International Journal of Multimedia and Ubiquitous Engineering
Vol. 10, No. 2 (2015), pp. 75-86
http://dx.doi.org/10.14257/ijmue.2015.10.2.07

Depth based Sign Language Recognition System Using SVM

Kisang Kim, Su-Kyung Kim, Hyung-Il Choi

School of Media, Soongsil University, Seoul, Korea
Illusion1004@gmail.com, ksg1603@ssu.ac.kr, hic@ssu.ac.kr

Abstract

In this paper, we propose a sign language recognition system using an SVM (Support Vector Machine) and a depth camera. In particular, we focus on the Korean sign language. For the sign language system, we suggest two methods, one for the hand feature extraction stage and the other for the recognition stage. Hand features consist of the number of fingers, finger length, palm radius, and hand direction. To extract hand features, we use Distance Transform and a hand skeleton. This method is more accurate and faster than a traditional method that uses contours. To recognize hand posture, we develop a decision tree with hand features. For more accuracy, we use SVM to determine the threshold value in the decision tree. In the experiment results, we show that the suggested method is more accurate and faster when extracting hand features and recognizing hand postures.

Keywords: *Sign Language, Hand Posture Recognition, SVM*

1. Introduction

Because of the recent growth of the mobile and smart TV markets and the convergence of smart devices and various other devices, smart devices are widely used in various places. Of the many smart device developments on character input methods, the sign language recognition system that uses the camera on a smart TV, in particular, is under active development [1-5]. Existing sign language recognition systems require users to equip special input devices such as color and data gloves. However, recent studies have focused on the use of sign language recognition systems without special devices such as color gloves. When a user utilizes special gloves, sign language recognition systems can easily obtain information on the user's hands and motions, but carrying such gloves constantly is an inconvenience to users. Moreover, the release of cameras installed with infrared ray sensor has made hand recognition devices such as color gloves meaningless.

The focus of recent studies is to recognize hand shapes and input them on a TV screen. Various methods have been suggested: the use of a neural network that learns and recognizes hand shapes [6]; hand recognition by removing the palm area and then extracting hand candidates [7]; the extraction and learning of hand features using SVM (Support Vector Machine) [8]; and the recognition of fingers by unfolding the contour of a hand [9]. The neural network method is unsuitable for recognizing various hand motions, such as rotation, and the hand needs to be fixed while rotating; therefore, this method is substantially inconvenient for the user. In addition, this method is limited in accommodating the count of recognizable hand shapes; consequently, it is inappropriate for applying to the actual character input system. An alternative for resolving this problem is a suggested method that recognizes hand motions by removing the palm of a hand. It is impossible to distinguish fingers when they are attached together, and this can cause errors. The method that recognizes fingers by unfolding the contour of a hand is a good algorithm for hand recognition, but as

ISSN: 1975-0080 IJMUE

with the neural network-based method, it is almost unable to recognize a revolving hand; consequently, this method is also limited in counting recognizable hand shapes. Lastly, the method that extracts hand features and has the sign language recognition system learn such features using SVM is effective, but this method is dependent upon a hand feature extraction method, and the count of its recognizable hand shapes is limited.

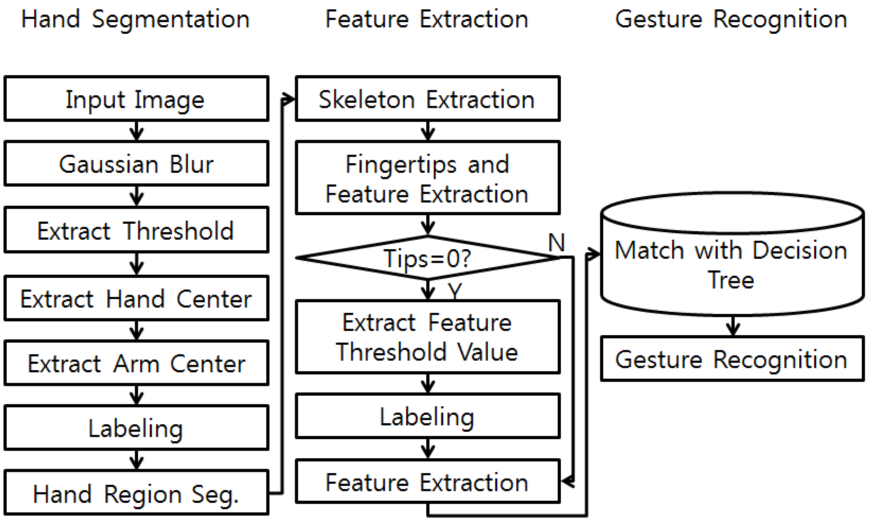


Figure 1. Proposed Method Structure

Figure 1 shows a process diagram for the method proposed in this study. This study proposes a hand shape recognition system that recognizes 28 templates: 14 consonants and 14 vowels. The hand region is extracted using the depth values of an image, provided that the hand region is always close to the camera for hand extraction. Then, the hand's features are extracted in order to recognize hand shapes. The hand feature extraction uses two methods: one is based on the hand skeleton and the other is based on a depth image. In Figure 1, the process for counting fingertips shows the recognition of hand features. The features to be extracted are finger count, finger length, angle, palm size, etc. The proposed method uses these features for recognition and proposes how to recognize hand shapes as close to the sign language as possible. A decision tree is used to recognize hand shapes. To make this decision tree, an SVM is used to recognize the branch point at each branch, and the decision tree generates more accurate branch points than those designated by a user. Using this decision tree, hand features are compared to the data added to the database. Section 2 and 3 explain the method for extracting hand features. Section 4 explains gesture recognition. The experiment result section presents the improvements introduced by our proposed system compared to existing systems, and shows the example test. The conclusion describes a future improvement.

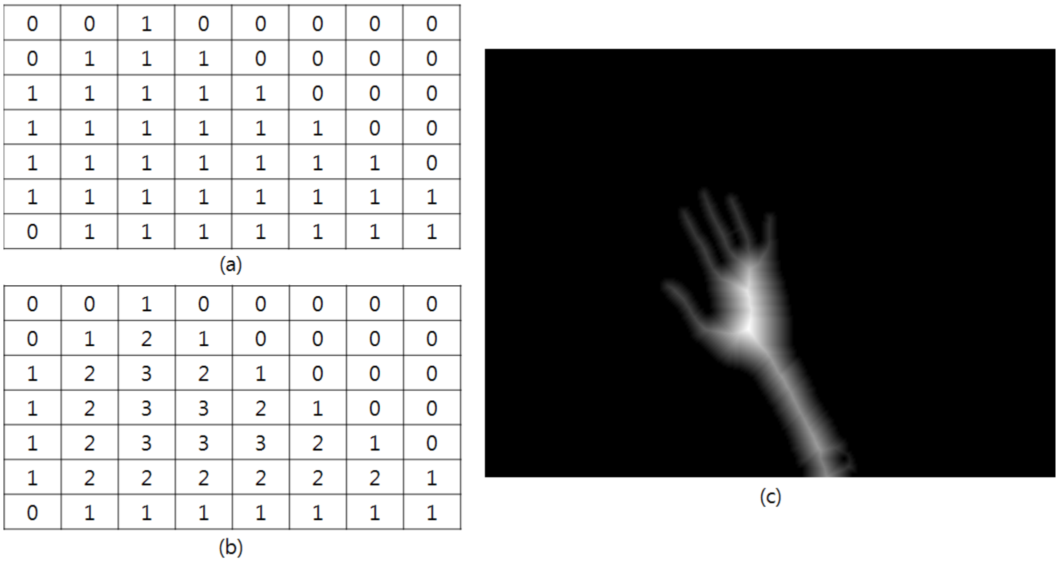
2. Hand Region Detection using Depth Image

In general, depth images contain much noise, and thus using such images without correction produces unwanted results. To prevent such problems, a smoothing calculation is performed. The Gaussian smoothing technique that uses the Gaussian kernel is widely used and it is also effective for removing noise. An image applied with the Gaussian smoothing is used to extract hand candidate regions through a binary-coded technique. This method is always based on the assumption that the hand is placed at the shortest distance from the camera. However, the threshold value (T) is extracted using the depth values and size of pertinent regions. Calculating threshold values generates an integral image (G). With the integral image, the minimum average depth value is extracted in the size of a certain region (ω) because, for pixel unit verification, the hand region is detected only partially when it is inclined. Equation (1) calculates the average depth value (S), which is the window of the size ω. The ω value varies by depth, but its maximum size is set to 30. If the value is greater than 30, it would include unnecessary regions, thus failing to extract the hand region correctly.

$$\mathrm{S}(\mathrm{x},\mathrm{y}) = \frac{G(x+\omega,y+\omega)-G(x+\omega,y-\omega)-G(x-\omega,y+w)+G(x-\omega,y-\omega)}{(2\omega+1)^2} \qquad (1)$$

The minimum of the S values obtained through Equation (1) is determined to be the threshold value. Because using only the threshold value extracts the hand region by a narrow margin, the sum of the threshold value and α is set as the final threshold value (T), where α equals to 50 in this study.

Determining the accurate hand region once a hand candidate region is extracted using the final threshold value entails removal of the unnecessary part—the arm region. The center of the arm region is detected in order to remove the arm region, which is the center point of the regions that carry the values between T-30 and T. The reason this removal technique works is that the arm always lies behind the hand from the point of the camera. When the arm region is detected, the center and the radius of the hand are extracted using distance transform. Distance transform is applied because the center of a palm is located at the deepest point in the hand region because of the fingers. Figure 2 shows an example of distance transform.



0	0	1	0	0	0	0	0
0	1	1	1	0	0	0	0
1	1	1	1	1	0	0	0
1	1	1	1	1	1	0	0
1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1
0	1	1	1	1	1	1	1

(a)

0	0	1	0	0	0	0	0
0	1	2	1	0	0	0	0
1	2	3	2	1	0	0	0
1	2	3	3	2	1	0	0
1	2	3	3	3	2	1	0
1	2	2	2	2	2	2	1
0	1	1	1	1	1	1	1

(b)

(c)

Figure 2. Example of Distance Transform. (a) Binary Image, (b) Distance Transformed Image, (c) Distance Transformed Hand Image

Finally, the hand region is extracted when the center and the radius of the palm and the center of the arm are calculated. The extraction method only leaves the palm region by removing the unnecessary arm region because the arm region is placed below the palm and the palm region is already known. The radius is the distance between the center of the arm and the center of the palm; the circular region centered on the arm is removed, except for the region within the radius × 1.2 from the center of the palm. In Figure 3, (a) and (b) are the ranges of the pertinent regions, and (c) and (d) the extracted hand regions. In addition, (a) and (c) are the results when the arm region is minimal because the hand is close to the camera, and (b) and (d) are the results when the arm region is large because the distance from the hand to the camera is far.

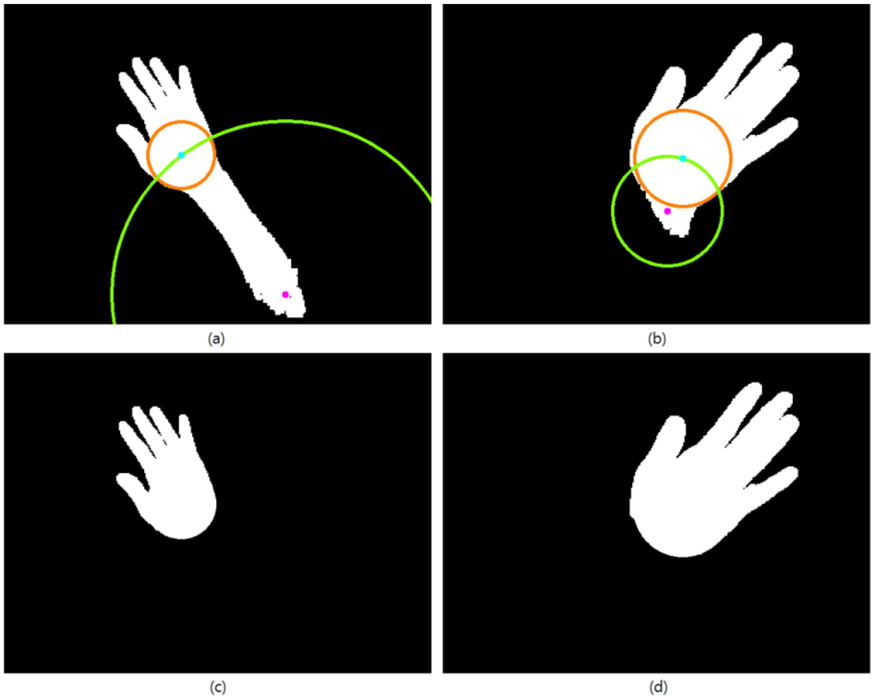


Figure 3. Example of Hand Region Extraction (a) Hand Region Features when Hand is Near to Camera, (b) Hand Region Features when Hand is Far from Camera, (c) Extracted Hand Region with (a), (d) Extracted Hand Region with (b)

3. Hand Feature Detection for Recognition

Extracting hand features in this proposed system largely involves two methods: one extracts the hand's skeleton and the other uses a threshold value different from the one used to extract the hand region. The reason for these two methods is the problem shown in Figure 4. That is, for Figure 4 (a), the palm and the camera are in the horizontal direction, and thus the former method is efficient for extracting hand features. However, for Figure 4 (b), the palm and the camera are in the vertical direction, making it difficult to extract the skeleton. Fortunately, there are only four such sign language templates, and they are not difficult to distinguish. Hence, their features are extracted using the threshold value.

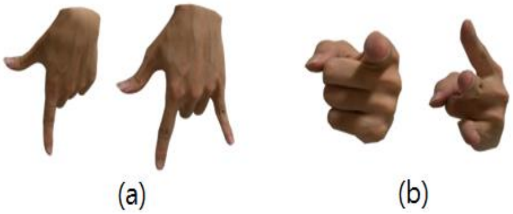


Figure 4. Sign Language Example

3.1. Hand Feature Detection using Hand Skeleton Detection

In this study, the hand skeleton is extracted prior to obtaining the hand features, especially the fingers. The reason for extracting the hand skeleton first is that extracting the contour first gives dull fingertips, thus entailing an integration process to sharpen the fingertips. Moreover, if the fingers are attached, an inaccurate design causes difficulties in distinguishing the fingers in the integration process. Figure 5 shows the problem of an incorrect fingertip extraction caused by several points detected for the thumb and the middle finger.

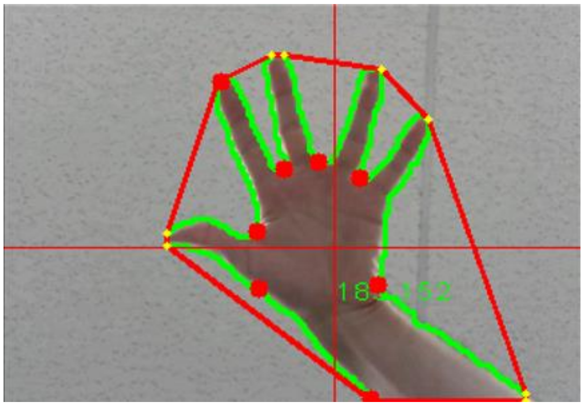


Figure 5. Problem of Contour-using Method

Extracting the hand skeleton uses the characteristic of distance transform. That is, an image applied with distance transform has a higher distance value that is close to the center and a lower value that is close to the background. Based on this characteristic, Equations (2) - (4) are used to extract the hand skeleton.

$$Q_1(x,y) = \begin{cases} 1 & if, D(x,y) \geq L/10 \\ 0 & otherwise, \end{cases} \tag{2}$$

$$Q_2(x,y) = \begin{cases} 1 & if, c \leq 2 \\ 0 & otherwise, \end{cases} \tag{3}$$

$$\mathrm{Q(x,y)} = \begin{cases} 1 & if, Q_1(x,y) = 1 \cap Q_2(x,y) = 1 \\ 0 & otherwise, \end{cases} \tag{4}$$

Equation (2) removes the parts unrelated to the hand skeleton. If extraction involves too much detail, the hand skeleton is inaccurately extracted and noise occurs. L is the palm radius obtained through distance transform in the previous process, and D(x,y) is an image generated through distance transform. Based on the current coordinates (x,y), Equation (3) compares the current D value to the neighboring D values in eight directions. For example, when the value of D(x,y) - D(x-1,y) is a positive number, the c value is increased by one. On the other hand, when the value is a negative number, the c value remains unchanged. The x and y-coordinates determine whether the point is part of the skeleton by comparing how big the value is compared to neighboring distance values. Equation (4) confirms that both Equations (2) and (3) are correct, thus recognizing that the coordinates belong to the skeleton. Figure 6 shows an image of a hand skeleton extracted through these equations.

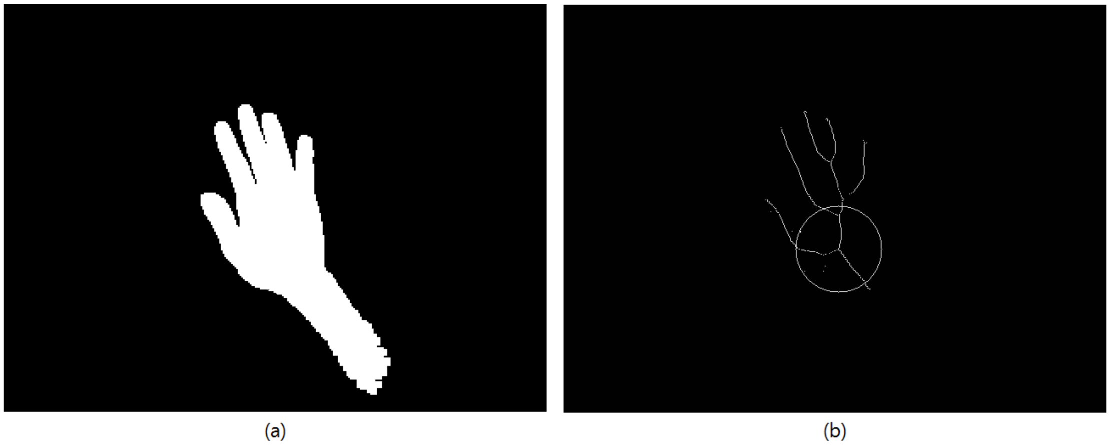


Figure 6. Hand Skeleton Extraction Example. (a) Hand Image, (b) Extracted Hand Skeleton

When the hand skeleton is extracted, it is used to extract fingers. Prior to extracting fingers, the Convex Hull algorithm is used to extract fingertips, which are the starting point of finger extraction. Equation (5) is the Convex Hull algorithm.

$$\mathrm{C} \equiv \left\{ \sum_{i=1}^{N} \lambda_i p_i : \lambda_i \geq for\ all\ i\ and\ \sum_{j=1}^{N} \lambda_j = 1 \right\} \quad (5)$$

In Equation (5), $p_1, \ldots, p_N$ are the locations of Q (x,y) that are the results of Equation (4), and N is the pixel count of Q(x,y). If the image in Figure 6 (b) is run through the Convex Hull algorithm to extract the fingertips, the palm region is extracted inaccurately, hindering an accurate fingertip extraction. To resolve this problem, as shown in Figure 7 (a), the palm region is expressed and inputted as a circle. When this image is run through the Convex Hull algorithm, as shown in Figure 7 (b), the palm region is incorrectly recognized as a fingertip. The fingertip candidates contained in the palm region are considered errors and thus removed, producing a fingertip extraction result as shown in Figure 7(c).

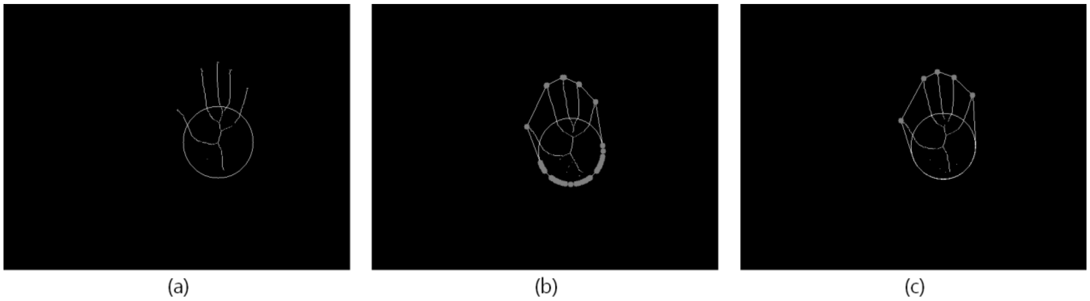


Figure 7. Fingertips Extraction Example (a) Input Image, (b) Convex Hull, (c) Extracted Fingertips

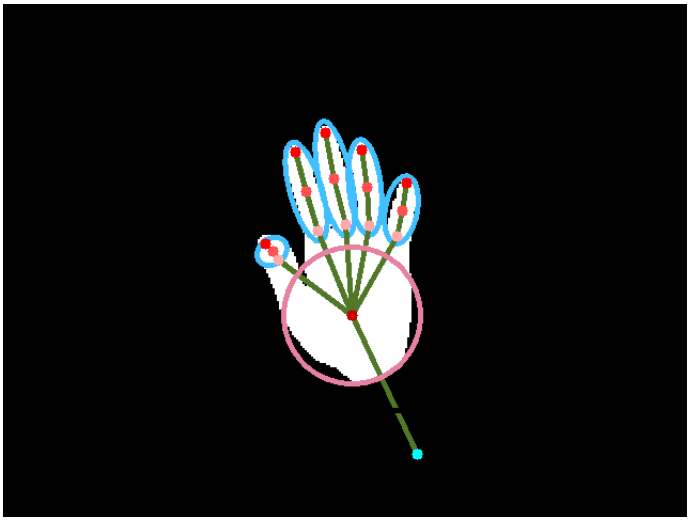


Figure 8. Result of Hand Feature Extraction using Skeleton

Once the fingertips are extracted, a search from the fingertips to the palm region is performed to finally extract the fingers. A recursive function is execute add and continued until the search goes into the palm region from the extracted fingertips, or until the skeleton ends. Using the search information, the length of each finger and its direction are determined. In addition, the direction of the longest finger is determined to be the representative direction, and all the information obtained is inputted to the hand shape recognizer. Figure 8 is the result of extracting hand features.

3.2. Hand Feature Detection using Depth Values

The hand in Figure 9 is located at a right angle to the camera, and thus its skeleton image cannot be extracted; for this case, depth values are used to detect hand features. The extraction of hand features using depth values entails less sign language templates than skeleton-based extraction; consequently, this method requires information such as finger count and distances between fingers.

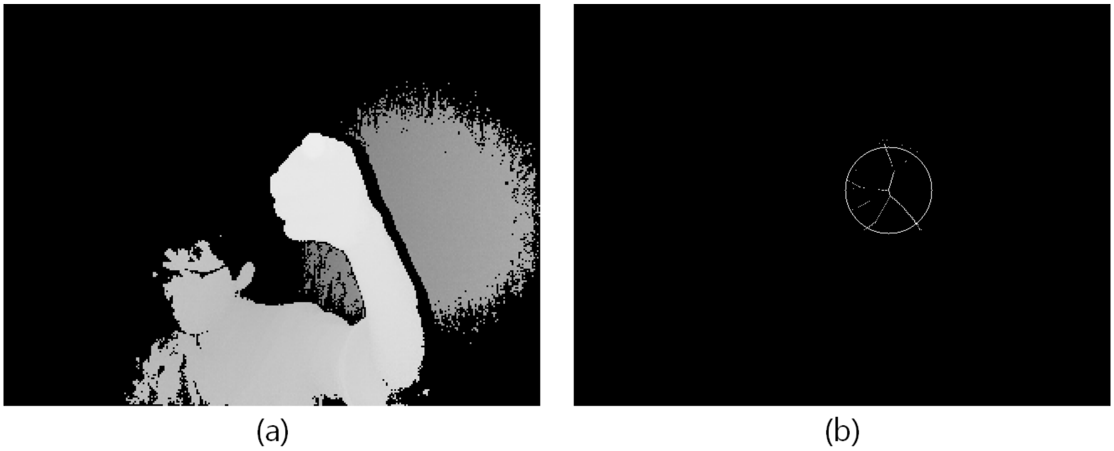


Figure 9. Failure Example of Extracting Hand Feature (a) Input Image, (b) Extracted Hand Skeleton

First, extracting fingertips using depth values detects the minimum depth value from the depth image, where the minimum depth value is limited to the values within the hand region. The minimum depth value is added to 55, making it the threshold value and creating a binary-coded image. Test results showed that 55 are suitable for dividing the amount of finger protrusion from the palm. From the binary-coded image, the finger count is obtained based on region shape using labeling. When a certain region has a ratio of 2:1 or greater for the major axis to the minor axis, the pertinent finger is considered attached, and thus the finger count is two. For three fingers, the Euclidean distance between each finger is measured and used as features. Figure 10 is the result of extracting hand information using depth values.

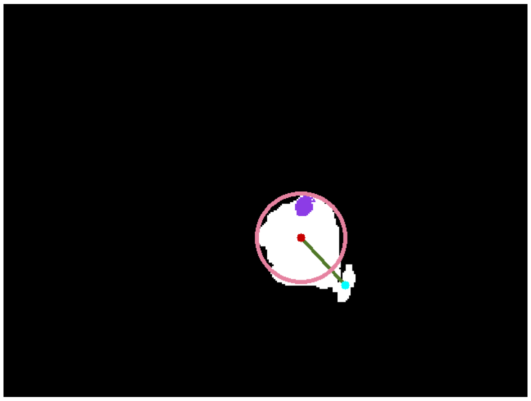


Figure 10. Extracted Hand Feature using Depth Threshold Value

4. Hand Shape Recognition

A decision tree is an easy and fast method for obtaining hand shapes. The decision tree used in this study, as shown in Figure 11, consists of four layers for skeleton-based features, and of two layers for depth value-based features.

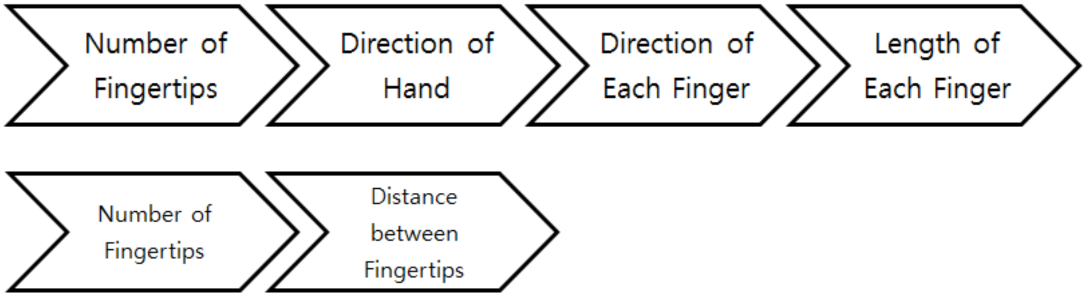


Figure 11. Steps of Hand Posture Recognition

However, the traditional decision tree has a problem with the axis being inclined because the branches are determined manually. As shown in Table 1, especially for skeleton-based hand region detection, characters that include K01, K02 and K04 are substantially inclined to one side. To resolve this problem, the proposed system uses SVM to set the branches. SVM is applied only for the classification of the hand angles layer and the classification of the finger angles layer for the skeleton-based feature detection decision tree. When SVM is applied, all gestures are recognized 100% for angles between -10° and 10°.

Table 1. Problem of Traditional Hand Posture Recognition along Degrees (Unit: %)

Gesture	-10°	-5°	0°	5°	10°	Gesture	-10°	-5°	0°	5°	10°
K01	100	100	100	0	0	K15	100	100	100	0	0
K02	100	100	100	0	0	K16	100	100	100	0	0
K03	0	100	100	100	0	K17	100	100	100	100	100
K04	0	100	100	0	0	K18	100	100	100	100	100
K05	100	100	100	0	0	K19	0	0	100	100	100
K06	100	100	100	100	100	K20	0	0	100	100	100
K07	0	100	100	100	0	K21	0	0	100	100	100
K08	50	100	100	50	0	K22	0	0	100	100	100
K09	100	100	100	100	100	K23	0	100	100	100	100
K10	100	100	100	100	0	K24	100	100	100	100	100
K11	100	100	100	100	100	K25	100	100	100	100	100
K12	0	100	100	100	100	K26	0	100	100	100	100
K13	100	100	100	100	100	K27	100	100	100	100	100
K14	0	100	100	100	100	K28	100	100	100	100	100

5. Results

To develop the proposed system, the operating system used was Windows 7 Enterprise K 32-bit OS, and the development tools were Visual Studio 2010, MFC library, and OpenCV 2.4.9. For the hardware, a DS325 infrared camera from SoftKinetic was used, and the computer specifications were Intel i5-3470 CPU with 3.47 GB memory. With the above hardware and software specifications, a hand was recognized from a distance of 11 cm to 30 cm from the camera, and the optimal distance for the best gesture recognition was 20 cm ±5 cm.

As shown in Figure 12, 28 templates that consisted of 14 consonants and 14 vowels were tested.

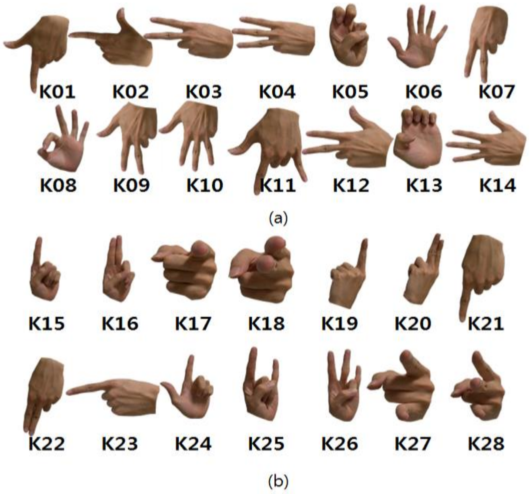


Figure 12. Sample Templates for Hand Posture Recognition (a) Consonant, (b) Vowel

The results show that the proposed system excels at recognizing rotating motions, unlike other existing systems. Whereas there is a problem recognizing certain gestures, such as "ㄱ" because of the inclination to one side, our proposed system demonstrated no problems with revolving gestures that are -10° to 10° from the center. The rotating gestures were measured in the direction of the revolving hand, where the position that corresponds to the x-axis was set to zero for the test. Therefore, we tested rotate angle between -25° and 25°. Table 2 shows the result of the rotation with proposed method.

Table 2. Result of Proposed Method along Degrees (Unit: %)

Gesture	-25°	-10°	0°	10°	25°
K01	75	100	100	100	75
K02	75	100	100	100	75
K03	100	100	100	100	100
K04	100	100	100	100	100
K05	50	100	100	100	50
K06	100	100	100	100	100
K07	100	100	100	100	100
K08	25	100	100	100	25
K09	100	100	100	100	100
K10	100	100	100	100	100
K11	100	100	100	100	100
K12	25	100	100	100	25
K13	100	100	100	100	100
K14	100	100	100	100	100
K15	75	100	100	100	75
K16	75	100	100	100	75
K17	100	100	100	100	100
K18	100	100	100	100	100
K19	75	100	100	100	75
K20	75	100	100	100	75
K21	75	100	100	100	75
K22	75	100	100	100	75
K23	75	100	100	100	75
K24	75	100	100	100	75
K25	100	100	100	100	100
K26	100	100	100	100	100
K27	100	100	100	100	100
K28	100	100	100	100	100

Figure 13 is the result of a test in which the phrase "눈" which means "eye," was inputted, and the characters were displayed on the screen.

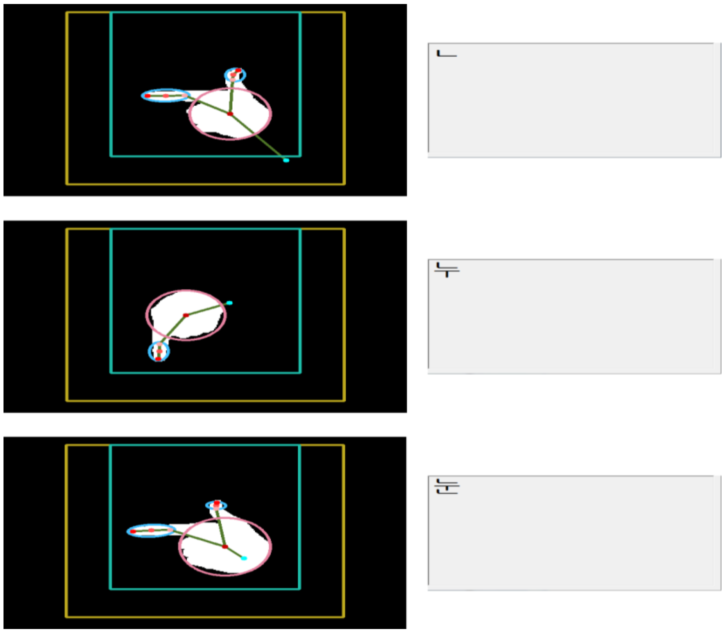


Figure 13. Hand Gesture Recognition Example

6. Conclusion

In this study, an improved algorithm for recognizing hand shapes was proposed, and a sign language recognition system was developed for testing. As a result of the test, although the proposed system was slightly affected by angles, its recognition rates were found to be excellent. Future study is required for a simpler way of inputting motions as well as shapes.

Acknowledgements

This work was supported by the Seoul R&BD Program (SS110013) and Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (2013R1A1A2012012).

This paper is a revised and expanded version of a paper entitled [Depth-based Static Sign Language Recognition System] presented at [NGCIT 2014, Oct 24-26, 2014 at Hochimin, Vietnam].

References

[1] M. M. Zaki and S. I. Shaheen, "Sign language recognition using a combination of new vision based features", Pattern Recognition Letters, vol. 32.4, **(2011)**.

[2] R. Yang and S. Sarkar, "Coupled grouping and matching for sign and gesture recognition." Computer Vision and Image Understanding, vol. 113.6, **(2009).**

[3] C.-F. Juang, S.-H. Chiu and S.-J. Shiu, "Fuzzy system learned through fuzzy clustering and support vector machine for human skin color segmentation", Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, vol. 37.6, **(2007).**

[4] C.-F. Juang and K.-C. Ku, "A recurrent fuzzy network for fuzzy temporal sequence processing and gesture recognition", Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, vol. 35.4, **(2005).**

[5] M.-H. Yang N. Ahuja and M. Tabb, "Extraction of 2d motion trajectories and its application to hand gesture recognition", Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 24.8, **(2002).**

[6] C. Nolker and H. Ritter, "Visual recognition of continuous hand postures", Neural Networks, IEEE Transactions on, vol. 13.4, **(2002).**

[7] Y. Fang, K. Wang, J. Cheng and H. Lu, "A real-time hand gesture recognition method", Multimedia and Expo, 2007 IEEE International Conference on. IEEE, **(2007).**

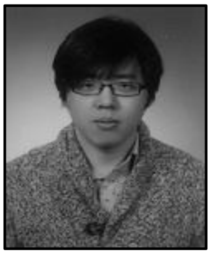
[8] P. Suryanarayan, A. Subramanian and D. Mandalapu, "Dynamic hand pose recognition using depth data", Pattern Recognition (ICPR), 2010 20th International Conference on IEEE, **(2010).**

[9] Ren, Zhou, Junsong Yuan and Zhengyou Zhang, "Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera", Proceedings of the 19th ACM international conference on Multimedia, ACM, **(2011).**

[10] X. Zabulis, H. Baltzakis and A. Argyros, "Vision-based hand gesture recognition for human-computer interaction", The Universal Access Handbook LEA, **(2009).**

[11] Z. Zafrulla, *et al.*, "A novel approach to american sign language (asl) phrase verification using reversed signing", Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on IEEE, **(2010).**

Authors



Ki Sang Kim, Ph. D. candidate at the Computer Vision Lab. in the School of Media at Soongsil University. He received his B. S. degree in Computer Engineering from Soongsil University in 2007, his M. S. degree in Media from Soongsil University in 2009.

Su Kyung Kim, M. S. candidate at the Computer Vision Lab. in the School of Media at Soongsil University. She received her B. S. degree in Computer Engineering from Korea Polytechnic University in 2013.

Hyung Il Choi, Professor in the School of Media at Soongsil University. He received his B. S. degree in Electronic Engineering from Yonsei University in 1979, his M. S. degree in 1983, and his Ph. D. in Electrical Engineering and Computer Science from the University of Michgan in 1987.