
웹 응용을 위한 MPEG-4 얼굴 애니메이션 파라미터 추출 및 구현

박경숙* · 허영남** · 김응곤***

Extraction and Implementation of MPEG-4 Facial Animation Parameter
for Web Application

Kyong-sook Park* · Yeong-nam Heo** · Eung-kon Kim***

이 논문은 2002년도 순천대학교 공과대학학술재단 연구비에 의하여 연구되었음

요약

본 연구에서는 기존의 방법에 비하여 값비싼 3차원 스캐너나 카메라를 이용하지 않고 정면과 측면 영상을 이용하여 3차원 모델을 생성하는 3차원 얼굴 모델러와 애니메이터를 개발하였다. 이 시스템은 특정한 플랫폼과 소프트웨어에 독립적으로 웹상에서 애니메이션 서버에 접속함으로써 3차원 얼굴 모델을 애니메이션할 수 있으며 자바 3D API를 이용하여 구현하였다.

얼굴모델러는 입력 영상으로부터 MPEG-4 FDP(Facial Definition Parameter) 특징점을 추출하여 일반 얼굴모델을 특징점에 따라 변형시켜 3차원 얼굴 모델을 생성한다. 애니메이터는 FAP(Facial Animation Parameter)에 따라 얼굴모델을 애니메이션하고 렌더링한다.

본 시스템은 웹 상에서 아바타를 제작하는 데 사용될 수 있다.

ABSTRACT

In this study, we developed a 3D facial modeler and animator that will not use the existing method by 3D scanner or camera. Without expensive image-input equipments, we can easily create 3D models only using front and side images. The system is available to animate 3D facial models as we connect to animation server on the WWW which is independent from specific platforms and softwares. It was implemented using Java 3D API.

The facial modeler detects MPEG-4 FDP(Facial Definition Parameter) feature points from 2D input images, creates 3D facial model modifying generic facial model with the points. The animator animates and renders the 3D facial model according to MPEG-4 FAP(Facial Animation Parameter).

This system can be used for generating an avatar on WWW.

*한영대학 인터넷 소프트웨어과 교수

**순천대학교 컴퓨터과학과 교수

I. 서론

최근의 다양한 멀티미디어에 대한 요구와 디지털 기술산업의 발달로 통신과 컴퓨터, 가전, 영화산업 등의 전통적인 경계는 의미가 없어졌다. 전통적으로 나누어 있던 이들 산업들의 결합으로 새로운 멀티미디어 및 통신에 대한 표준의 요구가 증대하고 있다. 이러한 요구에 부응하여 상호작용성(interactivity)과 높은 압축률, 범용 접근성을 제공하는 MPEG-4의 표준이 있다. MPEG-4는 동영상부호화 및 멀티미디어 데이터베이스에 대한 접근을 위해 객체기반의 부호화를 기반으로 하고 있다[1].

MPEG-4 SNHC를 이용한 3차원 동영상 부호화는 가상공간에서 3차원 그래픽처리, 3D 게임에서의 인물처리 등의 가상현실 기반 기술과 인터넷망을 통한 동화상 전송 등의 인터넷의 멀티미디어 서비스의 핵심 기술로 활용된다[1-8].

가상공간에서의 인간의 얼굴과 몸체를 만드는 대리인(avatar)의 구현은 MPEG-4 SNHC의 핵심 기술이라 말할 수 있다. 이러한 가상공간에서의 대리인을 구현하는 기술은 인간과 컴퓨터의 인터페이스와 상호작용적 게임, 멀티미디어 타이틀, 원격교육, 가상공간 하의 원격회의 등 많은 응용분야에서 필요로 한다.

따라서 본 논문에서는 기존의 방법에 비하여 3차원 스캐너나 카메라를 이용하지 않고 비용과 노력을 크게 절감하면서 실감나는 얼굴 모델링을 효율적으로 수행하는 웹 기반 3차원 얼굴 모델러와 애니메이션을 개발하였다. 사용자는 두 장의 사진을 입력으로 하여 웹브라우저만으로 3차원 얼굴 모델을 생성하게 되며, 생성된 3차원 모델에 대하여 웹 기반으로 애니메이션을 수행하게 된다. MPEG-4 SNHC 그룹에서 정의한 FDP(Facial Definition Parameter)와 FAP(Facial Animation Parameter)의 얼굴 파라미터들을 추출하고, Java 3D API를 이용하여 얼굴 애니메이션을 구현하여 웹상의 3차원 가상공간에서 활용하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구, 3장에서는 웹 기반 얼굴모델 생성에 대하여, 4장에서는 FAP를 이용한 웹 기반 얼굴 애니메이션에 대하여 기술하고, 5장에서는 결론을 맺는다.

II. 관련 연구

얼굴 애니메이션의 연구는 1974년 Parke의 파라메트릭 모델(parametric model)을 시초로 시작되었다. 얼굴 애니메이션 연구의 주된 방향은 감정과 입술 움직임 등을 처리하기 위한 정확하고도 효율적인 방법을 찾는 것으로, 대표적인 것으로는 얼굴에서 변형되는 부분들을 기준으로 얼굴 표현을 계층적 구조를 갖는 영역들로 구조화한 Platt and Badler[9]의 연구와 얼굴의 변형을 안면 근육들의 움직임의 결과를 해석하여 얼굴 변형에 영향을 미치는 근육들을 시뮬레이션한 Waters[10]의 연구 등을 들 수 있다. 이러한 연구들은 대부분 얼굴의 물리적 성질을 이용한 접근 방법을 택했다고 볼 수 있는데, 이러한 방법들은 실제로 인간의 얼굴이 변형되는 방식을 그대로 표현하므로 정확도 면에서 장점을 지니는 반면에 계산의 양이 많고 애니메이션 시스템 구현이 까다롭다는 단점도 갖는다[8].

얼굴의 애니메이션을 구현하기 위한 방법들은 실제로 얼굴 모습을 표현하는 피부 모델을 어떤 방법으로 만들었는가에 따라 좌우될 수밖에 없는데, 3차원 피부 모델 생성에 가장 흔하게 이용되는 다각형 메시의 경우 상대적으로 조작성 쉽지만 부드러운 곡면으로 이루어지는 얼굴을 사실적으로 표현하기 위해서는 모델을 구성하는 다각형의 수가 많아져야 한다. 1990년 Waite[11]는 B-스플라인(B-Spline) 곡면을 이용하여 얼굴 모델을 표현하였는데, 이 모델을 얼굴 표정들을 나타낼 때 비교적 적은 수의 데이터로 사실적인 표현을 할 수 있었으나 치아 및 눈썹 등과 같이 세밀한 부분의 처리가 불가능하였다.

MPEG-4는 초저속 동영상부호화 및 멀티미디어 데이터베이스에 대한 접근을 위해 객체기반의 부호화를 기반으로 하고 있다. MPEG-4는 다양한 기능의 표준화를 위해 여러 개의 전문그룹을 구성하고 있으며, MPEG-4 비디오와 오디오그룹에서는 각각 비디오와 오디오정보에 대한 효율적인 부호화 방법을 표준화하는 반면 SNHC 그룹에서는 자연적이거나 합성의 영상과 음향 정보의 효율적인 표현과 결합에 대해 표준화를 한다. 자연적인 영상과 합성의 영상은 자기 그 자체가 영상전체에 대한 객체가 되며 이들은 서로 다른 방법으로 압축 전송될 수 있다.

SNHC 그룹은 인간의 합성적인 얼굴과 몸체에 대한 표현과 애니메이션에 대한 기술과 TTS(Text to speech synthesis)와 그와 관련된 얼굴 애니메이션에 대한 기술, 텍스트(text)와 그래픽스의 결합, 합성 오디오에 대한 기술, 합성 텍스춰 부호화에 대한 기술 등을 표준화의 범위로 삼고 있으며, 인간의 합성적인 얼굴과 몸체의 표현과 애니메이션의 기술을 위한 얼굴과 몸체에 대한 API(Application Program Interface)로서 FDP와 FAP, BDP(Body Description Parameter)와 BAP(Body Animation Parameter) 등을 정의하고 있다.

표준얼굴은 다양한 움직임과 감정을 표현하는 시스템이므로 가상공간에서 인간의 실제 얼굴을 묘사하는 것은 힘든 작업이다. 얼굴의 기본적인 형태는 3차원 다각형 메쉬들의 결합으로 나타낼 수 있다.

SNHC에서는 얼굴의 모양, 텍스춰(texture), 그리고 표정을 기술하기 위하여 FDP(Facial Definition Parameter)와 FAP(Facial Animation Parameter) 등의 파라미터들을 정의하고 있다.

그림 2는 SNHC에서 정의하고 있는 얼굴의 형태와 애니메이션에 대한 특징점(feature point)을 보여준다.

III. 웹기반 얼굴모델 생성

3.1 얼굴 형상 모델의 구성

일반적으로 얼굴의 3차원 형상을 표현하는 3차원 와이어프레임 모델을 준비한다. 이 모델은 크게 나누어서 얼굴, 두피로 구성되어 있다. 먼저 얼굴의 3차원 형상모델의 정면과 측면을 그림 1에 나타내었다. 이 모델은 약 1703개의 꼭지점과 1038개의 삼각형으로 구성되어 있다.

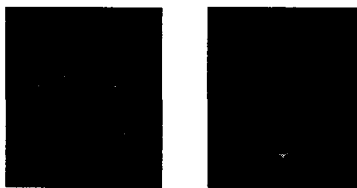
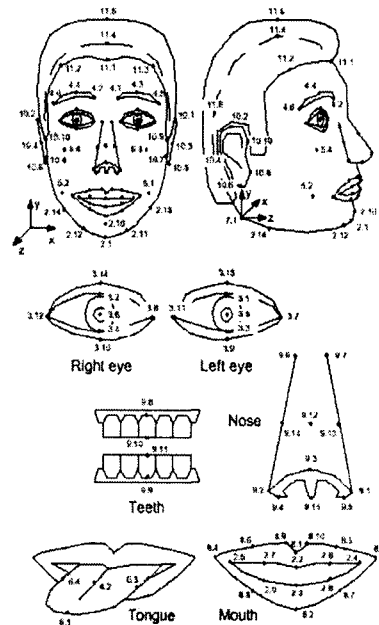


그림 1. 얼굴의 3차원 형상 모델의 측면상과 평면상

일반적인 얼굴의 3차원 형상을 만들기 위하여 모델의 얼굴표면에 종횡으로 선을 그어, 정면, 좌측, 우측으로부터 찍은 2장의 영상으로부터 얼굴의 3차원 형상정보를 계산한다. 형상모델의 정보는 크게 삼각형을 구성 방식을 결정하는 링크 정보와 각 꼭지점의 좌표값으로 나뉜다. 꼭지점의 좌표값 파일에는 각 점의 파라미터 2개와 3차원 x, y, z 값이 함께 저장된다.

3.2 얼굴 및 얼굴 요소의 특징점 추출

3.2.1 MPEG-4(Moving Picture Experts Group-4)



- FAP에 영향을 미치는 특징점
- 그 외의 특징점

그림 2. FDP와 FAP의 특징점

FDP들은 주어진 얼굴 모델을 특정한 얼굴로 만드는데 사용되며, 일반적으로 SNHC비트열내에서는 FDP 뒤에 압축된 FAP가 전송되어 온다. 그러므로, 만약 복호기에서 FDP를 받지 못하더라도 기본적인 얼굴의 형태를 나타낼 수 있는 FAPU를 사용하여 최소한의 작동을 보장할 수 있다.

FDP는 메쉬에서의 3차원 특징점, 3차원 메쉬, 텍스춰영상, 개인적인 특징 등을 포함한다. 3차원 메쉬

자체는 얼굴 모델이 아니라 얼굴의 형태를 정의하고 있으며, 만약 텍스처가 사용되면 텍스처의 좌표가 포함될 수 있다. 그림 1에서 보듯이 얼굴의 형태를 기술하기 위해 각 부분에 특징점들을 정의하고 있으며 이들은 3차원 메쉬에서의 꼭지점들을 나타낸다.

3.2.2 FDP(Facial DefinitionParameter)

얼굴 애니메이션을 지원하는 MPEG-4 디코더는 FAP를 해석할 수 있는 일반적인 얼굴모델을 가지고 있어야 한다. 얼굴의 외모와 형태를 변형하여 특별한 개인 및 캐릭터처럼 보이도록 하는데 FDP가 필요하다. FDP는 일반 얼굴 모델을 특별한 개인 얼굴로 변형하는데 사용된다. FDP는 세션 당 한 번만 전송되며 압축된 FAP의 스트림에 의해서 참조된다.

얼굴 모델러를 통하여 생성된 얼굴 모델을 본 연구에서는 메쉬형태의 데이터 구조에 따라 저장하거나, FDP 파일 형태로 저장이 가능하도록 하였다[12].

얼굴을 표현한 후 애니메이터에서 얼굴 표정을 편집하여 FAP를 자동으로 생성하여 애니메이션에 활용하는 방법을 제안한다.

얼굴 각 부분의 다양한 움직임을 표현하기 위해 FAP에서는 총 66개의 세부적인 얼굴 움직임 항목과 1개의 감정 항목, 1개의 viseme(visual phoneme)이 규정되어 있는데 보다 빠르고 실감나는 얼굴 움직임을 표현할 수 있는 파라미터를 선정한다.

우선 SNHC그룹에서는 얼굴의 애니메이션 파라미터인 FAP를 68개의 파라미터들로 기술하고 있다. 넷째 열의 FAP unit(FAPU)은 그 파라미터를 측정하는 기준이 되는 크기를 의미하며 이것은 얼굴의 기본적인 형태와 밀접한 관계를 갖고 있다. 다섯째 열은 그 파라미터가 단방향인지 양방향인지를 나타내며, 여섯째 열은 어느 방향이 양의 방향인가를 의미하고 있다. 그림 3과 표 1은 얼굴의 애니메이션 파라미터의 측정에 기준이 되는 FAPU를 나타내고 있다.

IV. 웹기반 얼굴 애니메이션

4.1 얼굴 애니메이션 파라미터

얼굴은 다양한 움직임과 감정을 표현하는 시스템이므로 가상공간에서 인간의 실제 얼굴을 묘사하는 것은 힘든 작업이다. 얼굴의 기본적인 형태는 3차원 다각형 메쉬들의 결합으로 나타낼 수 있다.

SNHC에서는 얼굴의 모양, 텍스처(texture), 그리고 표정을 기술하기 위하여 FDP(Facial Definition Parameter)와 FAP(Facial Animation Parameter) 등의 파라미터들을 정의하고 있다. 기본적인 얼굴 객체는 자연스러운 표정의 얼굴 모습을 나타내고 있으며, FAP에 의해 얼굴의 움직임과 표정을 만들 수 있다. FAP는 얼굴의 근육 운동과 밀접히 관련이 있으며, FAP의 파라미터는 FAPU(Facial Animation Parameter Unit)의 항으로 표현된다. 이 단위는 일관된 방법으로 FAP의 해석을 하기 위해서 정의된다.

얼굴 애니메이션 파라미터(FAP)는 얼굴의 각 부분이 무표정한 얼굴에 비하여 얼마나 이동하였는가를 수치로 나타냄으로써 검색하는 얼굴의 표정을 표현한다. 본 논문에서는 입력 얼굴 모델로부터 눈, 눈썹, 입, 코 영역의 얼굴 특징점을 추출하여 개인 얼

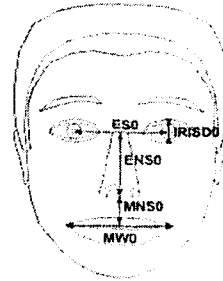


그림 3. FAP units

표 1. FAPU(Facial Animation Parameter Units)의 정의

ESo	눈 사이의 거리	ES=ESo/1024
ENSo	눈과 코 사이의 거리	ENS = ENSo/1024
MNSo	입과 코사이의 거리	MNS = MNSo/1024
MWo	입의 양쪽 거리	MW = MWo/1024

SNHC그룹에서 제공하는 FAP파일은 각각의 프레임별로 68개의 얼굴 애니메이션 파라미터에 대한 마스크 비트(mask bit)가 있어서 어떤 파라미터가 활성화될 것인가를 알려주고 있다.

```
#<frame number> <viseme> <expression> <open_jaw>
... <pull-r_ear>
#Header
2.0 sample 30 300
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 48 0 -140 -65 -29 20 -25 -27 -88 -58 -20 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 58 0 -170 -92 -39 59 18 -99 -18 -99 -70 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
2 49 72 -149 -122 -69 38 38 -66 -66 -75 -66 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
생략()
:
```

그림 4. FAP 파일의 내용

그림 4의 FAP 파일에서 #으로 시작하는 부분은 주석문을 나타내고 주석문이 끝나는 부분의 첫 번째 라인은 전체 FAP파일의 특성을 기술하고 있다. 이 파일은 버전이 2.0이고 FAP파일의 이름은 sample이며 프레임률(frame rate)은 30프레임/초, 총 프레임 수는 300장이다. 68개의 비트로 구성된 두 번째 라인과 세 번째 라인은 첫 프레임의 마스크 비트이며 FAP 68개의 파라미터중에서 어느 파라미터가 활성화 될 것인가를 알려준다. 그림 4에서는 open-jaw, lower_t_midlip, raise_b_midlip, stretch_l_cornorlip, lower_t_lip_lm, lower_t_lip_rm, raise_b_lip_lm, raise_b_lip_rm, raise_l_cornerlip, raise_r_cornerlip, thrust_jaw, shift_jaw, push_b_lip, push_t_lip, depress_chin, head_pitch, head_roll, head_jaw 등 19개의 파라미터가 활성화되었다. 네 번째 라인은 활성화된 얼굴의 애니메이션 파라미터들이 표 1에 기술된 FAPU를 단위로 하여 어느 만큼 이동하였는가의 구체적인 값이 나타나 있다[1]. 그러므로 순차적으로 프레임이 진행됨에 따라 얼굴이 약간 움직이는 모습을 보이게 된다.

이러한 형태로 구성되어 있는 FAP를 분석하여 얼굴의 애니메이션을 구현하는데 있어서 선행되어야

하는 것은 얼굴의 형태에 대한 표현이다. 그러므로 앞에서 설명한 FDP를 이용하여 얼굴 모델을 구현한 후 이를 바탕으로 애니메이션을 추가하였다.

4.2 얼굴 애니메이션

4.2.1 얼굴 모델 편집을 통한 애니메이션

3장에서 서술한 얼굴 모델러를 통하여 생성된 얼굴 모델을 본 연구에서는 메쉬 형태의 데이터 구조로 저장하거나, FDP 파일 형태로 저장이 가능하도록 하였다. 얼굴 모델러로 생성한 얼굴 모델에 대하여 사용자 인터페이스를 통하여 FDP 특징점을 표시하여 사용자가 FDP 특징점을 편집하여 키 프레임 설정하면 이 결과를 FAP로 자동 생성하게 되며 FAP 스트림을 FAP Decoder가 해석하여 애니메이션한다. 이를 위한 절차는 다음과 같으며 그림 5는 그 블록도를 나타낸 것이다.

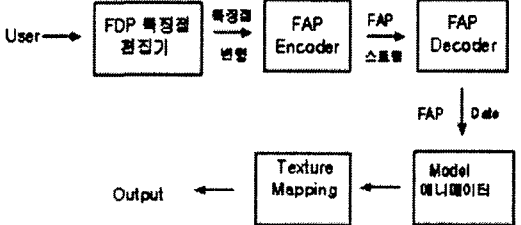


그림 5. 얼굴 모델 편집을 통한 애니메이션 블록도

- 가. 얼굴모델을 로드한다.
- 나. 얼굴모델에 FDP 특징점을 표시한다.
- 다. 키 프레임을 지정한다.
- 라. 눈, 코, 입 등의 특징점을 원하는 표정에 맞게 편집한다.
- 마. FAP를 생성한다.
- 바. 다~마의 과정을 반복하여 각 키프레임에 대한 특징점을 편집하여 FAP를 자동으로 생성한다.
- 사. 선형보간법을 이용하여 인비트윈(In-between)에 대한 FAP를 생성한다.
- 아. FAP Decoder를 통하여 AP를 해석한다.
- 자. 모델을 애니메이션한다.

4.2.2 얼굴 표정 파라미터 편집을 통한 애니메이션 키 프레임 별로 얼굴표정 파라미터를 편집하여

FAP를 자동으로 생성한 다음 이를 FAP Decoder를 통하여 애니메이션하도록 하였다. 얼굴표정은 기쁨(joy), 슬픔(sadness), 화(angry), 두려움(fear), 혐오감(disgust), 놀람(surprise) 등 6가지로 분류하며 표정 편집 패널을 통하여 표정파라미터를 키 프레임 별로 편집하여 이 결과를 FAP로 자동 생성하여 FAP 스트림을 FAP Decoder가 해석하여 애니메이션한다. 이를 위한 절차는 다음과 같으며, 그림 6은 그 블록도를 나타낸 것이고 그림 7은 얼굴모델을 로드하였을 때 얼굴 표정 파라미터를 편집하기 위한 화면을 나타낸 것이다.

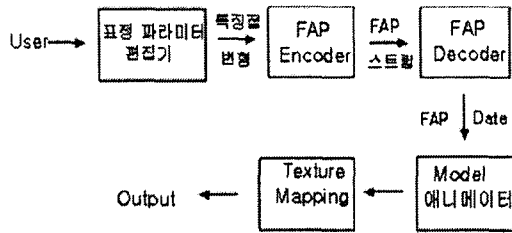


그림 6. 얼굴 표정 파라미터 편집을 통한 애니메이션 블록도

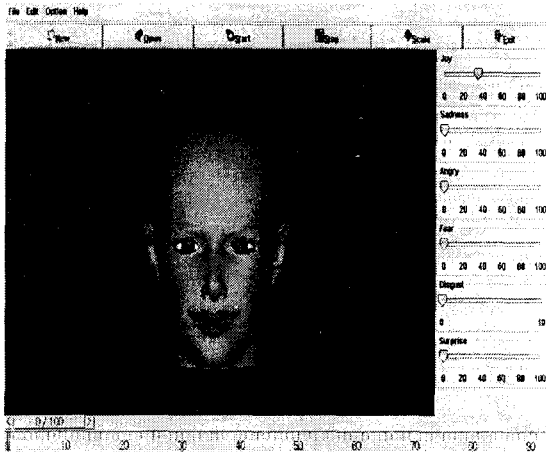


그림 7. 얼굴 표정 파라미터 편집 화면
 가. 얼굴모델을 로드한다.
 나. 얼굴모델에 대한 표정 파라미터를 표시한다. 파라미터값은 0부터 100까지로 한다.
 다. 키 프레임을 지정한다.
 라. 기쁨(joy), 슬픔(sadness), 화(angry), 두려움

(fear), 노여움(disgust), 놀람(surprise) 표정 파라미터를 표정에 맞게 편집한다. 파라미터를 조정함에 따라 얼굴 모델의 표정이 변화한다.

마. FAP를 생성한다.

바. 다~마의 과정을 반복하여 각 키 프레임에 대한 표정 파라미터를 편집하여 FAP를 자동으로 생성한다.

사. 선형보간법을 이용하여 인비트윈(In-between)에 대한 FAP를 생성한다.

아. FAP Decoder를 통하여 FAP를 해석한다.

자. 모델을 애니메이션한다.

4.2.3 얼굴 애니메이션 테이블(FAT)을 통한 애니메이션

빠른 다운로드와 실시간 상호작용성을 위하여 FAP로부터 FAT(Facial Animation Table)을 자동으로 생성하여 애니메이션하였다. 얼굴 애니메이션 테이블은 얼굴 모델이 공간적으로 어떻게 변형되었는지를 FAP의 진폭함수로서 정의한다. 각 얼굴 모델은 꼭지점들의 집합과 꼭지점을 연결하여 생기는 삼각형 정보로 표현된다. 이 정보는 기하학적 표현에서 IndexedFaceSet이라고 한다. 얼굴모델은 텍스처와 위상에 따라서 하나 이상의 IndexedFaceSet으로 구성된다. IndexedFaceSet과 FAP 각각에 대하여 FAT는 어느 꼭지점들이 어떻게 변형되는가를 정의한다. FAP로부터 표 2와 같은 FAT를 자동으로 생성하여 이를 애니메이션에 적용하였다.

표 2. FAT

Vertex n^u	1 st interval $[I_1, I_2]$	2 nd interval $[I_1, I_2]$...
Index 1	변위 D_{11}	변위 D_{12}	...
Index 2	변위 D_{21}	변위 D_{22}	...

4.2.4 Java 3D API를 이용한 애니메이션

애니메이션은 Java 3D API[14]의 Behavior 객체를 이용하여 구현하였다. 이 Behavior 객체를 이용하여 다양한 애니메이션을 구현할 수 있는데, Java 3D는 새로운 클래스를 생성하지 않고도 여러 가지 애니메이션을 구현할 수 있는 클래스들을 제공하고 있다. 물론 이러한 클래스들은 Behavior 클래스에

기반하고 있다.

우선 간단한 애니메이션을 구현하기 위해 사용할 애니메이션 클래스는 보간(interpolation) 클래스이다. 이 보간 클래스는 알파(Alpha) 객체와 함께 사용되어 시간에 기반한 애니메이션을 생성하는데 사용된다.

4.3 얼굴 애니메이션 시스템

본 웹기반 얼굴 애니메이션 시스템은 클라이언트/서버 구조로 클라이언트는 자바 애플릿을 통하여 웹 상에서 접근하고 서버는 자바 애플리케이션으로 접속을 통제하는 접속관리자, 작업 그룹의 동기화를 유지하며 공유된 작업공간을 확보하는 작업관리자, 그리고 3차원 얼굴 모델링을 수행하는 얼굴 모델러와 애니메이션을 수행하는 애니메이터로 이루어져 있다. 클라이언트의 사용자가 본 시스템에 접속하게 되면 자바 애플릿의 얼굴모델러가 실행되며, 사용자는 두 장의 사진을 입력으로 하여 웹브라우저만으로 3차원 얼굴 모델을 생성하고 생성된 얼굴 모델에 대한 애니메이션을 수행하게 된다. 얼굴 모델러와 애니메이터의 3차원 그래픽스 관련 모듈은 자바 3D API를 이용함으로써 웹 기반으로 실행할 수 있다. Java 3D 를 이용함으로써 얻어지는 장점은 3차원 그래픽스 프로그래밍을 위한 시간을 절약할 수 있으며 네트워크 환경에서 삼차원 그래픽을 이용할 수 있다는 점과 플랫폼에 독립적인 특징으로 인하여 성능 확장이 쉽다는 점이다.

4.3.1 시스템 구조

시스템은 클라이언트/서버 구조로 클라이언트는 자바 애플릿을 통해 웹 상에서 접근하고, 서버는 자바 애플리케이션으로 접속을 통제하는 접속관리자, 2차원 얼굴 영상을 받아 3차원 얼굴 모델을 생성하는 모델러, 여기서 얻어진 모델에 표정정보와 음성을 받아서 애니메이션하는 애니메이터로 이루어져 있다. 아래 그림 8은 웹기반 3D 애니메이션 시스템의 전체 구조를 보여주고 있다.

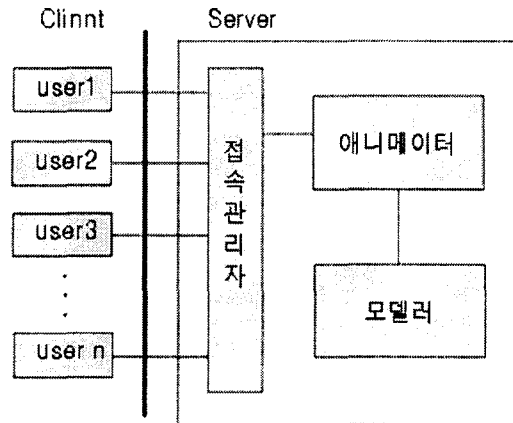


그림 8. 시스템 구조

4.3.2 서버의 구성

가. 접속관리자

접속관리자는 웹서버를 통하여 들어온 클라이언트의 서비스 요청을 받아 분석하여 메시지로 세션 관리자에게 보내고, 처리된 결과를 웹 문서의 형태로 클라이언트에 전송한다. 사용자의 ID에 따라 접속을 설정하거나 해제할 수 있고, 작업하는 동안 클라이언트의 접속을 계속 유지시키는 역할을 한다. 포트 번호를 이용해서 서버 소켓을 생성하고 포트에 접근하는 클라이언트를 기다린다.

여러 이용자가 접근할 수 있도록 각 노드들의 세션 연결에 관한 정보를 가지고 있으면서 클라이언트가 입장하거나 퇴장할 때 세션 정보를 관리한다. 그림 9는 접속관리자의 구조를 보여준다.

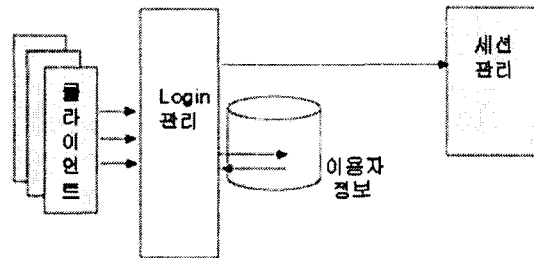


그림 9. 접속관리자 구조

나. 얼굴 모델러

얼굴 모델러는 정면과 좌측, 또는 우측 사진의 2

차원 영상을 먼저 입력받는다. 이 2장의 영상을 얼굴 부분과 배경영상을 색상에 따라 세그멘테이션하여 각각을 분리해 낸다. 분리된 영상 중 얼굴부분에서 눈, 코, 입의 위치와 윤곽선 부분 등의 특징점들을 찾아낸다. 특징점들의 위치를 찾아낸 후 얼굴의 일반 모델에서의 특징점과 입력된 영상에서 추출해 낸 특징점을 이용하여 일반모델을 전체적으로 주어진 영상에 맞게 변형시켜 3D 얼굴 영상을 생성한다. 그림 10은 모델러 구조를 보여주고 있다.

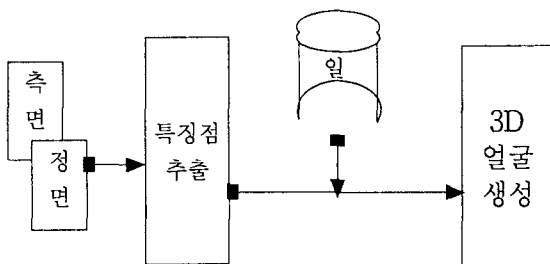


그림 10. 모델러 구조

다. 얼굴 애니메이터

얼굴에 대한 모델이 생성되면 이 모델에 대한 얼굴 표정을 키 프레임 별로 편집하여 자동으로 FAP를 생성하여 애니메이션하는 애니메이터에 의해 애니메이션을 수행하게 된다. 여기에는 얼굴모델 편집기, 얼굴 표정 파라미터 편집기, FAP Decoder 모듈이 있고, 자연스럽게 사실감을 주기 위해 텍스처 매핑을 하여 애니메이션이 되도록 한다.

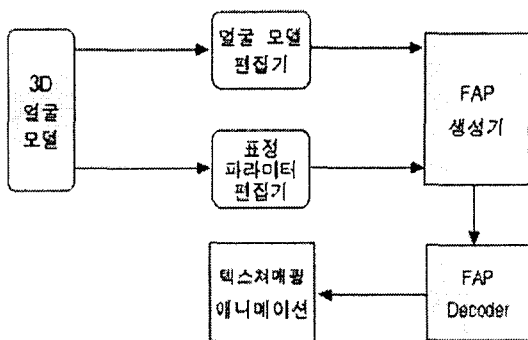


그림 11. 얼굴 애니메이터

본 애니메이션 시스템에서는 사용자가 얼굴 모델을 편집하거나 얼굴 파라미터를 편집할 때 용이한

조작을 위해서 사용자가 모델의 특징점이나 표정 파라미터를 변화시키면 파라미터의 값에 따라 얼굴의 각 부위를 움직이며 그 결과를 웹브라우저에 나타내도록 한다. 그림 11은 얼굴 애니메이터의 구조이다.

라. FAP Encoder와 Decoder 모듈

FAP Encoder는 얼굴 모델을 특징점을 이용하여 키 프레임 별로 편집하거나 얼굴 표정 파라미터를 편집할 때마다 FAP 파라미터를 자동으로 생성하게 된다.

FAP Decoder는 생성된 FAP를 해석하여 움직이는 얼굴 표정을 만들어 낸다.

V. 결론

본 연구에서는 기존의 방법에 비하여 3차원 스캐너나 카메라를 이용하지 않고 비용과 노력을 크게 절감하면서 실감나는 얼굴 모델링을 효율적으로 수행하는 웹 기반 3차원 얼굴 모델러와 애니메이터를 개발하였다. 사용자는 두 장의 사진을 입력으로 하여 웹브라우저만으로 3차원 얼굴 모델을 생성하게 되며, 생성된 3차원 모델에 대하여 웹 기반으로 애니메이션을 수행하게 된다. 얼굴 모델러와 애니메이터의 3차원 그래픽스 관련 모듈은 자바 3D API를 이용하였다.

두 장의 사진으로 3차원 얼굴 모델을 생성하기 위하여 얼굴 영역 내에서 색상 정보와 경계선 정보를 활용하여, MPEG-4의 SNHC에서 제안하고 있는 개개의 얼굴 특성을 표현하는 43개의 얼굴 특징점 파라미터 중 2차원 영상에서 추출 가능한 31개의 FDP를 추출하여 사용하였다. 추출된 2차원 얼굴 특징점들을 1038 개의 삼각형 메쉬로 이루어진 3차원 일반 얼굴 모델에 적용시켜 변형함으로써 개개의 얼굴에 해당하는 모델을 생성하였다.

얼굴에 대한 모델이 생성되면 이 모델에 대한 얼굴 표정을 키 프레임 별로 편집하여 자동으로 FAP를 생성하여 애니메이션하는 애니메이터에 의해 애니메이션을 수행하게 된다. 여기에는 얼굴모델 편집기, 얼굴 표정 파라미터 편집기, FAP Decoder 모듈이 있고, 자연스럽게 사실감을 주기 위해 텍스처 매

핑을 하여 애니메이션이 되도록 한다.

FAP Encoder는 얼굴 모델을 특징점을 이용하여 키 프레임 별로 편집하거나 얼굴 표정 파라미터를 편집할 때마다 FAP 파라미터를 자동으로 생성하게 된다. FAP Decoder는 생성된 FAP를 해석하여 움직이는 얼굴 표정을 만들어 낸다.

이 시스템은 게임이나 웹 상에서 자신을 나타낼 수 있는 아바타를 제작하기 위한 도구로 사용하거나, 3D 캐릭터의 제작, 영화, 컴퓨터 애니메이션 등과 같은 다양한 부분에 응용될 수 있다.

참고문헌

[1] 장학균, 김성엽, 연금숙, 한경희, 전영훈, 김미진, "인터넷상에서 가상환경 구현을 위한 3차원 애니메이션 저작도구 개발에 관한 연구", 정보통신부 연차보고서, 1999. 7.

[2] 우중제, "MPEG-4에 있어서 얼굴 표정 합성과 실시간 얼굴 동영상 애니메이션", 명지대학교 석사학위 논문 pp. 32-34, 1998.

[3] 김진우, 고한석, 김형근, 권용무, 안상철, "MPEG-4 기반 3차원 얼굴 모델 애니메이션".

[4] 안용기, "MPEG-4 SNHC에 있어서의 얼굴과 몸체의 애니메이션 설계 및 구현", 한양대학교 대학원 석사학위 논문, 1997.

[5] T. Goto, M. Escher, C. Zanardi, N. M. Thalmann, "MPEG-4 based Animation with Face Feature Tracking", CAS '99(Eurographics Workshop), pp. 89-98, Sept. 1999.

[6] A. Eliassen, S. Forchhammer, N. J. Dyhr, F. Nielsen", Danish Visual Speech Synthesis & MPEG-4 Facial Animation Plug-in for Realplayer G2", International Workshop on Synthetic-Natural Hybrid Coding and 3 Dimensional Imaging (IWSNHC3DI'99), pp. 37-40, Sept. 1999.

[7] 안용기, 이원석, 정제창, 이주상, 안정환, 호요성, 류지상, 이명호, 안치득, "MPEG-4 SNHC에 있어서의 얼굴과 몸체의 애니메이션 설계 및 구현", 1997년도 통신학회 추계종합학술발표회 논문집, 제16권 제2호, pp. 1116-1119, 1997년 11월.

[8] 한태우 "실감있는 얼굴 표정 애니메이션을 위한 3차원 얼굴 모델링 시스템", 한국과학기술원 석

사학위 논문, 1997.

[9] S.M. Platt and N. Badler, "Animating Facial Expressions," Computer Graphics, Vol.15, No. 3, 1981.

[10] K. Waters, "A Muscle Model for Animating Three-Dimensional Facial Expression," Computer Graphics, Vol. 21, No. 4, 1987.

[11] C.T. Waite, "Facial Action Control Editor, Face: a Parametric Facial Expression Editor for Computer Generated Animation," Master Thesis, MIT, 1990.

[12] 김응곤, 윤보열, 송승현, 정선인, 김종찬, 김세희, "웹기반 얼굴애니메이션 소프트웨어 개발", 정보통신부 2000년도 대학기초사업 최종 연구개발 보고서.

[13] <http://java3d.co.kr>, Java 3D 튜토리얼.

저자소개

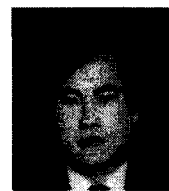
박경숙(Kyong-sook Park)

1982년 2월 조선대학교 이학사
 1985년 2월 조선대학교 이학석사
 1993년 ~ 현재 한영대학 인터넷 소프트웨어과 교수
 ※ 관심분야: 컴퓨터그래픽스, VR

허영남(Yeong-nam Heo)

순천대학교 컴퓨터과학과 교수
 ※ 관심분야: 컴퓨터그래픽스, 영상처리, VR

김응곤(Eung-kon Kim)



1980년 2월 조선대학교 공학사
 1987년 2월 한양대학교 공학석사
 1992년 2월 조선대학교 공학박사
 1984년 8월 ~ 1986년 8월 금성반도체(주) 연구원

1987년 3월 ~ 1991년 2월 국방과학연구소 선임연구원
 1991년 3월 ~ 1993년 2월 여수수산대학교 컴퓨터공학과 전임강사
 1997년 3월 ~ 1998년 2월 미국 University of California, Santa Cruz Post Doc.
 1993년 3월 ~ 현재 순천대학교 컴퓨터과학과 부교수
 ※ 관심분야: 컴퓨터그래픽스, 영상처리, 가상현실