

Dialogues for Embodied Agents in Virtual Environments

Rieks op den Akker and Anton Nijholt¹

Centre of Telematics and Information Technology (CTIT)
University of Twente, PO Box 217
7500 AE Enschede, the Netherlands
{infrieks, anijholt}@cs.utwente.nl

Abstract. This paper is a progress report on our research, design, and implementation of a virtual reality environment where users (visitors, customers) can interact with agents that help them to obtain information, to perform certain transactions and to collaborate with them in order to get some tasks done. We consider this environment as a laboratory for doing research and experiments with users interacting with agents in multimodal ways, referring to visualized information and making use of knowledge possessed by domain agents, but also, in the future, by agents that represent other visitors of this environment. As such, we think that our environment can be seen as a laboratory for research on users and user interaction in (electronic) commerce, educational and entertainment environments.

1 Introduction

We report about the progress of our research on dialogue agents in a virtual reality environment. The environment is a virtual theatre, implemented in VRML and Java-based extensions that allow inter-agent communication, speech recognition, speech synthesis, database access and animation. In this environment visitors can get information about theatre performances by asking questions in natural language (using the keyboard) to an information agent (Karin is her name) and they can make reservations. A second agent (the navigation agent) knows about the environment, can answer questions about the environment (using speech recognition and synthesis) and can guide the visitor to locations or information presentations. Obviously, any visitor has the freedom to walk around in this 3D environment and in this way, and not necessarily goal-directed, explore the theatre.

As will become clear from section 2, our approach in designing this environment has been bottom-up. At this moment we are in the process of designing, using an agent-oriented approach, a new version of this environment. In the new environment we want to exploit current technical possibilities that allow a multi-user environment, where both users and artificial agents (with the possibility to have both of them

¹ The research reported in this paper has been financed by the Dutch Telematics Institute (U-Wish project) and the VR Valley Twente Foundation in Enschede.

visualized as humanoids, i.e., 3D objects that resemble humans, both in their appearance as in their behavior) are part of a multi-agent system. This allows interaction between people who are virtually present in a scene, interaction between people and artificial domain agents and interactions with (shared) objects in the environment.

It will be clear that the aim to model and build such environments is a rather ambitious one. Research on many interesting issues can be pursued in such an environment. From a computationally linguistic point of view it is of course the presence of multiple dialogue partners that constitutes a challenge. Moreover, they may know about each other, they sometimes may see each other and they see their environment, inviting references to this visualized environment, shared behavior and shared tasks. Multimodality in interactions between agents (whether they are human or artificial) is another issue. Any utterance of a user may invoke an action of one or more agents, including speech output and synchronous lip movements, performing a certain task (made visible by animation of a humanoid), a change in an agent's facial expression, change of gaze direction, etc. Agent technology is another issue at hand. How can we 'control' the interaction between agents, the users and the (objects in the) environment, how can we allow users to introduce their own agents (see e.g. [10] in such a way that they can participate in the already existing environment? Moreover, our agents are designed to have some kind of 'interaction intelligence'. How can we integrate this intelligence in models of beliefs, desires and intentions? From an agent-oriented design to software engineering issues is a small step. In [11] some preliminary research is reported on the specification of our and similar complex virtual environments that allow interactions between multiple agents. Last but not least, standards have to be developed in order to be able to assemble independently developed components of inhabited virtual worlds.

2 Background

In [2] we reported about the original aims of our project and the approaches we took at that time. We discussed a natural language dialogue system that offered information about performances in some (existing) theatres and that allowed visitors to make reservations for these performances. Based on Wizard of Oz experiments we designed a system that incorporated rather traditional theory and approaches in computational linguistics and natural language processing. Unfortunately, theory and approaches did not allow to build a system that could be accessed by the general audience. That is, an audience interested in visiting cultural events and making reservations for these events, but not at all familiar with (shortcomings of) information and language technology and not at all ready to adapt its behavior to rather primitive interaction behavior of the interface to the theatre information system.

In the next phase of our research we decided to introduce a model of natural language interaction between system and user that was much more primitive from a linguistic point of view, but much more intelligent from a practical and pragmatic

point of view. In [6] this system was discussed.² In short, we introduced a natural language understanding system between the user and a database containing information about performances, artists and prices. The intelligence of this system showed in the pragmatic handling of user utterances in a dialogue. The ‘linguistic intelligence’ was rather poor, however the outcome of a linguistic analysis could be given to pragmatic modules which in the majority of cases (assuming ‘reasonable’ user behavior) could produce system responses that generated acceptable utterances for the user. With this we don’t mean that for any user utterance the next system utterance could be considered as a satisfactory answer or comment. Rather it should be considered as an utterance containing cues how to continue the dialogue in order to come closer to a satisfactory answer. The general idea behind this system was that users learn how to phrase their questions in such a way that the system produces informative answers. Certainly, we can design systems such that they ‘teach’ (preferably in a non-intrusive way) the users to do so. For instance, the system prompts can be designed in such a way that users adapt their behavior to the system, the prosody of system utterances (in a spoken dialogue) can invite users to provide once more information that they already assumed to be known by the system and, more generally, the system’s quality may improve by assuming that the user addresses context information that has been made available by the system, for example because information concerning the dialogue or its content has been visualized on the screen.

3 Embedding: A Dialogue System in a Virtual Reality Environment

We embedded our theatre information and booking system in a virtual reality (3-D) environment that allowed visitors to walk around in the theatre, to approach an information desk with an agent (Karin) with a talking face that is able to address the user in a natural language dialogue about available performances [7]. The theatre has been built, using the Virtual Reality Modeling Language VRML, according to construction drawings provided by the architects of the building. Visitors can explore this building and its environment, walk from one location to another, ask questions to available agents and objects, click on objects, etc. Karin, the receptionist of the theatre, has a 3-D face that allows simple facial expressions and lip movements that synchronize with a (Dutch) text-to-speech system that mouths the system’s utterances to the user. Presently, in our implementation of the system, there is no sophisticated synchronization between the (contents of the) utterances produced by the dialogue manager and corresponding lip movements and facial expressions of the Karin agent.

In Figure 1 we see Karin behind her desk. Near her we see some posters and a floor map of the theatre. Someone just entered the theatre. Therefore the door is open and it allows a view outside the theatre. Not shown is a monitor on the desk which allows previews of performances that may be suggested by Karin. Visitors do not necessarily

² The ‘parser’ of this system has become part of some commercial systems available from Carp Technologies (<http://www.carp-technologies.nl>). Among these systems are an automatic summarizer of arbitrary text and an assistant for navigating on a company’s website.

have to talk to Karin (using the keyboard for input). They can explore the building,

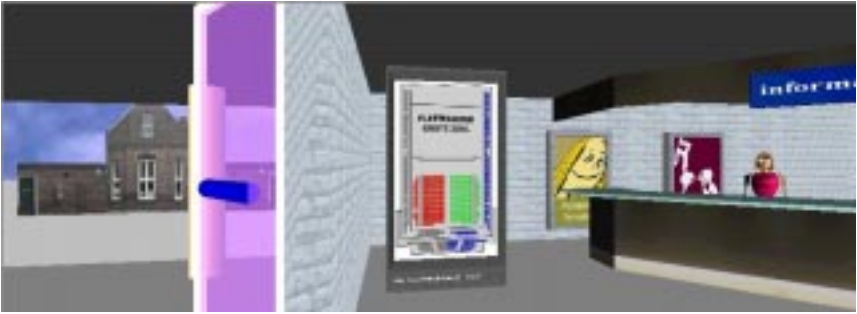


Fig. 1. Karin Behind the Information Desk

enter the performance halls, watch the stage from a particular position, play with the stage lights, etc. The view on the virtual world is part of a screen where there is a control panel for controlling and navigating this world. In addition there are some menu windows in which the user can type questions for Karin, where Karin's answers are displayed (besides being synthesized) and where a table (to which the user can make references) displays alternatives when there are too many performances that satisfy a user's request. These additions make the interaction between system and visitor multimodal.

The dialogue system, represented by Karin, has been embedded in the virtual world. However, no changes were made to the dialogue model. Clearly, this is necessary. The virtual theatre invites visitors to make references to the visual context (e.g., to the posters or the floor map), and to, for example, the visual appearance of the Karin agent. The dialogue system should allow this. We will return to this problem in the following sections.

It is not difficult to think of other agents that can play a useful role for visitors of our environment. Most obvious is an agent that helps the visitor to find his or her way in the virtual world. For that reason we introduced a navigation agent and an agent platform. The current navigation agent knows about the building and can be addressed using speech and keyboard input of natural language. No real dialogues are involved. The visitor can ask about existing locations in the theatre and when recognized a route to this location is computed and the visitor's viewpoint is guided along this route to the destination. The navigation agent has not been visualized as an avatar. Its viewpoint in the theatre is the current viewpoint from the position (coordinates) of the visitor in the world. The Java based agent framework provides the protocol for communication between agents. It allows the introduction of other agents. For example, why not allow the visitor to talk to the theatre seat map or to a poster displaying an interesting performance? Unlike its predecessor, the version of the virtual theatre with a speech recognizing navigation agent has not been made accessible to the general audience by putting it on the Web. Although speech recognition is done at the server (avoiding problems of download time, ownership, etc.) there are nevertheless too many problems with recognition quality and synchronization with the events in the system. However, further work on the navigation agent is in progress. Part of this work is on user preferences on navigation

in virtual worlds, part is on modeling navigation knowledge and navigation dialogues, part is on adding instruction models to agents (Evers [4]) and part is on visualization (Kiss [5]).

4 Dialogues with Agents in a Virtual Environment

In this section we discuss the consequences for the design of virtual environments if we allow human users to communicate with human-like agents (like Karin and a Navigation agent in our virtual theatre) using natural language. First notice that in any communication by means of natural language, there is an imaginative ('virtual') world: the worlds of objects and relations between them about which the participants communicate. The main problem we are confronted with when we allow users to communicate in their natural language with a software 'agent' do not come from embedding the agents in a virtual environment. Natural language is unlike a formal language a vague notion, and any formal language model will necessarily be incomplete. Even natural dialogue systems for such restricted domains as theatre information show how difficult it is to define a satisfying user, dialogue, and language model. This means that in designing natural language models for dialogue systems it is very important that the system can be extended and adapted easily on the basis of experiments with earlier versions.

Communication situated in a visible or otherwise observable (virtual) shared environment allows the communicating partners to support their communicative acts by other means of directing (like gazing or pointing) than linguistic reference. Introducing this multi-modal support for language communication in some cases helps the agents to understand each other but it introduces some new and challenging problems as well. One of them is the problem of coreferencing to shared visible objects. The phrase "that door" should be attached to some visible object in the environment and assumes that the agents share the visibility of this object. The 'geometrical' virtual environment (described in VRML code or in whatever virtual modeling language) must be described on an abstract conceptual and linguistic level as well. The agent should somehow be able to know what object the user points at even in case it is not in direct view of the agent and it should therefore be able to match this way of referring with the linguistic reference ("that door").

Natural language understanding is more than keyword recognition; we need syntax, i.e. a grammatical model of the language and pragmatics, i.e. a model of communicating agents. The grammatical model describes the relation between the order in which words and phrases in a linguistic utterance appear and their function in the whole utterance. It assumes that the words that occur in a user utterance are somehow related in a sensible way. This grammatical model underlies the first step in natural language understanding. In our case the parser for unification grammars (PATR like format) outputs a set of features structures: a syntactic/semantical analysis of (parts of) the input 'sentence'. The grammar and lexicon need to be easily adaptable for other domains, so different agents that can dialogue about different domains can share the general parts of the (Dutch) grammar and the lexicon.

The pragmatical model underlies the second step in understanding. This model assumes that an utterance is in a rational way related to the dialogue: it is a linguistic utterance of some communicative act that “fits” in the dialogue. This ‘fits in the dialogue’ does not exclude communicative acts that have the intention to control (in particular start or stop) the dialogue. A dialogue is a sequence of conversational acts. In a conversational act the actor addresses himself to an agent by means of an utterance. This level of conversational acts and pragmatics is in great part independent of the particular natural language that is used for communication. The agent is confronted with two problems: a) where do the words and phrases refer to and b) what is the intention of the actor with his act, i.e. what is (are) the conversational act(s) that underlies this utterance? The pragmatical model specifies the relation between the output of the parser - a message representing the syntactic/semantical structure of the user utterance - and the possible conversational acts as the possible source of this message.

An example: in our theatre information system the utterance “Are there any other opera performances tonight?” is interpreted as being an utterance that points to a conversational act in which the actor wants to know about opera performances tonight in a particular theatre (default, i.e. if not otherwise stated) because he is interested to go there in case there are such performances (default). The word “other” is only rational in case this utterance fits in a dialogue in which there was already spoken about opera performances. Moreover, the act assumes that the actor believes that the addressee can provide him the requested information. It will be clear that the pragmatical model should contain this knowledge about conversational acts in order to be able to deny this implicit assumption so the addressee can answer with: “I’m sorry I don’t know anything about opera performances” or to react by querying the database for other opera performances, than the one mentioned earlier in the current dialogue, or to react by: “What do you mean by ‘other’?” in case the agent does not know about other operas being discussed in the current dialogue.

The agent knows the structure (features and possible values) of the output delivered by the parser (by means of a type description) and can search for denotations of the values: in dialogue context, in the database of theatre information or in the set of actions it can perform. In most cases the parser will not give an unambiguous analysis of the input presented by the agent. For robust interpretation of elliptic utterances and non-grammatical input the agent has a conceptual model representing the relations between concepts in the domain. These concepts are referred to by words that occur in the lexicon of the parser.

The dialogue context consists of a focus stack containing linguistic items that can be referred to later on. Also references to objects in the virtual environment that are pointed at by the user (by mouse) are put on this focus stack, allowing simultaneous multi-modal interaction.

Unlike in the current implementation in which the dialogue acts of the agent are directly called by the user utterance, in the new design the agent decides what action to perform on the basis of his context-dependent interpretation of the user utterance. This allows a more flexible and intelligent system for action selection by the agent, based on his belief (dialogue and user knowledge) and his own intentions, supported by the knowledge in the pragmatical model.

Experiments with current dialogue systems show that it is important to distinguish between knowledge about the user that is confirmed, denied or only guessed by the agent on the basis of general (default) rules. The agent must decide whether to ask for confirmation in a implicit or explicit way, so the user can correct the agent if he has misunderstood him. This implies that the pragmatical model should model the intention of utterances like “No, I didn’t mean that” or “You are wrong” or the like. The recognition of conversational acts that are not about the primary domain (theatre performances, or objects in the virtual world) but about the acts, beliefs, or intentions of the participating agents in dialogue, is one of the most challenging problems in building useful natural language dialogue systems.

Any natural dialogue system - however restricted its primary domain - should allow the user to refer to these aspects of dialogue itself: language (naming; “I don’t know what you mean”), the participating agents (“What is your name?”) and the dialogue process itself (“As I said before ...”). This implies that these aspects of communication itself need to be modeled explicitly in a dialogue system.

5 Distributed Multi-user and Multi-agent Environment

In our environment we have different human-like agents. Some of them are represented as communicative humanoids, more or less naturally visualized avatars standing or moving around in the virtual world and allowing interaction with visitors of the environment. In a browser which allows the visualization of multiple users, other visitors become visible as avatars. We want any visitor to be able to communicate with agents and other visitors, whether visualized or not, in his or her view. That means we can have conversations between theatre agents, between visitors, and between visitors and agents. This is a rather ambitious goal which can not be realized yet, not only due to lack of theory as exemplified in the previous section, but also because current web technology does not allow free speech communication between multiple users and agents in virtual environments.

One of the main shortcomings from our point of view is the poor state of multi-user technology and the slow progress in establishing standards. VRML itself has become an ISO standard. It allows the modeling and implementation of 3D environments and of simple animations of objects. The environments can be visited with a standard web browser equipped with a VRML plug-in. More complex functionality can be obtained by connecting Java Applets to the plug-in using VRML’s External Authoring Interface (EAI). For example, in our virtual theatre the EAI has been used to build a version including speech recognition and in the current publicly web-accessible version it allows speech synthesis and synchronous lip movements for sentences that are generated by Karin’s dialogue system. Related to VRML other standards have been proposed or are under development. For our purposes, we are interested in:

- Humanoid Animation (H-Anim) standard [14]. This standard defines a structure and interface for humanoid like agents in VRML. It does so by defining a number of VRML node prototypes: Humanoid Node, Joint Node, Segment Node and Site Node. These nodes describe the visualization of the agent, the stiffness and

rotation of the joints (e.g., shoulder, elbow, knee), the segments (e.g., upper arm, jaw) and a viewpoint of the agent. An agent that conforms to the H-Anim standard can be plugged into a VRML world and controlled through its interface. Animations (not yet part of the standard) can be specified for the H-Anim agents.

- Living Worlds Standard [13]. At this moment Living Worlds is a working group rather than a standard. The aim of the working group is to define a conceptual framework and specify a set of interfaces to support the creation of multi-user and multi-developer applications in VRML. In [13] two concepts are mentioned: Interpersonal and Interoperable. The first concept refers to applications which support the virtual presence of many people in a single scene at the same time: people who can interact both with objects in the scene and with each other. Just to mention an example, when someone's avatar moves from one location to another, this movement should cause updates in the world that have to be made visible to all the clients that are connected to the world. The second concept refers to the possibility that such applications can be assembled from libraries of components developed independently by multiple suppliers. As a simple example, a user should be able to introduce his or her own VRML avatar in a world built by someone else. This requires control and adjustments of size, animations and possible interactions with the environment.

After some preliminary experiments with VRML multi-user environments (Sony Community Player, Blaxxun Contact, VNet) we now use the DeepMatrix system [9]. DeepMatrix is a multi-user virtual environment system based on Java and VRML. It has a client-server architecture which uses standards as TCP and UDP and which is compliant with the Living Worlds specifications. DeepMatrix offers users a choice of avatars. A user can supply his or her own avatar by providing a URL pointing to the VRML code of the avatar. Users (their avatars) are related to zones or rooms in the virtual world. Users that are related to the same room are updated on changes in this room. For example, a new user can enter the room, avatars move around, they initiate events (a door that opens because one of the avatars comes close to it), etc. The interface offered by DeepMatrix contains a chat area. Here users can type and read messages, see what other users are in the current room and they can explicitly activate some previously defined avatar behavior.

In Figure 2 we show a view of the stage of our virtual theatre using DeepMatrix. It shows an animated baroque dancer performing on the stage. This dancer has been imported and manually scaled down into our world (with permission) from the Baroque Dance Project of the Università degli Studi, Milano [1]. Close to the dancer we see a visitor's avatar which has been so impertinent to climb the stage in order to look at her more closely. This latter avatar has been built according to the H-Anim standard mentioned above. Its animations allow it to walk around following the coordinates of its owner's viewpoint position.

In the previous sections we talked about agents acting in our own virtual theatre. Karin was introduced as a 'visualization' of our existing dialogue system. She has extensive knowledge of performances that play in the theatre. She can move her lips and have some simple head movements in function of the dialogue. Once we had Karin it became clear that we needed an agent framework and we introduced a navi-

gation agent with some geographical knowledge and speech recognition capabilities. In fact, we have a multitude of potential agents. There is a piano player on stage with some simple predefined animations, there is a baroque dancer with animations synchronized with audio and there are visitors, able to move around, displaying walking movements with hands and legs. It will be clear that in order to maintain a virtual environment where we have a multitude of domain and user-defined agents we need some uniformity from which we can diverge in several directions and combinations of directions: agent intelligence, agent interaction capabilities, agent visualization and agent animation.



Fig. 2. The Virtual Theatre Stage in DeepMatrix: Visitor meets Baroque Dancer

Apart from dealing with problems in all kinds of subareas (e.g., those mentioned in section 4 of this paper) for our environment the following two lines of research have to be taken simultaneously in order to allow further useful research and extensions of our environment:

- Redesigning and extending our agent framework such that individual agents can represent (human) visitors (e.g., movements, posture, nonverbal behavior) and can stand for artificial, embodied domain agents that help visitors in the virtual environment (using multimodal interaction, including speech and language).
- Designing 3D VRML agents that are controlled according to the protocol of the agent framework, that can walk around in the virtual environment (either acting as a domain agent, hence displaying intelligent and autonomous behavior, or representing a visitor and its moving around in the environment). The geometry of the agents should be based on the H-Anim specification for a standard humanoid.

- Relating our agent framework to the theory of multi-agent systems and issues of autonomy, reactivity, pro-activity, social ability and learning. Some general frameworks for intelligent agents have been developed, among them the theory of belief-desire-intention agents which seems to be a good candidate (with different levels of abstraction) for our environment.

No existing multi-user environment system allows this advanced approach. When using DeepMatrix, to mention an example, we need separate channels for communicating with system agents and for communicating with other visitors using the chat extension. One reason to mention it again is that it is at least a serious attempt to comply with the Living Worlds specifications. The main elements of this specification deal with data distribution and scene synchronization. Below these elements are standards dealing with network and application protocols. Beyond these elements are standards dealing with the issues in the three lines of research mentioned above.

6 Gaze Behavior among Multiple Conversational Agents

Among the in- and output modalities we want to deal with in our future distributed virtual environments is gaze direction. This modality can help to resolve the problem to which of the visible agents a user directs a question. The role of gaze in dialogue and conversation has been studied by Cassell et al [3]. In Nijholt and Hulstijn [8] it is discussed how we can incorporate such results in annotated templates that are used for generation of system utterances in a dialogue system. Presently, we are doing experiments with a desk-mounted LC Technologies eyetracking system, where knowing where the visitor is looking at is detected by an infrared camera. On top of this camera is an infrared source projecting invisible light into the eye. This light is reflected by the retina and the cornea of the eye. These reflections make it possible to determine where a person is looking at. In particular, it is possible to determine to which agent a user is looking. This allows management of multi-user conversations in a virtual environment, where each user knows when and which other users are looking at him or her. This leaves to a certain degree open how the user is represented in the environment, but at least user gaze directional information can be conveyed. This approach allows visitors of our environment to address different task-oriented agents in such a way that speech recognition and language understanding are tuned to the particular task of the agent; therefore quality of recognition and understanding can increase considerably, since the agent may assume that words come from a particular domain and that language use is more or less restricted to this domain. That is, we can restrict lexicon and language model to the utterances that are reasonable given the agent. Obviously, we should try to visualize agents in such a way that it is clear from their appearance what they're responsible for and what a visitor can ask them. An attempt should be made to ensure that any agent is able to determine that he or she isn't the right agent to answer a visitor's questions and therefore should direct the visitor to an other task-oriented agent or to an agent having global knowledge of the task-oriented knowledge of the other agents in the virtual environment.

In the prototype we are using 3D texture-mapped models of humanoid faces. Muscle models are used for generating accurate 3D facial expressions. Each agent is

capable of detecting whether the user is looking at it, and combines this information with speech data to determine when to speak or listen to the user. To help the user regulate conversations, agents generate gaze behavior as well. This is exemplified by Fig. 3. Here, the agent speaking on the left is the focal point of the user's eye fixations. The right agent observes that the user is looking at the speaker, and signals it does not wish to interrupt by looking at the left agent, rather than the user. In this set-up we want to model a user and two agents, where the agents have related tasks. For reasons of experiment we want to make an explicit distinction between the information task and the reservation task of our information and transaction agent Karin.



Fig. 3. Gaze modelling in Conversations with More than One Agent

Hence, we have a Karin_1 and a Karin_2 who have to communicate with each other and with the visitor. Clearly, when during the reservation phase with Karin_2 it turns out that the desired number of tickets is not available or that they are too expensive, it may be necessary to go back to Karin_1 in order to determine an other performance. Although the separation of tasks may look a little artificial, it gives us the opportunity to experiment in the prototype environment and with a (modified) existing dialogue system, rather than being obliged to develop two new dialogue systems. Nevertheless, we can not expect a straightforward transfer of the research results in this prototype to the web-based environment of our virtual theatre. Depending on research on the agent framework and the design of human-like agents in this framework some of the results can be expected to be incorporated in the foreseeable future.

7 Conclusions

We surveyed our framework of research on issues related to dialogues with agents in virtual environments. Integration and scaling down of advanced research results to web-based environments are among the issues that play. Unlike many other virtual environments the public version of the environment has been made available to the general audience. This WWW environment (<http://parlevink.cs.utwente.nl/>) uses a database containing the performances that play in the local theatres of our home town. People can recognize the building, its performance halls and its environment. They can get information, by asking Karin, about performances, including reviews that are read by Karin. No real reservations can be made. The navigation part of the current system is also under scrutiny by the TNO Human Factors Research Institute in the

Netherlands. User evaluation studies will give directions for further research on navigation assistance in this particular environment. The original approach in our project was bottom-up. Now that we have gained sufficient experience we have decided to start a more comprehensive and top-down approach to agent-based virtual environments where we take our existing theatre environment again as a case study.

References

1. M. Bertolo, P. Maninetti and D. Marini. Baroque dance animation with virtual dancers. *Eurographics '99 Conference, Short Papers and Demos, Milan, 1999*, 117-120.
2. S.P. van de Burgt, A. Nijholt, T. Andernach, H. Kloosterman and R. Bos. Building dialogue systems that sell. *Proceedings NLP and Industrial Applications*, Moncton, New Brunswick, June 1996, 41-46.
3. J. Cassell and K.R. Thórisson. The power of a nod and a glance: envelope vs. Emotional feedback in animated conversational agents. *Applied Artificial Intelligence*, to appear.
4. M. Evers. The Jacob Project. URL: <http://wwwhome.cs.utwente.nl/~evers/jacob/index.html>
5. Sz. Kiss. Development of human-like agents in virtual environments. Manuscript, March 2000, University of Twente, Enschede, the Netherlands.
6. D. Lie, J. Hulstijn, R. op den Akker and A. Nijholt. A Transformational Approach to NL Understanding in Dialogue Systems. *Proceedings NLP and Industrial Applications*, Moncton, New Brunswick, August 1998, 163-168.
7. A. Nijholt, A. van Hessen and J. Hulstijn. Speech and language interaction in a (virtual) cultural theatre. *Proceedings NLP and Industrial Applications*, Moncton, New Brunswick, August 1998, 176-182.
8. A. Nijholt and J. Hulstijn. Multimodal Interactions with Agents in Virtual Worlds. Chapter in *Future Directions for Intelligent Information Systems and Information Science*, N. Kasabov (ed.), Physica-Verlag: Studies in Fuzziness and Soft Computing, 2000, to appear.
9. G. Reitmayr, S. Carroll, A. Reitemeyer and M.G. Wagner. Deep Matrix: An open technology based virtual environment system. *The Visual Computer Journal*, to appear.
10. N. Richard, P. Codognet and A. Grumbach. The InViWo virtual agents. *Eurographics '99 Conference, Short Papers and Demos, Milan, 1999*, 50-52
11. B.W. van Schooten. Process- and agent-based modeling techniques for dialogue systems and virtual environments. CTIT Report, March 2000, University of Twente, the Netherlands.
12. R. Vertegaal, R. Slagter, G. van der Veer and A. Nijholt. Why conversational agents should catch the eye. In: *Proceedings. ACM SIGCHI Conference CHI 2000 - The Future is Here*, The Hague, April 2000.
13. VRML Living Worlds Working Group proposal draft 2: Making VRML 97 Applications Interpersonal and Interoperable. URL: <http://www.vrml.org/living-worlds>, 1998.
14. VRML Humanoid Animation Working Group proposal draft 1.1, URL: <http://ece.uwaterloo.ca/~h-anim/>, 1998.