The Control of Avatar Motion Using Hand Gesture

ChanSu Lee, SangWon Ghyme, ChanJong Park
Human Computing Dept. VR Team
Electronics and Telecommunications Research Institute
305-350, 161 Kajang-dong, Yusong-gu, Taejon, KOREA
+82-42-860-5319
{chanse, ghyme, cjpark}@seri.re.kr

KwangYun Wohn
Dept. of Computer Science
Korea Advanced Institute of Science and Technologies
305-701, Kusong-dong, Yusong-gu, Taejon, KOREA
wohn@cs.kaist.ac.kr

Abstract

It is difficult to navigate virtual environment as in real world and to interact with other participant in virtual environment, especially wearing Head-Mounted Display (HMD). We developed Virtual Office Environment System (VOES), and avatar is used to navigate and to interact with other participants. For easy and intuitive control of avatar motion in the system, we use continuous hand gesture recognition system. State automata are proposed in hand gesture recognition to segment continuous hand gesture and to remove meaningless motion. Using avatar and gesture interface, this system provides natural navigation and interaction in virtual environment system.

Keywords: Hand gesture recognition, avatar, gesture interface, immersion system

1. INTRODUCTION

There are many attempts to develop realistic and easy interface in virtual environment. Speech recognition and force feedback is also attempted for easy interaction in virtual environment [12]. Human uses hands in manipulation of object and the use of hand gesture is often observed in everyday human communications. So one of the most effective and intuitive method to interact with virtual environment as in the real world is using hand [6]. Most previous researches to utilize hand gesture in 3D virtual world, however, have dealt with direct manipulation of 3D objects as the extension of 2D direct manipulation [7].

Recently there are many attempts to recognize hand gesture [16-18]. But gesture command recognition systems still are not used often in real application system for the difficulties in recognition of hand gesture. One of the most difficult problems in continuous hand gesture recognition is to find starting and ending point in continuous gestures, and segment continuous gesture into individual ones. To solve this segmentation problem, Hidden Markov Model [17-18], feature-based gesture analysis [1] and Artificial Neural Networks [14] are used. Still it is difficult to distinguish meaning hand gesture from meaningless simple movement. Glove devices are frequently used in immersive system. But for the lacks of notification which commands are valid it is difficulty to use glove device as input command generator. This paper attempt to solve the problem of distinguishing valid gesture from meaningless one using partition of motion phase and state automata for hand gesture.

Human wants to participate and represent himself in virtual environment, and want to communicate with other people. Avatar, which is computer graphic character, is used to represent participant in virtual environment. Many attempts are trying to effective generation of avatar motion [11] and to real-time control of avatar using tracker sensor [9][13].

We have developed a virtual environment system, Virtual Office Environment System (VOES). In this virtual environment, avatar is used to navigate and to interact with other participant [4] and to cooperate with other office workers in cyberspace. On wearing HMD, it is difficult to control avatar motion. So hand gesture interface system for the control of avatar motion is developed. This gesture interface system is useful, as it is able to distinguish meaning gesture command from meaningless movement.

This paper is organized as follows. The next section gives description for the VOES and motion engine, which generate realistic avatar motion. Section 3 shows overall hand gesture interface for the control of avatar motions. In the section, we define hand gesture and basic elements, and state automata for continuous hand gesture segmentation and removal of meaningless movement are explained. Next, basic element recognition and interpretation of motion and control of avatar is explained. In section 4, experiment results are shown. Finally, we give the summary of this paper and further works.

2. VIRTUAL OFFICE ENVIRONMENT SYSTEM (VOES)

2.1 Overview of VOES

We have developed the VOES, where user can do his work as in real office. The VOES is the virtual environment system using avatar [4] to navigate around a virtual office and to do interactions with other user's avatar as human does. In this system, we can find other participants' presence and activity, and generate motions and communicate with other participants. Some primitive motions are prepared in a motion DB to perform the avatar activities in the VOES.

The VOES has client-server architecture. A server records movement and location of avatar and informs existence of avatar to clients. And it also manages virtual environment. A client has 3 modules as shown in figure 1. Interface module receives user's input from mouse or keyboard or it gets input from glove and tracker. From user's input, interface module generates commands to control avatar motion. Gesture recognizer gets data for figure angle and hand position, and analyzes hand gesture and generates the meaning of gesture. Event handler receives event from mouse or keyboard and transfers event to command interpreter. Command interpreter receives commands from gesture recognizer or from event, and translates them into proper motion commands according to the environment. A motion engine generates motions requested by command interpreter. Browser is used to access server and communicate or interact with other user, and gives information of the environment to generate command.

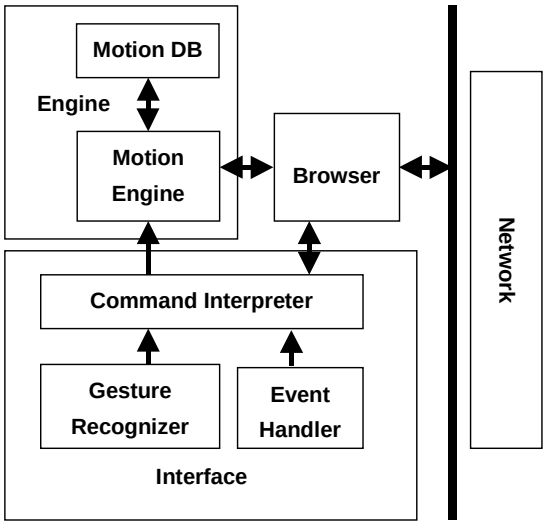


Figure 1: The Client of VOES

2.2 Motion Engine

For generation of realistic motion in avatar that has a skeletal structure, the metatree is used [6]. In the metatree, a center of motion isn't fixed as a hierarchical structure. Therefore, all avatars have a center of motion that may be the center of mass or another as like a heap or an ankle.

The motion engine is based on the metatree, and composed of sensor, motion flow controller and motion record processor. The motion record processor actually generates a motion by processing motion records in a motion DB.

2.3 Motion DB

The motion DB has the set of motion records needed for generating primitive motions. From kinematics analysis or motion capture data, these motion records can be generated. The primitive motions are used for navigation and interaction. For a complex motion, they can be combined with each other. In the motion DB of the VOES, 10 primitive motions are supported. Figure 2 shows 10 primitive motions. Among them, 'walk', 'side walk', 'jump', 'sit', 'turn' and 'view change' are motions for navigation. And 'bow', 'wave hand', 'bye', 'agree' and 'deny' are for interaction with other avatar.

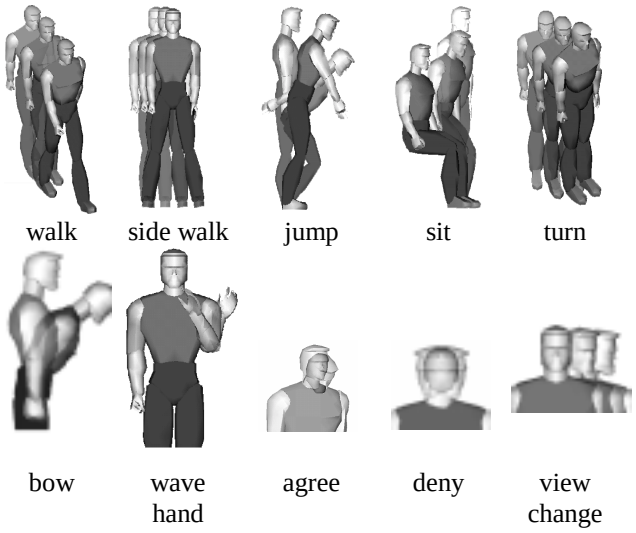


Figure 2: 10 primitive motions in VOES

3. HAND GESTURE INTERFACE SYSTEM

3.1 Definition of hand gesture.

Hand gesture is defined to control 10 primitive motion of avatar. To control avatar motions, we use posture attribute and direction attributes of hand gesture. Different postures mean different motions of avatar. Figure 3 shows the defined postures for avatar motion control. These postures are defined by modifying posture of Korean Sign Language (KSL). Direction of each gesture announces which direction the avatar moves. "Walk" and "side walk" use same posture for the similarity of meaning except direction. "Change view" is used to control camera viewpoint of the system. Camera viewpoint is changed to upper view, side view or avatar eye view according to the movement direction for

this posture. In figure 4, basic direction elements show defined 7 basic direction elements.

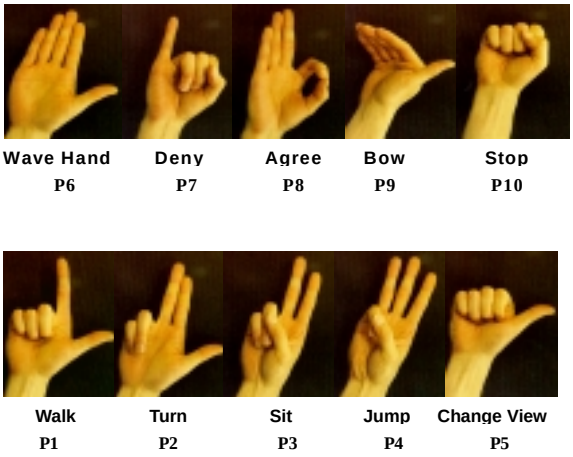


Figure 3: Basic posture

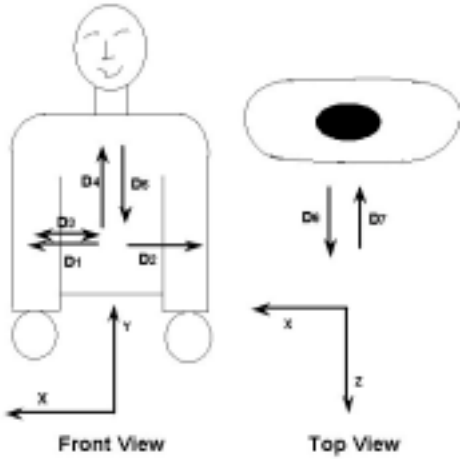


Figure 4: Basic Direction.

3.2 Overview of Gesture Interface System

The configuration of gesture recognizer system for the control of avatar motion is like figure 5. This consists of data acquisition stage, state estimation stage, meaning interpretation stage of gesture.

At the first stage, this system get angle data of each finger from CyberGlove™, and get position data from Polhemus Fastrak™ for one hand. At the second stage, this system estimates motion state by state automata that was done using speed and change of speed in motion. In this stage, continuous gestures are segmented into individual ones. And unintentional gestures are removed. At the third stage, each individual gesture's attributes are recognized using direction classifier and posture classifier and recognize the gesture meaning by interpreter. At last, using recognition result, this system generates command for avatar motion control.

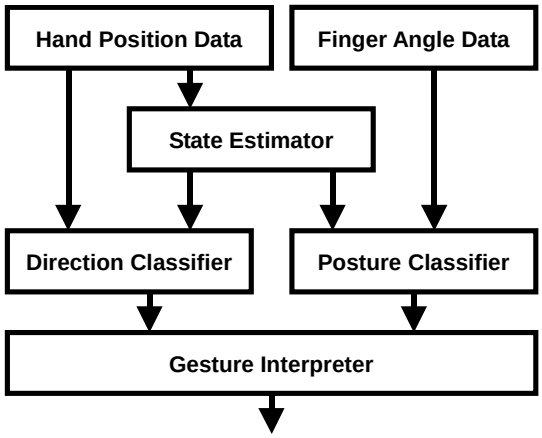


Figure 5: Configuration for Hand Gesture Recognizer

3.3 Segmentation of Continuous Gesture

Human can easily distinguish intentional hand gesture from simple meaningless movement. It is not fully investigated yet, however, how a machine can distinguish them automatically. This problem is finding the intentional gesture segment from the continuous arm movement.

In previous research to separate gesture movements from hand movement with no communicative intent, it was found that three distinct motion phase typically constitute a gesture [7]: preparation, stroke and retraction. Quek extract 6 rules to distinguish meaning gesture from meaningless movement [8]. 1. Movements that comprise a slow initial phase from a rest position proceed with a phase at a rate of speed exceeding some threshold (the stroke), and returns to the resting position are gesture laden. 2. The Configuration of the hand during the stroke is in the form of some recognized symbol. 3. Slow motions from one resting position and resulting in another resting position are not gestures. 4. Hand movements outside some work volume will not be considered as pertinent gestures. 5. The user will be required to hold a static hand gestures for some finite period for them to be recognized. 6. Repetitive movements in the workspace will be deemed gestures to be interpreted.

In previous our research for sign language [9], we analyzed Korean Sign Language (KSL), and found rules to distinguish meaning gesture from meaningless movement. By simplifying them to proper avatar motion control, we found similar rules to distinguish intentional hand gestures. In addition to Quek's rules, intentional gesture ends with distinguishable decrease in motion speed. And short time acceleration movement follows by short time deceleration

movement is not intentional gesture. And in our system, every intentional command gesture has position movement, which may be our system specific restriction.

To distinguish intentional gesture that observes above rules from meaningless motions that do not observes rules, we estimate motion state using motion phase. State automata are five-tuple as in equation 1.

$$(E, X, f, x_0, F) \quad (1)$$

where

E is a finite alphabet: *motion phase*

X is a finite state set: *motion state*

f is a state transition function

x_0 *is an initial state,* $x_0 \in X$

F is a set of final states, $F \subseteq E$

E is input which make translation of state and come from motion phase. Motion phase is partition of motion according to speed and change of speed. Speed and change of speed are calculated at each sampling time by equation 2. Speed is length of movement in any direction per second. Change of speed is difference of current speed related to previous one. Table 1 shows condition for segmentation of motion phase and event, which is used to automata input. Figure 6 shows partition example of motion phase. X is 9 motion states. And we define 10 states to distinguish motion states according to above rules. Table 2 describes 10 states. And transition function f can be represented graphically as in figure 7. Rules for intentional gesture that is acceptable language can be expressed by possible language [10]. And Meaning gesture satisfies possible language and reach q9, final state, in each individual gesture.

Table 1. Motion phase

Phase	Event	Condition	
		speed	change of speed
Stop	0	no, very slow	no , small acceleration or deceleration
Preparation	1	slow	small acceleration or deceleration
Stroke	2	very fast	large acceleration
Moving	3	fast	small acceleration or deceleration
End	4	slow	

$$speed: v(t) = \frac{\sqrt{(x(t)-x(t-1))^2+(y(t)-y(t-1))^2+(z(t)-z(t-1))^2}}{\Delta t} \quad (2)$$

$$change\ of\ speed: \Delta v(t) = v(t)-v(t-1)$$

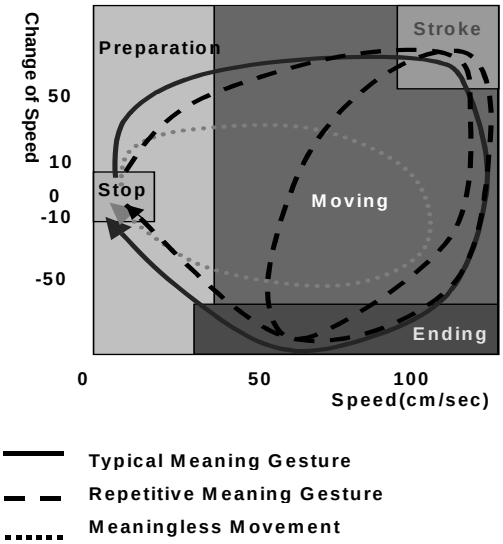


Figure 6: Meaning Gesture and Meaningless one

Table 2. Description of motion phase

State	Description
q0	no movement state
q1	slow movement in initial
q2	Stroke at the beginning
q3	moving motion without phase
q4	moving motion after stroke
q5	stroke motion after moving motion
q6	ending motion with deceleration
q7	repetitive motion
q8	end preparation motion
q9	end of meaning gesture

Possible language which means meaning gesture, can be expressed as regular form

$$q_0^+(q_0^* q_2^+ + q_3^+ q_5^+) q_4^+ q_6^+ (q_7^+ q_8^+ q_6^+)^* q_8^+ q_9 \quad (3)$$

This means that motion state of each gestures start from no movement state and after slow motion or movement, passes

stroke state, and moving motion. After moving state (q4), meaning gesture have end state (q6) with or without repetition of moving state (q7). Motion which reaches q9 state recognized above possible language and means that the gesture is meaning gesture. If motion starts at q0 and cannot reach q9 and end at q0 again, then it is meaningless movement.

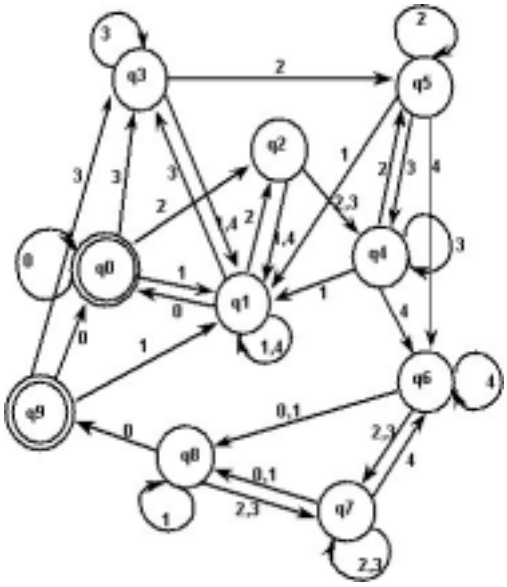
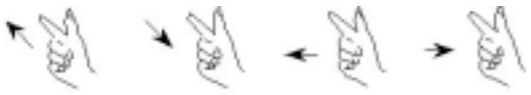


Figure 7: State Transition Diagram

3.4 Recognition of Gesture Meaning and Generation of Command

As shown configuration for hand gesture recognition and avatar control, after state estimator which distinguish meaning gesture from meaningless movement, hand motion classifier and hand post classifier are executed if the gesture is meaning gesture. Direction classification is done using feature extraction and classification based on fuzzy rule [9]. Posture classification is done using Fuzzy Min-Max Neural Networks [9][11]. Used data for posture classifier is 12 normalized angles for finger joints.

From direction classification and posture classification result, command meaning is interpreted. Figure 8 shows example of the walk command gesture interpretation. From posture classification P1, interpreter understands the gesture is for “walk” and sub classification of the meaning is done by classification result of direction. For examples, if direction D6 class is recognized, then the command for the avatar control is “walk forward”. And if direction D7 class is recognized, then “walk backward” command is generated. Using the result of gesture recognizer, command for avatar control is generated. Before generating command, collision detection and possibility of given gesture command is estimated. If possible, commands for avatar motion control are generated.



(a) walk forward (b) walk backward (c) walk left (d) walk right

Figure 8: Example of walk gesture interpretation

4. EXPERIMENTS

At first, we examined the ability to distinguish meaning gesture from meaningless movement. Experiments are done for the intended gesture commands, “walk forward” and “ stop” and “wave hand”, which is done continuously. During the gesture, we do some other motions that are not intended to control avatar.

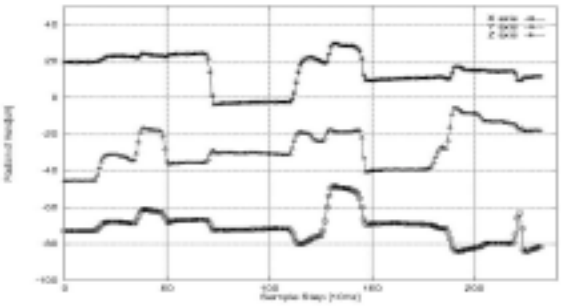


Figure 9: Sensing Position Data

Figure 9 shows sensing data of hand position for each axis. We get data from Polhemus Fastrak with 10Hz sampling rate. Figure 10 shows calculated speed and change of speed at every sampling time. In this figure, we can easily get when we make motion and when does not. But it is difficult to distinguish which is intended movement and which is not. We know about 10 movement is made from the figure. But we cannot distinguish 3 intentional gesture. Figure 11 is drown in phase plane with speed (not velocity) in horizontal axis and with change of speed (not acceleration) in vertical axis for the motion. In this plane, motion phase partition can be done as in figure 6. The partition result of motion phase is figure 12. In figure 12 we can get 11 distinguishable movements which are segmented by no motion phase. Still we have difficulty in distinguishing intentional meaning gesture. Figure 13 shows state transition according to state automata for hand gesture. By motion partition result, state is transferred according to state transition function as in figure 7. We can notice 3 intentional gestures that reach meaning end state q9 in the motion state transition as shown figure 13. In such a way

this system distinguish meaning gesture from continuous motion.

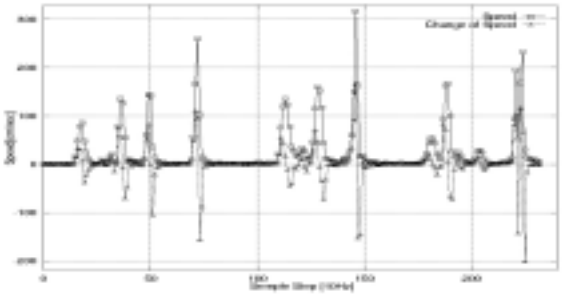


Figure 10: Speed and Change of Speed

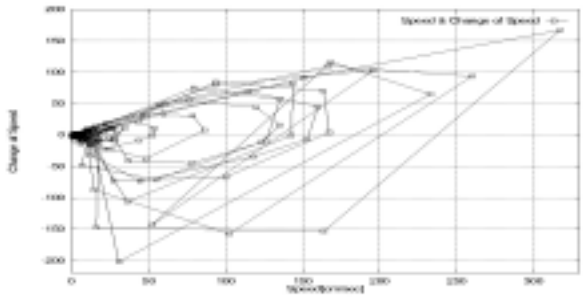


Figure 11: State transition for the gesture

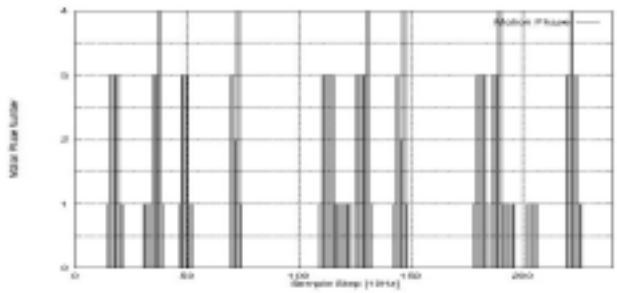


Figure 12: Motion Phase

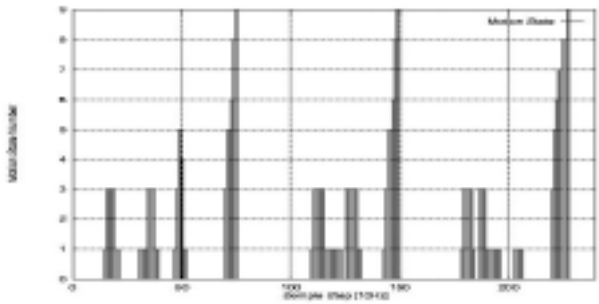


Figure 13: Motion State

Direction classifier and posture classifier recognizes the meanings of gesture. By combination of direction and posture element of gesture, 18 gesture commands are recognized for avatar control. Recognition rate of these hand gestures for 3 different persons is 94.1%. Errors are come from posture classification error mainly. It's because the difference of hand size or skeleton, which make difference in sensing data of each person for posture classification. Another errors are because of direction classification. The third is for the error to distinguish intentional gesture. In some gesture, especially "walk backward", are not well distinguished as intentional gesture. It's for slow down of speed at the end of motion.

5. CONCLUSION AND FUTURE WORK

We developed a virtual environment system, VOES, to develop realistic office in immersion environments. For the realistic navigation and interaction, avatar is used. And for the control of avatar motion easily in immersive system, we developed hand gesture interface system. In developing hand gesture interface system that can distinguish intentional gesture from meaningless movement, we partitioned movement into 5-motion phase according to speed and change of speed of motion. And using state automata for the hand gesture, we distinguish intentional gesture from meaningless movement. Average recognition rate for each command is 94.1%

Wearing HMD, camera-attached to avatars eye gives most realistic feeling. And during walking the change of height of avatar eye is larger than real motion. So it should reduce eye height variation to provide more realistic view. And direct manipulation of object using avatar's hand and arm slaving will be developed for more effective interaction in virtual environment using avatar. And this system can also be developed as gesture communication system.

6. REFERENCES

[1] A. Wexelblat, "Natural Gesture in Virtual Environments," in Proc. of VRST95 Conf., pp. 5-16, 1995.

[2] Adam Kendon, "Current issues in the study of gesture," In J-L Nespoulous, P. Person, & A. R. Lecours, editors, The Biological Foundations of Gestures: Motor and Semiotic Aspects, pp. 23-47, 1986.

[3] Chan-Su Lee at el., "Real-time Recognition System of Korean Sign Language based on Elementary Components," FUZZ-IEEE'97, pp.1463-1468, 1997.

[4] ChanJong Park at el., "The Avatar's Behavior and Interaction for Virile World," Proceedings of the Virtual Reality Society of Japan Second Annual Conference, pp. 242-240, July 1997.

[5] Christos G. Cassandras, "Discrete Event Systems," IRWIN, 1993.

[6] D. J. Sturman, “A Survey of Glove-based Input,” IEEE Computer Graphics & Applications, pp. 30-39, Jan. 1994.

[7] D. J. Sturman, “A Survey of Glove-based Input,” IEEE Computer Graphics & Applications, pp. 30-39, Jan. 1994.

[8] Francis K.H. Quek, “Toward a Vision-Based hand Gesture Interface,” in Proc. of VRST’94, pp.17-34, 1994.

[9] N. I. Badler et al., “Real-Time Control of a Virtual Human Using Minimal Sensors,” Presence, Vol. 3, No. 1, 1993.

[10] P. Simpson, “Fuzzy Min-max Neural Networks –Part 1: Classification,” IEEE Trans. on Neural Networks, Vol. 3, pp. 776-786, Sep. 1992.

[11] R. Boulic et al., “Integration of Motion Control Techniques for Virtual Human and Avatar Real-Time Animation,” in Proc. of ACM VRST’97 Lausanne Switzerland, pp. 111-118, Sep. 1997.

[12] R. Gupta et al., “Experiments Using Multimodal Virtual Environments in Design for Assembly Analysis,” Presence, Vol. 6, No. 3, pp. 318-338, June 1997.

[13] S. K. Semwal, “Mapping Algorithms for Real-Time Control of an Avatar Using Eight Sensors,” Presence, Vol. 7, No. 1, pp. 1-21, Feb. 1998.

[14] S. S. Fels and G. E. Hinton, “Glove-talk: A neural network interface between data-glove and a speech synthesizer,” IEEE Trans. Neural Networks, Vol.4, pp. 2-8, Jan. 1993.

[15] SangWon Ghyme et al., “The Real-Time Motion Generation of Human Avatars,” Proc. of the 13th Symposium on Human Interface, pp. 435-438, Osaka Japan, 1997.

[16] T. S. Hung et al., “Hand Gesture Modeling, Analysis and Synthesis,” in Proc. of Int. Workshop on Automatic Face-and Gesture-Recognition, Swiss Zurich, June, 1995.

[17] T. Starner et al., “Visual Recognition of American Sign Language Using Hidden Markov Models,” Int. Workshop on Automatic Face- and Gesture-Recognition, Swiss Zurich, June 1995.

[18] Yanghee Nam and KwangYun Wohn, “Recognition of Space-Time Hand-Gesture using Hidden Markov Model,” in Proc. of ACM VRST96 Conf., pp. 51-58, July 1996.