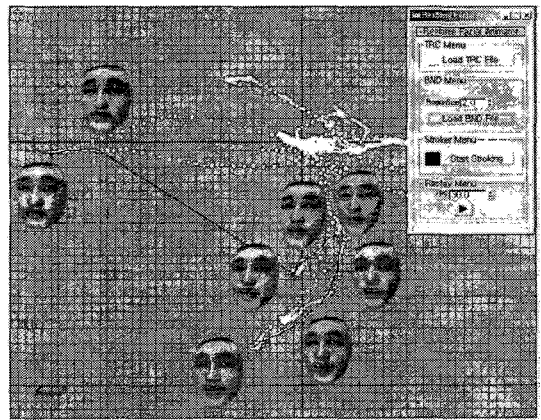


그림 6. 표정 제어를 위해 사용자의 향해 경로(파랑색)에 따라 나타나는 실시간 렌더링을 거친 3차원 아바타의 실시간 얼굴 표정 변화를 설명의 편리를 위해 향해공간에 표시함

그림 6은 약 2400여개의 얼굴 표정들이 분포된 2차원 표정 공간에서 3차원 아바타의 얼굴 표정 제어를 위해서 마우스를 사용하여 임의의 방향으로 향해 경로를 실시간적으로 생성할 때, 3차원 아바타의 얼굴 표정 변화를 실시간으로 보여주고 있다. 3차원 아바타의 얼굴 표정 제어는 그림 6의 스크립트 메뉴 'Start Stroking'을 사용하여 향해공간을 향해하는 사용자에게 의해서 실시간적으로 수행되어진다. 즉, 그림 6과 같이 하나의 표정에서 다른 표정으로 표정의 변화를 줄 때, 인접한 표정들을 향해하면 유사한 표정들이 실시간적으로 나타나고 거리가 멀어질수록 특정 표정으로 변화가 실시간적으로 진행된다. 그러나 인접한 표정을 거치지 않고 특정 거리만큼 점프를

하여 실시간 항해를 할 경우에는 매우 큰 표정의 변화가 있는 서로 다른 표정을 실시간적으로 보여준다. 또한 그림 6의 스크립트 메뉴 'Replay' 메뉴를 사용하여 사용자가 재생 속도(fps)를 마음대로 설정함으로써 항해경로에 해당되는 표정들의 실시간 재생을 반복적으로 수행하여 확인할 수 있도록 하였다.

## 6. 결 론

본 논문에서는 다량의 얼굴 모션 데이터들을 적당한 공간에 분포시키고, 사용자가 이 공간을 항해하면서 원하는 얼굴 표정들을 실시간적으로 선택하여 3차원 아바타의 얼굴 표정을 실시간적으로 제어하는 방법을 기술하였다. 약 2400여개의 얼굴 표정 프레임을 사용하여 항해 공간을 구성하였으며, 이 공간을 구성하기 위해서 PCA 투영 기법을 사용하였다. 3차원 아바타의 얼굴 표정을 실시간으로 제어하기 위해서 개발한 사용자 인터페이스는 표정공간을 위한 PCA 투영 기법을 실험적으로 확인하는데 유용하게 사용되었다. 또한 사용자 인터페이스는 사용자가 제어하고자 하는 3차원 아바타의 얼굴 표정을 직관적인 공간을 자유자재로 항해하면서 실시간적으로 제어가 가능하기 때문에 유용하고 효율적이라는 것을 사용자로 하여금 확인할 수 있었다.

본 논문의 연구 결과는 기존의 관련 연구와 비교할 때, 모션 데이터를 사용한 3차원 아바타 얼굴 표정의 실시간 제어가 가능하다는 것이 가장 큰 특징이다. 예를 들어 문자 위주의 채팅이나 메시지를 사용하여 상대방과의 의사소통을 하는 도중에 자신의 감정을 3차원 아바타를 사용하여 실시간적으로 실감나게 표현해줄 필요가 있는데, 이런 경우 본 논문의 연구 결과는 매우 적합하다. 그러므로 본 논문의 연구 결과는 3차원 아바타의 실시간 얼굴 표정 제어가 필요한 각종 응용 프로그램 분야에서 매우 유용하게 사용될 것이다.

## 참 고 문 헌

- [ 1 ] Demetri Terzopoulos, Barbara Mones-Hattal, Beth Hofer, Frederic Parke, Doug Sweetland, Keith Waters, "Facial animation : Past, present and future," Panel, SIGGRAPH97, 1997.
- [ 2 ] Frederic I. Parke, Keith Waters. "Computer facial animation," A K Peters, 1996.
- [ 3 ] Brian Guenter, Cindy Grimm, Daniel Wood, Henrique Malvar, and Frederic Pighin. "Making Faces," In SIGGRAPH 98 Conference Proceedings. ACM SIGGRAPH, July 1998.
- [ 4 ] 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.
- [ 5 ] Sannier G, Magnenat-Thalmann N., "A flexible texture fitting model for virtual clones," Proceedings of Computer Graphics International, IEEE Computer Society, pp. 167-176, 1997.
- [ 6 ] Wonseok Chae, Yejin Kim, Sung Yong Shin, "An Example-based Approach to Text-driven Speech Animation with Emotional Expressions", EUROGRAPHICS 2003 Volume 22, Number 3.
- [ 7 ] Hotelling, H., "Analysis of a complex of statistical variables into principal components", Journal of Educational Psychology 24, pp 417-441, 1933.
- [ 8 ] Pearson, K., "On lines and planes of closest fit to systems of points in space", The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science", Sixth Series 2, pp 559-572, 1901.
- [ 9 ] M. E. Tipping and C. M. Bishop, "Probabilistic principal component analysis," Journal of the Royal Statistical Society, Series B 61(3), pp 611-622, 1999.
- [ 10 ] Lucas Kovar Michael Gleicher Frederic Pighin, "Motion Graphs", ACM Transactions on Graphics 21(3) (Proceedings of SIGGRAPH 2002). July 2002.



김 성 호

- 1996년 상지대학교 이공과대학  
전산학과 졸업(학사)
- 1998년 숭실대학교 일반대학원  
컴퓨터학과(공학석사)
- 2001년 숭실대학교 일반대학원  
컴퓨터학과(박사수료)
- 1997년~2000년 숭실대학교, 숭

의여대 시간강사

- 2000년 1월~2001년 1월 (주)조오그래프 선임연구원
- 2000년 6월~2001년 9월 (주)애니큐브 부설 연구소장
- 2002년 3월~현재 숭의여대 정보통신계열 멀티미디어  
콘텐츠전공 겸임교수

2003년 9월~현재 (주)포도시스템즈 연구실장

관심분야 : 컴퓨터 그래픽스, 모션 캡처, 애니메이션, 가상현실, Web3D, 컴퓨터 비전, 영상처리, 멀티미디어 등