

# Robust pupil detection and gaze tracking under occlusion of eyes

Gyung-Ju Lee\*, Jin-Suh Kim\*\*, Gye-Young Kim\*\*\*

## Abstract

The size of a display is large, The form becoming various of that do not apply to previous methods of gaze tracking and if setup gaze-track-camera above display, can solve the problem of size or height of display. However, This method can not use of infrared illumination information of reflected cornea using previous methods. In this paper, Robust pupil detecting method for eye's occlusion, corner point of inner eye and center of pupil, and using the face pose information proposes a method for calculating the simply position of the gaze. In the proposed method, capture the frame for gaze tracking that according to position of person transform camera mode of wide or narrow angle. If detect the face exist in field of view(FOV) in wide mode of camera, transform narrow mode of camera calculating position of face. The frame captured in narrow mode of camera include gaze direction information of person in long distance. The method for calculating the gaze direction consist of face pose estimation and gaze direction calculating step. Face pose estimation is estimated by mapping between feature point of detected face and 3D model. To calculate gaze direction the first, perform ellipse detect using splitting from iris edge information of pupil and if occlusion of pupil, estimate position of pupil with deformable template. Then using center of pupil and corner point of inner eye, face pose information calculate gaze position at display. In the experiment, proposed gaze tracking algorithm in this paper solve the constraints that form of a display, to calculate effectively gaze direction of person in the long distance using single camera, demonstrate in experiments by distance.

▶ Keyword : Pupil center detection, Deformable template, Head pose estimation, Gaze tracking, Gaze calibration

## 1. Introduction

디스플레이가 대형화되면서 지하철이나 상점, 전시장 등의 공공장소에서 활용이 높아지고 있다. 과거에는 사용자가 디스플레이에서 제공하는 정보를 받기만 했다면, 현재는 사용자가 디스플레이에 나타나는 정보를 조작하고 그에 따른 결과를 받아보고 있다. 이처럼 디스플레이의 활용이 다양해지면서 디스플레이와 사람간의 상호작용도 중요한 연구 분야가 되었다. 아

직까지는 터치센서를 이용하여 사용자가 직접 디스플레이에 접촉하는 방식이 많이 사용되고 있지만, 체스처나 음성인식, 시선 추적 등을 적용하려는 시도가 늘어나고 있다[1-2]. 시선 추적은 눈동자의 움직임을 감지하여 시선의 위치를 파악하는 것으로 대부분 카메라로 획득한 영상을 통해 동공의 움직임과, 각막에 반사된 적외선 조명의 위치로 계산된다. 카메라와 적외선 조명이 사용자 얼굴보다 상단에 위치한 경우에는 눈꺼풀이나 속눈썹에 의해 눈동자 폐색이 발생하고, 얼굴 각도에 따라서 적외선 조명이 눈동자에 반사되지 않는 경우가 발생한다. 이와 같은

• First Author: Gyung-Ju Lee, Corresponding Author: Gye-Young Kim

\*Gyung-Ju Lee (lkj0917@ssu.ac.kr), Dept. of Computer Science and Engineering, Soongsil University

\*\*Jin-Suh Kim (kimjinsuh@ssu.ac.kr), Dept. of Computer Science and Engineering, Soongsil University

\*\*\*Gye-Young kim (gykim11@ssu.ac.kr), School of Software, Soongsil University

• Received: 2016. 09. 12, Revised: 2016. 09. 24, Accepted: 2016. 10. 09.

• This work was supported by the Global Leading Technology Program of the Office of Strategic R&D Planning (OSP) funded by the Ministry of Trade, Industry & Energy, Republic of Korea. (10042418)

이유로 대부분 시선추적 시스템은 사용자 얼굴의 정면 또는 하단에 위치하며 얼굴을 향한다. D. C. Cho 등은 60인치 TV 하단에 시선 추적기가 설치되고 이로부터 사용자가 1.4~3m 구간에 존재할 때 각막에 반사된 적외선 조명의 위치를 이용하여 시선의 방향을 계산하는 방법을 제안하였다[3]. 이 방법은 두 개의 광각 카메라와 팬-틸트 모듈이 장착된 협각 카메라, 그리고 두 개의 적외선 조명으로 구성되었고, 협각 카메라는 원거리에 위치한 사용자의 눈 영상을 얻기 위해 사용되었다. 이 방법은 0.8°~1.4°사이의 시선 추적 정확도를 보였다. H. C. Lee 등은 대형 디스플레이에 적용 가능한 시선 추적시스템을 제안하였다[4]. 광각 카메라를 이용하여 2m 이내에 존재하는 사용자의 위치를 파악하고, 협각 카메라를 이용하여 고품질의 눈 영역 영상을 획득하였다. 또한 디스플레이의 각 모서리에 적외선 조명을 설치하여 동공에 반사된 4개의 조명 반사점과의 매핑을 통해 시선의 위치를 계산하였고 이는 평균  $\pm 0.737^\circ$ 의 정확도를 보였다. W. O. Lee 등은 적외선 조명을 사용하지 않고 줌렌즈를 포함한 가시광 카메라를 이용하여 스마트 TV에 적용하기 위한 시선추적 방법을 제안했다[5]. 이 방법은 사용자 시선보정 절차에서 5개의 보정점을 사용하며 사용자가 각 보정점을 응시하는 동안 얼굴의 특징들을 저장하고, 이후에 얼굴의 특징 변화에 따른 시선의 위치를 예측한다. 카메라로부터 2m 떨어진 거리에서 디스플레이의 화면을 3x3으로 분할하여 시선추적 성능을 측정하였고, 90.5%의 정확도를 보였다. C. Hennessey 등은 대형 스마트 TV의 제어를 위해 TV 하단에 광각 카메라와 협각 카메라로 구성하였고, 광각 카메라는 마이크로소프트사의 키넥트를 이용하여 사용자의 위치를 탐지하였으며, 협각 카메라는 적외선 카메라에 팬-틸트 모듈을 장착하여 사용자의 얼굴을 추적하였다[6]. 이는 1°~2°이내의 시선 정확도를 보이고, 이 방법도 역시 각막에 반사된 적외선 조명은 시선의 위치를 파악하는데 중요한 역할을 한다. 그러나 대화면 디스플레이가 설치된 위치나 세로형 디스플레이에서는 시선추적기가 하단에 설치하기 어렵고, 이와 같은 경우 적외선 조명을 이용한 시선추적 방법은 적용하기 어렵다. 또한 적외선 조명을 사용하지 않더라도 시선추적의 정확성을 높일 필요가 있다.

본 논문에서는 디스플레이 형태에 따른 제약사항을 없애기 위한 시선추적 방법을 제시한다. 대형 디스플레이의 경우, 사용자는 디스플레이 전체를 응시하기 위해 디스플레이로부터 거리가 멀고, 화면의 어느 위치에 서 있을지 모르기 때문에 시선 추적을 위해 팬-틸트-줌(PTZ: Pan-Tilt-Zoom)카메라를 사용한다. 그림 1은 본 논문에서 제안하는 시선 추적 방법의 흐름도를 나타낸 것이다. 시선 방향은 눈의 안쪽 코너점과 동공의 중심 그리고 얼굴 포즈 정보를 이용하여 계산된다. 먼저, 카메라를 광각모드로 설정하여 카메라 FOV(Field of View) 내에 얼굴을 탐지한다. 얼굴이 검출되면 위치를 계산한 후에 카메라의 팬-틸트-줌을 제어하여 카메라를 협각모드로 전환한다. 카메라가 협각모드로 전환되면 얼굴은 계속 추적이 되고, 얼굴의 눈과 코, 턱과 같은 특징점들을 검출한다. 그런 다음 눈 영역에서

지와 가변적인 템플릿을 사용하여 동공의 중심을 검출하고, 얼굴의 특징점들과 3차원 얼굴 모델의 매칭을 통해 얼굴 포즈 정보를 추정한 후에 정보를 조합하여 시선의 방향을 계산한다. 본 논문의 구성은 다음과 같다. 2장에서는 MCT특징을 이용한 얼굴 검출과 얼굴 위치에 따라 팬-틸트-줌 카메라를 제어하는 방법에 대해 기술한다. 3장과 4장에서는 회귀 트리를 이용한 얼굴의 특징점 검출과 3차원 얼굴 모델과의 매칭을 통한 얼굴 포즈 추정 방법을 제안하고, 타원 정보와 가변 템플릿을 이용한 동공 중심 검출 및 시선 방향을 산출하는 방법을 다룬다. 5장에서는 실험결과를 보이며, 6장에서는 결론 및 향후 연구방향을 제시한다.

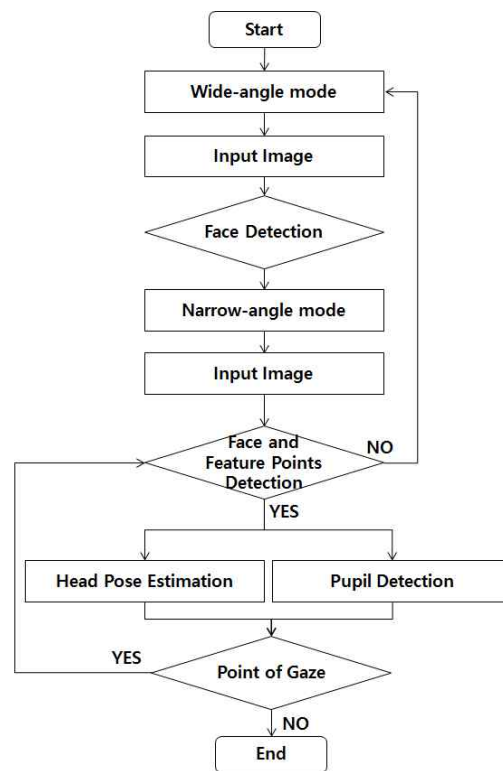


Fig. 1. Flow chart of gaze tracking

## II. Face detection and PTZ Camera control through MCT Features

초기에 카메라를 광각모드로 설정하여 FOV 내에 얼굴을 탐지하고, 얼굴의 위치에 따라 카메라의 팬-틸트-줌을 제어하여 협각모드로 전환한다. 영상에서 얼굴을 검출하기 위해 조명 변화에 강건한 MCT(Modified Census Transform)와 AdaBoost를 이용하여 얼굴 분류기 생성 및 검출을 수행한다[7-8]. 식 1과 같은 MCT는 지역 구조적 특징(local structure feature)으로 3x3 커널에서 {0, 1}의 이진 정보를 가진다. 이는 2<sup>9-1</sup>로

511개의 특징이 생성되고, 지역적 정보를 이용하기 때문에 조 명 변화에 강인하고, 계산이 간단하여 얼굴 검출에 높은 검출율 과 빠른 수행시간을 보인다.

$$\Gamma(x) = \otimes_{y \in n'} \zeta(\tilde{I}(x), I(y)) \quad (1)$$

여기서,  $x$ 는 커널의 중심,  $\Gamma(x)$ 는 변환된 MCT 값,  $n'$ 는 커널 중심  $x$ 와,  $x$ 의 이웃 픽셀을 나타낸다.  $I(x)$ 는  $3 \times 3$  커널 영역 내 의 평균값이고,  $\zeta(I(x), I(y))$ 는 비교함수로 만약  $I(x) < I(y)$ 라면 1, 그렇지 않으면 0을 반환한다. 입력 영상 에서 얼굴을 검출하기 위해서는 MCT를 이용하여 특징을 추출 한 후에, AdaBoost로 생성한 얼굴 분류기를 이용하여 얼굴을 판별한다. 분류기를 생성하기 위해 얼굴 영상과 얼굴이 포함하 지 않은 배경 영상을 입력으로 하며, 얼굴과 배경 영상의 분류 오류율이 50%보다 낮은 약한 분류기(weak classifier)들을 하 나의 강한 분류기(strong classifier) 생성한다. 강한 분류기는 아래와 같이 표현한다.

$$H(\Gamma) = \sum_X h_X(\Gamma(X))$$

$$h_X(r) = \sum_{t=1}^T \alpha_t w_t(r) I(X = X_t) \quad (2)$$

여기서,  $h$ 는 픽셀 분류기(pixel classifier)이고,  $r$ 는 511개 MCT특징의 인덱스이며,  $X = (x, y)$ 이다. 즉, 영상의  $(x, y)$ 에 해당하는 MCT 특징 값이 픽셀 분류기에 입력되면, 이 값을 약 한 분류기  $w$ 와 이에 해당하는 가중치  $\alpha$ 에 의해 픽셀은 얼굴 또는 배경으로 분류가 된다. 모든  $(x, y)$ 에 대한 픽셀 분류기를 하나로 구성하면 최종의 강한 분류기  $H$ 가 생성된다. 분류기를 생성하는데 사용되는 얼굴 학습데이터는 데이터의 분포가 클수 록 분류 오류율이 증가한다. 따라서 얼굴을 정면과 좌측면, 우 측면으로 분류하여 각각의 강한 분류기를 생성한 후에 그림 2 와 같이 캐스케이드(cascade)구조로 얼굴 검출을 수행한다. 입 력 영상에서 검출 윈도우 영역에 해당하는 MCT 특징이 입력되 면 정면 얼굴 분류기로 분류를 수행한다. 이때 얼굴로 분류되면 좌측면과 우측면 분류기는 수행하지 않고 검출 윈도우는 다음 위치로 이동하게 되며, 정면 얼굴 분류기에서 얼굴을 분류하지 못할 경우 좌측면 얼굴 분류기로 얼굴 분류를 진행한다.

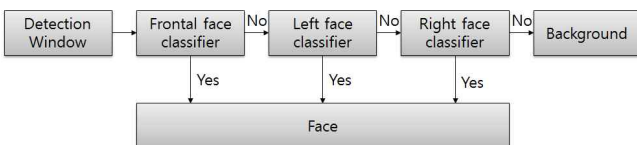


Fig. 2. Face detection process using a multi-classifier

3과 같다. 윈도우는 고정된 크기를 사용해 윈도우영역에 해당 하는 MCT 특징으로 분류를 진행한다. 이때 윈도우는 영상 전 체를 스캔하고, 영상은 검출 윈도우의 크기까지 다운샘플링을 진행하며 얼굴을 탐지한다. 광각 모드에서 얼굴을 검출하면 검 출된 위치 정보를 사용하여 카메라의 팬-틸트-줌을 제어한다. 식 3에서  $x$ 축의 회전각도 와  $y$ 축의 회전각도를 산출한다.

$$p_c = \frac{\theta_w}{W} \times u, \quad t_c = \frac{\theta_h}{H} \times v \quad (3)$$

여기서  $W$ 와  $H$ 는 영상의 가로와 세로의 화소 수이고,  $\theta_w$ 와  $\theta_h$ 는 카메라의 수평과 수직의 화각이다.  $u = I_x - F_x$ 와  $v = I_y - F_y$ 는 영상 중심( $I_x, I_y$ )과 얼굴 중심( $F_x, F_y$ )의 차 이 값이다.  $p_c$ 와  $t_c$ 를 통해 카메라를  $x$ 축으로  $p_c$ 각도,  $y$ 축으로  $t_c$ 각도만큼 회전을 한다. 그런 다음 식 4로 카메라의 확대 인자 를 계산한다.

$$z_c = \frac{H}{O_h} \times w \quad (4)$$

여기에서  $O_h$ 는 제어할 얼굴의 크기이고,  $w$ 는 가중치이다. 식 3-4를 진행하면 카메라는 광각 모드에서 협각 모드로 전환하 며, 얼굴 특징정보를 손실 없이 획득할 수 있다.



Fig. 3. Process of face detection

### III. Facial feature detection and head pose estimation using Regression trees

검출된 얼굴 영역에서 얼굴의 특징점을 검출하기 위해 본 논 문에서는 회귀트리를 이용한 얼굴 정렬(face alignment)방법을 적용한다[9-11]. 우선, 얼굴의 특징점을 그림 4와 같이 68개 의 점으로 정의한다.

검출 윈도우로 영상을 스캔하며 얼굴을 검출하는 과정은 그림



Fig. 4. Defined facial features

얼굴 영역이 주어졌을 때, 얼굴 정렬의 목표는 초기 형태(shape)가 실제 형태와 근접하게 만드는 것이다. 얼굴 영상  $I_f$ 와 얼굴 형태의 초기값  $S_0$ 가 주어지면, 각각의 복귀자(regressor)는 형태 증가분  $\delta S$ 를 영상의 특징으로부터 계산하고 얼굴의 형태를 변경한다.

$$S^t = S^{t-1} + R^t(I_f, S^{t-1}), t = 1, \dots, T \quad (5)$$

여기서, t번째 약한 복귀자(weak regressor)  $R^t$ 는 이전 형태  $S^{t-1}$ 를 새로운 형태인  $S^t$ 로 갱신한다. 복귀자  $R$ 을 학습하기 위해 Random Fern 알고리즘을 사용한다. 이를 이용하여 학습하는 방법은 첫 번째 노드에서 얼굴 영상의 2개의 쌍의 점을 임의로 추출하여 추출된 점의 화소값을 비교하여 임의로 선택한 임계값의 크기와 비교한다. 그 다음 두 번째 노드에서는 다른 두 개의 쌍을 추출하여 크기를 비교한다. 계속해서 다섯 개 점의 쌍을 비교하여 5번째 노드까지의 임계값과 비교하여 최종 중단 노드에 도달하게 된다. 중단 노드에 도달한 영상을 이용해 변형된 형태를 구하게 된다. 그림 5는 68개의 얼굴 특징점을 검출한 결과 영상이다.

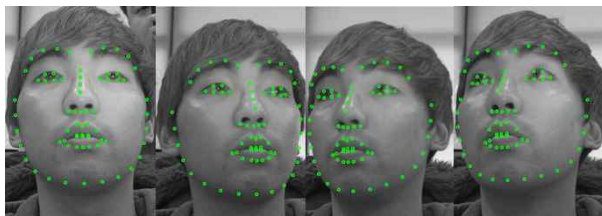


Fig. 5. The detected facial feature 68

68개의 얼굴 특징점을 검출한 후에 3차원 얼굴 모델과 매칭하여 3차원이 얼굴 포즈를 추정한다. 검출된 2차원 얼굴 특징점을 이용하여 3차원 얼굴 자세를 계산하기 위한 방법은 수학적 알고리즘과 최적화 알고리즘으로 개발되어왔다. 수학적 알고리즘은 계산을 반복적으로 함으로써 얼굴포즈를 추정하는 방법으로 속도가 빠르지만 잡음에 취약하다. 최적화 알고리즘의 포즈 추정 정확도는 초기 추측 값에 의존적이기 때문에 오류가 증가

할 수 있다. 따라서 얼굴포즈추정을 위해 수학적 알고리즘과 최적화 알고리즘을 통합하여 안정적인 계산과 속도가 빠른 POSIT(Pose from Orthography and Scaling with Iterations) 알고리즘을 이용한다. POSIT 알고리즘은 영상에서 추출한 특징점들과 3차원상의 실제 특징점들을 비교하여 2차원 특징점에 대응되는 3차원 객체의 자세  $R$ 과 위치  $T$ 를 추정한다. 이 알고리즘의 첫 번째 단계는 POS(Pose from Orthography and Scaling)과정이다. POS는 약한-투시 근사화(weak-perspective approximation)를 만족해야 한다. 이는 객체 상의 점들이 모두 동일한 깊이를 갖고, 크기 변화는 오로지 객체와 카메라 사이의 거리에 의해 발생한다고 가정하는 것이다. 만약 변형하는 객체의 경우 특징점을 원본 데이터로 정의할 수 없기 때문에 비교할 수 없다. 그래서 특징점의 위치의 변화가 크게 존재하는 변형 객체의 경우에는 이 알고리즘을 사용할 수 없다. 3차원의 자세를 계산하기 위해 최소한 4개 이상의 특징점이 영상 내에 존재해야 한다. 확대 또는 축소된 정사영의 투시 평면을 근사하여 선형 시스템의 해를 구하는 방법으로 객체의 회전행렬  $R$ 과 이동벡터  $T$ 를 구한다. 그 다음 단계는 "반복에 의한 POS(POS with Iteration)"이다. 3차원 객체의 자세와 위치 값이 수렴할 때 까지 반복을 수행하면 최종  $R$ 과  $T$ 를 추정할 수 있다.

$$R = \begin{bmatrix} R_{00} & R_{01} & R_{02} \\ R_{10} & R_{11} & R_{12} \\ R_{20} & R_{21} & R_{22} \end{bmatrix}, T = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (6)$$

3차원의 얼굴 포즈 각도는 그림 6과 같이 x축을 기준으로 상하로 회전하는 x축 회전각, y축을 기준으로 좌우로 회전하는 y축 회전각, 그리고 z축을 기준으로 얼굴의 기울어진 정도를 나타내는 z축 회전각으로 정의한다. 회전행렬  $R$ 과 식 7을 통해 3차원 얼굴 포즈 각도를 산출한다. 그림 7은 영상에서 검출한 얼굴 특징점과 POSIT 알고리즘을 통해 3차원 얼굴 포즈를 추정된 결과를 나타낸다.

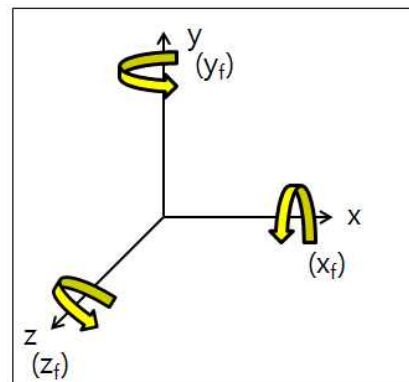


Fig. 6. Defined head pose

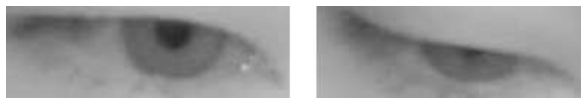
$$\begin{aligned} x_f &= \sin^{-1}(-R_{20}), \\ y_f &= \tan^{-1}\left(\frac{R_{21}}{R_{22}}\right), \\ z_f &= \tan^{-1}\left(\frac{R_{10}}{R_{00}}\right) \end{aligned} \quad (7)$$



Fig. 7. Head pose estimation using POSIT

#### IV. Pupil detection and gaze tracking with ellipse information and deformable template

앞 절에서 수행한 얼굴 특징점 검출 결과를 통해 눈 영역을 분리할 수 있고, 사용자의 시선에 따라 눈 영상은 그림 8과 같이 획득된다. 시선이 정면이나 카메라를 향하게 될 경우 그림 8의 a와 같고, 이 경우에는 홍채와 동공의 영역을 분리하고 타원 검출 알고리즘을 이용하여 동공의 영역 및 중심위치를 획득한다. 그러나 시선이 아래를 향하게 될 경우 그림 8의 b와 같이 동공이 눈꺼풀에 가려져 보이지 않는다. 이와 같은 경우에는 이동 및 변형이 가능한 템플릿을 이용하여 홍채를 검출하고, 동공의 위치를 추정한다.



(a) If gaze direction is toward the front  
(b) If gaze direction is toward the down

Fig. 8. Eye region images

그림 9는 동공 검출의 과정을 보여준다. 먼저, 그림 9의 a와 같이 홍채와 동공의 영역을 분리하기 위해 눈 영상에 대하여 Valley Map을 생성한다. 영상에서 낮은 값의 화소는 Valley Map에서 높은 값을 갖고, 영상에서 높은 값의 화소는 Valley Map에서 낮은 값을 갖는다. Valley Map은 식 8을 통해 생성된다.

$$V(x, y) = \frac{u(\mu - I(x, y)) \times |\mu - I(x, y)|}{\mu} \times 255$$

$$u(\mu - I(x, y)) = \begin{cases} 1, & \mu - I(x, y) \geq 0 \text{ 일 때} \\ 0, & \mu - I(x, y) < 0 \text{ 일 때} \end{cases} \quad (8)$$

식 8에서  $I(x, y)$ 는 입력 영상의 화소 값이고,  $\mu$ 는 입력 영상의 평균 화소 값이다. Valley Map을 생성하면 홍채영역을 설정한 후에 동공 영역을 분리한다. 동공 영역은 어두운 화소 값을 가지고 있기 때문에 Valley Map에서 높은 값을 가진다. 이를 이용하여 Valley Map 값의 상위 70%에 해당하는 영역의 중심을 홍채 영역의 중심위치로 설정하고, 홍채의 반지름은 식 9로 계산한다. 그림 9의 c는 식 9의 결과를 보여준다.

$$p_r = (\text{eye\_width} - 10) * 0.25 \quad (9)$$

그 다음 그림 9의 d와 e는 홍채영역에 대하여 Valley Map을 생성한 후에 가우시안 필터를 적용하여 잡음을 제거한 결과를 보여준다. 잡음이 제거된 Valley Map에서 에지를 검출하면 동공의 에지 뿐 아니라 눈꺼풀의 에지 정보도 같이 포함된 에지를 얻는다. 따라서 눈꺼풀에 해당하는 에지는 제거하고 남은 에지의 위치정보를 이용하여 타원을 검출한다[12]. 그림 9의 h는 홍채 및 동공을 검출한 결과 영상을 보여준다. 빨간색 원과 십자가는 홍채의 경계와 중심을 나타내고, 녹색 타원과 십자가는 동공의 경계와 중심을 나타낸다.

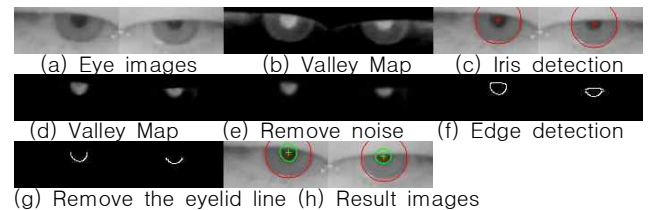


Fig. 9. Pupil center detection using the ellipse detection method

그림 9의 g에서 타원을 검출하기 위한 에지점의 개수가 임계값 이하일 경우에는 타원 검출이 제대로 수행되지 않는다. 이 경우에는 가변 템플릿을 이용하여 홍채를 검출 한 후에 동공의 중심을 추정한다. 템플릿은 그림 10처럼 원형 템플릿 주변에 에너지를 탐지하기 위한 윈도우가 존재하고, 이는 에너지의 크기에 따라 이동하거나 확대 및 축소를 한다[13]. 그러나 홍채의 크기는 식 9로 정해졌기 때문에 1~5번째 윈도우를 이용하여 이동을 수행한다.

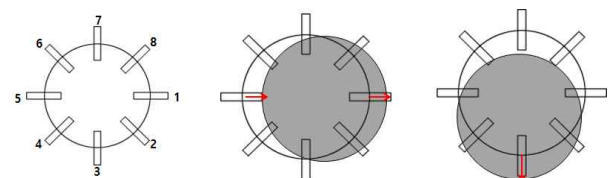


Fig. 10. Deformable template

가변 템플릿을 적용하기 위해 입력 영상에서 Valley Map을 생성하고, 이를 이진화한 영상에서 윈도우의 에너지는 식 10으로 계산한다. 가변 템플릿은 각 윈도우의 에너지가 0에 근사한 값을 가질 때 까지 이동하며 홍채의 영역을 탐색한다.

$$E_w = \frac{\sum(\text{each pixel value in the window})}{\text{number of pixels in the window}} - \frac{255}{2} \quad (10)$$

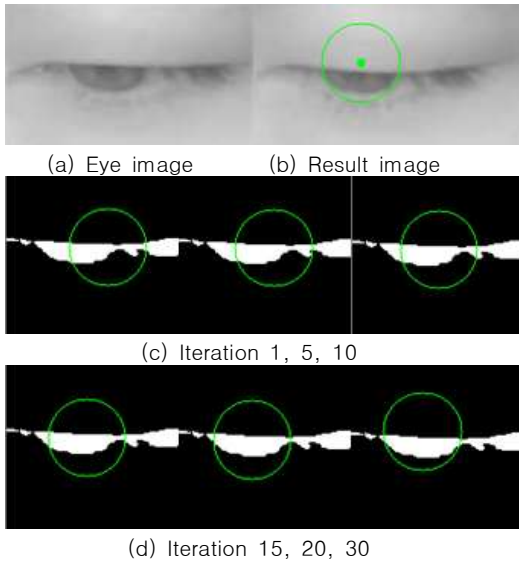


Fig. 11. Pupil center detection using the ellipse detection method

동공의 중심과 눈 안쪽 코너점, 얼굴 포즈 추정 값이 산출되면 이들 정보를 이용하여 시선의 방향을 계산한다. 다른 시선추적 문헌을 보면 안구의 크기와 움직임과 같은 기하학적 모델을 포함하는 눈의 특징으로부터 시선의 방향을 계산하는 것은 매우 어렵고 복잡하다[14-15]. J. Zhu 등은 이러한 문제를 해결하기 위해 시선의 방향을 추정하기 위해 눈 안쪽 코너점과 동공의 중심을 이용한 간단한 방식을 설계했고, 방식은 다음과 같다 [16].  $\{(\alpha_1, \beta_1), (x_1, y_1)\}, \{(\alpha_2, \beta_2), (x_2, y_2)\}$ 는 시선 보정에 사용되는 점  $P_1, P_2$ 에 대한 정보이며,  $\alpha, \beta$ 는 디스플레이 상의  $P_1, P_2$  좌표이고,  $x, y$ 는 눈 안쪽 코너점-동공 중심 벡터이다.

$$\alpha = \alpha_1 + \frac{x - x_1}{x_2 - x_1}(\alpha_2 - \alpha_1),$$

$$\beta = \beta_1 + \frac{y - y_1}{y_2 - y_1}(\beta_2 - \beta_1) \quad (11)$$

상기 방법은 간단하면서도 근거리에서 1.4°의 정확도를 보이지만, 얼굴 포즈 변화에 대한 자유도는 허락하지 않는다. 따라서 식 11에 얼굴 포즈 정보를 추가하여 얼굴의 움직임에 대한 자

유도를 높인다. 시선 보정에 사용되는 점은 그림 12와 같고, 디스플레이상의 시선 위치는 식 12로 나타낸다.



Fig. 12. Coordinates used in gaze calibration

$$\text{gaze}_x = (\alpha_{p12} + \alpha_{p34}) * 0.5 + \text{head}_x * 0.5,$$

$$\text{gaze}_y = (\beta_{p12} + \beta_{p34}) * 0.5 + \text{head}_y * 0.5 \quad (12)$$

카메라를 원점으로 사람의 위치가  $(X, Y, Z)$ 일 때,  $\text{head}_x, \text{head}_y$ 는 식 (13)으로 계산한다.

$$\text{head}_x = \alpha_1 + \frac{h_x - h_{x1}}{h_{x2} - h_{x1}}(\alpha_2 - \alpha_1),$$

$$\text{head}_y = \alpha_1 + \frac{h_y - h_{y1}}{h_{y2} - h_{y1}}(\alpha_2 - \alpha_1) \quad (13)$$

$$h_{x2} = \theta_x - (u_x * X),$$

$$h_{x1} = -\theta_x + (u_x * X)$$

$$h_{y2} = \theta_y - (u_y * Y),$$

$$h_{y1} = -\theta_y + (u_y * Y) \quad (14)$$

$$u_x = \frac{\theta_x}{w * 0.5}, \quad u_y = \frac{\theta_y}{h * 0.5} \quad (15)$$

여기서,  $w$ 와  $h$ 는 디스플레이의 너비와 높이이며,  $(X, Y, Z)$ 는 사람의 위치를 나타낸다. 이때  $\theta_x = 90 - \tan^{-1}\left(\frac{Z}{w * 0.5}\right)$

이고,  $\theta_y = 90 - \tan^{-1}\left(\frac{Z}{h * 0.5}\right)$ 이다.

## V. Experiment results

본 논문에서는 시선 추적 실험을 위해 실험 환경을 아래와 같이 구성하였다. 장비는 Canon VB-H43으로 팬-틸트-줌 제

어가 가능한 카메라와 850nm 파장의 적외선 조명, 디스플레이는 1920x1080 해상도의 42인치 디스플레이를 사용했다.

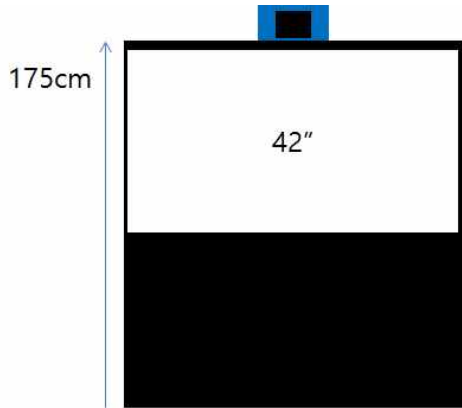


Fig. 13. Experimental environment

실험은 디스플레이와 사용자의 거리를 1.5m, 2m, 2.5m로 나누어 진행하고, 그림 14와 같이 9개의 점을 일정 시간 응시하며 정확도를 측정하였다. 그림 15는 각 거리에서 시선 추정의 결과가  $\pm 1^\circ$ ,  $\pm 2^\circ$ ,  $\pm 3^\circ$  범위에 어떻게 분포하는지 나타냈다. 그 결과 1.5m에서는 시선 추정은  $\pm 2^\circ$ 내에 가장 많이 분포되어 있고, 거리가 멀어질수록  $\pm 3^\circ$ 내에 분포하였다.

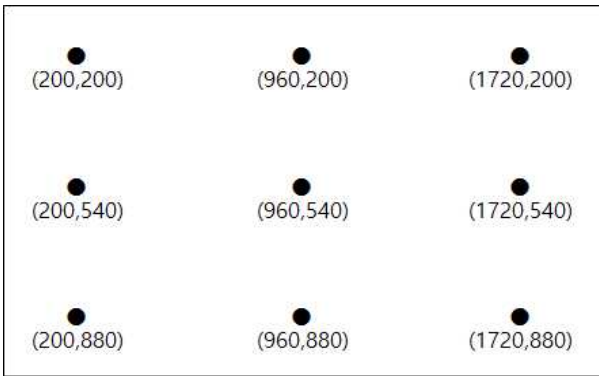


Fig. 14. Reference points on display

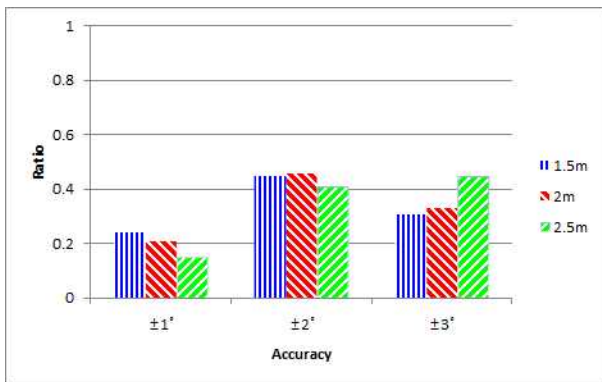


Fig. 15. By distance gaze tracking accuracy

균 위치를 나타낸 그림이다. 빨간색 십자가는 참조점 (reference points)이고, 파란색 원은 추정된 시선의 평균 위치를 나타낸 것이다. 표 1은 디스플레이상의 시선추정 오차거리와 각도를 나타낸 것이다. 이를 볼 때 시선 추정은 평균  $2.5^\circ$  정확도의 성능을 보임을 알 수 있다.

Table 1. Gaze distance and angle error

	1.5m	2m	2.5m
평균 오차거리 (pixel)	128	182	237
각도 (degree)	2.37	2.52	2.63

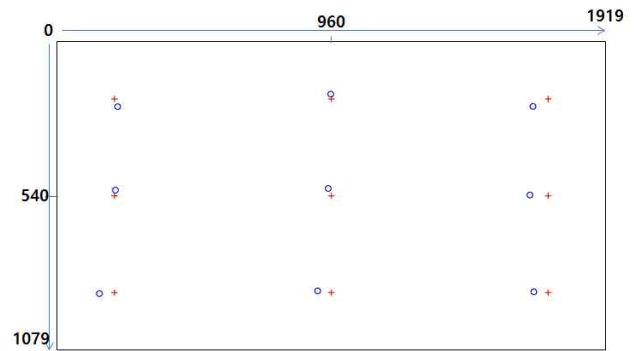


Fig. 16. Average point-of-gaze from 1.5m

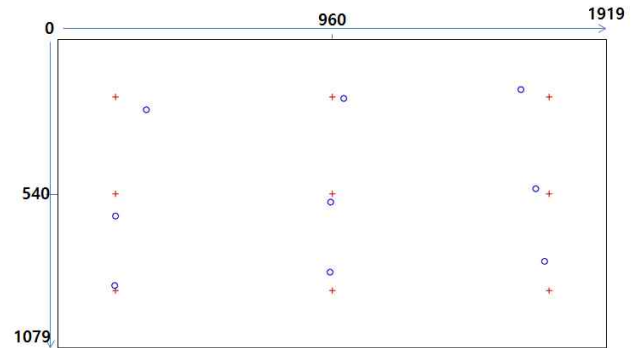


Fig. 17. Average point-of-gaze from 2m

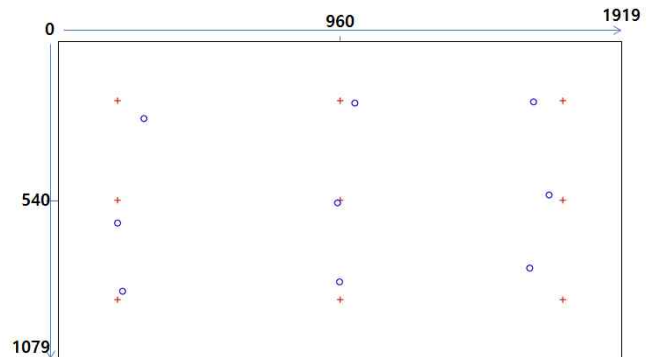


Fig. 18. Average point-of-gaze from 2.5m

그림 16-18은 거리 1.5m와 2m, 2.5m에서 추정된 시선의 평

## VI. Conclusions and future works

본 논문에서는 디스플레이 형태에 따른 시선추적의 제약사항을 해결하기 위한 알고리즘을 제시했고, 원거리에 있는 사용자의 시선을 추적할 경우 복잡해지는 장비의 구성을 간소화하기 위해 단일 팬-틸트-줌 카메라를 사용하였다. 또한 시선 추적 카메라가 디스플레이의 상단에 설치된 경우에 디스플레이의 형태에 따른 제약을 받지 않을 수 있었다. 시선을 추적하는 방법은 사용자가 나타나기 전에는 카메라가 광각모드로 사람이 탐지될 경우 협각모드로 전환하여 사람의 시선 정보를 획득하였다. 디스플레이와 사용자의 거리가 1.5~2.5m 사이일 때 시선추정의 정확도는  $\pm 3^\circ$  이내의 결과를 보였고, 이는 옥외 광고, 전시장 등에서 많이 사용되는 대화면 디스플레이에서 활용 가치가 있음을 확인했다. 향후 시선추적이 적용될 환경을 고려하여 사용자 시선 보정 절차를 생각하는 방향으로 연구가 필요하다.

## REFERENCES

- [1] M. Horsley, M. Eliot, B. A. Knight and R. Reilly, "Current Trends in Eye Tracking Research". Springer, Dec, 2013.
- [2] A. T. Duchowski, "A Breadth-First Survey of Eye-Tracking Applications", Behavior Research Methods Instruments Computers J. Psychonomic Soc. Inc., Vol. 34, pp. 455-470, Aug, 2002.
- [3] D. C. Cho and W. Y. Kim, "Long-Range Gaze Tracking System for Large Movements", IEEE Transactions on Biomedical Engineering, Vol. 60, No. 12, Jun, 2013.
- [4] H. C. Lee, W. O. Lee, C. W. Cho, S. Y. Gwon, K. R. Park, H. K. Lee and J. H. Cha, "Remote Gaze Tracking System on a Large Display", Sensors, Vol. 13, No. 10, pp. 13439-13463, Jul, 2013.
- [5] W. O. Lee, Y. G. Kim, K. Y. Shin, D. T. Nguen, K. W. Kim, K. R. Park and C. I. Oh, "New method for face gaze detection in smart television", Optical Engineering Vol. 53, No. 5, pp. 1-12, May, 2014.
- [6] C. Hennessey and J. Fiset, "Long Range Eye Tracking: Bringing Eye Tracking into the Living Room", Proc. of symposium on Eye tracking research & applications, pp.249 -252, Jan, 2012.
- [7] C. Küblbeck and A. Ernst, "Face detection and tracking in video sequences using the modifiedcensus transformation", Image and Vision Computing, Vol. 24 No. 6, pp.564-572, Jun, 2006.
- [8] Y. Freund and R. E. Schapire, "A Short Introduction to Boosting", Journal of Japanese Society for Artificial Intelligence, Vol. 14, No. 5, pp. 771-780, Sep, 1999.
- [9] V. Kazemi and S. Josephine, "One millisecond face alignment with an ensemble of regression trees", Proc. IEEE Conf. Comput. Vis. Pattern Recog., pp. 1867-1874, Jun, 2014.
- [10] D. F DeMenthon and L. Davis, "Model-based object pose in 25 lines of code," In Second European Conference on Computer Vision(ECCV), pp. 335-343, Oct, 1992.
- [11] K. J. Jeong, J. S. Choi and G. J. Jang, "Facial Expression Recognition using Face Alignment and AdaBoost," Journal of The Institute of Electronics and Information Engineers, Vol. 51, No. 11, Nov, 2014
- [12] A.W. Fitzgibbon and R.B. Fisher, "A buyers guide to conic fitting," in: Proc. fifth British Machine Vision Conference, pp. 513-522, Feb, 1995.
- [13] A. Yuille, "Deformable Templates for Face Recognition", J. Cognitive Neuroscience, Vol. 3, No. 1, pp. 59-71, Jan, 1991.
- [14] K.N. Kim and R.S. Ramakrishna, "Vision-based eye-gaze tracking for human computer interface", International Conference on Systems Man and Cybernetics, pp. 324-329, Oct, 1999.
- [15] Y. Matsumoto and A. Zelinsky, "An algorithm for real-time stereo vision implementation of head pose and gaze direction measurement", International Conference on Automatic Face and Gesture Recognition, pp. 499-504, Mar, 2000.
- [16] J. Zhu and J. Yang, "Subpixel eye gaze tracking," in: Proc. of the 5th IEEE International Conference on Automatic Face and Gesture Recognition, Washington, DC, pp. 131-136, May, 2002.

## Authors



Gyung Ju Lee received the M.S. degrees in Computer Science and Engineering from Soongsil University, Korea, 2013.

M.S. Lee joined the faculty of the Department of Computer Science at Soongsil University, Seoul, Korea, in 2010. She is currently the doctor's course in the Department of Computer Science, Soongsil University. She is interested in Computer Vision, Pattern Recognition, Image Processing.



Jin Suh Kim received the B.S. degrees in Computer Science and Engineering from Academic Credit Bank System, Korea, 2013.

B.S. Kim joined the faculty of the Department of Computer Science at Soongsil University, Seoul, Korea, in 2014. He is currently the master's course in the Department of Computer Science, Soongsil University. He is interested in Computer Vision, Pattern Recognition, Image Processing.



Gye Young Kim received the B.S., M.S. and Ph.D. degrees in Computer Science and Engineering from Soongsil University, Korea, in 1990, 1992 and 1996, respectively.

Dr. Kim joined the faculty of the Department of Computer Science at Soongsil University, Seoul, Korea, in 1983. He is currently a Professor in the Department of Computer Science, Soongsil University. He is interested in Computer Vision, Biometrics, Pattern Recognition, Augmented Reality, Image and Signal Processing.