Hand Gesture Recognition by Stereo Camera Using the Thinning Method

Xianghua Li, Jun-ho An, Jin-hong Min and Kwang-Seok Hong
School of Information and Communication Engineering
Sungkyunkwan University
Suwon, South Korea
gidghk88@skku.edu, amadasv@skku.edu, kaysi@skku.edu, kshong@skku.ac.kr

***Abstract*—In this paper, we propose a real-time hand gesture recognition system using a thinning method utilizing a stereo camera. We implement a depth map in the hand detection portion that uses a sum of absolute differences method based on the acquired right-left image to detect the foreground object. We use a convex hull to detect the region of interests (ROI) and calculate the depth of object in ROI to obtain hand images that are more accurate. Then, we remove the background image in ROI and get the foreground image as a hand image. Finally, we use a blob labeling method to obtain the clean hand image, without the noise caused by computing distance from stereo matching. The hand gesture recognition system uses the Zhang and Suen thinning algorithm to obtain the feature point, angle and distance. It recognizes five kinds of hand gestures. The proposed method achieves an average recognition rate of 82.93%.**

Keywords-component; hand gesture; thinning; depth map

I. INTRODUCTION

In recent years, vision-based hand gesture recognition is a challenging problem in the field of computer vision and pattern analysis. Hand gesture recognition can be used in many applications. The objective of research in hand gesture recognition is to apply to sign language recognition system, control of household electronic appliances, play games and human-computer interaction, etc. Stereo cameras are rarely used in the field of gesture recognition following the development of 3D applications developed to interact appropriately with these 3D environments. For instance, Sung-il Kang et al. [1] proposed a real-time method for a hand detection and gesture classification system by stereo camera using both the depth and color property able to reconstruct the gesture trajectory. Doe-Hyung LEE et al. [2] proposed a real-time hand gesture recognition system based on difference image entropy using a stereo camera and implemented a Chinese chess game interface.

We propose a hand gesture recognition system using a stereo camera. In this paper, we use a sum of absolute differences (SAD) method to implement a depth map after measuring the similarity between two cameras [3]. The result of the depth map in previous research based on the SAD method has noises caused by computing distance noises. However, we obtain a more accurate hand images with less noise in the proposed system. The hand gesture recognition system uses Zhang and Suen's thinning algorithm [4] to obtain feature points and use these feature points to recognize hand gestures.

The remainder of this paper is organized as follows. In section 2, we describe the limitations of existing research. The proposed hand detection method is presented in section 3. Section 4 presents the proposed hand gesture recognition algorithm. Experimental results and conclusions are presented in section 5 and 6, respectively.

II. RELATED WORK

A. Hand Detection

Figure 1 shows the hand detection using stereo camera. This system first detects the person using depth value. After detecting a person, it refines the depth map using both depth and color property, shown in figure 1(a). Finally, it detects the hand region and reconstructs the gesture trajectory [1].

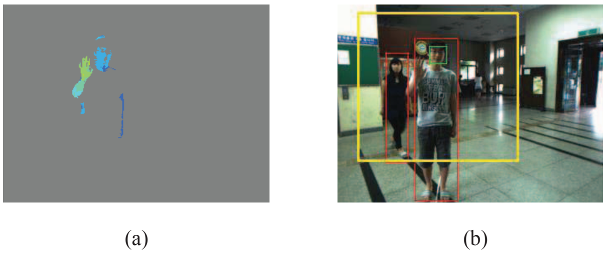

(a) (b)

Figure 1. (a) Depth map with the skin color (b) Detected hands (orange) and their trajectories (green).

B. Hand Recognition

This system implemented hand gesture recognition system by stereo camera. The recognition system used the difference image entropy of the input image and the average image. Figure 2 shows the four kinds of the average images [5].

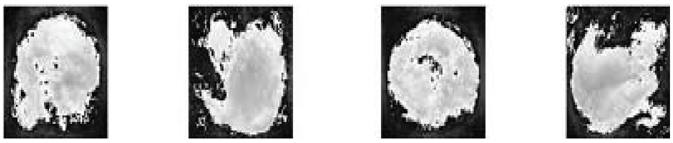


Figure 2. Left hand (vertical grab, vertical splay, horizontal grab and horizontal splay).

978-1-61284-774-0/11/$26.00 ©2011 IEEE

However, these systems implement a depth map using only the SAD method. However, errors occur in computing the distance from stereo matching using SAD method. In this paper, we use the SAD method to obtain the depth map to achieve more accurate hand detection image. We also consider noise reduction.

III. HAND DETECTION

The proposed method involves three main stages in using SAD for stereo matching images, calculating the distance between the stereo camera and hand to detect the hand image. Then, it detects the ROI and uses blob labeling to reduce noise.

A. SAD and Depth Map

In this paper, we use the SAD method to calculate the matching region in many computational stereo vision images in real-time. Our system makes the depth map using the left and right image displacement difference, in the SAD matching region, except for the left and right non-matching region. Equation (1) calculates that the smallest displacement difference; it is the matched displacement difference [3].

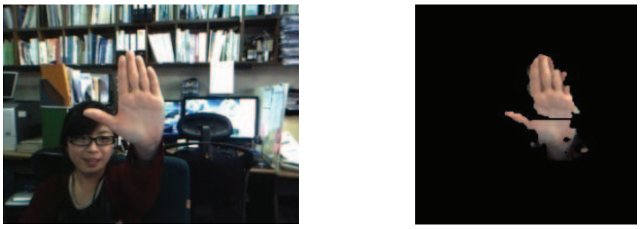
$$C(x,y,\delta)=\sum_{y=0}^{wh-1}\sum_{x=0}^{ww-1}\left|I_R(x,y)-I_L(x+\delta,y)\right| \quad (1)$$

Where *wh* and *ww* denote the window height and width in the image and δ are the differential values. $I_R(x,y)$ and $I_L(x+\delta,y)$ represent the intensity corresponding to coordinate of the right and left images.

The SAD method is selected due to its high-precision and its advantage of a real-time guarantee. We find the 3-dimensional coordinates in each frame using the displacement difference of the left and right images and the focal distance.1

$$d=\frac{b\times f}{d_l-d_r} \quad (2)$$

In our system *b* (base-line) =12 cm is used with the *f* (focal-length) =0.38 cm. The depth of the pixels can be calculated from Equation (2).However, we cannot calculate all pixel depth values. We can determine the approximate distance information using pixel interpolation due to the occlusion by the foreground object. We recognize the foreground object as a hand that can segmented from the background using depth values. Fig. 3 shows the rectification image and hand region image detects by depth map.

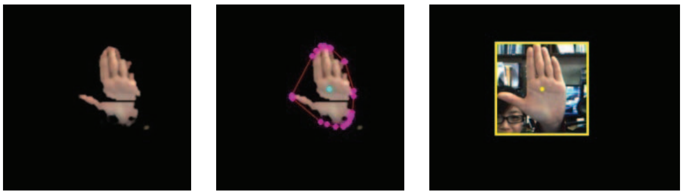


(a) (b)

Figure 3. (a) Rectification image using right and left images (b) Hand region detection using depth map.

B. Region of Interests(ROI)

We use the YCbCr space, instead of RGB space, to obtain the skin color region that can reduce the unnecessary hand image based on the acquired hand detection image. The next step uses the convex hull method to detect the boundary point and obtain the center coordinate of the hand. Finally, we calculate the largest distance between the center point and the boundary point as the radius. This radius is used to make a rectangle, as in ROI. Then, we use the distance value again to remove the background image in the ROI and detect the foreground image as the hand image. Figure 4 shows the resulting images.

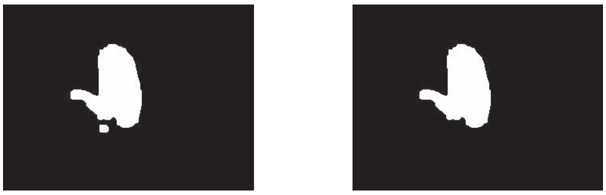


(a) (b) (c)

Figure 4. (a) Skin color detection (b) Convex hull (c) Detect region of interest.

C. Blob Labeling

The hand image in the ROI also has noise caused by the distance calculation error. Thus, we use the blob labeling method to resolve the background noise. The blob labeling method proposed by Rosenfeld and Pfaltz [6] performs two passes over a binary image. Each point is encountered once in the first pass. At each black pixel P, a further examination of its four neighboring points (left, upper left, top, and upper right) is conducted. If none of these neighbors carries a label, P is assigned a new label. Otherwise, those labels carried by neighbors of P are said to be equivalent. In this case, the label of P is replaced by the minimal equivalent label. A pair of arrays is generated for this purpose, one containing all current labels and the other the minimal equivalent labels of those current labels. Label replacements are made in the second pass. Then, we can obtain the largest blob as the hand image and remove the other small blobs. Finally, the clean hand image is detected, as shown in Fig. 5.



(a) (b)

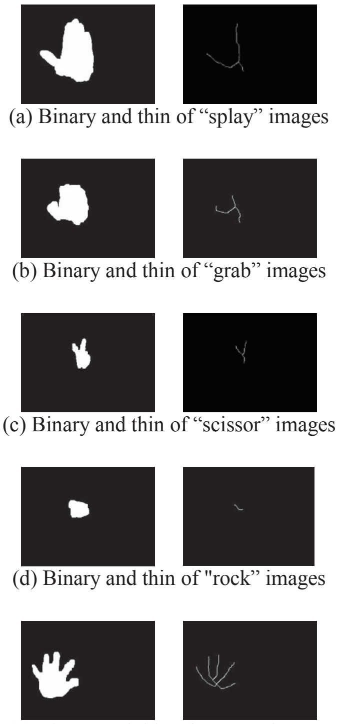
Figure 5. (a) Binary image (b) After blob labeling to remove small blobs.

IV. HAND GESTURE RECOGNITION

Once we have detected the hand using the stereo camera, we need to recognize the hand gesture images. The recognition candidate images are divided into five kinds of images, “grab”, “splay”, “scissor”, “rock” and “paper”. We use the thinning algorithm and detect the feature point to recognize these kinds of images. Finally, we calculate the angle distance between end-points and the diverging point.

A. *Thinning Results and Feature Point Detection*

We apply the Zhang and Suen (ZS) thinning algorithm[4] on the binary image. Fig. 6 shows the binary images and thin of binary images of the five kinds of hand gestures.



(a) Binary and thin of "splay" images

(b) Binary and thin of "grab" images

(c) Binary and thin of "scissor" images

(d) Binary and thin of "rock" images

(e) Binary and thin of "paper" images

Figure 6. The binary images of five kinds of hand gestures and the thin of the binary images.

We check the 8-connectivity neighbors to determine the end-points and joint points in the thin of the binary image to obtain the feature points. The end-point is one with only one 8-connectivity neighbors and represents the terminal pixel of a thin segment. The joint-point is a point on the thin segment with more than two 8-connectivity neighbors. We also obtain the number of end-points and joint-points. Figure 6 shows the "rock" image that has only two end-points and the "paper" image that has five end- points.

B. *Angle and Vertical Distance*

Three features are used to recognize gestures in this proposed algorithm. One of the features is the number of end-points. The second feature used in this paper is the angle. The third feature is the vertical distance from the end-point to the joint-point. We calculate the angle between the lines that join the joint-point to each end-point. In our system, we select the joint-point located at the bottom of the thin image, if there is more than one joint-point. The end-points that are located below the joint-point are not considered as end-points, as shown in Figure 7.

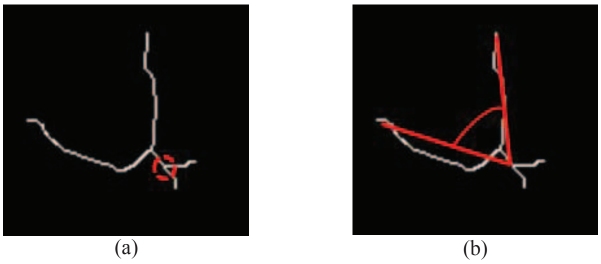


Figure 7. (a) Select the joint-point located a the bottom (b) Angle value(do not consider the end-points below the joint-point).

The vertical distance is the distance from the top of the end-points to the bottom joint-point is shown in Figure 8 presented by red lines. Figure 8 shows the thin images in the same depth value. This means the hand image is located at the same distance from the stereo camera. We find that the thin of "splay" image has a bigger distance value than the thin of "grab" image in the resulting images. Therefore, we can classify the two hand gestures using the depth value and distance.

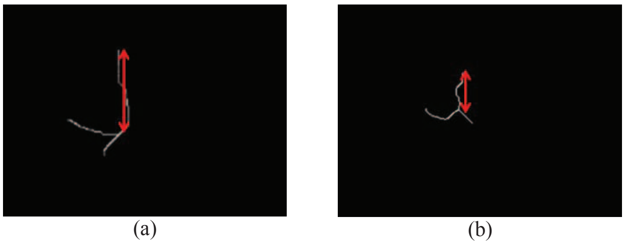


Figure 8. (a) Thin of "splay" image (b) Thin of "grab" image.

C. *Gesture Recognition using Feature Points*

In this part, we use the above three kinds of features to recognize hand gestures. Table 1 shows each of the hand gesture using each feature.

TABLE I. FEATURES USED IN EACH GESTURES

Gestures	Features used
paper	Number of end-points
rock	Number of end-points
scissor	Number of end-points, Angle
splay	Number of end-points, Angel, Vertical Distance
grab	Number of end-points, Angel, Vertical Distance

We use the number of end-points to classify the gestures "paper" and "rock". In this experiment, the result of the thin image mostly shows that "paper" has five to six, "rock" has two and "scissor", "splay" and "grab" have three or four end-points. Thus, we recognize the "paper" and "rock" hand gestures easily using only the number of end-points.

The "scissor" gesture is recognized using the number of end-points and angle value. The angle value is smaller than the "splay" and "grab", so we define the threshold value to distinguish it from the other two gestures. Figure 9 shows the thin images for these gestures.

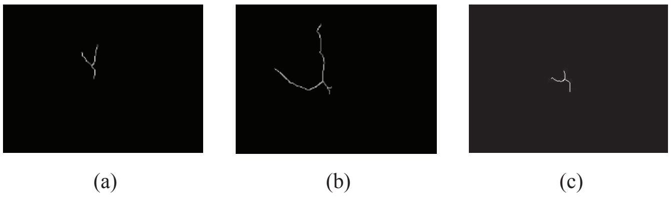


Figure 9. (a) Thin of “scissor” image (c) Thin of “splay” image (c) Thin of “grab” image.

Figure 9 shows these three thin images have different angles. The “scissor” gesture is mostly at an angle of 30 degrees to 50 degrees and the other two gestures mostly have angles of 60 degrees to 90 degrees.

We use the depth value, number of the end-points, angle and distance value to recognize the last two gestures, “splay” and “grab”. In this proposed method, the thin images show that “splay” and “grab” images are have a similar number of end-points and angle. Thus, we use the distance value to classify these two kinds of gestures, as shown in figure 8. In this experiment, we set the depth areas 20~30, 30~40, 40~50 cm from the stereo camera. Table 2 shows that the different depth values result in different vertical distance value using these features to recognize “splay” and “grab” gestures. In our system wh (window height) =240 pixel is used.

TABLE II. VERTICAL DISTANCE VALUE

Depth Area **(cm)**	**Splay** **Vertical Distance** **(pixel)**	**Grab** **Vertical Distance** **(pixel)**
20 ~ 30	1/4wh ~ 1/2wh	1/12wh ~ 1/5wh
30 ~ 40	1/6wh ~ 1/4wh	1/48wh ~ 1/10wh
40 ~ 50	1/8wh ~ 1/4wh	1/48wh ~ 1/10wh

V. EXPERIMENTAL RESULT

We used the Bumblebee stereo camera and a resolution of 320×240 pixels in the system developed. For algorithm implementation, we used Visual Studio 2008, OpenCV 2.0. The target for detection and recognition is divided into five kinds of image, "splay", "grab", "scissor", "paper" and “rock”. To evaluate the performance of the proposed hand gesture recognition system was tested using 500 test images, using 100 test images of each hand gestures. Table 3 shows the result of correct detection samples and the error samples; the hand detection rate is 93%. Table 4 shows the hand gesture recognition rate. The average recognition rate is 82.93%.

TABLE III. HAND DETECTION RESULT

Gestures	**paper**	**rock**	**scissor**	**splay**	**grab**
Correct sample					
Error sample					

TABLE IV. GESTURE RECOGNITION RATE

Gestures	**Recognition rate**
paper	92.00%
rock	81.33%
scissor	77.33%
splay	81.33%
grab	82.66%
Average Recognition Rate	82.93%

VI. CONCLUSION

In this research, we implemented a hand recognition system using a stereo camera in real-time. First, we performed hand detection using a depth map after matching the images obtained from a stereo camera in the left and right position. We use a convex hull to detect the region of interests (ROI) and calculate the depth of the object in ROI to obtain hand images that are more accurate. We detected the hand region in a complex environment. Hand detection achieved high performance. We use Zhang and Suen’s thinning algorithm to obtain the feature points to recognize the five kinds of gestures. The recognition achieved an average recognition rate of 83%. This system can be applied to games or any other control system.

Further work on the current project will implement game applications, such as Jang-Gi or chess.

ACKNOWLEDGMENT

This research was supported by the MKE(The Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the NIPA(National IT Industry Promotion Agency)" (NIPA-2011-(C1090-1121-0008)) and the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(2010-0021411).

REFERENCES

[1] Sung-il Kang, Annah Roh, Hyunki Hong, “using depth and skin color for hand gesture classification”, Consumer Electronics (ICCE), pp. 155-156, 2011.

[2] Doe-Hyung Lee, Kwang-Seok Hong, “Game interface using hand gesture recognition”, Computer Sciences and Convergence Information Technology (ICCIT), pp. 1092-1097, 2010.

[3] S Wong, S. Vassiliadis, S. Cotofana, “A Sum of absolute differences implementation in FPGA Hardware,” Euromicro Conference, 2002. Proceedings. 28th, pp. 183-188, September 2002.

[4] T.Y. Zhang and C.Y. Suen, “A Fast Parallel Algorithms for Thinning Digital Patterns. Research Contributions”, Communications of the ACM.27(3): pp. 236-239, 1984.

[5] Doe-Hyung Lee, Kwang-Seok Hong, “A Hand gesture recognition system based on difference image entropy”, Advanced Information Management and Service (IMS), pp.410, December 2010.

[6] A. Rosenfeld, and P. Pfaltz, Sequential Operations in Digital Picture Processing, Journal of the Association for Computing Machinery, pp. 471-494, December 1966.