International Journal of Semantic Computing
Vol. 2, No. 1 (2008) 71–90

World Scientific
www.worldscientific.com

AUTOMATIC GENERATION OF GAZE AND GESTURES FOR DIALOGUES BETWEEN EMBODIED CONVERSATIONAL AGENTS

WERNER BREITFUSS

Creative Informatics, University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-8656, Japan
werner@mi.ci.i.u-tokyo.ac.jp

HELMUT PRENDINGER

National Institute of Informatics
2-1-2 Hitotsubashi,Chiyoda-ku
Tokyo, 101-8430, Japan
helmut@nii.ac.jp

MITSURU ISHIZUKA

Creative Informatics, University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-8656, Japan
ishizuka@i.u-tokyo.ac.jp

In this paper we introduce a system that automatically adds different types of non-verbal behavior to a given dialogue script between two virtual embodied agents. It allows us to transform a dialogue in text format into an agent behavior script enriched by eye gaze and conversational gesture behavior. The agents' gaze behavior is informed by theories of human face-to-face gaze behavior. Gestures are generated based on the analysis of linguistic and contextual information of the input text. The resulting annotated dialogue script is then transformed into the Multimodal Presentation Markup Language for 3D agents (MPML3D), which controls the multi-modal behavior of animated life-like agents, including facial and body animation and synthetic speech. Using our system makes it very easy to add appropriate non-verbal behavior to a given dialogue text, a task that would otherwise be very cumbersome and time consuming. In order to test the quality of gaze generation, we conducted an empirical study. The results showed that by using our system, the naturalness of the agents' behavior was not increased when compared to randomly selected gaze behavior, but the quality of the communication between the two agents was perceived as significantly enhanced.

Keywords: Multimodal input and output interfaces; multi-modal presentation; processing of language and action patterns.

1. Introduction

Combining synthetic speech and human-like conversational behavior like gaze and gestures for virtual characters is a challenging and tedious task for human animators.

As virtual characters are used in more and more applications, such as computer games, online chat, and virtual worlds like "Second Life", the need for automatic behavior generation becomes more pressing. Thus, there have been some attempts to generate non-verbal behavior for embodied agents automatically. Systems like the Behavior Expression Animation Toolkit (BEAT) allow one to generate a behavior script for agents by just inputting text [3]. The drawback of most current systems and tools, however, is that they consider only one agent, or only suggest behaviors, such that the animator still has to select appropriate ones by him- or herself.

The aim of our work is to generate all non-verbal behavior for conversing agents automatically, so that someone writing a script to be performed by two agents can focus on creating the textual dialogue script and just feed it into the system. A salient feature of our system is that we generate the behavior not only for the speaker agent but also for the listener agent that might use backchannel behavior in response to the speaker agent. Employing two presenter agents holding a dialogue is advantageous, since watching (or interacting with) a single agent can easily become boring and it also puts pressure on users, as they are the only audience. Furthermore, two agents support richer types of interactions and "social relationships" between the interlocutors. Also TV-commercials, games, or news use two presenters, because of the increased interaction possibilities and entertainment value. The earliest use of dialogue for information presentation is presumably by Plato, where Socrates and his contemporaries use fictitious conversations to communicate Plato's philosophy. Empirical evidence exists for the claim that for learners, dialogue is a more effective communication medium than monologue [4, 5]. Comparing monologue to dialogue, [6], for instance, showed that dialogue stimulates students to write more in a free recall test and ask deeper questions in a transfer task.

In this paper, however, we confine discussion to the case where one user just watches the performance (dialogue) of two virtual agents, and does not interact with them. To assess the quality of our system we conducted an experiment. Twenty participants watched a presentation generated by our system. We randomly assigned them either to a version where the gaze behavior of the agents was informed by our gaze generator or to another version where the gaze was generated randomly. We speculated that the first (informed) version would increase the naturalness of the conversational behavior of the virtual characters and the quality of the communication between them. By "quality of the communication" we mean that the listener is paying attention to the speaker and the speaker addresses the listener in appropriate moments. In the study both versions used the same gestures, since we wanted to investigate the gaze behavior only. The dialogues were provided by a system developed at the Open University by Sandra Williams [28]. It generates a dialogue based on the medical history of a patient. While this system is designed to create shorter dialogues, for our purposes we used its original longer (unmodified) versions. The longer versions are sometimes repetitive, since patients in this database tend to have the same examinations over and over again.

This paper is organized as follows. In Sec. 2 we discuss related work. Section 3 describes our system and the way how gaze behavior and other non-verbal behavior are generated by means of a "walk through" example. In Sec. 4 we describe our empirical study on gaze generation. The results are presented and discussed in Secs. 5 and 6. Section 7 gives a short future outlook and concludes the paper.

2. Related Research

Existing character agent systems already support the automated generation of some behaviors, such as automatic lip-synchronization. The next step is to automatically generate agents' conversational behavior from text. In this section, we report on some previous attempts, which combine various disciplines like computer animation, psychology, and linguistics.

2.1. *Single agent systems*

The BEAT system [3] generates synthesized speech and synchronized non-verbal behavior for a single animated agent. It uses plain text as input, which is then transformed into animated behavior. First, text is annotated with contextual and linguistic information, based on which different (possibly conflicting) gestures are suggested. Next, the suggested behaviors are processed in a 'filtering' module that eliminates gestures that are incompatible. In the final step, a set of animations is produced that can be executed, after necessary adoptions, by an animation system. The BEAT system can handle any kind of text and generate a run-able agent script automatically. The system uses a generic knowledge base where information about certain objects and actions is stored, and the selected gestures are specified in a compositional notation defining arm trajectories and hand shapes independently, which allows the animator to add new gestures easily, or adjust existing ones. The main difference between the BEAT system and our system is that BEAT focuses mainly on generating the behavior for one agent whereas our system has the integration of speaker/listener behavior and synchronization as a core design feature. One important feature of our system is that it that there is a dependency between the speaker and listener behavior. Specifically, the listener behavior is determined according to the speaker behavior.

Although running two agents controlled by the BEAT system is possible, creating the required dependencies would involve a major change to the system. Thus, whereas the overall approach of both systems is similar — both systems take text as input and output a script that defines the behavior — the technical realization is different. Whereas the BEAT system is a straightforward pipeline where the output of the previous module provides the input to the subsequent module, our system allows for iterations and can reuse the generated output in the same module to refine it.

The PPP Persona [1] is a life-like interface agent that presents multimedia material to a user. The behavior of the agent during the presentation is controlled partly

by a script, written by the author of the presentation and partly by the agent's self-behavior. Behavior in the case of this agent is mostly acts such as pointing, speaking and expressing emotions and the automatically generated self-behavior which includes (1) idle-time actions to increase the personas life-like qualities, for example breathing or tipping a foot, (2) reactive behavior letting the agent react to external events like user reactions immediately, and (3) so-called navigation acts which display the movement of the agent across the screen, like jumping or walking. To generate this kind of behaviors a declarative specification language was used. The main difference to our system is that, except for idle-time behavior, the content author has to declare gesture and gaze behavior of PPP Persona manually.

[19] describes a system that converts Japanese text into an animated agent that synchronously gestures and speaks. For assigning an appropriate gesture to some phrase the authors employed communicative dynamism (CD) as introduced by McNeill [18] and results from an empirical study that identified lexical and syntactic information and their correlation with gesture occurrence. For every "bunsetsu", the Japanese equivalent for a phrase in English, the system adds a gesture at a certain possibility, which is derived from the results of the study and the CD value. Similar to our system the specific gestures are defined in a library and if no specific gesture can be found for the bunsetsu, a beat is added as default gesture. This system is similar to the BEAT System, and hence our system is different in the same way as it differs from the BEAT System, i.e., in the dependency of the listener on the speaker and the possibility of readjusting of agents' behavior.

2.2. *Multi agent systems*

Another system is the eShowroom demonstrator [15], which was developed as a part of the NECA Project. The application automatically generates dialogues in a car-sales setting between an agent who acts as a seller and a second agent acting as buyer. The user has the possibility to choose certain parameters like topic, the personality and the mood of the virtual characters, which control the automatically generated dialogues. Also the gestures and behavior of the two screen characters would be generated by the NECA eShowroom demonstrator. It uses three types of behavior or signals: (1) turn taking signals like looking to the other interlocutor at the end of the turn, (2) discourse functional signals, which are gestures that depend on the type of the utterance (type refers to dialogue acts like inform or request), (3) feedback gestures are also generated to signal that the listener is paying attention to the speaker. However, unlike our system, the gestures are added based on certain templates that are chosen by a dialogue planner. This approach hence lacks the flexibility in behavior generation that our system provides.

A different approach is suggested in [13]. This system supports the author in writing agent scripts by automatically generating gestures based on predefined rules, and using machine learning to create more rules from the set of predefined rules. It was used in the COHIBIT system, where the author first has to provide a script

containing the actions for two virtual characters. In the next step the author writes simple gesture rules using his or her expert knowledge. Using this corpus of annotated actions the system can learn new rules. In the third step the system suggests the most appropriate gestures to the author, which are, after resolving conflicts and filtering, added to the already existing ones. Finally it produces a script with the gestural behavior of both virtual characters. Similar to our work, two agents are used, but since we want to reduce the workload to the minimum, our system does not require any input from the author except the dialogue to be presented by our characters.

The SASO system described in [12] is a multi-agent system that enables users to train their negotiation skills with two virtual humans, who show both verbal and non-verbal behavior and have cognitive, emotional and conversational skills. The architecture of the system combines many technologies like speech recognition, natural language understanding and natural language generation to create an environment where human trainees can interact with agents to reach a common goal. In SASO the gestures of the virtual characters are controlled by the NVBG system [16] which applies rules based on theoretical foundations of movement space to select the appropriate gesture animations, postures, facial expressions, and lip synch timing for the virtual character. Similar to our system and the BEAT System it also uses a natural language parser to compute the input, and XML combined with XSLT to generate the output. The main difference from our approach is that the behavior generation system only considers one agent and hence does not generate any behavior based on the role of the agent.

2.3. *Related work on eye gaze and gestures*

[10] investigated the many different functions of gaze in conversation and its importance for the design of believable virtual characters. The gaze behavior of our agents is informed by empirically founded gaze models [11, 21, 27]. [11] analyzed gaze behavior based on two-person dialogues and found that gaze is used to regulate the exchange between the speaker and listener. In that work, different gaze patterns like the q-gaze (the speaker is looking at the person he/she is interacting with), and a-gaze (p is not looking at the interlocutor) were defined.

It was found that the speaker looks at the listener while speaking fluently, but looks away when starting to speak or during hesitation (influent speech). In this way, speakers can keep listeners' attention or, by looking away, gain time to think about what to say next. Another finding is that mutual gaze can regulate the level of emotionality between interlocutors.

The experiment described in [27] evaluates gaze behavior in multiparty environments, where four-person groups discussed current-affair topics in face-to-face meetings. Their results show that on average, interlocutors look about seven times more often at the speaker they listen to, than at others, and speakers looked about three times more at the addressed listener than at non-addressed listeners. Furthermore,

the total amount of time spent gazing at each individual in a group of three is nearly 1.5 times higher than if visual attention of the speaking person were divided by three. These results are very relevant for our gaze algorithm since they give us the basis for a 'two agents' situation and also provide the needed information for our gaze generation rules. The work in [21] developed a model of attention and interest based on gaze behavior. An embodied conversational agent may start, maintain, and end a conversation dependent on its perception of the interests of the other agents.

Other related research is [9], which introduces a behavior synthesis technique for conversational agents in order to generate expressive gestures, including a method to individualize the variability of movements using different dimensions of expression. The work described in [14] presents a gesture animation system that uses results from neurophysiologic research and generates iconic gestures from object descriptions.

3. Gesture Generation System

Our system consists of three different modules:

- Language Tagging module,
- Non-Verbal Behavior Generation module,
- Transformation to simple script or MPML3D module.

An XML-based scripting language called "Multimodal Presentation Markup Language" is used to control the behavior of our 3D agents [20]. We choose a modular pipelined architecture to support future extensions. The code of the system is written in Java, and the XML format is used to represent and exchange data between modules.

The Language Tagging module takes the input dialogue text and uses the language module from the BEAT toolkit [3] to annotate linguistic and contextual information. Next, the Behavior Generation module adds non-verbal behavior like eye gaze and gestures to the annotated input sentence. In the final step, an agent script file is produced. In our implemented system, we can produce an MPML3D file but also a simpler XML script that can be used as an interface to other systems. The MPML3D player displays the embodied characters agents.

In our system, gaze patterns are generated for two different types of roles: (1) the speaker, i.e. the agent that is speaking and addresses the other agent, and (2) the listening agent. We can currently generate gaze behavior and gestures for these two roles, based on a given set of rules. Gaze directions have certain probabilities of occurrence, which we derived from existing gaze models [11, 21, 27]. In order to avoid conflicts between certain gaze behaviors, like looking in two different directions at the same time, we assigned priorities to them. Typically, more specific gaze behaviors (such as looking at speaker/listener) have higher priority than, e.g., looking around. Moreover, we prioritize gazes that occur before starting the utterance,

```
FOR each THEMA node in the tree
  IF at the beginning of the utterance
  Or 70% of the time
    Look away from listener
FOR each RHEMA node in the tree
  IF at the end of the utterance
  Or 73% of the time
    Look at listener
```

Fig. 1. Gaze generation rule for the speaker.

i.e., speakers typically look away before starting a long utterance (in order to concentrate on planning their dialogue contribution). The rule in Fig. 1 (adapted from [3]) shows one example of how the gaze behavior for the speaker is generated.

In addition to generating the gaze behavior for the speaking agent we also have to consider the agent in the role of the listener. Since listeners typically look at speakers when they start an utterance (after taking the floor) to demonstrate their attentiveness, we developed rules like the one in Fig. 2.

We also added gaze rules for certain gestures enacted by the speaker. For instance, pointing gestures have to be accompanied by the correct gazes. In our presentation scenarios we mostly use rectangular slides in the centre between the agents and smaller objects around them. As all of those objects have a definite position either left or right to the agent, we can exploit this knowledge to add the correct gaze direction to the agents' behavior when they talk about or point at the object. However, since defining the objects' position in the scenery would increase the workload of the author, we also implemented the following straightforward principle. Every time a phrase such as "on my right side" or "to the left" occurs, we add a pointing gesture to the speaker's behavior tree. When the speaker's tree is completed, we recompile the listener's tree to adopt its gaze behavior to the pointing gestures, and add the gestures to the correct side. The gestures of our agents are generated in similar manner, broadly following rules proposed in [3].

```
FOR each THEMA node in the tree
  IF at the beginning of the utterance
  Or 80% of the time
    Look at speaker
FOR each RHEMA node in the tree
  IF at the end of the utterance
  Or 47% of the time
    Look at the speaker
```

Fig. 2. Gaze generation rule for the listener.

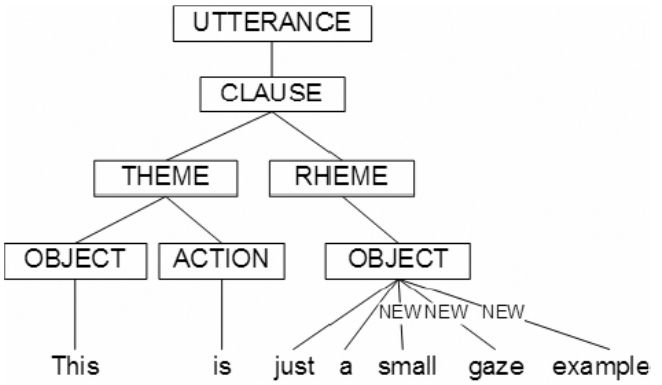

Fig. 3. The output tree of the language module.

Let us now walk through one simple example utterance and see how our system works. As input we take the sentence: "This is just a small gaze example." [2] First, the input is sent to the Language Tagger module, which annotates the sentence with linguistic and contextual tags. The output of this process is shown in Fig. 3. Here, "NEW" means that the word has not yet occurred in the conversation, and is thus a candidate for being accompanied by a "beat" gesture.

In the next step, we pass this newly constructed tree to our Behavior Generation module. It first generates a new tree with the gaze behavior and gestures (and speech parameters) for the speaker and a second tree for the listener. The tree for the listening character has the same structure as the speaker's tree, but contains the nodes for the non-verbal behavior that should be displayed by the listener agent.

Gestures are generated in two steps: first we add a beat every time some gesture is appropriate. After that the utterance is passed on to another layer that adds more specific gestures. To do this we provide a library, where we defined word bags associated with gestures. For instance, there is one word bag that contains the words "small, narrow, tiny" and the gesture for expressing something of little size. Hence, every time a word with the lemma of those words occurs in the sentence the beat gesture which has a lower priority is overwritten by the more specific gesture for small.

Figure 4 shows the speaker's tree, which was generated by our system for the sentence used in this short example. The root node of the tree is the utterance, and there is a speech pause between the theme and rheme of the sentence (see [3] for a discussion of speech parameters). The gaze behavior "Gaze away" and "Gaze at listener" is derived from the previously discussed rule (Fig. 1). The gesture behavior is generated according to dedicated gesture generation rules of the Behavior Generation module. In our example, a beat gesture is selected to accompany the word "just", and an iconic gesture (for describing something small) is suggested to co-occur with the phrase "small gaze example".

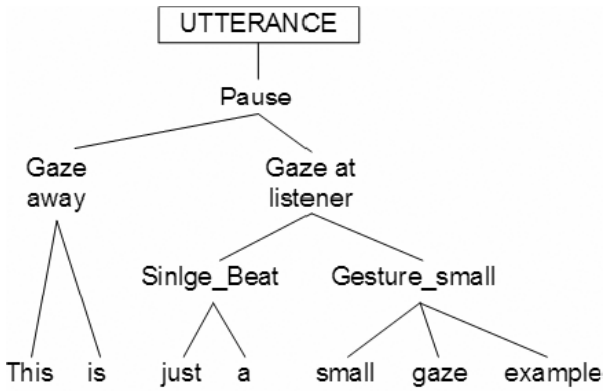

Fig. 4. Tree for the speaker behavior.

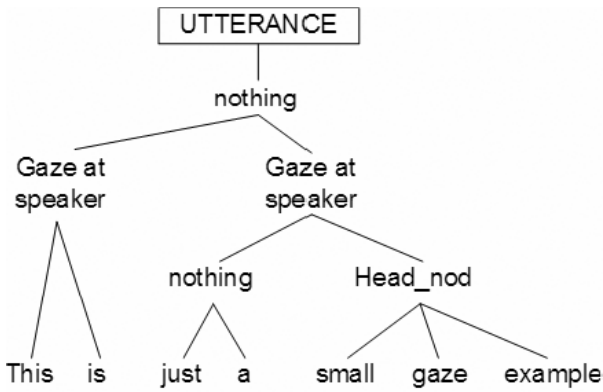

Fig. 5. Tree for the listener behavior.

The behavior tree of the listener agent is generated similarly to that of the speaker agent (see Fig. 5). It is based on the same tree that is output from the Language Tagging module of the speaker agent, but applies listener behavior generation rules instead of speaker rules. Again, we start with root node "UTTERANCE". During the speaker's speech pause, no behavior for the listener agent is defined. The listener's gaze behavior is added according to the rule in Fig. 2, i.e. the listener is looking at the speaker when the utterance begins. Accordingly, our system creates the label "Gaze at speaker". Since the listener agent is paying attention to the speaker, it continues to look at the speaker also in the "rheme part" of the utterance.

Thereafter, appropriate gestures are suggested for the listener agent. Whereas no gesture is suggested for the phrase "just a", the phrase "small gaze example" is accompanied by head nods. In our system, a head nod is a basic gesture type for the listener. It is the gesture with the lowest priority and is used when no other, more

specific gesture can be suggested. In the future, a dedicated "backchannel" knowledge base will be created to insert listener head nods in an informed, systematic manner.

After speaker and listener behavior trees are created, they are passed to the Transformation module, which compiles them into a synchronized MPML3D XML file or a simpler XML file.

Before generating the MPML3D Script we have to run the two trees through a small set of filters to handle any unexpected mistakes and to make sure no errors were forwarded to the script. We also use the filters to avoid minor technical problems like certain timing issues. Currently, the MPML3D Player cannot synchronize gestures that start at the beginning of a word and stop at the end of the same word.

This last module combines the speaker and listener tree by adding the actions of both agents for every utterance into one MPML3D structure called "task". The MPML Script contains parallel and synchronized actions which can be started and ended at the beginning, middle, or end of a certain word. First we add all the actions that should occur before the speaker starts to talk, mostly gaze behavior, like looking away from the speaker and idle gestures for the listener.

The next action that is added is speaking itself. In the following step, we add the gaze behavior, which has to be aligned with the appropriate words. Gaze is implemented by having the head turn to a certain direction. There is a set of parameters that can be used, like the vertical angle in which the head should be moved and the speed of the movement. As the last level we add the gesture for the speaking agent and the listening agent.

Figure 6 shows the MPML3D code, which our system generated for the sentence used in the example.

Our System can also produce a simpler script as output (see Fig. 7). It contains only 3 entries: (1) the text of the utterance; (2) a mood, which is generated by using the system described in [24], allowing a virtual character to display emotions; (3) the gesture with the highest priority.

The simple script is intended to be used for other agent systems, which can only display one gesture per utterance, or are limited with respect to gesture and speech synchronization, such as the agents in the Second Life virtual world.

Figure 8 shows our agents performing the example sentence.

4. Method

4.1. *Design*

In this study, we wanted to test two hypotheses: (1) our gaze model increases the naturalness of the presentation, and (2) it increases the perceived quality of the conversational behavior between the two agents.

Hence, in this study, we compared two different versions of a presentation. In one version, gaze behavior was generated by our system (the informed version). In the control version (control condition), gaze was generated in a random manner

```
<Task>
    <Action>ken.turnHead(20,0.2,0.3,0.2)</Action>
  <Parallel>
    <Action name="kenspeak">
    ken.speak("This is just a small gaze example")
    </Action>
    <Action startOn="kenspeak[0].begin"
            stopOn="kenspeak[9].end">
    ken.turnHead(20,0.2,1,0.2)
    </Action>
    <Action startOn="kenspeak[10].begin"
            stopOn="kenspeak[20].end">
    ken.turnHead(0,0.2,5,0.2)</Action>
    <Action startOn="kenspeak[0].begin"
            stopOn="kenspeak[20].end">
    yuuki.turnHead(0,0.2,1,0.2)
    </Action>
    <Action startOn="kenspeak[10].begin">
    ken.gesture("beat_one")
    </Action>
    <Action startOn="kenspeak[23].begin">
    ken.gesture("showsmallvertical")
    </Action>
    <Action startOn="kenspeak[14].begin">
    yuuki.gesture("headnod")
    </Action>
  </Parallel>
</Task>
```

Fig. 6. The MPML3D code for our example.

```
<utterance>
  <text> This is just a small gaze example</text>
  <mood>neutral</mood>
  <gesture>showsmallvertical</gesture>
</utterance>
```

Fig. 7. Simple XML code for our example.

(uninformed version). By "random" we mean that every time our system suggested a particular gaze behavior, a gaze direction was randomly chosen, which could be "look away" (to the left or to the right) or "look at the other agent", whereas in the informed version the gaze was chosen based on our predefined rules. We chose

Fig. 8. MPML3D Agents enacting the example sentence.

to use a random gaze condition based on previous experiments done in this field of research (see [8, 25, 26]).

The gestures used were the same in both versions, and consisted mostly of beats in the case of speaking character, and head nodes in the case of listening agent. We kept the set of the gestures used very limited, since as suggested in [5] too many gestures can distract the user and consequently have a negative effect on the perception of the overall presentation and gaze behavior.

4.2. *Participants*

Twenty people participated in the study, 18 males and 2 females, their age range from 22 to 35 years (mean age 28.3 years). Except for two external people, subjects were students or researchers from the National Institute of Informatics, Tokyo. Subjects received 1000 yen for participating.

4.3. *Materials*

The raw dialogues for the presentation were provided by an automated dialogue generation system [28], and contain the conversation between Yuuki, a female senior nurse and Ken, a male junior nurse.

The dialogue contained 106 utterances, and the duration of the presentation was around 5 minutes. The topic of the dialogue was about the medical history of a fictional patent that has breast cancer.

The following is a typical paragraph of the presentation. We wish to note again that for the purpose of the experiment (investigating gaze), we used the long, unmodified dialogue output by the system. This output was not meant to be shown to subjects when investigating, e.g., the effectiveness of the dialogue.

Yuuki: For May the 24th what does the medical record say?
Ken: On May the 24th she did a self examination.
Yuuki: What did she find?
Ken: A lump.
Yuuki: What does it say next?
Ken: On May the 19th she did another self examination.
Ken: And she still had a lump.
Yuuki: And then?
Ken: On June the 7th she did another self examination.
Ken: And she still had a lump.
Ken: From May the 20th to August the 5th she had a chemotherapy course.
Ken: What is a chemotherapy course?
Yuuki: A chemotherapy course is a treatment with drugs.
Yuuki: Is that clear?
Ken: Uhhuh.
Yuuki: What does it say next?
Ken: On June the 24th she had another examination.
Ken: And she still had lymphadenopathy.

4.4. *Apparatus*

The experiment was run on a Dell workstation with a dual-core processor. The material was presented to the subjects using a UXGA (1600×1200 pixels) flat screen color monitor. The speech for the agents was generated by Loquendo ([17]), a commercial text-to-speech (TTS) engine. The agents were controlled by our MPML3D Player ([20]).

For videotaping the participants we used a digital camera that was positioned behind subjects and a mirror, which was fixed on the right side of the monitor, so that we could capture the face and the shoulders of the subjects. Figure 9 depicts the setup of our study.

4.5. *Procedure*

Subjects entered the experiment room individually and received a written instruction about the procedure. The instruction given to the subjects was to watch the presentation as they would watch a presentation given by human presenters and they should keep an eye on the behavior of the agents.

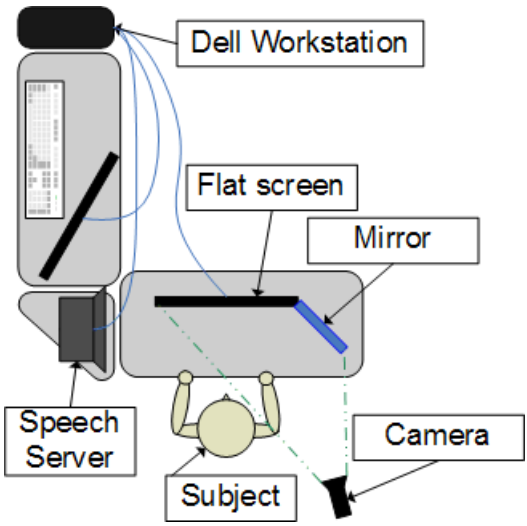

Fig. 9. Experimental setup.

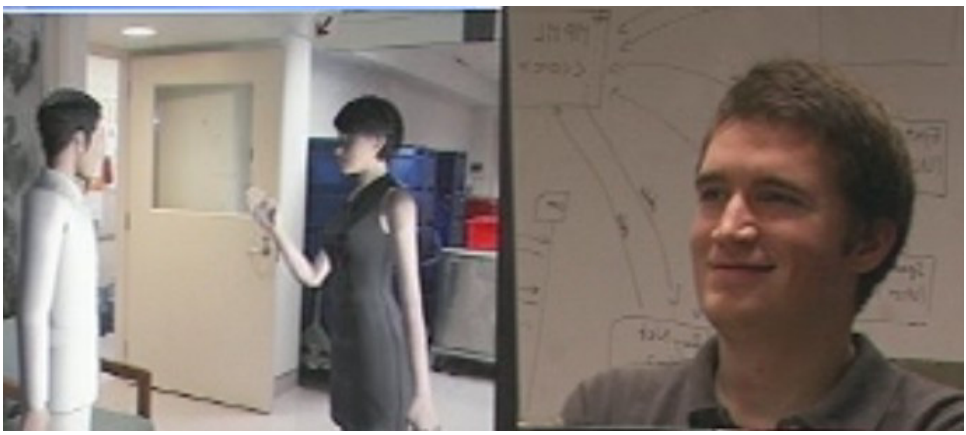

Fig. 10. Screen and participant.

While watching the dialogue between our two agents, the participants were videotaped for further analysis (Fig. 10: screen with presentation to the left, participant to the right). After watching the presentation, both groups of participants were asked to fill out a questionnaire with twelve questions.

1. The female agent (Yuuki) was friendly.
2. The male agent (Ken) was friendly.
3. The conversation between the two agents seemed very natural.
4. Sometimes I thought the agents react to each other in a strange way.

5. I felt that the two agents are a good team and communicate with each other well.
6. It seemed that the agents did NOT pay attention to each other.
7. I trusted the female agent (Yuuki).
8. I trusted the male agent (Ken).
9. I found the conversation easy to follow.
10. The conversation captured my attention.
11. I found that my attention wandered.
12. I found the conversation hard to understand.

The questionnaire consists of two types of questions. The type relating to the appearance of the agents and the communication between them was derived from a previous user study [7], using the same agents in a different context. The other type (Questions 1, 2, 7, 8) were not in the focus of this study, but were intended as "frame" questions only.

The answers were based on a Likert scale, and range from one ("strongly agree") to seven ("strongly disagree"). At the end of the questionnaire we also provided the possibility of free text entry, so that subjects could state their comments without restrictions. Each session of the experiment lasted around 15 minutes per person, and was conducted in our multimedia room.

5. Results

We performed a t-test[a] (two-tailed) to determine the statistical significance of the differences between the averages (significance level α set to 0.05). The averages of the answers to the questions in the questionnaire can be found in Fig. 11 where the x-axis gives the number of the question, and the y-axis shows the value for each question.

Figure 12 shows the means and standard deviations of the questions Q1 to Q12, where the first row gives the values, mean and deviation, of the uninformed version and the second row gives the values for the informed version.

We predicted that the gaze behavior generated by our system would generate a more natural dialogue and the agents would be perceived as communicating well with each other.

Regarding the first dimension (naturalness), we partly obtained significant results, while, surprisingly, the tendency in the answer is contrary to our expectation (Questions 3 and 4). The results for the question concerning the naturalness of the agents' behavior, the results for Question 3 showed that the uninformed version is perceived slightly more natural than the informed version. The result for

[a]The t-test tells us how likely it is that the means of the two populations are equal based on actual distance between the means and the within group variability of the two groups. The magnitude of $|t|$ increases as the distance between the means increases and the within-group variability decreases. As $|t|$ increases, the probability of the means being equal, decreases.

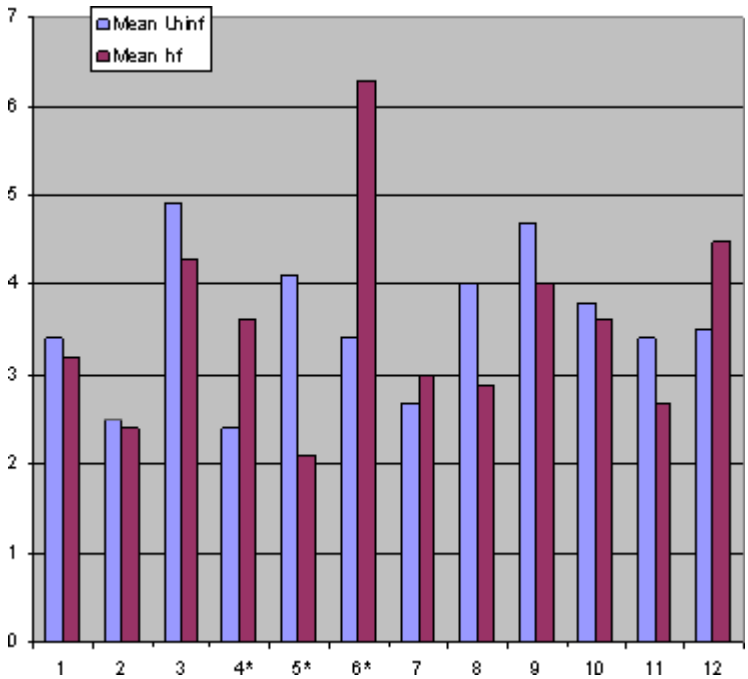

Fig. 11. The means for the questions.

Question nr.:	Q1	Q2	Q3	Q4 *
mean and deviation uniformed version	3.4 ± 1.1	2.5 ± 0.7	4.9 ± 1.4	2.4 ± 1.1
mean and deviation informed version	3.2 ± 1.4	2.4 ± 0.8	4.3 ± 1.6	3.6 ± 1.3
Question nr.:	Q5 *	Q6 *	Q7	Q8
mean and deviation uniformed version	4.1 ± 1.4	3.4 ± 1.4	2.7 ± 1.3	4 ± 1.4
mean and deviation informed version	2.1 ± 0.7	6.3 ± 0.5	3 ± 1.3	2.9 ± 0.9
Question nr.:	Q9	Q10	Q11	Q12
mean and deviation uniformed version	4.7 ± 1.6	3.8 ± 1.4	3.4 ± 1.3	3.5 ± 1.4
mean and deviation informed version	4 ± 1.8	3.6 ± 1.8	2.7 ± 1.3	4.5 ± 1.6

Fig. 12. Means and standard deviations.

Question 4 showed that the agents reacted significantly less strange (by contraposition, more natural) to each other in the uninformed version ($p < 0.05$).

The results for questions concerning the conversational behavior between the agents (quality of communication) are statistically significant. The results confirm the hypothesis that our system can significantly increase the level of perceived quality of conversational behavior between the two interlocutors For Question 5, $p < 0.01$, and for Question 6, $p < 0.0001$.

The questions regarding the friendliness of the agents (Questions 1 and 2), or about the trustworthiness of the agents (Questions 7 and 8), did not yield any significant results. Note, however, that the results for Question 8 indicate that the male character was nearly significantly ($p = 0.053$) more trustworthy in the version informed than in the uninformed version.

6. Discussion

The purpose of the experiment was to obtain empirical data on our newly implemented system, with a focus on the gaze behavior of the agents. The data from the questionnaires supports our expectations that the version with gaze behavior informed by our system would outperform the version with randomized gaze in terms of the quality of conversational behavior between the two embodied virtual characters. In particular, the result for Question 6 provides strong evidence that the participants noticed that the agents pay more attention to each other in the informed version.

The poor results, especially question three with an average of 4.3 for the informed version and 4.9 for the random gaze version, regarding the overall naturalness of the presented dialogues were somewhat surprising.

The free-text comments we received from the participants (as part of the questionnaire) gave three different reasons why they rated the naturalness as rather poor. One issue was the beat gesture, which seemed to be irritating, and the hand movement was too fast and too wide. A second problem was the voice generation, which did not produce satisfying results for technical medical terms. (In fact, this problem could have been avoided if we had provided the correct pronunciation of rare technical terms to the TTS engine beforehand.) Third, some subjects criticized parts of the dialogue as unnatural. They noted that there are too many repetitions and some of the answers given by the junior nurse (Ken) were irritating. There is one particular part in the dialogue, where the senior nurse explains the function of auxiliary lymph nodes, and the junior nurse answers with a short "Cool". As the video analysis showed, most participants found this part rather humorous, but others stated in their comments, that it is strange to use the word "cool" in the context of cancer. In order to rule out content as a confounding factor, we will be carefully in choosing dialogues in future studies.

The experiences with our study provide highly valuable insights for designing better studies with our non-verbal behavior generation system in the future.

A possible next step would be to compare this system with others that generate eye gaze (like [13] and [16]) to see whether our system can outperform others or not.

7. Conclusion and Future Work

There is ample evidence that agent-based multimodal presentations can entertain and engage the user, and are also an effective way to mediate information [23]. In this paper, we described our system that automatically generates gaze and gestures for two agents, in the roles of speaker and listener. It uses a dialogue script as its only input (from the content creator), and transforms it into a run-able multimodal presentation using two highly realistic 3D character agents.

In our future work, we plan to analyze the emotional content of text based on the work described in [24], and add emotional expressions to the agents' behavior in order to improve the naturalness of the performed dialogue. The emotion expressed in a sentence will also affect voice parameters, gaze, and gesture behavior. Conversational behavior is also influenced by the social role (instructor-student, employer-employee, etc.), the cultural background, and the personality of the interlocutors. Another venue of research relates to including a model of the user as a listener, who might be addressed by the agents.

Our next step, however, will address more feasible issues. In addition to extending the set of behavior generation rules for the listener agent, we want to align the behavior of the agents with respect to a slide show and virtual objects in a 3D environment. Here, we have to analyze phrases like "if you look at the slide" and generate appropriate behavior for the speaker and listener agent. Among others, the selected gaze behavior has to be timed and directed to specific locations in the 3D environment. In this way, "joint attention" (gaze) behavior will be implemented.

For all of our ideas, the focus will remain on the exploration of ideas that ultimately lead to a minimal workload for content creators, while ensuring high-quality, professional output in the form of natural and enjoyable multimodal presentations.

References

[1] E. André, J. Müller and T. Rist: The PPP Persona: A multipurpose animated presentation agent, in T. Catarci, M. F. Costabile, S. Levialdi and G. Santucci (eds.), *Advanced Visual Interfaces*, ACM Press, 1996, pp. 245–247.

[2] W. Breitfuss, H. Prendinger and M. Ishizuka, Automated generation of non-verbal behavior for virtual embodied characters, in *Proceedings of the 9th International Conference on Multimodal Interfaces*, 2007, pp. 199–202.

[3] J. Cassell, H. Vilhjálmsson and T. Bickmore, BEAT: The behavior expression animation toolkit, in *Proceedings of SIGGRAPH 2001*, 2001, pp. 477–486.

[4] R. Cox, J. McKendree, R. Tobin, J. Lee and T. Mayes, Vicarious learning from dialogue and discourse: A controlled comparison, *Instructional Science* **27** (1999) 431–458.

[5] S. Craig, B. Gholson and D. Driscoll, Animated pedagogical agents in multimedia educational environments: Effects of agent properties, picture features, and redundance, *Journal of Educational Psychology* **94**(2) (2002) 428–434.

[6] S. Craig, B. Gholson, M. Ventura, A. Graesser and the Tutoring Research Group, Overhearing dialogues and monologues in virtual tutoring sessions: Effects on questioning and vicarious learning, *International Journal of Artificial Intelligence in Education* **11** (2000) 242–253.

[7] T. Eichner, H. Prendinger, E. André and M. Ishizuka, Attentive presentation agents, in *Proceedings of the 7th International Conference on Intelligent Virtual Agents (IVA'07)*, Springer, 2007, pp. 283–295.

[8] M. Garau, M. Slater, S. Bee and M. A. Sasse, The impact of eye gaze on communication using humanoid avatars, in *Proceedings of the SIGCHI Conference on Human factors in Computing Systems (CHI'01)*, ACM Press, 2001, pp. 309–316.

[9] B. Hartmann, M. Mancini, S. Buisine and C. Pelachaud, Design and evaluation of expressive gesture synthesis for embodied conversational agents, in *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, ACM Press, 2005.

[10] D. Heylen, Head gestures, gaze and the principles of conversational structure, *International Journal of Humanoid Robotics* **3**(3) (2006) 241–226.

[11] A. Kendon, Some functions of gaze-direction in social interaction, *Acta Psychologica* **26** (1967) 22–63.

[12] P. Kenny, A. Hartholt, J. Gratch, D. Traum, S. Marsella and B. Swartout, The more the merrier: Multi-party negotiation with virtual humans, in *Proceedings of the 22nd AAAI Conference on Artificial Intelligence*, AAAI Press, 2007, pp. 1970–1971.

[13] M. Kipp, Creativity meets automation: Combining nonverbal action authoring with rules and machine learning, in *Proceedings of the 6th International Conference on Intelligent Virtual Agents (IVA 2006)*, Springer, 2006, pp. 217–242.

[14] S. Kopp, P. Tepper and J. Cassell, Towards integrated microplanning of language and iconic gesture for multimodal output, in *Proceedings of the International Conference on Multimodal Interfaces 2004*, ACM Press, 2004, pp. 97–104.

[15] B. Krenn, M. Grice, P. Piwek, M. Schroeder, M. Klesen, S. Baumann, H. Pirker, K. van Deemter and E. Gstrein, Generation of multi-modal dialogue for net environments, in *Proceedings of KONVENS-02*, 2002, pp. 91–98.

[16] J. Lee and S. Marsella, Nonverbal behavior generator for embodied conversational agents, in *Proceedings of the 6th International Conference on Intelligent Virtual Agents*, 2006, pp. 243–255.

[17] Loquendo Vocal Technologies and Services, URL (2008), www.loquendo.com.

[18] D. McNeill, *Hand and Mind: What Gestures Reveal about Thought*, The University of Chicago Press (1992).

[19] Y. Nakano, M. Okamoto, D. Kawahara, Q. Li and T. Nishida, Converting text into agent animations: Assigning gestures to text, *Journal of Humanoid Robotics* **3**(3) (2006) 241–26.

[20] M. Nischt, H. Prendinger, E. André and M. Ishizuka, MPML3D: A reactive framework for the Multimodal Presentation Markup Language, in *Proceedings of the 6th International Conference on Intelligent Virtual Agents*, Springer, 2006, pp. 218–229.

[21] C. Peters, C. Pelachaud, E. Bevacqua and M. Mancini, A model of attention and interest using gaze behavior, in *Proceedings of the 5th International Conference on Intelligent Virtual Agents*, 2005, pp. 229–240.

[22] H. Prendinger and M. Ishizuka (eds.), *Life-Like Characters. Tools, Affective Functions, and Applications. Cognitive Technologies*, Springer, Berlin-Heidelberg, 2004.

[23] T. Rist, E. André, S. Baldes, P. Gebhard, M. Klesen, M. Kipp, P. Rist and M. Schmitt, A review of the development of embodied presentation agents and their application fields, in Prendinger and Ishizuka [22], pp. 377–404.

[24] M. Shaikh, H. Prendinger and M. Ishizuka, A cognitively based approach to affect sensing from text, in *Proceedings of 10th International Conference on Intelligent User Interfaces*, ACM Press, 2006, pp. 349–351.

[25] I. van Es, D. Heylen, B. van Dijk and A. Nijholt, Gaze behavior of talking faces makes a difference, in *Conference on Human Factors in Computing Systems (CHI'02)*, Extended Abstracts, ACM Press, 2002, pp. 734–735.

[26] R. Vertegaal and Y. Ding, Explaining of eye gaze on mediated group conversations: Amount or synchronization, in *Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work*, ACM Press, 2002, pp. 41–48.

[27] R. Vertegaal, I. Weevers, C. Sohn and C. Cheung, Gaze-2: Conveying eye contact in group video conferencing using eye-controlled camera direction, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'03)*, ACM Press, 2003, pp. 521–528.

[28] S. Williams, P. Piwek, and R. Power, Generating monologue and dialogue to present personalised medical information to patients, in *Proceedings of the 11th European Workshop on Natural Language Generation*, 2007, pp. 167–170.