2장의 2차원 얼굴영상을 이용한 텍스쳐 생성과 자동적인 3차원 얼굴모델링

(Texture Mapping and 3D Face Modeling using Two Views of 2D Face Images)

원 선 희 † 김 계 영 ††
(SunHee Weon) (GyeYoung Kim)

요 약 본 논문에서는 직교하는 2장의 얼굴영상과 얼굴 특징 자동추출을 통하여 3차원 얼굴모델을 생성하는 기술을 제안한다. 제안하는 기술은 3차원 얼굴모델을 개인화하는 부분과 2장의 얼굴영상으로부터 얻은 텍스쳐 맵을 3차원 얼굴모델에 사상하는 부분으로 구성된다.

키워드 : 템플릿 스네이크, 3차원 얼굴 모델링, 텍스쳐 맵핑

Abstract In this paper, we propose 3d face modeling using two orthogonal views of 2D face images and automatically facial feature extraction. Th proposed technique consists of 2 parts, personalization of 3d face model and texture mapping.

Key words : Template Snake, 3D Face Modeling, Texture Mapping

1. 서 론

본 논문에서는 실감 있는 3차원 얼굴모델을 생성하는 방법에 관하여 기술한다. 얼굴 모델링과 관련한 기존 연구는 주로 실감나는 얼굴모델의 생성이나 실시간 얼굴 모델 생성, 자동화 시스템 등을 개발하는데 초점을 맞추고 있다. 그러나 이러한 얼굴 모델링에 관한 연구는 가상현실, 화상회의, 게임, 광고, 3차원 얼굴 인식과 같은 분야에서 많이 활용되고 있다. 하지만 아직까지 현실감 높은 얼굴모델을 생성하지 못하고 있으며, 주로 애니메이터에 의해 모델링 및 렌더링 도구를 이용하여 작업이 진행되므로 많은 노력과 시간이 소요된다.

기존의 3차원 얼굴 모델링에 관한 연구는 주로 레인지 스캐너 또는 digitizer probe, stereo disparity 같은 고가의 장비를 이용하여 3차원 좌표를 측정하여 생성하였다[1]. 최근에는 정, 측, 후면의 2차원 얼굴영상을 2장 이상 이용하여 얼굴 윤곽선과 특징들의 정보를 획득함으로써 3차원의 얼굴모델을 생성하는 기법[2,3]도 연구되고 있다. 3차원의 좌표 정보를 쉽게 획득할 수 있는 레인지 스캐너 같은 장비를 이용하면 상세한 기하 형태 및 좋은 텍스쳐 정보를 얻을 수 있다는 장점이 있다. 반면, 데이터가 너무 많아 취급이 어려운 문제점, 경계선을 수작업으로 수정하여야하며, 정합되는 지점으로 인한 영역의 손실되는 등의 단점이 있다. 2차원의 얼굴영상을 이용하는 경우에는 정확한 얼굴특징정보의 추출이 어려워 부자연스러운 3차원 얼굴모델을 생성하는 단점이 있다. 본 논문에서는 2장의 직교하는 2차원 얼굴영상을 이용하고 안정적인 얼굴특징정보의 추출을 통해 보다 쉽고 빠르게 3차원 얼굴모델을 생성하는 방법에 대해 제안한다.

제안하는 방법은 먼저, 정면과 측면 2장의 2차원 얼굴영상을 입력 받아 얼굴 특징점에 맞게 3차원 표준모델(generic model)을 변형시킴으로써 개인의 얼굴과 유사한 형태의 모델을 형성하는데, 이 과정에서 템플릿 스네이크 알고리즘[4]을 사용하여 개인화된 얼굴모델을 생성하기 위한 얼굴특징을 추출한다. 그 다음 얼굴의 대칭적인 특징을 이용하여 2장의 얼굴영상으로부터 얻어지는 텍스쳐 맵을 사용하여 현실감 있는 개인화된 3차원 얼굴모델을 생성한다.

본 논문의 구성은 5절로 구성되어 있다. 2절에서는 2차원 얼굴영상에서 얼굴특징요소들의 영역과 위치 및 윤곽선을 검출하는 방법을 설명한다. 3절에서는 3차원 표준모델을 변형하여 개인모델을 생성하는 방법과 2장의 얼굴영상을 이용한 텍스쳐 맵을 생성하는 방법에 대해 기술한다. 4절에서는 제안된 방법을 사용하여 실험한 결과와 연산시간을 분석하며, 5절에서 결론을 맺고 향후 연구에 대해 논의한다.

· 본 연구는 서울시 산학연 협력산업(10581CO93113)의 연구비 지원으로 수행하였습니다.
· 이 논문은 제34회 추계학술대회에서 '2장의 2차원 얼굴영상을 이용한 텍스쳐 생성과 자동적인 3차원 얼굴모델링'의 제목으로 발표된 논문을 확장한 것임

† 학생회원 : 숭실대학교 미디어학과
nifty12@ssu.ac.kr
†† 종신회원 : 숭실대학교 컴퓨터학과 교수
gykim11@ssu.ac.kr

논문접수 : 2009년 1월 15일
심사완료 : 2009년 7월 24일

Copyright©2009 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지: 컴퓨팅의 실제 및 레터 제15권 제9호(2009.9)

2. 2차원 얼굴영상에서의 얼굴특징 추출

얼굴영역 검출 기법 중 가장 널리 사용되고 있는 방법이 피부색상 정보를 이용한 검출 방법이다. 색상 정보를 이용한 방법은 빠르게 처리 할 수 있고, 얼굴의 기하학적인 변화에 비교적 많은 영향을 받지 않는 가장 효율적인 방법이다. 본 논문에서는 여러 가지 색상 모델 중 실험을 통하여 가장 적합한 YCbCr 색상 모델을 선택하였으며 얼굴영역 추출 시 피부색상과 직접적인 관련이 있는 색도 성분인 Cb, Cr 값만으로 피부색상을 모델링하여 얼굴영역을 추출한다.

이 같은 YCbCr 색상 모델을 사용하여 표본 데이터를 추출한 결과, 식 (1)과 같이 피부색의 범위를 설정하였다. 이 피부색상 참조맵을 가지고 피부색 영역을 분류를 하기에 앞서, 입력 받은 얼굴영상에 동적 윤곽선 추출 알고리즘(Snake)을 사용하여 얼굴 윤곽선을 추출한다. 추출된 얼굴영역 안에서 피부색상을 이용하여 얼굴영역과 얼굴특징요소영역을 분류할 수 있다. 픽셀 (x, y)의 Cb, Cr 값이 식 (1)에서 제안한 범위 안에 만족하면 얼굴영역이고 그렇지 않으면 얼굴특징요소의 영역이므로 얼굴특징요소영역은 0(검정색)으로 설정하여 영역을 분할(segmentation) 할 수 있다.

$$skin\,region\begin{cases}77 < Cb < 117\\ 143 < Cr < 173\end{cases} \tag{1}$$

그림 1은 식 (1)의 범위로 만든 피부참조 맵을 이용하여 얼굴영역과 얼굴특징요소의 영역을 구분하여 검출한 것으로 (a)는 피부색상에 해당하는 얼굴후보영역을 검출한 것이고, (b)와 (c)는 얼굴후보영역 안에서 식 (2)를 사용하여 수평 방향과 수직 방향으로 투영하여 얼굴특징요소의 후보영역을 최소포함사각형(MER) 형태로 추출한 것이다.

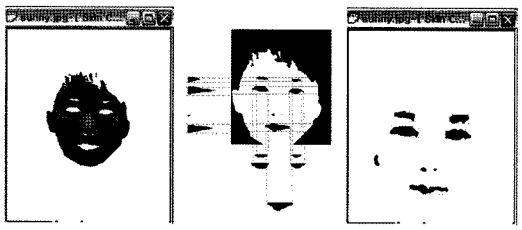

그림 1 얼굴 및 얼굴특징요소 후보영역 검출 결과. (a) 피부영역과 특징요소영역 분류 (b) 수직 및 수평 프로젝션 결과 (c) 얼굴특징요소 후보영역 검출결과

$$H_i = \sum_{j=0i=0}^{Y}\sum^{X}\frac{x_{(j,i)}}{255}, \quad H_j = \sum_{i=0j=0}^{X}\sum^{Y}\frac{y_{(i,j)}}{255} \tag{2}$$

앞서 추출한 얼굴특징요소의 후보영역 내에서 초기 템플릿 모델을 정의한 스네이크를 이용하여 각각의 얼굴특징요소의 윤곽선을 검출하는데 얼굴특징요소들의 각각의 위치적인 특성상 에지의 미흡성을 고려하여야 한다. 이를 위하여 본 논문에서는 기존 스네이크 알고리즘의 에너지 항에 변화를 주는 방법을 이용하여 이를 해결하였는데, 마지막 에너지 항인 Eimage 를 얼굴특징요소별로 다르게 정의하는 방법으로 식 (3), (4)와 같이 눈과 눈썹의 윤곽선 추출을 위한 Eeye_snake와 입의 윤곽선 추출을 위한 Emouth_snake를 제안하였다. 특히 눈과 눈썹의 경우는 눈가의 주름이나 머리에 의하여 눈썹이 가려지는 경우 또는 눈썹이 흐린 사람의 경우 정확한 에지를 추출해내기 어렵기 때문에 1차와 2차 에지 연산자를 병합하여 윤곽선을 추출하는 형태로 이미지 항을 정의하였다. 이때 사용되는 $\alpha,\beta,\gamma_1,\gamma_2$ 상수 값은 실험에 의하여 각각 1.0, 1.7, 0.5, 0.5로 설정하였다. 또한 입의 윤곽선은 눈이나 눈썹에 비해 비교적 뚜렷하기 때문에 1차 에지 연산자만을 이용하여 이미지 항을 정의한다. 식 (4)에 사용된 α,β,γ_1상수 값은 실험을 통하여 각각 1.0, 1.7, 1.0으로 정의하였다.

$$E_{eye_snake} = \int_0^1 (\alpha(s)\left|\frac{dv}{ds}\right|^2 + \beta(s)\left|\frac{d^2v}{ds^2}\right|^2 - (\gamma_1|\nabla I(v)| + \gamma_2|\nabla^2 I(v)|))ds \tag{3}$$

$$E_{mouth_snake} = \int_0^1 (\alpha(s)\left|\frac{dv}{ds}\right|^2 + \beta(s)\left|\frac{d^2v}{ds^2}\right|^2 - \gamma_1|\nabla I(v)|)\,ds \tag{4}$$

3. 3차원 개인 얼굴모델의 생성

본 연구에서는 2장의 2차원 얼굴영상을 이용하여 크기와 형태에 관계없이 얼굴의 눈, 눈썹, 코, 입 등 각 부위별로 와이어 메쉬 상의 정점을 조정함으로써 빠르고 정확하게 얼굴에 맞는 모델을 생성할 수 있는 방법을 제안하며 시스템은 그림 2와 같이 구성되어 있다. 정면과 측면에서 촬영된 2방향의 얼굴영상을 입력받아 각각의 영상에서 얼굴 윤곽선, 눈썹, 눈 등의 특징요소들의 특징점을 추출한다. 추출된 특징점의 위치정보를 바탕으로 3차원 표준모델의 크기를 변형시키고 표준모델상의 제어점을 이동시켜 입력영상과 같은 형상의 개인 얼굴모델을 생성한다. 생성된 개인 얼굴모델에 입력영상으로부터 얻은 텍스쳐 맵을 맵핑함으로서 실감나는 3차원 개인 얼굴모델이 생성된다.

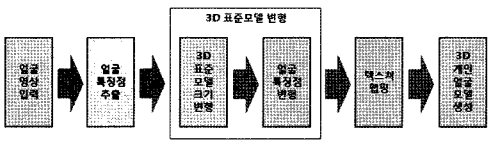

그림 2 제안하는 방법의 절차도

3차원 개인 얼굴모델을 생성하기 위해 필요한 표준모델은 총 926개의 vertex와 1,797개의 polygon으로 구성되었다. 이 얼굴모델을 3Ds Max의 export 포맷 중 하나인 ASE 파일로 export하여 이 파일을 분석함으로써 polygon mesh의 각 정점(vertex)의 좌표를 구할 수 있고, 이를 바탕으로 일반모델을 생성하였다.

3.1 3차원 모델의 변형

3차원 얼굴 모델링을 위해서는 표준 얼굴모델을 특정 개인의 얼굴에 맞게 정합하는 과정이 필요하다. 본 논문에서는 얼굴영상의 특정부분별로 나누어 조정하도록 와이어 메쉬 상의 점을 제어점(control point)라 정의한다. 머리, 얼굴의 형태는 매우 복잡하고 세밀하므로 제어점들은 얼굴의 형태와 크기, 특징 요소들 간의 관계를 고려하여 각 얼굴 부분별로 특징을 정확하게 정합할 수 있도록 하였으며 표 1과 같이 9개의 세분화된 구역으로 구분한다.

표 1 얼굴 부위별로 세분화된 얼굴특징요소의 제어점

영역 / 영상	윤곽선	눈썹	눈	코	귀	입	합계
정면	12	4	12	4	0	7	39
측면	6	2	3	3	4	4	22

추출된 2차원 얼굴특징정보를 바탕으로 3차원 표준 얼굴모델의 형태를 변형시키는 과정에서 모델의 부드러운(smoothing) 표면을 위하여 Radial Basis Function을 이용한다. 일반적인 변형과정 대응되는 두 모델의 포인트들의 개수가 같아야 하지만 이 방법에서는 같지 않더라도 서로 대응되는 포인트들의 위치만 정확하게 찾아준다면 보다 좋은 성능을 기대할 수 있는 기법이다[5,6]. 따라서 이미 앞 절에서 추출한 얼굴특징정보는 얼굴특징의 정확한 위치정보와 형태정보를 포함하고 있으므로 3차원 표준모델 메쉬의 정점들과 매칭을 통하여 모델의 형태를 변형시킬 수 있다. 표준 얼굴모델의 변형을 위해 RBF 함수를 사용한다. 얼굴모델에서 사용할 특징점은 위에서 정의한 것처럼 정면 39개, 측면 12개를 사용하고 각 특징점에 따라 각기 다른 영향을 받는 영역들로 클러스터링 한다. RBF 함수는 각각의 특징점들이 템플릿 스네이크로 이동된 특징점들과 대응되는 위치로 이동하고 각 특징점들의 영향을 받는 영역 내에 있는 정점들은 중간적인 위치로 사상시키는 보간법으로써, 변형하고자 하는 특징점을 $f(\vec{x})$라고 한다면, RBF 보간함수는 $F(\vec{x},\vec{c})$으로 정의된다. 여기서, $\vec{x}$는 특징점을 나타내는 벡터 $\vec{x}=x_1,x_2,\ldots,x_n$이고 $\vec{c}$는 각 특징점에 대해 영향을 받는 가중치 값($\vec{c}=c_1,c_2,\ldots,c_n$)이다. RBF 보간함수를 식 (5)와 같이 표현할 수 있는데 이때, N은 특징점의 개수를 나타내고, h(r)은 RBF의 기저함수를 나타내며 파라미터 r은 특징점과 정점들 간의 유클리디안 거리(Euclidean distance)를 의미한다.

$$F(\vec{x})=\sum_{i=1}^{N} c_i h(r) \tag{5}$$

본 논문에서는 RBF 함수의 기저함수로는 가우시안 함수를 사용하였으며, 수식으로 표현하면 식 (6)과 같다.

$$F(\vec{x})=\sum_{i=0}^{N} c_i e^{-\left\{\frac{r_i^2}{\sigma_i}\right\}} \tag{6}$$

식 (6)을 수행하기 위하여 특징점, 가중치, 너비매개변수를 필요로 하는데 특징점은 미리 설정해놓은 (x_i,y_i,z_i)이므로 가중치 c_i와 너비매개변수 σ_i를 구해야한다. 특징점들의 초기 위치를 $\vec{x}^{source}$이라 하고 이동된 후의 위치를 $\vec{x}^{target}$이라 할 때, 모든 특징점들은 변위를 알 수 있으므로 가중치 c_i를 식 (7)을 통해 구할 수 있다. 또한, 특징점이 이동함에 따라 특징점 클러스터에 속하는 정점들이 받는 영향의 범위와 관련된 너비매개변수 σ_i는 각 특징점들과 정점들 간의 유클리디안 거리 중 가장 먼 거리값을 이용하여 각 클러스터 마다 가중치를 이용하여 식 (8)과 같이 σ_i를 구한다. 따라서 특징점과 가까운 거리에 있는 정점들에는 큰 가중치를, 먼 거리에 있는 정점들에는 작은 가중치 값을 부여하여 특징점의 이동에 따라 정점들의 부드러운 변형이 일어나게 한다.

$$\vec{c}=h^{-1}\vec{x}^{target} \tag{7}$$

$$\sigma_i=\max\left\|\vec{x}_i^{source}-\vec{x}_j\right\|\times weight \tag{8}$$

3.2 텍스쳐 맵 생성

템플릿 스네이크를 적용한 특징점 제어를 통해 정합된 개인 얼굴모델을 이용하여 보여지는 평면에 대한 텍스쳐를 얻을 수 있다. 얼굴 모델링의 최종단계인 렌더링 단계에서는 쉐이딩 기법(shading)이나 텍스쳐 맵핑 기법(texture mapping)을 이용할 수 있는데 쉐이딩 기법은 특징이나 조명에 따라 픽셀 값을 계산하므로 얼굴같은 섬세한 부분을 표현하기 어려운 반면, 텍스쳐 맵핑은 3차원 좌표계의 모델 표면을 늘리거나 줄여서 붙여넣는 방식으로 맵핑시켜줌으로써 실제 모델과 차이 없이 현실감있게 표현할 수 있다. 따라서 본 논문에서는 다음과 같이 2차원 이미지들을 합성하여 뷰 독립적인 텍스쳐 맵(view-independent texture map)을 구성한다[7].

얼굴모델의 메쉬 상의 3차원 좌표정보와 2차원 얼굴영상의 텍스쳐 공간 사이의 사상은 원통 투영법(cylinder mapping)을 이용하여 정의한다. 얼굴모델의 메쉬 구조의 3차원 좌표를 원통에 투영시켜 얻은 값 (u,v)를 계산하고, 메쉬 상의 정점 p를 대응되는 2차원 영상 평면 위에 투영시킨 j 카메라의 위치에 대한 (x_j,y_j)를 계

산하여 p에 해당하는 색상값 $I_j(x_j, y_j)$를 구한다. 각 픽셀 (u,v)에 대해, 최종적으로 합성되는 텍스쳐 맵의 색상조합은 식 (9)와 같이 결정된다.

$$I(u,v) = \sum_j W_j(u,v) I_j(u,v) \quad (9)$$

여기에서, $W_j(u,v)$은 j 카메라의 입력 영상의 기여도에 따른 가중치로서 카메라 시선 단위 벡터와 3차원 얼굴모델 메쉬의 단위 법선 벡터의 내적에 기반하여 각 얼굴영상에 대한 각각의 메쉬의 합성 가중치를 정한다. 만약, 대응점 (u,v)가 영상에서 보이지 않을 때는 가중치는 0으로 설정하고 전체 가중치 합은 1로 정규화시킨다. 본 논문에서는 기존 연구에서 실험되어왔던 3장 이상의 영상을 사용하는 방법이 아닌 2장의 영상만으로 텍스쳐 맵을 생성하였다. 즉, 좌측면 얼굴영상의 텍스쳐 정보를 정면 영상의 좌표를 기준으로 대응되는 위치에 텍스쳐 정보를 투영시키고, 후면의 텍스쳐 정보는 머리 부분에 해당하는 특징점들의 텍스쳐 정보를 결합하여 빈 공간을 채워줌으로써 렌더링 과정에서 보다 빠른 처리가 가능하도록 하였다.

4. 실험 및 결과

본 논문에서 제안한 방법은 Pentium III 3.4GHz CPU에 OpenGL 가속보드가 설치된 시스템에서 Visiual C++ 6.0 기반으로 MFC 라이브러리와 영상처리 오픈 라이브러리인 OpenCV 1.0ver을 사용하여 구현 및 실험되었다.

그림 3은 템플릿 스네이크를 이용하여 전체 영상에서 얼굴 윤곽선 및 얼굴특징요소의 윤곽선을 찾아낸 결과로 실제로 이 윤곽선 및 위치를 정확하게 찾아내기 위한 반복 수행 횟수는 총 10번 이내임을 알 수 있다(그림 4). 또한 그림 3의 (c)에서는 추출된 특징점을 이용하여 표준 얼굴모델이 개인 얼굴모델 형태로 변형된 결과를 확인할 수 있다. 이 과정에서 특징점들이 윤곽선을 따라 이동하는 동시에 3차원 모델에서의 제어점도 이동되고 실시간으로 개인 얼굴모델의 형태가 생성된다. 따라서 특징점들의 위치 추출을 위한 반복 수행횟수와 제어점의 반복 수행횟수가 동일하다.

그림 5에서는 3방향의 얼굴영상을 이용한 방법과 2방향의 얼굴영상을 이용하여 얻은 결과물을 비교한 것으로 3방향 이상의 영상을 이용한 결과물과 비교하여 결과물의 자연스러움에 큰 차이가 없는 것을 알 수 있다. 표 2는 얼굴특징요소를 수동으로 추출하는 방법, AAM을 사용한 후 3차원 얼굴모델을 생성하는 방법[8,9]과 제안한 방법과의 수행속도를 비교한 것이다. 2차원 영상에서의 자동적인 얼굴특징추출이 이루어지고 우측 및 후면 영상의 텍스쳐 정합단계가 생략되어 훨씬 빠르게 결과물을 얻을 수 있는 것을 확인할 수 있다.

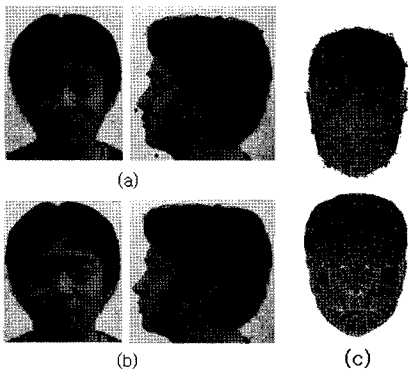

그림 3 얼굴특징요소 추출결과. (a) 특징점의 초기 위치 (b) 추출된 특징요소 (c) 개인 얼굴모델 생성

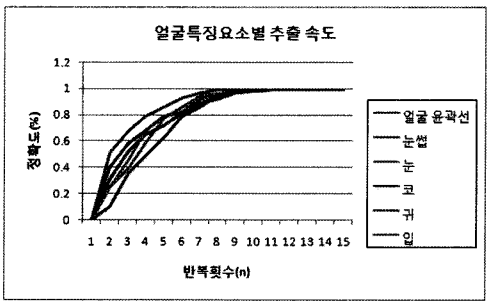

그림 4 얼굴특징요소의 위치 추출을 위한 수행속도

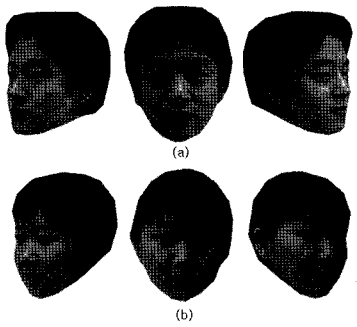

그림 5 다시점 영상을 사용한 시스템과의 결과 비교. (a) 2방향 사용 (b) 3방향 이상 사용

그림 6은 본 논문에서 제안된 알고리즘으로 얻은 결과물과 기존 논문[10]에서의 결과물을 비교한 것으로, (a)의 경우 정면과 측면에서의 형태를 미리 형성하고 텍스쳐 맵을 맵핑하여 실사영상과 유사한 얼굴모델을 생성할 수 있었다. 반면 (b)의 타 방법의 수행결과에서는 정면과 측면의 형태를 따로 설정하지 않고 측면의 형태와 텍스쳐를 그대로 대칭하여 얼굴모델을 생성하였으므로 실제 영상과는 확연히 다른 얼굴모델이 생성됨을 알 수 있다.

표 2 타 모델링 시스템과의 수행 속도 비교

항목	수동적 얼굴 모델링 방법	자동적 얼굴 모델링 방법	제안하는 방법
2차원 특징점 추출	측정 불가	15.3초	10.1초
얼굴모델 정합	89초	19초	10.1초 (자동 정합)
텍스쳐 맵 생성	모름	22초	12.3초
렌더링	모름	0.01초 미만	0.01초 미만
총 소요 시간	89+α 초	56.31초	31.21초

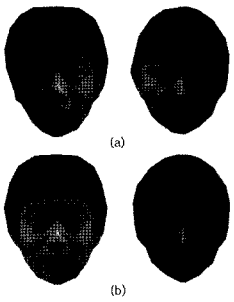

그림 6 타 알고리즘과의 수행결과 비교. (a) 제안된 알고리즘의 결과 (b) 타 알고리즘의 결과

5. 결론 및 향후 연구 계획

본 논문에서는 2장의 직교 얼굴영상을 이용하여 3차원 개인 얼굴모델을 자동적으로 생성하고 텍스쳐 맵을 2장의 실사 영상으로부터 획득하여 실감나는 개인 얼굴모델을 생성하는 방법을 제시하였다. 기존의 수작업에 의존하여 얼굴 특징을 추출하거나 학습과정이 필요했던 방법과 달리 다양한 얼굴 특징 요소의 형태에 동적으로 실시간 추출이 가능한 템플릿 스네이크 알고리즘을 사용하여 3차원 얼굴모델 형태의 편집 시간을 크게 줄일 수 있었고, 2장의 영상으로부터 획득한 텍스쳐 맵을 사용하여 사실감 있는 얼굴모델의 생성이 가능하였다. 또한 텍스쳐 맵을 생성할 때 기존의 2장의 영상을 이용한 방법의 경우 개인 얼굴 생김새와 관계없이 대칭하여 결과물을 획득한 것과 달리 얼굴모델의 형태를 먼저 정합한 후 텍스쳐 맵을 생성하였기에 훨씬 더 자연스럽고 실감나는 개인화된 얼굴모델을 얻을 수 있었다.

좀 더 텍스쳐의 실 얼굴영상의 사실감을 반영하기 위해 향후 연구 방향으로 두 인접 영상의 색 보정과정에서 인접하는 경계선 영역의 색 분포도를 고려하는 기법에 대한 연구가 필요할 것이다.

참 고 문 헌

[1] T. Russ, C. Boehnen, T. Peters, "3D Face Recognition Using 3D Alignment for PCA," *Proc. of the 2006 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR'06)*, vol.2, pp.1391-1398, 2006.

[2] A. Ansari, M. Abdel-Mottaleb, "Automatic facial feature extraction and 3D face modeling using two orthogonal view with application to 3D face recognition," *The Journal of the Pattern Recognition*, vol.38, pp.2549-2563, 2005.

[3] Z. Mandun, M. Linna, X. Y. Zeng, Y. S. Wang, "Image-Based 3D Face Modeling," *Proc. of the Int'l Conf. on Computer Graphics, Imaging and Visualization(CGIV'04)*, pp.165-168, 2004.

[4] S. H. Weon, K. S. Lee, G. Y. Kim, "Contour Extraction of Facial Feature Components Using Template Based Snake Algorithm," *ICCSA 2007, LNCS 4705, Part I*, pp.1034-1044, 2007.

[5] J. Y. Noh, D. Fidaleo, U. Neumann, "Animated Deformation with Radial Basis Functions," *Proc. of the ACM symposium on Virtual reality software and technology(VRST'00)*, USA : ACM Press, pp. 166-174, 2000.

[6] C. Chen, E. C. Prakash, "Personalized Cyber Face : A Novel Facial Modeling Approach using Multi-Level Radial Basis Function," *Proc. of the 2005 Int'l Conf. on Cyberworlds (CW'05)*, pp.475-482, 2005.

[7] Alvin W. K. Soh, Zhang Yu, Edmond C. Prakash, Tony K. Y. Chan, Eric Sung, "Texture Mapping of 3D Human Face for Virtual Reality Environments," *The Journal of Information Technology*, vol.8, no.2. pp.55-65, 2002.

[8] T. F. Cootes, G. J. Edwards, C. J. Taylor, "Active Appearance Models," *Proc. of the 5th European Conf. Computer Vision*, vol.2, pp.484-498, 1998.

[9] D. Cristinacce, T. Cootes, "Feature detection and tracking with constrained local models," *Proc. of the 17th BMVA'06*, pp.929-938, Scotland, Sept., 2006.

[10] D. H. Kim, J. H. Yoon, J. S. Park, "Generating Face Textures using Real Images," *Proc. of the 34th KIISE Fall Conference*, vol.34, no.2(B), pp. 156-161, 2007.