Realizing Emotional Autonomous Virtual Agents in a Multi-user Virtual Environment

Po-Lung Chen
Industrial Technology Research Institute
chenpolung@itri.org.tw

Tsai-Yen Li
National Chengchi University
li@nccu.edu.tw

ABSTRACT

Most of the Multi-user Virtual Environment Systems (MVE) today do not allow computer-controlled virtual agents to interact with real users in an emotional manner. In this paper, we aim to design autonomous virtual agents that can generate emotional behaviors through textual dialogs, route selection, and body gestures when interacting with other avatars controlled by real users or by the computer. These behaviors are expressed via interaction events delivered through the network. The interaction events are generated according to the mental state of the agent based on a psychological model including emotion, memory, and personal traits. We have also designed an update mechanism to interpret an interaction event and update the mental state of the virtual agent accordingly. This model has been implemented and tested in an experimental MVE platform based on VRML/X3D to evaluate the effectiveness of the design. With the mechanism of autonomous virtual agent proposed in this paper, we hope to enhance MVE with virtual agents possessing emotional behaviors that can be easily authored by regular users.

1: INTRODUCTION

The concept of intelligent *virtual characters* (or called *virtual agents*[1]) has attracted much attention recently. One of the applications of designing this type of virtual agents is to inhabit computer-controlled avatars in a Multi-user Virtual Environment (MVE). Ideally, these intelligent virtual agents should be able to behave autonomously and interact with other avatars controlled by the computer or by real humans. Although some commercial MVE systems, such as AcitveWorld [20] and Blaxxun [21], have provided programming interfaces for users to create virtual agents controlled by another program, no emotional interactions with the real users are allowed.

Several reasons hurdle the realization of these desirable features on most MVE. First, the creation of an intelligent agent presents the same problem as imitation of intelligence on machines, which is a difficult problem that all researchers in Artificial Intelligence (AI) strive to tackle. However, do we really need a complete AI solution for the virtual agent to look intelligent or human-like? We think the virtual agents could look plausible if they can have unpredictable but reasonable responses to the user's inquiries. In this work, we have chosen to model the virtual agent's psychological state and use it to generate emotional behaviors, such as route and gesture selection. Second, although virtual agents could be made plausible with limited, shallow AI techniques, realizing these functions may still rely on programmers with AI expertise to spend a considerable amount of time to implement them. It could be more desirable if the behaviors of the agents are created by a psychological reasoning engine that takes only a few parameters specified by regular users. Third, if we have such a mechanism to author the agents with ease, we also need to have a MVE that can support a tailored protocol to facilitate the communications among avatars as well as a mechanism to update the mental state according to the interactions.

In order to address the aforementioned issues, we have proposed a psychological model and an update mechanism for interactions between avatars (real or virtual) in MVE. First, we have designed a psychological model consisting of emotion, memory, and personal traits to affect the behaviors (motion and gesture) of the agents. Second, in the implemented system, the personal traits of each virtual agent are specified in a configuration file prepared by the designer at startup time. The mental states of these agents are updated at run time according to the interactions between the avatars. Third, we have implemented a virtual environment system that is used as the experimental testbed for autonomous virtual agents with emotional behaviors.

2: RELATED WORK

The topic of intelligent virtual characters in a MVE has attracted much attention in Computer Graphics as well as Artificial Intelligence. In [13], the authors proposed a system of animated characters consisting of three parts: X3D, DLP, and STEP. STEP is a scripting language for describing character animation while DLP, Distributed Logic Programming, is used to author event-driven logics for animated characters. The work in [5] also proposed to add new nodes into X3D for character animation. However, the focus of this work is on enhancing the animation functions of virtual characters in X3D, and no AI solution has been proposed. In [19], the authors have proposed the InViWo agent architecture

[1] In this paper, the term of "avatar" refers to the geometric entity representing the user in a MVE. Virtual characters and virtual agents all refer to the computer-controlled avatars.

aiming to present an agent with various types of user interfaces. Through the Marvin language, one can author the intelligence of a virtual character. However, it lacks a flexible way to script animations for the character. These researches all have their own features and focuses but most of them do no provide an easy and intuitive way for a novice user to author an intelligent agent. In addition, most of them do not account for emotion modeling.

Autonomy is a key feature for a virtual agent to look believable in a virtual environment. Although many factors can affect the autonomy of a virtual character, we have chosen to focus on emotion in this work [16]. Most of the previous researches on incorporating emotion into intelligent agents are goal-oriented, which means that emotion is affected by the fact that the goal is achieved or not. For example, the models in PEN [11] and OCEAN [4] are all closely related to goal achievement. However, for a virtual environment that can run for a long time, it is not easy to specify all the goals in advance, especially if we allow the real users to interact with the virtual agents at run time. In addition, these models have some limitations as pointed out by [2].

According to [17], appraisal and coping are the two typical ways that a real human deals with emotion. Appraisal is the process of generating emotion by assessing the person-environment relationship. The famous OCC emotion model [18] is mainly based on appraisal. On the other hand, coping is the process of dealing with emotion, either externally by forming intentions to act in the world or internally by changing the agent's interpretation of the situation. [16][17] are all examples focusing on this aspect. The most common way to express emotion (coping) is by facial expression. Many researches [7][8] have focused on creating facial expressions for various emotion states. However, without the helps of intonation and contexts, judging the emotion state of a virtual agent simply by facial expression could be difficult. On the other hand, in some occasions, expressing emotion with other contextual means such as dialogs, body gestures, and walk styles, could be more effective when viewed in a distance. However, not much previous work has addressed this issue expect for [3], which allows a user to display appropriate body motions for a virtual agent according to its mental state. Nevertheless, the work only focused on body motions and was not designed to be used in a MVE.

Several researches, such as VHML [23] and MPML [22], have attempted to address the issue of annotating emotion for the purpose of communication. The work in [6] used MPML to mark up emotion states that can be sent to other clients for display. Similarly, VHML has a subset of tags, called EML, for emotion markup. Most of the display mechanisms for emotion available on these clients focus on facial expressions and simple gestures that can be easily computed on a remote client. For motions that are generated automatically with high-level goals and possibly affected by emotion states, it could be a better idea to generate them by the originating client because of the relatively high computational load of generating such motions.

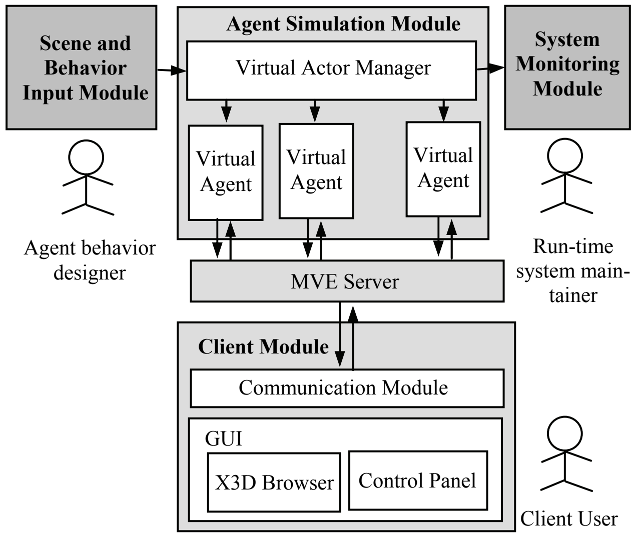

Fig. 1. System architecture of the experimental MVE system

3: SYSTEM OVERVIEW

The design goal of the system proposed in this paper is three-fold. First, we hope that the system can provide behavior-rich agents that can interact with the real users. These rich behaviors include dialogs selected from a given vocabulary, gestures available in the motion repository, and dynamically generated trajectories. Second, these behaviors should be generated at run time based on the emotion model of the intelligent agents and the interactions with other agents. Third, the behavior scripts should be intuitive to create by non-programmer designers.

3.1: System Architecture

The overall architecture of the proposed system is depicted in Fig. 1. The system consists of four main modules: *scene and behavior input module*, *system monitoring module*, *agent simulation module*, and *the client module*. The scene and behavior input module accepts the geometric description of the scene in a common 3D format such as VRML/X3D [24]. The geometric information is used to generate motion paths for the agents to trace. The designer of agent behaviors should also be able to specify the available dialogs, motion repository, and the default mental state for each virtual agent. The module of system monitoring is used for the system maintainer to monitor the agent status including the locations and future paths of each agent as well as their emotional states. The agent simulation module is in charge of the creation of virtual agents and the simulation of their behaviors by interacting with other avatars (including virtual agents and real users). The client module provides a graphical user interface for displaying the 3D world described in the X3D format,

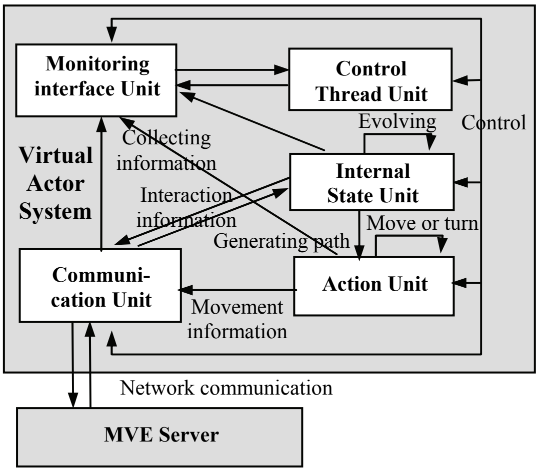

Fig. 2. Interoperation of various units in the virtual agent simulation module

and a control panel allowing the real user to interact with the virtual agents.

3.2: Simulation of Virtual Agent

The agent simulation module is the core for the generation of the autonomous behaviors of the agents. It consists of five interoperating units (as depicted in Fig.2) designed for specific functions: *control thread unit*, *internal state unit*, *action unit*, *monitoring interface unit*, and *communication unit*. The control thread unit is the simulation engine driving the progression of the simulation in other units. The internal state unit keeps the mental state of the virtual agent, which include emotion states, memory, and personal traits. This unit is also capable of interpreting other agents' message and updating its mental state accordingly. The emotion state is sent to the action unit to generate appropriate behaviors including dialogs, gestures, and motion paths. These behaviors are then sent to the communication unit for delivering the corresponding messages to the MVE server, who will in turn transit the messages to other clients. The monitoring interface unit collects information from the other four units and delivers the consolidated information to the external system monitoring module.

4: MENTAL STATE MODELING AND UPDATING MECHANISM

4.1: Modeling Mental State

The objective of this system is to create versatile behaviors for an agent based on its mental state, including emotion, memory, and personal traits. An agent should behave differently (coping) according to its personal traits under different emotion states or for different target subjects.

The number of basic emotion states reported in the literature ranges from six to a dozen [9][10][12]. The basic emotion states that most people agree upon include happiness, sadness, hate, fear, surprise, and anger. Among these emotions, surprise is the one that is more difficult to model and express. In addition, we consider hate in the memory model since it is usually associated with a target subject. Consequently, we adopt the remaining four emotion types as the basic states for the emotion of a virtual agent. In fact, according to how these emotions are elicited, we can further model the emotion of an agent with three independent dimensions allowing quantitative measures: Happiness (E_h) *vs.* Anger (E_a), Sadness (E_s) *vs.* Non-sadness (E_{ns}), Fearfulness (E_f) *vs.* Fearlessness (E_{nf}). The attribute on a dimension is quantitatively represented by a single parameter value.

In addition, we also use the following four attributes to model the memory of an agent upon other agents: Favorable impression (M_i), Familiarity (M_c), Sadness impression (E_{so}), and Fearfulness impression (E_{fo}). As will be explained in the next subsection, the favorable impression and familiarity are the evolving attributes that may affect the interpretation of the behaviors of other agents. E_{so} and E_{fo} are the emotion states of other agents that one can acquire from the last interaction.

Emotion elicitation and modulation may vary greatly for different individuals. Some people tend to get angry more easily while others may express their emotion more drastically. We regard these tendency and intensity as part of personal traits. For the emotions mentioned above, we have used the following ten attributes to describe personal traits on emotion: Tendency for happiness (T_h), Tendency for anger (T_a), Tendency for sadness (T_s), Tendency for non-sadness (T_{ns}), Tendency for fearfulness (T_f), Tendency for fearlessness (T_{nf}), Intensity of expressing happiness (I_h), Intensity of expressing anger (I_a), Intensity of expressing sadness (I_s), and Intensity of expressing fear (I_f).

In sum, we have three attributes for the emotion states, four attributes for memory, and ten attributes for personal traits. How to update them according to the interaction with other agents will be described further in the following subsections.

4.2: Defining Interaction Events

In this section, we first focus on defining an interaction event for reflecting the emotion state and intention of the virtual agent. These events are delivered in the form of textual dialogs associated with qualitative and quantitative attributes for the following three dimensions: *sadness*, *fear*, and *friendliness*. That is, the emotional attributes of each interaction event are represented as: (Sadness action type, sadness action intensity, fear action type, fear action intensity, friendliness action type, friendliness action intensity). An action and its intensity are used for each dimension to describe the user's intention. For example, the emotional attributes of the message: "You are dead!" could be coded as (none, 0, threatening, 80, friendliness, -70). This interaction event is selected when the agent is fearful and does not find you friendly. Several common actions for each

Table 1. Interactions for different emotions

Emotion Types	Actions	Meanings	Selection reason
Sadness	Complaint	Expressing one's sadness	One is sad.
	Encouragement	Attempting to reduce other's sadness	Other is sad, and one is friendly to him.
	Scold	Attempting to increase other's sadness	Other is sad, and one is not friendly to him.
Fear	Appeal	Expressing one's fear	One is fearful, and friendly to other.
	Begging	Expressing one's fear	One is fearful, but not friendly to other.
	Comfort	Attempting to reduce other's fear	Other is fearful, and one is friendly to him.
	Threatening	Attempting to increase other's fear	Other is fearful, and one is not friendly to him.
Friendliness	Friendship	Expressing one's friendliness	Must be expressed but the value could be 0 (neutral).

emotion category have been identified and shown in Table 1. Some of them (e.g. complaint) are used to express one's emotion state while others (e.g. encouragement) are used to affect others' emotions. The meaning of each action and the reason that it could be selected are also listed in the table. Note that we assume that the actions in each category are exclusive. For example, threatening and begging cannot exist at the same time. However, we can allow actions in different categories to coexist. For example, the message of "Don't be sad. I will help you." may imply encouragement as well as friendliness.

4.3: Selecting Interaction Events

Each virtual agent is given a library of interactions events at startup time. The interactions with another agent happen automatically when it gets into the view range of the virtual agent. As depicted in Fig. 3, the virtual agent selects an appropriate interaction event from the library according to its current mental state including its impression on the other agent's mental state. As mentioned in the previous subsection, the interaction events are classified into two categories according to the agent's intention: expressing one's emotion state and affect the other's emotion state. For each type of intentions, we design functions to map the mental state of the agent into the three basic emotion attributes for expressiveness: *sadness*, *fear*, and *friendliness*. We then select the most appropriate interaction event from the library that can reflect the intention of the current mental state. We first compute the overall impression (K), ranging from -100 to 100, on another agent by the use of the following formula:

$$K=((E_h)*(100-M_c)+(M_i*M_c))/100 \quad (1)$$

A virtual agent may select actions to affect another agent's emotion or express its own emotion according to the value of K. For example, it may select the action of "scold" to trigger negative reaction (NE) if K is larger

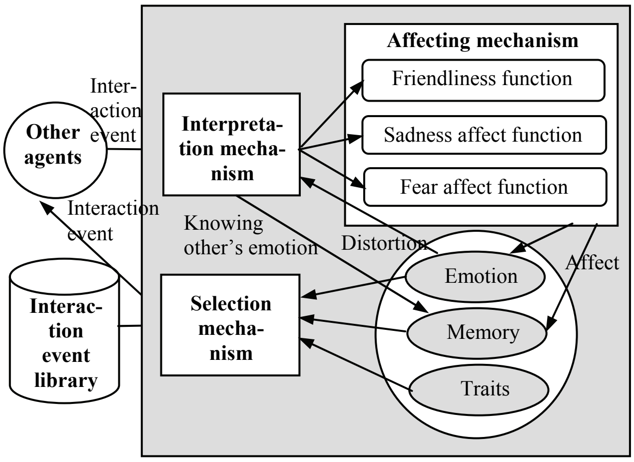

Fig. 3. Update mechanism of internal state with interactive events

than some threshold or "encouragement" to trigger positive reaction (PE) if K is smaller than some threshold. If K is between NE and PE, then no action in this category is selected. If an action is selected, its intensity is computed as $E_{so}*|K|*I_s/100$. The actions for the categories of sadness and fear are selected in a similar manner. For the category of friendliness, if K is greater than 0, the intensity of friendliness is computed as ($K * I_h$); otherwise, it is computed as ($K * I_a$). The agent may also take actions to express its own emotion. For example, to express its sadness, it can choose "complaint" and define the intention as ($E_s * I_s$). Similarly, according to the fact that K is great or less than 0, we will select the action of "appeal" and "begging", respectively, for the category of fear. The intensity is computed in a similar fashion.

4.4: Interpreting Interaction Events

The aforementioned interaction events affect mental states of the recipient agents. In addition to the delivered message itself, the current emotion state, memory, and personal traits all play some roles in interpreting the interaction events [16]. For example, a dialog message of "You are great!" may be interpreted differently if it is from a person you like or dislike. In other words, we think we usually distort the meaning of a message when we interpret it according to our mental state. According to [12], people tend to ignore the things that they do not like and think toward what they like. For example, a threat from a friend may be interpreted as a warning while an unpleasant comment from someone we dislike may be regarded as a threat. Therefore, emotion plays an important role in distorting the interpretation.

The impression of the virtual agent on other agents will affect how it interprets or distorts the received interaction events. In our system, the distortion rate is defined as a function D of the overall impression (K) defined in Eq.(1) multiplied by some tendency factor (N) to increase or decrease the effect according to personal traits on tendency. Assume that the original effect on an emotion attribute (happiness, sadness, or fear) carried by

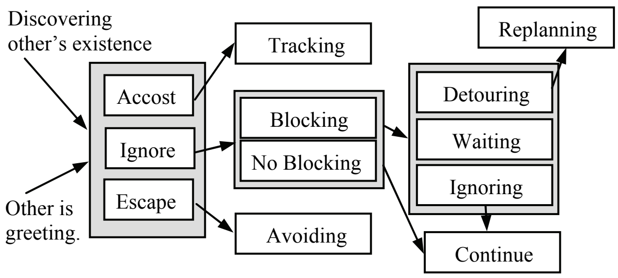

Fig. 4. Strategies of motion selection for different events

an interactive event is E_o. Then the final effect E_f after distortion is defined as

$$E_f = D(K) * N(T) * E_o \,, \tag{2}$$

where D and N are distortion and tendency functions designed by the scene designer while K and T are the overall impression and tendency for a specific attribute, respectively. In addition, the memory of the agent on the encountered agent, such as favorability and familiarity, will also be altered through interactions. In our system, we assume that the amount of incremented familiarity is defined as $\delta M_c = \delta M_{min} + (E_f * \delta R)$, where δM_{min} is the minimal amount for each interaction and δR is a user-specified increment rate.

5: ROUTE AND GESTURE SELECTION

In addition to dialog events, we also allow a virtual agent to express its emotion via route selection and body gestures. Route selection means automatic generation of a collision-free path for the agent based on its mental states and the other agents' locations. The agent can also use body gestures such as hand gestures or walking styles to reflect its mental state. The animations of movement and gestures for an agent are described in an extensible animation language called XAML [14] and realized in a multi-user virtual environment system (called IMNET[15]) supporting this animation language.

5.1: Selection of Route

Route selection results from reaction to the existence or actions of other agents. According to the interaction events and inter-personal relationship with other agents in the scene, a virtual agent could choose to use various strategies such as follow, escape, or accost to reflect its mental state. In our system, we have designed several route selection strategies, as shown in Fig. 4, to choose or compute an appropriate move path. When the agent discovers the existence of other agents or is greeted by these agents, it may choose to have the reaction of accosting, ignoring, or escaping. If it chooses to talk to the encountered agent voluntarily, the action to take would be "following". If it chooses to escape from the agent, it will change its route to avoid being seen. If it chooses ignoring, according to the fact that the path is obstructed or not, the agent could choose to take a detour by re-planning the path, waiting, or following.

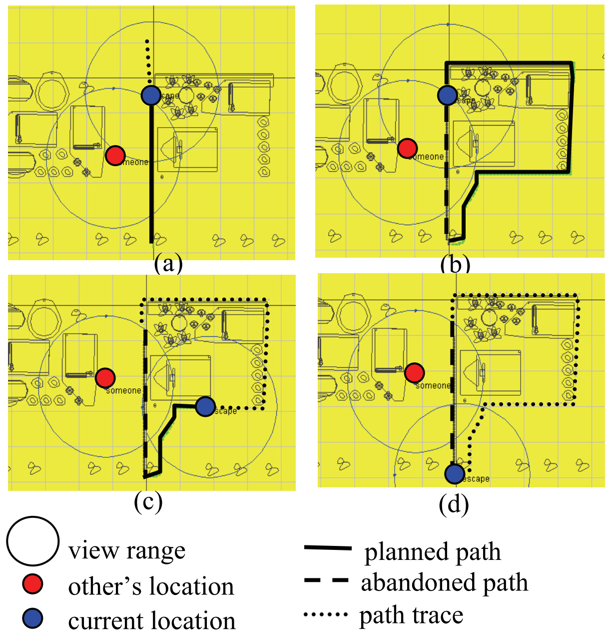

Fig. 5. Example of how a virtual agent detours to avoid someone he does not like

When an encountering event occurs, the virtual agent selects its reaction based on its mental state. In our system, if an agent is in the state of favorable impression, malice, scold, comfort, encouragement, threatening, or begging, it tends to talk to the encountered agent. In contrast, if the agent is in the state of fearfulness or it encounters someone it dislikes, then a "run-away" action may be chosen. If the agent is in a bad mood, it may not want to choose any active reactions and simply ignore the event. The rules for triggering reactions are mainly based on the impulse index defined as. $I_{mp} = (100\text{-}E_s)/100 * (|K|\text{+Max}(E_f, E_{fo}, E_{so}))$. If the impulse index is larger than some threshold, the agent will choose more vigorous reactions such as scolding and encouragement. Furthermore, we define the emergency index as a function of the impulse index: $I_{em} = (100\text{-}E_s)* E_f*I_{mp}$. According to the value of I_{em}, the agent may choose to wait for more time, re-plan its path, or keep tracing the old path.

A key module in implementing the autonomous virtual agent is the ability to plan its path for a given destination and intention. We have implemented an efficient path planner based on potential field techniques [1] to search for a collision-free path connecting the current location to the destination. For the case of escaping an encountered agent, we can use the same planning algorithm but define the view range (a circle) of the encountered agent as obstacles as well. An example of route selection created by the system is shown in Fig. 5. The agent re-plans its route and takes a detour as it encountered the static agent.

5.2: Selection of Gesture

The gestures available for selection in our system are classified into five categories: *greeting*, *speaking*, *fare-*

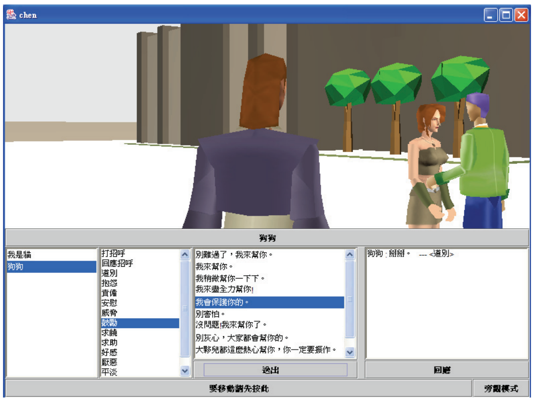

Fig. 6. Example of user interface for creating and displaying interaction events with dialogs and gestures

well, *walking*, and *subconscious motions*. First, the degree of friendliness is used to choose a greeting or farewell gesture from a library of motions in various degrees. We have designated at least an appropriate hand gesture for each dialog message such that they can be displayed along with the dialog. Furthermore, a walk motion with an adjustable speed has also been implemented. Lastly, subconscious motions are used when the agent is not walking or speaking. These motions are selected according to the three emotion attributes only (without memory) since this type of motions does not need to have a target subject. A screen dump of the client-side interface for interacting with the virtual agents with dialogs and 3D animations is shown in Fig. 6.

6: CONCLUSIONS

It has been a design goal for long to allow autonomous virtual agents to inhabit a MVE such that rich interactions and new applications can be developed. In this paper, we have proposed a novel interaction mechanism with a simple psychological model for a virtual agent to interact with others in an emotional and personal way. We have also realized such a virtual agent mechanism in a MVE platform that allows us to extend it to support these functions. Rich interactions and interesting scenarios have been observed. In the future, we would like to further research on how to parameterize the animation such that the emotion states can be expressed quantitatively. We would also like to investigate how to allow a virtual agent to present the interaction events of various forms in a more seamless way.

REFERENCES

[1] Barraquand, J., Langlois, B. Latombe, J.C.: Numerical Potential Field Techniques for Robot Path Planning. *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 22, No. 2. (1992).

[2] Bartneck, C.: Integrating the OCC Model of Emotions in Embodied Characters. *Proc. of the Workshop on Virtual Conversational Characters: Applications, Methods, and Research Challenges*, Melbourne (2002).

[3] Bécheiraz, P. and Thalmann, D.: A Behavioral Animation System for Autonomous Actors personified by Emotions. *Proc. of the First Workshop on Embodied Conversational Characters* (WECC '98), Lake Tahoe, California (1998).

[4] Costa P. T., and McCrae, R.R., Normal personality assessment in clinical practice: The NEO personality inventory, *Psychological Assessment*, vol.4, 5–13 (2003).

[5] Dashselt, R., Rukzio, E.: Behavior3D: An XML-Based Framework for 3D Graphics Behavior. *Proc. of the 8th Intl. Conf. on 3D Web Technology* (2003) 101–112.

[6] Descamps, S., Barakonyi, I., Ishizuka, M.: Making the Web Emotional: Authoring Multimodal Presentations Using a Synthetic 3D Agent. *Proc. OZCHI-2001 (Computer-Human Interaction, SIG of Australia)*, Perth, Australia (2001) 25-30.

[7] Egges, A., Kshirsagar, S., and Magnenat-Thalmann, N.: A Model for Personality and Emotion Simulation. *Proc. of Knowledge-Based Intelligent Information & Engineering Systems* (2003).

[8] Egges, A., Zhang, X., Kshirsagar, S., and Magnenat-Thalmann, N.: Emotional Communication with Virtual Humans, *Proc. of Multimedia Modeling Conf.* (2003).

[9] Ekman, P.: Expression and the Nature of Emotion. Ekman, P., and Scherer, K. (Eds.), *Approaches to emotion*, Hillsdale, N.J.: Erlbaum (1984) 319–343.

[10] Ekamn, P.: *Telling lies*. New York: Norton (1985).

[11] Eysenck, H.J.: Biological dimensions of personality. L. A. Pervin, (ed.), *Handbook of personality: Theory and research,* New York: Guilford. (1990) 244–276.

[12] Gleitman, H.: *Basic Psychology*, 3rd edition. New York, N.Y.: Norton, (1992).

[13] Huang, Z., Eliëns, A., Visser, C.: Implementation of a Scripting Language for VRML/X3D-based Embodied Agents. *Proc. of the 8th Intl. Conf. on 3D Web Technology*, (2003) 91 – 100.

[14] Li, T.-Y., and Liao, M.-Y.: A Scripting Language for Extensible Animation. *Proc. of 2003 Computer Graphics Workshop*, Taiwan, (2003).

[15] Li, T.Y., Liao, M.Y., and Tao, P.C.: IMNET: An Experimental Testbed for Extensible Multi-user Virtual Environment Systems. ICCSA 2005, LNCS 3480, O. Gervasi et al. (Eds.), Springer-Verlag Berlin Heidelberg, (2005) 957–966.

[16] Marsella, S. Gratch, J., Modeling Coping Behavior in Virtual Humans: Don't worry, Be Happy. *Proc. of the 2nd Intl. joint Conf. on AAMAS 2003*, (2003) 313 – 320.

[17] Oliveira, E., Sarmento, L. Emotion Advantage for Adaptability and Autonomy. *Proc. of the 2nd Intl. joint Conf. on AAMAS*, (2003) 305 – 312.

[18] Ortony, A., Clore, G., Collins, A: *The Cognitive Structure of Emotions*, Cambridge University Press (1988).

[19] Richard, N.: InViWo Agents: Write Once, Display Everywhere. *Proc. of the 8th Intl. Conf. on 3D Web Technology*, (2003) 123 – 127.

[20] ActiveWorlds, URL:<http://www.activeworlds.com>.

[21] Blaxxun, URL:<http://www.blaxxun.com>.

[22] MPML, Multimodal Presentation Markup Language, URL:<http://www.miv.t.u-tokyo.ac.jp/MPML/>.

[23] VHML, Virtual Human Markup Language. URL:<http://www.vhml.org>.

[24] X3D 2002. URL:<http://www.web3d.org/x3d/>.