얼굴의 움직임을 이용한 다중 모드 인터페이스에서의 응시 위치 추출

박강령, 남시욱, 한승철, 김재희
연세대학교 기계 전자공학부 인공지능 연구실

Gaze Detection Using Facial Movement in Multimodal Interface

Kang Ryoung Park, Si Wook Nam, Seung Chul Han, Gyeong Yong Heo and Jaihie Kim
AI Lab., School of Electrical and Mechanical Eng., Yonsei University,
134 Shinchon-Dong Sudaemoon-Ku, Seoul 120-749
E-mail : parkgr@seraph.yonsei.ac.kr

1. 서론 (Introduction)

시선의 추출을 통해 사용자의 관심 방향을 알고자 하는 연구는 여러 분야에 응용될 수 있는데, 대표적인 것이 장애인의 컴퓨터 이용이나, 다중 윈도우에서 마우스의 기능 대용 및, VR에서의 위치 추적 장비의 대용 그리고 원격 회의 시스템에서의 view controlling등이다. 기존의 대부분의 연구들에서는 얼굴의 입력된 동영상으로부터 얼굴의 3차원 움직임량(rotation, translation)을 구하는데 중점을 두고 있으나[1][2], 모니터, 카메라, 얼굴 좌표계간의 복잡한 변환 과정 때문에 이를 바탕으로 사용자의 응시 위치를 파악하고자하는 연구는 거의 이루어지지 않고 있다. 본 논문에서는 일반 사무실 환경에서 입력된 얼굴 동영상으로부터 얼굴 영역 및 얼굴내의 눈, 코, 입 영역 등을 추출함으로써 모니터의 일정 영역을 응시하는 순간 변화된 특징점들의 위치 및 특징점들이 형성하는 기하학적 모양의 변화를 바탕으로 응시 위치를 계산하였다. 이때 앞의 세 좌표계간의 복잡한 변환 관계를 해결하기 위하여, 신경망 구조(다층 퍼셉트론)을 이용하였다. 신경망의 학습 과정을 위해서는 모니터 화면을 15영역(가로 5등분, 세로 3등분)으로 분할하여 각 영역의 중심점을 응시할 때 추출된 특징점들을 사용하였다. 이때 학습된 15개의 응시 위치이외에 또 다른 응시 영역에 대한 출력값을 얻기 위해, 출력 함수로 연속적이고 미분가능한 함수(linear output function)를 사용하였다. 실험 결과 신경망을 이용한 응시위치 파악 결과가 선형 보간법[3]을 사용한 결과보다 정확한 성능을 나타냈다.

2. 얼굴 영역 및 얼굴내 특징점 추출

사용자의 응시 위치를 파악하기 위하여, 본 논문에서는 얼굴내의 특징점(양눈, 코, 입)의 위치 변화 및 특징점들이 형성하는 기하학적인 모양의 변화도를 이용한다. 이를 위해 본 논문에서는 먼저 얼굴 영역을 검출한 후 추출된 얼굴 영역내에 제한된 범위 내에서 양 눈과 코 및 입의 양 끝점을 추출한다.

2.1 차영상 정보와 칼라 정보를 이용한 얼굴 영역의 추출

본 논문에서는 시간적으로 연속된 두 영상간의 차영상 정보와 얼굴의 살색 정보를 이용하여 얼굴 영역을 검출한다. 이때 차영상 정보이외에 살색 정보를 같이 이용한 이유는 사용자의 뒷배경에서 움직임이 있는 물체를 얼굴 영역으로 오인식하는 경우를 막기 위해서이다. 입력된 얼굴의 살색 칼라 정보에 대한 RGB신호를 아래 식(1)과 같이 YIQ model로 변환함으로써 그림 1

처럼 얼굴의 살색 정보에 민감한 I성분 구간(110~150)을 바탕으로 얼굴 영역을 검출한다[4].

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.523 & 0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \cdots (1)$$

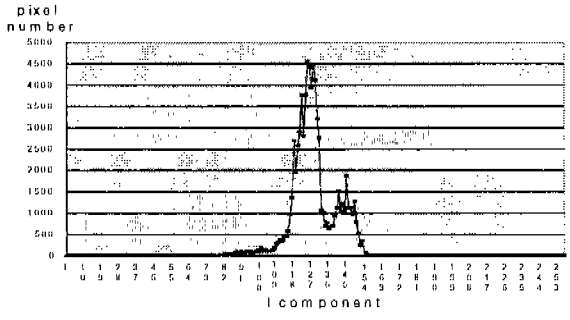


Fig. 1 얼굴 영역의 I 성분

이때, 차영상내에서 얼굴로 검출된 부분과 color model에서 얼굴로 검출된 부분에 대한 공통 부분(intersection)을 택함으로써 그림 2처럼 입력되는 얼굴 영상으로부터 빠르고 정확하게 얼굴 영역을 검출 할 수 있다.

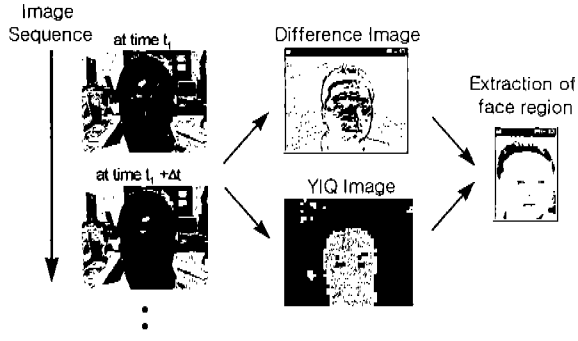


Fig. 2 얼굴 영역의 검출

2.2 수평·수직 히스토그램 분석법을 이용한 눈동자, 코 및 입의 양 끝점 추출

추출된 얼굴 영상은 히스토그램 평활화 및 이진화과정을 통해 이진 영상으로 변환한다. 이때, 아래 그림 3처럼 얼굴내의 눈의 위치에 대한 사전정보와 이진 영상에 대한 제한된 범위 내에서 수직, 수평히스토그램의 최고치를 계산함으로써 눈의 위치를 정확하게 추출할 수 있다. 양 눈이 검출된 후 그림 4처럼 입의 위치에 대한 존재 가능 범위를 설정한 후, 이 영역에 대한 이진화 및 수직방향 히스토그램으로 입선의 수직 위치를 먼저 추출한다. 추출된 입선의 수직 위치로부터 입의 양끝점을 추출하기 위해 입선에 대한 수평 히스토그램을 구하여 이 히스토그램이 어떤 threshold(수평히스토그램의 평균값)이상으로 갑작스럽게 변화되는 지점을 추출하여 이를 입의 양 끝점으로 결정한다. 콧구멍 역시 눈동자와 같은 방법으로 추출하였으며, 두 콧구멍의 근접도가 큰 관계로 양 콧구멍을 포함한 영역에 대한 수평 히스토그램을 통해 두 개의 최고치를 추출함으로써 양 콧구멍의 수평 위치를 파악해낸다.

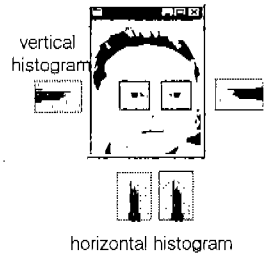


Fig. 3 눈 영역 검출

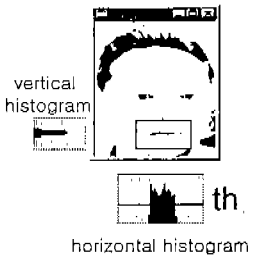


Fig. 4 입 영역 검출

다음 그림 5는 본 논문의 방법을 이용하여 추출된 사용자의 얼굴 영역 및 눈, 코, 입의 양 끝 위치를 나타낸 것이다.

Fig. 5 눈, 코, 입 영역 및 각 특징점을 추출하기 위한 탐색 영역

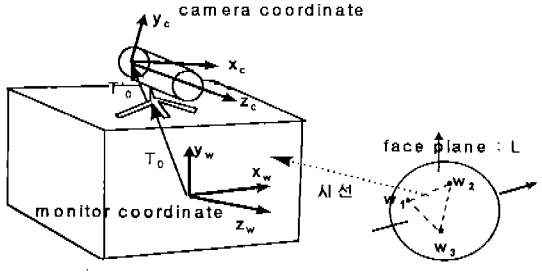
2.3 특징점의 움직임 추적

초기 영상에서의 특징점 추출 방법과는 달리 이후 연속 영상에서는 매번 얼굴 영역을 다시 추출하지 않고, 이전에 추출된 특징점부근을 탐색하는 방법을 이용하여 특징점의 움직임을 추적한다. 이때 이전에 추출된 특징점 부근에 정해진 크기의 window를 설정하여 이 window내의 히스토그램 분석을 통해 특징점들을 추적

한다.

3. 모니터, 카메라 및 얼굴 좌표계를 고려한 사용자의 응시 위치 파악

다음 그림 6과 같이 얼굴 좌표계에서의 3차원 좌표점(W_1, W_2, W_3)들로부터 얼굴 평면(L)이 결정되며, 이 평면의 normal vector가 모니터와 만나는 점이 사용자의 응시 위치가 된다. 그러나 실제의 경우, camera 좌표계에서 추출된 특징점으로부터 얼굴 좌표계의 3차원 좌표점(W_1, W_2, W_3)을 정확히 계산하기는 어려우며, 또한 이 경우 아래 식 (2)와 같이 사전에 알아야할 많은 parameter(예, 모니터와 사용자간의 거리, 카메라의 설치 각도 등)들이 있기 때문에 analytic한 방법으로 분석하기는 상당히 어렵다. 그러므로 본 논문에는 일반적으로 입출력간의 비선형 방정식을 해결하는데 우수한 능력을 발휘하는 신경망 구조(muti-layered perceptron)를 이용하여 카메라 좌표계에서 추출된 영상 정보로부터 사용자의 실제 응시 위치를 직접 구해낸다.



$$C = P \cdot T_0' \cdot R_\theta \cdot R_\alpha \cdot T_0 \cdot W \cdots (2)$$

C : camera에 투영된 Feature points

L : W_1, W_2, W_3로 구성된 얼굴 평면

T_0', T_0 : monitor coordinate에 대한 camera coordinate의 천이 행렬

R_θ, R_α : monitor coordinate에 대한 camera coordinate의 회전 행렬

P : 투영 변환 행렬

Fig. 6 모니터, 카메라, 얼굴 좌표계를 고려한 사용자의 응시 위치

4. 신경망 입력을 위한 특징값 및 정규화 과정

사용자의 응시 위치를 파악하기 위해 사용될 수 있는 얼굴내의 특징점으로는 양눈, 코, 입 그리고 귀 등을 이용할 수 있는데, 이중 귀의 경우는 얼굴이 좌우로 심하게 회전하게 되면 입력 영상으로부터 소실될 염려가 있으므로 본 논문에서는 양눈과 코, 입의 위치를 특징점으로 사용하였다. 아래의 그림 7.(a)와 그림 7.(b)는 각각 모니터의 정중앙과 일정 영역을 응시하는 순간에 추출된 특징점들의 위치를 나타낸 것이다. 이때, 추출된 특징점들로부터 응시 위치를 파악하기 위해 본 논문에서는 다음과 같은 20개의 특징값들을 신경망의 입력 노드로 사용하였다.

(a)모니터 정 중앙 응시 (b) 모니터 일정 영역 응시

Fig. 7 모니터 정중앙과 일정 영역을 응시하는 순간의 특징점의 위치 변화

▷ 모니터의 정중앙을 응시 할 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)

▷ 모니터의 일정 영역을 응시 할 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'_4 (입의 오른쪽 끝:X'_5, Y'_5)

특징값 1 ~ 5 : $X'_i - X_i$ (i = 1, 2, ... 5)
특징값 6 ~ 10 : $Y'_i - Y_i$ (i = 1, 2, ... 5)
특징값 11 : $S(\triangle P'_1P'_2P'_3) - S(\triangle P_1P_2P_3)$
특징값 12 : $S(\triangle P'_1P'_3P'_4) - S(\triangle P_1P_3P_4)$
특징값 13 : $S(\triangle P'_2P'_3P'_5) - S(\triangle P_2P_3P_5)$

특징값 14 : $S(\triangle P'_3P'_4P'_5) - S(\triangle P_3P_4P_5)$

특징값 15 : $S(\triangle P'_1P'_3P'_4)/S(\triangle P'_2P'_3P'_5) - S(\triangle P_1P_3P_4)/S(\triangle P_2P_3P_5)$

특징값 16 : $S(\triangle P'_3P'_4P'_5)/S(\triangle P'_1P'_2P'_3) - S(\triangle P_3P_4P_5)/S(\triangle P_1P_2P_3)$

특징값 17 : $\{(X'_1 + X'_4)/2 - X'_3\} - \{(X_1 + X_4)/2 - X_3\}$

특징값 18 : $\{X'_3 - (X'_2 + X'_5)/2\} - \{X_3 - (X_2 + X_5)/2\}$

특징값 19 : $\{(Y'_4 + Y'_5)/2 - Y'_3\} - \{(Y_4 + Y_5)/2 - Y_3\}$

특징값 20 : $\{Y'_3 - (Y'_1 + Y'_2)/2\} - \{Y_3 - (Y_1 + Y_2)/2\}$

그러나 사용자의 앉은 키와 모니터와의 거리에 따라 입력 특징값의 차이가 크다면 정확한 응시 위치를 나타낼 수 없을 것이다. 그러므로 본 논문에서는 정규화 과정을 통해 입력 특징값의 변화도를 수용하고자 한다. 그러나 실제의 경우 카메라가 모니터의 위에 설치되어 있는 관계로 사용자의 앉은 키에 따른 변화도는 크지 않은 결과를 나타냈으므로, 본 논문에서는 모니터와 사용자간의 거리에 따른 변화도만 정규화한다. 즉, 아래 식 (3)과 같이 초기에 사용자로 하여금 모니터의 15영역중 최우측 상단과 최좌측 하단을 응시하게 함으로써 얻어진 각 특징값들의 최대 최소치의 변위 차이로 입력 특징값들을 나눔으로써 정규화한다.

$$d_i = \frac{d_i}{|\max(d_i) - \min(d_i)|} \cdots (3)$$

단, $d_i\,(i=1,2,\ldots 20)$: 신경망 입력 특징값

5. 응시위치 파악을 위한 다층퍼셉트론

본 논문에서는 사용자의 응시위치 파악을 위해 그림 10과 같이 신경망 구조인 다층 퍼셉트론을 사용한다. 이때 신경망의 입력 노드로는 앞에서 설명한 20개의 특징값들을 사용하고, 출력노드를 통해 모니터의 X, Y 축 응시 위치를 나타낸다. 학습 데이터로는 19인치 모니터를 15등분(가로 5등분, 세로 3등분)하여 각 영역의중심점을 응시할 때 측정된 특징값들을 사용하며 출력함수로는 학습된 15영역이외의 응시 영역에 대한 정확한 출력값을 나타낼 수 있도록 미분 가능하며 연속적인 output function들을 사용하였다.

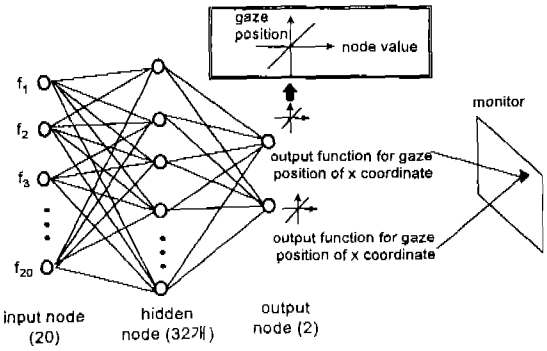


Fig. 8 다층 퍼셉트론 구조

신경망의 학습을 위해서는 식 (4)와 같은 generalized delta rule을 사용하였으며 총 10000~50000번의 반복 횟수등에 대하여 실험하였다.

$$w_{kj}(t+1) = w_{kj}(t) + \eta\ \delta_{pk}\ i_{pj} \cdots (4)$$

$w_{kj}(t)$: 반복 횟수가 t일때 각 노드에서의 가중치값

η : learning rate parameter (=0.01)

δ_{pk} : hidden units 혹은 output units에서의 error terms

i_{pj} : input layer 혹은 hidden layer에서 계산된 출력값

6. 실험 결과

신경망의 학습 과정에는 다양한 앉은 키와 자세를 갖는 총 150개의 학습 sample(10명분×15응시 영역)을 사용하였으며, 모니터의 각 영역에 대한 응시 위치의 정확도는 학습에 사용하지 않은 10명분의 test 데이터를 이용하여 실험하였다. 영상 크기는 320×240이며, pentium pro 200hz과 window 95환경에서 실험하였다. 영상 입력 장치로는 camcoder와 RT 300 video blaster capture board를 사용하였다. 매 영상마다 특징점들을 추출하는데 소요되는 시간은 0.08 sec정도이며, 신경망 출력을 통해 사용자의 응시위치를 파악하는데

걸리는 시간까지 합쳐서 0.09~0.10 sec 정도 소요됐다. 실제, online test에서는 camcoder를 통한 영상의 입출력 시간(0.11sec)까지 합하여 4~5 frames/sec의 처리 속도로 사용자의 응시위치에 대한 결과를 나타냈다. 향후 보다 고가의 영상 입출력 장비를 사용한다면 처리속도를 보다 향상시킬 수 있을 것으로 기대된다. 이때 본 연구 방법으로 입력 영상내에서 추출된 각 특징점들의 위치와 손으로 직접 찍은 위치와의 rms error는 다음과 같다.

(단위 pixel)

양 눈		양 콧구멍		입의 양끝점	
X축	3.97	X축	4.50	X축	4.2
Y축	3.80	Y축	4.01	Y축	3.76
rms error	5.49	rms error	6.03	rms error	5.64

※ 평균 얼굴 크기 (x축 144, y축 208)
평균 눈 크기 (x축 8, y축 4)
평균 콧구멍 크기 (x축 4, y축 4)
평균 입의 양 끝점의 크기(x축 2, y축 2)

Table 1. 평균 rms error

실험 결과에서 양 콧구멍의 위치에 대한 rms error가 가장 높았는데 이는 양 코 옆의 그림자 부분을 콧구멍으로 인식하는 경우가 많이 발생했기 때문이다. 신경망의 학습시에는 15영역이외의 응시 위치에 대한 결과값도 구하기 위하여 미분가능하고 연속적인 output function(linear output function)에대하여 실험하였다. 각각의 출력 함수들에 대해 실험은 두 가지로 나누어 하였다. 즉 학습 응시 위치와 testing 응시 위치가 같은 경우와 학습 위치는 앞의 15응시영역으로 하고 testing 위치가 다른 경우(10명분 × 17응시 영역)로 나누어 아래 표2와 같이 실제 응시 위치와 신경망에서 출력된 응시 위치와의 차이값을 계산하였다.

(단위 : inch)

linear interpolation method		neural network using linear function	
train data	1.83	train data	1.101
test data	1.84	test data	1.531

(a) 학습 응시 위치가 testing 응시 위치와 같은 경우

linear interpolation method	neural network using linear function
1.87	1.431

(b) 학습 응시 위치가 testing응시 위치와 다른 경우

Table 2. 선형 보간법을 사용하였을 경우와 linear output 신경망을 사용하였을 경우의 응시 위치 정확도

두가지 실험 결과를 평균하였을 경우 linear output function을 사용하였을 경우 선형 보간법보다 정확한 응시 위치 결과를 나타냈다(약 1.5인치 error). 이로부터 입출력간의 비선형관계를 해결하는데 신경망이 우수한 성능을 나타냄을 알 수 있었다. 또한, 본 연구에서는 실험 환경을 3차원 그래픽 워크스테이션(Silicon graphic ws : Indigo-II) 으로 확장하여 그림 9와 같이 사용자의 응시 위치 결과에 따라 3차원 그래픽 view를 moving 시키는 환경을 구현하였다. 3차원 그래픽은 SGI전용 open inventor로 작성하였으며, 실험 결과 기존의 mouse를 사용하였을 경우보다 얼굴의 움직임으로 3D view를 조종하였을 경우 보다 현실감나는 가상현실 환경을 제공함을 알 수 있었다.

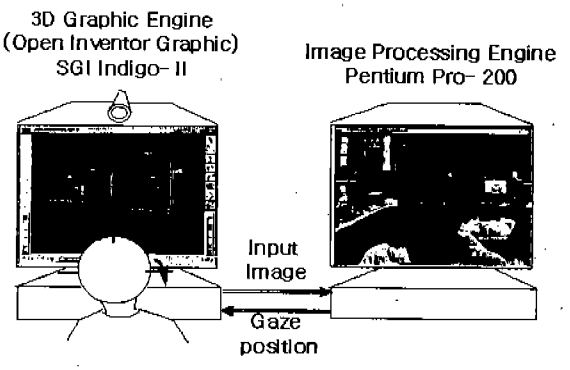


Fig. 9 가상 현실 환경에서 얼굴의 응시 위치 결과를

이용한 3D view controlling

7. 결론

본 논문에서는 입력 영상으로부터 얼굴 및 눈, 코, 입 영역 등을 추출하고, 추출된 특징점들의 위치 변화 및 각 특징점들이 형성하는 기하학적인 모양의 변화를 입력으로한 신경망 구조를 이용함으로써 그 사람의 실제 응시 위치를 파악하는 방법에 대하여 연구하였다. 실험 결과 linear output을 사용한 다층퍼셉트론 구조가 기타 다른 output function을 사용한 신경망 구조보다 우수한 성능을 나타냈으며, 선형 보간법을 이용한 결과보다도 정확한 성능을 나타냈다. 그러나 testing시 사용자의 자세가 심하게 움직이면 응시 위치 에러가 증가되는 문제점도 있었다. 향후, 이러한 사용자의 자세 변화도를 감지하여 효과적으로 보정해준다면 보다 정확한 응시위치 파악 결과를 얻을 수 있을 것으로 기대된다.

8. References (참고문헌)

[1] A. Azarbayejani, "Visually Contolled Graphics", in Proc. IEEE PAMI, Vol. 15, No. 6, June 1993

[2] Andrew Kiruluta, "Predictive Head Movement Tracking Using Kalman Filter", in Proc. IEEE Trans. on SMC, Vol.27, No.2, April 1997

[3] 박강령, 남시욱, 한승철, 김재희, "2차원 영상 정보와 선형 보간법을 이용한 사용자의 응시 위치 파악", 제 6회 인공지능, 신경망, 퍼지 시스템 종합학술대회 논문집, pp.255-258

[4] 남시욱, 박강령, 정진영, 김재희, "얼굴의 칼라정보와 움직임 정보를 이용한 얼굴 영역 추출"
1997년 대한전자공학회 하계 종합학술대회 논문집 pp.905-908