

## 3차원 모델을 사용한 애니메이션 캐릭터 얼굴의 합성

장 석 우\*, 김 계 영\*\*

### Synthesizing Faces of Animation Characters Using a 3D Model

Seok-Woo Jang \*, Gye-Young Kim\*\*

#### 요 약

본 논문에서는 3차원의 얼굴모델을 생성하여 사용자의 얼굴을 애니메이션 캐릭터의 얼굴에 자연스럽게 합성하는 새로운 방법을 제안한다. 제안된 방법에서는 먼저 정면과 측면의 직교하는 2장의 2차원 얼굴영상을 입력 받아 얼굴의 주요 특징을 템플릿 스테이크를 이용하여 추출하고, 추출된 특징점에 맞게 일반적인 3차원 얼굴 모델을 변형시킴으로써 사용자의 얼굴 형태에 적합한 얼굴 모델을 생성한다. 그리고 2장의 얼굴 영상으로부터 얻어지는 텍스처 맵을 3차원의 얼굴 모델에 매핑하여 현실감 있는 개인화된 얼굴 모델을 생성한다. 그런 다음, 개인화된 3차원의 얼굴모델을 애니메이션 캐릭터 얼굴의 위치, 크기, 표정, 회전 정보를 반영하여 캐릭터 얼굴에 자연스럽게 합성함으로써 현실감 있는 사용자 맞춤형 애니메이션을 제작한다. 실험에서는 제안된 캐릭터 얼굴 합성 방법의 성능을 검증하기 위해서 수행한 여러 가지 실험결과를 보인다. 본 논문에서 제안된 방법은 애니메이션 영화, 게임, 캐릭터를 이용한 여러 가지 응용 분야에서 유용하게 활용될 것으로 기대된다.

▶ Keywords : 얼굴 합성, 애니메이션 캐릭터, 사용자 맞춤형, 3차원 얼굴모델

#### Abstract

In this paper, we propose a method of synthesizing faces of a user and an animation character using a 3D face model. The suggested method first receives two orthogonal 2D face images and extracts major features of the face through the template snake. It then generates a user-customized 3D face model by adjusting a generalized face model using the extracted facial features and by mapping texture maps obtained from two input images to the 3D face model.

---

• 제1저자 : 장석우 • 교신저자 : 김계영  
• 투고일 : 2012. 6. 2, 심사일 : 2012. 6. 20, 게재확정일 : 2012. 7. 2.  
\* 안양대학교 디지털미디어학과(Dept. of Digital Media, Anyang University)  
\*\* 숭실대학교 컴퓨터학부(School of Computing, Soongsil University)

Finally, it generates a user-customized animation character by synthesizing the generated 3D model to an animation character reflecting the position, size, facial expressions, and rotational information of the character. Experimental results show some results to verify the performance of the suggested algorithm. We expect that our method will be useful to various applications such as games and animation movies.

▶ Keywords : Face Synthesis, Animation Character, User-Customized, 3D Face Model

## 1. 서 론

멀티미디어와 관련된 하드웨어 및 소프트웨어 기술의 빠른 발달로 그래픽, 동영상, 사운드, 애니메이션 등 다양한 종류의 디지털 미디어에 대한 수요가 지속적으로 증가하고 있다. 이 중에서 애니메이션(animation)은 컴퓨터 그래픽스 기술을 이용하여 사물의 움직임을 표현하기 위한 기법으로, 어떤 현상의 변화나 움직임을 강조하거나 현실 세계에서 쉽게 만들 수 없는 현상을 보이기 위하여 매우 효과적으로 사용될 수 있다. 뿐만 아니라, 애니메이션은 미적으로도 사람들에게 크게 호소할 수 있어서 매우 효과적인 미디어이다[1]. 따라서 애니메이션 콘텐츠의 사용 연령층이 점점 더 넓어지고 있으며, 그 응용 분야도 조금씩 증가하고 있는 추세이다.

기존의 문헌을 살펴보면 애니메이션과 관련된 많은 연구들을 찾아보기는 어렵지만 조금씩 진행되고 있다는 것을 확인할 수 있다. [2]에서는 2차원의 애니메이션 캐릭터 영상을 분석하여 캐릭터의 얼굴 부분을 추출한 후 기준 좌표계 및 대상 좌표계의 기하학적인 관계를 이용하여 3차원적인 얼굴의 회전정보를 측정하였다. 그런 다음, 측정된 애니메이션의 회전정보를 이용하여 3차원의 얼굴 모델로 표현하는 방법을 제안하였다. [3]에서는 단순화된 메쉬(mesh) 모델과 mpeg7에서 정의된 주색상(dominant color) 정보를 이용하여 애니메이션 캐릭터의 얼굴과 얼굴의 구성요소를 효과적으로 검출하는 방법을 제안하였다. [4]에서는 의미전달의 중요한 요소인 얼굴의 표정에 대한 상세한 이해에 대해 기술하였으며, 스타일화된 애니메이션 캐릭터와 사실적 애니메이션 캐릭터의 여러 가지 표정연출 유형에 관한 연구를 진행하였다. [5]에서는 애니메이션 캐릭터의 얼굴 표정을 결정짓는 결정요소, 즉 캐릭터의 눈, 코, 입, 귀와 얼굴 표정의 관계에 대한 연구를 진행하였으며, 결과적으로 눈이 가장 큰 영향을 미치며 다음으로는 입이 큰 영향을 미치는 것으로 조사되었다.

위에서 기술했듯이 애니메이션에 관한 연구가 조금씩 진행

은 되고 있으나 그 수준이 아직은 미미한 단계이며, 관련된 응용 분야들도 많지 않은 실정이다. 따라서 본 논문에서는 애니메이션과 관련된 새로운 응용 분야를 제안하고자 한다. 다시 말해, 애니메이션에 관한 성향이 강한 사람이라면 그 애니메이션의 주인공이 '나'였으면 하는 욕구를 누구나 한번쯤은 느껴 보았을 것이다. 이것은 사람들이 애니메이션 뿐만 아니라 드라마나 영화를 보는 중요한 원인 중의 하나이기도 하다. 즉, 어떠한 '주인공'과 '나'를 동시에 여겨 가상의 세계에서 대리만족을 하고 싶어 한다는 것이다.

이런 사실을 근거로 본 논문에서는 사용자의 얼굴을 3차원으로 모델링하고, 이를 2차원 기반인 애니메이션 캐릭터의 얼굴에 정합 및 합성하여 사용자 맞춤형으로 애니메이션을 재생성하는 방법을 제안한다. 그림 1은 본 논문에서 제안된 사용자 얼굴과 애니메이션 캐릭터의 얼굴을 합성하는 알고리즘의 전체적인 개요도를 보여준다.

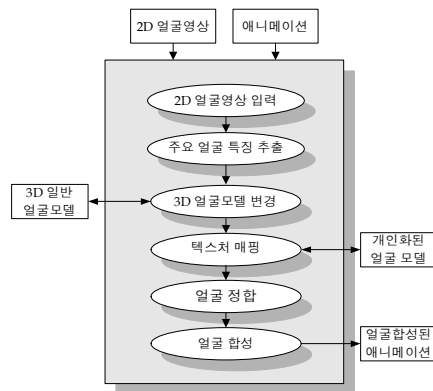


그림 1. 전체 개요도  
Fig. 1. Overall Flow

그림 1에서와 같이 제안된 알고리즘에서는 먼저 정면과 측면의 직교하는 2장의 2차원 얼굴영상을 입력 받아 얼굴의 주요 특징을 템플릿 스테이크[6]를 이용하여 자동으로 추출하고, 추출된 특징점에 적합하게 일반적인 3차원의 얼굴모델을 변형시킴으로써 사용자 얼굴의 형태에 적합한 3차원의 얼굴

모델을 생성한다. 그리고 2장의 얼굴 영상으로부터 얻어지는 텍스처 맵을 3차원의 얼굴 모델에 매핑하여 현실감 있는 개인화된 얼굴 모델을 생성한다. 그런 다음, 개인화된 3차원의 얼굴 모델을 애니메이션 캐릭터 얼굴의 위치, 크기, 표정, 그리고 회전 정보를 반영하여 자연스럽게 합성함으로써 현실감 있는 사용자 맞춤형 애니메이션을 제작한다.

1장에서는 본 연구를 수행하게 된 동기 및 배경, 그리고 전체적인 개요를 기술하였다. 2장에서는 사용자의 얼굴 특징에 맞게 3차원의 얼굴 모델을 생성하는 방법에 대해 설명한다. 3장에서는 RBF(radial basis function) 보간 기법을 이용하여 3차원의 얼굴 모델의 형태를 보다 자연스럽게 변형하는 방법에 대해 설명한다. 4장에서는 3차원 얼굴모델을 애니메이션 얼굴의 위치, 크기, 표정, 그리고 회전 정보를 반영하여 합성함으로써 사용자 맞춤형 애니메이션을 제작하는 방법에 대해 기술한다. 그리고 5장에서는 본 논문에서 제안된 방법의 성능을 평가하기 위해서 수행한 실험결과를 보이며, 6장에서는 결론 및 향후 연구방향을 제시한다.

### II. 3차원의 얼굴모델 생성

본 논문에서는 기본적인 3차원의 얼굴 모델을 생성하기 위해서 MPEG-4 SNHC(synthetic/natural hybrid coding) FBA (face and body animation) 그룹에서 표준화된 얼굴 객체의 FDP(facial definition parameters)를 이용한다[7]. FDP는 인간의 얼굴 객체를 묘사하기 위해서 얼굴 모델을 정의하는데, 얼굴의 3차원적인 특징점 정보를 포함하고 있다. 다시 말해, FDP는 실제 인간의 얼굴을 대상으로 해서 개개인의 얼굴 특징을 84개의 특징점을 이용하여 표현하였다. 즉, 최소한 이 84개의 특징점만을 이용하면 개개인의 독특한 얼굴 특징을 표현할 수 있다는 것인데, 그림 2는 FDP의 일부를 보여준다. 그림 2의 (a)는 정면 얼굴에 대한 FDP를 보여주고, 그림 2의 (b)는 혀에 대한 FDP, 그리고 그림 2의 (c)는 입에 대한 FDP를 보여준다.

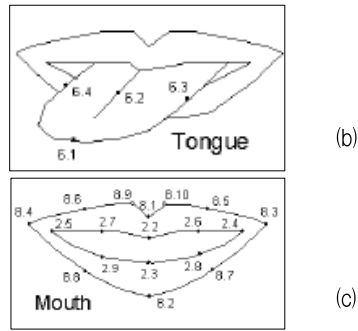
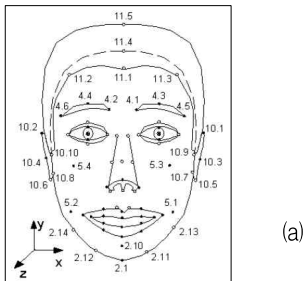


그림 2 Mpeg-4의 FDP  
Fig. 2 Mpeg-4 FDP

Mpeg-4의 FDP는 기본적으로 치아, 눈동자, 그리고 혀에 대한 특징점도 정의하고 있으나 본 논문에서는 그림 3의 (a)와 같이 얼굴의 정면과 옆면에서 보이는 특징점만을 활용하며, 이 특징점들 중에서 보다 중요하다고 판단되는 특징점들을 다시 선택하여 그림 3과 같이 간소화된 특징점을 가지는 3차원의 얼굴 모델을 생성하였다. 그림 3의 간소화된 3차원 얼굴 모델은 46개의 특징점, 그리고 5,491개의 정점(vertex)과 10,914개의 메쉬(mesh)로 구성되어 있다. 그림 3에서 흰색의 점들이 46개의 특징점을 표시하고 있다.



그림 3. FDP 기반의 3D 얼굴모델  
Fig. 3. 3D Face Model based on FDP

이와 같이 간소화된 3차원의 얼굴 모델을 생성한 후에는 그림 4와 같이 사용자에게 대한 정면 얼굴 영상을 받아들여 3차원 얼굴 모델의 특징점에 대응하는 점들을 템플릿 스네이크(template snake)[6]를 활용하여 추출하고, 이를 3차원의 얼굴 모델에 투영한다. 그리고 템플릿 스테이크는 기존 스네이크 알고리즘의 에너지 항에 변화를 주는 방법인데, 식 (1)은 템플릿 스네이크 중에서 눈과 눈썹의 윤곽선을 강건하게

추출하기 위한  $E_{eye-snake}$ 에 대한 수식을 보여주고 있다. 이외의 템플릿 스네이크에 대한 보다 자세한 내용은 참고문헌 [6]에 나와 있다.

$$E_{eye-snake} = \int_0^1 (\alpha(s) \left| \frac{dv}{ds} \right|^2 + \beta(s) \left| \frac{d^2v}{ds^2} \right|^2 - (\gamma_1 |\nabla I(v)| + \gamma_2 |\nabla^2 I(v)|) ds) ds \quad (1)$$

그리고 사용자의 측면 영상에 대해서도 위와 동일한 처리를 수행하여 3차원의 얼굴 모델을 생성한다. 본 논문에서는 3차원 얼굴 모델이 투영된 영상에 일정 부분 맞지 않거나 부자연스러운 형상을 유지할 경우에는 3차원 모델의 특징점을 마우스를 이용하여 수동적으로 이동시켜 3차원의 얼굴 모델의 형태를 자연스럽게 조정할 수 있는 기능도 추가하였다.



그림 4. 얼굴 특징점 추출  
Fig. 4. Extraction of Facial Feature Points

### III. 보간을 통한 얼굴모델의 변형

템플릿 스네이크를 통해 추출된 2차원 얼굴의 특징점들을 기반으로 3차원 표준 얼굴모델의 형태를 변형하는 과정에서 얼굴모델의 자연스러운 변형을 위해 RBF(radial basis function) 보간 기법을 활용한다. 일반적으로, RBF를 이용한 방법은 최소한의 정점으로부터 부드러운 기하학적인 변형을 할 수 있으며, 얼굴 변형을 쉽고 빠르게 할 수 있다고 알려져 있다[8].

본 논문에서는 기 추출된 얼굴의 특징점 정보가 사용자 얼굴에 대한 위치 및 형태 정보를 포함하고 있으므로 3차원 표

준 얼굴모델의 대응하는 정점들과의 정합을 통해 3차원 얼굴 모델의 형태를 사용자의 얼굴을 반영하도록 조정할 수 있다. 그리고 이 조정 과정에서 RBF 보간 기법을 활용하는데, RBF 함수는 3차원 표준 얼굴모델 상의 정점들을 대응하는 사용자의 얼굴 특징점의 위치로 이동하고, 각 특징점의 영향을 받는 영역 내의 정점들은 중간적인 위치로 사상시킨다.

일반적으로, RBF 함수는 변형하고자 하는 특징점들을  $f(\vec{x})$ 라고 한다면 RBF에 의한 보간 함수는  $F(\vec{x}, \vec{c})$ 로 정의할 수 있다. 여기서,  $\vec{x}$ 는 변형하고자 하는 특징점을 나타내는 벡터 ( $\vec{x} = x_1, x_2, \dots, x_n$ )를 의미내고,  $\vec{c}$ 는 각 특징점에 대한 가중치 벡터( $\vec{c} = c_1, c_2, \dots, c_n$ )를 의미한다. RBF 보간 함수는 수학적으로 식 (2)와 같이 표현할 수 있다.

$$F(\vec{x}) = \sum_{i=1}^N [c_i \times h(r_i)] \quad (2)$$

식 (2)에서 N은 정의된 특징점의 개수이고, h(r)은 RBF의 기저함수를 나타내며, r은 특징점과 정점들 간의 유클리디안 거리를 의미한다. RBF의 보간함수로는 선형거리( $r_i$ ), 다중이차( $\sqrt{r^2 + c^2}$ ), TPS(thin plate spline:  $r^2 \log r$ ), 가우시안 기저함수( $e^{-r_i^2/\sigma_i}$ ) 등이 있는데, 본 논문에서는 얼굴 형태 변형에 적합하다고 알려진 가우시안 기저함수를 사용한다. 따라서 가우시안 기저함수를 이용해 식 (2)를 전개하면 식 (3)과 같이 표현된다.

$$F(\vec{x}) = \sum_{i=1}^N [c_i \times e^{-r_i^2/\sigma_i}] \quad (3)$$

식 (3)의 해를 구하기 위해서는 특징점( $x_i, y_i, z_i$ ), 가중치( $c_i$ ), 너비 매개변수( $\sigma_i$ )의 값을 설정해야 하는데, 특징점은 기 설정되어 있으므로 가중치와 너비 매개변수를 계산해야 한다. 먼저, 가중치는 특징점들의 변위를 이용하여 계산할 수 있다. 즉, 특징점들의 초기 위치를  $\vec{x}_{source}$ 라 하고 변형된 후의 특징점들의 위치를  $\vec{x}_{target}$ 라 한다면 모든 특징점들의 초기 위치와 변형 후의 위치를 구할 수 있으므로 식 (4)를 이용하여 특징점에 대한 가중치를 계산할 수 있다.

$$\vec{c} = h^{-1} \times \vec{x}_{target} \quad (4)$$

그리고 특징점들이 이동함에 따라 주변에 위치한 정점들이 받는 영향의 범위와 관련된 너비 매개변수  $\sigma_i$ 는 식 (5)와 같이 각 특징점과 정점 사이의 유클리디안 거리 중에서 가장 먼

거리 값을 이용하여 계산한다. 다시 말해, 특징점과 가까운 거리에 있는 정점들에는 큰 가중치를 부여하고, 특징점과 먼 거리에 있는 정점들에는 작은 가중치를 부여하여 특징점의 이동에 따라 정점들의 부드러운 변형이 발생되도록 유도한다.

$$\sigma_i = \max_{i \neq j} \left\| \vec{x}_{i_{source}} - \vec{x}_j \right\| \times weight \quad (5)$$

RBF 보간을 통해 3차원의 얼굴모델을 조정된 후에는 사용자에게 대한 정면과 측면 영상을 얼굴모델에 텍스처(texture) 매핑하여 그림 5와 같은 개인화된 3차원 얼굴모델을 생성한다.



그림 5. 개인화된 3차원 얼굴모델  
Fig. 5. Personalized 3D Face Model

일반적으로, 3차원의 얼굴모델에 텍스처 데이터를 투영하는 방법에는 다시점으로 입력영상을 촬영하고, 다수의 입력영상을 구형, 상자형, 원통형 텍스처로 재 생성한 후, 재 생성된 텍스처를 3차원의 모델에 매핑하는 방법을 많이 사용한다[9]. 그런데 본 논문에서는 정면과 측면의 2장의 입력영상만을 사용하므로 구형, 원통형 등의 텍스처로 생성할 때 왜곡이 발생하기 쉽다. 따라서 제안된 시스템에서는 정면 영상을 그대로 3차원 모델의 정면에 투영하고, 측면 영상 역시 모델의 측면에 그대로 투영한 후 정면과 측면을 구분지어 텍스처 매핑을 수행하였다. 다시 말해, 모델의 경계를 정면과 측면으로 구분함으로써 하나의 면이 여러 텍스처를 혼합하는 방식이 아니라 정면 혹은 측면 중에서 선택된 하나의 텍스처를 사용하여 다중 텍스처의 보간 없이 텍스처링 할 수 있도록 하였다. 그리고 정면과 측면 텍스처의 경계를 구분하는 방법은 Ferradai [10]에서 제안한 특징점을 기준으로 정면과 측면의 경계를 구분하는 방법을 사용하였다.

## IV. 애니메이션 캐릭터의 얼굴 합성

이전 단계에서 획득한 사용자의 얼굴을 최대한 반영한 개인화된 3차원의 얼굴모델을 애니메이션 캐릭터의 얼굴로 대체하기 위해서는 애니메이션 캐릭터 얼굴과 얼굴의 구성요소를 검출해야 하며, 이를 바탕으로 애니메이션 캐릭터 얼굴의 위치, 크기, 표정, 회전정보를 인식해야 한다.

먼저, 애니메이션 캐릭터의 얼굴과 얼굴의 구성요소를 검출하기 위해서 그림 6과 같이 기존의 얼굴 메쉬모델의 형태를 애니메이션 캐릭터 얼굴의 특성에 맞게 간략화한 캐릭터 얼굴 메쉬모델을 정의하여 사용한다[3]. 본 논문에서는 애니메이션 캐릭터 얼굴의 3가지 구성요소인 눈썹, 눈, 입을 부각시키고, 나머지 구성요소는 제거함으로써 필수적인 특징점들만으로 구성된 애니메이션 캐릭터 얼굴의 메쉬모델을 생성한다.

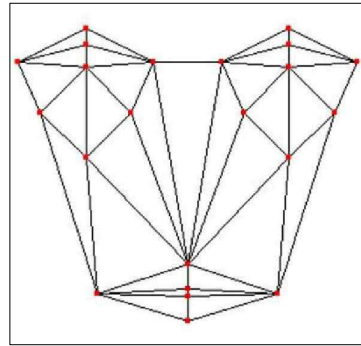


그림 6. 간략화된 메쉬모델  
Fig. 6. Simplified Mesh Model

그런 다음, 캐릭터 얼굴이 가진 가장 고유한 특징 중의 하나인 색상정보를 이용해 얼굴 및 얼굴의 구성요소를 검출하는데, 본 논문에서는 Mpeg-7에서 정의된 주색상 서술자인 DCD (dominant color descriptor)를 활용한다[11]. 주색상 서술자 F는 식 (6)과 같이 각 주색상의 값  $c_i$ , 주색상 값을 가지는 화소의 비율  $p_i$ , 주색상 값의 분산인  $v_i$ , 그리고 공간적인 색상의 응집도인  $s$ 로 정의된다. 또한,  $i$ 는 주색상을 나타내는 인덱스이다.

$$F = \{(c_i, p_i, v_i), s\}, \quad (i = 1, 2, \dots, N) \quad (6)$$

본 논문에서는 학습단계에서 그림 7과 같이 애니메이션 캐릭터 얼굴의 구성요소에 맞도록 메쉬모델을 수작업으로 정합시킨 후, 메쉬모델에 포함된 얼굴의 구성요소인 눈썹, 눈, 입에 대한 주색상을 추출한다. 그리고 메쉬모델에 포함되지 않은 영역의 주색상을 활용하여 캐릭터 얼굴에 대한 주색상을 추출한다. 그런 다음, 인식단계에서 새롭게 입력되는 애니메이션 영상으로부터 기 학습된 주색상 정보를 이용하여 캐릭터의 얼굴과 얼굴 구성요소의 후보를 그림 8과 같이 각각 추출한 후 유사도 분석[3]을 통해 최종적인 얼굴과 얼굴의 구성요소를 획득한다.



그림 7. 얼굴과 구성요소의 주색상 추출  
Fig. 7. Dominant Color Extraction of Face and Its Components



그림 8. 얼굴과 구성요소의 후보영역 추출  
Fig. 8. Extraction of Face and Its Component Candidates

주색상을 이용하여 애니메이션 캐릭터 얼굴과 얼굴의 구성요소를 추출한 후에는 애니메이션 캐릭터 얼굴의 표정을 인식한다[11]. 먼저, 이전 단계에서 추출된 캐릭터 얼굴의 각 구성요소 내에서 수평과 수직 에지에 민감하게 반응하는 프리윗(Prewitt) 에지 검출기를 이용하여 표정인식을 위한 특징점을 추출한다. 즉, 각 구성요소로부터 각 요소를 가장 잘 특징짓는 22개의 특징점을 추출하는데, 눈썹으로부터는 8개의 특징점(눈썹의 좌우 끝점 4개, 눈썹 중앙 부분의 위아래 특징점 4개)을, 눈으로부터는 8개의 특징점(눈의 좌우 끝점 4개, 눈

중앙부분의 위아래 특징점 4개)을, 그리고 입으로부터는 6개의 특징점(입의 좌우 끝점 2개, 윗 입술 중앙 부분의 상하 끝점 2개, 아랫 입술 중앙 부분의 상하 끝점 2개)을 추출한다. 그런 다음, 각 구성요소로부터 추출된 특징점들의 위치와 모양 정보를 신경망 학습을 통해 해당 AU(action unit)로 분류하고, 표정 AU 명세서를 이용하여 최종적으로 캐릭터의 표정을 인식한다. 본 논문에서는 캐릭터의 얼굴 형태를 최대한 고려하여 구성한 14개의 단일 AU와 8개의 복합 AU, 즉 총 22개의 AU를 신경망을 기반으로 학습하고, 정의된 표정 AU 명세서를 이용한 AU 조합을 통해 표정을 인식한다[11]. 그림 9는 애니메이션 캐릭터 얼굴에 대한 표정인식 결과의 예를 보여주고 있다.

애니메이션 캐릭터 얼굴의 3차원 회전정보 측정은 캐릭터 얼굴을 원형체라고 가정하고 캐릭터의 눈을 기준으로 얼굴의 상하좌우 및 기울어진 정도를 측정함으로써 구한다[2]. 제안된 방법에서는 어느 축으로도 기울어지지 않은 상태의 좌표계를 기준좌표계라고 명명하며, 캐릭터의 머리와 대응하는 기울어진 상태의 좌표계를 대상좌표계라고 명명한다. 그림 10에서 기울어지지 않은 파란색의 수평/수직선이 기준좌표계를 나



그림 9. 표정인식 결과  
Fig. 9. Results of Facial Expression Recognition

타내고, 기울어져 있는 빨간색의 수평/수직선이 대상좌표계를 나타낸다.



그림 10. 기준 및 대상 좌표계  
Fig. 10. Basis and Target Coordinate System

그런 다음, 기준좌표계를 기준으로 캐릭터에 대응하는 대상좌표계가 좌우로 이동한 정도로 얼굴의 좌우회전 정도를 측정하며, 상하로 이동한 정도를 분석하여 얼굴의 상하회전 정도를 측정한다. 그리고 대상좌표계의 세로축이 기울어져 있는 정도로 얼굴의 기울어진 정도를 측정한다.

$$S = -\tan^{-1}m \quad (7)$$

$$P = -\tan^{-1} \frac{ps}{\sqrt{pd^2 - ps^2}} \quad (8)$$

$$T = -\tan^{-1} \frac{ts}{\sqrt{td^2 - ts^2}} \quad (9)$$

식 (7), (8), (9)는 얼굴의 회전정보를 추출하는 식을 나타낸다. 식 (7), (8), (9)에서 S(swing)는 z축을 기준으로 캐릭터 얼굴이 회전한 정도를 나타내고, P(panning)는 y축을 기준으로 얼굴이 좌우로 회전한 정도를 나타내며, T(tilting)는 x축을 기준으로 얼굴이 상하로 회전한 정도를 나타내는 변수로 사용된다. 식 (8)에서  $pd$ 는 얼굴에 해당하는 원형체의 가로축 반지름이며,  $ps$ 는 얼굴이 기준 좌표계의 중심에서부터 대상 좌표계까지 x축으로 이동한 거리이다. 따라서 피타고라스의 정의를 이용하여  $P$ 를 구한다. 식 (9)에서  $ts$ 는 얼굴이 기준 좌표계의 중심에서부터 대상 좌표계까지 y축으로 이동한 거리이며,  $td$ 는 원형체의 세로축 반지름이다. 마찬가지로 피타고라스의 정의를 이용하여  $T$ 를 구한다. 마지막으로  $m$ 은 대상 좌표계의 중심을 기준 좌표계의 중심으로 평행 이동하였을 때의 직선의 기울기이며, 이 기울기  $m$ 에  $\arctan$ 을 적용하고 각도로 변환하여  $S$ 를 구한다.

이와 같이 애니메이션 캐릭터 얼굴과 얼굴 구성요소의 위

치 및 크기, 캐릭터 얼굴의 표정인식 결과, 캐릭터 얼굴의 x, y, z 방향으로의 회전정보를 추출한 후에는, 3장에서 생성한 개인화된 3차원의 얼굴모델을 위에서 추출한 정보에 맞게 조정 후 애니메이션 캐릭터 얼굴에 합성함으로써 사용자 맞춤형의 애니메이션 캐릭터를 재생성할 수 있다.

## V. 실험결과

본 논문의 실험을 위해 사용한 컴퓨터는 인텔 Pentium Core 2 Duo의 CPU와 4GB의 메모리를 사용하였고, 운영체제는 마이크로소프트사의 윈도우 XP Professional을 사용하였다. 그리고 마이크로소프트사의 Visual C++ 2008 컴파일러와 영상처리 오픈 라이브러리인 Open CV를 사용하여 제안된 애니메이션 캐릭터의 얼굴 합성 알고리즘을 구현하였다. 본 논문에서는 75분짜리 신데렐라 애니메이션 영상을 실험 영상으로 이용하였으며, 이 애니메이션 캐릭터 중에서 남자 주인공의 얼굴을 사용자의 얼굴로 합성하였다.

그림 11은 본 논문에서 제안된 3차원의 얼굴 모델을 이용한 얼굴 합성 방법을 신데렐라 애니메이션 영상에 실제로 적용하여 산출한 결과를 보여준다. 그림 11에서 각 영상의 아래에 표기한 숫자는 전체 애니메이션 영상 중에서 몇 번째 프레임인지를 나타낸다. 그림 11에서 확인할 수 있듯이 애니메이션 캐릭터 얼굴의 축소 및 확대, 회전, 이동, 그리고 캐릭터 얼굴의 표정이나 얼굴 구성요소의 움직임이 발생하여도 사용자의 실제 얼굴이 캐릭터의 얼굴에 자연스럽게 합성됨을 시각적으로 확인할 수 있다.







그림 11. 얼굴 합성 결과  
Fig. 11. Face Synthesis Results

본 논문에서는 제안된 애니메이션 캐릭터 얼굴 합성 방법의 성능을 평가하기 위해서 75분짜리 신데렐라 애니메이션 영상 중에서 남자 주인공의 얼굴이 다양하게 변화하는 영상들을 선택하고 이를 5분짜리 애니메이션 영상으로 재구성하여 실험을 하였다. 제안된 얼굴 합성 방법의 성능을 평가하기 위해서 8명의 판정인들에게 얼굴 합성 결과를 보여주고, 표 1과 같은 5가지 기준(매우 우수, 우수, 보통, 미흡, 매우 미흡)에 의거하여 정확도를 평가하도록 했다.

표 1. 성능 평가의 기준  
Table 1. Criteria of Performance Evaluation

평가기준	설명
매우 우수	캐릭터의 얼굴 합성이 매우 정확함
우수	미세한 오차는 있으나 얼굴 및 얼굴 구성요소의 움직임과 모양을 정확히 반영하면서 캐릭터의 얼굴을 합성함
보통	얼굴 및 얼굴 구성요소의 움직임과 모양을 모두 반영하면서 얼굴을 합성하나 오차가 일정부분 발생함
미흡	얼굴 및 얼굴 구성요소의 움직임과 모양 중에서 한 가지만 반영하면서 캐릭터의 얼굴을 합성함
매우 미흡	얼굴 및 얼굴 구성요소의 움직임과 모양을 전혀 반영하지 못하면서 합성

그런 다음, 식 (10)과 같이 판정인들이 각각의 평가기준별로 추출한 합성 애니메이션 프레임의 개수를 전체 애니메이션 프레임의 개수로 나누어 백분율로 환산하였으며, 8명의 판정인들의 정확도 결과에 대한 평균을 계산하였다. 식 (10)에서  $No(E(i))$ 는  $i$ 번째 판정인이 평가한 애니메이션 프레임의 개수를 나타내고,  $No(Tot)$ 는 애니메이션의 전체 프레임의 개수를 나타낸다.

$$accuracy = \frac{1}{5} \times \sum_{i=1}^5 \left\{ \frac{No(E(i))}{No(Tot)} \times 100 \right\} \quad (10)$$

그림 12는 식 (10)을 활용한 성능평가의 결과를 그래프로 보여준다. 그림 12에서 확인할 수 있듯이 판정인들 대부분이 본 논문에서 제안된 방법이 사용자의 얼굴을 애니메이션 캐릭터 얼굴의 움직임과 모양을 비교적 정확성 있게 반영하면서 합성한다고 판단하였다. 그림 12에서 '미흡'과 '매우 미흡'으로 판정된 프레임들에서는 제안된 방법이 캐릭터의 얼굴을 부정확하게 추출한 것에서 기인한다. 다시 말해, 애니메이션 캐릭터의 얼굴 검출이 제대로 수행되지 않아서 얼굴의 위치, 그리



고 x축, y축, z축으로의 회전 정보가 부정확하게 측정하였고, 결과적으로 사용자 얼굴의 합성 또한 만족스럽지 못하게 수행되었다.

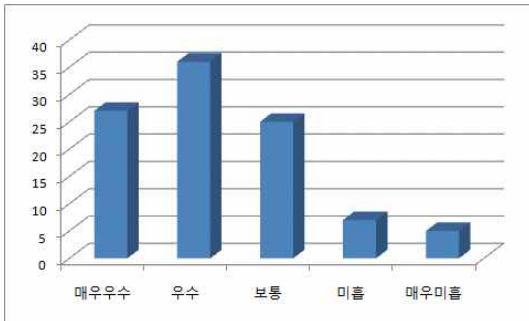


그림 12 성능 평가  
Fig. 12. Performance Evaluation

## VI. 결론 및 향후연구

본 논문에서는 3차원의 개인화된 얼굴모델을 생성하여 사용자의 얼굴을 애니메이션 캐릭터의 얼굴에 자연스럽게 합성하는 사용자 맞춤형의 애니메이션 캐릭터 생성 방법을 제안하였다. 제안된 방법에서는 먼저 정면과 측면의 직교하는 2장의 2차원 사용자 얼굴영상을 입력 받아 얼굴의 주요 특징을 템플릿 스테이크를 이용하여 추출하고, 추출된 특징점에 맞게 일반적인 3차원의 얼굴 모델을 변형시킴으로써 사용자의 얼굴에 최적화된 형태의 얼굴 모델을 생성한다. 그리고 2장의 얼굴 영상으로부터 얻어지는 텍스처 맵을 3차원의 얼굴 모델에 매핑하여 현실감 있는 개인화된 얼굴 모델을 생성한다. 그런 다음, 개인화된 3차원의 얼굴모델을 애니메이션 캐릭터 얼굴의 위치, 크기, 표정, 회전 정보를 반영하여 자연스럽게 합성함으로써 현실감 있는 사용자 맞춤형 애니메이션을 제작하였다.

향후 연구로는 보다 다양한 종류의 애니메이션 캐릭터에 제안된 사용자 얼굴 합성 알고리즘을 적용해 볼 계획이며, 알고리즘의 내부에서 사용하는 여러 파라미터들의 조율에 초점을 맞출 예정이다. 또한, 본 논문에서 제안된 시스템에서 사용된 개별 모듈의 안정화 및 성능평가를 보다 구체화할 수 있는 방안을 모색할 계획이다.

## 참고문헌

- [1] Y. C. Choi and S. B. Lim, "Multimedia Baerunter 2.0 - in the Era of IT Convergence," Life & Power Press, January 2010.
- [2] S.-W. Jang, S.-H. Weon, and H.-I. Choi, "Estimation of 3D Rotation Information of Animation Character Face," Journal of The Korea Society of Computer and Information, Vol. 16, No. 8, August 2011.
- [3] S.-W. Jang, H.-M. Shin, and G.-Y. Kim, "Face and Its Components Extraction of Animation Characters based on Dominant Colors," Journal of Korea Society of Computer and Information, Vol. 16, No. 10, pp. 93-100, November 2011.
- [4] S. Hong, "A Study on Pattern of Facial Expression Presentation in Character Animation," Journal of the Korea Contents Association, Vol. 6, No. 8, pp. 165-174, August 2006.
- [5] J.-H. Kim, "A Study on the Key Elements for Facial Expression in Animation Character," In Proceedings of the Spring Conference of the Korea Institute of Design, pp. 268-269, 2006.
- [6] S.-H. Weon, G.-Y. Kim, "Texture Mapping and 3D Face Modeling Using Two Views of 2D Face Images," Journal of Korean Institute of Information Scientists and Engineers, Vol. 15, No. 9, pp. 705-709, 2009.
- [7] F. Ofli, E. Erzin, Y. Yemez, and A. M. Tekalp, "Estimation and Analysis of Facial Animation Parameter Patterns," In Proceedings of the IEEE International Conference on Image Processing, pp. 293-296, September 2007.
- [8] O. Kwon, K. Min, J. Chun, and P. Park, "Expression Control of Facial Model Using Radial Basis Function in Real Time," In Proceedings of the Spring Conference of the Korean Society for Internet Information, Vol. 7, No. 1, pp. 363-368, April 2006.
- [9] Z. Fu, F. Ni, Q. Cao, and Y. Zhao, "The Facial Texture Analysis for the Automatic Portrait Drawing," Pattern Recognition, Vol. 43, No. 3, pp. 962-971,

March 2010.

- [10] S. Ferradal and J. C. Gomez, "TwinFaces: Seamless Textures for Rendering Head Models," In Proceedings of the IEEE International Workshop on Multimedia Signal Processing, pp. 176-181, 2006.
- [11] S.-W. Jang, G.-Y. Kim, and H.-S. Na, "Recognition of Facial Expressions of Animation Characters Using Dominant Colors and Feature Points," Journal of Korea Information Processing Society, Vol. 18-B, No. 6, pp. 375-384, December 2011.

### 저 자 소 개



장 석 우

2000년 8월:

송실대학교대학원 컴퓨터학과 (공학박사)

2009년 3월 - 현재: 안양대학교 디지털  
미디어학과 교수

관심분야 : 로봇비전, 증강현실, HCI,  
게임, 비디오 색인 및 검색,  
이러닝 등

E-mail : swjang@anyang.ac.kr



김 계 영

1996년 2월 :

송실대학교대학원 컴퓨터학과 (공학박사)

2001년 3월 - 현재 : 송실대학교 컴  
퓨터학과 교수

관심분야 : 컴퓨터 비전, 형태인식, 생체  
인식, 증강현실, 신호처리 등

E-mail : gykim11@ssu.ac.kr