

## The Estimation of Moving Velocity of Viewer by Using Two Eyes Image for 3D Display for Multiple Viewers

Se Hyuk Park, Young Bin Park, Kyung Moo Huh

School of Electronics and Electrical Engineering Dankook University

Telephone : (0417) 550-3548 / Fax : (0417) 550-3548

E-mail : huhkm@anseio.dankook.ac.kr

### Abstract

In this paper, we calculate the moving velocity of viewer by using two eye images obtained at different time through the camera. This process is necessary for future 3D display technique, in which moving viewer can see 3D image continuously. We firstly extract two eyes image and calculate the pixel coordinate of center point between two eyes. Next, we calculate the moving velocity in two dimension by comparing two center point coordinates obtained at different time.

### I. Introduction

3D display technique is a very important and fundamental technique, and has already been developed as a main research topic recently.

There are two typical types of 3D display, one is glasses 3D display method and the other is a nonglasses 3D display method. And also 3D image implementation using a holographic screen and holographic video is being developed. However, there are many problems in 3D display technique.

Among these, the autostereoscopic method has an advantage that we can enjoy 3D image without any additional device, but the method has a disadvantage of narrow viewing zone [4].

This disadvantage can be overcome with the detection of viewer's positional movement by head tracking. Head tracking is a method of measuring the movement of viewer by analyzing the image information through camera, particularly viewer's two eyes image.

In this paper, we develop a simplified algorithm to detect viewer's two eyes position to calculate the moving velocity of viewer.

### II. Calculation of moving velocity of viewer

#### II-1. Flow chart of moving velocity calculation.

Moving velocity of viewer is estimated by comparing two eyes position between the first image(at  $t=0$ ) and the second image(at  $t=1$ ) of continuous input images, and dividing the displacement by one frame time.

Fig. 1 is the flow chart of moving velocity calculation

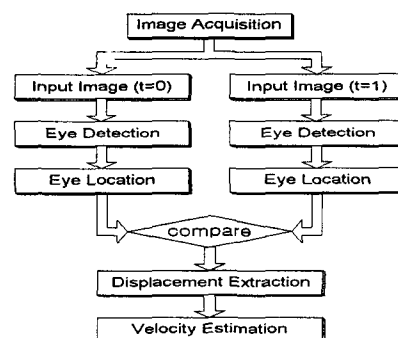


Fig. 1. Flow chart of moving velocity estimation.

## II-2. Eye detection algorithm

Eye detection algorithm is composed of preprocessing process and eye detection process.

### (1) Preprocessing

Preprocessing process is composed of threshold process and median filtering. To obtain the accurate pixel coordinate of eyes, we find the circle form of pupils in the image.

Because the gray level of pupils are nearly close to 0, the regions above threshold level are changed to white color(gray level 255), where the threshold level is decided experimentally. (The threshold level is defined for the average value of R, G, B values.) The resulting image is composed of dark part and a nearly original color part under threshold level.

Next, we decide another threshold level for each other's difference among R, G, B values and change the regions above threshold level to white color, then the regions of nearly original color are removed and only dark regions can remain in the resulting image.

After this process, we transform the image into a gray level image and execute median filtering to the resulting image.

Median Filtering is performed by sliding a window across the pixels of a picture and ranking the pixels in the window in ascending order. The median or center pixel will be placed in the output image in the location corresponding to the center of the window. Because the center value is selected, median filters have an odd number of pixels in the window. The principal function of median filtering is to force points with very distinct intensities to be more like their neighbors, thus eliminating intensity spikes [5].

After preprocessing process, hairs, eyebrows, eyes and the dark part of background only remain in the resulting image.

### (2) Eye detection

In the eye detection process, we scan the whole

image with an pupil-type mask block and find the pixel coordinates of two eyes. The mask block scans the whole image, and detects the position of each eye. If the first eye is detected, the coordinates of the eye is determined and the corresponding mask block region is changed into white color. Then, scanning around the region near the first eye, we inspect the existence of the second eye. If the second eye is detected, we finish the searching process, and if not detected, we repeat the above scanning process.

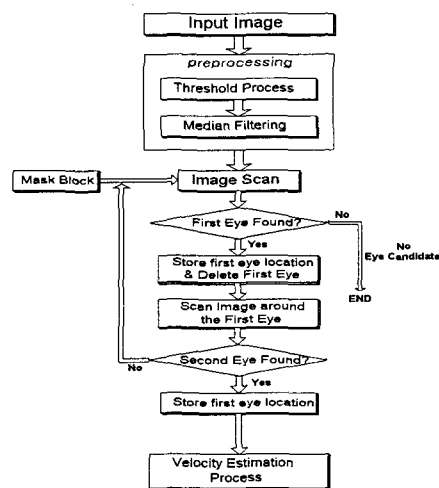


Fig. 2. Eye detection processing

## II-3. The calculation of moving velocity

In Fig. 3 is shown an analytical model for the calculation of moving velocity of viewer.

Let the coordinates of the eyes from the first input image be  $(c_1, r_1)$ ,  $(c_2, r_2)$ , and let the coordinates of the eyes from the second input image after  $\Delta t$  sec (1 frame time) be  $(c_3, r_3)$ ,  $(c_4, r_4)$ , then the coordinates of the center point between each two eyes are

$$m_1 = \left( \frac{c_2 + c_1}{2}, \frac{r_2 + r_1}{2} \right)$$

$$m_2 = \left( \frac{c_4 + c_3}{2}, \frac{r_4 + r_3}{2} \right), \quad (1)$$

and when the viewer is supposed to move in the horizontal direction, the pixel distance between two eyes becomes as (2)

$$d_p = \sqrt{|c_2 - c_1|^2 + |r_2 - r_1|^2} \quad [\text{pixel}] \quad (2)$$

If the real distance between two eyes is  $d_{eye}$ , the real distance per one pixel is calculated as (3).

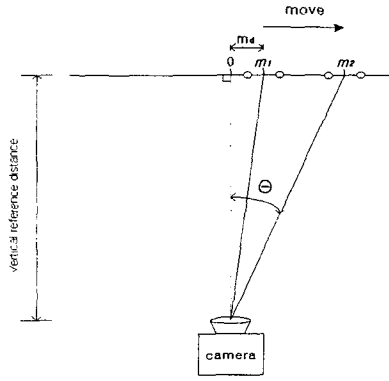


Fig. 3. Trace region

$$\begin{aligned} \text{Real distance per one pixel} &= \frac{d_{eye}}{d_p} \quad [\text{m/pixel}] \\ &= \frac{d_{eye}}{\sqrt{|c_2 - c_1|^2 + |r_2 - r_1|^2}} \quad [\text{m/pixel}] \end{aligned} \quad (3)$$

The moving velocity of viewer becomes as (4) in pixel distance,

$$v = \frac{|m_2 - m_1|}{\Delta t} \quad [\text{pixel/sec}] \quad (4)$$

and from the relation of (3), the real velocity can be calculated as follows.

$$\begin{aligned} v &= \frac{|m_2 - m_1|}{\Delta t} \frac{d_{eye}}{d_p} \\ &= \frac{|m_2 - m_1|}{\Delta t} \frac{d_{eye}}{\sqrt{(c_2 - c_1)^2 + (r_2 - r_1)^2}} \end{aligned} \quad [\text{m/sec}] \quad (5)$$

where,

$m_1, m_2$  : The coordinates of the center point

between each two eyes

$\Delta t$  : 1 frame time

$d_{eye}$  : real distance between two eyes

When viewer supposed to move in the horizontal direction as in Fig. 3., the rotational position  $\theta$  can be calculated as (6) in the following.

$$\theta = \tan^{-1} \left( \frac{|m_2 - m_1| + m_d}{\frac{d_{eye}}{\sqrt{(c_2 - c_1)^2 + (r_2 - r_1)^2}}} \right) \quad [\text{deg}] \quad (6)$$

Here,  $m_d$  is the pixel distance between  $m_1$  and the center point of the input image, and  $l$  is the vertical reference distance between the camera and viewer.

### III. Experimental result

#### III-1. Eye detection

The input image is  $320 \times 240$  true color. In the preprocessing process, we removed the high gray level region and the nearly original color region by thresholding process. And by applying  $3 \times 3$  median filtering method, we soften the distinct intensity regions through the whole image. And we detect the eye positions using mask block ( $18 \times 18$ ), which is decided experimentally.

Fig. 4 is the input image and the resulting image of preprocessing process. From this figure, we can know that the resulting image of preprocessing process includes only the hairs, eyebrows, eyes and the dark part of the background.

#### III-2. Moving velocity estimation

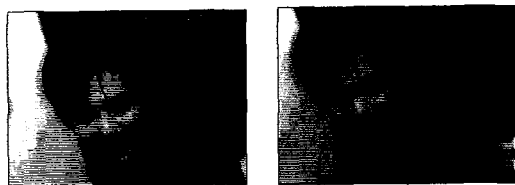
We detect the positions of two eyes and calculate the coordinates of center point between the two eyes from each image of Fig. 5. And then, we calculate the displacement and the moving velocity as described in the previous section.



(a) Input image (b) Preprocessing image  
Fig. 4. Input image and Preprocessing image

This experiment was executed under the following assumption.

- a) Real distance between two eyes( $d_{eye}$ ) is 0.065 m.
- b) Vertical reference distance between the camera and viewer( $l$ ) is 0.85 m.
- c) 1 frame time( $\Delta t$ ) is 0.2 sec.



(a) Input image (t=0) (b) Input image (t=1)  
Fig.5. First input image and Second input image

The experimental result of applying the proposed algorithm under the above assumption is shown in Fig. 6, from which we can know that the moving velocity of viewer and the rotational position(angle) of the viewer are 0.22 m/s and 3.08 degree.

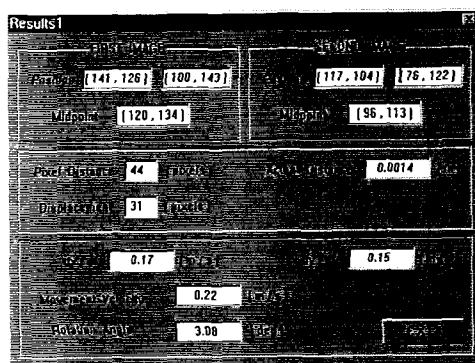


Fig. 6. The experimental result

#### IV. Conclusion

By the proposed eye detection algorithm we could find the accurate position of each eye near the vertical reference distance between the camera and viewer. However, in the case of very short distance or very long distance between the camera and viewer, this algorithm has much difficulty in finding the position of eyes.

#### Reference

- [1] D: Murray and A. Basu, "Motion tracking with an active camera", IEEE Trans. PAMI, Vol.16, No.5, pp.449-459, May 1994
- [2] R. C. Nelson, "Qualitative detection of motion by a moving observer", Proc. IEEE Conf. CVPR, Maui HI, pp.173-178, June 1991
- [3] R. S. Stephens, "Real-time 3-D object tracking", Image Vision Computing, Vol.8, No.1, pp.91-96, 1990
- [4] Tomohiko Hattori, "Streographic Image Taking and Output Devices Compatible with Conventional Infrastructure," Proc of International Workshop on 3D Imaging Media Technology, pp 143-150, march 1996.
- [5] Kenneth R. Castleman, "Digital Image Processing", Prentice Hall, 1996
- [6] M. J. Swain and M. A. Stricker, "Promising directions in active vision", Int'l J. computer Vision, 11:2, pp.109-126, 1993
- [7] C. Fermuller and Y. Aloimonos, "The role of fixations in visual motion analysis", Int'l J. Computer Vision, 11:2, pp.165-186, 1993
- [8] Q. Tian and M. N. Huhns, "Algorithms for sub pixel registration", CVGIP, Vol.35, pp.220-233, 1986
- [9] I. K. Sethi and R. Jain, "Finding Trajectories of feature points in a monocular image sequence" IEEE Trans. PAMI, Vol.9, No.1, pp.56-73, 1987
- [10] C. L. Morefield, "Application of 0-1 integer programming to multitarget tracking problems", IEEE Trans. Auto. Contr., Vol.22, No.6, June 1977
- [11] S. Yang, K. Baek, and Y. Soh, "Tracking of moving vehicle for automatic vehicle identification(AVI) system", Proc. Intelligent Transport Systems World Congress, Oct. 1996