Facial Feature Tracking for Eye-Head Controlled Human Computer Interface

Jong-Gook Ko*, Kyung-Nam Kim and R.S.Ramakrishna*****

Department of Information and Communications
Kwang-Ju Institute of Science and Technology (K-JIST)

E-mail: jgko@geguri.kjist.ac.kr*, knkim@kjist.ac.kr**, and rsr@kjist.ac.kr***

Abstract
*In this paper, we propose a robust, fast and cheap scheme for locating the eyes, lip-corners, and nostrils for Eye-Head Controlled Human Computer Interface on a facial image with non-constrained background. Many researchers have presented eye tracking methods. But the methods are not both robust and fast, and they also have many limitations. The method we propose uses complete graph matching from thresholded images. That is, after labeling the binarized image that is separated by a proper threshold value, the algorithm computes the similarity of between all pairs of objects. After that, the two objects that have the greatest similarity are selected as eyes. The average computing time of the image(360*240) is within 0.2(sec) and if the search window is reduced by estimation, the average computing time is within 0.1(sec). It has been tested on several sequential facial images with different illuminating conditions and varied head poses, It returned quite a satisfactory performance in both speed and accuracy. The algorithm is highly cost effective.*

Keywords: Thresholding, Labeling, Facial Feature Tracking, HCI.

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 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 and section 3 describes the proposed method of locating the facial features. Experimental results are provided in section 4. The paper concludes with section 5.

II. RELATED WORKS

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 [2][3][12] 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 a 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[4], Leung et.al.[5] and Sumi and Ohta[6] employed this approach.

The color-based detection system[7][8] 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.

The neural network approach detects faces by subsampling different regions of the image to a standard-sized subimage and then passing it through a neural network filter. Sung and Poggio,[9] and Rowley[10] reported this approach.

Motion-based approach [11] uses image subtraction to extract the moving foreground from the static background. But, this approach will not work well in the presence of a large number of moving objects in the image.

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.

III. FACIAL FEATURES LOCATING ALGORITHM

1. Locating the eyes

0-7803-5793-6 /99/$10.00 (c)1999 IEEE.

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. An example of a binarized image obtained in this manner is shown in Figure 1.

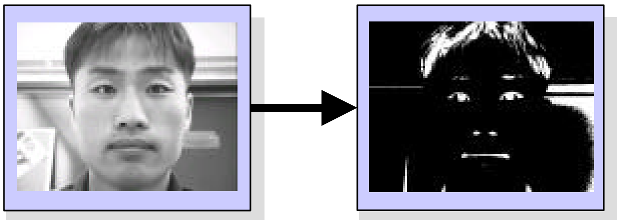

Figure 1. Binarized Image

1.2 Finding the candidates of the eyes

Edge detection is employed to find the candidates for the eyes[13]. 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 such as shown in Figure 1. 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 [13]. 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.

Figure 2 shows the result of locating the two eye blocks.

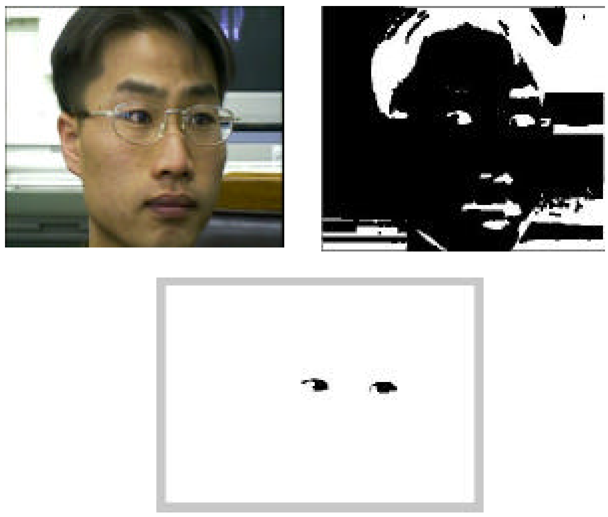

Figure 2. Locating the two eyes

2. Locating the Mouth and Lip-Corners

We can define a rough region for the mouth by using the eye information. Figure 3 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 as shown in Figure 4.

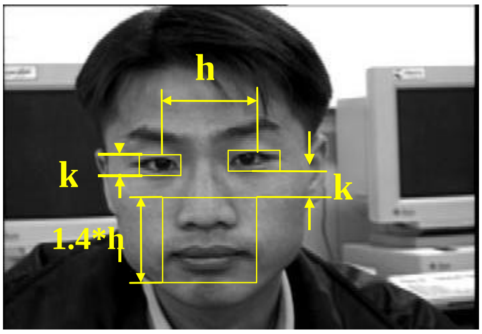

Figure 3. Defining the Mouth Region

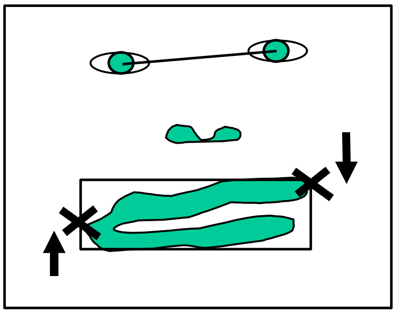

Figure 4. Locating the Lip-corners

3. Locating Nostrils

We can define the region for nostrils using the two eyes and the mouth position information as in Figure 5. But we should examine whether the nostrils in the image are appeared in one block or not.

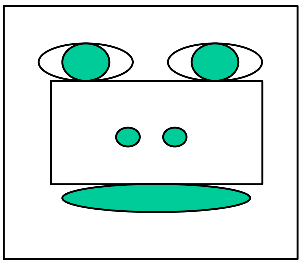

Figure 5. Define the region for nostrils

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. 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(12+ Hz) 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 6.

Figure 6. Experimental Results

V. Conclusions and Future Works

Real-time facial feature tracking for eye and head controlled human computer interface has been proposed. More intelligent gray-level threholding methods and verification techniques are desirable and head orientation must be included into the final target system. Zooming, panning, and tilting mechanisms of the camera might be useful for tracking large movement of the subject. Imagine the 'Interactive Room' in which the subject can turn on the TV, only by gazing at the TV switch. To handle this kind of large space, several cameras should track the subject cooperatively. This presents quite a difficult camera calibration problem. It is also possible to develop 3D interface on the basis of gaze difference between the two eyes. We can also expect better interface in car navigation systems which 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] Kwon-Il Kang, Jong-Hei Ra, Moon-hyun Kim, "Tracking the Eye Trajectories in dynamic images using adaptive template matching" Proceeding of the 9th Spring Conference of Korea Information Processing Society, 1988.

[3] Kyung-Nam Kim "Contributions to Vision-Based Eye-Gaze Tracking", Thesis for Master's Degree, 1999.

[4] 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.

[5] 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.

[6] Y.Sumi, Y.Ohata, "Detection of gface orientation and facial components using distributed appearance modeling", in Proc. Int. Workshop on Automatic Face and Gesture Recognition, Zurich, 1995,PP. 254-259.

[7] 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.

[8] 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.

[9] H.A.Rowley, S.Baluja, and T.Kanade. "Human face detection in visual scenes", Technical Report CMU-CS-95-158, CMU, July 1995.

[10]K.K.Sung, T.Poggio, "Example-based learning f for view-based human face detection, Technical Report A.I.Memo 1521, CBLC Paper 112, MIT, 1994.

[11] T.I.P. Trew, R.D.Gallery, D.Thanassas, E.Badique, Automatic face location to enhance videophone picture quality, in:Proc.4th Brit. Machine Vision Conf., Springer-Verlag, 1993,PP. 488-497.

[12] Sang-Ho Kim "Parallel Facial Feature Tracking", Thesis for Master's Degree, 1999.

[13] ETRI, "Electronic Helper System" the 2nd Report 1996.