UDC 004.946, DOI: 10.2298/CSIS1001177L

An Accelerometer-Based Gesture Recognition Algorithm and its Application for 3D Interaction

Jianfeng Liu[1], Zhigeng Pan[1], and Xiangcheng Li[2]

[1] State Key Lab of CAD&CD, Zhejiang University,
310027 Hangzhou, China
{liujianfeng, zgpan}@cad.zju.edu.cn
2 Sport System Simulation Lab, China Institute of Sports Scinence,
100061 Beijing, China
lxc@3s.org.cn

Abstract. This paper proposes an accelerometer-based gesture recognition algorithm. As a pre-process procedure, raw data output by accelerometer should be quantized, and then use discrete Hidden Markov Model to train and recognize them. Based upon this recognition algorithm, we treat gesture as a method of human-computer interaction and use it in 3D interaction subsystem in VR system named VDOM by following steps: establish Gesture-Semantic Map, train standard gestures, finally do recognition. Experimental results show that the system can recognize input gestures quickly with a reliable recognition rate. The users are able to perform most of the typical interaction tasks in virtual environment by this accelerometer-based device.

Keywords: Tri-axes Accelerometer, HMM, Gesture Recognition, 3-D Interaction, Virtual Reality.

1. Introduction

The VR (virtual reality) system has the widespread application in domains as CAD, E-learning, sports simulation, digital entertainment. There are instances of VR system such as VNM [1] (Virtual Network Marathon) and VBL (Virtual Biological Laboratory) developed by State Key Lab. of CAD&CG in Zhejiang University, Virtual Bicycle Training System designed by Sport System Simulation Lab of CISS. The main task of VR system including two aspects: first, organize and manage the virtual scene effectively, and render or describe the virtual objects as clearly as possible; secondly, complete the interaction task between the user and the virtual objects accurately [2].

Benefit from the rapid development of 3D realistic graphics in recent decades, the present VR systems are able to render large-scale virtual scene easily, and give the users strong immersion. Another primary task of VR system is 3D interaction in virtual environment. D. Bowman [2], [3] et al. defined “3D interaction” as: “Human–computer interaction in which the user’s

tasks are performed directly in a 3D spatial context." Different with traditional WIMP style interaction in 2D program, 3D interaction requires a richer interaction techniques and multi-modal user interface to complete various complicated interaction tasks in VR system which always cope with huge media. In order to enrich 3D interaction techniques in VR system, we presented an accelerometer-base interaction technology and applied it in an instance of VR system as a principal interaction device.

As a novel interaction method, gesture recognition has been researched by many researchers. Hofmann, F. et al. [4] presented the velocity profile based method which utilize HMM to recognize human gestures. Portillo-Rodriguez [5] et al. proposed a FSM based recognition method. Jani Mantyjarvi et al. treated the gesture recognition technology as an interactive method in a design environment [6], and developed a system named Smart Design Studio. Based on those previous works, Thomas Schlomer et al. [7] implemented the accelerometer-based gesture recognition algorithm and utilized the low-cost Wii controller as an interaction device in their experiments.

2. Accelerometer based gesture recognition

We utilize a sensor which integrated a tri-axes accelerometer chip as a hand held input device in our interaction system. When the human performs a gesture, the sensor will collect the data flow output by accelerometer chip, and send it to PC via wireless protocol. We consider this raw data stream fetched from sensor as an "input pattern".

Definition 1. Patten $P = \{V_t \mid 0 \le t \le T\}$, where V_t is the acceleration vector output by interaction device at time *t*, *T* is the time when data stream terminates.

According to the daily experience, the patterns generated by the movement of hand when human performing the same gestures satisfy certain statistical rules to some extent, based on it we propose the "standard pattern". The "standard pattern" is a class of pre-defined patterns, each one corresponding to a special "input semantics". When user performed a gesture, the sensor will send the "input pattern" to interaction system, then system will find out the most approximate "standard pattern", this also can be regarded as a procedure of recognition, and finally the interaction system get an input semantic according to the recognition result, system will interpret (execute) it and return a feedback information to the user.

2.1. Establish patterns and Preprocess

The accelerometer chip BOSCH SMB380 is selected in our system which has a high sampling frequency and sensitivity. The raw data output by the chip is

noisy, redundant, and approximately continuous. It is too complex to process them directly for our system. Here, a denoising procedure is applied.

When denoising, all elements V_t in input pattern P will be processed by a smoothing function like that:

$$S(V_t) = \begin{cases} V_t & t = 0 \\ S(V_{t-1}) + \alpha\left(V_t - S(V_{t-1})\right) & 0 < t \le T \end{cases} \quad (1)$$

α is a smoothing factor which in the range from 0 to 1.

A quantification process is necessary, because we use the discrete HMM-based approach. It requires that the quantified results should simply and countable, in other words it requires a codebook its size is fixed empirically. Since the quantification targets are 3 dimension vectors, and the correlation between them can be measured by the Euclidean distance. Partitional clustering method is suitable. We choose k-medoids clustering rather than k-meaning clustering, because it is more robust to noise and outliers as compared to k-means [8].

Consider that gestures in our system are performed in 3D space, and the output data is 3D vectors which imply the direction and velocity of the hand's motion, the cluster centers should be in 3D space, rather than 2D circle in most gesture recognition based on vision method [9]. Thomas Schlomer et al. [7] distribute the cluster centers in 3d sphere. In this paper we rely on this idea, and identified k (the number of cluster centers) = 14. Then perform clustering algorithm [8]:

Before clustering, for every input pattern P, calculate the maximum and minimum normal of its element vectors. Thus, the radius of the initial clustering sphere can be identified as (max normal + min normal)/2. It will make clustering quickly.

1. First, distribute k initial medoids uniformly in a 3D sphere. The initial medoids set $M_{ini} = \{m_1, m_2, \ldots, m_K\}$ can be identified according to the $R_{cluster}$.
2. Associate each vector in the input pattern to most similar medoid. The similarity here is defined using distance measure that is Euclidean distance.
3. Randomly select nonmedoid object O′.
4. Compute total cost *S* of swapping initial medoid object to O′.
5. If *S*<0, then swap initial medoid with the new one (if *S*<0 then there will be new set of medoids).
6. Repeat steps 2 to 5 until there is no change in the medoid.

After clustering, each candidate vector is associated to an indexed cluster. Quantification is on the basis of clustering results, the size of the codebook obviously equals to the number of clusters. For a vector (component in a pattern), identify its quantification result as the cluster index which it belongs to.

In summary, for a given "input pattern" P, after the steps of denoising and clustering, a desired pattern $P' = \left\{ ID_{V_t} \mid V_t \in P \right\}$ which is suitable for following steps can be established.

2.2. Patten classification

Statistical pattern recognition is based on statistical characterizations of patterns, assuming that the patterns are generated by a probabilistic system. A wide range of algorithms can be applied for pattern recognition, such as Support Vector Machine, Neural Network, and Hidden Markov Model. Since our pattern is composed of a set of discrete and simply data, we select one dimension and discrete HMM in our approach.

A HMM can be formulized as $\lambda = \{N, M, \pi, A, B\}$ [10]. *N* is the state set, *M* is the observation value set, π is the probability vector of initial states, *A* is the matrix for the state transition probability distributions, and *B* is the matrix for the observation symbol probability distributions.

Training

Definition 2. Gesture *G* is denoted as a two-tuple *G* = < *P*, *Att* >, where *P* is input pattern when perform a gesture, and *Att* is a set of additional information, such as input data provided by any other device at the same time.

In this paper, the item "standard gesture" means a class of gestures which correspond to a special "standard pattern". Every "standard gesture" is mapped to an interaction semantic. When system receives an input gesture, the system should give a "standard gesture" most approximate to input gesture.

For each "standard gesture", our participants perform it repeatedly. So finally we get a set of training data $T = < G_1, G_2, \ldots, G_n >$ for every "standard gesture".

Training approach as follows:

1. Establish a HMM for a "standard gesture" and initialize it.
2. Compose the multi-dimensional training data set $\{ O_1, O_2, \ldots, O_L \}$, where O_i is the first component *P* in G_i.
3. Consider that each elements in *T* are independent events, $P(O \mid \lambda)$ can be expressed like that $P(O \mid \lambda) = \prod_{l=1}^{L} P(O_l \mid \lambda)$ where *L* is the length of *T*. Reestimate the factors $\{A, B, \pi\}$ of HMM by Baum-Welch algorithm [10], and get a new HMM denoted as λ'.
4. If $P(O \mid \lambda') - P(O \mid \lambda) > \varepsilon, \lambda = \lambda'$, then repeat step 3. If $P(O \mid \lambda') - P(O \mid \lambda) \le \varepsilon$, λ' can be regard as the local optimal solution.

Finally, a set of optimized HMMs correspond to "standard gesture" can be obtained.

Recognition

The Recognition process might regard as gains an arbitrary input gesture (pattern), and find the optimal matching gesture (pattern) in the "standard gesture" database.

Formal description of the recognition process is as follows:

1. For a given input gesture G_x = < *P, Att* >, *P* was treated as an observation value sequence *O.*
2. Evaluate $\sigma_n = \arg\max P(O \mid \lambda_i)$, λ_i is one of the trained HMMs corresponding to "stand gestures".
3. Thus, the λ_i that give the maximum evaluation of σ_n can be considered as the result. And the input gesture was recognized.

The recognition system outputs a "standard gesture" in the database which the result HMM corresponding to.

Evaluation with a threshold

Think about that, for any input gestures, the above algorithm will give a solution. But it cannot guarantee that it is a optimal and harmless solution, since when user performed a misoperation and the system will recognize it and do some work that violate the user's original intention. To avoid it, a threshold should be set when evaluating the maximum probabilities. It can be regarded as a misoperation and ignore the input when the maximum expectation we have got in "recognition" step is not greater than the threshold empirical identified.

3. 3D Interaction System in Virtual Environment

3D human-computer interaction in virtual environment typically includes three main tasks [11]: virtual objects manipulation, viewpoint manipulation, application control. Viewpoint manipulation including the movement of viewpoint (et. Camera) and the control of other parameters such as FOV and Zoom [12]. The term application control means the communication between user and system which is not part of virtual environment.

We propose a 3D interaction model in virtual environment. Its goal is to complete above tasks effectively.

Before that, some concepts should be introduced:

Definition 3. Interaction Atom was denoted as A, means an atomic interaction operation. A* stands for the Interaction Atom set which should be determined early in the design of interaction system.

Definition 4. Execution Routine $E = \{A_0, A_1, ...A_t\}$, A_i is an interaction atom, and the entire routine was composed by a sequence of atomic interaction operations to implement a special interaction task.

Definition 5. Interaction Semantic is a four tuples as $s =< I, C, E, con >$, where *I* if the input data flow, *C* is the current context and *E* is the target execution routine and will be executed only if the condition *con* is *true*.

The architecture of the model is illustrated in Fig.1.

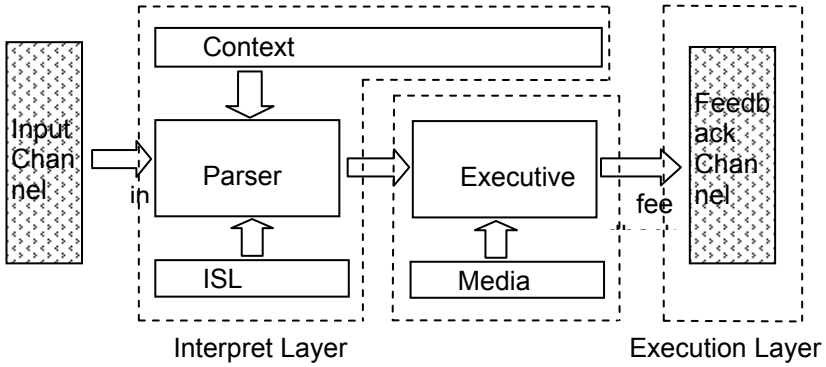

Fig. 1. The architecture of interaction model.

3.1. Interaction Semantics Library

The term “Interaction Semantic” refers to the “meaning” of the input data flow in particular context. The phrase “meaning” can be considered as a sequence of interaction commands and their parameters. Once an input event occurs, the Parser will interpret the input data flow, and extract the underlying Interaction Semantics in it by referring to current global context. The Executive receives Interaction Semantics from Parser, translates the semantics to Execution Routines, and completes the interaction tasks by executing the routines.

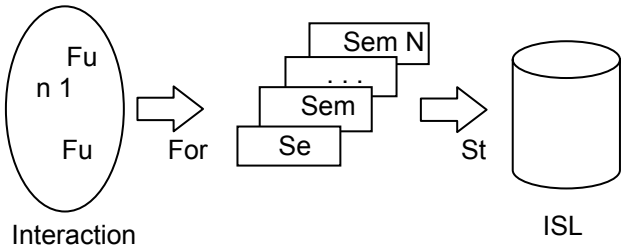

Fig. 2. creating interaction semantic library.

An ISL (Interaction Semantic Library) must be established when designing interaction system. Enumerates all interactive functions, translates them into IS, and eventually stored the ISs into ISL. The process was illustrated in Fig.2.

3.2. Interaction Feedback

The interaction result feedback is an important part of interaction process. A good interactive system needs clear and complete feedback. The feedback massage must contain result and final state of the task, the change of global context caused by the last interaction, the reasons when task failure, as well as some proposal of alternative plan.

4. Applications and Analysis

We applied 3D interaction method based on the gesture recognition algorithm which introduced above in the VR system –Virtual Digital Olympic Museum (VDOM).The 3D Interaction subsystem was demonstrated in Fig.3.

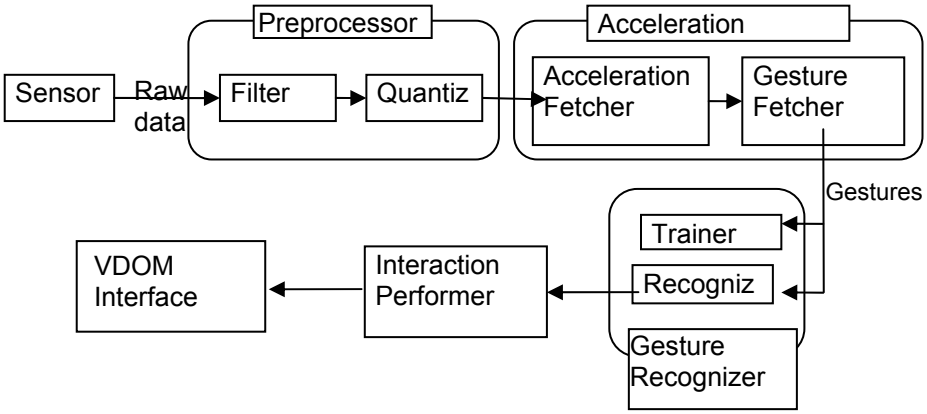

Fig. 3. 3D-Interaction Subsystem in VDOM.

4.1. Gesture-Semantic Mapping

In VDOM, interaction system provides interaction functions as follows:

<1>Agent control: turn left/right

<2>Object manipulation: selection/rotation/scaling/movement

For the functions, ISL was established. In the system, ISL was saved as XML file and named *.isl.xml. It is facile to deal with by script language.

Table.1 gives the mapping from “standard gesture” to “interaction semantic” where gestures (A)-(H) was illustrated in Fig.4.

Table 1. Gesture-Semantic Table

Gesture	Interaction Semantic
A	move left/turn left
B	move right/turn right
C	rotate left
D	rotate right
E	move up
F	move down
G	confirm/select/zoom in
H	cancel/delete/zoom out

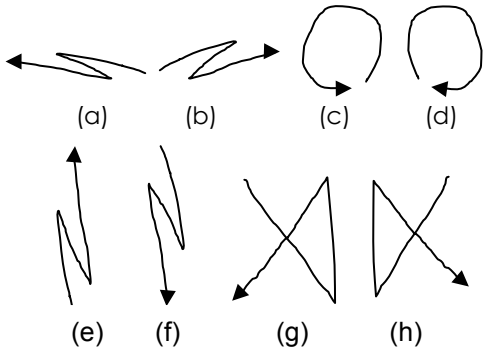

Fig. 4. Gestures.

4.2. Training

Table.2 shows the counter of training samples for each gesture, for gesture g (confirm) and h (cancel) are more complex than other gestures in Fig.3, more samples is helpful. After trained by the samples, we gain 8 HMMs corresponding to gestures A-H.

The training performance can be measured by $P(G \mid \lambda)$ which stands for the probability of HMM λ produce a observation sequence *G* . Fig.4 shows ln P(G| λ) with a varying number of HMM states. Training performance of each gesture was enhanced while increase the HMM State count from 8 to 12.While increase HMM count from 12 to 16, we gain an improvement in performance for gesture a,b and d, however we also got a loss in performance for gesture c,f,e. The count was identified as 12 by considering with both training effect and speed.

Table 2. Training Data Table

Interaction Commands	Training Data
move / turn left	150
move / turn right	150
rotate left	150
rotate right	150
move up	150
move down	150
Confirm	200
Cancel	200

Table 3. Recognition Results

Interaction Commands	Testing Data	Correct	Recognition Ratio
move / turn left	50	47	0.94
move / turn right	50	46	0.92
rotate left	50	49	0.98
rotate right	50	47	0.94
move up	50	48	0.96
move down	50	49	0.99
confirm	50	44	0.88
cancel	50	45	0.90

4.3. Results

After completing the training work, test each gesture with 50 samples separately. And the result was shown in Table.3. For gesture g and h, we got a recognition rate slightly less than 90 percents on average, and for the remaining we got a rate more than 95 percents.

As a whole, the 3D interaction technology based-on accelerometer was applied in VDOM successfully, and the system shown a reliable result. Users can use the accelerometer-based device barrier-free and perform the interaction with VDOM easily. Fig.6 contains the screenshots of VDOM. In the left side, the user is performing gestures which defined in Fig.4 with our accelerometer-based device, and the interaction results are illustrated in the right side.

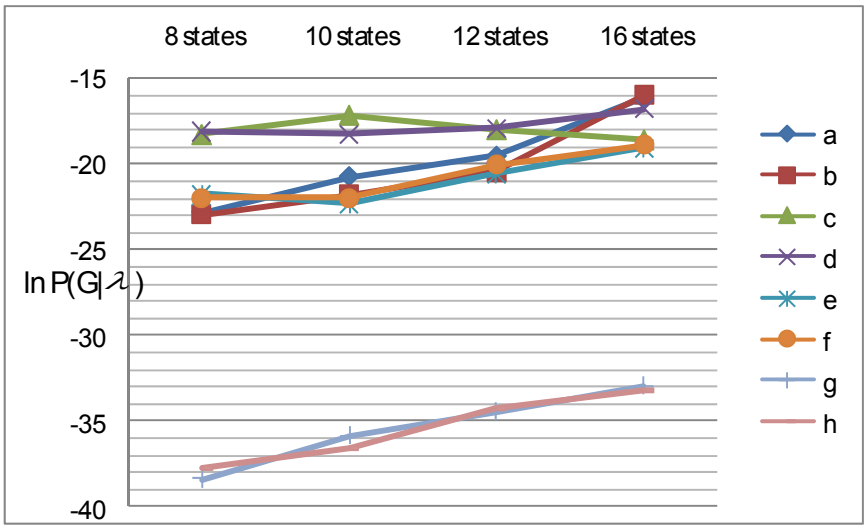

Fig. 5. ln P(G| λ) with a varying number of HMM states.

5. Conclusion

We have presented an accelerometer-based human-computer interaction method and its application in VR system. We also described the implementation of interaction subsystem. A Gesture-Semantic Table contains all predefined standard gestures and their corresponding semantics. Semantics and their respective execute routines are stored in ISL. Based on Gesture-Semantic Table and ISL the interaction subsystem can identify standard gestures in input data stream ,and translate them into interaction semantics and perform the interaction tasks finally. In our interaction subsystem, there are several reasons may cause a lower recognition rate: first, a deficient training process causes weakly correlation between HMM and the “standard gestures”; secondly, for complicated gestures, it may lose parts of information during quantification process; thirdly, the lack of samples from various users. In practical application, these following ways can improve the recognition rate: increase the sample count, involving various participants; increase the size of the codebook appropriately, in other words, increase the count of cluster centers in the clustering step, it may reduce the loss during quantification; for complicated gestures, increase the number of HMM state. The future work is to extend current gesture library and ISL, and to provide more interaction functions in 3D virtual world.

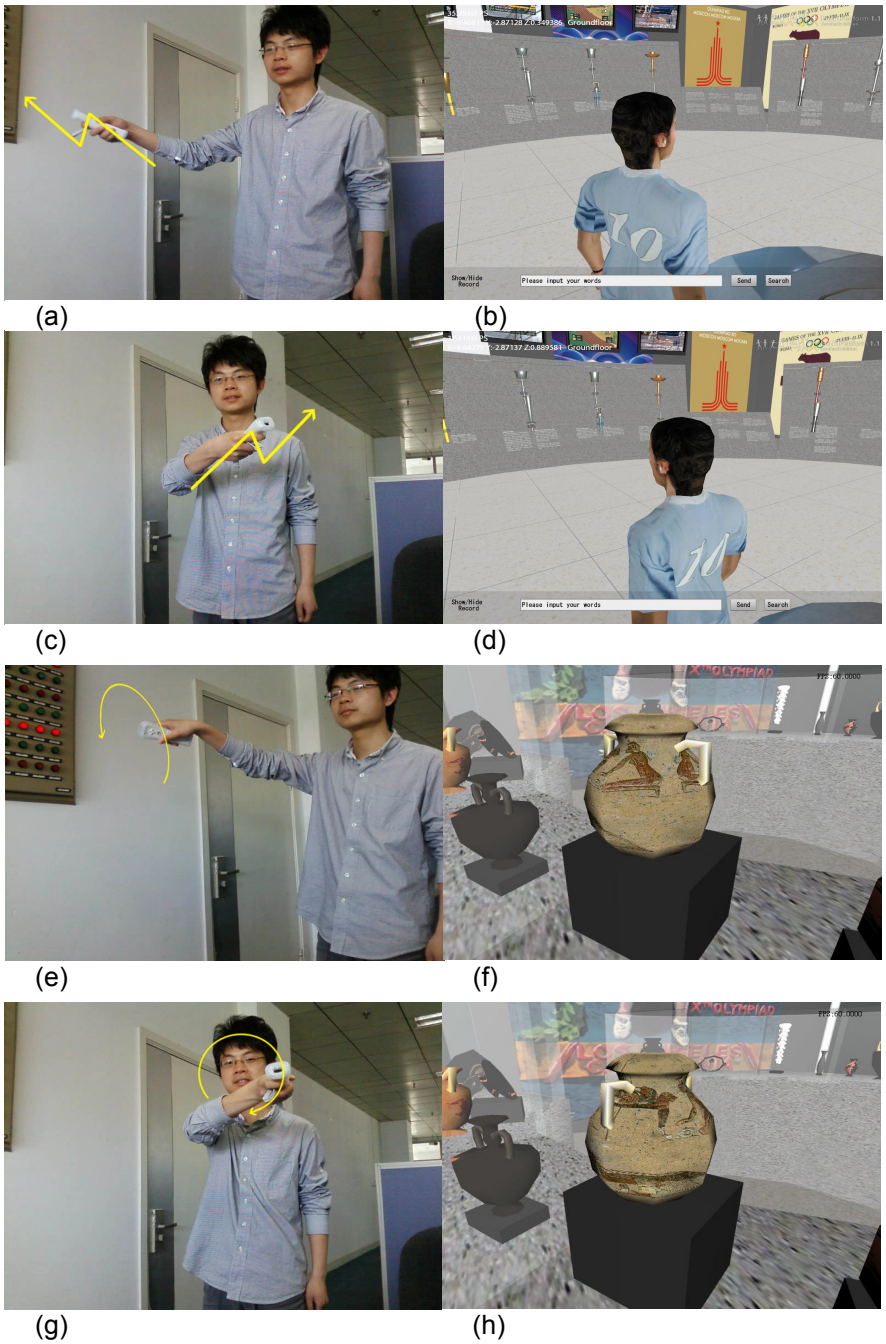

Fig. 6. Runtime Results.

6. References

1. He GQ, Pan ZG, Li YQ, Su SY: Synthesized Synchronicity Mechanism for Fitness-Oriented Virtual Network Game. Journal of Computer-Aided Design & Computer Graphics, Vol.12, No.2, 73-78. (2008)
2. Bowman, D.A., Kruijff, E., LaViola, J., Poupyrev, I.: 3D User Interfaces: Theory and Practice. Addison-Wesley, Boston, USA. (2004)
3. Bowman, D.A., Coquillart, S., Froehlich, B.: New Directions in 3D User Interfaces. Computer Graphics and Applications, IEEE, Vol.28, No.6, 20-36. (2008)
4. Hofmann, F., Heyer, P., Hommel, G.: Velocity Profile Based Recognition of Dynamic Gestures with Discrete Hidden Markov Models. International Gesture Workshop on Gesture and Sign Language in Human-Computer Interaction, Springer, London, UK, 81–95. (2004)
5. Portillo-Rodriguez, Sandoval-Gonzalez, O.: Development of a 3D real time gesture recognition methodology for virtual environment control.The 17th IEEE International Symposium on Robot and Human Interactive Communication, 279-284. (2008)
6. Jani Mantyjarvi, Juha Kela, Panu Korpipaa.: Accelerometer-based gesture control for a design environment. Personal and Ubiquitous Computing, Vol.10, No.5, 285-299. (2006)
7. Thomas Schlomer, Benjamin Poppinga, Niels Henze, Susanne Boll.: Gesture recognition with a Wii controller. Proceedings of the 2nd international conference on Tangible and embedded interaction, 11-14. (2008)
8. Ng, R.T., Han, J.: Efficient and effective clustering methods for spatial data mining. Proceedings of the 20th VLDB Conference, 144-155. (1994)
9. Hyeon-Kyu Lee, Jin H.Kim.: An HMM-based threshold model approach for gesture recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.21, No.10, 961-973. (1999)
10. Rabiner, L.R.: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proc. IEEE, Vol. 77, 257-285. (1989)
11. Chris, H.: A Survey of 3D Interaction Techniques. Computer Graphic Forum, Vol.16, 269-281. (1997)
12. LaViola, J., Acevedo, D..: Hands-Free Multi-Scale Navigation in Virtual Environments. Proceedings of ACM Symposium on Interactive 3D Graphics, North Carolina, USA, 9-15. (2001)

Jianfeng Liu was born in 1985. He is a M.S. candidate at the College of Computer Science and Technology, Zhejiang University. His current research interests include virtual reality and human-computer interaction.

Zhigeng Pan was born in 1965. He is a professor and doctoral supervisor at the College of Computer Science and Technology, Zhejiang University. His researches areas are virtual reality, argument reality.

Xiangcheng Li was born in 1974. He is currently the Director of Sport System Simulation Lab of CISS. His researches areas are virtual reality, sport simulation.

Received: April 28, 2009; Accepted: July 15, 2009.