

**자세와 표정변화에 강인한 눈 위치 검출**

정진권, 김재민, 조성원, 김대환, 김준범, 이진형  
 홍익대학교 전기정보제어공학과 지능정보처리 연구실

**Robust Eye Localization for various Pose and Expression**

Jinkwon Jung, Jaemin Kim, Seongwon Cho, Daehwan Kim, Joonbum Kim, Jinhyung Lee  
 Dept. of Electronin and Electric Engineering, A.I. Lab Hongik University

**Abstract** - 얼굴 영상에서 사람의 눈을 검출하는 것은 얼굴 인식의 전체적인 성능을 좌우하는 매우 중요한 사항이다. 눈 검출은 얼굴 영상의 특징이 변하기 때문에 항상 신뢰할 수 있는 결과를 얻는 것은 어려우며, 또한 실시간 얼굴 인식에 응용되기 위해서는 빠른 연산 시간도 고려되어야 한다. 본 논문에서는 빠르고 정확한 새로운 눈 검출 방법을 제안한다. 첫째, Ada-Boosting 알고리즘을 사용하여 얼굴 영역을 검출한다. 둘째, Intensity valley와 edge 정보를 사용하여 얼굴 영상의 회전(Rotation in plane)을 보상한다. 셋째, Intensity edge정보를 사용하여 두 눈의 수직, 수평라인을 검출한다. 넷째, 일반적인(generic) 사람 눈의 패턴을 이용하여 고안된 Filter로 두 눈의 위치를 검출한다. 본 논문을 통하여 새로 제안된 알고리즘에 대한 논의와 실험 결과를 통해 새로운 알고리즘이 눈 검출에 적합함을 제시한다.

**1. 서 론**

실제 얼굴 인식 시스템이나 얼굴 정보를 활용하는 많은 Application에서 얼굴 영상은 일정한 사이즈로 정규화되어야 한다. 얼굴 영상에 있어서 눈은 가장 확실한 특징점(LandMark)이기 때문에 사이즈 정규화 과정에 앞서 두 눈의 위치를 검출하는 것은 매우 중요한 과정이다.[1] 그러나 얼굴 영상은 표정, 크기, 포즈, 회전, 조명 상황, 안경 등 외부요인에 따라 변화가 심하기 때문에 빠르고 정확하게 눈 위치를 검출하는 것은 쉬운 문제가 아니다. 본 논문에서는 두 눈을 검출하는 빠르고 정확한 새로운 알고리즘을 제안한다. 이 알고리즘은 첫째로 Ada-Boosting[3] 알고리즘을 사용하여 얼굴을 검출한다. 다음으로 Intensity valley와 edge정보를 사용하여 얼굴 영상의 좌우경계, 회전, 눈의 수직, 수평라인을 예측한다. 마지막으로 일반적인 사람 눈의 패턴을 이용하여 고안된 Filter로 두 눈의 후보점을 찾고 위치를 검출한다.

제안된 알고리즘은 템플릿과 같은 모델 영상을 사용하지 않기 때문에 조명, 포즈, 인종에 강인하고, 별도의 데이터베이스를 구성하여 Training하는 과정이 필요하지 않다. 또한 적은 연산량으로 인해 얼굴 인식, 모바일 아바타, 애니메이션등의 실시간 시스템등에 응용될 수 있다.

**2. 본 론**

**2.1 Approximate Eye Localization**

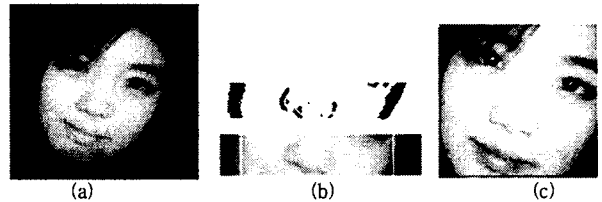
본 논문에서 제안하는 눈의 위치를 찾는 알고리즘은 다음과 같이 7단계로 나누어진다. 첫 번째 단계는 입력 영상에서 얼굴을 검출하는 것이다. 얼굴을 검출하기 위해서 Ada-Boosting알고리즘[3]을 사용하였다. 두 번째 단계는 정확한 얼굴 좌우의 경계선(Face Boundary)을 검출하는 것이다. 배경의 영향을 최소화하고 얼굴 영역안의 정보만을 사용하는 것이 눈의 위치를 찾는 데 중요하다. 본 논문에서는 에지정보를 사용하여 Face Boundary를 검출하였다. 세 번째 단계는 얼굴 영상의 회전(In plane)을 보상하는 것이다. Intensity valley 정보와 Intensity edge 패턴 정보를 사용하여 얼굴 영상의 회전을 보상하였다. 이것은 본 알고리즘의 이어지는 단계의 정확성을 높일 수 있는 중요한 단계이다. 네 번째 단계는 눈썹, 눈의 수직 위치를 검출하는 것이다. 벨리 histogram 정보를 사용하여 검출된 눈과 눈썹의 수직 영역을 기반으로 눈과 눈썹을 포함하도록 잘라낸 눈 영상을 만든다. 다섯 번째 단계는 앞 단계에서 만들어진 영상에서 좌우 눈의 수평 영역을 검출하는 것이다. 눈의 좌우 수평영역은 에지 histogram을 이용하여 검출한다. 여섯 번째와 마지막 단계는 검출된 눈 영상에서 정확한 좌우 눈의 위치를 검출하는 것이다. 이는 벨리 영역과 간단한 필터식으로 눈이 될 만한 후보점을 정하고 제안하는 필터식을 적용하여 눈의 위치를 찾는 과정이다.

**2.1.1 Detect Face Boundary for Eye Localization**

벨리와 에지는 정확한 얼굴 영역안에서 추출되어야 신뢰성이 높아지므로 먼저 얼굴의 좌우 경계(Face Boundary)를 검출하는 과정이 필요하다. 눈과 입의 에지 성분의 영향을 줄이기 위해 검출된 얼굴 영역의 가운데 부분만을 고려한 mid-face이미지에서 Sobel연산

자의 Gx 미분 연산자를 사용하여 수직 에지를 검출하고 다음 식(1)을 이용하여 [그림 1]과 같이 얼굴의 좌우 경계를 검출하였다.

$$E(x) = \text{the number of pixel}(x, y) \in \text{Edge, for all } y$$



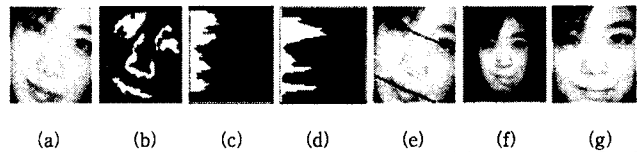
**<그림 1> (a) 원 얼굴 영상 (b) 좌우 얼굴 경계선 검출 과정 (c) 얼굴 경계선 검출 확대 영상**

**2.1.2 Estimation of Face Tilt Angle**

사람 얼굴 영상에서, 특히 저해상도 영상일 경우 눈, 눈썹 그리고 입은 수평방향으로는 길고 수직방향으로는 짧다. 또한 인접한 지역에 비해서 어두운 색깔을 띠게 된다. 이러한 특징 때문에 얼굴 영상에서 그레이 모폴로지 Black-hat 필터를 사용하면 수평으로 긴 벨리[4]가 검출된다. [그림 2]에서 보듯이 이것은 똑바로 선 얼굴에서 다음 식(2) V(y)의 분산이 최대라는 것을 의미한다. 일단 벨리가 추출되면 V(y)의 분산을 최대화하는 각도를 구한다.

$$V(y) = \text{the number of pixels}(x, y) \in \text{valley, for all } x$$

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N V^2(y) - \left[ \frac{1}{N} \sum_{i=1}^N V(y) \right]^2$$



**<그림 2> The estimation of face tilt angle: (a) 얼굴 경계 검출 영상 (b) 벨리 영상 (c) 기울어진 얼굴에서의 V(y) (d) 똑바로 선 얼굴에서의 V(y) (e) 기울어진 각도를 보여주는 세 라인 (f) 회전 정규화 된 얼굴 영상 (g) 경계, 회전 정규화 영상**

**2.1.3 Estimation of Vertical Eye Location**

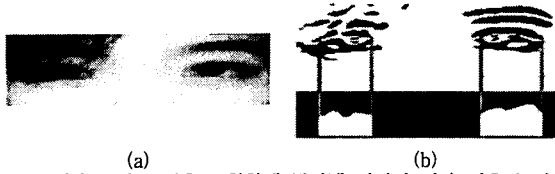
회전 정규화 된 얼굴 영상[그림 2.(g)]에서 눈, 눈썹, 코, 입의 수직 좌표는 식(2) V(y)의 국부 최대(Local Maximum)값과 일치하며 눈썹이나 눈에서 전역 최대(Global Maximum)값을 갖는다. 또한 일반적으로 눈썹과 눈 사이의 수직 거리는 눈과 코의 수직 거리에 비해 짧다. 이 특성을 이용하면 눈의 수직 좌표를 작은 범위내로 한정시킬 수가 있다. [그림 3]을 보면 조명이 변화하였을 때도 이러한 특성을 보이는 것을 알 수가 있다.



**<그림 3> Variation of the function V(y) due to the illumination and pose changes**

### 2.1.4 Estimation of Horizontal Eye Location

일단 눈의 수직 위치가 정해지면 두 눈의 수평 위치를 정하기 위해 [그림 2.(g)]의 얼굴 영상에서 두 눈썹과 두 눈만을 포함하는 영역을 잘라 낸다. 이 때 눈썹과 눈이 확실하게 포함될 수 있도록 앞 단계에서 결정된 눈과 눈썹의 수직 좌표보다 약간 크게 영역을 설정하였다. 이 영역만을 잘라낸 이미지에서 Sobel Gy미분 연산자를 이용하여 수평 에지를 검출하고 식(1)의 E(x)함수에 임계값을 적용하여 다음 그림과 같이 좌우 눈의 수평영역을 정하였다.



〈그림 4〉 (a) 눈과 눈썹을 포함하게 잘라낸 이미지 (b) 좌우 눈의 수평 영역 설정 과정

### 2.1.5 Eye Localization

얼굴 영상에서 두 눈을 포함하는 수직, 수평 영역이 검출되면 이 영역 안에서 밸리를 검출한다. 그리고 눈의 후보점은 검출된 밸리 영역 안에서 [그림 5]의 사각 필터를 이용한 식(3)의 Cost값이 주변 8픽셀보다 큰 지점으로 설정한다. 눈의 실제 모양은 타원형에 가깝지만 빠른 연산 시간을 위해 사각형 모양의 필터를 사용하였다[5].

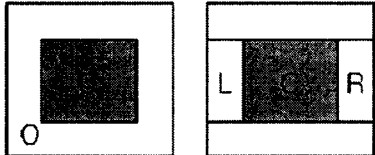
$$\text{Cost}(x, y) = (v_o(x, y) - v_c(x, y)) \times \frac{(u_o(x, y) - u_c(x, y)) / (Osize - Csize)}{u_c(x, y)}$$

$u_c(x, y)$  : 작은 사각형 안의 픽셀값의 합

$u_o(x, y)$  : 큰 사각형 안의 픽셀값의 합

Osize : 큰 사각형의 픽셀 개수

Csize : 작은 사각형의 픽셀 개수



〈그림 5〉 눈의 후보점과 최종 위치 결정에 사용된 사각형 필터

### 2.1.6 Eye Localization

눈의 최종적인 위치는 전 단계에서 검출된 눈의 후보점에서 다음 식(4)을 적용하여 결정된다. 이 식의 첫 번째 항은 (x,y)를 중심으로 하는 작은 사각형안의 밝기 정도를 측정하고 두 번째 항은 큰 사각형과 작은 사각형의 밝기값 차이를 나타낸다. 세 번째 항은 [그림 5]의 좌측과 우측 사각형의 밝기값의 차이를 측정하고 네 번째 항의 Cross(x,y)는 평균 교차 횟수(Mean Crossing Number)이다[1]. 마지막 다섯 번째 항은 작은 사각형안에 포함된 밸리 픽셀의 상대적인 많고 적음을 나타낸다. 결론적으로 모든 후보점 중 최소값을 갖는 위치가 최종 눈의 위치로 결정된다.

식(4)

$$\begin{aligned} \text{output}(x, y) &= \alpha \frac{u_c(x, y)}{\text{Avg}[u_c(x, y)]} \\ &+ \beta \frac{-[u_o(x, y) - u_c(x, y)]}{\max[u_o(x, y) + u_c(x, y)]} \\ &+ \gamma \frac{|u_L(x, y) - u_R(x, y)|}{[u_L(x, y) + u_R(x, y)]} \\ &+ \delta \frac{-\text{Cross}(x, y)}{\max[\text{Cross}(x, y)]} \\ &+ \lambda \frac{-\text{Valley}(x, y)}{\max[\text{Valley}(x, y)]} \end{aligned}$$

$u_L(x, y)$  : 왼쪽 사각형안의 픽셀 값의 합

$u_R(x, y)$  : 오른쪽 사각형안의 픽셀 값의 합

Valley(x, y) : 작은 사각형안의 밸리 픽셀의 수

### 2.2 Experimental Result

제안한 알고리즘의 눈 위치 검출의 성능을 평가하기 위해서 본 논문에서는 홍익대학교에서 직접 제작한 얼굴 데이터베이스로 실험하였다. 이 데이터베이스는 7가지 자세(정면, 좌, 우, 좌상, 좌하, 우상, 우하)와 4가지의 다른 조명 상태로 만들어졌다. 10명의 사람으로 총 280장의 이미지이다. 남자 3명, 여자 7명이고 이 중 안경을 쓴 사람은 2명으로 각각 두꺼운 볼테 안경과 얇은 무테 안경을 쓰고 있다. 실험 결과를 [표 1]에 나타내었다.

〈표 1〉 홍익 데이터베이스 240장을 사용한 실험 결과

| 안경 착용 | 총수  | 성공  | 실패 | 성공률(%) |
|-------|-----|-----|----|--------|
| 미 착용  | 224 | 219 | 5  | 97.8   |
| 착용    | 56  | 49  | 7  | 87.5   |
| Total | 280 | 268 | 12 | 95.7   |



〈그림 6〉 홍익 데이터베이스 눈 위치 검출 성공사례(정면, 좌, 우, 아래, 안경)

### 3. 결 론

본 논문에서는 눈의 위치를 검출하는 빠르고 정확한 알고리즘을 제안하였다. 실험 결과를 보면 정면의 얼굴 뿐만 아니라 -30~+30도의 회전(in depth)된 얼굴에서도 신뢰할 수 있는 결과를 보여주며 각각 다른 조명 상황에서 획득된 영상에서도 좋은 성능을 발휘하는 것을 볼 수 있다. 연산량면에서도 기존의 가변 템플릿 매칭이나 EigenSpace방법에 비해 적다. 단, 두꺼운 안경을 쓴 사람의 얼굴인 경우 허용되는 회전의 범위가 다소 줄어드는 경향을 보이며 안경테를 눈으로 오인할 수 있는 확률이 있다. 이런 문제는 얼굴에서 안경을 분리하는 방법[4]를 이용하여 안경을 제거한다면 해결될 수 있을 것이다. 그리고 조명 상태를 개선하는 알고리즘이 추가된다면 실험한 데이터베이스보다 더욱 극한 조명 상황에서도 눈을 찾을 수 있을 것이다. 눈의 위치 정보는 눈 자체의 위치 뿐만 아니라 여타 다른 얼굴의 정보를 얻을 수 있게 해 준다. 마지막으로 제안하는 알고리즘으로 찾은 눈의 위치에서 AAM(Active Appearance Model)을 이용하여 눈의 모양을 추출한다면 얼굴 인식이나 아바타 등의 기타의 여러 방면으로 활용될 수 있을 것이다.

### [참 고 문 헌]

- [1] Lin, C.H., Wu, J.L., "Automatic Facial Feature Extraction by Genetic Algorithm", IEEE Trans On Image Processing, Vol.8 No. 6, 834-845, 1999
- [2] Hamamoto, Y., "A Gabor Filter-based Method for Fingerprint Identification", Intelligent Biometric Techniques in Fingerprint and Face Recognition, eds. L.C. Jain et al., CRC Press, NJ, pp.137-151, 1999
- [3] Xiao, R., Li, M.j., Zhang, H.J., "Robust Multipose Face Detection in Images", IEEE Trans On Image Processing, Vol.14 No.1, 31-41, 2004
- [4] Kawaguchi, T. Rizon, M., "Iris Detection Using Intensity and Edge Information", Patter Recognition Vol.36 549-562, 2003
- [5] Viola, P., Jones, M., "Rapid Object Detection Using a Boosted Cascade of Simple Features", In Proc. 2001 IEEE Computer Vision and Pattern Recognition, vol. 1 HI, 511-518, Dec 2001