얼굴의 움직임을 이용한 응시점 추적

0

고종국*, R.S.Ramakrishna**
광주과학기술원 정보통신공학과
E-mail : jgko@geguri.kjist.ac.kr*, rsr@kjist.ac.kr**

Head Orientation-based Gaze Tracking

0

Jong-Gook Ko and R.S.Ramakrishna
Department of Information and Communications
Kwang-Ju Institute of Science and Technology(K-JIST)

Abstract

본 논문에서 우리는 제약이 없는 배경화면에서 얼굴의 움직임을 이용한 응시점 추적을 위해 얼굴의 특징점(눈,코,그리고 입)들을 찾고 head orientation 을 구하는 효과적이고 빠른 방법을 제안한다. 얼굴을 찾는 방법이 많이 연구 되어 오고 있으나 많은 부분이 효과적이지 못하거나 제한적인 사항을 필요로 한다. 본 논문에서 제안한 방법은 이진화된 이미지에 기초하고 완전 그래프 매칭을 이용한 유사성을 구하는 방법이다. 즉, 임의의 임계치 값에 의해 이진화된 이미지를 레이블링 한후 각 쌍의 블록에 대한 유사성을 구한다. 이때 두 눈과 가장 유사성을 갖는 두 블록을 눈으로 선택한다. 눈을 찾은후 입고 코를 찾아간다. 360*240 이미지의 평균 처리 속도는 0.2초 이내이고 다음 탐색영역을 예상하여 탐색영역을 줄일경우 평균 처리속도는 0.15초 이내였다. 그리고 본 논문에서는 얼굴의 움직임을 구하기 위해 각 특징점들이 이루는 각을 기준으로 한 템플릿 매칭을 이용했다. 실험은 다양한 조명환경과 여러 사용자를 대상으로 이루어졌고 속도와 정확성면에서 좋은결과를 보였다. 또한, 명암정보만을 사용하므로 흑백카메라에서도 사용가능하여 경제적 효과도 기대할수 있다.

Keywords: Thresholding, Labeling, Facial Feature Tracking, HCI, Head Pose Estimation.

I. INTRODUCTION

Multimodal user interface is attracting special attention in recent times, including hand/ head gesture, facial expression, voice and eye gaze. Conventional human computer interaction techniques such as keyboard and mouse are being seen as a bottleneck in the information flow between humans and computer systems. In many speech recognition systems, voice signals are recognized with high success rates. But, we can expect better recognition ratio in environment with noise such as car using eye-gaze and lip-reading.

Human gaze has also the potential to be a fast input mode of computers. Eye-head controlled interface is used in a wide array of applications: Computer Interface, Virtual Reality and Games, Robot Control, Disabled Aid, Behavioral Psychology, Teaching and Presentation and so on. Facial features locating capability is needed in all applications.

In this paper, we propose a facial features tracking and head pose estimation·schemes in order to do construct a novel image-based human computer interface controlled by eye and head, which is a subtask of a multimodal and intelligent interface of a car navigation system.

This paper consists of following. A brief description of related work is contained in section 2. Section 3 and section 4 describe the proposed method of locating the facial features and of head pose estimation respectively. Experimental results are provided in section 5. The paper concludes with section 6.

II. RELATED WORK

Due attention is being paid by the research community to face detection schemes, several kinds of approach to locate facial features have been proposed in this regard.

Template matching method that was introduced by Yuille D.S.[1] uses deformable templates. This method is independent of size, slope, and illumination. But, at first, it requires knowledge of initial template of face.

Feature-based approach searches the image for a set of facial features and groups them into face candidates based on their geometrical relationship. Yow and Cipolla[2], Leung et.al.[3] and Sumi and Ohta employed this approach.

The color-based detection system[4][5] selects pixels that have similarity to skin color, and subsequently defines a subregion as a face if it contains a large of skin color pixels. But different camera conditions produce significantly different color values even for the same person under the same lighting condition and human feature colors differ from person to person.

In head pose estimation, Feature-based method and template matching method are also used[6].

A significant fact is that the iris and the pupil are darker than any other features except hair. The idea is to use this information for locating facial features and we employ template matching method for head pose estimation.

III. FACIAL FEATURES LOCATING ALGORITHM

1. Locating the eyes

1.1 Thresholding

At first, the image should be binarized by a proper threshold value. It is important to find the proper threshold value in order to separate the eyes, nostrils and mouth from face. There are many methods to find the threshold value. We employed a heuristic P-Tile method. After finding the weight center of histogram, the value is subtracted by constant value until the eyes is located for the first time.

1.2 Finding the candidates of the eyes

Edge detection is employed to find the candidates for the eyes. But, it requires amount of computation intensive and it is difficult to find accurate edge pixels. A thresholding method was employed instead to get the candidate blocks of the eyes. We can assign unique a tag to each isolated block by labeling the binarized image. In finding the candidates of the eyes, eliminating the blocks that is not satisfied in condition of being the eyes is much efficient than the finding the proper block which is satisfied in condition of being the eye. So we need standard as follows:

Suppose that the two points [x1,y1],[x2,y2] are the top-left point and bottom-right point of a circumscribed rectangle respectively. Let l(x,y) be the tag of the pixel.

$$(i)\ Size(i) = \sum_{x=x1}^{x2} \sum_{y=y1}^{y2} F(l(x,y))$$

$$(if\ \ l(x,y) = i \quad then\ \ F(i) = 1)$$

$$Min \le Size(i) \le Max$$

$$(ii)\ Ratio = Max\_Vertical \,/\, Max\_Horizantal$$

$$Ratio \le 1$$

If the block does not satisfy the conditions (i) and (ii) , then the block is eliminated from the candidate set. Condition (i) implies that if the size of eye's block is between Max and Min value. By eliminating the blocks using the rough and simple size information, we can reduce the number of candidate blocks to a quarter. Condition (ii) means that the aspect ratio of the eye is less than 1.

1.3 Looking for similarity by Complete Graph Matching

After eliminating the unsatisfactory blocks, a complete graph is composed with the candidate blocks and similarity for each pair is computed. The standard for computing similarity is like following. Similarity is computed as follows:

$$1) Normal\_size(i,j) = Size(i) \,/\, Size(j)$$

$$2) Normal\_Average(i,j) = \frac{Average\_gray(i)}{Average\_gray(j)}$$

$$3) Normal\_Aspect\_ratio(i,j) = \frac{A.R(i)}{A.R(j)}$$

$$4) Normal\_Angle(i,j) = 1 - [y_{distance} \,/\, x_{distance}]$$

Normal_size(i,j) refers to similarity of two blocks in size while Normal_Average(i,j) and Normal_Aspect_ratio(i,j) refer to similarity of average gray value and aspect ratio between the blocks respectively. The small value is divided by larger value for normalization. Normal_Angle means the slope over x-axis. The pair of blocks that have the maximum sum of the above four factors are selected as the two eyes.

2. Locating the Mouth and Lip-Corners

We can define a rough region for the mouth by using the eye information. Figure 1 shows an example. We consider the largest blocks to be mouth in that region. After locating the mouth's block , we can find the lip-corners by scanning the first and last columns.

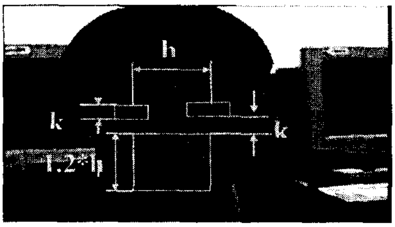

Figure 1. Defining the Mouth Region

3. Locating Nostrils

We can define the region for nostrils using the two eyes and the mouth position information. But we should examine whether the nostrils in the image are appeared in one block or not.

4. Verification

After locating the facial features such as the eyes, lip-corners and nostrils, we should check whether the facial features have been located correctly using the geometrical information. For example, we can prevent the eye brow from being selected as the eyes using the information that there are eye blocks under the eye brows.

IV. Head Orientation-based Gaze Tracking

Template Matching Method

We employed template matching using the angles of each pair of facial features. Each template consists of 6 angles like following Figure 2.

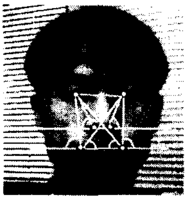

Figure 2. The angles of templates

Once we have created database of 11 templates representing different poses and gaze point. We do not need consider the distance from user to camera because the angles are independent of the distance. We compare the angles of input image with those of each template like following evaluation function:

$$E\_F(i)=\sqrt{\sum_{x=1}^{6}(T\_a(x)-I\_a(x))^2}$$

T_a(x) : xth angle of template image
I_a(x) : xth angle of input image
E_F(i) : evaluation value for i th template

The template that has the minimum evaluation value is selected as gaze point.

V. Experimental Results

Experiments were conducted on a single-processor, 166MHz Pentium PC equipped with CCD camera and Coreco Ultra II frame grabber. Experimental results show that we can locate and track the eyes, the nostrils, and lip-corners in images with different resolutions and different illuminations in real-time as soon as the face appears in the field of the view of the camera. The accuracy is above 95% without any identifying mark on the user's face. We have also tested person wearing the glasses or not wearing the glasses. In the case of the subject' black glasses, unsatisfactory results are returned. And if the people have a mustache, then we can not locate the mouth exactly. Some experimental results are shown Figure 3 and you can see the result of gazing using head orientation in Figure 4 and the rectangles mean the gaze points.

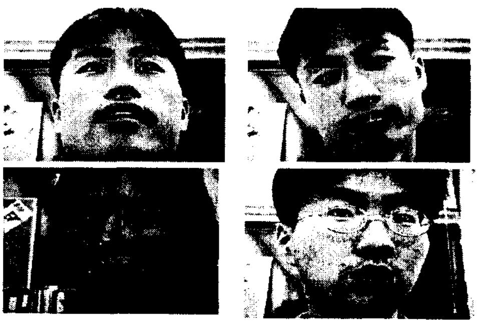

Figure 3. Experimental Results of Facial Features tracking

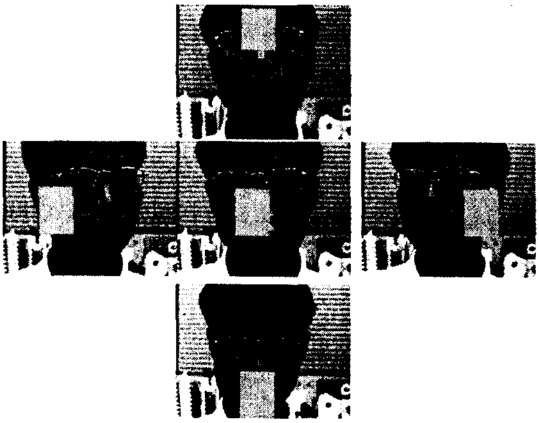

Figure 4. Experimental Results of Gazing

VI. Conclusions and Future Work

Real-time facial feature tracking and head pose estimation for eye and head controlled human computer interface has been proposed. More intelligent gray-level threholding methods and verification techniques are desirable and more distinct features for head orientation must be included into the final target system. We can also expect better interface in car navigation systems that incorporate eye-gaze information.

References

[1] 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.

[2] K.C.Yow, R.Cipolla, "Finding initial estimates of human face location, in Proc. 2nd Asian Conf. On Computer Vision, vol. 3. Singapore, 1995, PP.514-518.

[3] 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, PP. 637-644, MIT, Boston, 1995.

[4] Q.Chen, H.Wu, and M.Yachida. "Face detection by fuzzy pattern matching. In Proc. 5th Int. Conf. on Comp. Vision, pages 591-596, MIT, Boston, 1995.

[5] Y.Dai and Y.Nakano. "Face-texture model-based on SGLD and its application in face detection in a color scence. Pattern Recognition, 29(6):1007-1017, 1996.

[6] R.Lopez and T.S. Huang."3D Head Pose Computation from 2D Images: Templates Versus Features", ICIP,pages 220-224,WashingtonDC,Oct. 1995. IEEE.