

論文2002-39SP-3-3

얼굴의 3차원 위치 및 움직임 추정에 의한 시선 위치 추적 (Facial Gaze Detection by Estimating Three Dimensional Positional Movements)

朴康領*, 金在熹**

(Kang Ryoung Park and Jai Hie Kim)

요 약

시선 위치 추적이란 모니터상에 사용자가 쳐다보고 있는 지점을 파악해 내는 기술이다. 이 논문에서는 컴퓨터 비전 방법을 이용하여 사용자가 모니터 상에 어느 지점을 쳐다보고 있는지를 파악(시선 위치 추적) 하는 새로운 방법을 제안한다. 시선위치를 파악하기 위해 본 논문에서는 얼굴 영역 및 얼굴 특징점(양 눈, 양 콧구멍, 입술 끝점 등)을 2차원 카메라 영상으로부터 추출하였으며, 이들의 움직임으로부터 카메라 보정 및 매개변수 추정 방법등을 이용하여 초기 3차원 위치를 추정해 내었다. 이후 모니터 상의 한 지점을 쳐다 보기 위해 사용자가 얼굴을 움직이는 경우 이러한 얼굴의 3차원 움직임 량 역시 자동으로 추정하였다. 이로 부터 변화된 얼굴 특징점의 3차원 위치를 계산해 낼 수 있었으며, 이를 바탕으로 모니터 상의 시선 위치를 구하였다. 실험 결과, 19인치 모니터상의 임의의 지점을 사용자가 쳐다보았을 때, 약 2.01인치의 시선 위치에러 성능을 얻었다.

Abstract

Gaze detection is to locate the position on a monitor screen where a user is looking. In our work, we implement it with a computer vision system setting a single camera above a monitor and a user moves (rotates and/or translates) his face to gaze at a different position on the monitor. To detect the gaze position, we locate facial region and facial features(both eyes, nostrils and lip corners) automatically in 2D camera images. From the movement of feature points detected in starting images, we can compute the initial 3D positions of those features by camera calibration and parameter estimation algorithm. Then, when a user moves(rotates and/or translates) his face in order to gaze at one position on a monitor, the moved 3D positions of those features can be computed from 3D rotation and translation estimation and affine transform. Finally, the gaze position on a monitor is computed from the normal vector of the plane determined by those moved 3D positions of features. As experimental results, we can obtain the gaze position on a monitor(19 inches) and the gaze position accuracy between the computed positions and the real ones is about 2.01 inches of RMS error.

Keyword : Gaze Detection, camera calibration, parameter estimation, 3D motion estimation

* 正會員, LG 電子技術原

(LG Electronics Institute of Technology),

** 正會員, 延世大學校 電氣·컴퓨터工學科

接受日字:2001年3月8日, 수정완료일:2002年2月26日

1. 서 론

시선 위치 추적이란 컴퓨터 비전 방법에 의해 사용자가 모니터 상의 어느 지점을 쳐다보고 있는 지를 파악해 내는 기술이다. 이러한 시선 위치 추적 기술은 많

은 응용 분야를 가지고 있는데, 그 대표적인 예로는 손발을 사용하지 못하는 심신 장애자를 위한 컴퓨터 인터페이스, 다중 윈도우 환경에서 마우스 커서의 움직임을 사용자의 시선 위치 추적으로 대응하거나 혹은 공정 제어 환경과 같이 동시에 조정해야 할 버튼들이 많은 상황에서 사용자의 양손 이외에 제 3의 입력 수단으로 시선 위치 추적 기술을 이용할 수 있다^[1].

기존 대부분의 연구에서는 컴퓨터 비전 방법에 의해 얼굴의 3차원 움직임량(회전(rotation) 및 이동(translation))만을 파악하는 방법들이 주로 연구되었으며^[2-6], 이로부터 모니터 상에 사용자가 쳐다보고 있는 위치를 파악하는 연구는 최근 들어서 조금씩 수행되고 있다^[7-8]. Azarbajani와 Fukuhara등은^[2-3] 각각 확장 칼만 필터(Extended Kalman Filter)와 신경망(Neural Network)등을 이용하여 얼굴의 3차원 회전 및 이동량을 추정하는 연구를 수행하였으며, Ballard등은^[4] 연속적 근사 방법(successive approximation method)을 사용하여 시선 벡터의 회전량을 파악하였다. 또한 Gee등^[5]과 Heinzmann등^[6]은 아핀 투영 방법(affine projection algorithm)을 사용하여 시선 벡터의 3차원 회전량을 구하였다. 그러나 그들의 연구들은 전술한 바와 같이 사용자 얼굴의 3차원 상대적인 회전량 및 이동량만을 추정하였을 뿐, 모니터 상에 사용자가 쳐다보고 있는 위치는 계산하지 않았다. 이러한 얼굴의 3차원 움직임 추정 결과로부터 모니터 상의 시선 위치를 알기 위해서는 모니터와 사용자 얼굴 사이의 3차원 거리(depth) 파악, 모니터, 카메라 및 얼굴 좌표계 간의 결합 방법 등이 추가로 연구되어야 하기 때문이다. 이에 반하여 Rikert등^[7]은 신경망에 의해 학습된 변환 2차원 얼굴 모델(morphable face model)을 이용하여 모니터 상의 시선 위치를 파악하였다. 그러나 그들의 방법은 얼굴의 3차원 위치 및 움직임(회전 및 이동)을 파악하지 않는 상태에서 모니터 상에 쳐다보는 위치를 직접 계산하는 방법을 사용하였으며, 이러한 이유로 모니터와 사용자 얼굴 사이의 3차원 거리(depth)를 고정(50cm)시켜야 하고, 제한된 범위의 얼굴 움직임만 허용하여 만일 얼굴의 자연스러운 3차원 움직임이 발생하는 경우에 시선 위치 추출 에러가 증가되며, 또한 시선 위치 추적 시스템에 대해 학습된 제한된 사용자들만이 이용 가능한 문제점이 있었다. 이러한 문제점을 해결하기 위하여 Tomono등^[8]은 양안 카메라(stereo camera)를 이용하여 사용자 얼굴의 3차원 위치를 파악하고 이를 바탕으로

모니터상의 시선 위치를 파악하는 연구를 수행하였으나, 이러한 경우 두 대의 카메라 이용으로 시스템 가격이 상승하고, 두 대의 카메라로부터 영상이 입력되므로 영상 입력 속도가 저하되어 전체적으로 시스템 처리 속도가 감소되는 문제점이 있다.

이 논문에서는 이러한 문제점을 해결하기 위하여 단안 카메라 환경에서 사용자 얼굴의 3차원 위치 및 자연스러운 움직임(회전 및 이동)을 컴퓨터 비전 방법에 의해 추정하고, 이로부터 모니터상의 시선 위치를 파악할 수 있는 새로운 방법을 제안한다.

II. 시선 위치 추적

이 논문에서는 다음 그림 1과 같은 순서에 따라 모니터상의 시선 위치를 파악한다.

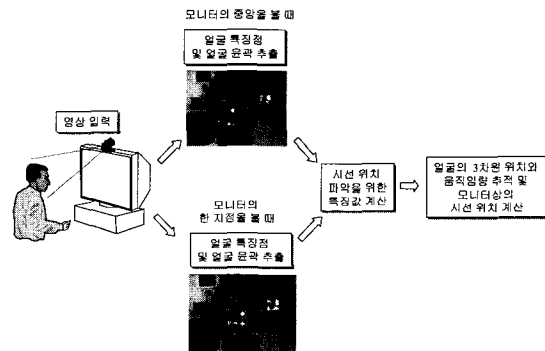
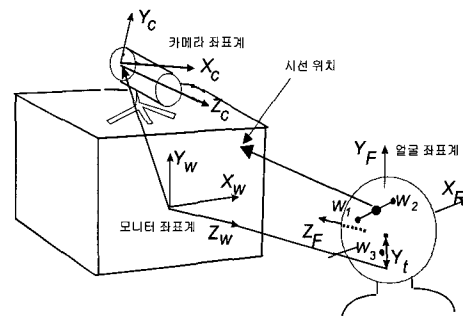


그림 1. 시선 위치 추적 방법의 개략도
Fig. 1. Overview of gaze detection.

본 논문에서는 전술한 바와 같이 단안 카메라(mono camera) 환경에서 얼굴 특징점의 3차원 위치/움직임 및 모니터상의 시선 위치를 파악하기 위하여 그림 2과 같은 얼굴, 카메라 및 모니터 좌표계 모델을 이용한다.



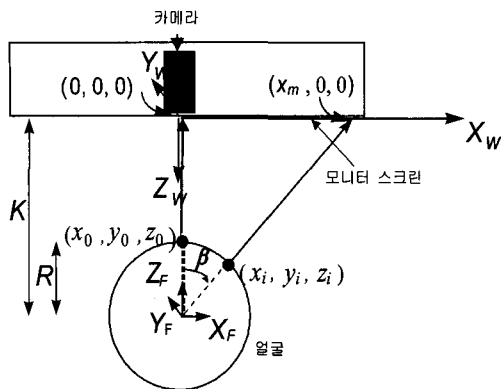


그림 2. 얼굴, 모니터 및 카메라 좌표계 및 모니터의 중앙과 모니터의 오른쪽 한 지점을 응시하는 순간에 관측된 사용자의 얼굴 회전 모델

Fig. 2. Face, monitor and camera coordinates and the rotational model when a user gazes at a monitor center and a different point.

1. 얼굴 특징점 및 얼굴 윤곽 추출

그림 1과 같이 모니터 위에 설치되어 있는 카메라로부터 얼굴 영상이 입력되면, 얼굴 특징점(양 눈, 코, 입) 및 얼굴 윤곽을 영상 처리 방법에 의해 자동으로 추출한다. 입력 영상으로부터 얼굴 특징점을 추출하기 위하여, 시간적으로 연속된 두 영상간의 차영상 정보와 살색 칼라 정보(YIQ칼라)를 이용하여 얼굴 영역을 검출하고 추출된 얼굴 영역내의 제한된 범위 내에서 수직, 수평히스토그램의 피크 위치를 파악함으로써 양 눈, 양 콧구멍 및 입의 양 끝점의 위치를 추출한다. 여기서 RGB대신 YIQ를 이용한 이유는 RGB 좌표계의 경우 칼라 정보와 밝기 정보가 혼합되어 있어 외부 광의 존재 여부에 따라 검출 오류를 많이 생성할 수 있으나 이를 YIQ로 변환하는 경우 밝기 성분은 Y좌표계로 색상 정보는 I, Q로 분해되어 배경으로부터 얼굴 영역을 검출하기가 용이하다. 특히 I성분은 밝기 성분이 제거된 red성분으로 간주될 수 있으므로 이를 이용하여 얼굴 살색 영역을 구분하기가 쉬워 진다(얼굴의 살색 정보에 민감한 I성분 구간(110~150)). 여기서 차영상정보와 얼굴의 살색 정보를 동시에 이용하는 이유는 이러한 살색 정보만 이용하는 경우 태양의 반사광이 생기는 경우 배경과 살색을 구분하기 어렵기 때문이다. 그러므로 본 논문에서는 YIQ좌표계에서 구분된 살색 영역을 바탕으로 얼굴의 대략적인 영역을 설정하고 이 내부에서 얼굴의 차영상 정보를 이용하여 대략적인 얼굴의 Box영역을 검출한다. 본 논문에서는 초기에 얼굴

영역을 추출할 경우에만 영상 capture board의 frame rate을 7 frames/sec의 속도로 자동 설정(얼굴 영역이 검출된 후에는 다시 30 frames/sec으로 재조정함)하므로 아무리 인위적으로 얼굴을 고정하려 해도 이 시간 간격 동안에는(132ms 간격)간에 얼굴의 움직임이 발생하여 차영상으로부터 움직임 정보를 검출할 수 있었다. 추출된 얼굴 영역에서 조도의 변화에 따른 영향을 줄이기 위하여 히스토그램 평활화(histogram equalization) 및 이진화(binanzation) 과정을 통해 이진 영상으로 변환하였다. 얼굴 영역에 대한 이진 영상이 구해진 후, 그림 3-(a)처럼 얼굴 내 눈의 위치에 대한 사전지식을 이용하여 이진 영상의 제한된 범위 내에서 수직, 수평 히스토그램의 피크 위치(peak position)를 파악함으로써 눈의 위치를 정확하게 추출할 수 있었다. 양 눈이 검출된 후 그림 3-(b)처럼 입의 위치에 대한 존재 가능 범위를 설정한 후, 이 영역에 대한 이진화 및 수직 방향 히스토그램으로 입선의 수직 위치를 먼저 추출하였다.

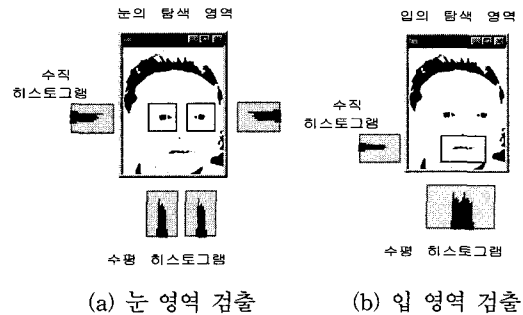


그림 3. 양 눈 및 입의 양 끝점 위치 검출
Fig. 3. Detect the both eyes and mouth corners.

추출된 입선의 수직 위치로부터 입의 양 끝점을 추출하기 위해 입선에 대한 수평 히스토그램을 구하여 임계치(수평히스토그램의 평균값)이상으로 갑작스럽게 변화되는 지점을 입의 양 끝점으로 결정하였다. 콧구멍 역시 눈동자와 같은 방법으로 추출하였으며, 두 콧구멍의 근접도가 큰 관계로 양 콧구멍을 포함한 영역에 대한 수평 히스토그램을 통해 두 개의 피크 위치를 추출함으로써 양 콧구멍의 수평 위치를 파악하였다. 기존의 연구^[12]에서는 눈 위치 추출시 안경 사용자의 경우, 안경알에 의해 반사되어 눈동자 확대소가 많이 소실될 수 있다. 이러한 문제점을 해결하기 위하여 눈 영역의 이진화시 p-tile method를 사용하여 확대소의 개수를

일정하게 유지함으로써 흑화소 소실 문제를 해결하였다^[9]. p-tile method는 임계치를 결정하여 영상을 이진화하고자 할 때, 전체 영역 크기 중 원하는 부분이 차지하는 면적이 p %가 되도록 유지하여 이진화하는 방법이다. 그림 4는 본 논문의 방법을 사용하여 검출된 사용자의 얼굴 영역 및 눈, 코, 입의 양 끝 위치를 나타낸 것이다. 얼굴 특징점의 추출 결과 양 눈과 입의 양 끝점 보다는 양 콧구멍의 검출 오차가 큰 것으로 나타났다. 이는 양 콧구멍의 경우 코 옆의 그림자 부분이 콧구멍으로 추출되는 경우가 있었기 때문이다. 그리고 test결과 모니터 내(19인치 기준)의 아래 끝 지점을 쳐다보는 경우에도, 입력 영상에서 콧구멍이 보이지 않을 정도로 얼굴을 숙이지는 않게 됨을 알 수 있었다.



그림 4. 추출된 얼굴 영역 및 눈, 코, 입 특징점 위치
Fig. 4. Detection of facial region and facial features.

또한 이후 입력 영상에서는 매번 얼굴 영역을 다시 추출하지 않고, 이전에 추출된 특징점의 위치로부터 현재 특징점의 위치를 예측하는 알고리즘(Discrete Kalman Filter^[11])을 사용함으로써 매 입력 영상마다 특징점의 움직임을 추적한다. 기존의 연구^[12]에서는 단순히 등가속도 움직임 예측 방법을 적용하여 특징점의 움직임을 추적하였으나, 본 논문에서는 DKF를 사용하였으며 이러한 경우 등가속도 움직임 예측 및 특징점의 위치 보정을 결합하여 보다 정확한 위치 추적이 가능하였다. 그런데, 차영상과 살색 칼라 정보를 이용한 방법으로는 오차가 많이 포함된 대략적인 얼굴 박스 위치만 추출되므로, 이로부터는 얼굴의 3차원 이동량을 위해 필요한 얼굴 윤곽의 정확한 무게중심 이동량을 구할 수 없다. 그러므로 이 논문에서는 다음과 같은 방법으로 그림 5에서와 같이 배경이 복잡한 입력 영상에서 정확한 얼굴 윤곽을 추출하였다. 얼굴 영상이 입력되면 2방향 소벨 경계 추출 연산자(2 directional sobel edge operator)를 이용하여 에지 영상을 얻는다(그림 5(b)). 이때, 실험 결과 얼굴 정면에 태양광이나

별도 조명과 같은 인위적인 광을 비추지 않는 이상 얼굴의 경계에 비해 얼굴 특징점에 의해 나타나는 얼굴 내부의 에지의 두께가 약하게 나타나기 때문에, 높은 이진화 임계치를 설정하여 사용하면 얼굴 내부의 에지 성분이 거의 남지 않게 된다(그림 5(c)). 이때의 이진화 임계치는 여러명의 test data로부터 실험적으로 구한 고정값을 사용한다. 그리고 이 순간 입력 영상에서는 얼굴 특징점(양 눈, 양 콧구멍, 입의 양 끝점)의 위치가 파악된 상태이므로, 추출된 양 눈의 위치를 중심으로 x, y방향의 외곽 방향으로 탐색할 때 처음 만나는 에지 부분을 얼굴 윤곽 위치로 정할 수 있게 된다(그림 5(d)). 이렇게 추출된 얼굴 윤곽의 기하학적인 중심을 얼굴의 무게 중심으로 정하여 사용하였다(그림 5(e)). 실험 결과 다양한 얼굴에 대해서도 얼굴 중심을 비교적 정확하게 찾을 수 있었다. 이때 그림 5(e)에서 알 수 있듯이 이진화 된 얼굴 윤곽(그림 5(c))은 부분적으로 연결되지 않고 끊어진 곳이 존재하였으나 이러한 부분은 얼굴 전체 중에 일부분에 해당되므로 전체 얼굴(머리 끝에서 입선 바로 밑) 영역의 무게 중심을 구하는 데에는 큰 영향을 주지는 않았다.

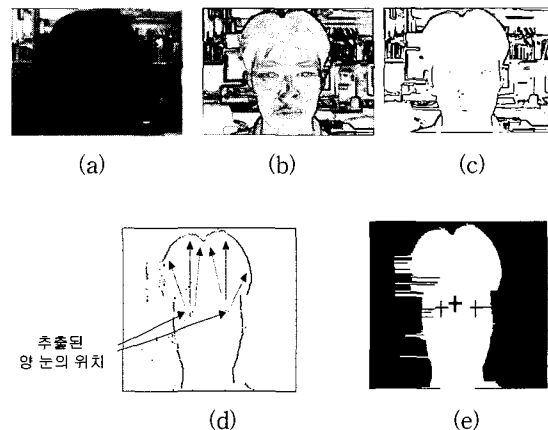


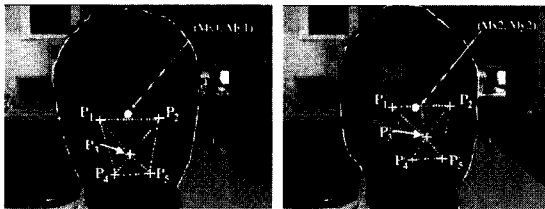
그림 5. 얼굴 윤곽선 및 무게 중심 추출 방법
Fig. 5. Facial contour and center detection by edge operation.

2. 시선 위치 파악을 위한 특징값 계산

전술한 바와 같이 사용자의 시선 위치를 파악하기 위해 이 논문에서는 양 눈과 코, 입 및 얼굴 윤곽의 무게중심 위치를 특징점으로 사용하였다. 이러한 특징점들은 사용자가 그림 6과 같이 모니터 중앙과 모니터상의 한 지점을 쳐다볼 때 각각 추출한다. 이때 사용자는

모니터 중앙을 매번 쳐다볼 필요는 없고, 오직 초기에 사용자가 모니터 앞에 처음 앉았을 때 자리 보정을 위해 한 번 쳐다본 정보를 이용한다. 이렇게 추출된 특징점의 위치들로부터 모니터 상의 시선 위치를 파악하기 위한 특징값들을 다음과 같이 계산한다.

아래의 그림 6.(a)와 그림 6.(b)는 각각 모니터의 정 중앙과 모니터상의 한 지점을 쳐다보는 순간에 추출된 특징점(양 눈, 양 콧구멍의 중심, 입의 양 끝점) 및 얼굴 무게중심의 위치를 나타낸 것이다. 이때, 추출된 얼굴 특징점과 무게 중심 위치로부터 모니터상의 응시 위치를 파악하기 위해 이 논문에서는 다음과 같은 18 개의 특징값들을 정하였다.



(a) 모니터상의 정 중앙 응시 (b) 모니터상의 한 지점 응시

그림 6. 모니터 정 중앙과 한 지점을 응시하는 순간의 얼굴 특징점 및 무게중심 위치
Fig. 6. The positions of facial features and geometrical center when a user gazes at a monitor center and a different point.

▶ 모니터의 정 중앙을 응시할 때

P_1 (왼쪽눈: x_1, y_1), P_2 (오른쪽눈: x_2, y_2), P_3 (코: x_3, y_3),
 P_4 (입의 왼쪽 끝: x_4, y_4), P_5 (입의 오른쪽 끝: x_5, y_5)
(얼굴 영역의 무게중심: Mx_1, My_1)

▶ 모니터의 한 지점을 응시할 때

P'_1 (왼쪽눈: x'_1, y'_1), P'_2 (오른쪽눈: x'_2, y'_2), P'_3
(코: x'_3, y'_3),
 P'_4 (입의 왼쪽 끝: x'_4, y'_4), P'_5 (입의 오른쪽 끝:
 x'_5, y'_5) (얼굴 영역의 무게중심: Mx_2, My_2)

특징값 1~5: $x'_i - x_i - (Mx_2 - Mx_1)$
($i = 1, 2, \dots, 5$)

특징값 6~10: $y'_i - y_i - (My_2 - My_1)$
($i = 1, 2, \dots, 5$)

특징값 11: $S(\triangle P'_1P'_2P'_3) - S(\triangle P_1P_2P_3)$,
($S(\triangle P'_1P'_2P'_3)$ 는 $\triangle P'_1P'_2P'_3$ 의 면적)

특징값 12: $S(\triangle P'_3P'_4P'_5) - S(\triangle P_3P_4P_5)$

특징값 13: $\{(x'_1 + x'_4)/2 - x'_3\} - \{(x_1 + x_4)/2 - x_3\}$

특징값 14: $\{(x'_3 - (x'_2 + x'_5)/2)\} - \{(x_3 - (x_2 + x_5)/2)\}$

특징값 15: $\{(y'_4 + y'_5)/2 - y'_3\} - \{(y_4 + y_5)/2 - y_3\}$

특징값 16: $\{(y'_3 - (y'_1 + y'_2)/2)\} - \{(y_3 - (y_1 + y_2)/2)\}$

특징값 17: $Mx_2 - Mx_1$

특징값 18: $My_2 - My_1$

특징값 1~5, 6~10은 모니터 중앙과 한 지점을 쳐다보는 순간 관측된 얼굴 특징점의 X, Y축 변위를 나타낸 것이다. 또한 특징값 11, 12는 얼굴 특징점들에 의해 형성되는 삼각형의 면적 변화도를 나타내며, 특징값 13~16은 양 눈 및 입에 대한 코의 X, Y축 변위를 나타낸 것이다. 여기서 특징값 11, 12는 얼굴의 이동시에는 변화를 나타내지 않으나, 얼굴의 회전이 발생하는 경우 그 크기가 변화되므로 얼굴의 회전량 추정을 위한 특징으로 사용된다. 일반적으로 얼굴의 회전(rotation)과 이동(translation)은 상호 독립적으로 발생하므로 이 둘을 상호 분리하여(separably) 처리하는 경우 복잡도(complex dimension)이 줄어들게 된다[9]. 이때 얼굴의 회전은 얼굴축의 움직임이 없는 상태에서의 얼굴 특징점의 움직임 형태로, 그리고 얼굴의 이동은 얼굴축의 움직임, 즉, 얼굴 무게중심의 움직임 형태로 표현될 수 있다. 그러므로 위의 18개의 특징값들 중 얼굴 특징점의 움직임에 의해 계산되는 1~16까지의 특징값은 얼굴의 회전(rotation) 성분을 반영하며, 얼굴 무게중심, 즉 얼굴 축의 X, Y축 변위를 나타내는 특징값 17, 18은 얼굴의 이동(translation)성분을 나타내게 된다. 기존의 연구^[12]에서는 총 22개의 특징값들을 사용하였으나 본 논문에서는 이 특징값들의 상호 연관 관계성을 조사하여 불필요한 특징값들을 제거함으로써 총 18개의 특징값들만 사용하였다.

3. 얼굴의 3차원 위치와 움직임량 추적 및 모니터 상의 시선 위치 파악

이렇게 계산된 18개의 특징값들을 이용하여 본 논문에서는 그림 7의 4 단계를 거쳐 모니터상의 시선 위치를 파악한다.

그림 7의 제 1 단계에서는 초기에 사용자가 모니터의 중앙을 쳐다보는 순간 얼굴 특징점(양 눈, 입의 중

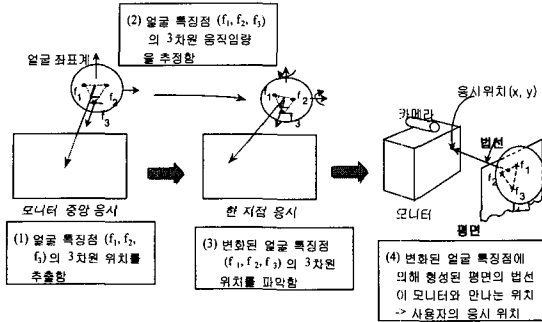


그림 7. 얼굴의 3차원 위치 및 움직임 추정에 의한 시선 위치 파악
 Fig. 7. Gaze detection by computing 3-D facial positions and motions.

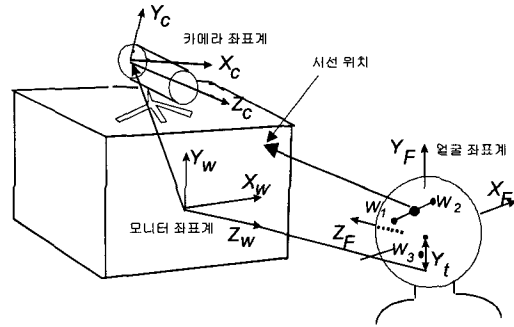
점)의 삼차원 위치를 계산해 낸다. 다음으로 제 2 단계에서는 사용자가 모니터의 한 지점을 쳐다보는 순간 얼굴의 삼차원 회전(rotation) 및 얼굴 축의 이동량(translation)을 파악한다. 제 3 단계에서는 제 1 단계에서 파악된 얼굴 특징점의 삼차원 위치와 제 2 단계에서 추정된 특징점의 회전 및 이동량을 바탕으로 아핀 변환(affine transformation)^[9]을 이용하여 변화된 얼굴 특징점들의 삼차원 위치를 계산한다. 이때, 변화된 얼굴 특징점들의 삼차원 위치를 구하기 위하여 제 2 단계에서 수행하는 얼굴 특징점의 3차원 움직임량 추정이 필요한 이유는 물체의 3차원 거리(depth) 및 위치를 산술적으로 파악할 수 있는 양안 카메라(stereo camera)와는 달리, 이 논문에서 사용하는 단안 카메라(mono camera)의 경우에는 물체의 3차원 위치를 직접 파악하기가 어렵기 때문이다^[9]. 이로부터 제 4 단계에서는 변화된 얼굴 특징점의 위치로부터 얼굴 평면 방정식을 구한 후, 이의 법선 벡터와 모니터가 만나는 위치를 사용자의 시선 위치로 결정한다.

3-1. 1 단계 : 얼굴 특징점의 3차원 위치 측정

전술한 바와 같이 단안 카메라(mono camera)의 경우에는 물체의 3차원 위치를 직접 파악하기가 어렵다. 그러므로, 이 논문에서는 초기에 얼굴 특징점의 3차원 위치를 파악하기 위하여 그림 8과 같은 얼굴, 카메라 및 모니터 좌표계 모델을 이용한다.

그림 8을 위에서 내려다보면 그림 9와 같이 된다.

실제 사용할 때에는 사용자는 초기에 모니터상의 서로 다른 5지점을 한번씩 쳐다보도록 되어 있으나, 그림 9에서는 사용자의 시선 위치와 얼굴 특징점의 3차원 위치 사이의 관계를 예를 들기 위해 두 지점을 쳐다보



$$X_W = X_F, Y_W = Y_F + Y_t, Z_W = -Z_F + K$$

그림 8. 얼굴 좌표계, 모니터 좌표계, 카메라 좌표계 모델
 Fig. 8. Face, monitor and camera coordinates.

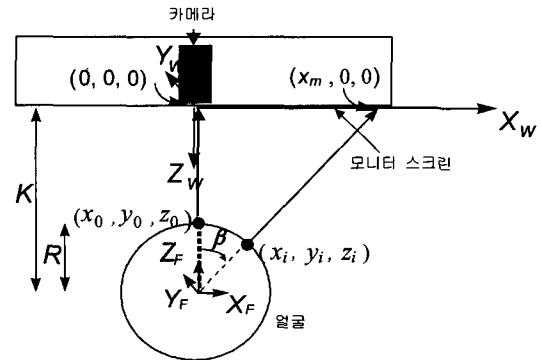


그림 9. 모니터의 중앙과 모니터의 오른쪽 한 지점을 응시하는 순간에 관측된 사용자의 얼굴 회전 모델
 Fig. 9. The rotational model when a user gazes at a monitor center and a different point.

는 것으로 설명하고자 한다.

그림 9와 같이 얼굴을 원으로 가정할 때, 사용자가 모니터의 중앙과 오른쪽 한 지점을 쳐다보는 순간 얼굴 좌표계 (X_F, Y_F, Z_F) 에서 정의된 하나의 얼굴 특징점의 위치 (x_0, y_0, z_0) 와 (x_i, y_i, z_i) 간에는 식 (1)과 같은 관계가 성립한다.

$$\begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \beta & 0 & \sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{bmatrix}$$

단, $\beta = \text{atan}(\frac{x_m}{K})$ (1)

얼굴 특징점의 위치 (x_0, y_0, z_0) 와 (x_i, y_i, z_i) 는 각각 얼굴 좌표계 (X_F, Y_F, Z_F) 에서 정의된 값들이므로

이들을 그림 8의 얼굴 좌표계 (X_F, Y_F, Z_F)와 카메라 좌표계 (X_c, Y_c, Z_c), 모니터 좌표계 (X_W, Y_W, Z_W)간의 변환 관계를 이용하여 식 (2)와 같이 모니터 좌표계에서의 위치값 ($x_{i_w}, y_{i_w}, z_{i_w}$)와 ($x_{0_w}, y_{0_w}, z_{0_w}$)으로 변환한다.

$$\begin{aligned} x_{0_w} &= x_0, & x_{i_w} &= x_i \\ y_{0_w} &= y_0 + Y_t & y_{i_w} &= y_i + Y_t \\ z_{0_w} &= -z_0 + K & z_{i_w} &= -z_i + K \end{aligned} \quad (2)$$

여기서 K 는 그림 8에서와 같이 모니터 좌표계에 대한 얼굴 좌표계의 Z 축 거리 정보이며, Y_t 는 모니터 좌표계에 대한 얼굴 좌표계의 Y 축 높이 정보이다. 이때 그림 7에서 $X_F = X_W$ 라고 가정하면(초기에 모니터 중앙에 얼굴이 위치하면), $x_0 \approx 0$ 이므로, 식 (2)의 결과는 식 (3), (4), (5)와 같이 전개된다.

$$x_{i_w} = \cos \beta \cdot x_0 + \sin \beta \cdot z_0 = \sin \beta \cdot z_0 \quad (3)$$

$$y_{i_w} = y_0 + Y_t \quad (4)$$

$$z_{i_w} = \sin \beta \cdot x_0 - \cos \beta \cdot z_0 + K = \cos \beta \cdot z_0 + K \quad (5)$$

또 이때, 그림 8에서 모니터 좌표계에 대한 카메라 좌표계의 상하각을 α , 좌우각을 θ (단, $\theta=0$)로 정의하고, 모니터 좌표계와 카메라 좌표계 간의 천이(translation) 행렬 값을 $T_x=0, T_y, T_z \neq 0$ (카메라를 모니터 중앙에 위치)으로 가정하면, 모니터 좌표계에서의 얼굴 특징점의 삼차원 위치 (w_i)와 카메라에 투영된 이차원 위치 (c_i) 사이에는 식 (6)과 같은 관계가 성립한다.

$$c_i = P \cdot R_a \cdot T_0 \cdot w_i \quad (6)$$

$$T_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -T_y \\ 0 & 0 & 1 & -T_z \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad R_a = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha & 0 \\ 0 & -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{f} & 1 \end{bmatrix}$$

(단, f 는 카메라의 초점거리, $c_i = [x_i, y_i, z_i, 1]^T$, $w_i = [x_{i_w}, y_{i_w}, z_{i_w}, 1]^T$)

이에 따라 식 (6)을 자세히 기술하면 식 (7)과 같다.

$$\begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha & -T_y \cdot \cos \alpha - T_z \cdot \sin \alpha \\ 0 & -\sin \alpha & \cos \alpha & T_y \cdot \sin \alpha - T_z \cdot \cos \alpha \\ 0 & \frac{\sin \alpha}{f} & -\frac{\cos \alpha}{f} & -T_y \cdot \frac{\sin \alpha}{f} + T_z \cdot \frac{\cos \alpha}{f} + 1 \end{bmatrix} \begin{bmatrix} x_{i_w} \\ y_{i_w} \\ z_{i_w} \\ 1 \end{bmatrix} \quad (7)$$

식 (7)에 의해 모니터 좌표계에서 정의된 얼굴 특징점의 삼차원 위치 ($x_{0_w}, y_{0_w}, z_{0_w}$)와 ($x_{i_w}, y_{i_w}, z_{i_w}$)가 이차원 카메라 좌표계에 투영된 위치를 각각 (x_0, y_0, f)와 (x_i, y_i, f) 라고 할 때, (8), (9)이 얻어진다.

$$\begin{aligned} x_0 &= \frac{f \cdot x_{0_w}}{(y_{0_w} \cdot \sin \alpha - z_{0_w} \cdot \cos \alpha - T_y \cdot \sin \alpha + T_z \cdot \cos \alpha + f)} \\ &= \frac{f \cdot x_{i_w}}{((y_0 + Y_t) \cdot \sin \alpha - (-z_0 + K) \cdot \cos \alpha - T_y \cdot \sin \alpha + T_z \cdot \cos \alpha + f)} \end{aligned} \quad (8)$$

$$\begin{aligned} x_i &= \frac{f \cdot x_{i_w}}{(y_{i_w} \cdot \sin \alpha - z_{i_w} \cdot \cos \alpha - T_y \cdot \sin \alpha + T_z \cdot \cos \alpha + f)} \\ &= \frac{f \cdot \sin \beta \cdot z_0}{((y_0 + Y_t) \cdot \sin \alpha - (-\cos \beta \cdot z_0 + K) \cdot \cos \alpha - T_y \cdot \sin \alpha + T_z \cdot \cos \alpha + f)} \end{aligned} \quad (9)$$

전술한 바와 같이 $x_0 \approx 0$ 이고 $z_0 = R$ 라고 가정하면, 그림 8에서와 같이 사용자가 모니터의 중앙과 한 지점을 응시하는 순간에 이차원 카메라에서 관측된 얼굴 특징점의 x 방향 움직임(d_x)과 모니터의 시선 위치에 대한 X 축 변위(x_m)사이에는 식 (10)과 같은 관계가 성립한다.

$$\begin{aligned} d_x &= x_i - x_0 = \\ &= \frac{f \cdot \sin[\text{atan}(\frac{x_m}{K})] \cdot R}{((y_0 + Y_t) \cdot \sin \alpha - (-\cos[\text{atan}(\frac{x_m}{K})] \cdot R + K) \cdot \cos \alpha - T_y \cdot \sin \alpha + T_z \cdot \cos \alpha + f)} \end{aligned} \quad (10)$$

그런데 위의 식 (10)에서는 d_x 와 x_m 을 제외하고 8개의 변수 ($f, K, T_y, T_z, Y_t, y_0, \alpha, R$)가 존재하며, 이는 크게 두 종류로 나누어, 사용자의 앉은 자세 및 모니터까지의 거리에 따라 달라지는 변수들과 사용자에게 상관없이 카메라 자체의 설치 위치 정보를 나타내는 변수들로 나눌 수 있다. 이중, 카메라 자체의 설치 위치 정보를 나타내는 변수들로는 α (카메라의 상하각), T_y, T_z (모니터 좌표계로부터 카메라 좌표계로의 이동 벡터(translation vector)), f (카메라의 초점 거리)등이 있다. 또한 사용자의 앉은 위치 등에 따라 달라지는 변수들로는 Y_t (모니터 좌표계에 대한 얼굴 좌표계의 Y 축 이동 벡터(translation vector)), y_0 (얼굴 좌표계 내에서 얼굴 특징점의 Y 축 높이), K (모니터 좌표계에 대한 얼굴 좌표계의 Z 축 거리), R (얼굴의 반지름)등이다. 이때 카메라의 설치 위치 정보를 나타내는 변수들은 초

기에 모니터 위에 카메라를 설치한 이후에는 변화되지 않기 때문에, 이 논문에서는 보정점들을 이용한 카메라 보정(camera calibration) 방법에 의하여 측정하였다. 이때 카메라 보정을 위해 사용한 장비는 그림 10과 같다. 이때, 보정점들의 이차원 좌표는 카메라에 의해 관측된 이차원 영상 정보로부터 구하고, 3차원 위치 추적 장비의 일종인 Polhemus sensor를 이용하여 측정된 모니터의 정중앙에서 보정판까지의 3차원 거리(3D depth) 정보로부터 보정점들의 삼차원 위치를 구하게 된다. Polhemus sensor는 전자기장 센싱 방법에 의해 삼차원 이동량(translation) 및 회전량(rotation)을 정확하게 측정할 수 있는 장비이다(평균 0.08 cm의 이동량 측정 에러 및 0.15°의 회전량 측정 에러). 이를 바탕으로 식 (11)을 이용하여 행렬 안에 존재하는 카메라의 설치 정보를 나타내는 파라미터인 α, T_y, T_z, f 등을 구할 수 있다. 식 (11)에서 $[x_i, y_i, z_i, 1]^T$ 는 보정점의 이차원 좌표를 $[x_{iw}, y_{iw}, z_{iw}, 1]^T$ 는 보정점의 삼차원 좌표를 나타낸다.

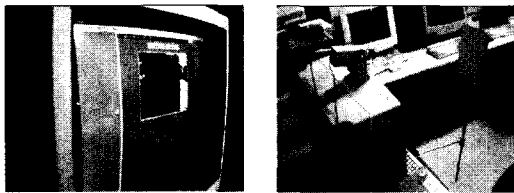
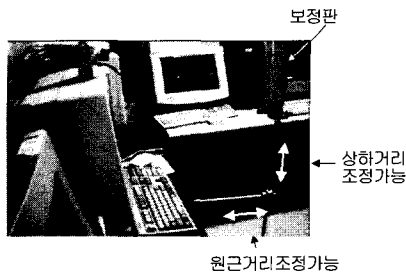


그림 10. 카메라의 설치 정보를 나타내는 파라미터를 구하기 위한 보정 장비

Fig. 10. Calibration equipments for computing camera set-up parameters.

$$\begin{pmatrix} x_i \\ y_i \\ z_i \\ 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha & -T_y \cdot \cos \alpha - T_z \cdot \sin \alpha \\ 0 & \sin \alpha & \cos \alpha & T_y \cdot \sin \alpha - T_z \cdot \cos \alpha \\ 0 & \frac{\sin \alpha}{f} & -\frac{\cos \alpha}{f} & -T_y \cdot \frac{\sin \alpha}{f} + T_z \cdot \frac{\cos \alpha}{f} + 1 \end{pmatrix} \begin{pmatrix} x_{iw} \\ y_{iw} \\ z_{iw} \\ 1 \end{pmatrix} \quad (11)$$

이후 식 (10)에서 남은 미지의 변수들로는 Y, y_0, K, R 등의 4개가 있는데, 이 변수들은 사용자마다 다른

자세 및 모니터까지의 거리등에 따라 달라지는 값이기 때문에, 이 논문에서는 초기에 사용자로 하여금 모니터 상의 미리 알고 있는 5지점(모니터의 중앙, 좌측 상단의 서로 다른 2지점, 우측 하단의 서로 다른 2지점)을 쳐다보게 하는 순간 취득된 4쌍의 (d_x, x_m) 을 식 (10)의 입력으로 사용하고 나머지 미지의 변수인 Y, y_0, K, R 를 매개변수 추정 방법에 의하여 구한다. 이때 (10)은 비선형 방정식이므로 일반적인 선형 회귀법으로는 미지의 변수를 구할 수 없다. 그러므로 이 논문에서는 기존의 매개변수 추정 연구들에서 많이 사용하는 Gauss-Newton method, Steepest Descent method, Davidon-Fletcher-Powell method등을 비교하여 가장 정확한 매개변수 추정 성능을 나타내는 알고리즘(Davidon-Fletcher-Powell method)을 사용하였다^[10].

기존의 연구^[12]에서는 모니터 상의 3지점을 쳐다보았을 때 얻어진 값을 바탕으로 매개변수들을 추정하였으나 추정시 때에 따라 local minima에 빠져서 정확하지 않은 결과를 나타내는 적이 있었으므로 이를 개선하기 위해 본 논문에서는 모니터 상의 5지점을 쳐다보도록 하였다. 이로부터 구한 모니터에서 얼굴 특징점까지의 Z축 거리 ($Z = K - R$)와 앞에서 구한 카메라 파라미터들(f, T_y, T_z, α), 그리고 식 (6)으로부터 얼굴 특징점의 3차원 위치를 구할 수 있게 된다. 그러나, 이렇게 추정한 얼굴 특징점의 삼차원 위치 자체에도 에러(평균 2.79 cm의 RMS 에러)가 있기 때문에, 이 논문에서는 보다 정확한 특징점의 위치를 찾기 위하여 그림 11과 같이 위치 보정 방법을 사용한다.

그림 11과 같이, 계산된 얼굴 특징점(f_1, f_2, f_3)으로

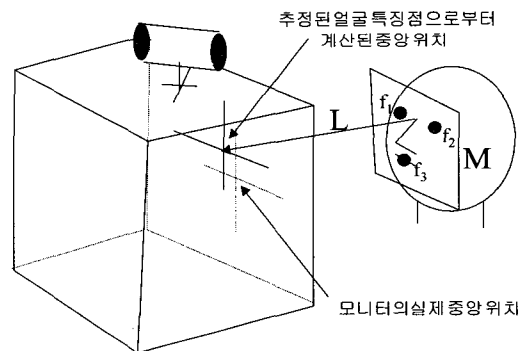


그림 11. 얼굴 특징점의 위치 에러 보정

Fig. 11. The position error compensations of facial features.

부터 얼굴 평면(M)과 얼굴 평면의 법선(L)을 구할 수 있으며(본 논문에서는 얼굴 특징점(f_1, f_2, f_3)가 얼굴 평면(M)에서는 coplanar하다고 가정함), 법선(L)이 모니터 평면과 만나는 위치가 사용자의 시선 위치가 된다. 그런데, 이때 얼굴 특징점의 삼차원 위치는 그림 8에서처럼 사용자가 모니터 중앙을 볼 때 추정된 값이다. 그러므로, 얼굴 특징점(f_1, f_2, f_3)으로부터 계산된 시선 위치는 모니터의 중앙에 놓여야 하며, 만일 그렇지 않다면 시선 위치가 모니터 중앙에 놓이도록 얼굴 특징점의 위치를 보정하게 된다.

3-2. 2 단계 : 얼굴의 3차원 움직임량 추정

(1) 신경망에 의한 얼굴의 3차원 회전량 추정

일반적으로 모니터상의 한 지점을 쳐다보기 위하여 사용자가 얼굴을 움직이는 경우 발생한 얼굴의 회전(rotation)과 얼굴 축의 이동(translation)은 상호 분리하여(separably) 처리 가능하다^[9]. 이처럼 얼굴의 회전과 이동을 분리하여 처리한 이유는 회전과 이동은 상호 독립적으로 발생이 논문에서는 이때 발생한 얼굴의 3차원 회전량을 구하기 위하여 신경망의 일종인 다층 퍼셉트론을 이용한다. 신경망의 입력 노드로는 앞에서 설명한 대로 그림 6의 18개 특징값들 중 얼굴의 3차원 이동량을 추정하기 위해 사용되는 특징값 17, 18을 제외한 16개를 사용하였으며, 은닉 노드는 12개를, 그리고 출력 노드로는 X, Y축 회전각을 나타내는 2개를 사용하였다. 여기서 Z축 회전을 고려하지 않은 이유는 자연스럽게 모니터상의 한 지점을 쳐다볼 때에는 주로 X, Y회전만 발생하지 Z축 회전을 거의 발생하지 않기 때문이다. 기존의 연구^[12]에서는 특징값들 사이의 상관관계를 고려하지 않고 총 20개의 특징값을 신경망의 입력 노드로 사용하였으나 본 논문에서는 특징값의 상관관계를 고려하여 불필요한 특징값을 줄임으로써 총 16개의 입력 노드를 사용할 수 있었다. 실험 결과 정확도 저하는 없었으며 처리 속도는 입력 노드수가 줄어들어 많이 향상됨을 알 수 있었다. 신경망의 학습 데이터로는 Polhemus sensor를 얼굴에 부착하여 모니터상의 42개 시선 위치를 10명의 사용자가 각각 쳐다보는 순간 취득된 Polhemus 측정 데이터와 카메라를 통해 입력된 영상을 이용하였다. 이때 학습된 42 위치에 대한 얼굴 회전각이외의 회전각 역시 출력으로 나타낼 수 있도록 미분 가능하며 연속적인 출력함수를 사용하였다. 추정된 3차원 회전량의 정확도는 역시 Polhemus

sensor와 비교하였다.

(2) 얼굴의 3차원 이동량 추정

일반적으로 3차원 공간상에서 얼굴의 움직임은 얼굴의 회전(rotation)과 얼굴 축의 이동(translation)에 의하여 표현되며, 전술한 바와 같이 얼굴 축의 3차원 이동량은 얼굴 윤곽 무게중심의 2차원 이동량(그림 6의 특징값 17, 18)으로부터 파악될 수 있다. 일반적으로 카메라 영상에서 파악된 2차원 위치나 이동량으로부터 실제 3차원 위치 및 이동량을 구하기 위해서는 2차원에서 3차원 변환을 위한 각종 변수들, 카메라 파라미터들(a, T_y, T_z, f)과 3차원 Z축 거리(3D depth)등을 알아야 한다. 그런데 이 논문에서는 그림 7의 1 단계에서 카메라 파라미터들(a, T_y, T_z, f)과 모니터에서 얼굴까지의 Z축 거리를 자동으로 파악했기 때문에 이를 이용하여 얼굴 윤곽 무게 중심의 2차원 이동량으로부터 얼굴의 실제 3차원 이동량을 구하였다.

3-3. 3 단계 : 얼굴 특징점의 변화된 3차원 위치 파악

그림 7의 제 1 단계에서 구한 얼굴 특징점(양 눈, 입의 중심점)의 초기 3차원 위치는 모니터 좌표계에서 파악된 값이므로, 이 논문에서는 그림 7의 모니터와 얼굴 좌표계 간의 변환 식을 이용하여 얼굴 좌표계에서의 위치로 변환한다. 이렇게 변환된 얼굴 특징점의 3차원 좌표는 식 (12)와 같이 2 단계에서 구한 3차원 회전량(rotation) 및 이동량(translation)결과를 바탕으로 한 아핀 변환 과정(affine transformation)을 거쳐 사용자가 모니터상의 한 지점을 쳐다보는 순간 변화된 얼굴 특징점의 3차원 위치가 된다^[9].

$$\begin{bmatrix} x'_i \\ y'_i \\ z'_i \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha \\ 0 & -\sin \alpha & \cos \alpha \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \\ 0 \end{bmatrix} \quad (12)$$

($i = 1, 2, 3$) 왼쪽 눈, 오른쪽 눈, 입의 중심

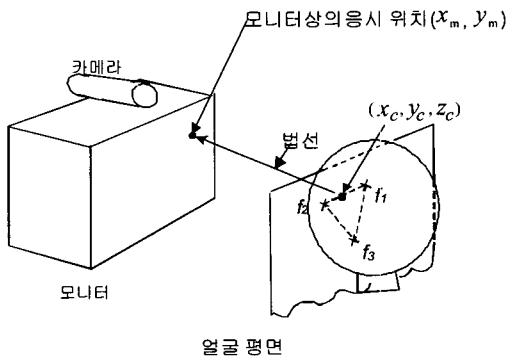
3-4. 4 단계 : 모니터상의 시선 위치 파악

식 (12)에 의해 변화된 얼굴 특징점의 3차원 위치(x'_1, y'_1, z'_1), (x'_2, y'_2, z'_2), (x'_3, y'_3, z'_3)는 다시 그림 8의 모니터와 얼굴 좌표계 간의 변환 식을 이용하여 모니터 좌표계를 기준으로 한 얼굴 특징점의 좌표값으로

변환한다. 이로부터 식 (13)과 같이 양 눈, 입의 중심점으로 구성된 하나의 얼굴 평면 방정식을 구할 수 있게 되며,

$$Ax + By + Cz = D \quad (13)$$

그림 12 및 식 (14)와 같이 얼굴 평면의 법선 벡터 방향 (A, B, C) 을 가지고 양 눈 (f_1, f_2) 의 중심점 (x_c, y_c, z_c) 을 지나는 직선과 모니터와 만나는 점이 사용자가 쳐다보는 모니터상의 시선 위치 (x_m, y_m) 가 된다.



$$x_m = -\frac{A}{C} * z_c + x_c, \quad y_m = -\frac{B}{C} * z_c + y_c \quad (14)$$

(단, $x_c = \frac{x'_1 + x'_2}{2}, y_c = \frac{y'_1 + y'_2}{2}, z_c = \frac{z'_1 + z'_2}{2}$)

그림 12. 모니터상의 시선 위치 파악
Fig. 12. Gaze detection on a monitor.

여기서 시선 벡터의 지점을 양 눈과 입의 무게중심으로 하지 않고 양 눈의 중심으로 정한 것은 사람의 시선 방향이 양 눈의 중심에 위치하기 때문이다.

III. 실험 결과

1. 카메라 보정에서 구한 파라미터의 정확도

그림 10에서 설명한 카메라 보정에 의해 구한 카메라 파라미터들의 정확도를 측정하기 위해, 구해진 파라미터들과 식 (11)에 의해 계산된 보정점들의 3차원 위치와 Polhemus sensor를 이용하여 측정된 보정점들의 실제 3차원 위치사이의 RMS error를 구하였다. 실험 결과 구해진 보정점들의 삼차원 위치와 Polhemus sensor 측정값과의 최소자승 에러(RMS error)는 표 1과 같다.

표 1. 보정점들의 삼차원 위치와 Polhemus sensor 측정값과의 RMS 에러

Table 1. The RMS errors between the real 3-D positions of calibration points and the measured ones by Polhemus sensor.

(단위 cm)		
X	Y	평균
0.079	0.1679	0.1237

2. 얼굴 특징점의 초기 삼차원 위치 추정의 정확도

19인치 모니터에서 50~70cm정도의 거리만큼 떨어져 있는 10명분의 사용자에게 대해 실험한 결과, 그림 11의 방법에 의해 보다 정확한 얼굴 특징점의 삼차원 위치 (평균 1.15 cm의 RMS 에러)를 얻을 수 있음을 알 수 있었다. 표 2는 보정된 얼굴 특징점들의 삼차원 위치와 Polhemus sensor에 의해 구한 특징점의 실제 위치의 예를 나타낸 것이며, 표 3은 보정된 얼굴 특징점의 삼

표 2. 보정된 얼굴 특징점의 삼차원 위치와 Polhemus sensor로부터 구한 특징점의 삼차원 위치의 예

Table 2. The real 3-D positions of facial features and the measured ones by Polhemus sensor.

방법	얼굴 특징점			실험자
	왼쪽 눈	오른쪽 눈	입의 중심	
보정된 얼굴 특징점의 삼차원 위치	(41, 104, 51.2)	(-41, 104, 51.2)	(0, 44, 51.4)	실험자 1
Polhemus sensor로부터 구한 삼차원 위치	(35, 109, 51.7)	(-36, 108, 51.7)	(0.1, 36, 52.2)	
보정된 얼굴 특징점의 삼차원 위치	(29, 101, 51.3)	(-29, 101, 51.3)	(0, 32, 51.6)	실험자 2
Polhemus sensor로부터 구한 삼차원 위치	(37, 11, 51.6)	(-36, 11, 51.7)	(0.4, 37, 52.6)	
보정된 얼굴 특징점의 삼차원 위치	(28, 106, 62.2)	(-28, 106, 62.2)	(0, 24, 62.3)	실험자 3
Polhemus sensor로부터 구한 삼차원 위치	(38, 11, 61.7)	(-38, 11.1, 61.8)	(0.6, 41, 63)	
보정된 얼굴 특징점의 삼차원 위치	(30, 105, 65.2)	(-30, 105, 65.2)	(0, 28, 65.4)	실험자 4
Polhemus sensor로부터 구한 삼차원 위치	(36, 9.9, 65.2)	(-37, 9.8, 65.2)	(0.7, 3.2, 65.9)	
보정된 얼굴 특징점의 삼차원 위치	(42, 107, 69.2)	(-42, 107, 69.2)	(0, 23, 69.5)	실험자 5
Polhemus sensor로부터 구한 삼차원 위치	(36, 9.7, 68.2)	(-35, 9.8, 68.2)	(0.5, 2.9, 68.4)	

차원 위치와 Polhemus sensor로부터 구한 위치와의 RMS 에러를 나타낸 것이다.

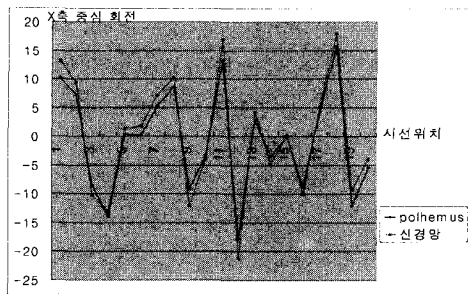
표 3. 보정된 얼굴 특징점의 삼차원 위치와 Polhemus sensor로부터 구한 특징점의 삼차원 위치와의 RMS 에러

Table 3. The RMS errors between the real 3-D positions of facial features and the measured ones by Polhemus sensor.

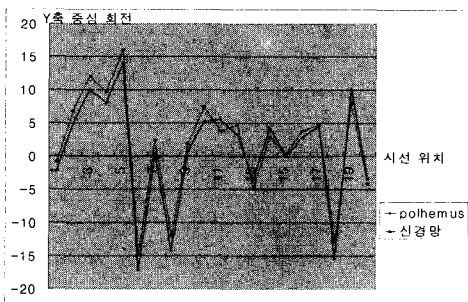
(단위 cm)

X축	Y축	Z축	평균
0.64	0.5	0.81	1.15

3. 신경망에 의한 얼굴의 3차원 회전량 추정의 정확도
실험 결과 신경망에 의해 추정된 회전각과 Polhemus sensor에 의해 측정된 회전각사이에는 평균 2.98 도의 RMS error가 존재함을 알 수 있었다. 그림 13은 한 실험자의 회전량 추정 결과를 나타낸 것이다.



(a) The estimation accuracy of rotation about X_axis(α)



(b) The estimation accuracy of rotation about Y_axis(β)

그림 13. 얼굴 회전량 추정의 정확도
Fig. 13. The accuracy of 3D rotation estimation by neural network.

4. 얼굴의 3차원 이동량 추정의 정확도
추정된 3차원 이동량의 정확도는 역시 Polhemus

sensor와 비교하였으며, 실험결과 평균 1.8cm(X축으로는 1.1 cm, Y축으로는 1.42cm)의 RMS error를 나타냈다. 이때 Y축 방향으로의 이동량 추정에러가 X축 방향으로의 에러보다 다소 큰 이유는, 얼굴의 Y축 방향(수직 방향)이동량이 큰 경우에 입력 영상에서 얼굴뿐 아니라 목의 일부 영역까지 포함되어, 얼굴 윤곽의 무게 중심 위치를 파악하는 데에 오차가 포함되었기 때문이다.

5. 모니터상의 시선 위치 추정의 정확도

그림 12에서 설명한 시선 위치 추정의 정확도는 19인치 모니터 앞에 약 50~70 cm떨어진 거리에서 10명의 사용자가 쳐다보는 실제 위치와 시선 위치 추적 알고리즘에 의해 파악된 위치사이의 RMS error로써 측정하였다. 이때 시선위치 추정의 정확도는 표 4와 같이 모니터상의 임의의 지점을 쳐다볼 때 얼굴의 회전만 존재하는 경우와 회전 및 이동이 같이 발생하는 경우로 나누어 실험하였다.

표 4. 모니터 상의 시선 위치 파악 정확도

Table 4. The gaze detection accuracies on a monitor.

(단위 : cm)

	얼굴의 회전만 존재하는 경우		얼굴의 회전과 이동이 같이 발생하는 경우	
	X축	Y축	X축	Y축
시선 위치 추적 에러	2.4	3.4	3.5	3.7
	4.2		5.1	

실험 결과, 얼굴의 회전만 존재하는 경우보다 회전과 이동이 같이 발생하는 경우에 시선 위치 에러가 조금 증가됨을 알 수 있었다. 이 에러는 입력 영상에서 얼굴 특징점 및 얼굴 윤곽 추출의 에러, 그림 7의 1, 2단계에서 구한 얼굴 특징점의 3차원 위치 및 움직임 추정 에러들이 합쳐진 결과이다. 실험 결과 기존의 연구^[12]에 비해 성능이 향상됨을 알 수 있었다. 또한 이 결과는 Rikert의 논문^[7]에서 나타낸 시선 위치 추적 성능(5.08 cm 에러)과 비슷한 결과를 나타낸다. 그러나 Rikert의 방법은 얼굴의 3차원 위치 및 움직임에 대한 고려 없이 카메라에서 관측된 2차원 얼굴 영상으로부터 모니터상의 시선 위치를 직접 파악하므로, 모니터와 사용자 얼굴까지의 거리는 항상 고정(50cm)시켜야한다는 단점

이 있으며, 얼굴의 자연스러운 움직임(회전 및 이동)이 발생하는 경우 시선위치추적에러가 증가되는 문제점이 있다. 동시에 그들의 방법은 사용자 얼굴의 뒤 배경에 복잡한 물체가 없는 것으로 제한 조건을 두고 있으며 처리 시간이 상당히 오래 걸리는 문제점이 있다 (333MHz alphastation에서 약 1분). 그러나 본 논문에서 제안하는 시선위치추적방법은 배경이 복잡한 사무실 환경에서도 사용가능하며, 약 3초 이내의 처리 시간 (200MHz Pentium PC)이 소요됨을 알 수 있었다. 또한 본 논문에서는 모니터에서 사용자 얼굴까지의 거리를 변화시켜가면서 모니터 상의 시선 위치 정확도를 측정 하였다. 실험 결과 모니터와 사용자 사이의 거리가 55 cm 일 때의 시선 위치 에러는 4.75cm, 60cm 일 때의 시선 위치 에러는 4.79cm, 65cm 일 때의 시선 위치에러는 4.80cm인 것으로 나타났으며, 이로부터 모니터와 사용자얼굴까지의 거리변화에 따른 시선 위치 에러 증가분은 거의 없는 것으로 조사되었다. 또한 사용자가 모니터 상의 위치를 쳐다볼 때 다양한 표정변화를 나타내는 경우에도 시선위치에러의 증가현상은 없는 것으로 조사되었다. 본 논문에서는 얼굴을 움직이지 않고 눈동자만 움직여서 시선방향을 바꾸는 것을 고려하지 않았으며, 이러한 경우에 대비하기 위해서는 향후 눈동자 움직임만을 추적하는 연구를 병행해야 할 것으로 기대된다.

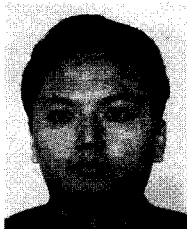
IV. 결 론

이 논문에서는 사용자 얼굴의 3차원 위치 및 자연스러운 움직임(회전 및 이동)을 컴퓨터 비전 방법에 의해 추정하고 이로부터 모니터상의 시선 위치를 파악할 수 있는 새로운 방법을 제안하였다. 실험 결과, 모니터와 사용자 얼굴까지의 거리를 50~70 cm정도 유지한 환경에서 19인치 모니터상의 입의 지점을 사용자가 쳐다 보았을 때, 약 2.01인치의 시선 위치 파악 에러 성능을 얻었다. 향후 얼굴 특징점 및 윤곽 추출의 정확도를 향상한다면 보다 정확한 시선 위치 추출 성능을 얻을 수 있을 것으로 기대된다.

참 고 문 헌

- [1] Jaihie Kim, K. R. Park, S. Leclair, "Process Control via Gaze Detection Technology", The Second International Conference on Intelligent Processing and Manufacturing of Materials, pp.1263-1269, July, 1999.
- [2] A. Azarbajejani, "Visually Controlled Graphics", IEEE Trans. PAMI, Vol. 15, No. 6, pp. 602-605, June, 1993.
- [3] T. Fukuhara, T. Murakami, "3D-motion estimation of human head for model-based image coding", IEE Proc., Vol. 140, No. 1, pp. 26-35, 1993.
- [4] P. Ballard, G. Stockman, "Controlling a Computer via Facial Aspect", IEEE Trans. on System Man and Cybernetics, Vol. 25, No.4, pp. 669-677, 1995.
- [5] A. Gee, R. Cipolla, "Fast visual tracking by temporal consensus", Image and Vision Computing, Vol. 14, pp. 105-114, 1996.
- [6] J. Heinzmann, A. Zelinsky, "3-D Facial Pose and Gaze Point Estimation using a Robust Real-Time Tracking Paradigm", Proceedings of the International Conference on Automatic Face and Gesture Recognition, pp. 142-147, 1998.
- [7] T. Rikert, M. Jones, "Gaze Estimation using Morphable Models", Proceedings of the International Conference on Automatic Face and Gesture Recognition, pp. 436-441, 1998.
- [8] A. Tomono, F. Kishino, "Gaze Point Detection Algorithm Based on Measuring 3D Positions of Face and Pupil", IEICE Transactions on Information and Systems, D-II, Vol. J75-D-II, No. 5, pp. 861-872, 1992. 5.
- [9] Ramesh Jain, Machine Vision, McGraw-Hill International Edition, 1995.
- [10] Steven C. Chapra, Raymond P. Canale, "Numerical Methods for Engineers", McGraw-Hill International Editions, 1989.
- [11] R. BROWN et al., Introduction To Random Signals and Applied Kalman Filtering. Wiley, Third Edition.
- [12] 박강령, 김재희, "단안 카메라 환경에서의 시선위치 추적", 한국통신학회 논문지, 제26권 10호 pp. 1390-1398, 2001년 10월

저 자 소 개



朴 康 領(正會員)

1994년 2월 : 연세대학교 전자 공
학과 졸업. 1996년 2월 : 연세대학
교 전자 공학과 석사. 2000년 3
월 : 연세대학교 전기·컴퓨터공학
과 박사. 2000년 3월~현재 : LG
전자 기술원 Digital Vision Group

선임 연구원. <관심분야 : Biometric영상처리, 패턴인식,
컴퓨터 vision>

金 在 薰(正會員)

1982년 8월 : Case Wetsern Reserve Univ. Electrical
Eng. 석사. 1984년 5월 : Case Wetsern Reserve Univ.
Electrical Eng. 박사. 1984년 3월~현재 : 연세대학교
전기·컴퓨터공학과 교수. <관심분야 : Biometric영상처
리, 패턴인식, 컴퓨터 vision>