

얼굴 표정의 자동 인식을 통한 PDA 상에서의 3차원 얼굴 애니메이션

이 돈 수[†] · 최 수 미^{††} · 김 해 광^{†††} · 김 용 국^{†††}

요 약

본 논문에서는 일곱 가지의 기본적인 감정 정보를 자동으로 파악하고 얼굴을 PDA 상에서 렌더링할 수 있는 얼굴 표정의 인식 및 합성 시스템을 제시한다. 얼굴 표정 인식을 위해서 먼저 카메라로부터 획득한 영상으로부터 얼굴 부분을 검출한다. 그리고 나서 기하학적 또는 조명에 의한 보정을 위해 정규화 과정을 거친다. 얼굴 표정을 분류하기 위해서는 Gabor wavelets 방법을 enhanced Fisher 모델과 결합하여 사용할 때가 가장 좋은 결과를 보였다. 본 표정 분류에서는 일곱 가지 감정 가중치가 결과로 제시되고, 그러한 가중 정보는 모바일 네트워크를 통하여 PDA 상으로 전송되어 얼굴 표정 애니메이션에 이용되어진다. 또한 본 논문에서는 고유한 얼굴 캐릭터를 가진 3차원 아바타를 생성하기 위하여 카툰 셰이딩 기법을 채택하였다. 실험 결과 감정 곡선을 이용한 얼굴 표정 애니메이션은 선형 보간법 보다 감정 변화의 타이밍을 표현하는데 더 효과적인 것으로 나타났다.

키워드 : 표정 인식, 얼굴 애니메이션, 비사실적 렌더링, 모바일 플랫폼

3-D Facial Animation on the PDA via Automatic Facial Expression Recognition

Don-Soo Lee[†] · Soo-Mi Choi^{††} · Hae-Kwang Kim^{†††} · Yong-Guk Kim^{†††}

ABSTRACT

In this paper, we present a facial expression recognition-synthesis system that recognizes 7 basic emotion information automatically and renders face with non-photorealistic style in PDA. For the recognition of the facial expressions, first we need to detect the face area within the image acquired from the camera. Then, a normalization procedure is applied to it for geometrical and illumination corrections. To classify a facial expression, we have found that when Gabor wavelets is combined with enhanced Fisher model the best result comes out. In our case, the out put is the 7 emotional weighting. Such weighting information transmitted to the PDA via a mobile network, is used for non-photorealistic facial expression animation. To render a 3-D avatar which has unique facial character, we adopted the cartoon-like shading method. We found that facial expression animation using emotional curves is more effective in expressing the timing of an expression comparing to the linear interpolation method.

Key Words : Face Recognition, Facial Animation, Non-Photorealistic Rendering, Mobile Platform

1. 서 론

인간과 인간 또는 인간과 컴퓨터 간의 상호작용을 위해 얼굴 표정을 통하여 사용자의 감정 상태를 자동으로 파악하고 이를 시각적으로 표현하는 기술은 매우 중요하다. 얼굴 표정에 관한 연구는 사회적인 측면에서 매우 중요하기 때문에 인지심리학자들의 오랜 연구 주제가 되어 왔다. 이러한 오랜 연구들을 토대로 최근에는 컴퓨터를 이용하여 자동으

로 인간의 얼굴 표정을 인식 및 분류하고 나아가 다양한 장비에서 인간의 자연스러운 감정을 표현할 수 있는 표정 합성에 대하여 관심이 집중되고 있다[1].

얼굴 표정의 인식에 관한 연구에는 영상으로부터 전체적으로 표정을 인식하는 Holistic 접근방법과, 얼굴의 특징점들을 트래킹하는 기하학적 접근방법이 있다[11]. 얼굴 표정을 분류하는 방법은 공간적 접근방법과 공간-시간적 접근방법으로 나누어 볼 수 있다. 공간적 접근방법에서는 Neural Networks 들이 자주 사용되었는데 이를 단독으로 적용했거나, Principal Component Analysis(PCA), Independent Component Analysis (ICA), Gabor Wavelets Filter 등과 합성하여 사용하였다. 또한 표정 인식의 복잡도에 따라 레이어를

[†] 준 회원 : 세종대학교 컴퓨터공학부
^{††} 정 회원 : 세종대학교 컴퓨터공학부 조교수
^{†††} 정 회원 : 세종대학교 컴퓨터공학부 부교수
 논문접수 : 2005년 6월 13일, 심사완료 : 2005년 10월 31일

나누어서 처리하거나, Support Vector Machines (SVM)을 사용하여 연구되어졌다. 공간-시간적 접근방법에는 얼굴의 움직임을 Spatio-Temporal 모션 에너지로 표현하여 표정을 분류한 방법, Hidden Markov Model (HMM)을 이용하여 얼굴 표정을 여섯가지 기본 표정으로 분류한 방법, HMM을 이용하여 얼굴 표정을 FACS(Facial Action Coding System)의 Units으로 분류하는 방법, 계층적인 Dynamic Bayesian Network를 이용한 방법 등이 있다.

얼굴 표정의 합성에 관한 접근방법은 기하학적 접근방법과 영상기반 접근방법으로 나누어 볼 수 있다[2, 10]. 기하학적인 접근방법에는 키프레임 보간법, 파라메트릭 모델, 물리적 근육 모델, 유한요소 모델, 시뮬레이티드 근육 모델 등이 있다. 키 프레임 보간법은 빠르지만 제약이 많고, Parke와 Waters 에 의해 제안된 파라메트릭 모델은 독립적인 파라미터 값들의 조합에 의해 좀 더 자유스러운 표정을 생성할 수 있었다. 물리적 근육 모델에는 근육에 적용되는 힘을 탄성 스프링에 적용함으로써 피부 표면의 변형을 표현하는 매스-스프링 모델(mass-spring model), 벡터 근육을 얼굴 메쉬와 독립적으로 움직이므로써 근육의 움직임을 컴팩트하게 표현하는 벡터 근육 모델, 세밀한 해부학적 안면 구조와 얼굴의 동역학을 고려함으로써 사실적이지만 많은 계산이 필요한 계층적 스프링 메쉬 근육 모델(layered spring mesh muscle) 등이 있다. 유한요소 모델은 얼굴 안면을 유한요소로 분할한 후 시간의 흐름에 따른 동역학적인 요소들의 관계를 수치적으로 적분한 것이다. 이외에 근육을 시뮬레이션하기 위하여 Free Form Deformation(FFD) 또는 스플라인을 이용한 모델 등이 있다. 영상기반 접근방법에는 타겟 영상간의 모핑을 이용한 2D & 3D 모핑 방법, 얼굴 피부색 변화를 위해 텍스처 맵핑 또는 폴리곤 칼라 변화를 이용한 방법, 와이어 프레임 모델을 주어진 얼굴 영상에 적합시킴으로써 표정을 합성하는 방법, 액터의 얼굴 표정의 특징점들을 트래킹하여 얼굴의 표정을 합성하는 performance-driven 방법 등이 있다.

본 논문에서는 비디오 영상으로부터 얼굴 표정을 사용자 개입 없이 자동으로 인식하여 시간에 따라 변화하는 감정 정보를 추출하고, 이때 얻어진 감정 곡선을 기반으로 PDA 상에서 얼굴 표정을 3차원 애니메이션 하는 전체 파이프라인을 제시한다. 3차원 애니메이션을 위해 자주 사용되는 두 모델 간의 선형 보간법은 점진적으로 변화하는 얼굴 표정을 생성하기 때문에 표정 변화의 타이밍을 잘 나타내기 어렵다. 애니메이터들은 이러한 어색한 얼굴 애니메이션을 보완하고자 미리 중간 표정 즉, 선행 동작(anticipation)을 준비하여 보간하는 기법을 사용하고 있다[12]. 본 논문에서는 자동 표정인식 시스템으로부터 인식된 감정 정보를 이용하여 표정 변화의 타이밍을 보다 잘 표현하였다. 그리고 사용자의 감정을 강조하여 전달하는 한편, 그래픽 하드웨어의 지원이 없는 PDA 상에서 구동 가능하도록 만화적인 비사실적 렌더링 방식을 채택하였다.

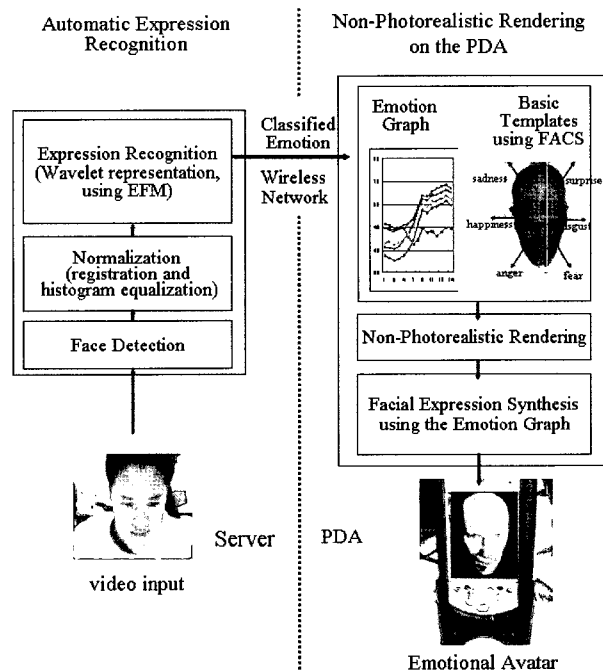
본 논문의 구성은 다음과 같다. 2장에서는 전체 시스템의

구조에 대해서 설명한다. 3장과 4장에서는 얼굴 표정의 자동 인식과 감정 곡선에 기반한 표정 애니메이션 방법에 대하여 각각 기술한다. 5장에서는 구현 결과를 제시하고, 6장에서는 결론 및 향후 연구 방향을 제시한다.

2. 표정 인식 및 합성 시스템 구조

전체 시스템은 크게 두 부분으로 구성된다. 즉, 서버에서 얼굴 표정을 자동으로 인식하는 부분과 PDA 상에서 얼굴 표정을 합성하는 부분으로 나누어진다. (그림 1)은 이러한 시스템의 전체 구조도를 보여준다.

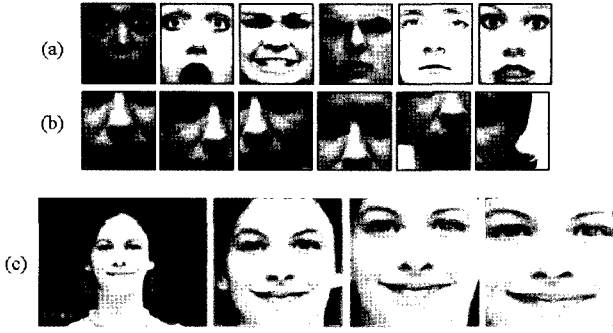
표정 인식 과정에서는 서버 PC에 부착된 카메라로 입력되는 사용자 얼굴 표정을 자동으로 인식하기 위하여 먼저 실시간 입력 영상에서 얼굴 부위를 검출하고, 표정 분류기에 맞는 형태로 정규화 한다. 그리고 학습된 무표정 및 여섯 가지 기본 감정(surprise, fear, disgust, anger, happiness, sadness) 데이터를 바탕으로 입력 영상과 학습된 영상과의 유사한 정도를 정량적으로 측정한다. 유사도에 따른 감정 곡선은 네트워크를 통해 PDA 로 전송된다. 애니메이션 과정에서는 미리 생성된 감정 템플릿 모델을 사용하여 전송된 감정 정보의 변화에 따라 3차원 얼굴을 합성한다.



(그림 1) 표정 인식 및 합성 시스템의 구조도

3. 얼굴 표정의 자동 인식

본 논문에서 제안하는 자동 얼굴 표정 인식 시스템은 세 가지 단계로 구성된다. 첫 번째 단계는 입력되는 영상 내에 포함된 얼굴 영역을 추출하는 얼굴 검출(face detection) 과정이고, 두 번째 단계는 추출된 얼굴 내의 특징점을 찾아



(그림 2) SVM 신경망을 통한 얼굴 영역 검출

영역을 정규화(normalization)하는 과정이다. 마지막 과정은 이렇게 정규화 된 얼굴을 분류기를 적용하여 표정을 분류(classification)하는 것이다[3].

얼굴 검출 과정에는 SVM(Support Vector Machine) 기반의 기계학습으로 훈련된 신경망이 사용되었다[14]. (그림 2) (a)와 (b)는 각각 얼굴영상과 비 얼굴 영상의 예를 보여 준다. (그림 2) (c)는 얼굴 영역이 검출된 후 정규화 과정을 위해 두 눈과 입의 위치 정보를 검출하는 장면이다.

(그림 3) (a)는 정규화 과정으로 두 눈과 입의 위치 정보를 이용하여 생성된 축을 통해 회전 및 크기의 영향을 고려하여 얼굴 영역 이미지를 일정한 사각형 영역에 피는 것이다. 정규화 과정은 얼굴 표정인식 시스템의 전체 성능을 좌우하는 중요한 단계이다. 얼굴 검출과 정규화 과정을 거친 이미지는 일정한 간격으로 그려진 격자(20x20)에 Gabor Wavelet을 적용하여 콘볼루션된 출력은 이 시스템의 세 번째 단계인 얼굴표정 분리기(facial expression classifier)로 보내진다. 본 연구에서는 PCA(Principle Component Analysis) 기법을 향상시킨 EFM(Enhanced Fisher Discriminant Model)을 사용하였다.

PCA는 주성분이라 부르는 직교 기저 벡터의 집합을 생성하는데 이것은 모든 투영 표본의 분산을 최대로 하는 것이다. 이미지를 표현하는 랜덤 벡터로 만들고, 여기서 N 은 이미지 공간에 대응하는 차원이다. 공분산 행렬 X 는 다음 식 (1)과 같이 정의된다.

$$\sum X = E\{[X - E(X)][X - E(X)]^t\} \quad (1)$$

여기에서 $E()$ 는 기대 연산자이며 t 는 전치 연산 $\sum \in R^{NXN}$ 로 표시한다. 랜덤 벡터 X 의 PCA는 아래 식 (2)의 형태로 공분산 행렬 $\sum X$ 를 인수화한다.

$$\sum X = \Phi \Lambda \Phi^t \quad (2)$$

$\Phi = [\Phi_1 \Phi_2 \dots \Phi_N] \in R^{NXN}$ 는 고유벡터의 직교 행렬이며 $\Lambda = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_N\} \in R^{NXN}$ 는 내림차순으로 정렬된 $(\lambda_1, \lambda_2, \dots, \lambda_N)$ 대각성분에서의 고유값 대각행렬이다. 다음 식 (3)에서 첫 번째 m 에 따르는 고유벡터 행렬 P 가 정의된다.

$$P = [\Phi_1, \Phi_2, \dots, \Phi_m] \quad (3)$$

PCA 어플리케이션은 다음식 (4)와 같이 차원 감소의 형태를 가진다.

$$Y = P^t X \quad (4)$$

(그림 3)의 (b)에 보여 지고 있는 것처럼 저차원 벡터 $Y \in R^m$ 는 원래 데이터 X 의 가장 두드러진 특징들을 가진다. 축소된 이미지 공간의 차원을 결정하는 EFM을 표현하기 위해 저차원의 특징을 표현하는 랜덤 벡터를 Y 라 하고, 각 클래스에서의 이미지 수와 클래스들을 각각 w_1, w_2, \dots, w_L 와 N_1, N_2, \dots, N_L 이라 표시하자. 그리고 M_1, M_2, \dots, M_L 과 M 은 클래스들의 평균과 전체 평균이다. 내부 클래스(within-class)와 중간 클래스(between-class)의 공분산 메트릭스 $\sum w$ 와 $\sum b$ 는 다음식 (5)와 (6)과 같이 정의된다.

$$\sum w = \sum_{i=1}^L P(w_i) E\{(Y - M_i)(Y - M_i)^t w_i\} \quad (5)$$

$$\sum b = \sum_{i=1}^L P(w_i) (M_i - M)(M_i - M)^t \quad (6)$$

EFM 첫 번째 대각행렬 내부클래스의 공분산 메트릭스 $\sum w$ 는 식 (7)과 (8)이다.

$$\sum w E = E \Gamma \text{ and } E^t E = I \quad (7)$$

$$\Gamma^{-1/2} E^t \sum w E \Gamma^{-1/2} = I \quad (8)$$

여기서 E, Γ 는 $\sum w$ 의 고유벡터와 고유값 대각 행렬들이다. EFM은 식 (9)와 같은 중간클래스의 공분산 행렬을 계산한 후에 진행 한다.

$$\Gamma^{-1/2} E^t \sum b E \Gamma^{-1/2} = K_b \quad (9)$$

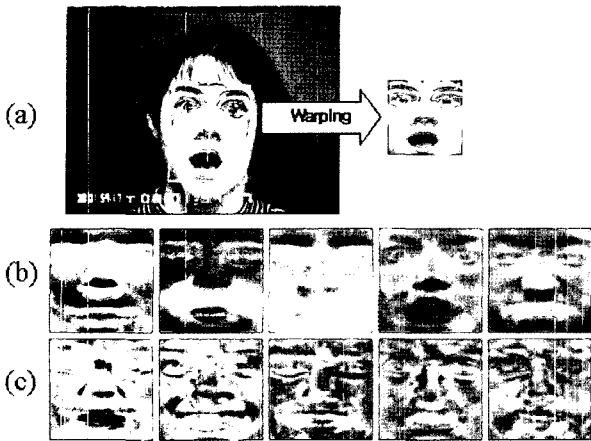
새로운 중간클래스의 공분산 행렬 K_b 를 식 (10)처럼 대각행렬화 한다.

$$K_b \Theta = \Theta \Delta \text{ and } \Theta^t \Theta = I \quad (10)$$

여기서 Θ, Δ 는 K_b 의 고유벡터와 고유값 대각 행렬들이다. EFM의 전체 변환 행렬은 다음 식 (11)과 같이 정의된다.

$$T = E \Gamma^{-1/2} \Theta \quad (11)$$

(그림 3) (b)는 임의로 주어진 영상에 PCA를 적용한 영상이고 (c)는 EFM을 적용한 영상이다. EFM에 Gabor wavelet filter를 추가하여 적용 했을때 10% 정도 인식률이 높아졌다.



(그림 3) 얼굴 영역 정규화 및 표정 분류 과정

<표 1> 유사도 측정 (k=80)

(%)	EFM	Gabor +EFM
L2	90	93

얼굴 표정인식에 사용된 DB는 Cohn-Kanade[9]의 얼굴표정 DB로 7개의 표정에 대해 각각 100개씩 700개를 트레이닝 이미지로 선택하여 얼굴영역을 검출하여 정규화하고 Gabor 필터링한 후에 EFM을 위한 컴포넌트를 생성하였다.

컴포넌트 개수는 80개로 하였고 생성된 Gabor-EFM 표정모델에 입력된 얼굴 이미지를 투영하여 각 모델과 유사도(L2)를 계산하여 가장 근접된 표정을 선택하였다. 인식률 테스트는 Cohn-Kanade DB에서 트레이닝에 사용하지 않은 나머지 이미지중 각 표정별로 100개씩 700개의 이미지로 하였다. 그 결과는 다음과 같다.

4. 감정 곡선에 기반한 PDA 상에서의 얼굴 애니메이션

본 장에서는 서버로부터 전달된 감정 정보를 이용하여 PDA 상에서 얼굴 표정을 애니메이션 하는 방법에 대하여 설명한다. PDA는 그래픽 연산을 위한 하드웨어적인 가속기가 없기 때문에 본 논문에서는 최소한의 폴리곤 얼굴 모델을 구성하고 간단한 근육 모델을 기반으로 기본적인 표정 템플릿을 생성하였다. 여기에 만화적인 비사실적 렌더링 기법을 적용하여 친근하고, 개성적인 3차원 얼굴 표정 애니메이션을 PDA 상에서 구현하였다.

4.1 기본 표정 템플릿 생성

먼저 표정 인식기에 의해 분류된 기본 표정들을 위해서 Facial Action Coding System (FACS)과 근육 모델에 기반한 표정 템플릿을 생성한다. FACS는 얼굴 표정을 기술하기 위한 방법으로 Ekman과 Friesen에 의해 개발된 방법이다. 이 방법은 인간의 얼굴에서 움직일 수 있는 액션 유닛(action units; AUs)에 기초하고 있다[5, 6]. <표 2>는 기본

<표 2> 기본 감정과 연관된 액션 유닛의 예

Emotions	Upper	Midsection	Lower
Surprise	AU1+AU2		AU26
Fear	AU1+AU2+AU4	AU20	AU15 and/or AU16
Disgust	AU4	AU9 and/or AU10	AU17
Anger	AU2+AU4, AU5	AU10	AU24 or AU25
Happiness		AU6	AU11, AU12
Sadness	AU1+AU2+AU4		AU15

Upper Face> AU1: Inner Brow Raiser, AU2: Outer Brow Raiser, AU4: Brow Raiser, AU5: Upper Lid Raiser
 Midsection> AU6: Cheek Raiser, AU9: Nose Wrinkler, AU10: Upper-Lip Raiser, AU20: Lip Stretcher
 Lower Face> AU11: Nasolabial Furrow Deepener, AU12: Lip Corner Puller, AU15: Lip Corner Depressor, AU16: Lower Lip Depressor, AU17: Chin Raiser, AU24: Lip Pressor, AU25: Parting of Lips, AU26: Jaw Drop

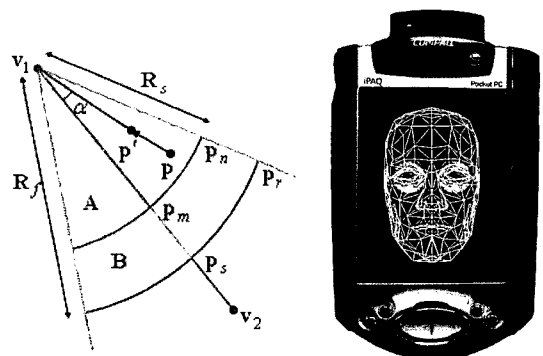
적인 여섯 감정을 표현할 때 얼굴 부위(윗부분, 중간부분, 아랫부분)에서 자주 나타나는 움직임과 이와 연관된 액션 유닛의 예를 보여준다. 음영 부분은 주된 암시가 되는 움직임들로 놀라는 표정에서는 눈썹이 올라가고, 애타한 표정을 지을 때는 윗 입술이 올라가면서 코등을 당기게 된다. 그리고 기쁜 표정에서는 보편적으로 입술의 양 끝이 올라가게 된다.

기본 템플릿 생성을 위해 사용한 근육은 얼굴의 뼈와 피부사이에 존재하는 선형 근육을 이용하였다[15]. 즉, 선형 근육은 뼈의 한점 v1에서 피부의 한점 v2 로 이어지는 벡터로 정의된다 (그림 4) (a). 이때 벡터의 영향을 받는 부분은 삼각 함수로 정의되며 (그림 4) (a) 에서처럼 삼각뿔 영역으로 정의된다.

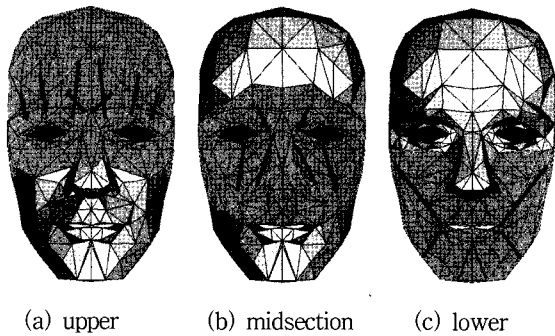
(그림 4) (a)에서 (v1, p, ps) 영역에 속하는 임의의 점 p 가 새로운 점 p' 로 옮겨지는 것은 식 (12)에 의해 계산된다.

$$p' = p + \cos(\alpha)kr(pv_1 / \|pv_1\|) \tag{12}$$

$$r = \begin{cases} \cos((1-D)/R_s) & \text{for } p \text{ insidezone}(v_1, p_n, p_m) \\ \cos((D-R_s)/(R_f - R_s)) & \text{for } p \text{ insidezone}(p_n, p_r, p_s, p_m) \end{cases} \tag{13}$$



(a) 선형 근육 모델 (b) PDA 상의 얼굴 매쉬
(그림 4) 근육 기반의 얼굴 모델



(그림 5) 얼굴의 영역 분할

여기서 α 는 벡터 (v_1, v_2) 와 (v_1, p) 사이의 각도이며, D 는 $\|v_1 - p\|$, k 는 피부의 탄성도를 표현하는 고정된 상수이다. r 은 반경방향 변위 파라미터로 점 p 가 영역 R_s 또는 R_t 에 있는 지에 따라서 식 (13)과 같이 계산되어진다. 선형 근육 모델은 폴리곤의 수에 따라 계산량이 많아짐으로 적은수의 폴리곤 모델에 잘 적용된다.

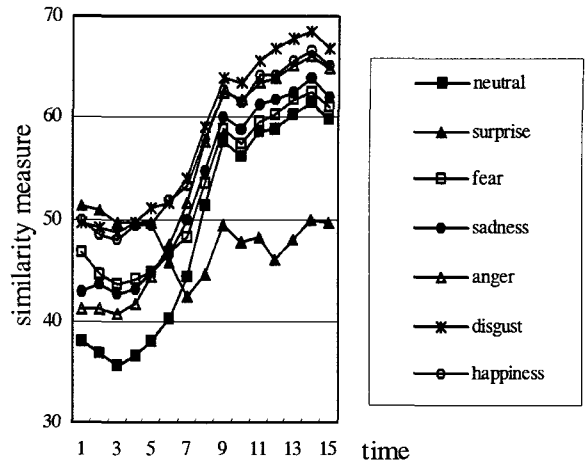
기본 템플릿을 위한 얼굴 메쉬는 762개의 폴리곤과 476개의 정점으로 구성되어 있다(그림 4) (b). 이들 중에 296개 정점과 474개의 폴리곤이 전체 얼굴을 구성하며, 180개의 정점과 288개의 폴리곤은 눈을 구성한다. 본 논문에서는 계산 속도를 높이기 위하여 3차원 얼굴 메쉬를 (그림 5)와 같이 세 개의 영역(upper, midsection, lower)으로 나누어 처리하도록 하였다[13]. 예를 들어 눈썹 부분에 속해 있는 근육이 움직일 경우 턱 아래 부분에 있는 정점들에는 영향을 미치지 않는다. 그러므로 각 폴리곤에 플래그 값을 지정하여, 불필요한 연산을 줄임으로써 계산 속도를 향상시킬 수 있다. 그리고 놀라는 표정과 같이 입을 벌리는 표정을 생성하기 위해서는 아랫 입술 부분의 폴리곤을 턱의 회전축에 바인딩하는 방법을 사용하였다. 이때 자연스러운 입술 모양을 생성하기 위해 아래 입술을 구성하는 폴리곤에 바인딩 가중치 값을 달리하여 적용하였다.

4.2 감정 곡선에 기반한 비선형적인 표정 애니메이션

키 프레임 모델 간을 보간하는 방법은 3차원 애니메이션을 위해 널리 사용되는 방식이다. 즉, 시작 표정을 E_1 , 목적 표정을 E_2 라고 했을 때, 중간 표정 E 는 식 (14)와 같이 계수 α 를 변경시킴으로써 생성되어진다.

$$E = \alpha E_1 + (1 - \alpha) E_2 \quad 0.0 < \alpha < 1.0 \quad (14)$$

일반적으로 보간 계수 α 는 애니메이션 프레임 타임을 이용한다. 이러한 선형 보간법은 점진적으로 변화하기 때문에 부드러운 중간 표정을 생성할 수 있다. 그러나 일반적인 사람의 표정은 선형적으로 변하지 않기 때문에 선형 보간식을 그대로 적용하면 감정 변화가 이루어지는 타이밍을 표현하기가 어렵다. 예를 들어, 놀라는 표정의 경우 점진적을 놀란다가 보다는 갑자기 이루어지게 되고 이러한 타이밍의 차이



(그림 6) 자동 표정인식 시스템으로부터 얻어진 감정곡선

는 애니메이션 결과를 보면 더 뚜렷하다. 본 연구에서는 표정을 자동 인식하여 얻은 감정 곡선을 기반으로 중간 표정을 생성함으로써 표정 변화 타이밍을 잘 포착할 수 있도록 하였다.

(그림 6)은 중립 표정에서 놀라는 표정으로 변화하는 감정 곡선의 예이다. 감정 곡선은 중간 표정을 생성하는 가중치 값으로 이용되는데, 수평 축은 시간의 변화를 나타내고 수직축은 기본 표정들과의 유사도를 나타낸다. 각 감정과의 유사도는 자동 표정 인식 시스템에 의해 측정되므로 실시간 변화하는 감정 가중치를 각 프레임별로 구할 수 있다. 각 프레임 타임 t 에서 수직축의 값이 가장 적음(즉, 유사도 높음) 그래프가 현재의 주된 감정을 나타낸다. 이렇게 함으로써 비선형 표정 애니메이션을 생성할 수 있다.

주어진 애니메이션 프레임 타임 t 에서의 실제 파라미터 값 $F_p(t)$ 는 식 (15)와 같이 가중 평균을 이용하여 표정들을 합성한다. 식 (15)에서 n 은 인식기를 통해 분류되는 표정의 갯수이며, T_{ep} 는 목적 표정을 위한 파라미터이고, $W_{ep}(t)$ 는 시각 t 에서의 해당 표정에 대한 가중치이다.

$$F_p(t) = \sum_{e=1}^n (W_{ep}(t) T_{ep}) / \sum_{e=1}^n W_{ep}(t) \quad (15)$$

일반적으로 사람과 유사한 아바타를 생성하기 위해서는 사실적인 렌더링(photorealistic rendering) 방법이 많이 사용되어진다. 이와 반면에 얼굴을 비사실적으로 렌더링(non-photorealistic rendering)하면 사용자의 감정을 과장되게 또는 재미있게 전달할 수 있다는 장점이 있다. 뿐만 아니라 여기서 사용한 카툰 셰이딩 기법은 비교적 계산량이 적어 상대적으로 계산 성능이 낮은 PDA에서 잘 적용될 수 있다 [7,8]. 식 (16)은 카툰 셰이딩에 사용된 식으로 정점에 대한 분산 셰이딩 방정식을 수정한 것이다.

$$C_i = a_g a_m + a_l a_m + (\max\{L \cdot n, 0\}) d_l d_m \quad (16)$$

여기서 C_i 는 정점의 색이다. a_g 는 주변광의 상관계수이고,

a_i 과 d_i 은 광원에 대한 주변광과 분산광의 상관계수를 나타낸다. 그리고 a_m 과 d_m 은 물체의 재질에 대한 주변광과 분산광의 상관계수이다. L 은 정점의 광원 벡터이며, n 은 정점의 법선 벡터이다. 실시간에는 정점간의 색을 계산하는 대신에 미리 계산된 1차원 텍스처 맵을 사용한다.

3차원 표면 모델을 이루는 각 폴리곤 마다 조명 벡터와 법선 벡터의 각도에 따라 식 (16) 과 같이 1차원 텍스처 맵을 적용한다. 여기서 일반적으로 Color 1은 그림자와 같이 어두운 색상을 Color 3은 하이라이트를 표현하는 밝은 색상을 사용한다. L 과 n 의 내적 연산 값을 이용하여 그림자 색, 기본 색, 하이라이트 색을 참조하는 과정은 (그림 7)과 같다.

$$\begin{aligned}
 &\text{If } 0 \leq L \cdot N < \alpha \text{ then Color 1} \\
 &\text{If } \alpha \leq L \cdot N < \beta \text{ then Color 2} \\
 &\text{If } \beta \leq L \cdot N \leq 1 \text{ then Color 3}
 \end{aligned} \tag{16}$$

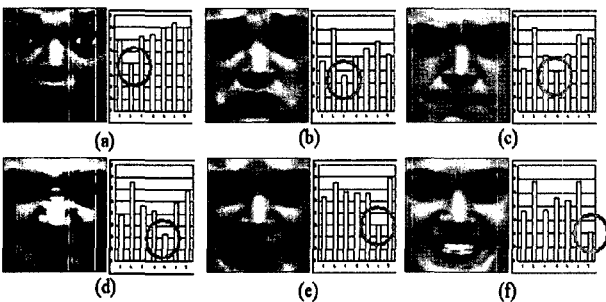


(그림 7) 1차원 텍스처 맵

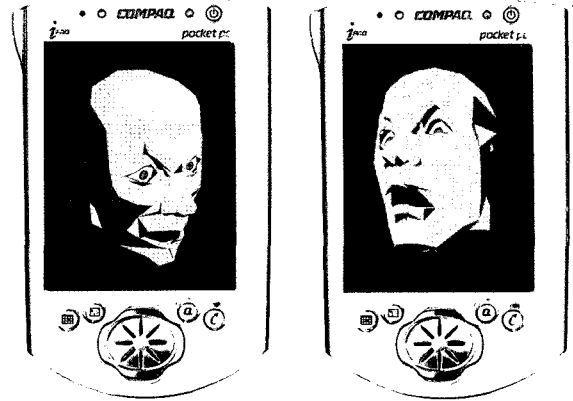
5. 구현 및 결과

본 논문의 실험을 위한 입력 비디오 영상은 Cohn-Kanade [9] 데이터베이스를 사용하였다. (그림 8)은 기본 감정에 대한 표정 영상과 3장에서 기술한 자동 표정인식 시스템으로 인식한 결과를 후측에 그래프로 보여주고 있다. 영상 옆의 막대 그래프는 7 가지 기본 표정((bar 1) neutral, (bar 2) surprise, (bar 3) fear, (bar 4) sadness, (bar 5) anger, (bar 6) disgust, (bar 7) happiness) 과의 유사도를 나타내며, 높이가 가장 낮은 막대가 현재의 표정으로 판단되어 진다. (그림 8) (a)의 놀라는 표정이 입력되는 경우 경우 놀람을 가르키는 두 번째 막대 (bar 2)가 가장 수치가 낮게 나왔다.

클라이언트 측의 PDA로는 HP사의 iPAQ 3630(200Mhz, 32Mbytes)를 사용하였고, Embedded Visual C++ 3.0과 PocketGL™을 이용하여 구현하였다. PDA의 계산 성능을 고려하여 본



(그림 8) 자동 표정인식 시스템의 인식 결과



(그림 9) PDA 상에서의 3차원 얼굴 합성 : 놀람

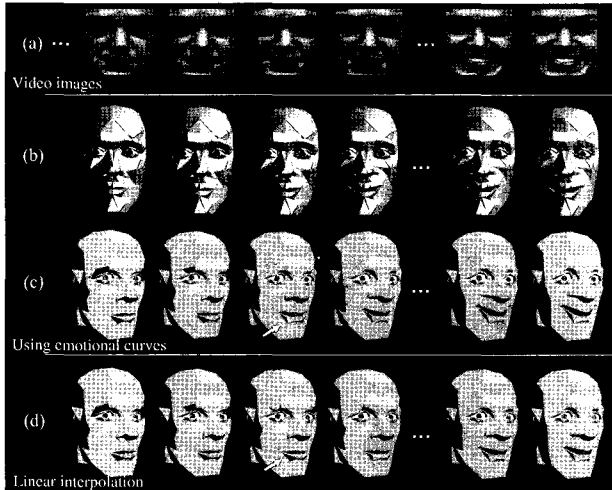
논문에서는 최소한의 정보(762 폴리곤, 476 정점)로 구성된 얼굴 메쉬를 사용하였으며, 부동 소수점 연산을 위한 최적화를 수행하였다[10]. 그리고 자동 표정인식 시스템과 PDA와의 통신은 서버 클라이언트 구조로 구축하였다. 통신은 IEEE 802.11b 무선랜 환경에서 수행되었으며, 서버측의 자동 표정인식 시스템에서 사용자의 표정을 인식하는 과정이 끝나면, 클라이언트 PDA에 소켓 통신을 통해 인식된 감정과 감정 곡선 정보를 전송하고, 클라이언트 측에서는 이러한 정보를 기반으로 비사실적인 얼굴 표정을 합성한다. 이와같이 PDA를 이용한 만화적인 아바타 (그림 9)는 사용자의 감정을 이입한 개성적인 아바타로써, 익명성을 보장하면서도 사이버 공간에서 감정을 전달하는데 유용하게 사용될 수 있다.

감정 곡선에 기반한 얼굴 애니메이션을 선형 보간법에 비교한 결과는 (그림 10), (그림 11)과 같다. 각 그림은 일련의 얼굴 비디오 영상에 따른 3차원 얼굴 애니메이션 결과를 감정 곡선 기반 방법과 선형 보간법으로 나누어 보여준다. (a)는 비디오 영상을 보여주고, (b)는 감정 곡선에 기반하여 플랫폼 웨이딩한 결과이고 (c)는 감정 곡선에 기반하여 카툰 웨이딩한 결과이다. (d)는 선형 보간법을 사용하여 카툰 웨이딩한 결과이다. (b)와 (c)의 화살표 부분을 보면 선형 보간법에 비하여 감정 곡선에 기반한 결과가 (a)의 표정 변화를 더 잘 포착하는 것을 알 수 있다. (그림 11)은 놀라는 표정에 대하여 실험한 결과로 (그림 10)과 마찬가지로 감정 곡선에 기반한 방법이 놀라는 타이밍을 더 잘 포착함을 알 수 있다. 두 실험 결과 모두 비교적 적은 수의 폴리곤 모델과 카툰 웨이딩을 적용하여 개성적인 아바타를 모바일 플랫폼 상에서 구현할 수 있었다. 현재의 PDA는 하드웨어적인 그래픽 가속기나 정확한 실수 연산이 가능하지 않기 때문에 사실적인 얼굴 애니메이션을 하기에는 아직 제약이 많이 있다. 하지만 이러한 부분은 앞으로 고성능 PDA가 등장함에 따라 해결될 수 있으리라 전망한다.

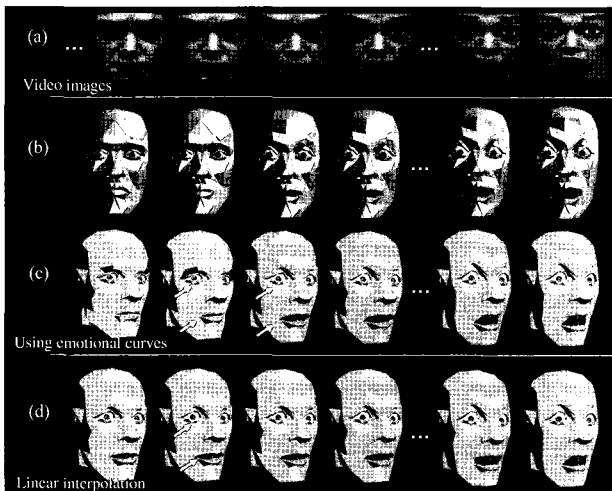
<표 3>은 현재 일반적으로 사용되는 PDA를 이용하여 얼굴 애니메이션을 수행했을 때의 초당 프레임 수를 보여준다. 부동 소수점에 대한 최적화를 수행하지 않으면 4~5 fps, 부동 소수점 최적화를 포함하면 8~9 fps, 그림 5에서와

<표 3> 부동 소수점 최적화와 얼굴 영역 분할에 따른 속도 비교

IPAQ 3630h series : (CPU : 206Mhz, ARM SA1110/RAM : 32 Memory/OS : Window CE 3.0/Model : 762polygos, 476vertices)	
얼굴 애니메이션	4~5 fps
부동 소수점 최적화 포함	8~9 fps
부동 소수점 최적화 및 얼굴 영역 분할 포함	9~10 fps



(그림 10) 감정 곡선에 기반한 애니메이션과 선형 보간법의 비교 : 기쁨



(그림 11) 감정 곡선에 기반한 애니메이션과 선형 보간법의 비교 : 놀람

같이 얼굴 영역을 나누어 계산을 수행하면 9~10 fps 의 성능을 보였다[4]. <표 3> 에서처럼 최적화를 병행했을 때 초당 3에서 4 프레임 정도의 성능 개선 효과를 볼 수 있었고, 여기서 얼굴 영역 분할을 하여 근육을 적용했을 경우 1~2 fps의 개선 효과가 나타났다.

6. 결론 및 향후 연구

본 논문에서 제시한 표정의 자동 인식에 기반한 3차원 얼굴 애니메이션은 인식과 표정합성을 단일 파이프라인으로

통합하고, 실시간으로 표정의 변화를 포착할 수 있다는 장점이 있다. 특히, 표정의 자동 인식 결과로 생성되는 감정 곡선 정보를 이용함으로써 기존의 선형 보간법에 의한 얼굴 애니메이션에 비하여 감정이 변화하는 타이밍을 더 잘 표현할 수 있었다. 또한 현재 휴대성과 편의성을 함께 갖춘 PDA 상에서 적은 수의 폴리곤 얼굴 모델을 사용하여 개성적이면서도 친근한 감성 아바타를 제공하였다. 이러한 감성 아바타는 점차 변화되어 가고 있는 유비쿼터스 컴퓨팅 환경에서 효과적인 커뮤니케이션 수단으로 사용되어질 것이다. 향후 연구로는 실시간으로 사용자의 눈동자 및 머리의 움직임을 트래킹하고 이를 얼굴 애니메이션에 통합한 다양한 형태의 감성 인터페이스를 개발하고자 한다.

참고 문헌

- [1] J. Harber, D. Terzopoulos, "Facial Modeling and Animation," SIGGRAPH 2004 Course Notes, pp.16-39, 2004.
- [2] F.I. Parke and K. Waters, "Computer Facial Animation," A K Peters, 1996.
- [3] S. H. Lee, Y. G. Kim, and G-T. Park, "Facial Expression Recognition based upon Gabor Wavelets Based Enhanced Fisher Model," LNCS 2869, pp.490-496, 2003.
- [4] P. Lalonde and R. Dawon, "A High Speed, Low Precision Square Root," Graphics Gems, Academic Press, 1998.
- [5] P. Ekman and W. Friesen, "Unmasking the Face. A guide to recognizing emotions from facial clues," Palo Alto. Consulting Psychologists Press, 1975.
- [6] I. Pandzic, R. Forchhemier, "MPEG-4 facial animation - the standard, implementations, and applications," John Wiley & Sons, 2002.
- [7] I. Adam, C. Marshall, M. Harris, and Blackstein, "Stylized Rendering Techniques for Scalable Real-Time 3D ACM Animation," In Symposium of Non-Photorealistic Animation and Rendering(NPAR) 2000, pp.13-20, 2000.
- [8] I. Buck, A. Finkelstein, C. Jacobs, et al., "Performance-driven hand-drawn animation," In Symposium of Non-Photorealistic Animation and Rendering (NPAR) 2000, pp.101-108, 2000.
- [9] T. Kanade, J. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis," Proc. Int'l Conf. Face and Gesture Recognition, pp.46-53, 2000.
- [10] J. Y. Noh, U. Neumann, "Survey of Facial Animation," Tech Report, University of Southern California, 1998.
- [11] W. Zhao, R. Chellappa, P. J. Phillips, A. Rosenfeld, "Face Recognition : A Literature Survey," ACM Computing Surveys, 2003.
- [12] 최재진, "MAYA Character Animation_2", 영진닷컴, pp.282-303, 2002.
- [13] T.D. Bui, D. Heylen, M. Poel, and A. Nijholt, "Exporting Vector Muscles for Facial Ani-mation," Smart Graphics 2003, LNCS 2733, pp.251-260, 2003.
- [14] V. Vapnik. "Statistical Learning Theory," John Wiley & Sons, 1998.

[15] K. Waters, "A muscle model for animating three-dimensional facial expressions," SIGGRAPH'87, Vol.21, pp.17-24, 1987.



이 돈 수

e-mail : dslee28@hampex.com
2002년 세종대학교 전산과학과(학사)
2004년 세종대학교 컴퓨터공학과(석사)
2005년 햄팩스㈜ 분당연구소 멀티미디어 개발본부 주임연구원
관심분야: 컴퓨터 그래픽스, 얼굴 애니메이션, DMB 등



최 수 미

e-mail : smchoi@sejong.ac.kr
1993년 이화여자대학교 전자계산학과 (학사)
1995년 이화여자대학교 전자계산학과 (석사)
2001년 이화여자대학교 컴퓨터학과(박사)
1998년 6월~1998년 12월 Fraunhofer Institute for Computer Graphics, Germany, 방문연구원
2001년~2002년 이화여자대학교 정보통신연구소 연구전임강사
2002년~2004년 세종대학교 컴퓨터공학부 전임강사
2004년 3월~현재 세종대학교 컴퓨터공학부 조교수
관심분야: 유비쿼터스 휴먼 인터페이스, 3차원 의료영상, 가상/증강 현실 등



김 해 광

e-mail : hkim@sejong.ac.kr
1986년 한양대학교(학사)
1994년 Paul-Sabatier University 전산학 (D.E.A)
1997년 Paul-Sabatier University 전산학 (박사)
1985년~1991년 삼성전자 종합연구소 주임연구원
1997년~2000년 현대전자 정보통신연구소 책임연구원
2000년~2004년 세종대학교 컴퓨터공학부 조교수
2004년~현재 세종대학교 컴퓨터공학부 부교수
관심분야: 멀티미디어 등



김 용 국

e-mail : ykim@sejong.ac.kr
1982년 고려대학교 전기공학(학사)
1984년 고려대학교 전기공학(석사)
1997년 University of Cambridge(박사)
1984년~1986년 LG전자 중앙연구소 연구원
1986년~1989년 KT 연구소 전임연구원
1995년~1997년 네덜란드 Helmholtz Robot Institute, EU Research Fellow
1998년~2001년 미국Smith-Kettlewell Vision Institute, Research Associate
2001년~현재 세종대학교 컴퓨터공학부 부교수
관심분야: Human-Computer/Robot Interaction, 얼굴 및 표정인식 Machine Learning