

User-Friendly Interface Control Using Segmented Skin Areas

Seok-Woo Jang¹ and Sang-Hong Lee^{2*}

¹*Department of Software, Anyang University
22, 37-Bungil, Samdeok-Ro, Manan-Gu, Anyang 430-714, Republic of Korea*

^{2*}*Department of Computer Engineering, Anyang University
22, 37-Bungil, Samdeok-Ro, Manan-Gu, Anyang 430-714, Republic of Korea*

¹*swjang7285@gmail.com, ^{2*}shleedosa@anyang.ac.kr*

Abstract

This paper proposes a more natural game interface capable of controlling a game in the way of segmenting hand region robustly, tracing the segmented hand region, and recognizing a user's gesture. The proposed method is able to detect a user's eyes, extract reliable skin color samples from the region around the detected eyes, and to create the skin color model most suitable to users. The created skin model is used to segment a user's hand region robustly. The segmented hand region is then traced so as to recognize a user's static and dynamic gestures, which are used as a game interface. To evaluate the proposed method, a conventional method and the proposed one were compared. As a result, the proposed method extracted a user's hand region more robustly than the conventional one and thereby more accurately recognized a user's gestures. The proposed technique in this paper is expected to be used as a tool to control various game programs to which a similar interface can be applied.

Keywords: Segmentation, Feature extraction, Hand region, Skin area, Color image

1. Introduction

With the web population, the rapid development of wireless internet technology, and the spread of tablet PC and smartphone, the environment where users are able to play games always regardless of time and place is established. With the support of the environment, various types of games have been released, and the demands for games have been on the increase. In particular, the age groups of game users vary from children to the elderly, and users request more diverse game contents. Accordingly, the game industry is more booming than other industries [1-4].

Of various game techniques, an interface of controlling a user's gesture taken for a game is very important. Generally, a user plays a game with the use of a keyboard, a mouse, or a joystick. With the lapse of time, a user wants to play a game in a more natural circumstance. In particular, people with the physical disabilities want to have a more convenient circumstance where they are able to play a game with the use of their gesture or their hand motion, rather than the use of the interface device fixed to a computer [5-7].

Hand gesture recognition is one of the useful techniques to implement a convenient and effective game interface, and there has been much relevant research. In [8], 3-axis acceleration sensors were attached to a user's wrist and ankle in order to develop a tangible game that supports interaction between a user and a game. Based on these sensor data, a user's motion was measured so as to classify a user's motion situation. In [9], the vision-based interface for Tetris game was proposed. The proposed interface was able to recognize a user's hand gesture in the images that were received continuously in real time

Received (October 11, 2017), Review Result (December 19, 2017), Accepted (January 15, 2018)

* Corresponding Author

from a camera, and to use the recognized gesture as a game command. Six commands for Tetris game are defined as static gestures which are a user's hand posture and as dynamic gestures which are a user's hand posture and motion. In [10], in order to recognize a user's hand gestures, simulated annealing-based multi-direction search technique was proposed. This technique showed relatively stable recognition results, but it required a lot of calculation. As a result, it was not suitable for real-time application. In [11], in order to design an interactive game application, the system 'GERMANE' with the interface capable of tracing a user's hand gestures in real time was developed. This system exploited a web camera to photograph a user's hand motion, and a game screen was projected to a separate large LCD.

As described earlier, in order to provide a more natural game interface, many techniques using a user's gestures or motion were proposed. However, there are still restrictions to be solved. First, although such an interface is more convenient than an interface attached to a computer like mouse or keyboard, it is necessary to attach a particular device to a part of the body. Secondly, the system with the game interface using a user's hand motion only without any attached device exploits the fixed skin color distribution model already defined in order to extract a user's hand region. As a result, it is impossible to segment the hand region robustly for surroundings.

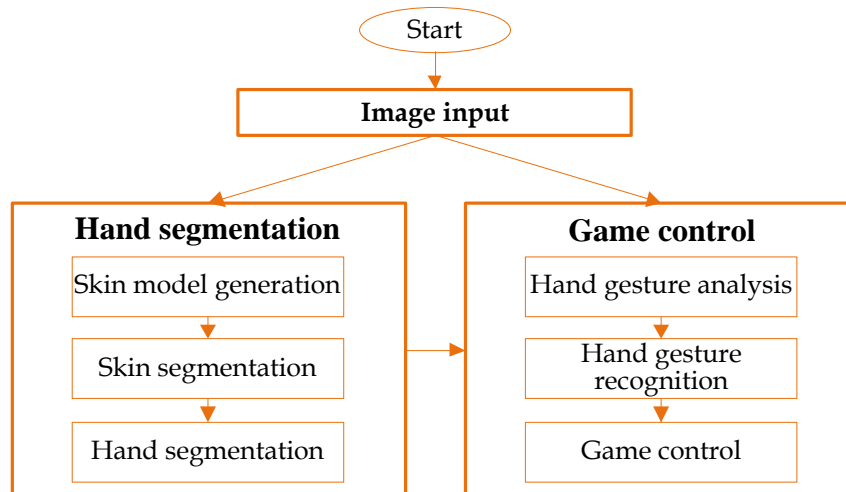


Figure 1. Overall Flow of the System

In this paper, we describe the method of implementing the natural and robust game interface with the use of a user's hand motion only. Unlike conventional techniques, the method proposed in this paper does not exploit the fixed skin color distribution model, but adaptively creates a user's skin color distribution model to split the hand region. To do that, a user's upper body image based on its facial region is obtained before a game starts. In this way, the user's eyes are detected, and the skin color samples around the eyes are extracted in order to create the skin color model suitable to the user. With the use of the created skin model, the hand region is segmented robustly. Subsequently, the hand image is obtained as input data in order to classify and recognize a type of hand gesture. The result is used as an interface to control the game. Figure 1 shows the overview of the game interface proposed in this paper.

This paper is comprised of as follows: Section 1 describes the motivation and background of this paper, and the overview; Section 2 explains the method of obtaining a user's hand region with the use of adaptive skin color model; Section 3 describes the technique of analyzing main gestures necessary for a game interface; Section 4 shows the results of the experiment which was conducted to evaluate the performance of the robust

hand region segmentation-based game interface proposed in this paper; Section 5 describes the conclusion and future research direction.

2. Hand Region Segmentation

In this step, the work of splitting a user's hand region robustly is executed in order to implement a game interface. Unlike conventional hand region extraction methods using the pre-defined fixed skin color distribution model, the method proposed in this paper obtains an image to detect a user's eyes and then extracts the skin color around the eyes to create the user's skin color model adaptively. To detect a user's eyes, this paper uses the eye map proposed by Hsu [12]. To do that, an input image is first converted to YC_bC_r color space [13-14], and then color and gray values are used to generate two eye maps. Equation (1) is used to extract a color eye map. In Equation (1), C_b and C_r are C_b and C_r values in the YC_bC_r color space. C_b^2 , $(255-C_r)^2$, and C_b/C_r are normalized to a value between 0 and 255.

$$EyeMapC = \frac{C_b^2 + (255 - C_r)^2 + (C_b / C_r)}{3} \quad (1)$$

Equation (2) is used to create an eye map in the way of executing dilation and erosion operations [15-17] with the use of gray value. In Equation (2), \otimes means dilation and \oplus represents erosion, and $g(x, y)$ is a gray value of an input image. The color value in the eye region includes a dark region and a bright region. Accordingly, dilation and erosion are applied to the gray value of the image in order to extract an eye map on the basis of the contrast.

$$EyeMapL = \frac{Y(x, y) \otimes g_\sigma(x, y)}{Y(x, y) \oplus g_\sigma(x, y) + 1} \quad (2)$$

Two eye maps created in Equation (1) and Equation (2) are integrated as shown in Equation (3) in order to create the final eye map.

$$EyeMap = (EyeMapC) \text{ and } (EyeMapL) \quad (3)$$

Subsequently, the eye map is binarized with the use of a threshold in order to extract candidate eye regions. Whether the extracted candidates are real eyes is judged with the use of the geometric structure factors of the eyes in the face, including length, density, and aspect ratio.

In this paper, a skin sample is extracted from the five-fold enlarged square region of the least enclosing rectangle for eye after the eyes are detected. Generally, skin color is highly likely to be distributed in the region selected around the eyes. If there are glasses or other background regions, it is necessary to classify the selected region into skin region and non-skin region. Therefore, the proposed method calculates the distance between the selected skin sample and average skin color with the use of Equation (4) and creates a skin map.

$$SkinMap = 255 - \frac{|\hat{C}_r - C_r| + |\hat{C}_b - C_b|}{2} \quad (4)$$

In Equation (4), \hat{C}_r and \hat{C}_b represent C_r and C_b values of general skin color, respectively. A skin map has a value ranging from 0 to 255. The closer to average skin color a selected sample is, the closer to 255, the value is.

To select skin sample pixels from a skin map, this paper exploits the histogram binarization proposed by Otsu [18-21]. Otsu method statistically selects an optimal

threshold to binarize contrast histogram without previous knowledge, and is known to have excellent performance when histogram has two probability densities.

To model skin color, it is necessary to find a proper color space and a cluster related to skin color in the space. Generally, the skin color sample distribution in the $C_b C_r$ space has an ellipse, but the position of an ellipse is different depending on test images. Therefore, the proposed method selects a skin sample directly from each input image so as to create an adaptive skin color model. The proposed skin color model is generated with the use of the skin samples selected from the regions around the eyes, and the center of an elliptical cluster and how widely it is distributed are determined. The skin color model in this paper is defined as shown in Equation (5).

$$\frac{(x-ec_x)^2}{a^2} + \frac{(y-ec_y)^2}{b^2} \leq 1 \quad (5)$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} C_b - C'_b \\ C_r - C'_r \end{bmatrix}$$

$$C'_b = \frac{1}{K} \sum_{t=0}^{K-1} C_b(t), \quad C'_r = \frac{1}{K} \sum_{t=0}^{K-1} C_r(t)$$

In Equation (5), (C_b, C_r) represents C_b and C_r values of a test image, and θ is a rotation angle of an ellipse. (a, b) means the long axis and short axis of an ellipse, and (ec_x, ec_y) is an offset value of the rotation error of an ellipse. (C'_b, C'_r) means the center of a skin model. t means the index to display each skin sample and K represents the count of the selected skin colors.

As such, after an elliptical model for skin color distribution is created, the hand region segmentation for an input image is executed with the use of the created model. Therefore, it is possible to segment the hand region robustly for various surrounding conditions.

3. Hand Gesture Analysis

Generally, a hand gesture is used as the most intuitive way to operate or instruct an object [22-24]. A gesture shows its possibility as an intuitive and efficient tool for interaction between computer and human in the real world through application programs like 2D and 3D mouse, TV operation, and computer game.

A hand gesture can be classified into a static gesture to show spatial information, or a hand posture, into a dynamic gesture to show temporal information. This paper exploits a static gesture and a dynamic gesture in an integrated way so as to implement a vision-based effective game interface.

In this paper, four static gestures and four dynamic gestures were defined as shown in Figure 4. First, static gestures include click, stop, pause, and move gestures. Click gesture is used to perform mouse-clicking operation. As shown in Figure 4 (a), the gesture is taken in the way of putting all fingers together and opening the hand. Stop gesture is used to stop playing a game. As shown in Figure 4 (b), the gesture is taken in the way of stretching out all fingers and opening the hand. Pause gesture is used to pause a game. As shown in Figure 4 (c), the gesture is taken in the way of clenching the fist and stretching the index finger. Move gesture is used to move a mouse. As shown in Figure 4 (d), the gesture is taken in the way of clenching the fist. Dynamic gestures are taken in the way of clenching the fist and moving up, down, left and right as shown in Figure 4 (e) to (h), and each gesture is equal to the move of a mouse in each direction.

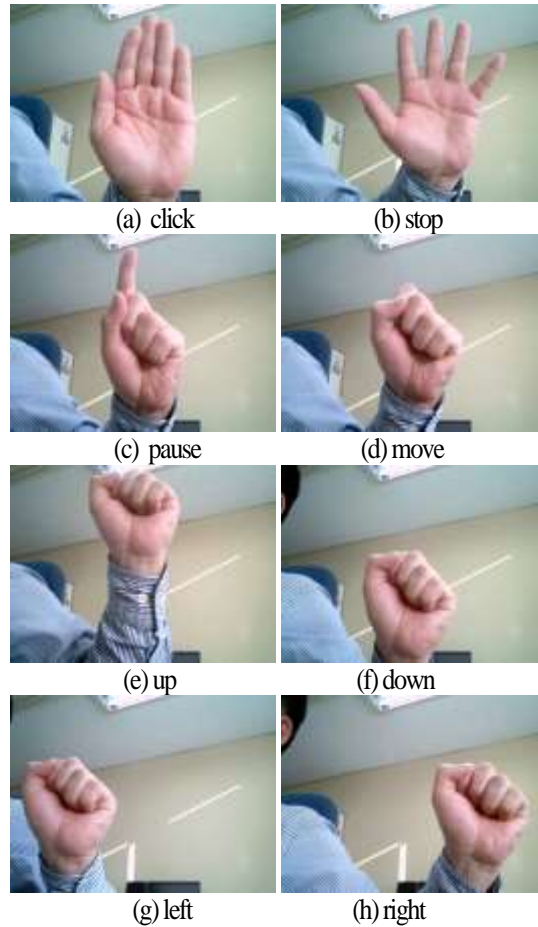


Figure 2. Gesture Types

In Equation (6), W and H represent the width and height of an image. CG_x and CG_y mean the centers of the gravity of the hand region, IC_x and IC_y are the center positions of an input image. The position feature has a value ranging from 0 to 1.

$$\frac{1}{\sqrt{2}} \times \sqrt{\left(\frac{|CG_x - IC_x|}{W}\right)^2 + \left(\frac{|CG_y - IC_y|}{H}\right)^2} \quad (6)$$

As for the size of the hand region, the rate of the size of the hand region for the whole image is used. Usually, the size of the hand region accounts for more than a certain rate of an image. Accordingly, when the size defined in Equation (7) is more than a certain threshold, it is highly likely to include the segmented hand include. In Equation (7), N means the count of the pixels of the hand region.

$$\frac{N}{W \times H} \quad (7)$$

As for the ratio of long axis and short axis, the ratio of the length and width of the segmented hand region is calculated. The ratio of long axis and short axis is calculated in the way of calculating the height and width of the least adjacent rectangle of the hand region as shown in Equation (8). This feature is useful to identify a fist from other hand shapes. If the hand region expresses a fist, its feature value is close to 1; if it has other hand shapes, the value is less than 1. In Equation (8), LER means the least enclosing

rectangle of the hand regions, and $L_{hor}(LER)$ and $L_{ver}(LER)$ represent the width and length of the least enclosing rectangle.

$$\frac{L_{hor}(LER)}{L_{ver}(LER)} \quad (8)$$

As for density, the ratio of the segmented hand region and the least enclosing rectangle of the region is calculated. Density represents how dense a region is, and is defined as shown in Equation (9) [25]. In Equation (9), if this feature value is 1, it means that the density of the hand region is the highest.

$$\frac{N}{L_{hoz}(LER) + L_{ver}(LER)} \quad (9)$$

As for perimeter, the ratio of the square of the perimeter of the hand region and the least adjacent rectangle area is calculated as shown in Equation (10). This feature is useful to identify the fist or the opened hand or the hand with stretched fingers. The feature value of the hand region with stretched fingers is larger than that of the fist or of the opened hand. In Equation (10), perimeter represents the perimeter of the segmented hand region.

$$\frac{perimeter^2}{L_{hoz}(LER) + L_{ver}(LER)} \quad (9)$$

Dynamic gestures are a user's fist gestures moving up, down, left and right. Therefore, when a user's fist is moved, the trajectory of the center of gravity of the hand continues to be accumulated and then its direction is calculated.

4. Experimental Results

The computer used for this experiment has Intel Pentium Core 2 Duo CPU, 8GB memory, and Microsoft Windows 7 as an operating system. The programming languages to implement the system proposed in this paper were Microsoft Visual C++ 2015 and OpenCV. In addition, to photograph a user's hand region, Microsoft LifeCam VX-1000 web cam easy to be obtained was used.

In this paper, the interface of Infection game [26] as a dialogue box-based application was implemented with the trace of a user's hand region. Infection game is a well-known board game, being played with the following rules:

- First, a bacterial cell can be moved one space or two spaces left, right, forward, and backward, or diagonally.
- Secondly, in case of one space move, a bacterial cell is copied; in case of two-space move, it is moved.
- Thirdly, the other player's bacterial cells around a newly placed bacterial cell are infected and changed.
- Fourthly, if there is no space to move, the game ends.
- Fifthly, when the game ends, a player who has more bacterial cells on board becomes a winner.

The main screen of the game interface system implemented in this paper has three windows. The left window and the right top window are the main screens of the Infection game, and the right bottom window shows the detection of the hand region with the use of skin color distribution, and the traced detected hand region. In the right bottom window, the small red circle at the center represents the center of the hand region, and the large circle on the hand region boundary is used to easily judge how much the hand region is included in the circle.

Table 1 shows the table of the average recognition rate of each gesture in the proposed system. In the environment good to split the hand region relatively and with the optimal maintenance of the hand moving speed, this experiment was conducted. Of static gestures, move gesture had the highest recognition rate and pause gesture had the lowest recognition rate. The reason why move gesture had high recognition rate was that the ratio of long axis and short axis and density were good. Pause gesture had often low recognition rate when index finger segmentation was not performed rightly. Dynamic gestures had good recognition rates generally. However, when the hand region with severe reflection background was not accurately segmented, the rate of gesture recognition was low.

Table 1. Table Label

Gesture	Average recognition rate (%)
click	94.02
stop	96.13
pause	93.12
move	95.55
top	96.21
down	97.07
left	96.52
right	95.05

In order to conduct comparative evaluation of the proposed game interface performance, Chan method [9] and Hsu method [12] were implemented. The two methods were used to split a user's hand region, but the hand region recognition process was performed by the proposed method.

In this paper, each method was applied to play Infection game multiple times in different places. The reason for different places was to take into account a change in surroundings. In this experiment, a user's hand was moved slowly, normally, and fast. For performance evaluation, whether each algorithm recognizes four static gestures and four dynamic gestures successfully was measured in terms of average percentage. In this paper, each of four users took four static gestures and four dynamic gestures 30 times (10 times: slowly, 10 times: normally, 10 times: fast), and the mean of the recognition rates was calculated. Accordingly, each gesture was tested 120 times repeatedly. Figure 3 shows the graphs of the comparative results.

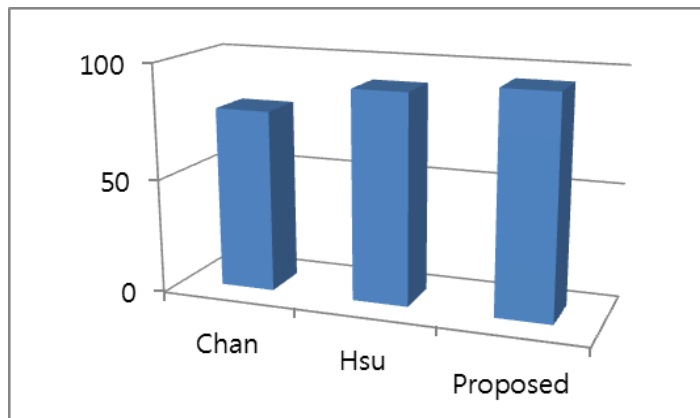


Figure 3. Performance Comparison

Chan method used the fixed threshold for skin color distribution in HSV space when a user's hand region was split. The method worked well in a stable environment. However,

when surroundings changed or a user changed, its accuracy lowered much. Hsu method used the ellipse model in YC_bC_r space to generate a skin color distribution model. Accordingly, the method had higher accuracy of hand region segmentation than the method using a simple and fixed threshold, but had lower performance than the proposed method using an adaptive skin color model.

Although the proposed method can have little different performance depending on surroundings, multiple experiments revealed that it was able to split a user's hand region and recognize hand gestures in a relatively stable way. Generally, the proposed method had lower accuracy of hand region segmentation when a user's hand region moved very fast, when background had a region similar to skin color, and when light reflection was too much.

The method proposed in this paper had no big change in recognition rate depending on user change. Of course, it was better to extract the hand region of a user with large hands than a user with small hands, and it was easier to create a skin color distribution model with a user wearing no glasses. Nevertheless, the factors little influenced the overall recognition rate. A change of external circumstances like light and background and a hand moving speed influenced the recognition rate.

5. Conclusion

This paper proposed the game interface capable of controlling a game more naturally with the use of the algorithm which can robustly split a user's hand motion without any attached device, and verified its appropriateness by applying the proposed one to the board game 'Infection'. The proposed method exploits the skin color around the detected eye to create a user's skin color model adaptively and split the hand region. Therefore, it is possible to detect the hand region robustly for surroundings. The detected hand region is analyzed in order to identify a static gesture and a dynamic gesture. Based on the analysis result, a game is controlled. It is expected that the interface system using a user's hand region recognition proposed in this paper is applicable to not only Infection, but can be used as a useful tool to control various games to which a similar interface can be applied.

In the future, more types of gestures will be designed in order for the interface enable to control a game. The proposed interface system will be applied to a more complicated game in order to verify its usability. In addition, prediction algorithms like Kalman filter will be applied so as to extract a user's fast moving hand region robustly.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2016R1D1A1A09917838). In addition, this paper is a revised and expanded version of a paper entitled [A Method of Extracting Human Hand Regions Based on Adaptive Skin Distribution] presented at International Conference on Green and Smart Technology, Jeju Island, Korea, on 1st December, 2017.

References

- [1] H. Yuan, X. Wei, F. Yang, J. Xiao, and S. Kwong, "Cooperative Bargaining Game-Based Multiuser Bandwidth Allocation for Dynamic Adaptive Streaming Over HTTP", *IEEE Transactions on Multimedia*, vol. 20, no. 1, (2018), pp. 183-197.
- [2] E.-Y. Kim and D. Ashlock, "Changing Resources Available to Game Playing Agents: Another Relevant Design Factor in Agent Experiments", *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 9, no. 4, (2017), pp. 321-332.

- [3] X. Nan, X. Guo, Y. Lu, Y. He, L. Guan, S. Li and B. Guo, "Delay-Rate-Distortion Optimization for Cloud Gaming with Hybrid Streaming", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 12, (2017), pp. 2687-2701.
- [4] K. S. Krishnan, A. Saha, S. Ramachandran and S. Kumar, "Recognition of Human Arm Gestures Using Myo Armband for the Game of Hand Cricket", *Proceedings of the IEEE International Symposium on Robotics and Intelligent Sensors*, Ottawa, Canada, (2017), pp. 389-394.
- [5] R. Horie and R. Nawa, "A Hands-on Game by Using a Brain-Computer Interface, an Immersive Head Mounted Display, and a Wearable Gesture Interface", *Proceedings of the IEEE 6th Global Conference on Consumer Electronics*, Nagoya, Japan, (2017), pp. 1-5.
- [6] E. H. El-Shazly, M. M. Abdelwahab, A. Shimada and R. Taniguchi, "Early Gesture Recognition with Adaptive Window Selection Employing Canonical Correlation Analysis for Gaming", *Electronics Letters*, vol. 52, no. 16, (2016), pp. 1379-1381.
- [7] P. Lu, Y. Chen and C. Dong, "Stereo Vision-based 3D Game Interface", *Proceedings of the International Symposium on Multi-Spectral Image Processing and Pattern Recognition*, vol. 7496, (2009), pp. 749609-749609-8.
- [8] S.-H. Kim and B.-K. Chae, "Design of Game Interface Based on 3-Axis Accelerometer for Physical Interactive Game", *Journal of Korean Institute of Intelligent Systems*, vol. 19, no. 4, (2009), pp. 538-544.
- [9] S.-H. Kim, J.-S. Chang and H.-J. Kim, "Vision-based Interface for Tetris Game", *Proceedings of the Korean Institute of Information Scientists and Engineers*, vol. 31, no. 1(B), (2004), pp. 775-777.
- [10] W. Liang, Y. Jia, F. Sun, B. Ning, T. Liu and X. Wu, "Visual Hand Tracking Using MDSA Method", *Proceedings of the IMACS Multi-conference on Computational Engineering in Systems Applications*, (2006) pp. 255-259.
- [11] A. T. S. Chan, H. V. Leong and S. H. Kong, "Real-time Tracking of Hand Gestures for Interactive Game Design", *Proceedings of the IEEE International Symposium on Industrial Electronics*, (2009), pp. 98-103.
- [12] R.-L. Hsu, M. Abdel-Mottaleb and A. K. Jain, "Face Detection in Color Images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, (2002), pp. 696-706.
- [13] B. Chen, X. Qi, X. Sun and Y.-Q. Shi, "Quaternion Pseudo-Zernike Moments Combining Both of RGB Information and Depth Information for Color Image Splicing Detection", *Journal of Visual Communication and Image Representation*, vol. 49, (2017), pp. 283-290.
- [14] J.-U. Yun, H.-J. Lee, A. K. Paul and J.-H. Baek, "Face Detection for Video Summary Using Illumination-Compensation and Morphological Processing", *Pattern Recognition Letters*, vol. 30, no. 9, (2009), pp. 856-860.
- [15] S. Lou, X. Jiang and P. J. Scott, "Algorithms for Morphological Profile Filters and Their Comparison", *Precision Engineering*, vol. 36, no. 3, (2012), pp. 414-423.
- [16] R. Su, C. Sun, C. Zhang and T. D. Pham, "A New Method for Linear Feature and Junction Enhancement in 2D Images Based on Morphological Operation, Oriented Anisotropic Gaussian Function and Hessian Information", *Pattern Recognition*, vol. 47, no. 10, (2014), pp. 3193-3208.
- [17] E. Hamuda, B. M. Ginley, M. Glavin and E. Jones, "Automatic Crop Detection under Field Conditions Using the HSV Color Space and Morphological Operations", *Computers and Electronics in Agriculture*, vol. 133, (2017), pp. 97-107.
- [18] H. Cai, Z. Yang, X. Cao, W. Xia and X. Xu, "New Iterative Triclass Thresholding Technique in Image Segmentation", *IEEE Transactions on Image Processing*, vol. 23, no. 3, (2014), pp. 1038-1046.
- [19] A. M. A. Talab, Z. Huang, F. Xi and L. HaiMing, "Detection Crack in Image Using Otsu Method and Multiple Filtering in Image Processing Techniques", *Optik - International Journal for Light and Electron Optics*, vol. 127, no. 3, (2016), pp. 1030-1033.
- [20] N. Otsu, "A Threshold Selection Method from Gray-Level Histogram", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 1, (1979), pp. 62-66.
- [21] X.-C. Yuan, L.-S. Wu and Q. Peng "An Improved Otsu Method Using the Weighted Object Variance for Defect Detection", *Applied Surface Science*, vol. 349, (2015), pp. 472-484.
- [22] C.-J Tsai, Y.-W. Tsai, S.-L. Hsu and Y.-C. Wu, "Synthetic Training of Deep CNN for 3D Hand Gesture Identification", *Proceedings of the International Conference on Control, Artificial Intelligence, Robotics and Optimization*, Prague, Czech Republic, (2017), pp. 165-170.
- [23] P. D. S. H. Gunawardane and N. T. Medagedara "Comparison of Hand Gesture Inputs of Leap Motion Controller and Data Glove into a Soft Finger", *Proceedings of the IEEE International Symposium on Robotics and Intelligent Sensors*, Ottawa, Canada, (2017), pp. 62-68.
- [24] H.-I. Suk, B.-K. Sin and S.-W. Lee, "Hand Gesture Recognition Based on Dynamic Bayesian Network Framework", *Pattern Recognition*, vol. 43, no. 9, (2010), pp. 3059-3072.
- [25] X. Teng, B. Wu, W. Yu and C. Liu, "A Hand Gesture Recognition System Based on Local Linear Embedding", *Journal of Visual Languages and Computing*, vol. 16, no. 5, (2005), pp. 442-454.
- [26] Y.-S. Kim, "A Complete Guide to Visual C ++ 6", Youngjin Press, Seoul, (2004).

Authors



Seok-Woo Jang, he received the B.S., M.S., Ph.D. degrees in Computer Science from Soongsil University, Seoul, Korea, in 1995, 1997, and 2000, respectively. From October 2003 to January 2009, he was a Senior Researcher with the Construction Information Research Department at Korea Institute of Construction Technology (KICT), Ilsan, Korea. Since March 2009, he has been a Professor in the Department of Software, Anyang University, Korea. His primary research interests include robot vision, augmented reality, video indexing and retrieval, cluster computing, biometrics and pattern recognition.



Sang-Hong Lee, he received the B.S., M.S., and Ph.D. degrees in Computer Science from Gachon University, Korea in 1999, 2001, and 2012, respectively. He is currently an assistant professor in the department of computer engineering at Anyang University, Korea. His research focuses on neuro-fuzzy systems, stocks prediction systems, and biomedical prediction systems.