

HMM을 이용한 얼굴에서 입 특징점 검출에 관한 연구

김희철*, 정찬주*, 박종서*, 김문환*, 배철수**, 나상동*
*조선대학교 컴퓨터공학부
**관동대학교 정보 통신공학과
e-mail:sdna@mail.chosun.ac.kr

A Study on Mouth Features Detection in Face using HMM

Hea-Chel Kim *, Chan-Ju Jung*, Jong-Se Kwag*, Mun-Hwan
Kim*, Chul-Soo Bae**, Snag-Dong Ra*
*Dept of Computer Engineering, Chosun University
**Dept of Electronic Com. Kwangdong University

Abstract

The human faces do not have distinct features unlike other general objects. In general the features of eyes, nose and mouth which are first recognized when human being see the face are defined. These features have different characteristics depending on different human face.

In this paper, We propose a face recognition algorithm using the hidden Markov model(HMM). In the preprocessing stage, we find edges of a face using the locally adaptive threshold scheme and extract features based on generic knowledge of a face, then construct a database with extracted features. In training stage, we generate HMM parameters for each person by using the forward-backward algorithm. In the recognition stage, we apply probability values calculated by the HMM to input data. Then the input face is recognized by the euclidean distance of face feature vector and the cross-correlation between the input image and the database image. Computer simulation shows that the proposed HMM algorithm gives higher recognition rate compared with conventional face recognition algorithms.

1. 서 론

얼굴 검출, 인식 시스템은 자동현금인출기 혹은 제한 영역의 출입제한, 특별구역(은행, 가게 등)의 출입허가, 특정 DB의 관리 목적으로 컴퓨터 등에 사용자 허가를 위해 사용되어진다.

본 논문에서는 정면 얼굴 이미지 상에서 인식에 있어서 중요한 영역들 중 하나인 입 부분 추출에 초점을 맞추어 HMM 방식을 이용하여 얼굴에서 입을 인식하고 검출하는 시스템을 연구한다. 입력 얼굴 영상으로부터 입 영역을 검출하고, 검출된 영역에서 특징 점과 벡터를 추출하여 입력영상과 데이터베이스의 영상을 비교하여 입력영상의 명암, 크기 그리고 회전 같은 약간의 변화에 대해서도 동일한 결과를 유추해내어 일관된 얼굴 인식의 기반을 제공한다. 전통적인 평면 기반의 방법과 비교하여 HMM 기반의 방법으로 검출과 인식 작업의 결과를 나타내고, 또 여러 시스템에서 능률적으로 사용이 가능하도록 실험의 결과를 통해 입증한다.

2. HMM의 확률적 모형과 기능 모델

2.1. HMM의 개요

인식을 위한 HMM은 다양한 형태로 구성될 수 있으나, 본 논문에서는 얼굴의 각 인식단위로 5개의 상태를 가지는 left-to-right 모델을 사용한다. 각 단위의 HMM이 연결된 형태이며 그림 1과 같이 나타내었다.

2.2. HMM의 학습

인식을 위한 학습하기 위해, 즉 각각의 얼굴 구성 요소에 해당 하는 HMM들의 파라미터 값들을 추정하기 위해서는 요소들의 학습 데이터 집합이 필요하다. 학습 데이터 집합내의 구성하는 단위 요소들을 학습시키기 위해서는 모든 단위 요소를 나열로 나타내야 한다. 본 논문에서는 계산 량이 적고 비교적 인식률이 높은 segmental K-means 알고리즘[12] 을 사용하였다. 이 알고리즘은 영상으로부터 입력을 받아 자동적으로 분할하여 학습하며, 이러한 과정을 반복하여 HMM 파라미터를 추정하기 때문에 그 과정은 다음과 같다. 첫째, 초기 분할 블록에서는 분할 정보가 주어지지 않은 경우 얼굴을 구성하는 관측 열을 HMM 상태 별로 균일하게 분할한다. 여기서는 각 구성요소마다 하나의 학습 데이터베이스를 가정한다. 둘째, HMM 파라미터 추정 블럭은

HMM 파라미터인 출력 확률분포와 전이 확률을 구하는 과정이다. 셋째, 재분할 블록에서는 발음 사전에 따라 구성요소의 서브 단위 HMM을 연결하여 벡터비 복호화 과정을 통해 각 학습 데이터를 재분할한다.

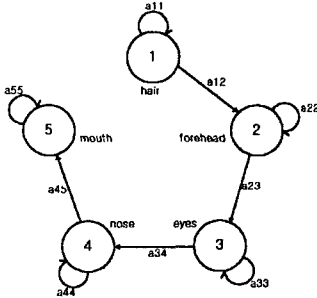


그림 1. 얼굴 인식을 위한 HMM

3. 특징점과 특징 벡터

3.1. 특징점 추출

특징점이란 영상 안의 물체가 가지고 있는 코너 점(corner point)이나 주변화소에 비해 큰 밝기의 변화를 갖는 화소와 같은 점으로서, 이 점들은 영상 안에서 다른 점들과 구별되어지는 점이다. 특징 점들은 스테레오비전, 3차원 인식, 움직임 추정과 같은 컴퓨터비전 분야에서 널리 사용되는데, 특히, 영상 인식이나 카메라 캘리브레이션등의 분야에서 이들의 역할은 중요하다. 영상 안에서 특징 점을 추출하는 방법은 다음과 같이 크게 세 가지로 분류할 수 있다.

3.1.1 에지 지도(edge map)를 이용하는 방법

이 방법은 먼저 영상으로부터 높은 미분 값을 갖는 에지를 뽑아내어 이들을 체인코드(chain code)로 표현한다. 그런 다음 이 체인코드 안에서 지역적 최대 곡률(local maximum curvature)을 갖는 점들을 특징 점으로 삼거나[14], 혹은 이 체인코드에 다각형 근사(polygonal approximation)를 수행하여 각 라인 세그먼트를 얻어낸 다음 그 라인 세그먼트들의 교차점을 특징 점으로 삼는다.

3.1.2 그레이레벨(gray level)을 직접 다루는 방법

최근에 제안된 대부분의 특징 점 추출기들은 이 분류에 속한다. 특징점들을 뽑아내기 위해 그레이레벨의 영상을 직접 다룬다. 즉, 각 화소에 대한 특징점 정도(feature strength 혹은 featureness)를 얻어내기 위해 그레이레벨 값들로부터 정의되는 지역적 측정치를 사용한다.

3.1.3 모델 기반의 접근방법

이 방법들은 특징 점들을 추출하기 위해 특정 모델을 사용한다. 즉, 영상 안의 특징 점들이 특정 모델을 갖는다고 가정하고 이 모델에 부합되는 점들을 찾아내어 그 점을 특징 점으로 판단한다. 이 기법을 사용하면 서브픽셀(subpixel)단위의 정확도를 갖는 특징 점 지역화가 가능하다.

3.2 특징점 추출기

영상 안에서 한 점 $p(x, y)$ 와 그 점을 중심으로 하는 주변 윈도우 W 을 고려해본다. W 안에서 이 다른 한 점 $q(x, y)$ 라 정하고, 또 $I(p)$, $I(q)$ 가 각각 점 p, q 에서의 밝기 값을 나타낸다고 하자. 그러면 p 와 q 사이의 밝기 값 유사도 $S(p, q)$ 는 식(1)과 같이 정의된다.

$$S(p, q) = \begin{cases} 1 & \text{if } |I(p) - I(q)| < t \dots (1) \\ 0 & \text{otherwise} \end{cases}$$

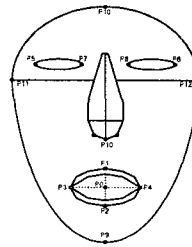
여기에서 t 는 밝기 값의 유사정도를 나타내는 밝기 임계값(brightness threshold)이다. 이 식으로도 만족할 만한 결과를 얻을 수 있지만, 좀더 안정적인 유사도 계산식은 식(2)과 같다.

$$S(p, q) = e^{-\frac{|I(p) - I(q)|}{t}} \dots (2)$$

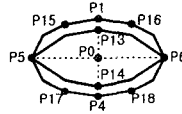
이제 점 p 에서의 "USAN(Univalue Segment Assimilating Nucleus)"은 식(3)과 같이 계산된다.

$$N(p) = \sum_{q \in \mathcal{N}(p)} S(p, q) \dots (3)$$

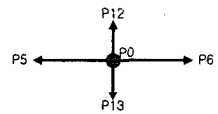
이 값이 주변 윈도우 면적의 반보다 작으면 점 p 는 특징 점으로 판단하고, 또 이 값이 작을수록(SUSAN : smallest USAN) 특징 점의 정도가 커지게 된다.



a) 얼굴의 기본 특징 점



b) 입의 특징 점



c) 입의 특징 벡터

그림 2. 특징 점

얼굴의 인식이 필요한 기본적인 특징 점은 그림 2와 같다. 먼저 점 P5와 P6를 잇는 선은 얼굴의 가로축으로 얼굴의 기울어짐과 두 눈의 위치판단에 사용된다. 점 P11과 P12는 얼굴의 폭을 점P9와 점P10은 얼굴의 길이를 각각 나타낸다. 두 눈 사이의 거리는 P7과 P8, 입의 크기는 P3와 P4 그리고 입과 코와의 거리는 P1과 P10으로 각각 표현된다.

본 논문에서 초점을 맞춘 입의 특징은 특징 벡터로써 표현이 가능한데, 이 특징들은 입의 크기, 두께, 그리고 벌어짐이나 무무린 정도 등을 알 수 있다.

4. 얼굴 특징 점 추출 과정

4.1. 입력 영상 보정

크기에 대한 보정은 데이터베이스 작성 시 입력된 얼굴의 영상과 인식 시 입력되는 영상의 크기가 다름으로 해서 가져오는 오차를 극복하기 위한 방법으로 두 눈 사이의 거리를 미리 데이터베이스화하는 방법을 이용하였다. 밝기에 대한 보정은 두 가지로 나뉘어지는데, 하나는 그 평균값을 조정하는 것이고, 다른 하나는 밝기의 변화율을 보정하는 것이다. 밝기 변화율은 조명명이 밝을 때 대체로 커지고 어두울 때 작아지므로 밝기 평균에 따라 그 변화율에 가중치를 두는 방법을 취하였다.

$$difference = \left| \frac{image - image\ mean}{image\ mean} - \frac{template - template\ mean}{template\ mean} \right| \dots (4)$$

입력 영상과 데이터베이스의 영상들을 비교하여 그 차이 (*difference*)가 가장 적게 나는 것을 선택한다.

본 논문에서는 기존의 템플릿 매칭방법에서와 같이 얼굴의 기울어짐 정도를 눈의 위치로부터 계산하였다. 즉, 두 눈의 위치를 찾은 후, 두 눈 사이를 잇는 선분과 수평선상이 이루는 각을 계산하고 그 각도만큼 회전시키는 방법을 사용하였다. 이를 위해서 반드시 필요한 것은 여러 각도에 대한 눈의 모델이 필요하게 된다. 이는 기울어진 영상에서 회전에 대한 모델 없이 눈을 찾으려면 올바른 검출이 이루어지지 않기 때문에 각 기울어진 각도에 대한 모델을 설정하여 눈의 위치를 찾는다.

4.2. 얼굴 영역의 분리

얼굴 영상은 순수한 얼굴 영역 영상만을 사용하기 때문에 일반적인 입력영상에서 순수한 입 영상을 얻기 위해서는 얼굴 영역 추출 알고리즘을 적용한다. 입력 영상에 있어서 복잡한 배경은 배제시킨다는 가정 하에 Kelly의 방법을 변형하여 적용하였다. 이때, 일반적인 사람의 얼굴이 타원형에 근사하다는 점에서 이지 영상을 타원 모델과 비교한다. 타원과 실제 얼굴의 값, 그리고 사람마다의 형태 차이를 감안하여 유사도가 0.7 이상의 객체 중 가장 큰 객체를 얼굴이라 가정한다.

4.3. 전처리 과정

영상에 대한 기본적인 보 정을 마친 다음 실질적인 검출 작업이 수행된다. 본 논문 얼굴인식을 위한 지역적 특성으로 입을 사용한다. 이것은 정면 얼굴영상에서 조명이나 얼굴 영상의 크기와 무관하게 그 모양이 유지되는 특징구성을 갖고 있고, 그 특징에 대한 활용도가 높기 때문에 얼굴인식에 있어서 그 비중이 크다 하겠다. 여기서 제안한 입을 인식하기 위해 얼굴 내에서 입의 특징 점을 보이고 특징 벡터를 사용한다.

그림 3은 전처리 과정의 영상들로서, 그림 3(b) 결과에서 볼 수 있듯이 얼굴 구성 객체가 강조되어 나타났다. 하지만, 이러한 특징 외에도 얼굴의 윤곽선이나 기타 작은 객체들이 같이 나타난다. 그러나 그림 3(d)와 같이 전처리 과정의 최종단계 후에는 입이 강조되고 다른 객체들은 거의 사라진 것을 확인할 수 있다. 하지만 아직 입에 비해 작은 객체와 오른쪽, 왼쪽으로 45° 정도 기울어진 뺨에 해당하는 객체들이 남아있는 것을 확인할 수 있다.



(a) 얼굴 입력 영상 (b) 에지 추출 결과



(c) skeleton 영상 (d) 이진 화 모폴로지 영상

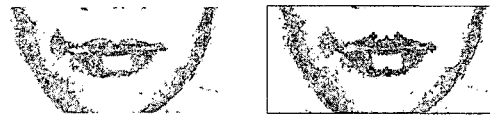
그림 3. 입 특징 점 추출을 위한 전처리 과정

4.4. 얼굴 영상에서 입 추출

얼굴 영상에서 각 객체의 위치를 찾기 위해 각 특징 점의 기하학적인 위치 관계를 이용하고 있다. 전처리 과정의 결과 영상 중 후보영역 안에 존재하는 객체들은 눈이나 입뿐만 아니라 의미 없는 데이터가 함께 섞여있다. 본 논문에서는 기준 점의 위치와 특징점이 있을 만한 위치를 후보영역으로 설정한다. 입의 기준 점은 입력영상의 사람 얼굴이라는 가정에서 특징들이 있을 만한 위치를 기준으로 설정한다. 영역의 중심점을 입의 기준 점으로 설정하고 각 영역에 들어가는 가장 큰 원을 기준 점으로부터 각 특징이 존재할 수 있는 후보객체 거리 임계 치로 설정한다. 따라서 거리 임계 치를 설정한 원의 반지름은 영상 넓이 25% 만큼의 크기이기 때문에 추출범위가 확정된다. 입 추출 알고리즘은 전처리 과정의 결과 영상에서 입의 경우 입술의 가운데 부분에서 끊어지는 현상이 발생한다. 따라서 전처리 결과의 영상으로부터 입을 찾을 경우 원래의 입보다 작은 반쪽 짜리 입을 찾는 경우가 많다. 입 추출 알고리즘의 복잡 도는 후보 객체의 개수에 비례해서 $O(n^2)$ 으로 결정한다.



a) 입 영역 분리



b) 입 영역 윤곽선 추출 c) 입 영역 특징 점 추출
그림 4. 입 특징 점 추출 과정

5. 실험 및 결과

인간이 얼굴을 구분하는데 사용할 수 있는 요소는 각 특징을 이루는 눈, 코, 입의 생김새뿐만 아니라 전체적인 배치도 중요하다. 이러한 점을 인식의 단계에 적용하기 위해 그림 10과 같이 얻어진 입 특징을 여러 개의 벡터로 표현한다. 이를 위하여 얻어진 후보 얼굴 중 정확한 얼굴을 찾아내기 위해 얼굴 특징 점만을 포함하는 영상을 얻어낸다. 이것은 얼굴 입력 영상 중 특징 점만을 포함하는 영역을 잘라낸 특징 점 영상이다. 영상의 +기호는 특징 점 추출 알고리즘에 의해서 얻어진 특징 점의 위치를 나타낸다. 여기서 상관관계를 계산하기 위해 얻어진 얼굴 특징 영상은 동일한 크기로 정규화 한다.

MI_i 는 영상자체의 유사도 이다. 이는 입력 얼굴 영상과 데이터베이스에 존재하는 i 번째 영상과의 상관관계로 정의되며 식 (9)과 같이 나타낼 수 있다.

$$MI_i = \frac{E(IT_i) - E(I)E(T_i)}{\sqrt{sd(I) \times sd(T_i)}} \dots (9)$$

I : 입력 영상 데이터

T_i : 영상 데이터베이스의 i 번째 영상
 $E()$: 평균
 $sd()$: 분산

여기서 $E(IT_i)$ 는 입력 영상과 데이터베이스 영상의 각 픽셀의 곱 영상의 전체 평균값이다. 이때 평균과 분산은 각 영상의 전체 픽셀의 평균과 분산이다.

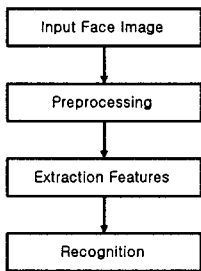


그림 5. 영상 처리 과정

영상 자체의 유사 도를 구하기 위해 얼굴 특징 점 영상을 특정 크기로 정규화해야 한다. 본 논문에서 사용한 정규화 크기는 40*40 픽셀 상관관계를 사용한다. 이는 70명의 서로 다른 사람 특징 점 얼굴의 입 영상 픽셀 상관관계를 실험해서 40*40보다 작아지게 되면, 상관관계는 0.4 이상의 값이 된다.

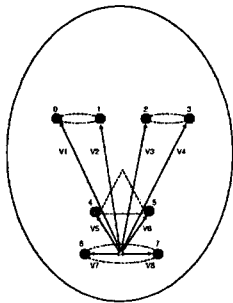


그림 6. 얼굴 특징 점 벡터

제안한 얼굴 인식 방법을 실험하기 위하여 16사람으로부터 한 사람 당 11개씩 모두 176개의 정면 얼굴 입 영상을 사용한다. 얼굴의 입 영상 입력 방법은 디지털카메라 2대를 이용해서 특수한 환경을 만들지 않은 평범한 형광등이나 보조광학 장치의 불빛아래에서 입력받는다. 먼저 입력 영상에서 입 부분만을 따로 떼어내고, 다음으로 분리된 영상에서 윤곽선을 추출해내고, 마지막으로 특징 점을 검출해내는 과정을 거친다. 여기서 특징점을 찾는 과정에 HMM을 이용하여 보다 효율적인 특징 점 추출이 가능하다. 이러한 과정을 통해 수행한 176개의 결과 얼굴 영상에 대해 실제 입의 위치를 정확히 찾아낸 경우는 97%이상이다. 그림에서 확인할 수 있듯이 얼굴 영상의 밝기, 크기에 무관하게 입의 양 끝점을 찾아내고 있음을 볼 수 있다. 또 얼굴의 선택 위치는 얼굴을 포함한 입 영상이면 만족하는 결과를 나타낸 것이다.

그림 7은 실험결과를 나타낸 그래프이다. 하나의 실선은 한 입력 영상에 대한 16개 데이터 베이스의 영상과 유사도 측정결과를 나타낸다. 그래프의 세로 축은 유사도가 높고, 가로축은 다른 사람 번호가 된다. 유사도가 0으로 나타난 부분은 후보영상에서 제외되어 0으로 설정된 값이다. 그림 7은 3번 사람의 유사도 측정 결과를 나타내는 것으로 모든 실선이 3번 사람 위치에서 최

대치가 된다. 인식결과 표정의 변화에 의해 입 특징점이 위치가 변화되어, 후보 입 영상을 얻지 못한 경우 에러 인식율이 커진다.

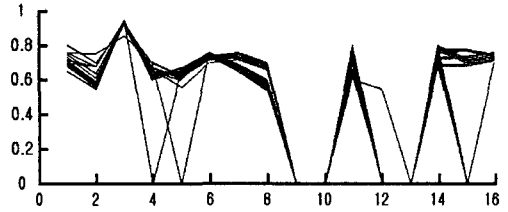


그림 7. 입력 영상과 데이터베이스의 유사도 측정결과

6. 결 론

본 논문에서는 얼굴 형태를 분석하여 기울어진 영상이나 크기에 무관하게 입의 위치를 검출하는 방법을 제안하였다. 제안한 알고리즘은 입의 얼굴 영상 입력에서 입의 특징을 HMM을 사용하여 빠르게 찾을 수 있다. 얼굴에 존재하는 객체들이 위치할 후보영역을 지정하여 그 영역 안에서 찾고자하는 특징 점 객체를 선택하도록 하였다. 그러나, 영상이 흐리거나 불분명할 경우에는 각 특징 점 선정에 문제점이 발생한다. 향후 연구해야 할 내용은 특징 점과 특징 벡터를 이용하여 다양한 움직임을 갖는 입 영상을 인식하는 연구와 인식된 입 영상을 확장하여 동영상에서의 변화하는 입 모양을 통해 발음하는 음성을 유추해내는 연구와 얼굴의 특징 점만으로 얼굴을 재구성하기 위해서는 더욱 많은 특징 점의 자동 추출이 필요하고, 심하게 회전된 얼굴과 안경을 쓴 얼굴 영상 등 변형된 영상에 대해서도 인식을 가능하게 하는 연구가 필요하리라 본다.

[참고문헌]

- [1] Cipolla R., Pentland A. Editors, "Computer Vision for Human-Machine Interaction," Cambridge Univ. Press, Cambridge, 1998.
- [2] Mase K., Watanabe Y., Yasuhito Y., "Headreader:Real-time Motion Detection of Human Head from Image Sequences," Systems and Computers in Japan, Vol. 23, No. 7, 1992.
- [3] R. Chellappa et. al, "Human and Machine Recognition of Faces : A Survey," Proceedings of the IEEE, 1995
- [4] Xiaobo Li and Nicholas Roeder, "Face contour extraction from Front-view images," Pattern recognition, Vol. 28, No. 8, pp.1167-1179, 1995.
- [5] R. Rose, D. Paul, "A Hidden Markov Model Based Keyword Recognition System," in Proc. Int. Conf. Acoustics, Speech, and Signal Processing, 1990, Vol. 1, pp.129-132
- [6] L. bahl, P. Brown, P. Souza, and R. Mercer, "Maximum mutual information estimation of hidden Markov model parameters for recognition," in Proc. Int. Conf. Acoustics, Speech and Processing, pp. 49-52, apr. 1986.
- [7] L. Rabiner, "A tutorial on Hidden Markov Models and selected Applications in speech Recognition," Proc. IEEE 77, pp. 257-286, 1989
- [8] G. D. Forney, "The Viterbi Algorithm," Proc. of IEEE, Vol. 32, No. 2, pp263-271, Apr. 1989.