

An Enhanced Method for Detecting Iris from Smartphone Images in Real-Time

Seong-Hoon Kim[†] · Gi-Tae Han^{**}

ABSTRACT

In this paper, we propose a novel method for enhancing the detection speed and rate by reducing the computation in Hough Circle Transform on real-time iris detection of smartphone camera image. First of all, we find a face and eyes from input image to detect iris and normalize the iris region into fixed size to prevent variation of size for iris region according to distance from camera lens. Moreover, we carry out histogram equalization to get regular image in bright and dark illumination from smartphone and calculate minimal iris range that contains iris with the distance between corner of the left eye and corner of the right eye on the image.

Subsequently, we can minimize the computation of iris detection by applying Hough Circle Transform on the range including the iris only. The experiment is carried out in two case with bright and dark illumination. Our proposed method represents that detection speed is 40% faster and detection rate is 14% better than existing methods.

Keywords : Eye Detection, Iris Detection, Hough Circle Transform, Biometrics, Smart Car

스마트폰 영상에서의 개선된 실시간 눈동자 검출 방법

김성훈[†] · 한기태^{**}

요 약

본 논문은 스마트폰 영상의 실시간 눈동자 검출에서 허프 원 변환 연산의 연산량 축소를 통한 속도 및 검출율 개선 방법을 제안한다. 눈동자를 검출하기 위해서는 입력 영상에서 얼굴과 눈을 검출하고, 눈 영역의 크기에 따라 눈동자의 크기가 변하는 것을 방지하기 위해 일정크기로 눈 영역을 정규화하며, 다양한 조명환경에서 눈동자가 검출이 가능하도록 히스토그램 평활화를 실시하고, 눈의 양쪽 끝점간의 거리를 구하여 영상에서의 실제 눈동자의 크기를 포함할 수 있는 최소한의 눈동자 크기 범위를 계산하여 허프 원 변환에 적용함으로써 연산량을 최소화 하였다. 제안한 방법을 밝은 조명과 어두운 조명에서 실험한 결과 기존 방법들과 비교하여 눈동자 검출 속도는 40% 이상, 검출율은 14% 이상 향상된 것을 보였다.

키워드 : 눈 검출, 눈동자 검출, 허프 원 변환, 생체인식, 스마트 카

1. 서 론

최근 IT융합기술의 발전에 따라 다양한 산업에서 생체인식 기술을 IT기술과 융합한 신 기술을 선보이고 있다. 특히 눈동자 인식 기술을 이용하여 스마트 디바이스의 화면을 제어할 수 있는 아이 스크롤링(eye scrolling), 아이 포즈(eye pause)와 같은 사용자 편의 제공 기술이 등장하고 있다. 또한 스마트카에서 IT기술 융합으로 운전자의 안전운전 서비

스가 가능한 고 안전 지능형차량 (Advanced Safety Vehicle : ASV)의 개발이 활발하게 이루어지고 있다. ASV 개발에는 여러 가지 안전기술이 탑재되며, 이 기술은 사고 예방 및 회피, 충돌 등 위험상황으로부터 운전자 및 탑승자를 보호하여 교통사고 및 피해를 줄일 수 있는 기술로 평가되고 있다.

주요 안전기술 중 하나인 시선추적시스템은 운전자 상태를 감시하는 시스템으로 운전자 정면에 설치된 카메라를 이용하여 운전자의 눈, 입과 같은 얼굴의 특징점을 추출하여 내부의 분석 장치에 의해 운전자의 졸음상태, 주시방향, 깜박임 등을 감지하여 위험상황으로 판단될 경우 운전자에게 경고하는 기술이다[1, 2].

이러한 기술들을 사용하기 위해서는 얼굴 검출, 눈 검출, 입 검출, 눈꺼풀 검출 등이 실시간으로 이루어져야하며, 차량 외부로부터의 조명, 날씨 등 다양하게 변하는 조명환경에서 강건하게 검출할 수 있어야한다.

* 본 논문은 2013년도 가천대학교 교내연구비 지원(GCU-2013-R183)에 의한 결과임.

† 준 회 원: 가천대학교 전자계산학과 석사과정

** 정 회 원: 가천대학교 IT대학 컴퓨터미디어융합학과 교수

논문접수: 2013년 7월 16일

수정일: 1차 2013년 8월 2일

심사완료: 2013년 8월 3일

* Corresponding Author : Gi-Tae Han(gthan@gachon.ac.kr)

기존에 연구된 눈동자 검출 방법에는 눈 영역에서의 영역 분할(segmentation) 방법과 눈동자의 형태를 이용한 검출 방법 등이 있다. 영역분할을 이용한 눈동자 검출 방법은 영상의 색상이나 명도를 이용하여 검출하는 방법으로 단순히 눈동자의 영역을 검출하기 위해서는 이 방법이 적합하지만, 눈동자의 중심점 및 눈동자의 크기와 같은 부가적인 정보를 얻기 위해서는 별도의 계산을 필요로 한다. 또한 눈동자의 형태를 이용한 눈동자 검출의 대표적인 방법이 허프 원 변환을 이용한 방법인데, 눈동자가 원의 형태를 갖고 있다는 특징을 이용하여 눈동자를 검출한다. 이 방법은 형태적 특징을 이용하여 눈동자를 검출하게 되므로 눈동자 검출 후 중심점 및 반지름과 같은 부가적인 정보를 얻을 수 있다는 것이 장점이지만 원의 검출 시 지정되는 반지름의 범위가 넓을수록 혹은 이미지의 복잡도가 높을수록 계산량이 많아지고 검출율이 떨어진다는 단점이 존재한다.

눈동자 검출 방법 중에 Retno Supriyanti가 최근에 발표한 제안한 방법은 검출되는 눈의 영역의 크기에 따라 눈동자의 크기가 변하는 것을 고려하지 않고 반지름 범위를 넓게 지정하므로 검출되는 눈동자의 위치가 실제 눈동자의 위치와 일치하지 않는 문제가 있었다[6].

Klaus Toennies가 제안한 방법은 카메라의 초점 거리 값을 이용하여 카메라와 눈동자의 거리가 0.5m에서 5m 사이에 있다는 것을 가정하고, 해당 거리와 눈동자의 직경에 대한 비율을 계산하여 반지름 범위를 지정하게 되므로 넓은 반지름 범위를 사용하게 되어 연산량이 많다는 문제가 있었다[7].

본 논문에서는 스마트폰 기반 실시간 생체인식 기술에 활용하기 위해 눈 영상의 보정을 통해 복잡도를 낮추고, 눈의 양쪽 끝점 간의 거리를 구하여 원 검출 시 반지름 범위를 최소화함으로써 눈동자 검출의 계산량을 줄여 실시간에서의 빠른 눈동자 검출 방법을 제안하고 있다.

2. 관련 연구

2.1 Face detection algorithm

기존의 얼굴 등의 객체 검출 방법에는 대표적으로 haar-like feature를 이용한 검출 방법과 MCT(Modified Census Transform)영상을 이용한 검출 방법이 있으며, 두 방법 모두 adaboost를 기반으로한 객체 분류기(classifier) 생성 방법이다[3, 4].

Haar-like feature 방법은 Viola와 Jones가 제안한 haar-like feature에 의해 검출할 객체의 특징들을 학습시켜 약 분류기(weak classifier)를 만들며, 약 분류기에 의해 강 분류기(strong classifier)를 생성하고 만들어진 강 분류기를 선형구조(cascade)로 연결하여 입력된 영상에서 객체를 검출하는 방법이다[4].

특징 값은 Haar-Like Feature의 흰 영역에 포함되는 픽셀들의 합에서 검은 영역에 포함되는 픽셀의 합의 차로 표현된다. 또한 이러한 픽셀의 합 계산은 적분 영상에 의해서 매우 빠른 속도로 연산되며, haar-like feature에 의해 계산

된 특징 값들을 adaboost 알고리즘에 적용하여 객체를 검출하기 위한 분류기를 생성한다.

Adaboost 알고리즘은 여러 개의 약 분류기(weak classifier)를 선형 조합하여 강 분류기(strong classifier)를 만드는 알고리즘으로, 여러번의 반복을 통하여 약 분류기를 생성한다. 이 때 약 분류기가 나타내는 에러를 최소화 하는 약 분류기를 획득하기 위하여 이전의 약 분류기에서 나타난 오류 값이 클수록 더 높은 가중치를 주어 다음 약 분류기 생성 시 더 개선되어야 할 분류기로 판단하여 오류 값을 점점 최소화 시키는 새로운 약 분류기를 생성하고 최종적으로 생성된 여러 개의 약 분류기들을 선형 조합하여 강 분류기를 생성한다[10]. 생성된 강 분류기는 결정 트리 형태로 만들어지며 객체 검출 시 각 스테이지마다 존재하는 특징들의 위치와 임계값을 이용하여 검출할 객체인지 판별하게 되며, 모든 스테이지에 대하여 특징 값을 비교할 필요 없이 검출할 객체의 주요 몇 개의 특징만을 고려하여 판별하는데 이용한다.

이러한 특징 기반의 얼굴 검출 방법은 다양한 조형에서도 얼굴 검출이 가능하다는 장점이 있으며, Jianguo Li가 최근에 발표한 방법[5]은 SURF(Speeded Up Robust Features) 알고리즘을 기반으로 얼굴의 특징을 학습하여 에러를 발생시키는 약 분류기를 최소화하여 최종적으로 얻어지는 강 분류기 성능을 향상시켰다. 이 방법은 얼굴 검출 시 haar-like feature보다 더 적은 개수의 특징점을 이용하여 기존의 얼굴 검출 방법보다 향상된 검출 속도와 검출율을 나타낸다.

2.2 허프 원 변환(hough circle transform)

허프 원 변환은 영상에서 원의 형태를 검출하기 위해 중심점과 반지름을 찾는 방법으로 Fig. 1과 같이 반지름 r을 갖는 원 C에 대해서 원의 둘레에 위치하는 n개의 점 P_n을 중심으로 같은 r의 값을 갖는 n개의 원을 그렸을 때 n개의 원들이 서로 교차하는 지점 P_c(point of cross)를 알 수 있으며, 원 C의 중심이 가장 많은 교차점을 갖게 된다[11].

즉, 반지름 r의 값을 갖는 원 C_n의 중심점 (x, y)는 점(a, b)에서 반지름 r의 크기를 갖는 원의 둘레에 존재한다. 이를 식으로 표현하면 아래 Equation (1)과 같다.

$$x = a + r \times \cos\theta, y = b + r \times \sin\theta \tag{1}$$

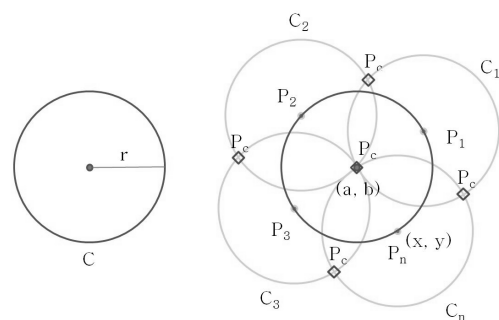


Fig. 1. Hough circle transform example

이 때, 점(x, y)가 위치할 수 있는 원의 중심(a, b)는 반지름 r과 0~360° 까지의 θ 값에 대하여 다음과 같이 Equation (2)로 표현할 수 있다.

$$a = x - r \times \cos\theta, b = y - r \times \sin\theta \quad (2)$$

위 Equation (2)를 이용하여 원의 중심점을 찾기 위해 먼저 입력 영상의 외곽선(edge) 정보를 추출한다. 따라서 추출된 외곽선의 모든 점(x, y)에 대해 반지름 r의 크기 일 때, 0~360° 까지의 θ 값을 갖는 지점 (a, b)에 대해 hough space에 투표(vote)를 실시하여 높은 투표값이 나타나는 위치가 원의 중심점임을 알 수 있다.

3. 제안하는 방법

본 논문에서 제안하는 방법은 Fig. 2와 같이 입력 영상에서 얼굴 검출, 눈 검출, 눈동자 검출과 같은 순서로 처리된다.

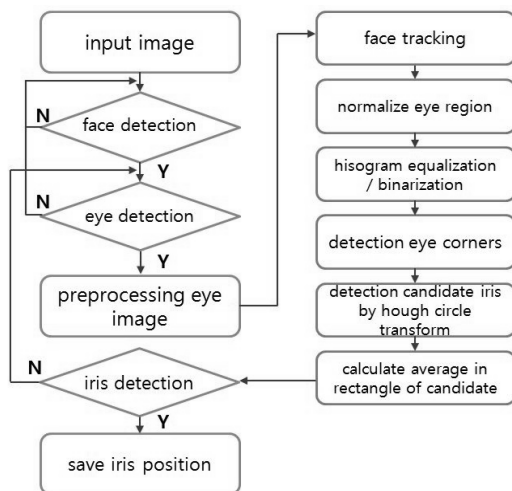


Fig. 2. Flowchart of algorithm

먼저 haar-like feature로 학습된 얼굴 및 눈 검출기를 이용하여 얼굴 영역과 눈 영역을 검출하고 눈동자를 검출하기 위해 영상의 전처리 과정을 수행한다.

전처리 과정에서는 첫 번째 프레임이 아닌 경우에 이전에 검출된 눈의 위치와 현재 검출된 눈의 위치 간의 차이를 구하여 이전에 검출된 얼굴 영역의 위치를 계산된 차이만큼 이동시켜 얼굴을 트래킹 한다. 다음으로 얼굴의 전후 이동에 따라 눈동자크기 변화를 줄이기 위해 검출된 눈의 영역을 일정한 크기로 정규화 시키며, 눈동자 검출율과 성능을 향상시키기 위해 히스토그램 평활화 및 이진화를 수행한다.

눈동자를 검출하기 위해서는 먼저 검출할 원의 반지름 범위를 지정해야 하며, 이는 양쪽 눈 끝점간의 거리를 이용하여 구해진다. 눈의 양쪽 끝점은 이진화 시킨 눈 영역의 영상에서 눈의 양쪽 끝점을 검출한다. 일반적으로 눈동자의 직경과 안구의 직경의 비율이 1 : 2.5 라는 점[8, 9]과 양쪽

끝점 간 거리를 이용하여 최소 범위의 눈동자 반지름 범위를 추정하고, 이를 통해 일정범위 내에서의 허프 원 변환을 수행하여 눈동자 후보들을 검출한다.

마지막으로 검출된 눈동자 후보들에 대하여 눈동자의 위치를 결정하기 위해 후보들의 원 내부의 일부 사각 영역에 대하여 평균 픽셀 값을 계산하고, 평균값이 일정 임계값 이하로 나타나는 후보들 중에서 가장 낮은 평균을 보이는 위치를 눈동자의 위치로 결정한다.

3.1 눈동자 검출 방법

눈동자 검출을 위해 먼저 얼굴 및 눈 검출이 우선 수행되며 이 과정에서 haar-like feature로 얼굴 및 눈을 검출한다. 눈 검출 시 얼굴영역 내에서 눈과 유사한 특징을 갖고 있는 영역이 존재한다면 양쪽 눈 외에 오검출 영역이 검출될 수 있으므로, 이를 방지하기 위해 Fig. 3과 같이 검출된 얼굴 영역을 상단 영역과 하단 영역으로 나누어 상단 영역에서 검출된 눈 영역만을 올바르게 검출된 눈 영역으로 판단한다. 또한 검출된 눈 영역들의 위치 관계를 이용하여 오른쪽 눈, 왼쪽 눈을 결정한다.

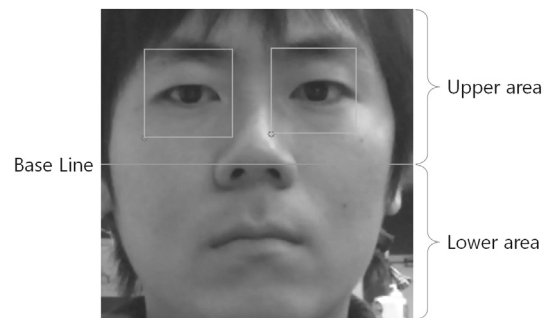


Fig. 3. Decision of right eye and left eye

본 논문에서는 얼굴 트래킹을 위해서 Fig. 4와 같이 이전에 검출된 양쪽 눈의 위치와 현재 검출된 양쪽 눈의 위치의 차를 구하여 얼굴 영역의 차 위치 만큼 갱신한다. 따라서 첫 프레임 또는 눈을 검출하지 못했을 때에만 얼굴 검출 알고리즘을 수행하도록 함으로써 전체적인 알고리즘 성능이 향상 되도록 하였다.

눈동자 검출 시 얼굴의 전후 이동에 따라 눈 영역의 크기가 변하게 되는데, 이에 따라 눈동자의 크기 또한 변하게 되므로 검출된 눈 영역은 Equation (3)를 이용하여 Fig. 5와 같이 일정 크기로 정규화 시켜 눈동자의 크기를 항상 일정한 크기로 나타나게 함으로써 허프 원 변환 시 검출할 원의 반지름 범위의 변화를 최소화할 수 있다. 본 논문에서는 실험을 통하여 눈동자 검출 속도 및 정확성에서 만족할 수준의 성능을 보이는 크기로 정규화 하였다. Equation (3)에서 W' 과 H' 은 정규화 시킬 눈 영역의 가로 값, 세로 값이며, 현재 검출된 눈 영역의 가로 값과 세로 값인 W 와 H 를 이용하여 정규화를 시키기 위한 비율 값 S_W 와 S_H 를 구할 수 있다.

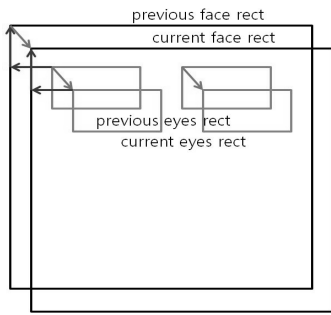


Fig. 4. Face tracking by using eyes coordination

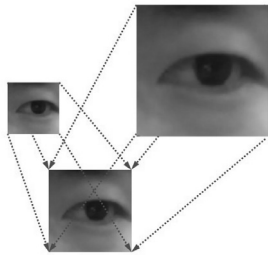


Fig. 5. Normalization of eye region scale

$$W' = W \times S_W, H' = H \times S_H \quad (3)$$

$$S_W = \frac{W'}{W}, S_H = \frac{H'}{H}$$

또한 정규화 된 눈 영역에서 눈썹 및 노이즈 등으로 인한 눈동자 검출의 정확도 저하를 방지하고 알고리즘의 연산량을 최소화하기 위해 Fig. 6과 같이 눈의 영역을 수평으로 4등분 하여 중심부의 2개 영역을 관심영역으로 설정하였다.

또한 관심영역으로 설정된 눈 영역을 Fig. 7(A)와 같이 grayscale로 변환하고, 조명에 강인하게 눈동자를 검출하기 위해서는 영상의 조명 및 대비개선이 필요하므로 Fig. 7(B)와 같이 히스토그램 평활화(histogram equalization)를 이용하여 눈 영역 영상의 대비를 보정하였다.

보정된 눈 영역 영상에서 눈동자와 눈의 외곽선 추출을 위해 Fig. 8과 Fig. 9와 같이 밝은 조명 상태와 어두운 조명 상태에서의 눈 영역 영상에서 눈동자와 눈의 외곽선 영역이 추출되는 최적의 임계값(threshold)을 실험을 통하여 지정하고, 임계값을 이용하여 이진화 하였을 때, Fig. 10과 같이 불필요한 피부 영역은 제거되고 눈동자와 외곽선 영역만 추출되는 것을 알 수 있다.

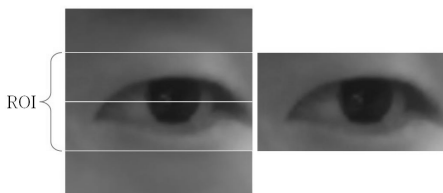


Fig. 6. ROI of eye region

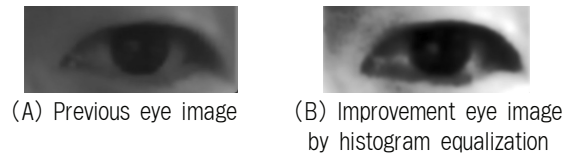


Fig. 7. Histogram equalization of eye image

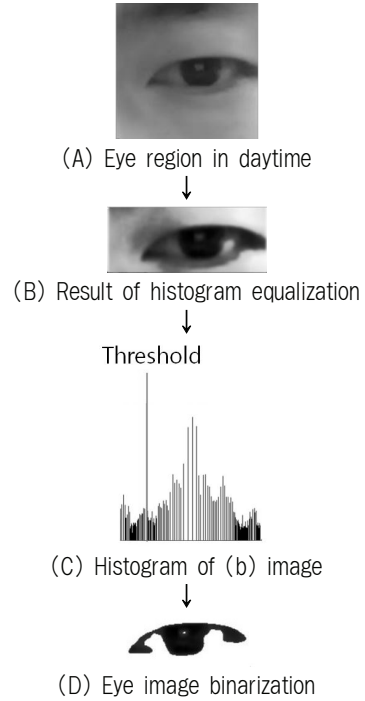


Fig. 8. Decision of threshold in daytime

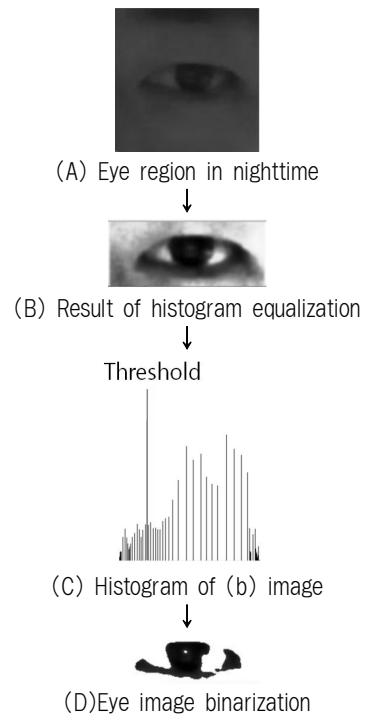


Fig. 9. Decision of threshold in nighttime

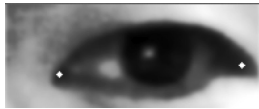


Fig. 10. Binarization of eye image

눈동자를 포함하는 눈동자 범위를 구하기 위해 이진화 처리된 눈 영상에서 가장 오른쪽에 위치한 검은 픽셀과 가장 왼쪽에 위치한 검은 픽셀을 다음 Fig. 11과 같이 눈의 양쪽 끝점으로 추출한다.



(A) Finding eye corners in binary eye image



(B) Result of extraction eye corners

Fig. 11. Extraction eye corners

검출된 양쪽 끝점에서 눈동자의 크기 범위는 다음 Fig. 12와 Equation (4), (5), (6)에 의해 구해진다. P_x 와 P_y 는 왼쪽 끝점의 x, y값을 나타내며, P'_x 와 P'_y 는 오른쪽 끝점의 x, y값을 나타낸다. 이를 이용하여 두 점간의 차이 Δx , Δy 와 두 끝점간의 거리 L 을 구한다.

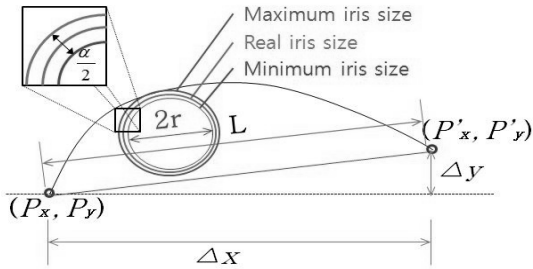


Fig. 12. Calculation iris radius range

$$\Delta x = |P_x - P'_x|, \Delta y = |P_y - P'_y| \quad (4)$$

$$2r = \frac{L}{2.5}, L = \sqrt{\Delta x^2 + \Delta y^2} \quad (5)$$

$$r \leq r_{real} \leq r + \alpha \quad (6)$$

일반적으로 사람의 눈동자 직경은 10mm 정도이며, 안구의 직경은 25mm로 1 : 2.5의 비율을 갖는데[8, 9], 여기서 두 끝점간 거리 L 은 안구를 감싸고 있는 피부에 의해 구해진 거리이므로 안구의 직경보다 항상 작다는 가정을 할 수 있다. 따라서 비율 값을 사용하여 두 끝점간의 거리 L 을

2.5로 나누었을 때 구해지는 눈동자의 직경은 실제 눈동자의 직경보다 항상 작으므로 이를 눈동자 검출을 위한 최소 눈동자 직경으로 지정한다. 최대 눈동자 직경은 아래 Table 1과 같이 영상의 최초 10 프레임 동안 최소 눈동자 직경에 α 값을 1부터 10까지 증가시켜가면서 더했을 때, 가장 높은 검출율을 나타내는 α 값을 Fig. 13과 같이 지정한다.

Table 1. The result of iris detection by α value

α value	1	2	3	4	5
average of detection rate	79%	80%	82%	84%	84%
α value	6	7	8	9	10
average of detection rate	85%	87%	91%	90%	90%

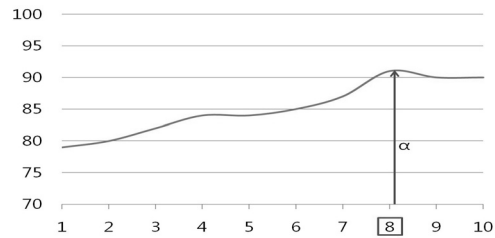


Fig. 13. Decision α value

전처리과정이 모두 끝나면 추정된 반지름 범위를 이용하여 허프 원 변환을 수행하여 아래 Fig. 14와 같이 눈동자의 후보 위치를 검출해낸다.

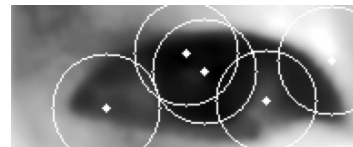


Fig. 14. Detection iris candidates

후보 위치 중 눈동자의 위치를 찾기 위해 원 내부의 평균 픽셀 값을 계산하여 눈동자의 위치를 결정하게 되는데 Fig. 15와 같이 히스토그램 평활화로 보정된 눈 영상에서 눈동자 영역은 항상 가장 어둡게 나타난다는 특징을 이용하여 후보 눈동자 위치에서의 실제 눈동자 위치를 찾게 된다. 눈동자의 위치를 결정하는 알고리즘은 다음과 같다.

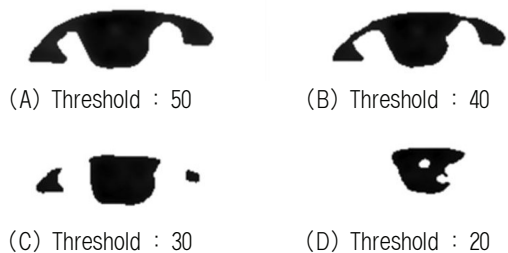


Fig. 15. Binary eye image by threshold

[Algorithm for searching iris]

- ① 중심점 (x, y) 와 반지름 r 을 갖는 원 C 를 원소로 하는 눈동자의 후보들의 집합 S_C 를 구한다.

$$S_C = \{C_1, C_2, \dots, C_n\} \quad C_i = \{x_i, y_i, r_i\}$$

- ② 집합 S_C 의 각각의 원소 C_1, C_2, \dots, C_n 들에 대하여 다음 Equation (7)에 의해 원에 내접하는 정사각형 R_1, R_2, \dots, R_n 을 구한다.

$$R_i = (x'_i, y'_i, width_i, height_i) \tag{7}$$

$$width_i = height_i = \sqrt{\frac{(2r_i)^2}{2}}$$

$$x'_i = x_i - \left(\frac{width_i}{2}\right), y'_i = y_i - \left(\frac{height_i}{2}\right)$$

- ③ 구해진 R_1, R_2, \dots, R_n 에 대하여 다음 Equation (8)에 의해 n 개의 직사각형 R'_1, R'_2, \dots, R'_n 를 구한다.

$$R'_i = (x'_i, y''_i, width_i, height'_i) \tag{8}$$

$$height'_i = \frac{height_i}{2}, y''_i = y'_i - height'_i$$

- ④ 직사각형 R'_1, R'_2, \dots, R'_n 에 대하여 Grayscale 영상에서 각각의 직사각형 영역 내 평균 픽셀 값을 원소로 하는 집합 $S_{avg}(I_{R'_i})$ 를 구한다.

$$S_{avg}(I_{R'_i}) = \{avg(I_{R'_1}), avg(I_{R'_2}), \dots, avg(I_{R'_n})\}$$

- ⑤ 집합 $S_{avg}(I_{R'_i})$ 의 원소들에 대하여 다음과 같이 임계값 T 이하인 원소들만을 선택한 집합 $S_{avg}(I_{R'_i})'$ 를 구한다.

$$S_{avg}(I_{R'_i})' = \{avg(I_{R'_1}), avg(I_{R'_2}), \dots, avg(I_{R'_m})\}$$

단, $avg(I_{R'_j}) \leq T \quad (j = 1, \dots, m)$

- ⑥ 집합 $S_{avg}(I_{R'_i})'$ 의 원소의 개수가 1개이면, 해당 원소에 해당되는 원 C 를 눈동자의 위치로 저장하고, 원소의 개수가 2개 이상일 때, 원소들 중 최소값을 가지는 원소에 해당되는 원 C 를 눈동자의 위치로 저장한다.

위 알고리즘 ① 단계에서는 허프 원 변환을 통해 얻은 눈동자의 후보들을 집합 S_C 로 구성하며 하나의 후보는 원의 중심점 x, y 와 반지름 값 r 을 갖는다. ② 단계에서는 원에 내접하는 정사각형을 아래 Fig. 16과 같이 구하며, 정사각형 R_i 는 정사각형의 좌상단 꼭지점 위치 x', y' 과 가로, 세로 값 $width, height$ 를 가진다.

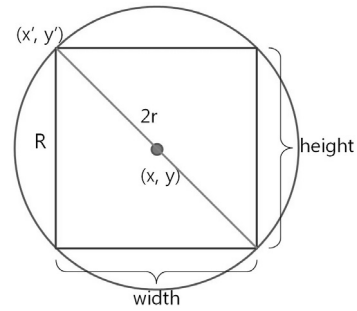


Fig. 16. Inscribing square

③ 단계에서는 Fig. 17과 같이 정사각형을 수평으로 등분하여 아래쪽의 직사각형을 선택하며, ④ 단계에서 이 직사각형 내부에 위치하는 픽셀들의 평균값의 집합 $S_{avg}(I_{R'_i})$ 을 구한다. 이렇게 하는 이유는 Fig. 16과 같이 정사각형에서의 평균값을 구하게 될 경우 눈꺼풀에 의하여 평균값이 커지게 되어 눈동자의 검출율이 떨어지므로 Fig. 17과 같이 직사각형의 영역을 사용한다.

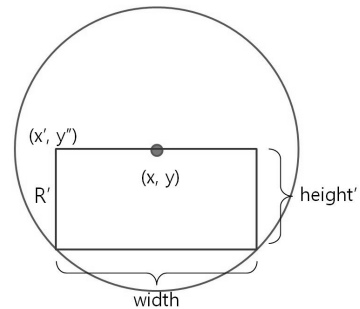


Fig. 17. Division square

또한 눈동자 영역의 평균 픽셀값은 Fig. 15와 같이 20이하임을 알 수 있으므로, ⑤ 단계에서 임계값 T 를 20으로 지정하여 만족하지 않는 후보위치를 제외하며, ⑥ 단계에서 남은 후보 위치들의 개수가 1개이면 후보에 해당되는 원 C 를 눈동자의 위치로 선택한다. 이 때 눈동자의 후보 위치가 2개 이상일 경우 가장 낮은 평균값을 나타내는 후보를 눈동자의 위치로 하여 Fig. 18과 같이 찾을 수 있게 된다.

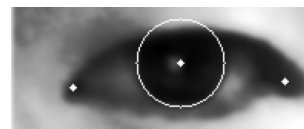
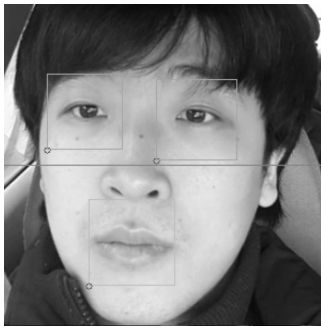


Fig. 18. Decision of iris position in candidates

4. 실험결과

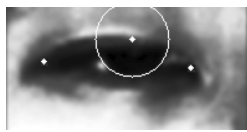
제안한 방법은 밝은 조명상태와 어두운 조명상태에서 실험하였다. Fig. 19는 밝은 조명 상태에서 눈동자를 검출한 결과이고 Fig. 20은 어두운 조명 상태에서 검출한 결과이다.



(A) Face detection and eye detection

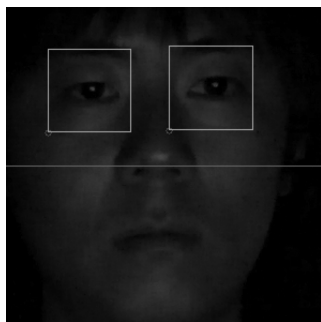


(B) Iris detection of right eye

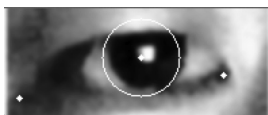


(C) Iris detection of left eye

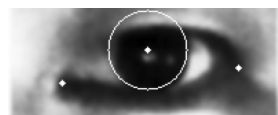
Fig. 19. Iris detection in the daytime



(A) Face detection and eye detection



(B) Iris detection of right eye



(C) Iris detection of left eye

Fig. 20. Iris detection in the nighttime

본 논문에서 제안하는 방법의 구현은 편의를 위해 일반적으로 사용되고 있는 OpenCV를 이용하여 구현하였으며, 처리 환경은 1.6GHz Quad-Core, 2GB RAM의 안드로이드 기반 스마트폰에서 640*480 크기의 영상을 사용하였다.

아래 Table 2는 조명의 상태가 밝을 때와 어두울 때 100 프레임 길이의 정면만 응시하고 있는 눈동자 영상에 대해 기존의 Kalus Toennies 방법[7]과 Retno Supriyanti 방법[6] 및 본 논문에서 제안하는 방법으로 눈동자를 검출하였을 때의 검출 속도와 검출율을 나타낸다. Kalus Toennies가 제안한 방법은 일반적인 실내 조명에 대해서만 고려하였기 때문에 어두운 조명에서는 눈동자를 검출할 수 없었다.

한편 제안한 방법에서도 아래 Fig. 21과 같이 강한 조명 환경이나 노이즈에 의해 눈의 양쪽 끝점을 올바르게 추정할 수 없는 경우에는 눈동자의 크기 범위를 잘못 추정하므로 정확한 눈동자 검출이 어려웠다.

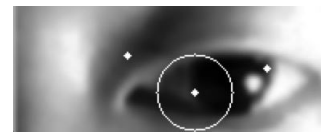


Fig. 21. False detection of iris

5. 결론 및 향후 연구방향

본 논문에서는 haar-like feature를 이용하여 얼굴 및 눈을 검출하고, 눈 영역의 크기 정규화 및 양쪽 눈 끝점 검출을 통해 눈동자의 크기 범위를 추정함으로써 허프 원 변환에서의 계산량을 줄여 눈동자 검출에 대한 성능을 향상시켰다. 또한 히스토그램 평활화를 이용하여 다양한 조명환경에서도 눈동자 검출이 가능하도록 하였다. 하지만 강한 조명에 의해 눈의 특징점이 소실되거나, 노이즈 등의 이유로 눈의 양쪽 끝점이 올바르게 검출되지 않는 경우에는 눈동자 크기 범위가 잘못 추정될 수 있다는 문제점을 안고 있다. 향후 연구에서는 조명에 견고한 정확한 눈의 양쪽 끝점을 추출하는 방법에 대하여 연구를 진행할 예정이다.

참고 문헌

[1] HwangSoo Jeon, ETRI, Industry Environment Research Team, Senior Research Engineer, "Intelligent Vehicle Safety System Development Trend" National IT Industry Promotion Agency, 2012. 8.
 [2] Korea Electronics Technology Institute, "Vehicle Safety System Industry Trend" 2010. 12.

Table 2. Result of process speed and detection rate

		existing method 1 (Kalus Toennies)	existing method 2 (Retno Supriyanti)	proposing method
bright illumination	average of process speed per second	8 frames	15 frames	21 frames
	average of detection rate	81%	84%	92%
dark illumination	average of process speed per second	-	14 frames	20 frames
	average of detection rate	-	71%	90%
require camera parameter		O	X	X

[3] Bernhard Froba and Andreas Ernst, "Face Detection with the Modified Census Transform", IEEE International Conference on Automatic Face and Gesture Recognition, pp.91-96, 2004.

[4] Paul Viola and Michael Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features", in Computer Vision and Pattern Recognition, pp.511-518, 2001.

[5] Jianguo Li, Tao Wang and Yimin Zhang, "Face Detection using SURF Cascade", IEEE International Conference on Computer Vision Workshops, pp.2183-2190, 2011.

[6] Retno Supriyanti and Budi Setiawan, "Detecting Pupil and Iris under Uncontrolled Illumination using Fixed-Hough Circle Transform", International Journal of Signal Processing, Image Processing and Pattern Recognition, Vol.5, No.4, pp.175-188, 2012.

[7] Klaus Toennies, "Feasibility of Hough-Transform-Based Iris Localisation for Real-Time-Application", IEEE, Pattern Recognition, pp.1053-1056, 2002.

[8] Zinn Walter and Solomon Herbert, "Eye Care, Eye Glasses and Contact Lenses" City: Lifetime Books, 1965.

[9] John Daugman, "How iris recognition works", Proceedings of 2002 International Conference on Image Processing, Vol.1, pp.21-30, 2002.

[10] Yoav Freund and Robert E. Schapire, "Experiments with a New Boosting Algorithm", In Proc. 13th Int. Conf. On Machine Learning, pp.148-156, 1996.

[11] Carolyn Kimme, Dana Ballard and Jack Sklansky, "Finding Circles by an Array of Accumulators", Communication of the ACM, Vol.18, No.2, pp.120-122, 1975.



김 성 훈

e-mail : limiteddaily@gmail.com

2013년 가천대학교 컴퓨터미디어학과
(학사)

2013년~현 재 가천대학교 전자계산학과
석사과정

관심분야: 영상처리, 컴퓨터비전 등



한 기 태

e-mail : gthan@gachon.ac.kr

1982년 충남대학교 계산통계학과(학사)

1990년 한양대학교 전자계산학과
(공학석사)

2001년 한양대학교 전자계산학과
(공학박사)

2009년~2010년 University of Texas at Austin, Researching
professor

2010년~현 재 가천대학교 IT대학 컴퓨터미디어융합학과 교수
관심분야: 컴퓨터비전, 영상처리, 스마트객체 응용기술, 모바일
컨텐츠 등