

아바타 통신에서의 얼굴 표정의 생성 방법

김진용*, 유재휘**

A Generation Methodology of Facial Expressions for Avatar Communications

Jin-Yong Kim*, Jae-Hwi Yoo **

요약

사이버 공간 내에서 텍스트나 영상 이외에 보조적인 통신 수단으로써, 아바타 통신을 이용할 수 있다. 실시간 아바타 영상 통신을 위하여, 전체 영상이나 압축 영상을 전송하는 대신에 아바타의 애니메이션 파라미터(팔의 동작 및 제스처를 위한 각도 또는 얼굴 표정을 위한 파라미터 등)만을 전송하는 지적 통신방식을 이용하기도 한다. 본 논문에서는 팔, 다리 등의 파라메타 등의 몸의 동적 움직임을 보조할 수 있는 수단으로써, 송신자의 감정을 표현할 수 있는 아바타의 얼굴 표정 생성에 대하여 제안한다. 얼굴 표정은 AU(Action Unit)에 의하여 표현할 수 있으며, 여기에서는 모양과 구조가 다른 아바타 모델에서 표정을 생성하기 위한 AU를 찾아내는 방법을 제안한다. 그리고, 감정 표현의 극대화를 위하여 얼굴 표현의 효율을 위하여, 눈썹, 눈, 코, 입에 대한 코믹 스타일을 가지는 아바타 모델을 제안한다. 또한, 얼굴 표정 합성을 위한 파라메타들도 제시하도록 한다.

Abstract

The avatar can be used as an auxiliary methodology of text and image communications in cyber space. An intelligent communication method can also be utilized to achieve real-time communication, where intelligently coded data (joint angles for arm gestures and action units for facial emotions) are transmitted instead of real or compressed pictures. In this paper, for supporting the action of arm and leg gestures, a method of generating the facial expressions that can represent sender's emotions is provided. The facial expression can be represented by Action Unit(AU), in this paper we suggest the methodology of finding appropriate AUs in avatar models that have various shape and structure. And, to maximize the efficiency of emotional expressions, a comic-style facial model having only eyebrows, eyes, nose, and mouth is employed. Then generation of facial emotion animation with the parameters is also investigated.

▶ Keyword : 아바타 통신(Avatar Communication), 얼굴표정(Facial Expression), AU(Action Unit)

• 제1저자 :김진용
• 접수일 : 2005.04.14, 심사완료일 : 2005.05.20
* 여주대학 산업경영정보과, ** 여주대학 컴퓨터정보관리과

I. 서론

얼굴 표정은 인터넷 사이버공간에서 개인을 상징하는 아바타(avatar) 통신의 보조 수단으로써 효율적으로 사용될 수 있다[1]. 특히 얼굴 표정을 가진 수화 영상은 단지 팔과 손의 움직임만을 보여주는 영상에 비해 이해하기가 쉽다. 또한 코믹 만화나 애니메이션 영화에서는 과장된 얼굴 표정이나 몸짓으로 사실적이지는 않지만 보다 확실하게 감정을 전달하고 있다[2]. 이때, 얼굴 구성 요소들에 대한 움직임을 표현하기 위해 FACS(Facial Action Coding System)의 AU(Action Unit)들의 조합을 이용한다[3, 4].

아바타 통신은 다양한 분야에서 활용될 수 있으나, 현재 수화 통신에서 그 효과를 톡톡히 보고 있다[1]. 즉, 실시간 수화 영상 통신을 위하여 수화 영상이나 압축 영상을 전송하는 대신에 수화 애니메이션 파라미터만을 전송하는 지적 통신방식을 이용한다. 즉 송신 측에서는 수화 영상과 얼굴 영상을 생성할 수 있는 수화 관절 각 파라미터와 표정 AU만을 전송하면 수신 측에서는 이 파라미터를 이용하여 다시 수화 애니메이션을 재생하는 방법이다. 이 때 애니메이션을 위한 관절 각 파라미터 값은 애니메이터의 경험에 의해 결정하였고, 얼굴 표정은 한 종류로 고정되는 등의 문제점이 있었다[5].

기존의 표정 에디터는 MetaCreations사의 Poser, 3D Planet사의 3D Create 및 Blaxxun사의 Avatar Studio 등이 있다[6]. 이들 시스템에서는 아바타의 표정을 생성하기 위해 AU를 미리 정의해서 표정을 생성하기 때문에 아바타 모델에 AU가 정의되어 있지 않으면, 표정을 생성할 수 없다. 따라서 표정을 생성할 수 있는 아바타 모델의 종류를 제한하고 있다. 그러나 가상공간상에서 활동하는 아바타는 크기, 폴리곤 수, 구조 등에서 다양한 종류가 있다. 따라서 임의의 아바타 모델에 표정을 생성하기 위해서는 필요한 AU를 찾아내는 작업이 필요하다[7].

본 논문에서는 모양과 구조가 다른 아바타 모델에서 표정을 생성하기 위한 AU를 찾아내는 방법을 제안한다. AU를 찾기 위하여 폴리곤 정보나 색상 정보, 또는 이 두 정보를 모두 이용한다. 또한 지적 통신방식에서 찾아낸 AU만을 전송하면 수신 측에서는 자기 모델에 이 AU를 적용하여 원

하는 감정 표현을 재생할 수 있도록 한다. 또한, 사람마다 표정이 다양하듯이 아바타 종류에 따라서 표정의 형태가 다르게 하며, 표정을 나타낼 수 있는 범위는 아바타의 얼굴 크기에 따라서 다르게 생성할 수 있도록 한다.

얼굴 표정 생성에서 특히 AU의 6번인 “뺨을 올림”, AU의 26번인 “턱을 내림”에 대한 움직임은 자동화가 어려운 조작이지만 아바타 통신에서 얼굴 표정을 전달하기 위해서 중요한 역할을 수행한다. 따라서 본 논문에서는 구조 혹은 모양이 다른 아바타 모델에 얼굴 표정을 생성하기 위해서 코, 볼, 턱의 영역을 추출하는 방법을 제안하고, 이렇게 찾아낸 볼, 턱을 가지고 포물형 편미분식(Parabolic Partial Differential Equations)과 Runge-Kutta법을 이용하여 보다 자연스러운 표정을 생성할 수 있는 방법을 제안한다. 그리고 3차원 모델에 표정의 한계값을 지정하여 다양한 아바타 모델마다 자동적으로 표정을 생성할 수 있게 한다.

이하 2장에서는 인터넷상 가상 공간상에서 구현한 지적 아바타 수화 통신 시스템을 고찰하고, 제 3장에서는 얼굴 모델을 설명한다. 제 4장에서는 다양한 아바타 모델에서 코믹한 얼굴표정을 위한 AU를 찾아내는 방법을 제안한다. 그리고 제 5장에서는 실험 결과를 고찰한 후, 제 6장에서 결론을 맺는다.

II. 지적 아바타 통신 시스템

인터넷 가상공간에서 지적 통신 방식에 의한 수화 제스처 통신 시스템은 중앙 집중적인 클라이언트-서버 구조를 가진다. 여기서 서버의 역할은 클라이언트로부터 오는 수화 문장을 각 클라이언트가 요구하는 언어권의 수화 파라미터로 번역하여 내보내 주는 것이다. (그림 1)은 인터넷상에 구현된 한-일간의 지적 수화 통신 시스템 구조이다[8].

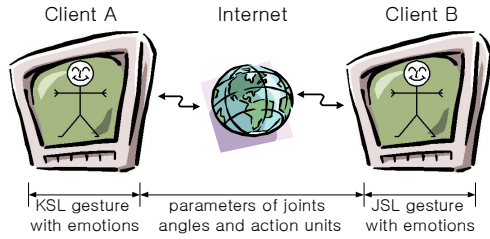


그림 1. 인터넷상에 구현된 한-일간의 지적 수화 통신 시스템.
Fig. 1. Intelligent communication system between Korean and Japanese on the internet.

여기서, 클라이언트 A는 한글 수화 (Korean Sign Language: KSL)를 구사하는 사이트이고, 클라이언트 B는 일본 수화 (Japanese Sign Language: JSL)를 쓰는 사이트이다. 이 때 실시간 통신을 위하여 수화 영상 대신에 수화 애니메이션을 위한 수화 제스처 및 얼굴 표정 파라미터를 전송하는 지적 통신 방식을 이용한다. 즉 클라이언트 A, B에 각각 캐릭터 모델을 준비하여 놓고 송신 측에서는 보낸 수화 및 표정 파라미터를 이용하여 수신 측에서 수화 영상을 재생한다.

III. 얼굴 모델

얼굴 영상을 실시간으로 디스플레이하기 위해서는 가능한 적은 수의 제어 점으로 구성되는 모델 영상이 필요하다. 반면에 보다 사실적인 얼굴 표정을 재현하기 위해서는 충분한 수의 폴리곤 모델이 필요하다. 따라서 본 연구에서는 적은 수의 폴리곤으로 감정을 효율적으로 표현할 수 있는 코믹 모델을 이용한다. 코믹한 모델이란 과장된 얼굴 표정을 합성하기 위해 감정 표현에 꼭 필요한 눈썹, 눈꺼풀, 눈동자, 입의 제어 점만을 사용한 모델이다.

기존 방식에서는 사진과 같은 사실적인 감정 표현을 위해 AU6 (뺨을 올린다)와 같이 얼굴 근육의 미세한 움직임을 표현하기 때문에 많은 AU의 조합이 필요하였다. 그러나 코믹 모델에서는 얼굴 근육의 필요한 부분만을 강조하여 과장된 표현을 하기 때문에 보다 적은 수의 AU만으로도 효과적인 감정 표현이 가능하게 된다.

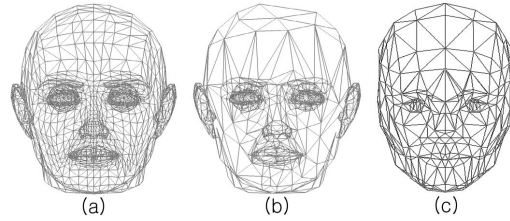


그림 2. 감정 표현을 위한 3차원 wireframe 얼굴 모델 (a) 3,800, (b) 1,900, (c) 296 폴리곤 모델.
Fig. 2. 3D facial wireframe for expressing the emotions. (a) 3,800 (b) 1,900, (c) 296 polygon model.

(그림 2)는 폴리곤 수가 3,800, 1,900, 그리고 296개인 3차원 wireframe 얼굴 모델이다

여기서 (a)와 (b)는 동일한 구조의 모델로써 폴리곤의 수를 다르게 한 모델이고, 마지막으로 (c)는 구조와 폴리곤 수를 모두 다르게 한 모델이다

IV. 얼굴 표정 재생 방법

3.1 AU 검출 방법

(그림 3)은 3차원 모델에 포함되어 있는 폴리곤 정보와 색상 정보를 이용하여 AU와 AU 주변을 추출하고, 6가지 표정을 생성하는 절차를 나타낸 것이다. 여기서 3D Model에는 3차원 아바타 모델을 생성하기 필요한 정보들이 포함되어 있다. 즉, 폴리곤, 벡터스 및 색상 정보를 가지고 있다.

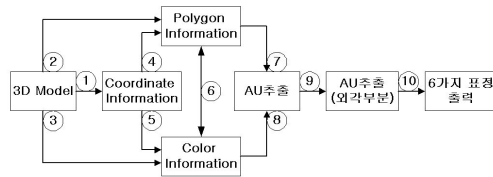


그림 3. 얼굴 표정 생성을 위한 AU 추출 절차.
Fig. 3. Procedure of AU extraction for generating facial emotions.

단계 ①, ②, ③에서는 버텍스, 폴리곤, 색상 정보를 참조하며, 단계 ④, ⑤에서는 AU를 찾을 수 있도록 위치적 정보를 제공한다. 그리고 단계 ⑥에서는 폴리곤과 색상정보를 같이 사용하여 상호 보완적으로 참조하여 그룹핑을 수행한다. 또한 단계 ⑦, ⑧은 폴리곤 정보와 색상정보를 이용하여 그룹핑하며, 단계 ⑨, ⑩은 AU 추출과 AU 주변 추출을 한다.

여기서 그룹핑이란 폴리곤들 간에 혹은 색상들 간에 떨어져 있으면 다른 그룹으로 분류하는 것이며, AU 주변(외각부분)이란 AU가 움직임으로써 영향을 미칠 수 있는 주변이 되는 영역을 의미한다.

<표 1>은 AU를 찾아내기 위한 3가지 방법이며, (그림 3)의 3D Model에서 제공되는 정보들 중에서 어떠한 부분을 참조하는가에 따라서 3가지로 나눌 수 있다. 즉, 폴리곤 정보 이용하여 AU를 찾지 못하면, 색상 정보를 이용하여 AU를 찾는다. 만약 색상 정보를 이용하여 AU를 찾지 못하면, 폴리곤 정보와 색상정보를 같이 이용하여 AU를 찾는다.

(그림 4)는 코의 꼭짓점과 그룹핑된 그룹들 사이에 있는 상관정보를 이용하여 중요한 얼굴 구성요소를 찾는 과정을 나타낸다.

표 1. AU를 찾아내기 위한 3가지 방법
Table 1. 3 methods for finding AUs.

AU를 찾는 방법	AU 찾는 순서
폴리곤을 이용하는 방법	① ② ④ ⑦ ⑨ ⑩
색상 정보를 이용하는 방법	① ③ ⑤ ⑧ ⑨ ⑩
폴리곤과 색상 정보를 모두 이용하는 방법	① ② ③ ④ ⑤ ⑥ ⑦ ⑧ ⑨ ⑩

(그림 5)는 AU 주변을 추출하기 위해서는 (그림 2)의 단계 ⑦, ⑧에서 찾아낸 AU의 위치를 기준으로 사각형의 크기를 생성하는 방법을 나타낸다. 여기서 사각형은 주변을 포함하는 크기를 의미하며, 사각형의 크기는 눈썹과 눈꺼풀, 그리고 입술의 평균위치에 따라서 크기가 다르다. 눈썹과 눈꺼풀 주변의 크기는 눈썹과 눈꺼풀의 높이 차이이며, 입술 주변의 크기는 윗입술과 아랫입술의 높이 차이로 나타낼 수 있다. 사각형은 큰 사각형과 작은 사각형으로 분리하며, AU 움직임이 100%인 경우, 작은 사각형은 66% 움직이며, 큰 사각형은 33% 움직인다.

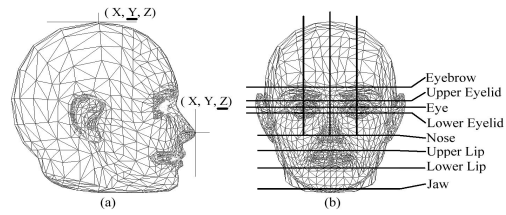


그림 4. 얼굴 구성요소의 추출. (a)는 코의 꼭짓점 위치를 추출, (b)는 코의 꼭짓점 기준과 그룹들의 위치 비교
Fig. 4. Extraction of facial components.

(a) Extraction of vertex of nose
(b) Comparison of vertex of node and other groups.

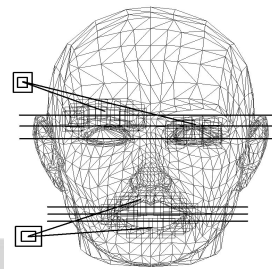


그림 5. AU 주변을 추출하는 사각형
Fig. 5. Rectangular for extracting AUs.

AU 주변을 추출하기 위해서는 AU의 위치를 기준으로 한다.

턱을 추출하는 방식은 인접한 폴리곤을 찾아 레벨을 지정하고 Normal Vector 계산을 사용한다.

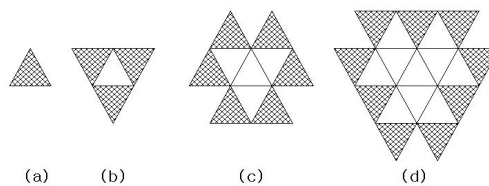


그림 6. (a) 레벨0, (b) 레벨1, (c) 레벨2, (d) 레벨3
Fig. 6. (a) level 0 (b) level 1 (c) level 2 (d) level 3

아래 입술의 가장 근접한 폴리곤을 찾아서 레벨을 0으로 처리하며, 레벨0 폴리곤에 멀어질수록 가중치를 부여한다. 레벨0을 기준으로 Normal Vector를 사용하여 비교한다.

즉, 레벨0과 레벨1을 Normal Vector(N)를 비교하며, 레벨1과 레벨2, 다음으로 레벨2와 레벨3을 비교하는 순서로 하고. 하나에 폴리곤에 3개의 Vertex(V)를 각각 $V1 = (X1, Y1, Z1)$, $V2 = (X2, Y2, Z2)$, $V3 = (X3, Y3, Z3)$ 라고 할 때, 이 벡터들 사이에 외적을 취하면

$$N = (V2 - V1) \times (V3 - V1)$$

이 된다. 여기서 X, Y, Z중에서 Z축만을 기준으로 턱을 추출하기 때문에

$$N = (Z2 - Z1) \times (Z3 - Z1)$$

이 된다. 여기서

$$0 < |N| < 0.0005$$

를 만족할 경우 이 벡터를 턱을 구성하는 벡터스로 판정하였다. 이 때 범위는 모델에 따라서 다를 수 있으며, 실험적으로 구한 값이다.

3.2 얼굴 영역 추출

296개의 벡터를 가지고 있는 모델에서 폴리곤과 색상을 각각 따로 이용한 경우에 AU를 찾아낼 수 없으며, “폴리곤, 색상, 좌표 정보를 참조”하여 AU를 찾아낸다. 첫 번째 단계에서는 폴리곤을 이용하여 그룹핑 시켜서 6개의 그룹으로 분류한 후, 두 번째 단계에서 색상으로 다시 그룹핑하여 12개의 그룹을 생성한다. 세 번째 단계에서 좌표정보와 Normal Vector를 이용하여 턱, 코, 볼의 위치를 추출하여 총 15개의 그룹을 생성한다.

먼저 코 영역을 추출하는 방법은 얼굴 정면에서 가장 낮은 부분을 기준으로 Normal Vector의 z축값을 가지는 NVz의 값을 결정한다.($0 \leq NVz < 0.00005$)

다음으로 볼 영역을 추출하는 두 가지 절차가 있다. 첫 번째는 윗입술과 아랫입술 사이의 평균값과 그리고 위 눈꺼풀과 아래 눈꺼풀 사이의 평균값 경계를 벗어나지 못한다. 두 번째는 코와 입술 주변(윗입술 주변, 아랫입술 주변, 턱 부분을 제외한 부분에서 코를 기준으로 NVz를 결정한다.($0.00005 \leq NVz \leq 0.005$)

마지막으로 턱 영역을 추출하는 두 가지 절차가 있다. 첫 번째는 윗입술과 아랫입술 사이의 경계를 기준으로 위의 영역을 포함하지 않는다. 두 번째는 외각 아랫입술 영역에 바

로 밑에 있는 벡터를 기준으로 NVz의 값을 결정한다.($0 \leq NVz \leq 0.005$)

3.3 볼에 대한 처리

이전의 작업은 볼의 영역을 찾아낸다. 이후에 3차원 모델에 찾아낸 볼 영역의 움직임을 위해서 포물형 편미분식을 사용하였다. 그래서 볼의 움직임이 가능하도록 하였다.

포물형 편미분식은 다음과 같다.[4]

$$u_i^{j+1} = u_i^j + \alpha \frac{(\Delta t)}{(\Delta x)^2} (u_{i+1}^j - 2u_i^j + u_{i-1}^j)$$

여기서 입력으로 α , i , j , Δx 그리고 Δt 가 필요하다. 여기서 α 는 볼록한 정보를 나타내며, 값이 작을수록 볼록한 정도가 날카로운 모양을 나타내고, 0.1에서 10까지의 실험치로 1을 지정하였다. i 는 거리 x 에 대한 절점 번호를 나타내고, j 는 시간 t 에 대한 반복횟수번호를 의미한다. Δx 는 거리구간이며, Δt 는 시간구간이다. 여기서 Δx 와 Δt 식은

$$\Delta t \leq \frac{(\Delta x)^2}{2\alpha}$$

$$\Delta x = length / (n - 1) \quad \text{이다.}$$

여기서 입력으로 $length$ 는 볼의 반지름이며, 길이의 분할 횟수 n 을 나타낸다.

다음으로 시간 $t=0$ ($j=1$)에서의 $(u_i^j, u_{i-1}^j, u_{i+1}^j)$ 값으로부터 u_i^j 는 분할된 영역의 정점이고, u_{i+1}^j 는 이전의 정점이며, u_{i-1}^j 는 다음의 정점이다. 마지막으로 시간 $(t + \Delta t)$ 에서 u_i^{j+1} 값을 계산한다 ($i = 2 \dots n$).

그러나 볼의 움직임을 나타내는 동작은 4가지 있으며, ㉓움직임은 “입술 주변 이동” ㉔움직임은 “볼의 중심에서부터 볼록하게 이동”, ㉕움직임은 “볼의 중심에서부터 x와 y축으로 이동”, 마지막으로 ㉖움직임은 “입술 주변에서부터 x와 y축으로 이동”이다. 이들 움직임 중에서 ㉔와 ㉕는 포물형 편미분식에서 생성된 정보를 참조한다.

(그림 7)은 4가지 동작들을 볼의 움직임에 영향을 끼쳐 변화하는 모습을 나타낸다.

(그림 7(a))는 ㉓와 ㉔움직임을 얼굴의 X축과 Y축을 기

준으로 표시하고, (b)는 ㉑와 ㉒움직임을 얼굴의 X축과 Z 축을 기준으로 표시하고, 마지막 그림으로 (c)는 ㉑와 ㉒ 그리고 ㉓움직임을 얼굴의 Y축과 Z축을 기준으로 표시한 결과이다.

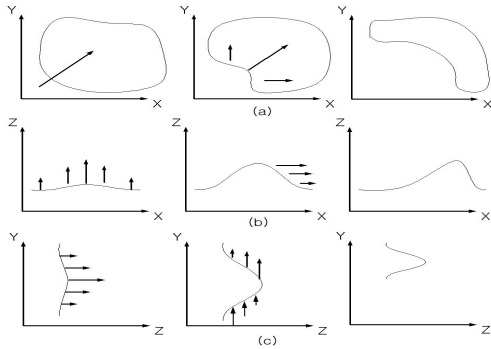


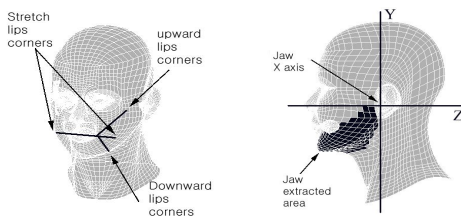
그림 7. 볼의 이동을 보기 위한 XY, XZ, YZ축에서 볼 때의 결과
Fig. 7. Results for movement of cheek in aspect of XY, XZ, YZ

3.4 턱에 대한 처리

AU가 움직이는 범위와 각도가 고정되어 있으므로, 다양한 모델에 AU의 움직임 범위를 제한하기 위해 입술의 한계 값은 (그림 8(a))에서 정한다. 턱의 한계 값은 코와 턱의 기준축을 0°으로 하고 (그림 8(b))에서 턱의 기준 축으로 Runge-Kutta법을 사용하며, 실험치로 75°을 초과하지 못하게 하였다.

(그림 8(b))에서 턱의 기준축 (X, Y, Z)에서 X와 Y는 코를 중심으로 하고, Z는 얼굴의 앞면과 뒷면의 중앙을 기준으로 선정하였다.

턱의 움직임을 기준축 기준으로 움직임과 함께 자연스러운 움직임을 지정할 수 있도록 턱의 움직임 사이에 보안을 할 수 있는 Runge-Kutta법을 사용하였다.



(a) (b)
그림 8. (a) AU에 대한 움직임의 한계 값과 (b) 턱의 영역과 기준 축
Fig. 8. (a) Limit value for movement of AUs and (b) domain of chin and base line

턱의 새로운 좌표 (X', Y', Z') 위치를 계산한다.

$$\begin{aligned} X' &= Xv \\ Y' &= d_n * \cos(t_n) \\ Z' &= d_n * \sin(t_n) * (\sin(ph)) \end{aligned}$$

초기각도(θ₀)를 라디안 각도로 변환한 t_n과 턱과 기준축 사이의 거리값인 d_n을 사용하다. 여기서 턱이 옆으로 이동하는 것을 고려하지 않기 때문에 X'는 기존의 좌표와 동일하다. 또한 t_{n+1}을 생성하기 위해서 Runge-Kutta법을 사용한다[4].

$$t_{n+1} = t_n + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6}$$

여기서, k₁, k₂, k₃ 그리고 k₄는 다음과 같다.

$$\begin{aligned} k_1 &= h * f(\Omega_n, t_n) \\ k_2 &= h * f(\Omega_n + \frac{k_1}{2}, t_n + \frac{h}{2}) \\ k_3 &= h * f(\Omega_n + \frac{k_2}{2}, t_n + \frac{h}{2}) \\ k_4 &= h * f(\Omega_n + k_3, t_n + h) \\ \Omega_{n+1} &= \Omega_n + h \end{aligned}$$

여기서 사용된 입력은 t_n, Ω_n, h 그리고 N틀 입력이 필요하며, θ₀를 이용하여 t_n을 산출한다. Ω_n는 0으로 대입하고, h와 N은 각각 0.2 그리고 1로 대입한다. 즉, h가 누적되어 N이 될 때까지 반복하게 된다. 마지막으로 턱의 기준축과 턱의 벡스 Yv와 Zv의 차이로 각각 cy와 cz를 산출하고, 이를 이용하여 턱과 기준축 사이의 거리 d_n = √(cy² + cz²)를 알아낸 이후에 θ₀를 산출한다. 결과는 실험에서 고찰한다.

V. 실험

5.1 표정 생성

다음 (그림 9)는 (그림 4)와 (그림 5)에서 찾아낸 AU와 AU 주변을 사용하여 3800, 1900, 그리고 296 버텍스를 가진 3차원 wireframe 모델과 비교한 것이다. 좌측부터 눈썹, 눈꺼풀, 입술, 턱을 추출하는 화면이다.

(그림 10)은 (그림 3)에서 ⑩번 과정에서 표정을 생성한 화면이며, 좌측부터 기쁨, 슬픔, 역겨움, 화남, 놀람, 그리고 두려움을 나타내는 6가지 표정 화면이다.

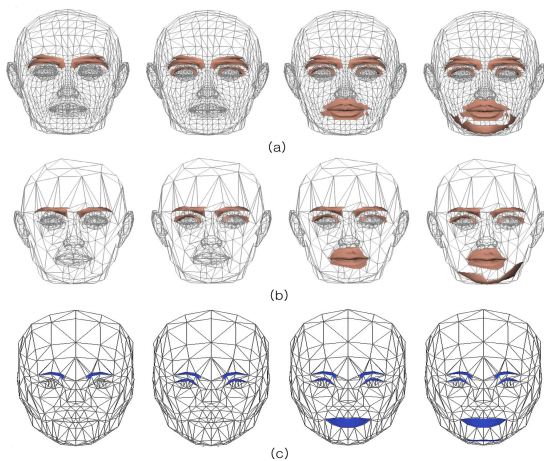


그림 9. AU와 AU 주변을 찾아내는 과정.
(a) 3800개, (b) 1900, (c) 296개 버텍스를 가진 모델
Fig. 9. Process for finding AUs and AU's neighborhoods.

표정을 나타낼 때 사용할 수 있는 6가지 표정은 내적상태 기술단어의 차원값 분포[5]에서 도표로 제시할 수 있다. 이 도표에 의하면 화남과 혐오(역겨움)은 분포가 비슷하며, 두려움(무서움)과 슬픔(울적함)도 비슷한 위치에 있다. 표정을 나타낼 때 동일한 AU를 사용하는 것과 다른 AU를 사용하는 것을 분류하여, 다른 AU를 사용하는 것을 과장된 표현함으로써 6가지 표정들을 분리시킬 수 있다.

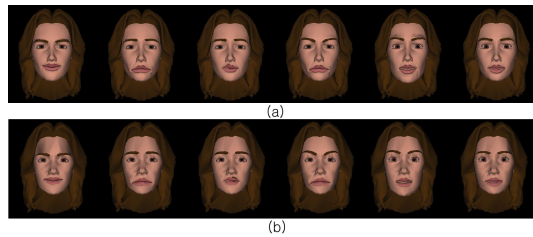


그림 10. 6가지 표정을 생성
(a) 3800개 버텍스, (b) 1900개 버텍스
Fig. 10. Results of generating 6 expressions using (a) 3800 vertex (b) 1900 vertex

<표 2>는 6가지 표정에서 나타낼 수 있는 표정의 종류를 나열하고, 가장 잘 어울리는 표정을 음영으로 처리하였다. 여기서 AU(2, 4)란 화남과 혐오에서 공통적으로 사용하는 AU이고, AU(1, 4)는 두려움과 슬픔에서 같이 사용하는 AU이다. 또한 AU(1, 2)는 기쁨과 놀람에서 같이 사용하는 AU를 의미한다.

6가지 표정은 AU공통과 표현 방법의 종류를 결합하여 표정을 생성할 수 있다.

표 2. 표정을 나타낼 때 사용할 수 있는 AU의 종류
Table 2. Kinds of AUs can be used for generating facial expressions.

AU 공통	얼굴 표정의 표현 방법							
	1	2	3	4	5	6	7	
화남	24	5	15	20	5,15	5,20	15,20	5,15,20
혐오	24	10	18	48	10,18	10,48	18,48	10,18,48
두려움	1,4	20	25	26	20,25	20,26	25,26	20,25,26
슬픔	1,4	15						
기쁨	1,2	12	25	12,25				
놀람	1,2	5	26	5,26				

<표 3>은 296개의 버텍스를 가지고 있는 모델은 폴리곤과 색상을 각각 따로 이용한 경우에 AU를 찾아낼 수 없으며, <표 1>에서 “폴리곤과 색상 정보를 참조”하여 AU를 찾아낸다. 첫 번째 단계에서는 폴리곤을 이용하여 그룹핑 시켜서 6개의 그룹으로 분류한 후, 두 번째 단계에서 색상으로 다시 그룹핑하여 12개의 그룹을 생성한다.

5.2 타 방법과의 비교

본 논문의 성능을 평가하기 위하여, 우선 일반적인 AU를 이용하는 방법 및 3D MAX를 이용한 방법과 상호 비교

하도록 한다. 표정에 의한 감정 전달의 효과이기 때문에 수치적인 자료보다는 영상에서의 감정 표현의 정도를 판단하도록 한다.

(그림 11)은 <표 2>에서 음영으로 처리된 표정과 <표 3>에서 폴리곤과 색상을 이용한 그룹핑 된 내용을 가지고 6가지 표정을 생성한 것이다. 일반적인 AU를 이용하는 방법은 감정 표현에 있어서 구분이 다소 애매한 것을 볼 수 있다.

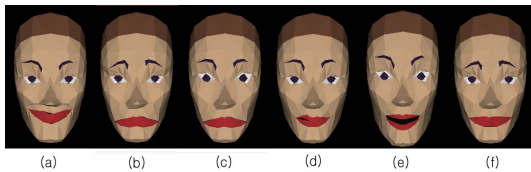


그림 11. 296개 모델에 과장된 표현으로 생성한 6가지 표정. (a) 기쁨, (b) 슬픔, (c) 역겨움, (d) 화남, (e) 놀람, (f) 두려움
 Fig. 11. 6 kind of expressions using 296 vertex model. (a) joy (b) sad (c) disgust (d) angry (e) surprise(f) fear

표 3. 폴리곤과 색상을 이용한 그룹핑 생성(296개 버텍스를 가진 모델)
 Table 3. Polygons and groups using color.(model having 296 vertex)

방식 그룹번호	폴리곤과 색상 정보를 모두 이용하는 방법	
	폴리곤(첫 번째)	색상(두 번째)
1	머리카락	머리카락
2	얼굴	얼굴
3		왼쪽눈썹
4		오른쪽 눈썹
5	왼쪽 눈	왼쪽 눈꺼풀
6		왼쪽 홍채
7		왼쪽 흰자
8	오른쪽 눈	오른쪽 눈꺼풀
9		오른쪽 홍채
10		오른쪽 흰자
11	윗입술	윗입술
12	아랫입술	아랫입술
그룹 합계	6개	12개

(그림 12)는 3D MAX를 이용하여 “JOY” 표정을 생성한 결과이다. (그림 12)의 결과를 살펴 보면, 볼의 영역이 거의 소멸된 것을 볼 수 있다. 다시 말해, “JOY”에 대한 의미 전달은 명확하지만, 자연스러운 표정 생성이 결여된다.

MAX를 이용하여 표정을 생성하는 방법으로 두 가지 방법을 사용하였다. 첫째로 FFD(Free Form Deformation modifier) 박스를 이용한 입술과 볼에 대한 움직임 제어

하는 영역을 선정하고, 둘째로 Affect Region을 이용한 선택된 영역에서 볼과 입술에 대한 움직임을 제어하였다.

(그림 13)은 본 논문에서 제안된 방법으로 생성된 “JOY” 표정을 나타낸 것이며, 표정에 대한 의미전달과 자연스러운 표정생성이 가능하였다.

VI. 결론

인터넷 가상공간에서 이루어지는 지적 수화 제스처 통신에서는 수화 영상을 전송하는 대신에 수화 제스처 및 얼굴 표정 애니메이션을 생성할 수 있는 관절 각 및 AU 파라미터만을 전송한다.

가상공간에서 다양한 아바타 모델들이 있으며, 아바타 모델마다 AU를 미리 지정하여 하는 것은 매우 어렵다. 논문에서는 다양한 아바타 모델에 AU를 미리 지정하지 않아도 표정을 생성하기 위해서 아바타 모델에 AU를 찾아내기 위한 3가지 방법과 표정 생성하는 과정을 제안하였다.

그리고 6가지 표정 중에 화남과 혐오, 그리고 두려움과 슬픔에서 분포가 비슷한 위치에 있으며, 표정을 나타내는데 표정들 간에 다른 AU를 사용하는 부분을 과장된 표현으로 함으로써 6가지 표정을 분리시켰다.

제안한 방법은 일반적인 AU 방법 및 3D MAX의 방법에 비하여 감정 전달에 있어서 효율적임을 알 수 있었다.

그러나 특별한 구조를 가지고 있는 모델에는 본 방법을 적용하기 어렵다. 또한 현재 제시되고 있는 아바타 모델은 Texture를 지원하지 않으므로 실제 사진과 같은 표정을 나타내기 어렵다. 따라서, 이에 관한 연구가 추후 진행되어야 한다.

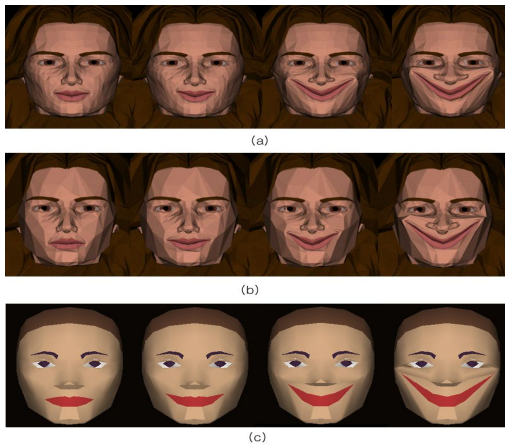


그림 12. 3800, 1900, 296개 버텍스를 가진 모델에 3D MAX로 구현한 "JOY" 표정 생성 결과
 Fig. 12. Results for "JOY" expression using 3D MAX in 3800, 1900 and 296 vertex models.

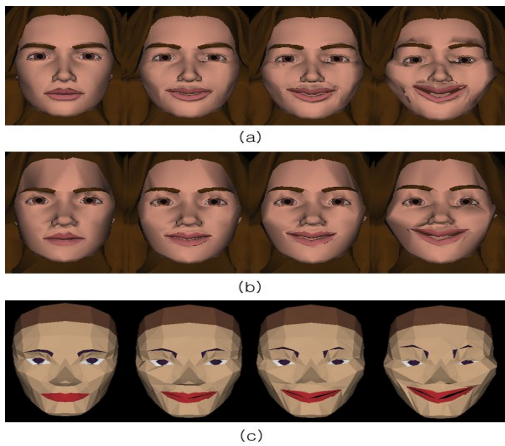


그림 13. 3800, 1900, 296개 버텍스를 가진 모델에 포물형 미분 방정식을 이용한 "JOY" 표정 생성
 Fig. 13. Results for "JOY" expression using differential equation in 3800, 1900 and 296 vertex models.

참고문헌

- [1] 金商雲, 吳芝英, 青木 由直, "인터넷상에 3차원 모델을 이용한 한-일 실시간 수화 통신 시스템의 구축을 위한 기초적인 검토", 대한전자공학회논문지-S, vol. 36, no. 7, pp. 811 - 820, 1999. 07.
- [2] 高光訓, 金 商雲, 棚橋眞, 青木 由直, "효율적인 수화 통신을 위한 코믹한 감정 표현", 대한전자공학회국제 종합학술대회 논문집, vol. 22, no. 1, pp. 651-654, 1999. 6.
- [3] 李容厚, 金商雲, 青木由直, "인터넷 가상공간에서 지적 아바타 통신을 위한 코믹한 얼굴 표정의 생성법", 대한전자공학회논문지-S, vol. 40, no. 1, pp. 59-73, 2002.
- [4] P. Ekman and W.V. Friesen, Unmasking the Face, New Jersey: Prentice Hall, 1975
- [5] I. Essa and A. Pentland, "Coding, Analysis Interpretation, Recognition of Facial Expressions," IEEE Trans, Pattern Analysis and Machine Intelli., vol. 19, no.7, pp.757-763, 1997.
- [6] S. K. Wilcox, Web Developer.Com Guide to 3D Avatar, John Wiley & Sons, Inc., pp.99 - 184, 1998.
- [7] M. Wang, Y. Iwai, and M. Yachida, "Expression Recognition from Time-Sequential Facial Images by Use of Expression Change Model", Proc. Int'l Conf. Automatic Face and Gesture Recognition, pp. 324-329, 1998.
- [8] Sang-Woon Kim, Ji-Yong Oh, Shin Tanahashi, and Yoshinao Aoki, "Preliminary study on a sign-language chatting system between Korea and Japan for avatar communication on Internet." IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, vol. E83-A, no. 2, pp. 386 - 389, Feb. 2000

저 자 소개



김진용

1994년 8월~현재: 여주대학 산업
경영정보과 부교수

1998년 8월 동국대학교 산업공학과
공학박사

소속학회: 한국컴퓨터정보학회,
대한산업공학회



유재휘

1995년 3월~현재: 여주대학 컴퓨
터정보관리과 부교수

2002년 2월 명지대학교 컴퓨터공학과
박사과정 수료

소속학회: 한국컴퓨터정보학회, 한
국정보처리학회, 한국정보과
학회

K C I