

범용 USB PC 카메라를 이용한 얼굴 특징점의 추적

양정석⁰ 이철우
전남대학교 컴퓨터공학과 지능영상 미디어연구실
{island33, leecw}@kiss.or.kr

Facial Feature Tracking from a General USB PC Camera

Jeong-Seok Yang⁰ Chil-Woo Lee
Intelligent Image Media Lab., Dept. of Computer Engineering, CNU.

요 약

In this paper, we describe an real-time facial feature tracker. We only used a general USB PC Camera without a frame grabber. The system has achieved a rate of 8+ frames/second without any low-level library support. It tracks pupils, nostrils and corners of the lip. The signal from USB Camera is YUV 4:2:0 vertical Format. we converted the signal into RGB color model to display the image and We interpolated V channel of the signal to be used for extracting a facial region. and we analysis 2D blob features in the Y channel, the luminance of the image with geometric restriction to locate each facial feature within the detected facial region. Our method is so simple and intuitive that we can make the system work in real-time.

1. 서 론

이 논문에서는 얼굴의 특징점(눈, 콧구멍, 입가장자리)를 찾는 방법에 대하여 기술한다. 얼굴의 특징점을 찾는 연구는 아주 오래 전부터 수행 되어온 연구 분야중의 하나이다. 얼굴의 특징점 탐색은 그 자체가 갖는 의미보다도 이러한 특징점을 이용해서 머리 포즈의 인식, 얼굴 표정 인식등 다른 어플리케이션제작에 응용될 수 있다는 점에 더욱 의미가 있다고 할 수 있다. 얼굴 특징점을 찾기 위한 기존의 연구들을 살펴 보면 다음과 같다. Henry A. Rowley와 Shumeet Baluja와 Takeo Kanade는 Gray-scale image에서 영상을 각자 작은 윈도우로 나누고 각각의 윈도우에 얼굴 영역이 포함되어 있는지를 판단하기 위하여 신경망을 이용하였다.[1] 이는 학습과정이 꼭 필요하다. Nuria Oliver, Alex P. Pentland와 Francois Berard는 pixel clustering과 2D blob features을 이용하여 얼굴 특징을 찾고, 입모양을 특징 벡터로 하여 HMM을 이용 하여 얼굴표정이나 행동양식을 인식하는데 사용하였다.[2] Yuille D.S.는 모델의 크기, 휘도치등의 영향을 받지 않은 Deformable Templates를 제안했다. 이 방법은 초기 템플릿에 대한 정보가 필요하다라는 단점이있다.[3] 그리고 이 논문에서 사용하는 방법과 비슷한 특징 후보들을 찾고 그 후보들 중에서 기하학적인 구조를 이용하여 특징점을 결정하는 방법등이 있다[4,5]

이 논문에서는 YUV color모델을 이용하여 얼굴영역을 찾고 그 영역안에서 2D 블랍 특징들을 분석하여 실시간

에 얼굴 특징점 후보를 찾고 이를 각 특징점의 위치 분포 제약을 이용하여 특징점을 찾아내는 알고리즘을 제안한다. 본 논문의 구성은 다음과 같다. 2장에서는 얼굴영역 검출및 특징점 검출에 사용할 영상신호의 생성과 얼굴영역의 검출 방법 관해서 기술하고, 3장에서는 얼굴영역안에서 각각의 얼굴 특징점을 찾는 방법에 대해서 기술하고, 4장에서는 실험환경과 잘 못 찾아진 결과들에 대해서 언급한다.

2. YUV신호를 이용한 얼굴 영역 검출

색상정보를 이용하기 위해서는 RGB 컬러모델 보다는 YIQ나 YUV, HSI등의 컬러 모델이 적합하다. 범용 PC 카메라를 이용할 경우 카메라로부터 얻을 수 있는 신호는 YUV 4:2:0(v) format이다. 이중 Y 채널은 휘도치를 나타내고 U와 V채널이 색상정보를 나타낸다. U 채널보다는 V 채널이 살색 정보에 민감하다. 그래서 V 채널을 이용하였다. V 채널을 이용하기 위해서는 보간이 필요하다. V채널을 보간하기 위하여 인접 화소 보간법을 사용하였다. Y 채널은 얼굴 특징점을 탐색하기 위해 사용되지만 U채널은 화면에 영상을 나타내기 위하여 RGB 영상으로 변환할 때만 사용되었다. 그리고 얼굴 영역을 설정하기 위하여 보간된 V채널에 대하여 상한값과 하한값을 설정하는 Clipping을 이용하였다. 그리고 Opening, Closing같은 처리를 한후 가장 큰 블랍을 얼굴 영역으로 설정하였다.

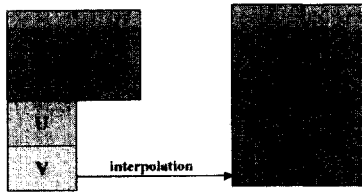


그림1. Y와 V 영상신호의 획득

3. 얼굴특징점의 추출

얼굴 특징점을 찾기 위한 이진화, 레이블링, 블랍 분석 등의 모든 처리는 미리 찾아진 얼굴 영역안에서 이루어진다. 이렇게 처리 영역을 한정함으로써 계산량을 현저히 줄일 수 있다.

3.1 눈알(pupils)의 위치 결정

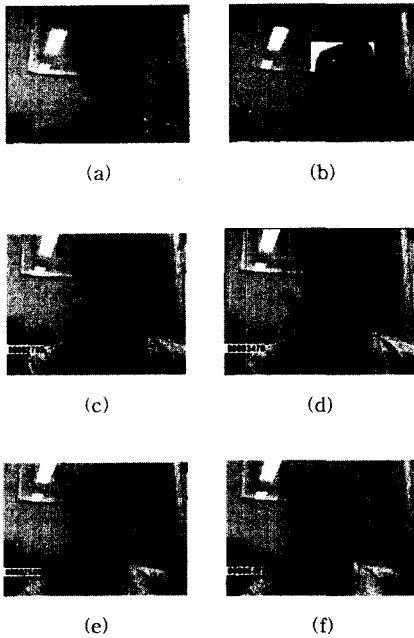


그림2. (a) V channel을 Clipping하여 찾은얼굴. (b) Y채널을 눈 후보영역에 대하여 이진화 시킨 영상. (c)와(d) 카메라와의 거리가 가까울 때. (e)와(f) 카메라와의 거리가 멀때

눈알의 위치를 찾기 위해서 얼굴이 가지고 있는 위치 제약을 최대한 활용했다. 눈을 찾기 위해서 얼굴 후보영역의 위쪽 1/3에 해당하는 영역의 Y채널을 이진화 했다. 영상 전처리(Opening, Closing)을 적용한 후에 그림1의 (b)에 보이는 것처럼 6개 이상의 블랍이 나타났다. 각

각의 블랍은 머리카락, 눈썹, 눈 그리고 기타 잡영(noise)이 나타났다. 이 중에서 눈의 블랍을 찾기 위해 아래의 조건을 이용하였다. i 와 j 는 각각 블랍 인덱스를 나타내고, DISTANCE는 거리 함수, XCOORD는 가로좌표, YCOORD는 세로좌표를 나타낸다. 식(1)은 블랍을 노이즈로부터 구별하는 식이고, 식(2)는 왼쪽 오른쪽을 구분하기 위한 식이다. 식(3)은 왼쪽 눈과 오른쪽 눈 사이의 거리제약이며 Thre1, Thre2, Thre3은 각각 거리 제약을 위한 문턱치이고 얼굴영역의 길이를 이용하여 유동적으로 결정된다. 식(4)는 가장 아래쪽에 있는 블랍을 찾기 위함이다. 그림2는 다양한 한 방향과 거리에서의 눈의 위치 탐색 결과를 나타내고 있다.

$$\begin{aligned} \text{Min} < \text{Size}(i) < \text{Max} & \quad (1) \\ \text{XCOORD}(i) < \text{XCOORD}(j) & \quad (2) \\ \text{Thre1} < \text{DISTANCE}(\text{COB}(i), \text{COB}(j)) < \text{Thre2} & \quad (3) \\ \text{if } \{ \text{DISTANCE}(\text{XCOORD}(i), \text{XCOORD}(j)) < \text{Thre3} \\ \text{AND } \text{YCOORD}(i) < \text{YCOORD}(i+1) \} \text{ Then} & \\ i \leftarrow i+1 & \quad (4) \end{aligned}$$

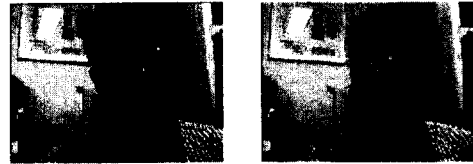


그림3. 입술라인과 콧구멍의 예상 위치 판독

3.2 콧구멍의 위치 결정

눈의 위치가 찾지면 얼굴의 영역과 눈의 위치를 이용하여 콧구멍과 입술라인의 대략적인 위치를 예측할 수 있다. 그리고 대략적인 위치주변에서 Vertical Profile의 최고치 위치를 구하여 정확한 입술라인과 콧구멍의 위치를 예측할 수 있다. 찾아진 입술라인과 콧구멍의 예상 위치를 그림2에서 확인 할 수 있다. 입술라인과 눈의 위치를 기준으로 콧구멍의 대략적인 위치를 예측하고 그 주변의 블랍을 분석하여 콧구멍을 찾는다. 블랍해석의 의사코드는 아래와 같다.

```
While( 블랍수 > 2 )
    Erosion을 수행
End While

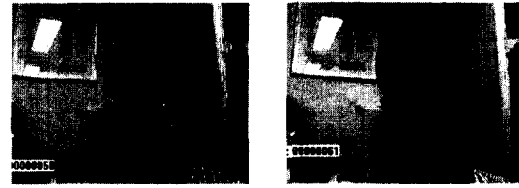
If(블랍수 == 2) Then
    왼쪽콧구멍위치 = 왼쪽블랍의 위치
    오른쪽콧구멍위치 = 오른쪽블랍은
End If
```

3.3 입가장자리의 추출

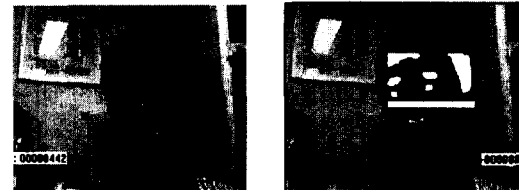
입가장자리(Lip Corner)의 위치는 Vertical Profile을 이용하여 안정적으로 찾을 수가 있었다. 그래서 립라인을 따라서 중앙에서 바깥쪽으로 코너를 탐색해 나간다. 왼쪽 모서리를 탐색할때는 이진화된 영상에서 값이0인 위치를 찾아서 라인을 따라 왼쪽으로 탐색을 수행한다. 그리고 왼쪽 끝에 이르렀을때 위와 아래 부분의 최소 값이 0이 아닌 값인지를 판별하여 그 쪽 방향으로 탐색을 계속해 나간다. 그래서 입술의 왼쪽 끝자리에 이르렀을때 탐색을 멈추게 된다. 오른쪽도 같은 방법을 사용하였다.

4. 실험 및 결론

실험에는 320 X 240 크기의 영상을 사용하였고, 카메라는 Philips PC Camera를 이용하였다. 그림3에서는 다양한 포즈와 카메라와의 거리에서의 탐색 결과를 확인할 수 있다. 그림4는 대표적인 에러를 보여 준다. 그림4의 (a)는 오른쪽 눈의 블랍과 눈썹의 블랍이 연결되어서 눈의 위치를 눈썹과 눈의 사이로 판정 한 결과이다. (b)는 얼굴 영역 자체의 탐색에 실패한 경우이고 콧구멍과 Lip corner가 찾아진 것은 탐색조건에 맞지 않는 경우 전의 위치를 유지하고 있는 것이다. (c)는 콧구멍후보 라인을 잘 못 찾은 것을 보여준다. 이러한 이유 때문에 콧구멍은 후보 위치를 이용하지 않고 눈의 위치와 입술의 위치를 기준으로 예측한 위치를 사용했다. 그래서 콧구멍의 위치를 안정적으로 찾을 수 있었다. (d)는 얼굴 영역을 잘못 찾아서 이진화 대상 영역에서 입술 부분이 제외 된 것을 보여 주고있다 이 경우에도 입술 코너 부분의 탐색은 실패하게 된다.



(a) (b)



(c) (d)

그림5. 잘못된 탐색 결과

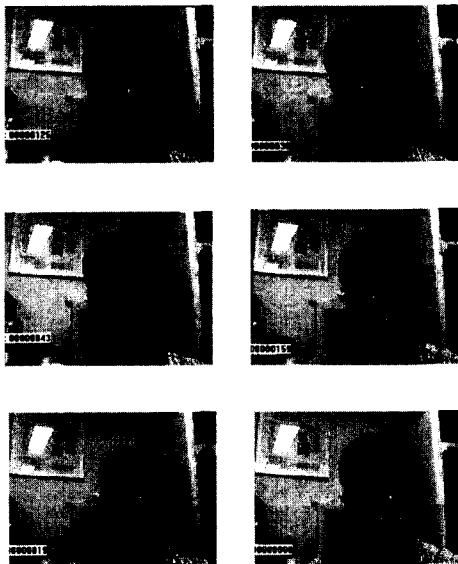


그림4. 다양한 거리와 포즈에서의 탐색 결과

참고문헌

- [1] Henry A. Rowley and Shumeet Baluja and Takeo Kanade "Human Face Detection in Visual Scenes", Technical Report CMU-CS-95-158R, CS department, CMU, 1995.
- [2] N. Oliver, A. Pentland and F. B'erard, "LAFTER: Lips and face real time tracker with facial expression recognition.", Proc. CVPR, 97.
- [3] A.L. Yuille, D.S. Cohen and P.W. Halinan, "Feature extraction from face using deformable template", Proc. IEEE Computer Soc. Conf. On computer Vision and Patt. Recog., 1989, PP. 104-109.
- [4] T. Leung, M. Burl, and P. Perona, "Finding faces in cluttered scenes using labelled random graph matching." In Proc. 5th Int. Conf. on Comp. Vision, pages 637--644, MIT, Boston, 1995.
- [5] Y. Sumi and Y. Ohta. Detection of face orientation and facial components using distributed appearance modeling. In Proc. Int. Workshop on Auto. Face and Gesture Recog., pages 254--259, Zurich, 1995.
- [6] Goto T., Kshirsagar, S., and Magnenat-Thalman, N., "Real Time Facial Feature Tracking and Speech Acquisition for Cloned Head", IEEE Signal Processing Magazine, Special Issue on Immersive Interactive Technologies, 2001.