

애니메이션을 위한 통계적 모델에 기반을 둔 3D 얼굴모델링

3D Face Modeling based on Statistical Model for Animation

오두식¹, 김재민¹, 조성원¹, 정선태²
Du-Sik Oh¹ and Jae-Min Kim¹ and Seoung-Won Cho¹ and Sun-Tae Chung²

¹홍익대학교 전기정보제어공학과

E-mail: odoos@hongik.ac.kr

²승실대학교 정보통신공학부

E-mail: cst@ssu.ac.kr

요 약

본 논문에서는 애니메이션을 위해서 얼굴의 특징표현(Action Units)의 조합하는 방법으로 얼굴 모델링을 하기 위한 3D 대응점(3D dense correspondence)을 찾는 방법을 제시한다. AUs는 표정, 감정, 발음을 나타내는 얼굴의 특징표현으로 통계적 방법인 PCA (Principle Component Analysis)를 이용하여 만들 수 있다. 이를 위해서는 우선 3D 모델상의 대응점을 찾는 것이 필수적이다. 2D에서 얼굴의 주요 특징 점은 다양한 알고리즘을 이용하여 찾을 수 있지만 그것만으로 3D상의 얼굴 모델을 표현하기에는 적합하지 않다. 본 논문에서는 3D 얼굴 모델의 대응점을 찾기 위해 원기둥 좌표계 (Cylindrical Coordinates System)을 이용하여 3D 모델을 2D로 투사(Projection)시켜서 만든 2D 이미지간의 워핑 (Warping) 을 통한 대응점을 찾아 역으로 3D 모델간의 대응점을 찾는다. 이것은 3D 모델 자체를 변환하는 것보다 적은 연산량으로 계산할 수 있고 본래 형상의 변형이 없다는 장점을 가지고 있다.

키워드 : 3D 얼굴모델링, 3D 대응점, TPS, FACS

1. 서 론

2D 공간에서 표현된 얼굴 모습은 3D 공간보다 정교함이나 사실감에서 떨어진다. 3D로 표현된 얼굴은 프로그램으로 조명과 얼굴의 자세를 다양하게 조절할 수 있으므로 자연스러운 애니메이션을 위해서는 3D 얼굴 모델링이 필요하다.

지금까지 얼굴모델링에 관한 많은 연구가 계속 진행되고 있다. Mikael Rydfalk가 발표한 Candidate model은 적은 면(approximately 100 polygons)으로 얼굴의 특징을 선택하여 모델링함으로써 얼굴의 특징 표현하는데 있어서 적은 연산량으로 얼굴을 표현한다[1]. 하지만 Candidate model은 적은 면때문에 다양한 자세에서의 얼굴모델은 자연스럽지 못한 단점이 있다. Yong Cao가 제안한 모델은 적정 모델(fine mesh : approximately 10000 polygons)을 이용하여 얼굴의 주요 특징을 면이 아닌 특징 점들을 연결한 선으로 나타내었고 입력되는 음성을 분석을 통해 발음을 판단하여 사전에 등록된 3D 발음 모델을 이용하여 적정 모델을 3D 변형함으로써 자연스런 3D 얼굴 모델링을 하였다[2].

FACS(Facial Action Coding System)는 AUs들을 원소로 가진다. Ekman and Friesen이 제시한 이 모델은 64가지의 얼굴의 움직임을 표현하였다[3]. 얼굴의 특징이라 할 수 있는 눈, 눈썹, 코, 입 등이 각각 움직이는 모습과

자세를 나타내고 있다.

Yongmian Zhang는 AUs를 가지고 사람의 표정을 정의하였다. 사람의 표정은 하나의 AUs로는 표현이 불가능하다. 따라서 여러 AUs의 조합으로 구성된 얼굴의 표정을 구성하여 사람의 표정을 만듦으로써 자연스러운 표정을 만들었다.

본 논문의 구성은 다음과 같다. 2장에서는 논문을 관련된 이론을 기술하며 3장에서는 본 논문에서 제안하는 3D 대응점을 찾는 알고리즘을 기술한다. 4장에서는 실험을 통한 결과를 보여주고 5장에서는 결론 및 고찰 순으로 구성되어 있다.

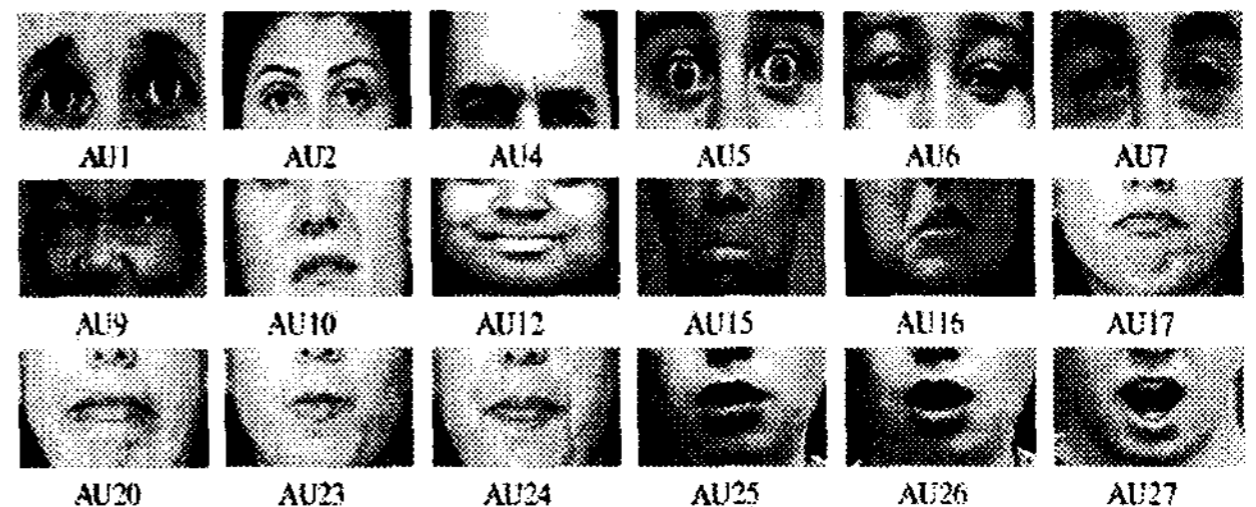


그림 1. Ekman and Friesen 제안한 AUs.

2. 관련이론

2.1 Procrustes 정규화

D.G. Kendall 에 의하면 형상(shape)은 다음과 같이 정의 된다. “형상(shape)은 물체로부터 이동, 스케일링, 회전 등의 영향이 모두 걸러진 후에 남는 모든 기하학적 정보”. 이 정의에 따르면, 형상은 유클리디언 유사 변환

감사의 글 : 본 연구는 2007년도 정부재원(중소기업청 산학협력실 지원사업)으로 중소기업청 및 서울시 지원을 받아 연구되었음.

(Euclidean similarity transformation)에 불변이다. 사람의 손의 다양한 형상으로 보일 수 있다. 같은 모양의 손 일지라도 회전 되었거나 크기가 바뀌거나 이동되어 다른 형상을 가지고 있다.

다수의 동일 물체 클래스의 형상들을 정렬하는 것을 일반 프로크루스테스 분석(generalized procrustes analysis)라 한다. 일반 프로크루스테스 분석은 다음과 같이 수행된다.

- 1) 하나의 형상을 평균 형상의 초기 추정으로 선택한다.
- 2) 각 형상의 무게 중심(centroid) 계산
- 3) 모든 형상들의 무게 중심이 원점이 되도록 이동한다.
- 4) 모든 형상을 norm2을 이용하여 형상 크기를 정규화 한다.
- 5) 각 형상을 초기 추정 형상과 비교하여 회전, 이동, 스케일링 한다.
- 6) 정렬된 형상들로부터 평균 형상을 다시 계산한다.
- 7) 처음 추정 형상과 비교하여 형상의 변화가 threshold 값을 넘지 않으면 종료한다.

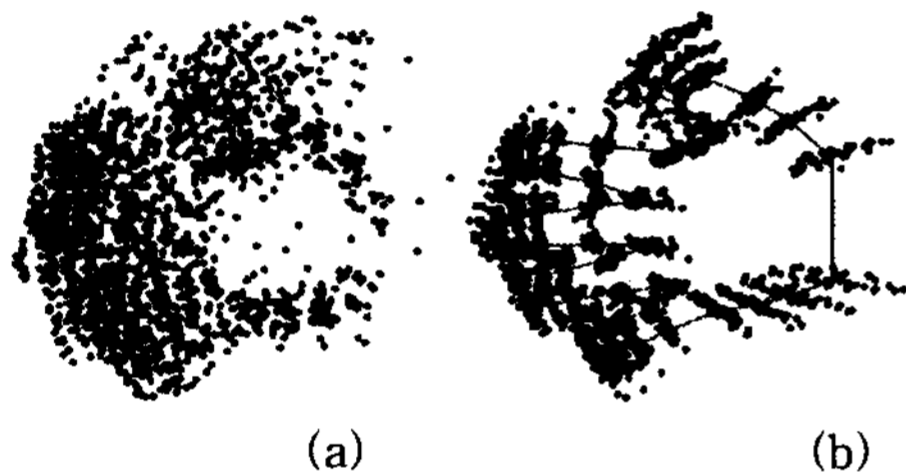


그림 2. 프로크루스테스 분석 전(a)과 후(b)의 비교 그림 좌측 그림은 프로크루스테스 분석전이어서 손을 나타내는 여러 특징점이 아무렇게나 흐트러져 있는것을 볼 수 있다. 그러나 우측 그림은 프로크루스테스 분석을 통해서 손의 특징 점들이 비슷한 위치로 정렬되어있는 것을 보여준다[4].

2.2 TPS(Thin Plate Spline)

TPS는 비강체(non-rigid) 스플라인 매핑 함수 클래스로 공간상에 흩어져 있는 점들을 매우 미끈하게(smoothing) 연결하는 곡면을 구하는데 유용한 도구이다. TPS는 쉽게 계산할 수 있으며, 선형 항과 비선형 항으로 분리되며, 전역적으로 미끈한 매핑이 되도록 하는 최소 비선형 항을 포함한다.

2D 공간에 참조 얼굴과 샘플 얼굴이 있다고 가정하자. 평균얼굴의 평균점을 $Q = \{Q_1, Q_2, \dots, Q_n\}$ 라 하고, 위 평할 샘플영상의 제어 점을 $P = \{P_1, P_2, \dots, P_n\}$ 라 하자. 이때 P 에서 Q 로의 미끈한 벡터 매핑 함수 f , $f(P_i) = Q_i$ for $i = 1, 2, \dots, n$ 가 있으면 샘플 얼굴의 모든 제어점은 f 에 의해 참조 얼굴 형상 공간으로 매핑된다.

이제 함수 f 를 TPS로 구한다고 하면 다음과 같이 표현된다.

$$f(O) = \sum_{i=1}^n \Omega_i U(\|O - P_i\|) + A_0 + A_1x + A_2y + A_3z \quad (1)$$

식 (1)에서 O 와 P_i 는 2차원 점으로

$O = (x, y)^T, P_i = (x_i, y_i)^T$ 이다.

또한, $U(\|O - P_i\|) = \|O - P_i\|$ 이며, 가중치 Ω_i 는 다음의 제약 조건을 만족해야 한다.

$$\sum_{i=1}^n \Omega_i = 0, \sum_{i=1}^n \Omega_i x_i = 0, \sum_{i=1}^n \Omega_i y_i = 0 \quad (2)$$

이제, 함수 f 는

$f(P_i) = Q_i$ for $i = 1, 2, \dots, n$ 를 만족하므로, 식(2)를 만족하는 계수 $\Omega_i (i = 1, 2, \dots, n)$, A_0, A_1, A_2, A_3 는 다음의 선형 관계식을 만족한다.

$$\begin{bmatrix} K & P \\ P^T & 0 \end{bmatrix} \begin{bmatrix} \Omega^\alpha \\ A^\alpha \end{bmatrix} = \begin{bmatrix} Q^\alpha \\ 0 \end{bmatrix} \quad (\alpha = x, y) \quad (3)$$

이때, K 는 i, j 번째 항이

$K_{ij} = U(\|(x_i, y_i) - (x_j, y_j)\|)$ 으로 표현되는 $n \times n$ 행렬이며, P 는 i 번째 행이 $(1, x_i, y_i)$ 인 $n \times 3$ 행렬이다.

또한 Ω^α 는 $\Omega^\alpha = (\Omega_1^\alpha, \Omega_2^\alpha, \dots, \Omega_n^\alpha)^T$ 이며, Ω_i^α 는 Ω_i 의 $\alpha = (x, y)$ 좌표를 나타내며, 마찬가지로 $A^\alpha = (A_0^\alpha, A_1^\alpha, A_2^\alpha, A_3^\alpha)^T$ 이며, A_i^α 는 A_i 의 $\alpha = (x, y)$ 좌표를 나타내며,

$Q^\alpha = (Q_1^\alpha, Q_2^\alpha, \dots, Q_n^\alpha)^T$ 이며, Q_i^α 는 Q_i 의 $\alpha = (x, y)$ 좌표를 나타낸다.

식 (3)를 풀면 식 (1)의 벡터 TPS 각 계수 벡터 $\Omega_i, A_0, A_1, A_2, A_3$ 를 구할 수 있다.

3. 제안하는 알고리즘

본 논문에서 제안하는 3D 형상간의 대응점 찾기는 다음과 같은 과정으로 이루어진다. 그림 3는 3장에서 제안하는 알고리즘의 대략적인 모습을 나타내었다.

- step1 3D 데이터변환
- step2 수동 점찍기
- step3 TPS
- step4 3D - 2D 간의 대응점 찾기

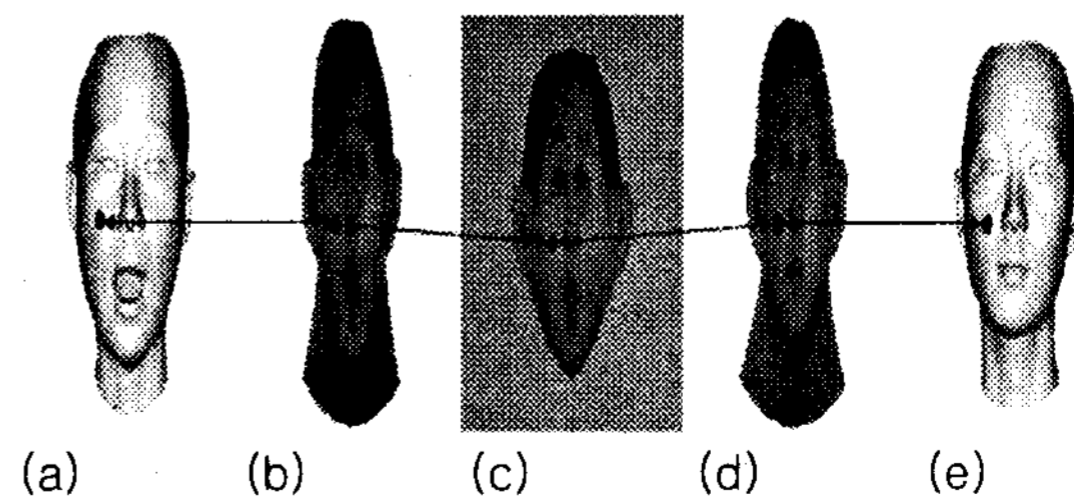


그림 3. 3D 형상간의 대응점 찾는 과정 step1. (a) -> (b) (3.1절) step2. (b) step3. (b) > (c)

step4. (a) -> (c) (3.2절)

3.1 원기둥 좌표계(Cylindrical Coordinates System)를 이용한 2D 영상(Image) 생성

장비로 획득한 얼굴 데이터는 얼굴의 굴곡면에 대해서 각 점(vertex)과 각 점의 칼라 정보(RGB) 그리고 점으로 구성된 면(face)의 정보를 가지고 있다. 이를 원기둥좌표계를 이용하여 2D에 사상(Projection)하여 2D 영상을 만들었다. 그림4의 (a) 사진에 얼굴의 모습에서 흰색으로 된 구멍(hole)을 볼 수 있다. 이는 2D로 사상할 때 영상의 양자화(quantization) 때문에 나타나는 현상이다. 즉 흰색으로 된 영상의 픽셀에는 3D 데이터가 존재하지 않는다는 것을 의미한다. 3D 데이터를 생성할 때 형상을 얼마나 조밀하게 스캔하는 문제와 2D 영상사이즈를 얼마큼 할 것인가의 문제에 따라 흰색으로 된 픽셀의 양은 달라진다. 본 논문에서는 실험적으로 최적이 영상을 얻을 수 있는 영상의 크기를 선택하여 실험하였다.

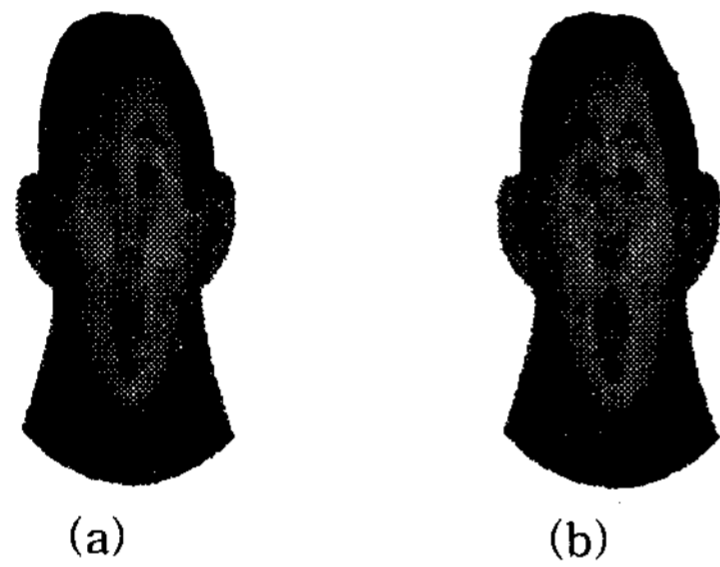


그림 4.(a) 원형좌표계에 매핑된 2D 텍스처 사진 (Cylindrical Image)이다.
(b) 수동으로 원형에 매핑된 2D 텍스처 사진에 84의 제어 점을 찍은 사진이다.

3.2 3D 대응점 찾기 (3D Dense Correspondence)

본 논문에서는 얼굴의 조명변화나 표정에 상관없이 주어진 제어 점을 가지고 TPS과정을 통해 데이터를 얻었다. 과정은 다음과 같다.

step1에서 생성된 파일을 S 로 정의한다. S_i 는 다음과 같이 구성된다.

$$S_i = \{x_i, y_i, z_i, r_i, h_i, t_i, rc_i, gc_i, bc_i, ix_i, iy_i\}$$

x, y, z 는 3D의 vertex를 의미한다. r, h, t 는 3D의 vertex를 원형좌표계로 식4로 변환한 값을 나타낸다.

rc, gc, bc 는 3D의 vertex에서의 RGB값을 의미한다. 마지막으로 ix, iy 는 2D 영상으로 사상된 x축 y축 좌표를 나타낸다.

$$r = \sqrt{(x^2 + y^2)} \quad h = z$$

$$t \text{ (degree)} = \frac{180}{\cos^{-1} \frac{x}{r}} \times \pi$$

(4)

S_i 는 $S = \{S_1, S_2, \dots, S_n\}$ 데이터 묶음으로 만들어진다.

step3에서 실행하면 각 표정의 ix, iy 의 좌표는 평균 얼굴(reference shape)의 rx, ry 로 역사상 된다.

step4에서 step1에서의 x, y, z 는 step3의 결과의 rx, ry 로 사상된다. 즉, step4의 결과로 평균얼굴의 각

픽셀의 rx, ry 좌표의 점들은 3D형상의 x, y, z 좌표가 나온다.

TPS의 정확도는 본 논문에서 제안하는 3D 대응점을 찾는 알고리즘에서 가장 중요한 부분이다. 4장에서는 제안하는 알고리즘의 결과와 정확도에 대해서 나타내었다.

4. 실험

본 장에서는 제안하는 알고리즘으로 만든 3D 얼굴 모델과 워핑의 정확성을 보여줄 것이다.

그림 6은 제안하는 알고리즘으로 생성된 3D 대응점의 결과를 보여준 그림이다. 이 결과의 장점은 다양한 발음이나 표정에서도 적용이 가능하다.

워핑의 정확도는 제안하는 3D 대응점을 찾는 알고리즘에서 가장 중요하다. 표 1은 동일점을 이용하여 워핑 결과의 에러를 확인한 것이다. 두 워핑 모두 워핑의 오차가 거의 한 픽셀정도 차이나는 것을 볼 수 있다. 이는 계산상 양자화로 인한 문제라 할 수 있고 3D 좌표로 하면 기준점의 바로 옆 점에 해당되며 거의 무시될 수 있다.

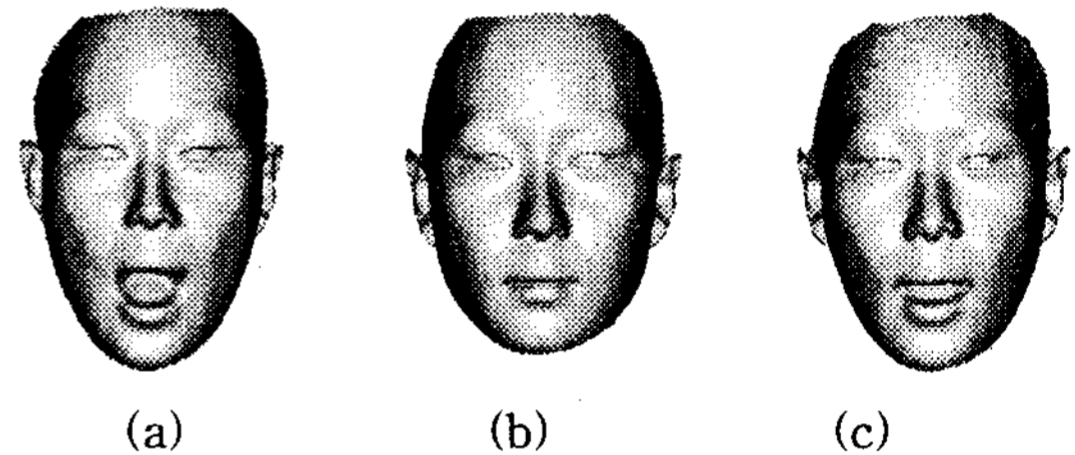


그림 6. 제안하는 방법으로 얻은 각 발음에서의 3D 대응점 (a) 아 (b) 무표정 (c) 에

표1. 워핑의 정확도

(T :Thin Plate Spline M: Mesh Warping)

종류 \ 거리(%)	0	1	$\sqrt{2}$	기타
T	88.2	8.3	2.7	1.8
M	93.2	4.5	1.2	1.1

TPS을 선호한 이유는 Mesh Warping의 경우 mesh범위가 벗어나는 영상의 부분들은 워핑이 될 수 없기 때문에 데이터의 상실을 가져올 수 있고 또한 mesh를 매번 생성해야하는 번거로움이 있다.

5. 결론 및 고찰

자유다면체에서 다각형 표면을 따라가는 3D 변형은 같은 표정에서는 뛰어난 성능을 보인다. 하지만 다양한 3D 데이터(표정의 다른 경우)에서는 정확한 3D 대응점을 찾을 수 없다[5]. 하지만 제안한 방법의 데이터를 변화하는 과정에서 오차를 가지고 있다. 이를 보완하기 위해서 제안하는 방법을 초기 값으로 하여 3D 굴 곡면을 찾아가는 방법이 연구되어야 할 것이다.

참 고 문 헌

- [1] J. Ahlberg, "CANDIDE-3 -- an updated parameterized face", Report No. LiTH-ISY -R-2326, Dept. of Electrical Engineering, Linkoping University, Sweden, 2001.
- [2] YONG CAO, " Expressive Speech-Driven Facial Animation", ACM Transactionson Graphics, Vol.V, No.N, Month20YY.
- [3] P.Ekman and W.V.Friesen, Facial Action Coding System. Palo Alto, CA: Consulting Psychologists Press 1977
- [4] Stegmann, Mikkel B., Gomez , David Delgado: A Brief Introduction to Statistical Shape Analysis, Technical University of Denmark, Lyngby, 2002.
- [5] 김재민, "3차원 다각형 표면의 자유 변형 추정 ("Estimation of Free-Form Deformation of 3D Polyhedral Surfaces" 특허 출원번호 10-2001-0071117