# A New NUI Method for Hand Tracking and Gesture recognition Based on User Experience

Wenkai Xu and Eung-Joo Lee

*Department of Information & Communications Engineering, Tongmyong University,
Busan 608-711, Korea
xwk6298@hotmail.com, ejlee@tu.ac.kr*

### *Abstract*

*Human gesture recognition is a non-verbal part for interaction or movement that can be used to involves real world and virtual world. In this paper, we explain a study on natural user interface (NUI) in human gesture recognition using RGB color information and depth information by Kinect camera from Microsoft Corporation. To achieve the goal, hand tracking and gesture recognition have no major dependencies of the work environment, lighting or users' skin color, libraries of particular use for natural interaction and Kinect device, which serves to provide RGB images of the environment and the depth map of the scene were used. An improved CamShift tracking algorithm combined with depth information is used to tracking hand motion, and then an associative method of HMM and FNN is propose for gesture recognition step. The experimental results show out its good performance and it has higher stability and accuracy as well.*

***Keywords:*** *NUI, depth information, CamShift, gesture recognition, Kinect*

## 1. Introduction

Massive technology shift has played a dominant role in all disciplines of science and technology. The use of hand gesture is an active area of research in the vision community, mainly for the purpose of sign language recognition and Human-Computer Interaction (HCI). The history of interaction and interface design is a flow and step from complex interaction to simple interaction between human and computer [1, 2]. The word natural interaction came from Natural User Interface (NUI) that use human body interaction and voice interaction, verbal and non-verbal communication, becoming a one of Human-Computer Interaction area. It is an evolution from Graphical User Interface (GUI).

HCI is emerged as a new field with the motivation to bridge the communication gaps among the humans and computers. Gesture and posture recognition are application areas in HCI to communicate with computers. A gesture is spatiotemporal pattern which maybe static, dynamic or both. Static morphs of the hands are called postures and hand movements are called gestures. In gesture recognition, Liu, *et al.*, [3] developed a system to recognize 26 alphabets by using different HMM topologies. Hunter, *et al.*, [4] used HMM for recognition in their approach where Zernike moments are used as image features for sequence of hand gestures. In the last decade, several methods of potential applications in the advanced gesture interfaces for HCI have been suggested but these differ from one to another in their models. Some of these models are Neural Network [5], Hidden Markov Model (HMM) [6] and Fuzzy Systems [7]. Hidden Markov Model (HMM) is one of the most successful and widely used tools for modeling signals with spatiotemporal variability [8]. It has been successfully applied in the area of speech recognition and is one of the mostly successfully used methods in the research area of dynamic gesture recognition.

Traditional user interfaces for controlling a virtual camera in the 3D virtual space are achieved by using devices such as mice and keyboards. Since the equipments are not familiar with usage of the user interface in 3D space, gesture based user interfaces have been used to give better immersion to the users by allowing them to more naturally and friendly control the virtual camera and/or objects in 3D space. There are many attempts to support gesture user interfaces by using image based approaches [11-18].

In this paper, an improved CamShift tracking algorithm combined with depth information is used to tracking hand motion by Kinect, and then an associative method of HMM and FNN is propose for gesture recognition step, which combines ability of HMM model for temporal data modeling with that of fuzzy neural network for fuzzy rule modeling and fuzzy inference.

## 2. Hand Tracking by Using Improved Camshift Algorithm based on Depth Data

To track hand in video frame sequences, the image data has to be represented as a probability distribution. Distributions derived from video image sequences change over time, so the mean shift algorithm should be adapted dynamically to the probability distribution when it is tracking. CamShift tracking algorithm based on color performs well in solving the bottom problems of computer vision. Due to its robust and real-time quality, CamShift has become a basic tracking method which can adapt to the continuous variation of the shape and size of the target, compute fast and has strong anti-jamming capability, guaranteeing the stability and real-time of the system. CamShift algorithm is a dynamic change in the distribution of the density function of the gradient estimate of non-parametric methods. The course of algorithm is as follows:

1. Choose an initial search window W1;

2. Run the Mean-shift algorithm;

3. Resize the search window according to the result of Step (2), and get a new window W2;

4. Use W2 as the initial search window for the next video frame and repeat the algorithm. The tracking result is displayed as below.

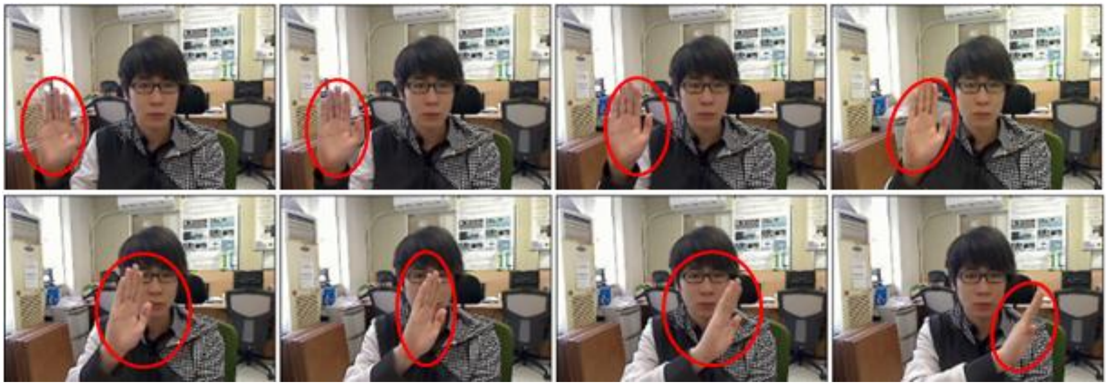
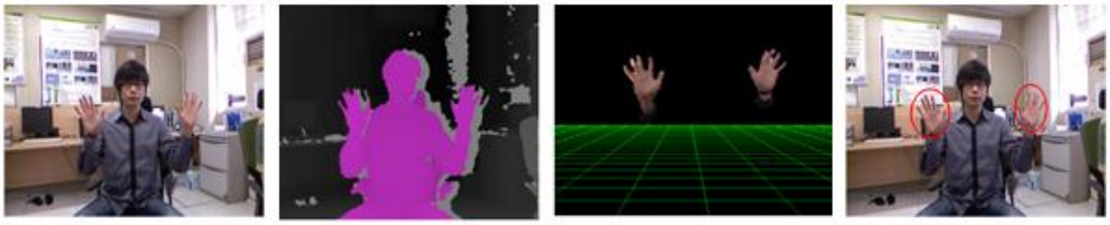


**Figure 1. The Experimental Results using CamShift Tracking Algorithm**

Because the CamShift algorithm is based on color images, tracking error will easily occur when there is similar color in background as Figure 1. Considering the object is usually separated from the surrounding environment in depth, and has fixed moving range, so threshold segmentation in depth map can accurately distinguish the area of objective from the background. According to reference [9], we combined depth information with traditional CamShift tracking algorithm by using Kinect. The tracking results are shown as Figure 2.



(a) (b) (c) (d)

**Figure 2. The Experimental Results using Improved CamShift Tracking Algorithm: (a). Original Image, (b). Depth Image, (c). Hands Segmentation, (d). Tracking Result**

By using depth data, the improved CamShift algorithm can avoid the interference from background and the object that similar to skin color. Figure 2(b) shows us the depth image of video sequence, basing on the different information on depth orientation we can define threshold to segment the objective we concern. For simplifying the processing, we transform the depth image to grayscale; meanwhile the depth information will be transformed to gray value from 0 to 255. As hands area are the nearest to camera, and then it is the human body, the background is the furthest to camera, we define that depth concentrate on three ranges: [0, 70] for the human hand area; [100, 175] for the human body; most grayscale distributes in [180, 255], standing for the background which contains a number of unuseful information, as it is the furthest object to Kinect camera.

## 3. Feature Extraction and Gesture Recognition

### 3.1 Feature Extraction based on Orientation and Velocity

There are many feature types such as the chain code, mesh code, momentum, and the MRF (Markov random fields), but all these features are based on only three basic features from a gesture trajectory: location, orientation and velocity, as shown in Figure 3. In Figure 3, $(X_{t-1}, Y_{t-1})$ indicates the geometric center of hand at time *t-1*, $(X_t, Y_t)$ indicates the geometric center of hand at time *t*, as well as $(X_{t+1}, Y_{t+1})$. $\theta_{1t}$ and $\theta_{2t}$ are angle feature between two successive points. $D_t$ is the distance between two points, and  is the velocity feature. The moment distance $D_t$, changing angle $\theta_{1t}$, $\theta_{2t}$ and velocity feature are calculated for each position by the following equations.

$$D_t = \sqrt{(X_t - X_{t-1})^2 + (Y_t - Y_{t-1})^2} \tag{1}$$

$$\theta_{1t} = \arg\tan(\frac{Y_t - Y_{t-1}}{X_t - X_{t-1}}) \tag{2}$$

$$\theta_{2t} = \arg\tan(\frac{Y_{t+1} - Y_t}{X_{t+1} - X_t}) \tag{3}$$

$$V_t = \sqrt{(X_t - X_{t+1})^2 + (Y_t - Y_{t+1})^2} \tag{4}$$

We analysis these features contained in gesture trajectory, and combining these feature to set up coordinates system for gesture recognition state.

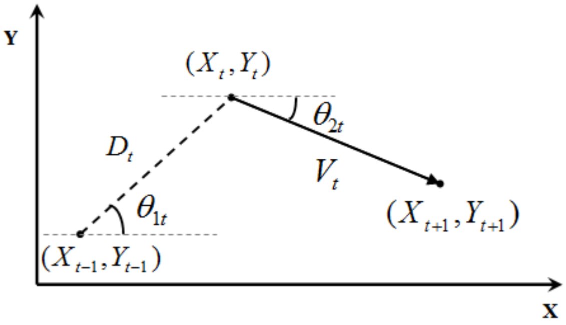


**Figure 3. Basic Feature between Two Points of a Gesture Sequence**

As described above, the orientation feature is represented by the angle between two successive points. We use a chain code to convert these angles into feature codes. A chain code is a useful coding method for converting angle values to feature code. A chain code is based on a grid and represents 4-connectivity or 8-connectivity according to the connectivity. An 8-connectivity code is assigned to each of eight possible directions between two points, where 0 is assigned to movement to the right, and the code increases by one going counterclockwise as shown in Figure 4 (a).

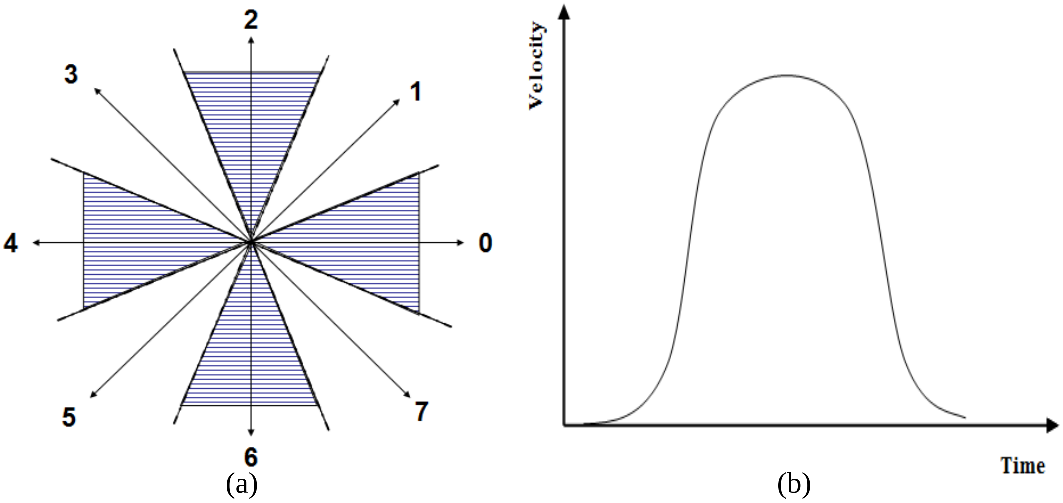


**Figure 4. (a) Eight-directional Chain Codes, (b) The Velocity Change Rate of Simple Gesture**

The velocity feature is based on the fact that each gesture is made at different speeds. For example, a simple gesture, such as a “leftwards” gesture, has an almost stable velocity during the gesture generation. This velocity feature is measured by the distances of two successive points, such as points $(X_t, Y_t)$ and $(X_{t+1}, Y_{t+1})$. Figure 4(b) shows the differences in velocity values for simple gestures.

The velocity feature v is generated by the following equation.

$$V_{\max} = \underset{t=1}{\overset{n}{Max}}(V_t) \tag{5}$$

$$v = \frac{V_t}{V_{\max}} \tag{6}$$

Here, $V_t$ is the value from Eq. (4) and $V_{\max}$ is the maximum velocity in one gesture. The v value has the result of normalizing size. The v values obtained above are between 0.0 to 1.0, and can be converted into feature codes by partitioning the range from 0.1 to 1.0.

In this paper, we combine the orientation feature with velocity feature for gesture feature extraction. As Figure 4(a) shown, we regard any path in each shadow area as the standard eight-orientation chain codes we defined for simplify. Meanwhile the velocity feature is used to detect initial signal and finished signal; on the other hand, it is used to discriminate the movement on depth orientation as well.

### 3.2 Gesture Recognition using HMM-FNN

As we know, Fuzzy Neural Network has strong ability for fuzzy rule modeling and fuzzy inference due to its integration of fuzzy set theory and Neural Network together. Since traditional FNN cannot model temporal data and conventional HMM do not own ability for fuzzy inference, we integrate the two models together to represent complex gesture trajectory and perform inference by the integrated HMM-FNN model [10], which is shown in Figure 5, for the recognition of dynamic gesture.

HMM-FNN model includes five layers. Its first layer, second layer and HMM layer constitute the fuzzy preprocessing part, third layer and fourth layer constitute fuzzy inference part, fifth layer is the defuzzification part of HMM-FNN and produce distinct output.

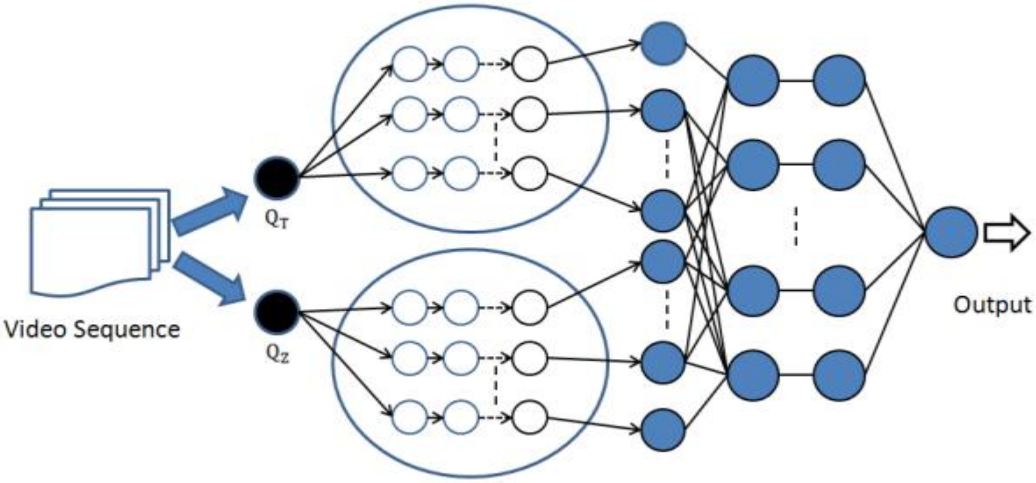


**Figure 5. HMM-FNN Model we Proposed**

As we can see, Qt and Qz are corresponding to the two movement components of dynamic gesture: Qt express the changing sequence of 2D gesture trajectory and Qz express the changing sequence of the depth information.

Training of proposed HMM-FNN model includes two parts: Firstly, the training of HMM model, it is the re-estimation of parameters in state transition matrix, output probability matrix or GMM's expectation and variance. Secondly, the weights of FNN will be trained after HMM training. Back-Propagation (BP) algorithm is chosen for the training of connecting weights. When it reaches maximum iteration number or converge, the training process will stop.

## 4. Experimental Results

The hand recognition system is running on the hardware environment of Intel (R) Core (TM) 2 (2.93GHz), a Kinect camera, and the software environment of Windows 7 (32bit) and Visual Studio 2008 using Kinect SDK.

In our experiments, we firstly tested the improved CamShift algorithm and original CamShift tracking method respectively on the computer. Table 1 shows us the results. The system process speed is 30 fps. The experimental results show out it has better performance than CamShift algorithm, and it has higher stability and accuracy as well.

**Table 1. Tracking Experiments Results**

| | Original Camshift | Improved Camshift |
|---|---|---|
| Corr. Rate | 96.7% | 99.4% |
| Miss Rate | 3.3% | 0.6% |

Secondly, we test gesture recognition rate. In this paper, we define 8 kinds of gestures: six basic gestures are shown as Figure 6, expressing leftwards, rightwards, upwards, downwards, zoom in and zoom out.; Besides, two kinds of gestures based on depth space also are designed as “Click” and double “Click”.

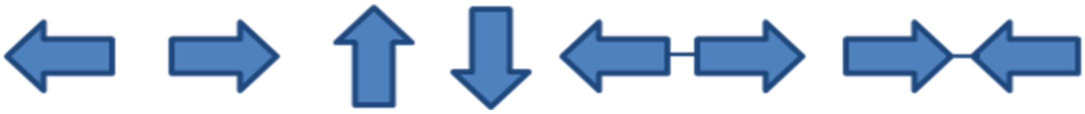
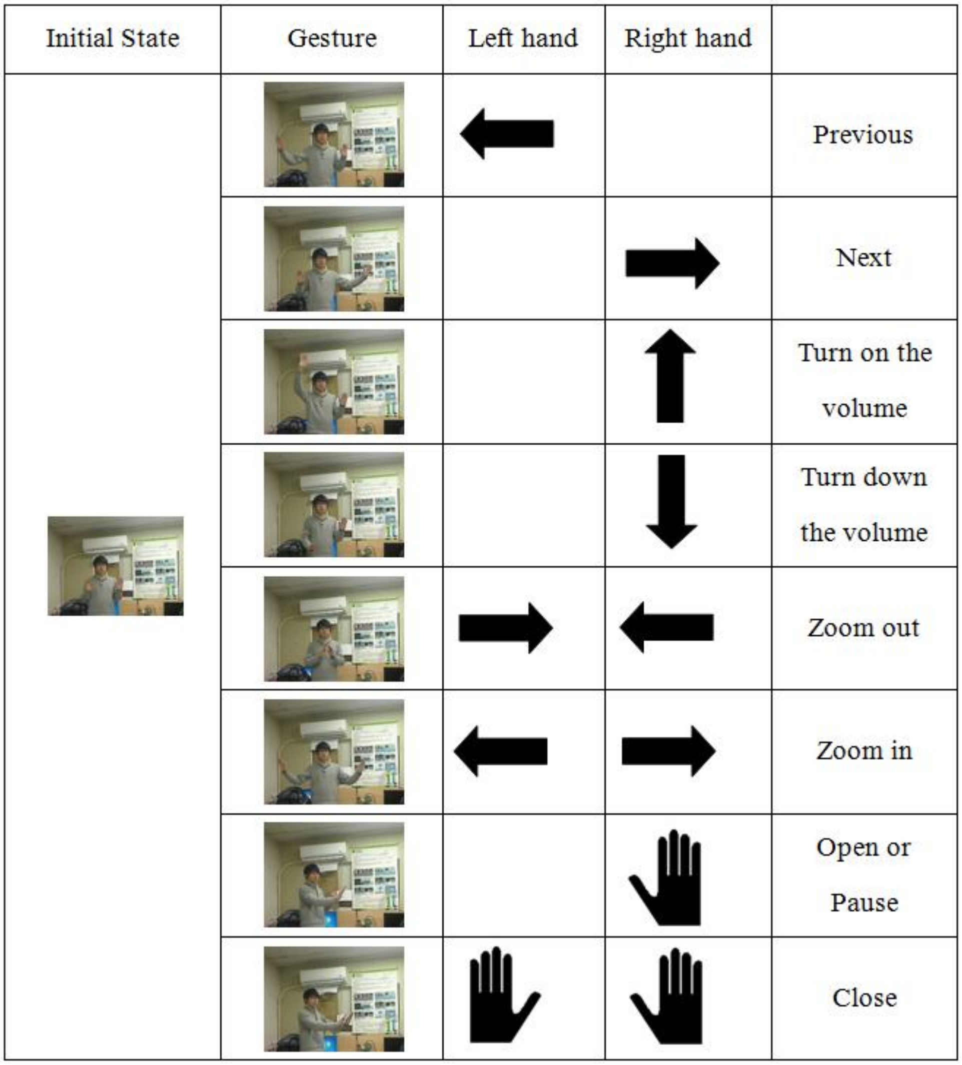


**Figure 6. Six Basic Gestures we Defined**

In this study, based on the method we proposed, we implement an Intelligent Media Controlling System based on computer vision, including MP3 player, Movie player, E-Book. As eight types have been defined, six of which have been demonstrated: upwards, downwards, leftwards, rightwards, zoom in and zoom out. Besides, there are another two types of movement in the Z-axis direction defined in our system: moving towards or away from the camera by one hand and two hands. In the controlling system we presented, we regard these five types gesture as different kinds of meanings for control signal. We defined the upwards gesture as “Turn on the volume”, downwards gesture as “Turn down the volume”, leftwards gesture as “Previous” and rightwards gesture as “Next”, zoom-in as “enlarge the frame” and zoom-out means “Shrink the frame” the type of movement in the Z-axis direction is defined as the signal of “Play Or Pause” and double click is defined as “Cancel or Quit” The gesture actions are shown as Figure 7, and Intelligent Media Controlling System experimental results are shown as Figure 8.



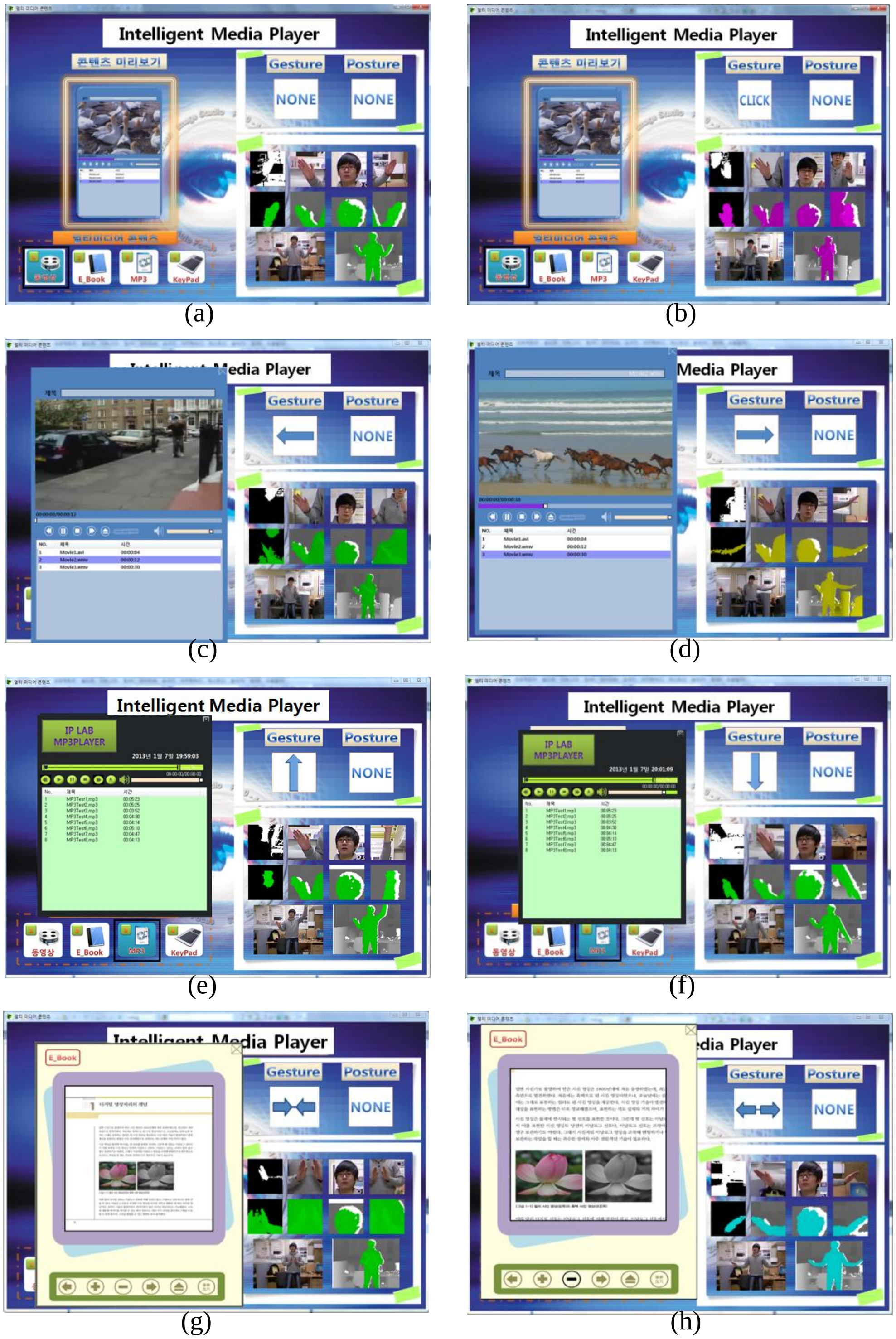
| Initial State | Gesture | Left hand | Right hand | |
|---|---|---|---|---|
| | | ← | | Previous |
| | | | → | Next |
| | | | ↑ | Turn on the volume |
| | | | ↓ | Turn down the volume |
| | | → | ← | Zoom out |
| | | ← | → | Zoom in |
| | | | ✋ | Open or Pause |
| | | ✋ | ✋ | Close |

**Figure 7. Gesture Actions for Controlling Command**

The 8 continuous gestures are recognized in three tests. In the first test, 10 data of each gesture are trained and then the trained 10 data with untrained 10 data are used in test; In the second test, 15 data are trained and then the trained 15 data with the untrained 5 data are tested; finally, 20 data are trained and then the trained 20 data are tested. Table 2 shows us the experimental results.

**Table 2. Gesture Recognition Experiments Results**

| Test | Recognition Rate | | |
|---|---|---|---|
| | Trained Data | New Data | Total |
| 1 | 78/80 | 68/80 | 146/160 |
| 2 | 117/120 | 37/40 | 154/160 |
| 3 | 156/160 | | 156/160 |



(a) (b) (c) (d) (e) (f) (g) (h)

**Figure 8. Gesture Recognition for Controlling System: (a) GUI, (b) "Click" Function, (c) Leftwards for "Previous", (d) Rightwards for "Next", (e) Upwards for "Volume on", (f) Downwards for "Volume down", (g) Zoom out, (h) Zoom in**

## 6. Conclusion

Gesture recognition is an important application area in HCI to communicate with computers. In this paper, we explain a study on natural user interface (NUI) in human gesture recognition using RGB color information and depth information by Kinect. Improved CamShift algorithm combined depth information is proposed for hand tracking, based on depth data, the background which it is similar to the skin color can be distinguished from the front ground. Next, we select orientation feature and velocity feature for feature extraction, an 8-connectivity code is assigned to each of eight possible directions between two points; velocity features are regarded as initial signal and finish signal. Finally, HMM-FNN model is used for gesture recognition, through large numbers of experiments, the experimental results show out its good performance and it has higher stability and accuracy as well.

## Acknowledgements

This research was supported by the Tongmyong University Research Grants 2012 (Small-Scale Research 2012).

## References

[1] A. Valli, “The design of natural interaction”, Multimedia Tools Appl., vol. 38, **(2008)**, pp. 295-305.

[2] H. S. Yoon, J. Soh, Y. J. Bae and H. S. Yang, “Hand gesture recognition using combined features of location, angle and velocity”, Pattern Recognition, vol. 34, **(2001)**, pp. 1491-1501.

[3] N. Liu, B. Lovel and P. Kootsookos, “Evaluation of hmm training algorithms for letter hand gesture recognition”, In proceeding of 3rd IEEE International Symposium on Signal Processing and Information Technology, Darmastadt, Germany, **(2003)**, pp. 648-651.

[4] E. Hunter, J. Schlenzig and R. Jain, “Posture estimation in reduced-model gesture input systems”, International Workshop on Automatic Face and Gesture Recognition, Zurich **(1995)**, pp. 290-295.

[5] X. Deyou, “A Network Approach for Hand Gesture Recognition in Virtual Reality Driving Training System of SPG”, International Conference on Pattern Recognition, Hong Kong, **(2006)**, pp. 519-522.

[6] M. Elmezain, A. Al-Hamadi and B. Michaelis, “Real-Time Capable System for Hand Gesture Recognition Using Hidden Markov Models in Stereo Color Image Sequences”, WSCG Journal, vol. 16, **(2008)**, pp. 65-72.

[7] E. Holden, R. Owens and G. Roy, “Hand Movement Classification Using Adaptive Fuzzy Expert System”, Expert Systems Journal, vol. 9, **(1996)**, pp. 465-480.

[8] L. R. Rabiner, “A tutorial on hidden Markov models and selected applications in speech recognition”, Proc. IEEE, vol. 77, **(2012)**, pp. 257-286.

[9] W. Xu, E. -J. Lee, “Continuous Gesture Recogntion System Using Improved HMM Algorithm Based in 2D and 3D”, International Journal of Multimedia and Ubiquitous Engineering, vol. 7, **(2012)**, pp. 335-340.

[10] W. Xu and E. -J. Lee, “Continuous Gesture Trajectory Recognition System Based on Computer Vision”, Appl. Math. Inf. Sci., vol. 6, **(2012)**, pp. 339-346.

[11] J. L. Nespoulous and A. R. Lecours, “Gesture: nature and function”, In The Biological Foundations of Gestures: Motor and Semiotic Aspects, Lawrence Erlbaum Assoc., **(1986)**, pp. 49-62.

[12] P. Garg, N. Aggarwal and S. Sofat, “Vision Based Hand Gesture Recognition”, World Academy of Science, Engineering and Technology, vol. 49, **(2009)**.

[13] G. Berry, “Small-wall, A Multimodal Human Computer Intelligent Interaction Test Bed with Applications”, Dept. of ECE, University of Illinois at Urbana-Champaign, MS thesis, **(1998)**.

[14] B. Stenger P. R. S. Mendonca and R. Cipolla, “Model-Based 3D Tracking of an Articulated Hand”, In Proceedings of British Machine Vision Conference, Manchester, UK, September, vol. 1, **(2001)**, pp. 63-72.

[15] C. C. Wang and K. C. Wang, “Hand Posture recognition using Adaboost with SIFT for human robot interaction”, Springer Berlin, ISSN 0170-8643, vol. 370, **(2008)**.

[16] A. L. C. Barczak and F. Dadgostar, “Real-time hand tracking using a set of co-operative classifiers based on Haar-like features”, Res. Lett. Inf. Math. Sci., vol. 7, **(2005)**, pp. 29-42.

[17] C. S. Bourennane and L. Martin, “Comparison of Fourier descriptors and Hu moments for hand posture recognition”, In Proceedings of European Signal Processing Conference (EUSIPCO), **(2007)**.

[18] J. S. Jeong, C. Park and K. H. Yoo, “Hand Gesture User Interface for Transforming Objects in 3D Virtual Space”, Communication in Computer and Information Science, vol. 262, **(2011)**, pp. 172-178.

# Authors

### Wenkai Xu

Wenkai Xu received his B. S. at Dalian Polytechnic University in China (2006-2010) and Master degree at Tongmyong University in Korea (2010-2012). Currently, he is studying in Department of Information and Communications Engineering Tongmyong University, Korea for doctor degree. His main research areas are image processing, computer vision, biometrics and pattern recognition.

### Eung-Joo Lee

Eung-Joo Lee received his B. S., M. S. and Ph. D. in Electronic Engineering from Kyungpook National University, Korea, in 1990, 1992, and Aug. 1996, respectively. Since 1997 he has been with the Department of Information & Communications Engineering, Tongmyong University, Korea, where he is currently a professor. From 2000 to July 2002, he was a president of Digital Net Bank Inc. From 2005 to July 2006, he was a visiting professor in the Department of Computer and Information Engineering, Dalian Polytechnic University, China. His main research interests include biometrics, image processing, and computer vision.