

논문-98-3-2-07

## 얼굴 특징 벡터를 이용한 효율적인 얼굴 인식 알고리즘

전승철\*, 박성헌\*

### An Efficient Algorithm of Face Recognition Using Facial Feature Vectors

Seung-chul Jun\* and Sung-Han Park\*

#### 요 약

사람의 얼굴은 일반 객체와는 다르게 정확히 구별되는 특징이 없다. 따라서 일반적으로 사람 얼굴에 관한 연구에서는 인간이 사람의 얼굴을 볼 때 가장 먼저 인식을 하는 눈, 코, 입을 특징으로 정하고 있다. 이러한 특징은 사람에 따라 다르게 나타나며 주위환경에 영향을 받는다. 따라서 이러한 사람의 특징을 정확히 찾아내는 것이 중요하다. 본 논문에서는 얼굴 특징점의 기하학적 성질을 이용하여 눈, 코, 입의 특징점을 효율적으로 찾아내는 알고리즘을 제안하고 있다. 이러한 특징점을 이용해서 얼굴 특징점 벡터와 얼굴 특징점 영상을 얻어낸다. 이 후 임의 입력 사람 얼굴에 대하여 얼굴 특징점 벡터의 유클리디안 거리와 미리 기록된 특징점 영상과의 상관관계를 이용해 유사도를 계산해서 얼굴을 인식한다. 제안하는 방법은 기존의 방법보다 계산 복잡도가 적으며 또한 정확한 인식을 얻는다.

#### 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. Therefore, the exact detection of human features are very important. In this paper we propose an efficient algorithm for finding the features of eyes, nose and mouth using the geometrical characteristics of a human face. The facial feature vector and the facial feature image are made with these features. 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. The proposed algorithm has less complexity and higher recognition rate than the existing methods.

#### I. 서 론

사람의 얼굴 인식은 얼굴 영상 입력을 통한 인증과 같은 보안이나 많은 얼굴 중에서 비슷한 사람의 검색과 같은 곳에 활용되고 있으며 또한 향후 로봇 공학에서 로봇이 임의 사람을 알아보는 부분에서 사용될 수 있다. 이러

한 필요성으로 얼굴인식 기술은 그 가치가 더해가고 있어 많은 연구가 수행되었다.<sup>[1][8]</sup> 하지만 사람 얼굴에는 일관성이나 보편성이 존재하지 않아서 그 연구진행에 많은 어려움이 있다. 즉, 같은 사람의 얼굴이라도 그 사람의 표정이 다를 때마다 그 특징은 다르게 나타나며 시간에 따라 다른 모습을 가지고 있다. 따라서 얼굴을 인식하기 위해서 사용되는 얼굴 특징은 모든 사람이 일반적으로 가지고 있고, 사람을 구분하는데 중요한 비중으로 사용되는 값이어야 한다. 이러한 관점에서 얼굴에 존재하는 눈, 코, 입의 위치는 가장 중요한 특징점으로 볼 수 있다.

이러한 특징점을 얻기 위해 템플릿을 이용하는 방법<sup>[1]</sup>이 있다. 템플릿 매칭을 위해서는 미리 만들어진 객체의

\* 한양대학교 전자계산학과  
Dept. of Computer Science & Engineering, Hanyang University

템플릿을 입력 영상의 모든 위치에 대해서 매칭을 시도하게 된다. 이때 객체의 크기를 알 수 없기 때문에 미리 지정된 가로 세로 크기만큼 템플릿을 변화하면서 매칭을 시도한다. 때문에 하나의 템플릿을 매칭하기 위해서는  $O(n^4)$ 의 복잡도를 가진다. 템플릿 매칭의 방법은 정확한 모양을 찾아내는 반면, 높은 복잡도를 가짐으로서 처리 시간이 오래 걸리는 단점이 있다. 특징점을 얻는 다른 방법으로 허프 변환을 이용하는 방법<sup>[2]</sup> 등이 연구되었다. 허프 변환을 이용해서 특징점을 찾기 위해서는 특징점을 이루는 객체를 하나의 도형으로 간주한 후 전체 영상을 조사해야 한다. Li<sup>[1]</sup>는 눈을 타원과 원으로 구성된 도형으로 간주하고 입을 하나의 타원으로 이루어진 것으로 간주하고 있다. 이후 각 도형에 대해 허프 변환을 적용한다. 이때 전체 영상에 적용하기보다는 파라미터로 주어진 후보 영역에 대해서만 눈과 입의 위치를 찾아내고 있다. 허프 변환으로 주어진 도형을 찾기 위해서는 하나의 수식에 대해서 영상에 존재하는 모든 점에 대해서 가능한 도형의 형태로 계산을 수행하기 때문에  $O(n^3)$ 의 복잡도를 가지고 있다. 뿐만 아니라, 주어진 도형의 크기를 모를 경우 이것 또한 하나의 차원을 차지하기 때문에 더 많은 복잡도를 가질 수도 있다. 또한 허프 변환의 경우, 입력 영상의 이진화 과정을 어떻게 하느냐에 따라서 특징점 추출의 결과의 정확도가 많은 영향을 받게 된다. 즉, 얼굴 영상 안에 존재하는 특징점 주변에 존재하는 작은 객체들에 의해서 눈, 입 등이 잘못 얻어지는 경우가 발생한다. 동영상 이용해서 얼굴의 특징을 찾는 방법<sup>[3]</sup>에서는 입력되는 연속적인 얼굴 영상 중에서 얼굴 영역을 얻어내고 특징점들간의 상호 관계를 정의해서 그 상호 관계가 얼굴 특징과 가장 유사한 후보 영역들을 얼굴 특징점으로 찾아주고 있다. 동영상에서는 얼굴이 움직이지 않더라도 배경에 비해 미세한 움직임을 가지고 있다. 이에 착안하여 얼굴의 전체적인 윤곽을 찾아내고 이후 정의된 얼굴의 상호 관계는 몇 개의 파라미터를 가지고 에너지를 계산한다. 하지만 이러한 파라미터 설정에 따라 얼굴 옆에 존재하는 작은 데이터들이 얼굴 특징점으로 오인될 수 있는 가능성이 존재한다. 본 논문은 이러한 문제점을 해결하고 입의 상황에서 입력으로 들어오는 여러 사람의 얼굴에서 눈, 코, 입을 빠른 속도로 안정적으로 찾아내는 알고리즘을 제안한다. 또한 여기서 구한 특징으로부터 얼굴인식을 위하여 특징점들의 전체적인 배치를 보여주는 특징점벡터를 구하는 방법을 제안한다.

본 논문은 2장에서 얼굴의 기하학적인 특징을 이용하여 기준점을 설정한 후 이 기준점과 얼굴에 나타나는 작은 객체들의 유클리디안 거리를 계산함으로써 눈, 코, 입을 찾아내는 알고리즘에 대해서 설명한다. 또한 각 알고리즘의 복잡도를 계산해 봄으로써 다른 알고리즘과 성능을 평가한다. 3장에서는 찾아진 특징을 벡터로 표시하여 얼굴을 인식하는 알고리즘에 대해서 설명하고, 마지막으로 4장에서 실제 얼굴에 적용한 실험 결과를 보여주고 있다.

## II. 얼굴 특징점 추출

본 논문에서는 얼굴 인식을 위한 지역적 특성으로 눈, 코, 입을 사용한다. 이들은 정면 얼굴영상에서 조명이나 영상 크기에 무관하게 그 모양이 유지되는 특징을 갖고 있기 때문이다. 또한 사람이 얼굴을 인식하는데 가장 먼저 바라보는 특징이다. 그림 1은 본 논문에서 제안한 눈, 코, 입을 추출하기 위해 사용하는 얼굴 내에서의 이들의 특징점을 보이고 있다. 눈의 양끝 점과 코 아래 부분의 양 끝 점, 그리고 입의 양끝 점을 얼굴의 특징점으로 정의하고 있다.

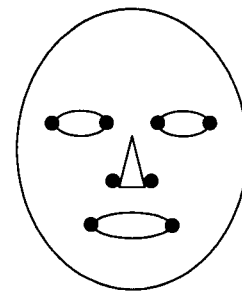


그림 1. 얼굴의 특징점  
Fig. 1. Face features

이렇게 정의된 특징을 찾기 위해서 본 논문은 각 특징별로 알고리즘을 제안하고 있다. 따라서 전체의 알고리즘의 복잡도는 특징점별로 제안된 알고리즘의 복잡도의 합으로 표현된다. 특징점 추출 알고리즘은 이진화 레이블링된 영상에서 각 객체 별로 제약조건을 만족하는 객체를 선택하도록 구성되어 있다. 그러므로 최대  $n^2$ 의 객체가 있을 수 있고 복잡도는  $O(n^2)$ 으로 표현할 수 있다.

### 1. 전처리 과정

본 논문에서 사용하는 얼굴 영상은 순수한 얼굴 영역 영상만을 사용한다. 일반적인 입력 영상에서 순수한 얼굴 영상을 얻기 위해서는 얼굴 영역 추출 알고리즘을 적용한다. 얼굴 영역을 추출하는 방법은 복잡한 배경에서 얼굴을 얻어내는 방법<sup>[2]</sup>과 피부의 색을 정보를 이용해서 얼굴 영역을 얻어내는 방법이 있을 수 있다. 본 논문에서 사용한 방법은 입력 영상에 하나의 얼굴만이 있다는 가정에 복잡한 배경영상에서 얼굴 영역만을 얻어내는 방법을 적용하였다.

다음으로 눈, 코, 입을 찾기 위해서는 먼저 얼굴에 존재하는 여러 가지 특징 중 눈, 코, 입과 유사한 객체들만 남기고 나머지는 제거해서 이진형태로 만드는 전처리 과정이 필요하다. 본 논문에서는 이러한 전처리 과정을 크게 3

개의 연산 과정으로 수행한다. 첫 번째 얼굴의 특징점을 강조한 에지 연산을 하고, 두 번째는 에지의 능선 부분을 얻어내는 skeleton 연산을 한 후 마지막으로 끊어진 특징점을 연결하는 모폴로지 연산을 한다.

눈, 코, 입 추출을 위해서 입력 얼굴 영상을 눈, 코, 입만이 강조되어있는 영상으로 만드는 과정이 필요하다. 영상에 존재하는 일반적인 객체의 테두리만을 얻어내기 위해 얼굴영상에 Sobel 알고리즘을 그대로 적용할 경우 얼굴영상에 존재하는 미세한 에지 성분까지 결과 영상에 나타나게 된다. 따라서 본 논문에서는 Soble 알고리즘을 변경하여 가로 방향의 마스크를 적용한 결과의 절대값을 결과 영상에 나타낸다. 이렇게 하면 입력 영상에 존재하는 가로 방향 에지 성분만이 검출되고, 그 중 밝은 밝기 값에서 어두운 밝기 값으로 변하는 에지만이 출력 영상이 된다.

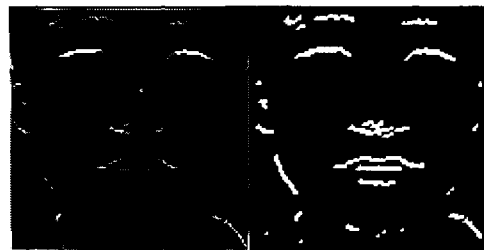
얻어진 가로 방향 에지 성분의 특징을 정확히 얻어내기 위해 에지의 skeleton만을 얻어내는 과정이 필요하다. 이때 단순히 이진화해서 thinning 알고리즘을 적용하기보다는 출력 에지의 밝기 값이 최대가 되는 능선을 따라 가면서 skeleton을 얻어냄으로써 에지의 특성을 좀더 정확히 반영시킬 수 있다. 강조된 영상은 눈썹, 눈, 코밑, 입 부분과 가로 방향 특징을 가진 객체들이 나타난다. 사람 얼굴의 특징상 세로 방향으로 구성된 특징보다는 가로 방향 특징을 가지는 객체가 더 많이 존재하므로 이때 얻어지는 특징들은 눈, 코, 입을 포함하는 영상이라고 생각할 수 있다. 그러나 눈, 코, 입은 연속적으로 같은 밝기 값으로 구성되어 있다는 보장이 없다. 즉, 연속적으로 연결된 선으로 나타나지 않기 때문에 skeleton 영상을 이진화 한 다음 모폴로지 dilation을 수행함으로써 끊어진 선들을 연결한다. 이렇게 함으로써 한번의 dilation 수행으로 미세하게 끊어진 선들을 연결함과 동시에 선을 좀 더 강조시키는 결과를 얻을 수 있다. 이때 눈, 코, 입 뿐만 아니라 다른 객체들도 같이 강조되는 결과를 얻게 되는데, 이러한 객체들은 이후 눈 추출 및 입 추출 과정에서 제거된다.

그림 2는 전처리 과정의 영상들이다. 그림 2-(b) 결과에서 볼 수 있듯이 얼굴 영상 눈, 코, 입이 강조되어 나타나 있음을 볼 수 있다. 하지만, 이러한 특징 외에도 얼굴의 윤곽선이나 기타 작은 객체들이 같이 나타나 있음도 볼 수 있다. 그러나 그림 2-(d)와 같이 전처리 과정의 최종단계 후에는 눈, 코, 입이 강조되고 다른 객체들은 거의 사라진 것을 확인 할 수 있다. 하지만, 아직 눈, 코, 입에 비해 작은 객체와 오른쪽, 왼쪽으로 45° 정도 기울어진 뺨에 해당하는 객체들이 남아 있는 것을 확인할 수 있다.

따라서 전처리 과정의 최종 얻어진 영상에서 실제 특징이라고 생각할 수 있는 눈, 코 입의 위치를 얻어내기 위해서 각 부분 별로 별도의 알고리즘을 적용하여 각각의 위치를 찾아내는 처리가 필요하다. 각 특징을 찾는 순서는 눈, 입, 그리고 눈과 입 사이에 존재하는 코를 찾는 과정으로 이루어진다. 그 각각의 방법은 다음과 같다.



(a) 얼굴입력 영상 (b) 에지 추출 결과



(c) skeleton 영상 (d) 이진화 모폴로지 영상

그림 2. 얼굴 특징점 추출을 위한 전처리 과정

(a) Input face image (b) Edge image (c) Skeleton image (d) Binary morphology image

Fig. 2. Preprocessing for extracting face features

## 2. 눈, 코, 입 추출

눈, 코 입의 위치를 찾기 위해 각 특징점의 기하학적인 위치 관계를 이용하고 있다. 그림 3은 본 논문에서 정의한 기준점의 위치와 특징점이 있을 만한 위치를 정의한 후보 영역을 정의한 그림이다. 각 특징점은 기준점으로부터 일정 거리만큼 떨어진 영역 안에 존재할 수 있다고 정의하고 눈과 입의 기하학적인 모양에 근거해서 눈을 찾아내는 알고리즘을 제안한다. 이때 사용되는 거리 값은 왼쪽 눈은 입력 영상의 중점을 기준으로 왼쪽 상단에 존재한다고 가정해서 값을 결정한다. 사람의 눈은 얼굴 중심에 대칭적인 특징을 가지고 있으므로 왼쪽 눈에 대한 알고리즘을 구성하고, 오른쪽 눈은 기준점의 위치만 변경하여 적용한다.

눈, 입의 기준점은 입력 영상이 사람 얼굴이라는 가정에서 특징들이 있을만한 위치를 기준으로 설정한다. 입력 사각형 영역을 4 등분 할 때 왼쪽 눈은 왼쪽 상단 영역에 위치하게 된다. 만약 그렇지 않다면, 영상은 사람의 얼굴이라고 가정할 수 없다. 같은 방법으로 오른쪽 눈은 오른쪽 상단 쪽에 위치하게 되고, 입은 하단 좌우 영역에 걸쳐서 나타나게 된다. 이러한 점을 기반으로 각 영역의 중심점을 눈, 입의 기준점으로 설정하고 각 영역에 들어가는 가장 큰 원을 기준점으로부터 각 특징이 존재할 수 있는

후보객체 거리 임계치로 설정한다. 따라서 거리 임계치 설정한 원의 반지름은 영상 넓이 25% 만큼의 크기가 된다.

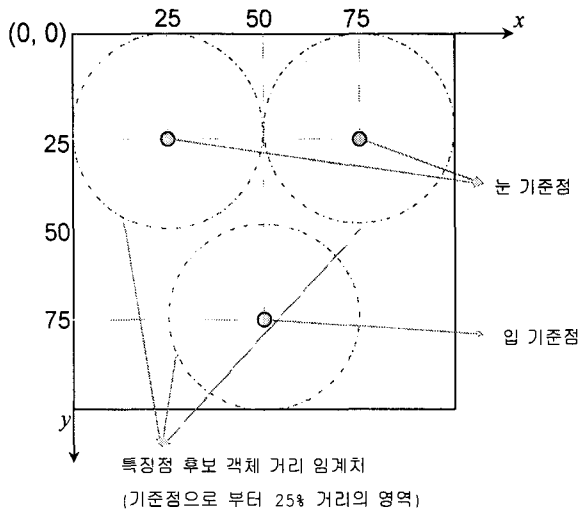


그림 3. 특징점 기준점과 후보 영역  
Fig. 3. The origins of features and candidate area

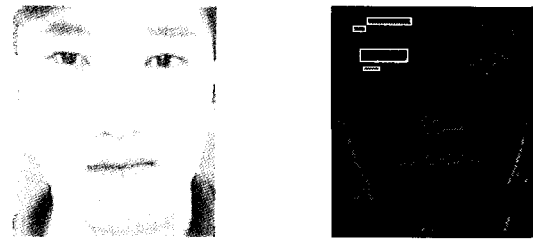
### 2.1 눈 추출 알고리즘

전처리 과정의 결과 영상 중 눈 후보영역 안에 존재하는 객체들은 눈뿐만 아니라 의미 없는 데이터가 함께 섞여있다. 눈 후보 영역에 존재하는 객체들 중에서 가로 방향 에지 특성을 가장 강하게 띠는 객체는 눈썹과 눈으로 볼 수 있다. 그러므로 전처리 과정을 거친 결과는 눈과 눈썹 객체가 가장 넓은 면적을 차지하는 객체가 된다. 또한 눈과 눈썹을 둘러싸는 사각형은 세로보다 가로가 넓은 사각형을 이루고 있다. 이러한 특징을 이용해 다음과 같은 눈 추출 알고리즘을 제안한다.

- i) 후보영역 안에 존재하는 모든 객체를 후보로 설정한다.
- ii) 후보객체의 면적을 기준으로 내림차순으로 정렬한다.
- iii) 객체를 둘러싸는 사각형의 높이가 넓이의 50% 이상이면 제외한다.
- iv) 가장 넓은 2개의 객체를 눈과 눈썹으로 설정한다. 이 중 세로로 위쪽 객체를 눈썹 아래쪽 객체를 눈으로 설정한다.
- v) 얻어진 후보객체가 1개 이상이 아니면, 특징점 얻기를 중단한다.

그림 4 (b)는 왼쪽 눈에 해당하는 후보영역에 들어 있는 모든 객체 중 눈 후보객체를 얻은 결과이다. (b)는 눈 추출 알고리즘을 적용해 객체를 둘러싸는 사각형 중 가로

방향으로 특징을 가지는 후보 객체들만을 표시한 영상이다. 눈 추출 알고리즘은 이진화 레이블링된 영상에서 각 객체 별로 제약조건을 만족하는 객체를 선택하도록 구성되어 있다. 그러므로 최대  $n^2$ 의 객체가 있을 수 있고 복잡도는  $O(n^2)$ 으로 표현할 수 있다.



(a)입력얼굴 영상 (b) 왼쪽 눈 후보 객체

그림 4. 왼쪽 눈 후보객체 (a) Input face image  
(b) candidate objects of left eye  
Fig. 4. The candidate objects of left eye

### 2.2 입 추출 알고리즘

입 추출 알고리즘은 눈 추출 규칙과 유사하다. 하지만 특이한 점은 전처리 과정의 결과 영상에서 눈은 중간에 끊어지는 현상이 없지만, 입의 경우 입술의 가운데 부분에서 끊어지는 현상이 발생한다. 따라서 전처리 결과의 영상으로부터 입을 찾을 경우 원래의 입보다 작은 반쪽 짜리 입을 찾는 경우가 많다. 이러한 문제점을 해결하기 위해서 후보 입 영역 오른쪽 혹은 왼쪽에 또 다른 후보 객체가 존재하면 그 두 개의 객체를 하나로 병합한다. 여기서 제안하고 있는 입 추출 알고리즘도 눈 추출 알고리즘과는 마찬가지로 입의 기하학적인 특징을 이용하고 있다. 입 추출 알고리즘의 복잡도도 눈 추출 방법과 동일하므로 후보 객체의 개수에 비례해서 표현되어  $O(n^2)$ 으로 표현된다.

### 2.3 코 추출 알고리즘

코는 입과 눈 사이에 존재한다. 이러한 점을 감안하여 눈과 입 사이에 코가 존재할 수 있는 후보 영역을 정의한다. 전처리 과정을 거친 결과 중 코 후보 영역만을 분리시킨 후의 영상은 코 아래 쪽 부분이 강하게 강조되어 나타난다. 이 강조된 지점을 얻어내기 위해 다음과 같은 알고리즘을 제안한다.

- i) 후보 영역에 존재하는 모든 데이터를 가로 방향으로 투영시킨다.

- ii) 투영된 그래프에서 최대 값의 위치가 코가 존재하는 세로 방향 위치이다.
- iii) 코 후보영역에서 얻어진 세로 위치의 위아래 세 화소만큼의 데이터를 얻어낸후 세로 방향으로 투영하여 코 양 끝 지점을 얻어낸다.

그림 5는 얻어진 눈과 입 위치에서 코의 후보영역을 얻어 코의 위치를 찾아내는 과정을 나타낸다. 그림에서 확인할 수 있듯이 코 후보 객체들을 가로 방향으로 투영시키면, 코 아래 부분이 최대의 크기를 갖는 그래프가 얻어진다. 이 최대치를 기준으로 위의 과정 iii)을 수행하면 코 아래 좌우 점을 찾아 낼 수 있다. 코를 추출하는 알고리즘은 코 위치의 영상을 가로세로 투영법에 의해서 추출하고 있으므로 영상의 크기  $n$ 에 비례하여  $O(n)$ 의 복잡도를 가진다.

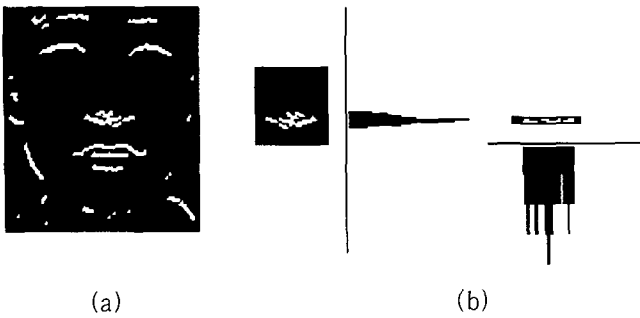


그림 5. 코 추출 과정  
 (a) 이진화 영상 (b) 코 위치 영상의 가로 세로 투영 과정  
 Fig 5. The extraction of nose  
 (a) binary image (b) The projection processing for the position of nose

얼굴의 특징점을 찾는 전체 복잡도는 각 알고리즘의 복잡도의 합인  $O(3n^2 + n)$  으로  $O(n^2)$ 이라고 말할 수 있다. 이는 허프 변환을 이용하여 특징점을 찾는 방법의 복잡도  $O(n^3)$ 보다 작은 복잡도를 가지므로 더욱더 빠른 속도로 특징점을 추출할 수 있다.

### III. 얼굴 인식

제안하는 얼굴 인식 알고리즘은 두 단계의 과정을 거쳐 수행된다. 먼저 얼굴에 존재하는 특징점을 벡터화 시킨 데이터를 이용하여 얼굴 전체 생김새의 유사도를 구하고 두 번째는 특징점만을 포함하는 얼굴 영상을 이용하여 특징점 자체 생김새의 유사도를 구한다.

이러한 데이터는 인식을 원하는 사람들의 여러 장의 얼굴 영상을 이용해서 특징점 벡터와, 얼굴 특징점 영상을

미리 데이터베이스에 저장한다. 특징점 벡터의 유사도 측정은 유클리디안 거리를 계산한 값을 이용하고 특징점 얼굴 영상의 유사도 측정은 상관관계의 측정을 이용한다. 두 개의 유사도를 가중치를 이용해서 더한 값이 최종 유사도가 된다.

얼굴 인식을 수행할 때는 먼저 특징점 벡터 유사도만을 이용해서 후보 얼굴을 찾아내고 찾아낸 후보 얼굴에서 두 번째로 특징점 영상의 유사도를 구해서 최종 유사도를 얻어낸다. 이때 얻어진 유사도 중 최대 값을 가지는 얼굴이 인식된 얼굴이 된다.

#### 1. 얼굴 특징 벡터

인간이 얼굴을 구분하는데 사용할 수 있는 요소는 각 특징을 이루는 눈, 코, 입의 생김새뿐만 아니라 전체적인 배치도 매우 중요하다. 이러한 점을 인식의 단계에 적용하기 위해 그림 6과 같이 얻어진 얼굴 특징을 여러 개의 벡터로 표현한다. 이를 위하여 얻어진 후보 얼굴 중 정확한 얼굴을 찾아내기 위해 얼굴 특징점만을 포함하는 영상을 얻어낸다. 그림 7은 입력 얼굴 영상 중 특징점만을 포함하는 영역을 잘라낸 특징점 영상이다. 영상의 +기호는 특징점 추출 알고리즘에 의해서 얻어진 특징점의 위치를 나타낸다. 여기서 상관관계를 계산하기 위해 얻어진 얼굴 특징 영상은 동일한 크기로 정규화 한다.

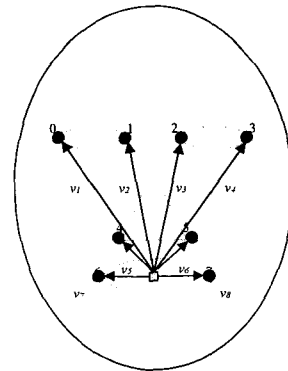


그림 6. 얼굴 특징점 벡터 (a) Face feature vectors  
 그림 7. 얼굴 특징점 영상 (b) Face feature image

한 사람의 특징 벡터  $V$ 는 입 중앙으로부터 각 특징점들 간의 위치를 벡터로 사용하여 다음과 같이 정의한다.

$$V = [v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8]. \quad (1)$$

본 논문에서는 한 사람에 대한 특징점의 오차를 줄이기 위해서 동일한 사람의 여러 장의 얼굴 영상을 입력받아 각  $V$ 를 구해서 평균을 구한다.

$$F = [f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8]. \quad (2)$$

여기서

$$f_i = \frac{1}{n} \sum_{j=1}^n v_j^{(i)}$$

이며  $i$ 는  $i$ 번째 얻은 영상이고  $j$ 는  $j$ 번째 벡터이다. 이렇게 구한  $F$ 는 한사람의 얼굴 특징 벡터가 되어 이후 유사도 계산에서 사용된다.

## 2. 얼굴 유사도 측정

입력 얼굴 영상이 누구인가를 판단하기 위해서 데이터 베이스에 기록된 각 얼굴 영상들과 유사도를 계산한다. 이 중 가장 큰 유사도 값이 나온 얼굴 영상을 입력 얼굴 영상과 같은 사람으로 판단한다. 이를 위하여 본 논문에서는 특징점 영상 자체의 유사도와 특징점 분포 유사도를 정의한다.

$MI_i$ 는 영상자체의 유사도 이다. 이는 입력 얼굴 영상과 데이터 베이스에 존재하는  $i$ 번째 얼굴 영상과의 상관관계로 정의되며 다음과 같이 나타낼 수 있다.

$$MI_i = \frac{E(IT_i) - E(I)E(T_i)}{\sqrt{sd(I) \times sd(T_i)}} \quad (3)$$

- $I$  : 입력 얼굴 영상 데이터
- $T_i$  : 데이터베이스 얼굴 영상 데이터베이스의  $i$ 번째 영상
- $E()$  : 평균
- $sd()$  : 분산

여기서  $E(IT_i)$ 는 입력 영상과 데이터 베이스 영상의 각 픽셀의 곱 영상의 전체 평균 값이다. 이때 평균과 분산은 각 영상의 전체 픽셀의 평균과 분산이다. 한편  $I$ 번째 영상의 특징점 분포의 유사도  $MF_i$ 는 입력된 영상의 8개의 벡터들의 특징점 벡터와 데이터베이스에 존재하는 영상의 특징점 벡터간 유클리드 거리로 정의된다.

$$MF_i = \sum_{j=1}^8 |v_j - f_j^{(i)}| \quad (4)$$

여기서  $v_j$ 는 입력 얼굴 영상의  $j$ 번째 벡터이고  $f_j^{(i)}$ 는  $i$ 번째 데이터베이스 얼굴 영상의  $j$ 번째 벡터를 말한다. 8개 벡터 유클리드 거리의 합을 0과 1사이로 정규화 한다. 그러기 위해서 데이터 베이스 얼굴 영상 중 최대 MF 값을 얻어 그 값으로 나눈다. 여기서 유사도 값이 1에 가까울수록 유사 정도가 높게 되도록 한다.

이러한 유사도를 사용하는 최종 유사도  $M_i$ 는 얼굴 특

징점 영상의 상관관계와 특징점 벡터의 유클리드 거리의 합으로 다음과 같이 정의된다.

$$M_i = \alpha MI_i + \beta \left( 1 - \frac{MF_i}{\max(MF)} \right) \quad (5)$$

여기서  $\max(MF)$ 는 입력 영상과 데이터베이스의 영상의 유사도 검사 값 중 가장 큰 유사도 값이다.  $\alpha$ 와  $\beta$ 는 각 유사도의 가중치이다. 특징점 영상자체의 유사도와 특징점 분포의 유사도를 같은 비중으로 사용하기 위해서 본 논문에서는  $\alpha, \beta$ 를 모두 0.5로 사용한다.

## IV. 실험

영상 자체의 유사도를 구하기 위해 얼굴 특징점 영상을 특정 크기로 정규화 해야한다. 본 논문에서 사용한 정규화 크기는 40 \* 40 픽셀로 사용한다. 이는 70명의 서로 다른 사람 특징점 얼굴 영상의 픽셀 상관관계를 실험해서 40\*40 보다 작아지게 되면, 상관관계는 0.4 이상의 값이 된다. 정규화의 기준은 두 눈 사이의 거리를 사용한다. 얼굴의 기준위치에 해당하는 두 눈 사이의 거리를 모두 동일하게 크기를 변경하여 여러 사람들 간의 상관관계를 비교한다. 두 영상이 같은 사람의 얼굴 영상이라면 두 영상의 상관관계 값은 1에 가깝게 나타나고, 다른 형태의 얼굴일 수록 0에 가까운 값이 나타난다. 즉 영상의 크기가 작아질수록 다른 얼굴임에도 불구하고 상관관계 값이 커지게 된다. 이는 영상의 크기가 작아지면서 특징점들이 자신의 특성을 잃어가기 때문이다.

본 논문에서 제안한 얼굴 인식 방법을 실험하기 위하여 16 사람으로부터 한 사람 당 11개씩 모두 176개의 정면 얼굴 영상을 사용한다. 얼굴 영상 입력 방법은 카메라를 이용해서 특수한 환경을 만들지 않은 평범한 형광등 불빛 아래에서 입력받는다. 그림 8은 실제 입력 영상에 대한 실험 결과 중 일부를 나타낸 그림이다.

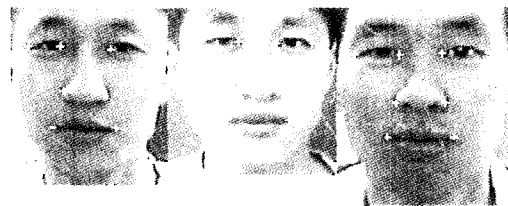


그림 8. 얼굴 특징점 추출 결과  
Fig. 8. The result of extracting face features

이와 같이 수행한 176개의 결과 얼굴 영상에 대해 실제 눈, 코, 입의 위치를 정확히 찾아낸 경우는 97% 이상이다. 그림에서 확인할 수 있듯이 얼굴 영상의 밝기, 크기에 무

관하게 눈의 양 끝점과 입의 양 끝점, 코의 양 끝점을 찾아내고 있음을 볼 수 있다. 또 얼굴의 선택 위치는 대략적으로 얼굴을 포함하는 영상이면 만족하는 결과를 낸다.

얼굴 인식 실험을 위해서 얼굴 특징점 추출 알고리즘에 의해서 얻어진 176장의 데이터로 16 사람의 데이터베이스를 구성한다. 이 후 새로운 입력 영상에 대해서 데이터베이스에 존재하는 16 사람에 대한 유사도를 조사한다. 전체 176장의 새로운 얼굴 영상에 대해서 실험한 결과 평균 91.88 % 이상의 인식률로 얼굴을 인식한다. 그림 9는 실험 결과를 나타낸 그래프이다. 하나의 실선은 한 입력 영상에 대한 16개 데이터베이스 영상과의 유사도 측정 결과를 나타낸다. 그래프의 세로 축은 유사도 값이고, 가로축은 다른 사람 번호가 된다. 유사도가 0으로 나타난 부분은 후보영상에서 제외되어 0으로 설정된 값이다. 그림 9는 3번 사람의 유사도 측정 결과를 나타내는 것으로 모든 실선이 3번 사람 위치에서 최대치를 뽐을 볼 수 있다. 인식 결과 표정의 변화에 의해 얼굴 특징점이 위치가 변화되어, 후보 얼굴 영상을 얻지 못한 경우 오인식율이 커진다.

V. 결 론

본 논문에서 제안하는 알고리즘은 입의 얼굴 영상 입력에서 얼굴 특징에 해당하는 눈, 코, 입을 템플릿 매칭 방법에 비해서 빠르게 찾을 수 있다. 허프 변환 방법이나, 템플릿 매칭 방법은 입력 영상의 상태에 따라 잘못 찾아지는 문제점을 가지고 있지만, 여기서는 얼굴의 특징만을 강조시키는 전처리 단계를 거치기 때문에 적은 오류를 가지고 특징점을 찾아낸다. 이렇게 찾아진 특징점을 이용하여 얼굴 특징점 벡터와 얼굴 특징점 영상을 얻어내 데이터베이스를 구성한다. 새로운 입력 영상에 대해서 동일한 작업을 적용하여 유사도를 측정하여 최대값을 가지는 영상을 찾거나 하는 사람으로 결정한다. 이를 위하여 본 논문에서는 얼굴에 존재하는 특징점들의 위치적인 특성과 특징점 자체가 가지는 특징을 모두 포함한 얼굴인식 방법을 제안한다. 때문에 특징점 자체만의 유사도를 사용해서 인식하는 경우보다 오류를 감소시킬 수 있고, 눈, 코, 입

중 한가지가 비슷한 사람도 구분할 수 있어 정확한 얼굴 인식을 수행한다.

향후 연구해야 할 내용은 얼굴의 특징점만으로 얼굴을 재구성하기 위해서는 더욱 더 많은 특징점의 자동 추출이 필요하다. 또한 심하게 회전된 얼굴 영상에 대해서도 인식 가능하게 하는 연구가 필요하다.

참 고 문 헌

- [1] Xiaobo Li and Nicholas Roeder, "Face contour extraction from Front-view images," *Pattern Recognition*, vol. 28, no. 8, pp. 1167-1179, 1995.
- [2] Guangzheng Yang and T. S. Huang, "Human face detection in a complex background," *Pattern Recognition*, vol. 27, no. 1, pp. 53-63, 1994.
- [3] 최정일, 김수환, 이필규, "얼굴의 특징 추출을 위한 얼굴의 구성 요소 분할", *제 10회 영상처리 및 이해에 관한 워크샵*, pp. 215-220, 1998년 1월.
- [4] X. Xie, R. Sudhakar and H. Zhuang, "On improving eye feature extraction using deformable templates," *Pattern Recognition*, vol 27, no 6, pp. 791-799, 1994.
- [5] G. Chow and X. Li, "Towards a system for automatic facial feature detection," *Pattern Recognition*, vol. 26, no. 12, pp. 1739-1755, 1993.
- [6] R. Brunelli and T. Poggio, "Face Recognition: feature versus templates," *IEEE Transacion on Pattern Analysis and Machine Intelligence*, vol. 15, no. 10, pp. 1042-1052, Oct. 1993.
- [7] R. Kothari, and J. L. Mitchell, "Detection of eye locations in unconstrained visual images," *IEEE ICIP-96, Switzerland*, pp. 519-522 Sep. 1996.
- [8] B. Ju Lee, Jechang Jeong, and Byunguk Choi, "Extraction of feature areas from human face image using a modified hough transform," *IWCVT-96, Korea*, pp. 24-29 Dec. 1996.

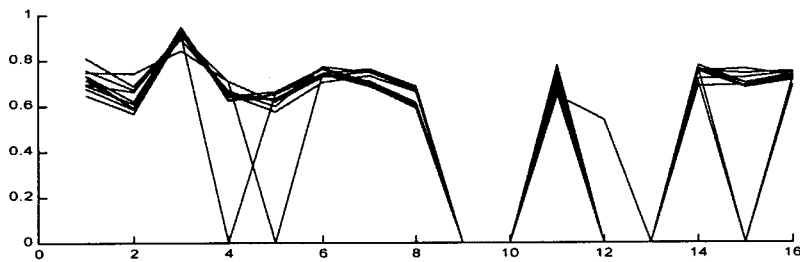


그림 9. 입력 영상과 데이터베이스의 유사도 측정 결과  
Fig 9. The similarities between input image and database images

---

저 자 소 개

---



**전 승 철**

1971년 1월 11일 생  
1996년 2월 한양대학교 전자계산학과 졸업(공학사)  
1998년 2월 한양대학교 대학원 전자계산학과 졸업(공학석사)  
1999년 2월 ~ 현재 한양대학교 대학원 전자계산학과 박사과정  
주관심분야 : Computer Vision, Human Face Recognition, Video Indexing



**박 성 한**

1947년 6월 2일 생  
1970년 2월 한양대학교 전자공학과 졸업(공학사)  
1973년 8월 서울대학교 대학원 전자공학과 졸업(M.S.)  
1984년 5월 텍사스 주립대학 전기 및 컴퓨터공학과 졸업( Ph. D.)  
1974년 3월 ~ 1978년 8월 경북대학교 전자공학과 전임강사  
1984년 5월 ~ 1984년 8월 미국 텍사스 주립대학 Instructor  
1984년 8월 ~ 1986년 2월 금성사 중앙 연구소 수석 연구원  
1989년 8월 ~ 1990년 7월 미국 텍사스 주립대학 Visiting Researcher  
1995년 3월 ~ 1997년 2월 한양대학교 공학대학 학장  
1986년 3월 ~ 현재 한양대학교 전자계산학과 교수  
주관심분야 : 멀티미디어 통신, 컴퓨터 비전, B-ISDN