Designing and Implementing Embodied Agents Learning from Experience

Dirk Heylen
University of Twente
PO BOX 217
7500 AE Enschede, The Netherlands
heylen@cs.utwente.nl

Anton Nijholt
University of Twente
PO BOX 217
7500 AE Enschede, The Netherlands
anijholt@cs.utwente.nl

ABSTRACT
In this paper, we provide an overview of part of our experience in designing and implementing some of the embodied agents and talking faces that we have used for our research into human computer interaction. We focus on the techniques that were used and evaluate this with respect to the purpose that the agents and faces were to serve and the costs involved in producing and maintaining the software. We discuss the function of this research and development in relation to the educational programme of our graduate students.

Categories and Subject Descriptors
H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems - Animations. I.2.11 [Artificial Intelligence]: Distributed Artificial Intelligence - Intelligent agents.

General Terms
Design, Human Factors, Algorithms, Performance, Theory.

Keywords
Embodied agents, talking faces, VRML, human-computer interaction, speech and natural language processing, standards.

1. INTRODUCTION
To study various issues in the field of human-computer interaction, including multi-modality, speech and natural language, affective computing, our group has created a virtual (VRML) environment inhabited by a number of (intelligent) agents, some of which are embodied and capable of conversing face-to-face with visitors of the environment. Besides the actors, living in this habitat, we have designed and implemented other agents and talking faces, often based on characters previously build. This has provided quite a lot of experience in porting this type of software across different platforms and maintaining, modifying, updating or reengineering it.

In the remainder of this introduction, we sketch some of the aspects of our research and the choices we have made in setting up our research project that have a direct or indirect bearing on the design and implementation of our agents and the virtual environments.

Our research group is made up of researchers with quite divergent interests and backgrounds. Domains of study vary from graphics to agent technology, from neural networks and machine learning to speech and natural language processing, from information retrieval to formal specification languages. Many of the projects are part of the educational programme and therefore involve close collaboration with graduate students (mostly computer science students). These students are working on projects (multi-disciplinary design and implementation projects, masters theses, or some other kind of assignment) only for a limited amount of time, from two or three months to a year. As far as ensuring continuity of the whole enterprise is concerned, this has a number of repercussions. First of all, students need to have – or be able to acquire – some basic to advanced knowledge on human-computer interaction, natural language processing, graphics, artificial intelligence (or a combination of these subjects). Only part of this is taken care of in the computer science curriculum. Secondly, the projects should be interesting and educationally relevant for the student. For instance, a student cannot be given an assignment simply to implement some unproblematic straightforward boring piece of software or to fix the bugs in the software from another student project. Instead, the projects have to pose a challenging research or engineering problem. Thirdly, the students don't hang around for a long time after the project has finished. This means, amongst other things, that we have to put great emphasis on good software engineering practices (design, documentation, good testing, etc.) in order to make the products useful afterwards. Some more consequences of this way of organising the research and development will be given below.

The kind of virtual environment that was chosen for most of our experiments into human-agent interactions was a web-based 3D VRML world. One of the motivations behind this choice was that the environment was constructed not just for experimentation but also for demonstration purposes and should therefore be widely accessible with low to no investments in terms of additional software requirements on the part of the visitors. Choosing an internet-accessible interactive world already imposes a number of important constraints. On a general level it more or less restricts the choice of software that can be used effectively and it determines the way the different components are put together given a typical client-server architecture. Furthermore, speed and bandwidth considerations play a role in the design of agents, talking faces and what they can be made to perform realistically given the current restrictions on data transfer and download time.

We illustrate these issues by a discussion of some of the projects that we and our students have been involved in over the last couple of years.

In the next sections we will discuss some of our experience in building embodied agents, focussing on how all these different constraining factors influence the design and implementation and how the specifics of the workprocess enforces certain choices.

2. AGENTS AND TALKING FACES

In the following paragraphs we will introduce a selection of our agents and talking faces, highlighting their technical characteristics and the way they were build. We discuss some of the typical problems in constructing them and we evaluate the method used to build them in terms of effort needed and results achieved.

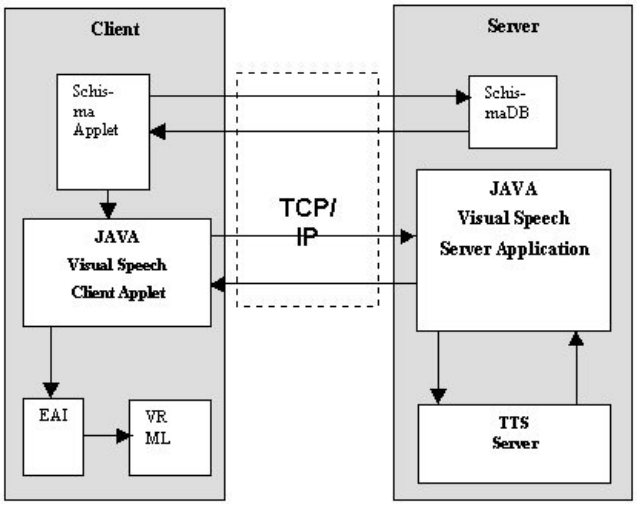
2.1 Karin

One of our first steps in studying natural interactions between users and machines led to a key-board driven, natural language information system (SCHISMA) that was able to inform users about performances in the local theatres and to make reservations ([7]). This dialogue system was subsequently taken as the basis for the natural language component of our agent, Karin.

Our main objectives in modeling Karin were to put a face and body to our dialogue system and build a basis for further research in multi-modal interactions and the combination of natural language and non-verbal communication. We therefore wanted an embodied agent living in our virtual environment that was able to speak out loud, with lip synchronisation, the responses to the requests typed in by visitors and that was capable of displaying some facial expressions accompanying the speech. The requirements were further that the repertoire of expressions be expandable, the relation between verbal dialogue acts and the non-verbal cues could be changed easily. Together with the general constraints on our undertaking mentioned in the introduction this resulted in the cartoon face for Karin ([10]) displayed below.

The face was modelled in 3D Studio. It is constructed out of simple basic forms – eyebrows, nose, lips, nose, torso – that can be manipulated separately. This provides the basis to define a repertoire of facial expressions. A sample of facial expressions was designed in 3D Studio and exported to VRML. By means of a JAVA-applet these can be called for directly but the different components can also be controlled separately by the applet to define a whole range of expressions. The eye-brows can be raised, the eyes can blink, eyes can be wide open or closed to any degree. The eyes can gaze in any direction. The nose also has some flexibility. The mouth can take on 5 different positions which gives a crude but adequate clustering of visemes (of course, more positions can be animated if needed). The shade of the head can change to make the face blush. All in all, the face is thus capable of showing quite a range of expressions, which can be called for any time during a dialogue.

There are several reasons why a cartoon face was preferred above a more realistic face. First, a detailed face would lead to a very large number of coordinates that have to be coded in the VRML files, which slows down downloading considerably. Also, animating the face can take a lot of computing power that may not be available to the average visitor of the environment. Secondly, it is quite costly to build a completely realistic face. Moreover, if realistic faces are aimed for, than they should be nearly perfect, as all technical imperfections have greater effects on the perception of the face. So besides technical difficulties there were also other reasons to abandon natural faces. With a cartoon face it is much easier to achieve qualitative results in line with what visitors expect from cartoon faces.



A second aspect of the design and implementation of Karin, besides the implementation of the face, is the distributed architecture of the virtual enviroment and more particularly the position of the dialogue modules and the text-to-speech synthesis. The figure above shows how on the client side (that of the visitor) a SCHISMA applet takes care of dialogue management. It takes care of user input (typed sentences), which it processes and tries to map (in the course of a dialogue) onto a database query for the database on performances. Processing the input is a two-step process, where first the user input is rewritten to a canonical form (by a series of rewrite rules) and then the canonical form is processed to be mapped on a query or another appropriate action (for instance a request for more information). The database contains essentially, all the knowledge Karin possesses about the world. The SCHISMA applet implements the dialogue management system and thus Karin's knowledge of language and

interaction. The output it produces – textual reply + codes for non-verbal behavior – are passed on to the Speech client applet. This filters out the text, sends it to the server, where the text-to-speech (TTS) system Fluency[1] turns the text into speech and sends it back. After the visual speech client applet receives the speech signal it takes care of synchronising instructions for speech and lip movements and sends all the information it received (from the Schisma applet and Speech Server) to the External Authoring Interface. This takes care that the animations and sounds are played in the VRML-browser.

What is particularly problematic about this set-up is the fact that the TTS-system does not run locally on the machine of the visitor. This can cause important, unnatural delays because TCP/IP connections are not always reliable or fast. This particular design choice was enforced because we cannot assume that every visitor has a TTS-system for Dutch on his computer (although this situation may be changing soon). This restriction is due to the principles of web-publishing. Most of the content published on the web is there to be freely accessible. The maker pushes it on the web and hopes to target as many visitors as possible. The visitor should not be forced, therefore to invest (financially or otherwise) in products to get access. Media players are often free of charge, authoring tools on the other hand are not. This shows again how technological restrictions influence the design characteristics and quality of the product.

Karin has been used for some experiments on multi-modal and non-verbal interaction. It has proven possible to modify and extend both the dialogues and the repertoire of non-verbal cues. Currently, for instance, we are working on Karin's gaze behaviour while conversing for which we needed extra animations and longer verbal replies. Studies on gaze behaviour in humans, like [5], have found interesting patterns in how gaze is used to control the organisation of turn-taking or to signal emotional and relational attitudes. In one research track that we are currently involved in, we investigate how these findings can be made to use in our Karin agent to regulate and influence the interaction in subtle ways. In particular, we want to implement algorithms that can simulate natural behavior on the basis of only a limited number of superficial cues. In our first serie of experiments, which we are currently engaged in, we look only at some general parameters of the dialogue state and the information structure of Karin's utterances[2]. We want to compare algorithms that take this information into account with even less knowledgeable, more robust, algorithms that make estimates of appropriate behaviour knowing only the duration of what Karin will say. In our second serie we also want to take into account the prosodic characteristics of the speech input of Karin's interlocutor.

Although, Karin's dialogue-management capabilities are rather crude, they are certainly sufficient for particular experiments (for instance, when we can more or less control what Karin is going to say). The cartoon face also offers enough expressive potential. Instructions for verbal and non-verbal output are coupled together by a basic command *tell(text-string,non-verbal-instruction)*. This works best if the text-string corresponds to a complete utterance. If complex non-verbal cues are to be synchronised (for instance a sequence of facial expressions) this does not form an unsurmountable problem because the non-verbal-instruction is not restricted to basic signals. However, very detailed synchronisation between utterance and non-verbal cues cannot be achieved. The use of the system is therefore restricted to situations where these details do not matter.

All in all, we can say that Karin is a successful implementation of an embodied agent, where success is measured in terms of the goals that we set for it. It offers adequate functionality for particular research questions and has served as a platform for education and demonstration. Because of its limitations not all experiments can be implemented in it. This is why we have build and are still designing and implementing some other agents and talking faces.

2.2 Gina

The second talking face that we want to discuss in more detail was produced by a group of graduate students as part of a software engineering assignment. In these kinds of assignments, a group of 4 to 5 students has to implement some working system together with the required software documentation in about 10 weeks. Time and resources for such projects are thus somewhat limited, though they can be sufficient to build some interesting software modules. These types of assignments have a dual function. On the one hand, students have to go through the usual stages in a software engineering process (requirements study, analysis and design, implementation, testing, documentation, etc.), on the other hand, students have to research some problem or other in computer science. The projects we propose in our group always have to do with research issues of the staff members: dialogue systems, speech, agents, robotics, neural networks, machine learning, etcetera. One of the projects we proposed this year was to build an expressive talking face that would be highly flexible and programmable by simple instructions. Whereas Karin, was originally designed to embody a particular dialogue system in a specific environment that is internet accessible, we wanted to see whether a basic talking face could be made that was specifically designed for experiments and research on non-verbal communication and that was set up in a modular way so that different components could be replaced easily.

Because the time available is rather limited and the project involves a number of complex issues, students have to make smart use of what is already available and can be reused easily.

"To improve the modularity, extensibility and reusability of Gina we decided to adhere to the component philosophy. This means that several components are created each with their own interface. This way it is easy to subdivide tasks. As long as people stick to the interface nothing can go wrong. This also means that in the future, people could decide to create other implementations of these interfaces. Or people could just use those components they need and leave the rest alone." ([12])

The system was made up out of four components: GinaFace, GinaMuscleController, GinaTTS, and GinaChat. GinaFace corresponds more or less to the graphics engine and it handles the request from the MuscleController to control the facial expressions and from the text to speech system to render the visemes. The MuscleController maps tags for non-verbal cues, i.e. facial expressions provided by GinaChat onto muscle positions. The Text-To-Speech module not merely maps the text output provided by GinaChat to speech but also to visemes which are

[1] See http://www.fluency.nl.

[2] See [13] for the role of information structure in gaze behaviour.

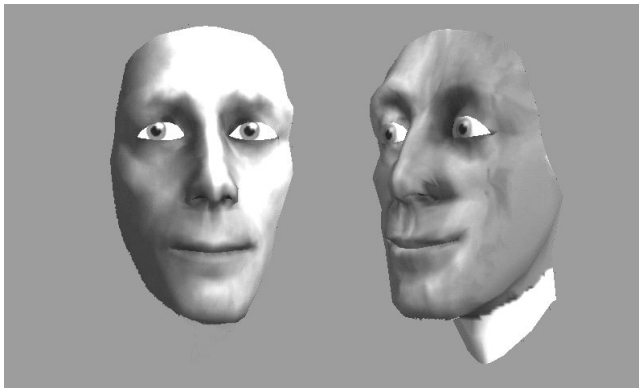
passed on to GinaFace. GinaChat can thus be identified as the dialogue manager that responds to input from the human interlocutor by means of both verbal and non-verbal responses.

What is interesting about this project, is the way existing components were incorporated. For speech, Microsoft's Speech SDK 5.0 (the system talks English not Dutch) was used. What appeared from the project is that for adequate synchronisation between different output channels, one is often forced to write software that is platform (hardware and operating system) specific. The dialogue management was largely based on existing chat-robot technology. A.L.I.C.E. bot[3] was used and adapted to provide the discourse functionality. ALICE is freely available under the terms of the GNU Public License. It utilizes AIML (Artificial Intelligence Markup Language) to form responses to questions and inputs. The fact, that such software is freely available, easy to understand and modify, and easy to fit in with other software components, makes it attractive to integrate in projects like this one.

In this project the students managed to build a complete system in quite a short period of time (though not all requirements were met in detail) using software components that were available. However, this has resulted in a system which is rather platform specific and the portability to other environments has come second place.

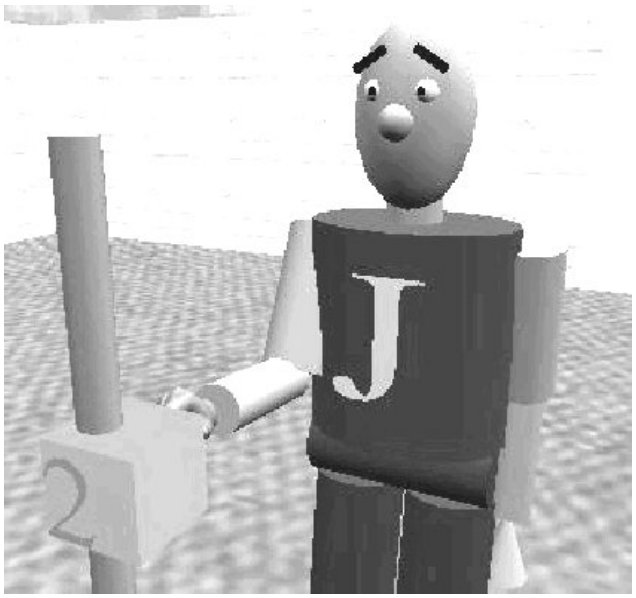
2.3 Other Agents and related research

Karin and Gina are just two of several embodied conversational agents that have been build at our department. We have chosen these two to discuss some of the issues in designing and implementation which we wanted to draw special attention to. We will now briefly introduce our other agents and discuss some further aspects.



Fred and Holy are two agents that are build to research gaze-behaviour in multi-party conversations with agents. This research is to a large part carried out at Queen's University (Kingston, Ontario) by Roel Vertegaal ([14]) in collaboration with our department ([9,15]). For the graphics and speech parts, use was made of Keith Waters's ([17]).

These faces have been used to model gaze interactions in 3 party conversations (2 agents, 1 human). Where the gaze direction of the human is monitored by means of an eye-tracker. Recently, the faces have been used in experiments to let the agents learn appropriate gaze behaviour during interactions with humans by means of association networks.



Jacob provides instruction and assistance for tasks that the user has to learn to perform ([4]). The user interacts with Jacob by performing actions as well as by using natural language. The main objectives in building this agent were (i) to research adequate task and instruction models that together make up the mind of an instruction agent, (ii) to research a specific form of multi-modal interaction, where the reader cannot just converse with the agent using several channels, but can also modify the world in specific ways.

The graphical part made use of standard components only. It was created to comply with the H-Anim standard ([16]) and is not very involved otherwise. The adequacy and genericity of the implementation of the instruction model (based on research on intelligent tutoring systems) will be put to the test in a new project that involves a tutoring agent that will teach visitors to play the piano.

Other agents include a navigation agent and an assistant agent. These were not visualised by a talking face. They were implemented to guide visitors through the virtual environment and used as part of a user study ([2,8]).

Besides work on specific agents, research in our group is concerned with specific components that are involved in building conversational agents. We would like to point out in this context, particularly the following work on dialogue systems and emotions.

Dialogue The dialogue management implemented in Karin and Gina uses robust and relatively unsophisticated techniques. Also Karin uses a rewrite system, much like the one used in ALICE, to map user utterances to normalised forms which are then processed further by the dialogue manager that looks for key phrases and concepts. Given an understanding of the standard flow of the dialogue, the system determines the state of the dialogue and the current function of the user's utterance (a request for information, an answer to a question, a confirmation, etc.) More linguistically based and complex architectures are being researched as well. Natural language parsers and grammars (for Dutch) are being

[3] See http://www.alicebot.com/.

implemented that will be used in the next generation of talking faces at our institute ([1]).

A special line of research is concerned with annotating dialogues (a lot of which were collected by Wizard of Oz techniques and also by logging the dialogue sessions with Karin and the other agents) with tags on conversational acts to be used for machine learning and to induce grammars from corpora ([3]). We hope to extend this type of work in the future by annoted corpora that include information on non-verbal cues.

Emotions Besides studying the expressions of emotions by our talking faces and otherwise embodied agents, we are also studying computational models of emotions. An architecture for a system simulating the emotional state of an agent that acts in a virtual environment was constructed ([6]). It is an implementation of an event-appraisal model of emotional behaviour by Ortony, Clore and Collins ([11]). The primary motivation was not to build a computational model of an emotional agent, but a methodology to use for building emotional agents by means of learning algorithms. The system uses neural networks to learn how the emotional state is influenced by the occurrence of internal and environmental stimuli. In current research, we are extending this work and incorporating these emotional systems as part of the minds of our embodied agents that are capable of expressing these emotions both verbally and non-verbally.

Summary In this and the previous paragraphs on embodied agents we have identified and illustrated a number of issues that arise when building embodied conversational agents in our research and education setting (which is probably not very different from most other university settings).

1. Dealing with graphics and speech, it is not always easy to implement platform independent applications.
2. Because of this dependence, also updates in software or hardware may make maintenance of systems difficult.
3. Some freeware software is available that can be plugged in to function as components to build interesting prototypes or useful systems for experimentation. It would be nice to see more of such components.
4. The use of existing (third party) software may put restrictions on the platform it is to run on or on the other software components it is to interact with. This may restrict the portability of the system.
5. Wherever feasible it is a good idea to stick to (emerging) standards, that enable software to be distributed and re-used more widely. Our experience shows that this common wisdom has beneficial effects.
6. Crude techniques can already be useful for experimentation and research though this may involve setting up clever tests that bypass the shortcomings of the system.
7. For web-based applications, fast access is important and software requirements on the client side should be kept to a minimum.

These are some of the more practical lessons we have learned from our experience in building embodied conversational agents.

3. CONCLUSIONS

In the previous sections we have discussed how, in the particular educational and research setting in which our research group operates, talking faces or animated agents have been developed by students and staff (focussing on work by the former in this paper). This does certainly not represent exhaustively the kind of research and development work that is going on in our group. The constraints that come along however with this kind of practice, provide their own specific requests for the community to deliver some basic components that need not be highly sophisticated but that can be integrated into other systems, free of charge, and that can be easily adapted to the new situation. This educational setting particularly highlights these features, but we believe that they are valuable for research purposes as well. The value of such systems as build by students or of the simple components they are made up from is that they can also be used in tandem with more complex components that implement more sophisticated or experimental theories. For instance, in the research from PhD-students and members of staff (senior researchers) more sophisticated models of natural language understanding systems, machine learning techniques or agent architectures are designed and implemented. However, this kind of research is often directed towards details. There is thus a complementary division of labour. On the one hand we have build some large systems incorporating many components (a complete virtual environment inhabited by various agents) restricted to simple, basic principles and on the other hand we have more involved subsystems that need to be integrated into a more global framework to be tested properly. The optimal situation is one in which modules or components can be freely plugged together to form complete systems. This requires agreement on the identification of components and how they work together, i.e. a specification of a type of reference architecture. Certainly the use of (emerging) standards on all different levels are helpful as well. It might further lead to implementations of some interfaces that facilitate the exchange and coupling of different modules. This will certainly make it possible to bring the research and development on embodied conversational agents some steps further.

4. REFERENCES

[1] Akker, R. op den and Nijholt, A. Dialogues with Embodied Agents in Virtual Environments. In: Proceedings 2nd International Conference on Natural Language Processing: NLP 2000: Filling the gap between theory and practice. D.N. Christodoulakis (ed.), LNAI 1835, Springer, 358-369, 2000.

[2] Akker, R. op den, J. Zwiers, B. van Dijk and A. Nijholt. Design issues for intelligent navigation and assistance agents in virtual environments. In: Proceedings Learning to Behave: Interacting Agents. TWLT 17, A. Nijholt, D. Heylen & K. Jokinen (eds.), University of Twente, Enschede, 2000.

[3] Doest, H. ter, Towards Probabilistic unification-based parsing. PhD Thesis. University of Twente, 1999.

[4] Evers, M. and Nijholt, A. Jacob - an animated instruction agent for virtual reality. In: Proceedings 3rd International Conference on Multimodal Interfaces (ICMI 2000), Beijing, Lecture Notes in Computer Science 1848, Springer, 2000.

[5] Kendon, A. *Conducting Interaction*. Cambridge University Press, 1990.

[6] Kesteren, A.-J., Op den Akker, R., Poel, M. and Nijholt, A. Simulation of emotions of agents in virtual environments using neural networks. In: Learning to Behave: Internalising

Knowledge. Proceedings Twente Workshops on Language Technology 18 (TWLT 18), 2000.

[7] Lie, D., Hulstijn, J., Op den Akker, R, and Nijholt, A. A Transformational Approach to NL Understanding in Dialogue Systems. Proceedings *NLP and Industrial Applications*, Moncton, New Brunswick, 163-168, 1998.

[8] Luin, J. van, Op den Akker, R., Nijholt, A. A dialogue agent for navigation support in virtual reality. Proceedings ACM SIGCHI Conference CHI 2001.

[9] Nijholt, A, Heylen, D. and Vertegaal, R. Inhabited interfaces: Attentive conversational agents that help. In: Proceedings 3rd International Conference on Disability, Virtual Reality and Associated Technologies - ICDVRAT2000, Alghero, Sardinia, 2000.

[10] Nijholt, A., M. van den Berk and A. van Hessen. A natural language web-based dialogue system with a talking face. Proceedings Text, Speech & Dialogue, P. Sojka et al (eds.), Brno, Czech Republic, 1998, 415-420.

[11] Ortony, A., Clore, G.L., and Collins, A. The Cognitive Structure of Emotions. Cambridge University Press, 1988.

[12] Soeteman, S., Hölzenspies, Ph., Haarmeijer, C. Suter, J., Willemsen, M. *Gina, a chat robot with facial expressions* Internal report TKI. University of Twente, 2001.

[13] Torres, O., Cassell, J. , Prevost, S. Modeling Gaze Behavior as a Function of Discourse Structure." First International Workshop on Human-Computer Conversations. Bellagio, Italy, 1997.

[14] Vertegaal, R. Look Who's Talking to Whom. PhD Thesis, University of Twente, Enschede, 1998.

[15] Vertegaal R., Slagter R., Van der Veer, G. and Nijholt, A. Why conversational agents should catch the eye. In: Proceedings ACM SIGCHI Conference CHI 2000.

[16] VRML Humanoid Animation Work Group, http://ece.uwaterloo.ca/~h-anim/, 1998.

[17] Parke, F.I. and Waters, K. *Computer Facial Animation*. A.K. Peters 1994.