

## 적외선 조명 카메라를 이용한 동공 마우스 시스템

김충범\*, 김성훈\*\*, 한수환\*

\*동의대학교 멀티미디어공학과

\*\*경북대학교 소프트웨어공학과

e-mail: toonme@hanmail.net, shkim1454@knu.ac.kr, swhan@deu.ac.kr

### An Eye Mouse System Using an Infrared Illumination Camera

Choong-Bum Kim\*, Seong-Hoon Kim\*\*, Soowhan Han\*

\*Dept. of Multimedia Engineering, Dong-Eui University

\*\* Dept. of Software Engineering, Kyungpook National University

#### 요 약

장애인이 지속적으로 증가하면서 장애인들을 위한 복지 시설들 또한 증가하고 있다. 그러나 장애인들이 컴퓨터를 이용하는 데에는 아직까지 많은 불편함을 겪고 있으며, 그 원인으로는 장애인들이 컴퓨터를 보다 편리하게 사용할 수 있는 장치들이 부족하다는 것을 들 수 있다. 따라서 이 논문에서는 적외선 조명 카메라를 이용하여 장애인들이 눈동자와 눈꺼풀의 움직임을 마우스를 제어할 수 있는 동공 마우스 시스템을 제안하고 실험을 통하여 그 가능성을 보였다. 제안한 시스템에서는 적외선 조명 카메라, Canny 알고리즘과 퍼지 추론 기법이 이용되었으며, 눈이 포함된 영상에서 Canny 알고리즘으로 동공을 검출하여 마우스 위치를 추정하고 퍼지 추론 기법으로 눈 깜박임을 추정하여 마우스 포인터 이동 및 클릭 기능이 구현되었다. 제안된 기법들을 이용하여 실험하고 그 결과를 분석하였다.

키워드 : Canny 알고리즘; 안구 마우스; 퍼지 추론 기법; 동공 검출

#### I. 서론

한국보건사회연구원의 <2005 장애인 실태조사>에서 우리나라 장애인의 수를 총 215만 명으로 추정하였다[1]. 2000년에 비해 장애인은 70만 명 정도 증가하였고 2008년인 지금은 장애인의 수가 지속적으로 늘어나고 있는 추세이다. 실태 조사에 의하면 우리나라의 장애인 수는 매년 약 1,000명씩 늘어나고 있으며 우리나라 장애인 출현율은 다른 나라에 비하여 상대적으로 높지 않은 것으로 볼 수 있다. 그러한 이유로 우리나라의 장애인 복지시설은 상당히 낮은 수준에 머무르고 있다.

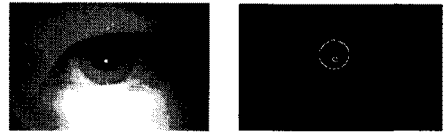
또한 이 실태 조사에 따르면 장애인 2명중 1명이 인터넷을 사용하고 있으며, 특히 발달장애인 4명중 3명이 인터넷을 사용하고 있다고 발표되었다. 하지만 몸을 가누기 힘든 장애인들은 컴퓨터를 이용하고 싶어도 하지 못하고 있는 실정이다. 현재 미국, 일본 등과 같은 선진국에서는 이 논문에서 제

안하는 시스템과 유사한 제품들이 이미 상품화되어 시장성이 높다고 한다. 그러나 우리나라에서는 수요의 문제로 인하여 아직 적극적으로 개발되고 있지 않다.

눈동자를 이용한 마우스 제어와 관련한 시선 추적 등을 다룬 논문들이 현재 많이 발표되고 있다[2][3]. 하지만 이 논문에서 제안하고자 하는 것과 같은 동공 마우스 시스템에 대해서는 아직 많은 연구가 이루어지지 않고 있다. 이 논문에서 제안하는 적외선 조명 카메라를 이용한 동공 검출 기법과 마우스의 클릭 기능을 구현하기 위한 퍼지 추론 기법은 이미 부분적으로 알려진 기법들이다. 이 논문에서는 이러한 알려진 기법들 중의 하나로 윤곽선 검출을 위한 Canny 알고리즘을 이용해서 동공을 검출한다. 그리고 눈의 깜박임으로 마우스 버튼 클릭의 기능을 구현하기 위해서 퍼지 추론 기능을 이용하였다[4].

이 논문에서는 기존의 기법들과 다른 점으로 적외선 조명 카메라를 컴퓨터에 부착시키는 것이 아니라 머리에 부착시킨

다. 카메라를 눈의 부근에 부착시켜 눈만을 영상으로 입력 받음으로써 시스템 구현상 많은 부분이 간소화 된다. 일반적으로 컴퓨터에 설치된 카메라로부터 동공을 찾기 위해서는 얼굴을 인식한 후에 눈을 찾고 찾은 눈에서 동공을 추출해야 된다. 이 논문에서는 그러한 과정이 필요 없도록 카메라의 위치를 변경함으로써 시스템의 속도와 성능을 향상시키고자 하였다.



〈그림 2〉 소벨 연산자를 통해 이진화 된 영상

## II. 동공 검출

적외선 조명 카메라를 이용하게 되면 외부 가시광선의 영향을 줄이고, 멜라닌 색소의 양에 의해 결정되는 사용자의 홍채 색상에 관계없이 뚜렷한 동공 경계의 눈 영상을 취득할 수 있다. 동공 검출에 대해서는 홍채 인식으로 인해 많은 논문과 자료들이 발표되어 있다. 이 논문에서는 기존의 다양한 방법들 중에서 Canny 알고리즘과 템플릿 매칭 알고리즘을 사용하였다.

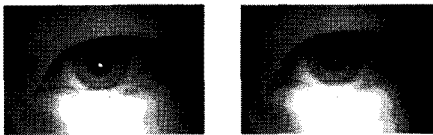
### 2-1. Canny 알고리즘을 이용한 동공 검출

Canny 필터(5)는 경계(edge) 검출 필터의 일종으로 가우시안(Gaussian) 필터를 이용하여 영상을 평활화 한 후, 소벨(Sobel) 연산자를 사용하여 경계를 검출하는 과정으로 이루어 진다(2).

입력받은 영상  $I(x,y)$ 는 동공 경계 오차를 줄이기 위해 식(1)과 식(2)에 의해 그림 1과 같이 가우시안 필터링이 수행된다.

$$G(x,y) = \sigma^2 e^{-\frac{x^2+y^2}{2\sigma^2}} \dots\dots\dots (1)$$

$$I_G(x,y) = G(x,y) \times I(x,y) \dots\dots\dots (2)$$



〈그림 1〉 가우시안 필터링 된 입력 영상

가우시안 필터링 된 영상은 식(3)과 식(4)의 소벨 연산자에 의해 윤곽선이 검출되고, 검출된 영상은 임계값을 통해 이진화된다.

$$S_x = I[i-1][j+1] + I[i][j+1] + I[i+1][j+1] - I[i-1][j-1] - I[i][j-1] - I[i+1][j-1] \dots\dots (3)$$

$$S_y = I[i+1][j+1] + I[i+1][j] + I[i+1][j-1] - I[i-1][j+1] - I[i-1][j] - I[i-1][j-1] \dots\dots (4)$$

그런 다음 그림 2와 같이 소벨 연산자에 의해 윤곽선이 검출 된다. 그 검출된 윤곽선에 의해서 동공의 중심과 반지름을 구할 수 있다. 중심과 반지름을 구하기 위해서 전체 탐색 알고리즘을 이용한다. 전체 탐색 알고리즘은 입력받은 영상의 픽셀을 전부 탐색하여 특징점을 찾는 것이다. 위의 검출된 영상에서 전체 탐색 알고리즘을 이용하여 검출하여 표시를 하게 되면 그림 3과 같은 결과를 얻을 수 있다.



〈그림 3〉 전체 탐색 알고리즘을 이용한 동공 검출

하지만 전체 탐색 알고리즘을 이용하게 되면 탐색 시간이 오래 걸린다. 따라서 이 논문에서는 4방향으로 검색하는 방식을 이용하여 특징점 4개만을 구한 결과를 얻음으로써 전체 탐색 알고리즘을 사용할 때 보다 빠른 검출 결과를 얻을 수 있었다.

### 2-2. 템플릿 매칭 알고리즘을 이용한 동공 검출

템플릿 매칭 알고리즘에는 MAD 방법, MSE 방법과 NCC 방법 등의 3가지 방법이 있다(6). 그 중에서 가장 성능이 우수하다고 알려진 NCC 방법을 이용하였다. MAD 방법과 MSE 방법은 대응되는 픽셀의 밝기 값을 서로 빼서 차이 값을 더하는 알고리즘이다. 그렇게 되면 밝기 값의 차이가 크게 되면 오차가 심하게 된다. NCC 방법은 각각의 밝기 값을 정규화 해서 계산하는 알고리즘으로 MAD 방법과 MSE 방법에서 발생하는 오차를 줄인 알고리즘이다. NCC 방법은 식(5)와 식(6)을 이용한다. 템플릿 매칭을 이용하여 동공을 검출한 결과는 그림 4와 같다.

$$r = \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{a}| \cdot |\mathbf{b}|} = \frac{\sum_{i=1}^M \sum_{j=1}^N a(i,j)b(i,j)}{\sqrt{\sum_{i=1}^M \sum_{j=1}^N a(i,j)^2} \sqrt{\sum_{i=1}^M \sum_{j=1}^N b(i,j)^2}} \dots\dots\dots (5)$$

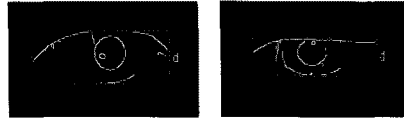
$$a(i,j) = g(i,j) - m, \quad b(i,j) = t(i,j) - i$$

$$r = \frac{(MN) \sum g t - (\sum g) \sum t}{\sqrt{[(MN) \sum g^2 - (\sum g)^2][(MN) \sum t^2 - (\sum t)^2]}} \dots\dots (6)$$



〈그림 4〉 탭플릿 이미지를 이용한 동공 검출

으로 찾게 된다. 그림 6과 같이 눈의 영역을 구한 후 눈의 높이를  $d$ 로 둔다.



〈그림 6〉 검출된 눈의 영역

### III. 마우스 기능 구현

마우스의 기능은 클릭, 더블클릭, 포인터의 움직임, 드래그 등이 있다. 이러한 마우스 기능을 눈으로 표현하기란 쉽지 않다. 눈으로 마우스의 기능을 모두 표현하기 위해서는 많은 제약 요소가 따른다. 그래서 클릭과 마우스 포인터의 위치 제어만 구현하였다.

클릭을 제어하기 위해서는 눈의 깜박임으로는 제어를 하기가 힘들다. 사람은 눈을 수시로 깜박이게 된다. 그리고 눈이 피로하게 되면 깜박임의 횟수가 증가 하는 것에서도 문제가 생기게 된다. 그래서 클릭을 정확하게 제어하기 위해서 깜박임 시간( $t$ )을 고려하였다. 거리( $d$ )와 시간( $t$ )을 이용하여 클릭을 추정하게 된다. 클릭 여부를 결정하기 위한 규칙은 표 3과 같으며 각 입력 요소와 출력 요소에 대한 퍼지 멤버십 함수는 그림 7, 8, 9와 같다.

#### 3-1. 동공의 움직임과 스크린 좌표

동공 마우스 시스템이라고 하게 되면 동공의 움직임이 마우스 포인터 이동 요소가 될 것이다. 눈이 보는 시선에 의해서 마우스 포인터를 이동하게 하는 것이 이 시스템에서 상당히 중요한 부분이다. 정확하게 시선의 위치에 따라 마우스 포인터를 이동 시킨다는 것은 아주 어려운 일이다. 그림 5와 같이 스크린의 모서리 부분에 4방향의 시선을 맞추게 된다. 그렇게 되면 시선의 범위를 구할 수 있다. 이 범위로 비례식을 이용해서 마우스 포인터 이동이 구현 된다. 식(7), 식(8), 식(9), 식(10)를 통해서 포인터의 좌표를 구할 수 있다.

〈표 3〉 거리와 시간을 이용한 경험적 퍼지 규칙

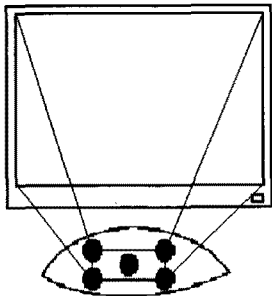
$t \backslash d$	ZE	S	M	L
VL	C	M	O	O
L	C	C	M	O
M	C	C	M	M
H	C	C	C	C
VH	C	C	C	C

$$W: w = x : 1 \dots\dots\dots (7)$$

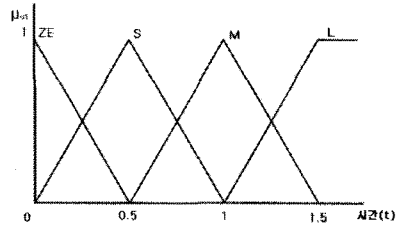
$$x = \frac{W}{w} \dots\dots\dots (8)$$

$$H: h = y : 1 \dots\dots\dots (9)$$

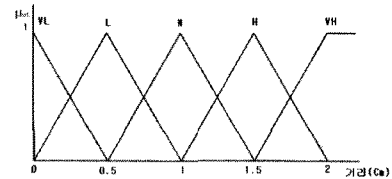
$$y = \frac{H}{h} \dots\dots\dots (10)$$



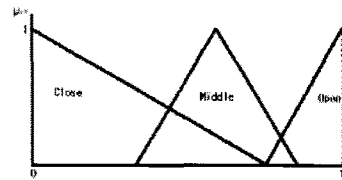
〈그림 5〉 눈 시선과 스크린의 좌표



〈그림 7〉 시간( $t$ )에 대한 퍼지 함수



〈그림 8〉 거리( $d$ )에 대한 퍼지 함수



〈그림 9〉 출력 퍼지 함수

#### 3-2. 경험적 퍼지 추론 규칙을 이용한 클릭 제어















눈의 깜박임으로 클릭을 제어 하려고 한다. 눈의 깜박임을 측정하기 위해서는 먼저 눈을 찾아야 한다. 눈을 찾는 방법은 2에서 언급한 것과 같이 Canny 알고리즘을 이용해서 윤곽

#### IV. 실험 및 결과 고찰

동공 검출 실험을 위하여 10명으로부터 각 20장씩 획득된 전체 200장의 영상을 사용하였다. 또한 실시간 처리 여부를 확인하기 위하여 1분 분량의 동영상 4개를 사용하였다. 이상의 실험 영상을 이용하여 Canny 알고리즘과 템플릿 매칭 알고리즘을 각각 사용하여 그 결과를 비교하였다.

그림 10에 나타나 있듯이 템플릿 매칭보다는 Canny 알고리즘의 인식률이 우수하게 나왔다. 이 논문에서는 템플릿의 크기를 고정하였기 때문에 밝기 값을 정규화하여도 눈동자 크기 차이에 따른 오인식이 많이 발생하였다.

마우스의 포인터 위치 제어를 위해 앞에서 설명한 방식으로 구현한 후 마우스 포인터를 움직여 보았다. 비례 방식으로 계산된 좌표로 인해 동공의 미세한 움직임에서도 스크린에서의 마우스 포인터는 움직임이 크게 변하는 현상이 나타났다. 이로 인해 마우스 포인터의 세밀한 움직임을 구현 하는 것이 쉽지 않은 문제임을 파악할 수 있었다. 따라서 마우스 포인터를 더욱 정밀하게 움직일 수 있도록 하기 위해서는 추가 연구가 필요할 것으로 생각한다.

Canny	템플릿 매칭
	
	
	
	
	
	
	
인식률 : 87.5 %	인식률 : 50 %

〈그림 10〉 동공 검출 결과 비교

#### V. 결론

몸이 불편한 장애인들에게도 컴퓨터를 쉽게 사용할 수 있게 하기 위해서 동공 마우스 시스템에 대해 제안하고 실험을 통해 그 성능을 분석하였다. 그 결과 제안한 기법들을 활용한

동공 마우스 시스템에 대한 가능성을 확인할 수 있었다.

그러나 이 논문에서는 마우스 포인터의 제어량을 추정하는 기법에서 눈과 스크린의 거리에 따라 실제 화면의 좌표와 동공의 위치가 차이나는 문제를 반영하고 있지 않기 때문에 마우스 포인터 위치의 정밀한 제어에 어려움이 많이 발생하였다. 따라서 눈과 스크린의 거리에 따른 이러한 문제가 발생하지 않는 시스템을 제안할 수 있는 기법에 대해서는 추가적으로 많은 연구가 필요할 것으로 생각한다.

#### References

- [1] <http://www.kihasa.re.kr/html/jsp/>
- [2] 박강령, "얼굴 및 눈동자 움직임에 의한 시선 위치 추적," 전자공학회논문지 제41권 SP편, 제2호, pp. 79~8, 2004.
- [3] 원정우, "홍채인식을 위한 개선된 전처리와 최적 웨이블릿 밴드 추출 알고리즘," 홍익대학교 석사학위 논문, 2003.
- [4] Timothy J. Ross, "Fuzzy Logic with Engineering Applications(2nd ed.)", Wiley, 2004.
- [5] Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing, 3/E, 2007.
- [6] 이찬희, 신상호, 우영운, 장경식, "얼굴 영상에서 유전자 알고리즘을 이용한 눈동자 검출," 한국해양정보통신학회 추계학술대회논문집, pp.444-447, 2008.