

## 2차원 영상 기반 3차원 개인 얼굴 모델 생성 및 애니메이션

### 2D Image-Based Individual 3D Face Model Generation and Animation

김진우<sup>\*,\*\*</sup>, 고한석<sup>\*</sup>, 김형곤<sup>\*\*</sup>, 안상철<sup>\*\*</sup>

<sup>\*</sup>고려대학교 전자공학과, <sup>\*\*</sup>한국과학기술연구원 영상미디어연구센터

#### 요 약

본 논문에서는 사람의 정면 얼굴을 찍은 컬러 동영상에서 얼굴의 각 구성 요소에 대한 특징점들을 추출하여 3차원 개인 얼굴 모델을 생성하고 이를 얼굴의 표정 움직임에 따라 애니메이션하는 방법을 제시한다. 제안된 방법은 얼굴의 정면만을 촬영하도록 고안된 헬멧형 카메라(Head-mounted camera)를 사용하여 얻은 2차원 동영상의 첫 프레임(frame)으로부터 얼굴의 특징점들을 추출하고 이들과 3차원 일반 얼굴 모델을 바탕으로 3차원 얼굴 특징점들의 좌표를 산출한다. 표정의 변화는 초기 영상의 특징점 위치와 이후 영상들에서의 특징점 위치의 차이를 기반으로 알아낼 수 있다. 추출된 특징점 및 얼굴 움직임은 보다 다양한 응용이 가능하도록 최근 1단계 표준이 마무리된 MPEG-4 SNHC의 FDP(Facial Definition Parameters)와 FAP(Facial Animation Parameters)의 형식으로 표현되며 이를 이용하여 개인 얼굴 모델 및 애니메이션을 수행하였다. 제안된 방법은 단일 카메라로부터 촬영되는 영상을 기반으로 이루어지는 MPEG-4 기반 화상 통신이나 화상 회의 시스템 등에 유용하게 사용될 수 있다.

#### 1. 서론

최근 3차원 얼굴 모델을 통신상에서 보다 적은 데이터를 기반으로 빠르게 생성하고 애니메이션하는 방법이 많은 이들에 의해 연구되고 있다. 이와 같은 추세에 따라 얼마 전 1단계 표준화가 마무리된 MPEG-4에는 인간의 얼굴의 형태와 표정을 나타내는 FDP(Facial Definition Parameters)와 FAP(Facial Animation Parameters)라는 파라미터에 대한 규정이 포함되어 있으며 이는 화상 회의, 영상 기반 통신 등에 유용하게 사용될 전망이다 [1][2].

FDP와 FAP를 추출하기 위해 몇몇 연구단체에서 제안된 기존의 방법들은 주로 2장 이상의 영상을 이용하거나 얼굴에 마커(marker)와 같은 부착물을 사용한다는 단점이

있다[3][4]. 본 논문은 1개의 헬멧형 카메라(head-mounted camera)를 사용하여 얼굴의 정면 모습만을 촬영한 2차원 컬러 동영상으로부터 각 프레임(frame)에서 FDP와 FAP를 빠르고 정확하게 추출하여 3차원 개인 얼굴 모델을 생성하고 대상자의 표정을 따라 애니메이션하는 방법을 제안하고 있다. 제안된 방법은 얼굴에 마커 없이 정면에서 촬영된 영상만으로 개인 얼굴 모델을 생성하고 애니메이션할 수 있다. 또한 MPEG-4에서 제한하는 FDP와 FAP라는 표준에 따르므로 타 분야에 매우 유용하게 사용될 수 있다.

제안된 방법은 크게 두 부분으로 나뉘어진다. 첫 번째 단계는 얼굴의 정면 모습이 포함된 한 장의 프레임에서 얼굴의 형태를 나타내는 특징점과 FDP를 추출한 후 개인 얼굴 모델을 생성한다. 이후 연속되는 프레임들에

대해 수행되는 두 번째 단계에서는 눈, 입과 같은 부위 주변에서 표정을 대변하는 특징점 들만을 골라 그 위치를 template matching으로 구하고 이를 사용해 FAP를 계산하고 이를 기반으로 모델 애니메이션을 수행한다. 제안된 방법의 구조가 그림 1에 나타나 있다.

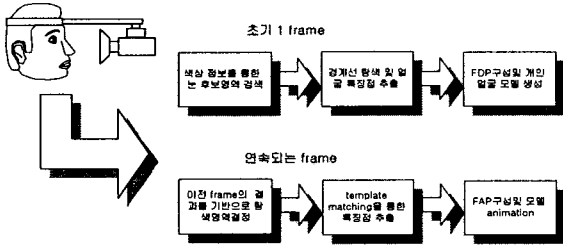


그림 1 제안하는 방법의 구성도

## 2. FDP 추출 및 개인 얼굴 모델 생성

### 2.1 색상과 형태를 이용한 눈 위치 확인

입력 영상은 먼저 휘도 성분에 의한 영향을 배제하기 위해 식 (1)과 같이 색상 공간을 휘도 성분으로 정규화한다[5].

$$r = \frac{255R}{R+G+B}, g = \frac{255G}{R+G+B} \quad (1)$$

정규화를 거친 영상 내에서 눈의 색상인 검은색과 흰색 영역을 강조하기 위해 본 논문에서는  $(m_r, m_g) = (85, 85)$ ,  $\sigma_r = 6$ ,  $\sigma_g = 6$ 의 값을 가지는 2차원 가우시안 함수를 사용하며 이를 통해 우리가 원하는 색상의 영역만을 강조하는 변환을 색상 변환(Color Transform)이라고 한다. 이 과정을 수식으로 표현한 것이 식(2)이며 해당 모델과 입력 영상에 이를 적용하는 과정을 그림 2에 나타내었다[5].

$$Z(x, y) = G(r(x, y), g(x, y)) \\ = \frac{1}{2\pi\sigma_r\sigma_g} e^{-\frac{1}{2} \left[ \left( \frac{r(x, y) - m_r}{\sigma_r} \right)^2 + \left( \frac{g(x, y) - m_g}{\sigma_g} \right)^2 \right]} \quad (2)$$

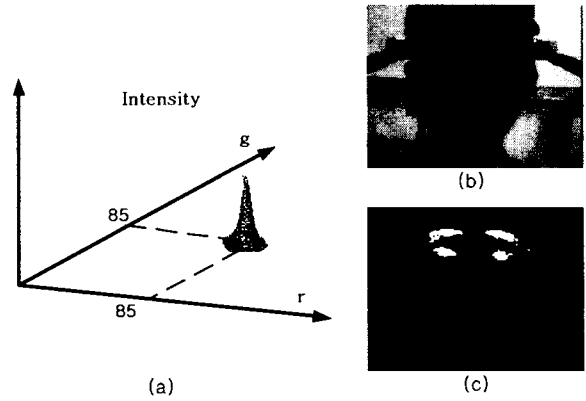


그림 2 (a) 검은색을 강조하기 위한 가우시안 함수, (b) 입력 영상, (c) 선택된 눈 후보 영역

형태 정보 또한 눈 위치 확인 과정에서 중요한 정보가 될 수 있다. 이러한 형태 정보로서 실험 전 다양한 사람들의 정렬된 얼굴 영상에 대하여 평균 영상과 Principal component를 추출하여 사용한다[6].

먼저 그림 2(c)의 후보 영역들을 대상으로 2개씩 group을 지어 이를 눈으로 가정한다. 각 경우에 대해 입력된 얼굴 영상은 Principal component를 추출하기 위해 영상을 정규화했던 방식과 동일하게 정렬된다. 그 후 정렬된 영상을 Principal component로써 재구성하여 나타낸다. 영상을 Principal component를 사용하여 재구성하는 과정이 식 (3)부터 식 (5)에 설명되어 있다.

$$\Gamma = [Y_1 Y_2 \dots Y_N]^T \quad (3)$$

$$\omega_i = u_i (\Gamma - \Psi) \quad (i = 1, 2, \dots, L) \quad (4)$$

$$\Phi_f = \sum_{i=1}^M \omega_i u_i + \Psi \quad (5)$$

$\Gamma$  : 입력영상     $\Psi$  : 평균영상

$u_i$  : i번째 principal component

$\Phi_f$  : 재구성된영상

원래 영상  $\Gamma$  과 재구성된 영상  $\Phi_f$  의 차이를 관찰하면 실제 눈을 기준으로 정렬한 영상의 경우 그 차이값이 거의 없으나 눈이 아닌

다른 쌍을 기준으로 정렬한 영상은 그 차이가 매우 크다. 그림 3의 상단의 입력 영상들 중 눈이 정확하게 검색된 얼굴인 경우는 거의 정상적으로 재구성된 영상이 되었으나 그렇지 않은 영상은 재구성된 영상이 원래의 입력 영상과 큰 차이를 가지는 것을 볼 수 있다. 이와 같은 근거로 원래 영상과 재구성된 영상의 차이가 가장 적은 쌍을 눈 영역으로 선택한다.



그림 3 검색 영상과 이를 재구성한 결과

## 2.2 얼굴 특징점 추출

본 논문에서 탐색하는 각 특징점들의 위치는 FDP 표준의 얼굴 특징점을 최대한 반영한 것이다. 입력된 얼굴 영상에 대하여 시스템은 먼저 각 구성요소들의 영역을 분리한 후, 각 영역의 경계선을 추출하여 각 경계선 상에 위치한 얼굴 특징점들의 2차원 좌표를 얻는다. 그림 4의 a)와 b)는 FDP에서 규정하고 있는 특징점과 본 논문에서 추출하는 40개의 특징점들을 표시한 것이다.

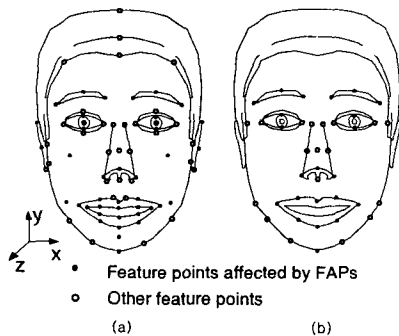


그림 4 (a) FDP 얼굴 특징점, (b) 추출하는 특징점

앞장에서 확인된 눈 영역의 위치와 얼굴의 지정학적 정보를 이용하여 시스템은 먼저 각

구성 요소들의 영역을 분리한다. 각 구성 요소 영역의 깨끗한 경계선을 얻기 위한 전처리 과정으로 각 영역에는 Histogram Stretching 기법이 적용되며 경계선 탐색을 위해 Sobel operation이나 Prewitt operation 등의 일반적인 경계선 추출 방법을 조합하는 Logistic function을 사용한다[7]. 그림 5에 나타낸 결과를 비교하여 볼 때 이 방법은 한 가지 방법만을 통해 검출하는 경우보다 정확한 경계선이 검출할 수 있음을 알 수 있다.

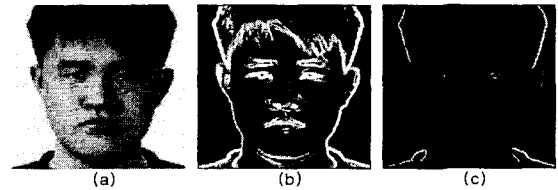


그림 5 (a) 입력 영상, (b) Logistic function으로 생성된 경계선 영상, (c) Sobel operation으로 생성된 경계선 영상

얼굴의 각 특징점들은 대개 각 구성 요소의 위나 아래쪽 혹은 옆쪽의 끝에 위치하고 있으므로 우리는 각 구성 요소의 Edge 영상을 가로와 세로 축 상으로 누적시킴으로써 특징점들을 구할 수 있다[5]. 그림 6에 입 영역의 특징점을 구하는 과정을 예를 들어 설명하고 있다.

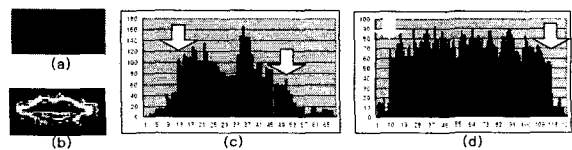


그림 6 (a) 입 영역, (b) 입 영역의 경계선 영상, (c) 수평 누적 histogram, (d) 수직 누적 histogram

## 2.3 FDP 구성 및 개인 얼굴 모델 생성

영상에서 추출된 각 특징점들은 모두 2차원 좌표계로써 표현되어 있다. 단일 영상만으로는 3차원 정보를 정확하게 추출하는 것이 불가능하므로 본 논문에서는 영상으로부터 추

출한 2차원 특징점들에 해당하는 위치를 일반 얼굴 모델에서 검색하여 그 깊이 정보를 가지고 z좌표를 추가하는 방법으로 3차원 얼굴 특징점들을 구해낸다. 이는 대부분의 사람들의 각 특징점 깊이 정보가 일반적으로 유사할 것이라는 가정 하에 이루어진다. FDP에서 추가적으로 규정되어 있는 특징점들의 위치는 검색된 각 점들의 위치 관계를 통해 추정될 수 있다. 본 논문에서 사용한 일반 모델인 IST 모델과 입력 영상에서 추출된 특징점 및 이를 이용하여 생성된 개인 얼굴 모델의 모습을 그림 7에 나타내었다.

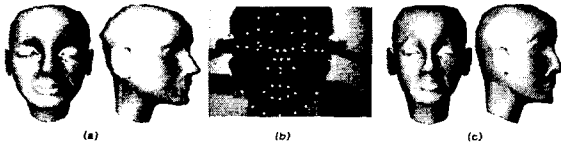


그림 7 (a) IST 일반 얼굴 모델, (b) 추출된 얼굴 특징점, (c) 생성된 개인 얼굴 모델

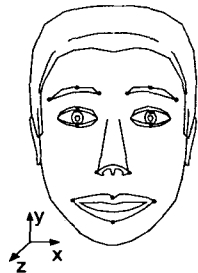


그림 8 표정 검색에 사용되는 얼굴 특징점

### 3. FAP 추출 및 모델 애니메이션

주요 얼굴 구성 요소들의 움직임을 나타내는 FAP는 각 frame에서 촬영된 얼굴의 각 특징점들을 검색하고 그 위치를 초기 frame의 무표정한 얼굴에서 추출한 특징점 위치와 비교함으로써 구할 수 있다. 이 때 FDP를 구성하는데 사용되었던 특징점들을 모두 추출하지 않고 눈, 입 등의 얼굴 구성 요소 주변에 위치하는 특징점만을 추출한다. 대부분의 표정은 얼굴 구성 요소의 움직임에 기인하므로 우리

는 적은 양의 계산으로도 효과적인 표정 검색을 수행할 수 있다. 본 논문에서 표정을 나타내기 위해 선택한 특징점은 모두 16개이며 각 위치가 그림 8에 표시되어 있다. 각 특징점들은 template matching 기법을 통해 추출된다.

#### 3.1 Template Matching을 통한 특징점 추출

검색할 각 특징점들의 template로써 첫 번째 frame에서 검색된 각 특징점들 주변의 일정 크기의 영역을 사용할 수 있다. 또한 시간이 지나면서 변화하는 표정에 대한 영향을 줄이기 위해 검색 과정을 하기 이전 다양한 사람으로부터 여러 가지 표정의 변화 시 생성되는 특징점 주변 template들을 추출하여 사용한다. 이러한 방법은 여러 사람의 얼굴 형태가 서로 달라도 비슷한 표정을 지을 때 특징점 주변의 좁은 영역에서는 그 형태가 유사하다는 가정 하에 성립 될 수 있다. 여러 가지 보조 template들에 관한 예시 영상을 그림 9에 나타내었다.

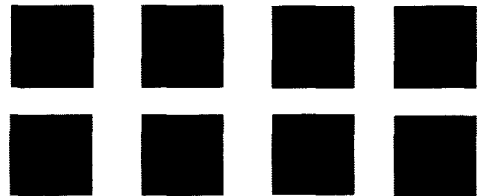


그림 9 다양한 표정 변화에 사용되는 보조 template

#### 3.2 FAP 생성 및 모델의 Animation

초기 프레임에 비해 각 특징점들이 움직인 정도는 개인 얼굴의 구성요소간 거리를 기초로 계산되는 FAPU라는 단위로 전환된다. 그 후 얼굴 구성 요소 각 부분의 움직임을 FAP에서 정의한 형식에 맞추어 표현한다. 얼굴 각 부분의 다양한 움직임을 표현하기 위해 FAP에서는 총 66개의 세부적인 얼굴 움직임 항목과 1개의 감정 항목, 1개의 viseme(visual

phoneme)이 규정되어 있다. 본 논문에서 제안하는 방법은 이들 중 35가지의 얼굴 움직임을 FAP로 생성하고 이를 이용하여 개인 얼굴 모델의 표정 애니메이션을 수행하였다.

#### 4. 실험 방법 및 결과

제안된 방법은 Windows NT를 기반으로 하는 Pentium 200 PC를 기반으로 구현되었다. 카메라로부터 프레임을 얻고 영상들을 대상으로 하는 연산을 고속으로 처리하기 위해 TI사에서 제작한 TMS320C80 DSP board를 사용하였으며 또한 영상 내 얼굴의 위치와 방향을 고정시키기 위해 헬멧형 카메라(head-mounted camera)를 자체 제작하였다.

다양한 입력 영상을 대상으로 수행된 실험에서 얼굴 특징점들은 2 pixel의 오차 이내에서 약 93 % 이상의 성공률로 추출되었다. 에러 발생의 가장 주요한 원인은 턱, 코 부분 등의 영역에서 경계선이 비교적 흐릿했기 때문이었다. 320x240 크기를 가진 컬러 동영상의 초기 프레임을 대상으로 FDP의 얼굴 특징점 구성 및 3차원 개인 얼굴 모델 생성 과정은 약 2초 이내에 수행될 수 있었다. 실험 대상에 대해 모델을 생성하고 texture를 입힌 결과를 그림 10 (a)와 (b)에 나타내었다. FAP를 추출하는 부분은 주어진 하드웨어에서 초당 12 frame의 속도로 동작하였다. 그림 11은 입력되는 다양한 표정에 대하여 검색되는 특징점과 이를 이용해 모델의 움직임을 구현한 결과이다.



그림 10 (a) 실험 대상, (b) 생성된 개인 모델

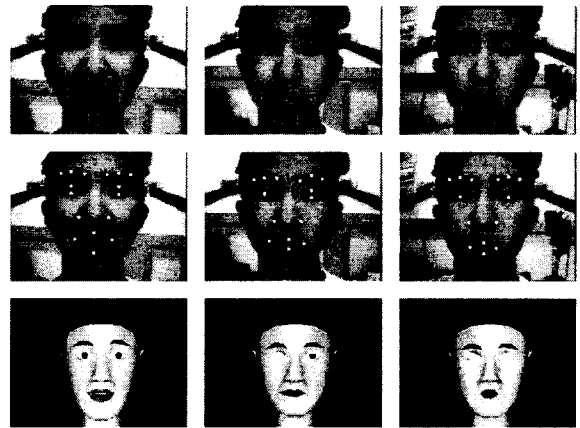


그림 11 다양한 표정의 얼굴 영상으로부터 추출된 특징점 및 모델 애니메이션

#### 5. 결론

지금까지 얼굴이 포함된 2차원 컬러 동영상으로부터 대상의 형태와 표정에 대한 정보를 얻어 3차원 개인 얼굴 모델을 구성하고 동작시키는 방법에 대해 다루었다. 본 논문이 제안하는 방법은 마커(marker)등의 부착물 없이 카메라와 일반 PC만으로 3차원 개인 얼굴 모델을 생성할 수 있다는 장점을 가지고 있으며 또한 MPEG-4에서 제한하는 FDP와 FAP라는 표준에 따르므로 이 후 화상 회의, 화상 통신 시스템 등에 매우 유용하게 사용될 수 있다.

본 시스템에 대한 개선점으로서 보다 사용자의 편의를 고려하기 위해 헬멧형 카메라 없이 자동적으로 영상 내 얼굴 위치 및 방향을 보정할 수 있는 방법이 고안되어야 할 것이다.

#### 참고 문헌

[1] MPEG-4 System Sub-Group, "Information Technology - Generic Coding of Audio - Visual Objects Part1 : Systems," ISO/IEC JTC1/SC29/WG11 N2501, 1999.  
 [2] MPEG-4 System Sub-Group, "Information Technology - Generic Coding of Audio - Visual Objects Part2: Visual," ISO/IEC

JTC1/SC29/WG11 N2502, 1999.

- [3] Won-Sook Lee, Marc Escher, Gael Sannier, Nadia. M. Thalmann, "MPEG-4 Compatible Faces from Orthogonal Photos," Proc. International Conference on Computer Animation(CA99), Geneva, pp.186-194, May 1999.
- [4] Adam Eliassen, Søren Forchhammer, Niels-Jørn Dyhr, Finn Nielsen, " Danish visual speech synthesis & MPEG-4 Facial animation plug-in for Realplayer G2," International Workshop on Synthetic - Natural Hybrid Coding and Three Dimensional Imaging (IWSNHC3DI'99), pp. 37-40, Sept. 1999.
- [5] 함 상진, 김 형곤, "3차원 개인 얼굴 모델 자동 생성," 전자공학회 논문지, 제 36권 S편 제1호, pp. 104-114, Jan. 1999.
- [6] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, Vol. 12, No. 1, pp. 71-86, Jan. 1991.
- [7] James C. Bezdek, Ramachandran C. and Yianni A., "A Geometric Approach to Edge Detection," *IEEE Trans. Fuzzy Systems*, Vol 6, no.1, pp. 52-77, Feb. 1998.