New Generation Computing, 24(2006)97-128
Ohmsha, Ltd. and Springer

Invited Paper

Describing and Generating Multimodal Contents Featuring Affective Lifelike Agents with MPML

Mitsuru ISHIZUKA
School of Information Science and Technology, The University of Tokyo, Japan

Helmut PRENDINGER
National Institute of Informatics, Japan

Received 30 November 2005

Abstract In this paper, we provide an overview of our research on multimodal media and contents using embodied lifelike agents. In particular we describe our research centered on MPML (Multimodal Presentation Markup Language). MPML allows people to write and produce multimodal contents easily, and serves as a core for integrating various components and functionalities important for multimodal media. To demonstrate the benefits and usability of MPML in a variety of environments including animated Web, 3D VRML space, mobile phones, and the physical world with a humanoid robot, several versions of MPML have been developed while keeping its basic format. Since emotional behavior of the agent is an important factor for making agents lifelike and for being accepted by people as an attractive and friendly human-computer interaction style, emotion-related functions have been emphasized in MPML. In order to alleviate the workload of authoring the contents, it is also required to endow the agents with a certain level of autonomy. We show some of our approaches towards this end.

Keywords: Lifelike Agent, Multimodal Contents, Content Description Language, Emotion, Affective Computing.

§1 Introduction

Embodied lifelike agents are emerging in multimodal interfaces and in new multimodal media contents.[9,52~55,60,67] They allow for a natural way of information presentation and interaction with humans through their multimodal expressions, such as speech, body gestures and facial expressions in addition to existing media components such as texts, graphics, images and video clips. They act as friendly and intelligent guides, tutors, stewards, partners, etc. in complex

information worlds. Embodiment of the agents provides effective means of imitating human skills such as presenting information or engaging in a conversation.

Through leading researches and developments in the last decade, their positive effect has been demonstrated and recognized, and some component technologies have become available. As seen in computer games, current media technologies allow for the creation of attractive interactive multimedia contents if professional creators devote their efforts and time. Currently, however, it is rather cumbersome for ordinary people to produce attractive multimodal contents with lifelike agents. Most of the systems so far have been constructed as individual systems employing different ways of content description, character systems, and so on.

There are the following findings regarding the cognitive effect and significance of lifelike agents.

- Albert Mehrabian's study in the 1960s on the role and significance of nonverbal communication.[31] (He unveiled that nonverbal components convey more than 50% of the information in our daily communication.)
- The idea of the so-called *Media Equation*, that is, "media=real life", reported in Byron Reeves and Clifford Nass' book in the 1990s.[59] (Human beings naturally tend to conceive an interacting media object as a "social actor", i.e. they apply social interaction protocols, rather than treating it as a "lifeless" artifact.)
- The *Persona Effect*.[24] (A lifelike agent when integrated to, for example, educational media contents has the effect of enhancing the engagement and motivation of the interacting user.)

To let a lifelike agent become a truly friendly and intelligent partner in our information space, further studies are required from several viewpoints. In this paper, we will introduce our research and developments, which have been centered on a description language called MPML (Multimodal Presentation Markup Language),[12,13,50,54,61,69,71] a language for creating multimodal contents with lifelike character agents.

The aim of MPML is twofold, that is, 1) to provide an easy-to-use tool for non-expert users to produce and distribute attractive multimodal contents with lifelike agents, and 2) to develop a technological core for integrating various components and functionalities required for a multimodal system. Whilst multimodal presentations (including some interactions) are one of the main target applications, they tend to be monotone. As a consequence, presentations may become boring when they last for a long time, say, more than 5 or 10 minutes. Thus affective functions, especially emotion-related functions of lifelike agents, are incorporated into our system to avoid such a monotonous presentation or interaction, and to increase the agents' lifelikeness and believability. MPML is an XML-based language and not restricted to a particular character system. Yet, in order to adapt MPML to various environments, such as Web contents, a 3D environment, a mobile-phone environment, and the physical world with a humanoid robot, several versions of MPML have been developed while keeping its basic style.

§2 Overview of MPML (Multimodal Presentation Markup Language)

Facing the explosion of Web contents written in HTML (and recently XML), we have been considering that, in order to spread multimodal contents in the information space, it is of key importance to provide many people with an easy-to-use tool for authoring and producing contents, and to establish a standardized content description scheme for allowing wide distribution. Along this direction, there have been some developments of description languages for multimodal contents with character agents world-wide. Some of them, including our MPML, are designed on the basis of XML (Extensible Markup Language), since the adoption of XML is beneficial for use with Web browsers and in combination with available Web-related functions. Other languages besides our MPML include VHML,[28] CML/AML,[3] APML,[11] RRL-NECA,[44] BEAT,[10] PAR,[4] STEP,[18] and so on. The differences among these languages are their description levels, targeted users (professional creators or ordinary people), character agents being supported for use, functions related to dialogue, intelligence and emotion, and interaction with an environment other than the character agents themselves (such as the background of a presentation).

Standardization is preferable, though its discussion has not been progressing well. There are basically two different approaches towards standardization, depending mainly on targeted users; one side emphasizes intricate functionalities so that even professional creators can produce their contents satisfactorily, whereas the other side emphasizes the ease and convenience of use for many non-professional users. The choice of a character agent system often reflects the adopted approach. The former approach allows for a low and fine level control of the character agent, such as a parameterized description for body gesture, facial animation, speech, and so on. The latter approach usually provides a more abstract level of control over the character agent. A typical character agent system employed within the latter approach is the Microsoft Agent package[32] or similar ones, where the gesture behavior of the agent is selected from a set of pre-defined patterns (normally 30-50 patterns). While the two approaches could in principle be merged in the future, by incorporating the characteristics of the other side, there are presently no concrete efforts in that direction.

MPML, which advocates the latter approach, is a medium-level description language allowing ordinary people to write multimodal presentation contents easily. It has been designed by considering the following requirements.

- *Ease of use*. Presentations featuring animated character agents should be easy to write and not assume advanced programming skills, and should thus be similar to writing Web contents with HTML.
- *Intelligibility*. The tags of the markup language should provide names and abbreviations that clearly indicate their meaning (semantics).
- *Extensibility*. The markup language should support authors, if necessary, in specifying new functionalities. The language should also allow for the

possibility of future extensions, such as incorporating new tags.
- *Easy distribution.* The described content should be easy to install and play on popular platforms.

Since MPML is an XML-based markup language, its description is intuitive and easily understandable. Tags in MPML are designed following the conventions well known from HTML, so that they are easy to learn and remember. (When using the graphical editor for MPML, a user does not need to care about the tag names or even the MPML description.) MPML is extensible because functions not supported by MPML can be described in JavaScript when necessary, and added to the MPML description. Although the basic Web version of MPML content can be played with the popular Internet Explorer (version 5.5 or higher) from Microsoft, there is a restriction regarding the use of the character agent system. The use of the Microsoft Agent package[32] is assumed by default; other character agent systems can be used with appropriate driver programs, which translate MPML commands into the corresponding commands of each individual agent system.

As the main target application area of MPML is (Web-based) multimodal presentations, our language has incorporated a subset of SMIL[60] for multimedia synchronization, and control functions of presentation materials (typically HTML files) and interaction. In order to enable affective and attractive contents while avoiding monotonous or flat agent presentations, MPML supports the authoring of emotional behaviors of agents.

§3 Tags in MPML

In this section, we will briefly introduce the tags being used in MPML. The top-most tag pair of MPML descriptions is `<MPML>` and `</MPML>`. Other major tags can be classified into four categories at present: agent action tags, presentation control tags, interaction tags, and emotion-related tags.

Agent action tags

The agent can perform several types of actions. A tag used to make the agents act will be called *action tag*. The available action tags as defined in current MPML are as follows.

Play: The agent can play an animation, such as "greet", "congratulate", "acknowledge", "decline", "confused", "alert", "think", "happy", "surprised", "anger", "point left", "point right", "look left", and "look right", among others. These body and facial gestures effectively add meaning to the presentation. When using this tag type, the author should know what the agent he/she is using is capable of doing: actions undefined for a particular agent cannot be used. The following example describes one tagging structure.

```
<PLAY agent="rocky" act="greet" />
```

Most agent characters including Microsoft agent characters will typically have about 30-50 pre-defined actions, which fall under one of the following categories:

- Locomotion: "moveup" (character performs a step), "moveleft", "show" (character appears), "hide" (character disappears), among others.
- Deictic gestures: "gestureright" (character points to the right by hand gesture), "gestureleft", and so on.
- Gestures realizing communicative functions: "acknowledge" (nod for giving feedback), and so on.
- Propositional gestures: For example, the sequence of "thisthatfirst" (left-hand stroke) and "thisthatsecond" (right-hand stroke) may represent a contrastive gesture corresponding a sentence 'on the one hand ··· on the other hand'.
- Emotional gestures: "sad" (character displays sad face and hanging shoulders), "pleased" (big smile), "anger" (character displays angry face and stems hands on hips), "fear" (character lifts arms), among others.

For some agent characters, 'idle' gestures are defined, such as looking left or right, folding arms, and so on. These idle gestures will be randomly invoked for periods where no specific actions are performed, as a way of increasing the lifelikeness of the character agents.

Move: The agent moves on the screen to the destination spot.

```
<MOVE agent="rocky" spot="spot3" />
```

In this case, the value "spot3" of the spot argument is defined beforehand; otherwise, a direct specification of the destination spot is possible using x and y arguments instead, for instance, as `{x="240" y="150"}`. The "agent" specification in the above tags can be omitted when it is obvious, for example, only one agent is assigned to a scene (as will be described later).

Speak: The agent speaks one or several sentences through a TTS (text-to-speech) system. The text to be spoken will, if necessary, be displayed in a balloon appearing adjacent to the agent.

```
<SPEAK agent="rocky">
 My name is Rocky. I am glad to meet you today.
</SPEAK>
```

When the text in the `<SPEAK>` tag is modified by an `<EMOTION>` tag (as described later), its speech parameters are set accordingly to synthesize an affective voice. Some actions can also co-occur with speech, or occur before or after speech, when a strong emotion arises.

Think: Unlike "speak", the agent does not speak aloud in this case. Instead, the text will appear in the balloon only. This is useful, for example, when several agents would like to 'say' something at the same time. Indeed using the `<THINK>` tag, several balloons can appear and only one agent will be speaking, keeping the voice understandable, instead of having everyone speak at the same time.

```
<THINK agent="rocky">
 I hope Merlin will soon leave so that I can be alone.
```

```
</THINK>
```

Listen and Heard: In order to accept speech input from a human user, the tags below are provided, which are used as follows.

```
<LISTEN agent = "rocky">
 <HEARD value = "* hit *">
  ...
 </HEARD>
 <HEARD value = "* stand *">
  ...
 </HEARD>
 </LISTEN>
```

Here "*" indicates a wild card matching any word. The <LISTEN> tag lets the agent enter into the listening mode. If a speech input text recognized by the speech recognizer matches with a value specified in a <HEARD> tag, then the actions described inside the <HEARD> tag are executed. A flexible description scheme allowing for a wide range of matching types is provided for the value attribute in the <HEARD> tag, though we cannot show all its details here. If no match is found for a certain period of time, then the system closes the listening mode and proceeds to the next step with indicating the unsuccessful closing of the listening mode. While the <HEARD> tag typically assumes an input text from a speech recognizer, it can be replaced by an input text obtained through an input text box or selection buttons.

Presentation control tags

The presentation is organized like a theatre play with different scenes, which is useful to separate a long presentation into several parts. Scenes are also used as references denoting assembled pieces of presentations. In each scene, pieces of presentations or agents' actions can be performed sequentially or in parallel.

Scene: This defines a scene with its background. Within the scope of this tag, a set of presentation pieces is defined.

```
<SCENE id="introduction" agent="rocky">
 ...
</SCENE>
```

(The "id" argument here can be omitted if unnecessary.)

Seq: The tag <SEQ> is used to define a sequential presentation. Within the start tag <SEQ> and the end tag </SEQ>, the order will be taken one after another. We also indicate which agents will be used as part of this tag.

```
<SEQ agents="rocky, merlin">
 ...
</SEQ>
```

Par: This tag defines a part of the presentation where two or more threads

of actions will be executed in parallel. It is impossible, however, to execute the given order in parallel if two different orders are given for the same agent; only one order may be executed at a time in this case.

```
<PAR agents="rocky, merlin">
 ...
</PAR>
```

The combination of <SEQ> and <PAR> tags enables a synchronization function as realized in SMIL, and allows for a complex scripting structure of the presentation.

Interaction tags

In an MPML presentation, there are usually two different components, i.e., the agents and the background. The agents are actors; they speak, move, make gestures, and thus present. In addition, the background plays an important role as supportive information for the presentation, as in the case of human presentation.

In the basic Web-version of MPML, the background consists of a Web page typically written in HTML. Background change is simply invoked by the <PAGE> tag specified in the MPML script. The common way of a page change by clicking an anchor of a Web link is also permitted in an MPML presentation; in this way the user may have control over the flow of a presentation.

For MPML being not only easy to use but also a powerful language, a simple interface with JavaScript functions is provided. As a result, if the background contains a JavaScript script realizing some complex functions, it is possible to integrate and synchronize the presentation with some event happening on the Web page. We will give some examples of possible usages of an interactive background below.

Page: This tag defines the background of the presentation. The page used can be any Web page and it appears behind the agents.

Wait: The first possible type of interaction between an MPML script and JavaScript is the <WAIT> tag. It simply consults the value of the variable or function included in the "target" attribute until it switches to a value different from "0". An example is as follows.

```
<PAGE id="page1" ref="MPML_evaluation_emotion.html">
   <WAIT target="Pressed" />
</PAGE>
```

(The "id" argument here can be omitted if unnecessary.)

Consult: This tag is the MPML equivalent of the well-known C "switch" instruction. It consults the data or function given in the "target" attribute, and then checks the <TEST> tags in order to select the one to be executed. The <CONSULT> tag can either perform one test only, and then go on even if no match is found, or wait until a match is found, depending on its "mode" attribute.

Test: This tag is to be used with the <CONSULT> tag as parent. It compares the value of the "target" attribute contained in the <CONSULT> tag with the value

of the "value" attribute of the <TEST> tag, and executes the script included in the <TEST> tag if they are identical.

```
<CONSULT target="dangerous" mode="pass">
  <TEST value="true">...
  </TEST>
</CONSULT>
```

Execute: This tag executes a JavaScript instruction. It should be mainly used to call a function in the background page of the presentation. Its use, however, is limited only by the need of the author.

```
<EXECUTE target="Alert(  )" />
```

Txt: This tag is used to introduce variable content into the speech of the agent. It permits some customization of the dialogue, for example, to include names or some answers of the user as shown below.

```
<SPEAK agent="rocky">
   Welcome, <TXT target="user-name" />!
</SPEAK>
```

Emotion, mood and personality tags

The presentation by lifelike agents has the risk of being monotonous or flat. Endowing an agent with emotion is an important way for enhancing the lifelikeness and believability of the agent. The display of agent emotions invokes emotions of interacting human users, and thus brings an effect of enhancing friendliness, motivation, entertainment, and so on.

Emotions are often named by words such as "happiness", "sadness", "surprise", "anger", "disgust", or "fear". The categorization of emotions, however, is not uniform, i.e., it is different from researcher to researcher, since there is no generally agreed upon common ground. In the emotion model called "the cognitive appraisal theory" or the "OCC model" after its three advocates that appeared in the book published in 1988,[43] the most comprehensive emotion theory consisting of twenty-two types of emotion is defined on the basis of a systematic analysis. MPML basically supports emotional expressions based on those twenty-two types of emotion; one reason of this choice is that the model is also used in an artificial emotion module called SCREAM, to be described later. Mainly for the purpose of sensing the user's emotional states, a simple emotion model based on the dimensions of valence and arousal[23] is also used in systems using MPML.

The emotion of the agent can be directly specified using an <EMOTION> tag as follows.

```
<EMOTION assign="rocky: angry">
<SPEAK agent="rocky">
  You stole my book, James!
</SPEAK>
</EMOTION>
```


or

```
<SPEAK agent="rocky">
 <EMOTION assign="angry" />
  You stole my book, James!
</SPEAK>
```

In the first case, the speak actions inside the <EMOTION> tag are modified according to the specified emotion. The speech parameters are set according to the given emotion. Table 1 illustrates a setting of speech parameters that is based on Murray and Arnott.[36] Due to the limitation of the TTS engine, presently only speech rate (speed), pitch average, pitch range, and speech intensity are controlled. In addition, for some types of emotion, appropriate actions are inserted before and/or after the speak action depending on the case.[71] While the present quality of synthesizing emotional speech is not always entirely satisfactory, the actions generated in accordance with the emotion compensate for the (sometimes poor) quality.

Table 1 Speech Parameter Setting for Emotional Speech

Emotion	*Fear*	*Anger*	*Sadness*	*Happiness*	*Disgust*
Speech rate	much faster	slightly faster	slightly slower	faster or slower	very much slower
Pitch average	very much higher	very much higher	slightly lower	much higher	very much lower
Pitch range	much wider	much wider	slightly narrower	much wider	slightly wider
Intensity	normal	higher	lower	higher	lower
Pitch changes	normal	abrupt on stressed syllables	downward inflections	smooth upward inflections	wide downward terminal inflections

In the second case above, the emotion specified continues for a short time period, and influences the speech parameters.

An emotion reflects a state of the mind and is a short-term phenomenon. On the other hand, mood is a phenomenon spanning a longer period. It works like a background emotional state when no strong emotion is occurring. The main function of mood in MPML presentations is to define how the author wants the agent to behave when no emotion occurs. Currently, only three types of mood are defined: "happy mood", "neutral mood" (default), and "unhappy mood".

The way of combining mood and emotion is simple. When there is no emotion occurring, the parameters defined for the mood is used. Here, a slight perturbation is added to increase the lifelikeness. When an emotion occurs, the parameters are switched quickly according to the emotion. The mood is specified, if necessary, as follows.

```
<MOOD assign="rocky: happy; merlin: neutral">
```

```
...
</MOOD>
```

Personality is another important feature of the agents. It is one of the characteristics of a human being that makes everyone act in his/her own way, and thus makes everyone unique. Unlike emotion or mood, the association of a personality with a person/agent lasts semi-permanently. When there are more than two agents appearing on a screen, it is beneficial to characterize each agent differently; otherwise, all the agents behave in similar manners and speak in similar ways.

One personality model originating from psychological research is the Five Factors model,[30] which is descriptive and widely used. It deals with five dimensions for describing the personality: extraversion, agreeableness, conscientiousness, neuroticism, and openness. In practice, however, often only two or three dimensions are employed.[5]

Personality-related functions incorporated into MPML are still on a preliminary level. Among the personality traits of the Five Factors model, MPML presently supports only two dimensions, i.e., agreeableness and activity (from extraversion). Activity affects the average frequency of agent behaviors. Agreeableness has an impact on the duration of the emotional states; that is, when the agent gets angry, it will soon recover from this negative emotion, and when the agent is joyful, this positive emotion will persist longer. Personality is specified within the agent definition tag to be placed in the header part of an MPML description as follows.

```
<AGENT id="rocky" character="rocky" spot="center"
   voice="{C2D4EF00-B025-11D4-BE23-0000F44780AB}"
   personality="active" />
```

Here the "spot" argument indicates the first position at which the agent appears.

§4 Several Versions of MPML

4.1 Basic Web Version

The basic and typical version of MPML is designed for describing multimodal Web contents, which is played normally with Microsoft's Internet Explorer (version 5.5 or higher, capable of interpreting XML) on a PC display. There are two ways of generating a presentation from a MPML description. The first one is to use XSL (Extensible Stylesheet Language) as a plug-in software of Internet Explorer, whereby XSL is a part of the XML technology. In this case, an MPML script is fed directly into the Web browser. The second way is via a converter software, which reads an MPML script and then generates a JavaScript code to perform a presentation in the Web browser. The backgrounds of the MPML presentation are normally composed of HTML files. Useful objects such as buttons, text-boxes, etc. can be placed in the browser window. Figure 1 shows screenshots of multimodal presentations written in the basic Web version of MPML.

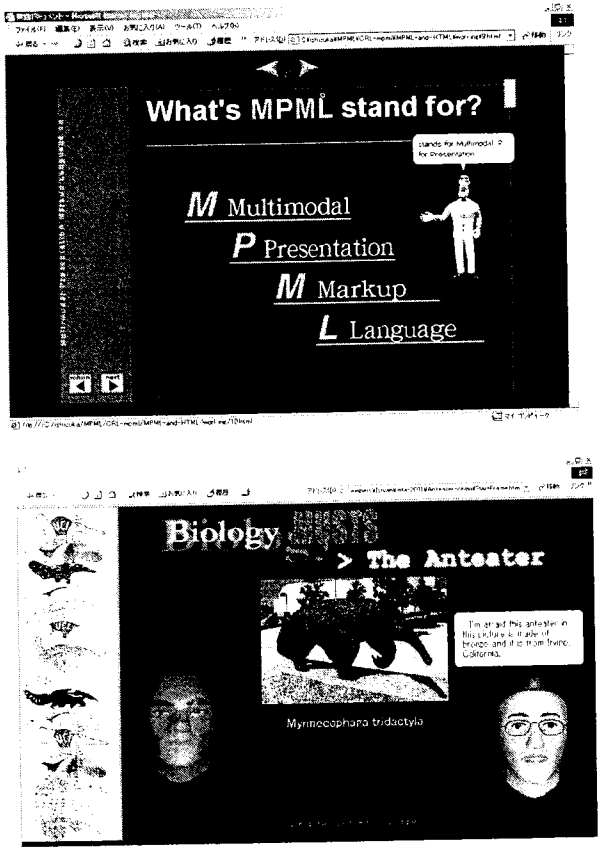

Fig. 1 Screenshots of MPML Basic Web Version

A graphical editor, a display snapshot of which is depicted in Fig. 2, has been developed for this basic Web version.[12,13] By using this graphical editor, people can produce MPML-based multimodal Web contents without knowing the tags or other description format of MPML, just like producing Web contents with a graphical Web editor, which does not require knowledge regarding HTML.

In order to fit to various environments in which multimodal contents can be played, the basic Web version of MPML has been extended in several ways, while keeping its basic description style.

4.2 MPML-VR

MPML-VR, where VR stands for "virtual reality", is a version of MPML for 3D VRML space.[39,42] The function of the <MOVE> tag in this version is changed to the movement of the agent in a defined 3D VRML space, rather than in a 2D display space. The movement in this case is basically in the horizontal space on a floor defined in VRML. In addition, the modes of the movement such

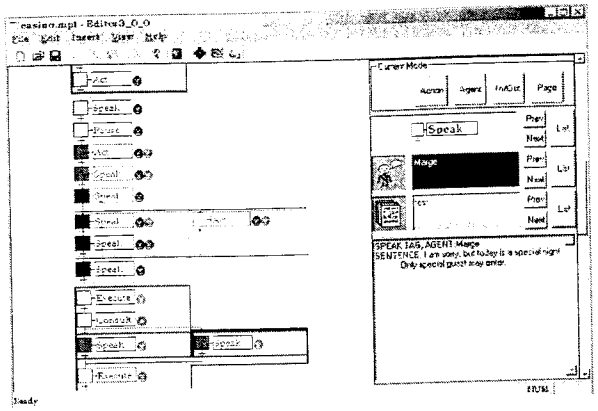

Fig. 2 Graphical Editor for MPML Basic Version

as "walk" and "run" can be specified in the <MOVE> tag as exemplified below.

```
<MOVE agent="aya" how="walk" location="pos-center" />
```

Here the "location" argument indicates the destination of the movement.

A new tag introduced in MPML-VR is <SET-VIEWPOINT>, which specifies a new camera position (and angle) in the VRML space as follows.

```
<SET-VIEWPOINT location="camera5" transition="ON" />
```

Another tag unique in MPML-VR is <OBJECT>, which allows to place a 3D object externally defined as a VRML object into the VRML space and to change its state in the presentation. For example, a television panel can be placed and a movie file can be played on it as follows.

```
<SCENE id="scn-movie">
<OBJECT id="TV" url="television.xml" location="pos-table1" />
   ...
   ...
<CALL object="TV" method="url" value="sample.mpg" />
<CALL object="TV" method="start" value="1" />
   ...
</SCENE>
```

Here <CALL> lets the object execute the designated method.

We also developed our original 3D agent characters along with MPML-VR. Characters conformed to H-anim, the standard design format for human figures in VRML space, could be used for this purpose, though our original characters are different from this standard. Figure 3 illustrates two display snapshots of the MPML-VR.

Fig. 3 Screenshots of MPML-VR

4.3 MPML-Mobile

Recently, most mobile phones are equipped with a mobile-phone Web Browser, and have become a major information and communication device other than personal computers, especially in Japan. Many types of Web contents are now provided also for the mobile phones. From the viewpoint of business with paid Web contents, the information service for the mobile phones is of importance, because a reliable payment system is established via the operating companies of the mobile phones. We therefore decided to develop a MPML-Mobile version in cooperation with a small company in 2002. The development also aimed to establish a de facto standard in the description language of multimodal contents with lifelike characters. While the standardization was not progressing well for the Web contents, we felt it might be possible for the mobile-phone Web contents, taking advantage of the leading position of mobile-phone information

services in Japan.

In MPML-Mobile,[62,63] dynamic flow control functions are enhanced to respond to user's actions. To change the behavior of the character agent or the background of the display scene, according to the user's actions on the Web page components or pushing a control button, an approach similar to that used in JavaScript has been adopted. That is, for this purpose, <FROM> tags with extended attributes like onClikGoto, onSelectGoto, onDeSelectGoto and onChangeGoto are embedded in the background Web page. The value of each of these attributes is a scene ID or a sub-scene ID. The <FORM> tag makes the character agent jump to the specified scene when the user takes an action.

Although voice output is technically feasible, a text in the <SPEECH> tag is displayed in a balloon text box in the MPML-Mobile version. MPML-Mobile players have been developed using J2ME (Java2 Mobile-edition) for the mobile handsets of major mobile-phone operating companies in Japan (NTT DoCoMo, KDDI-au and Vodafone). The program consisting of an MPML-Mobile parser and a converter into executable Java codes is run on the mobile handsets. Some agent characters have been developed for the multimodal information services on the mobile phones. These are 3D agent characters that perform with small computational power on mobile phones. Figure 4 shows some snapshots of the MPML-Mobile version.

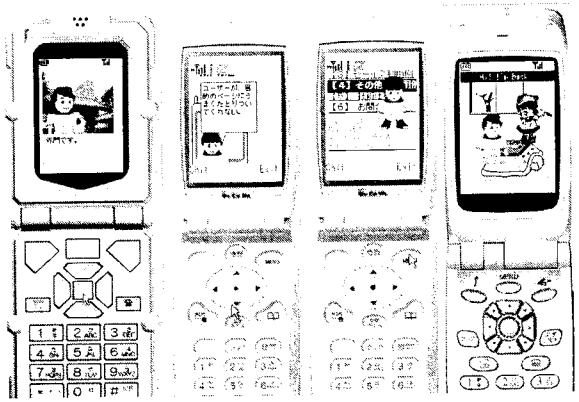

Fig. 4 Snapshots of MPML-Mobile

MPML-Mobile was deployed in a commercial service for KDDI-au's mobile phones in 2004; the company we collaborated with in the development offered this service.

4.4 MPML-HR

Humanoid robots are emerging recently, especially in Japan. They can perform the same role as lifelike agents, but operate in physical worlds, and thus give us a stronger impression and effect different from the software character agents that work on a display. At present, there exist no convenient tools for non-

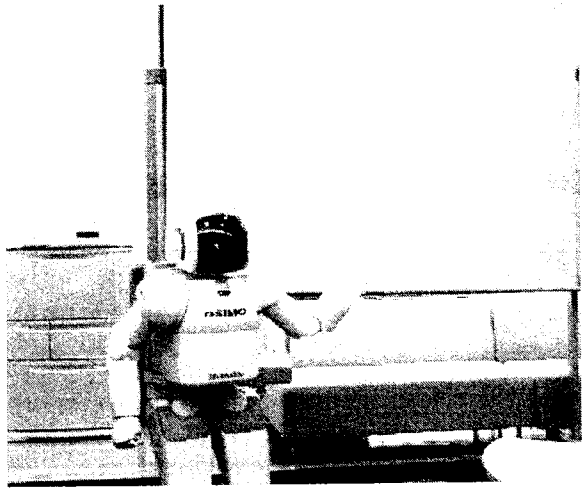

Fig. 5 ASIMO's Presentation Written in MPML-HR

professionals to describe and generate the behavior of humanoid robots. Thus we decided to apply our MPML technology to produce such a tool for the humanoid robots, primarily aiming at the application of multimodal presentations in the physical world. The result is MPML-HR (HR stands for "humanoid robot"),[22,38] a version of MPML.

MPML-HR allows people to describe multimodal presentation contents with a biped humanoid robot and a presentation display in a physical world. The first biped humanoid robot operated by MPML-HR was Fujitsu's HOAP-1; second and present one is Honda's Asimo which is the first and most advanced biped humanoid robot in the world. The research and development of MPML-HR using Asimo have been carried out in cooperation with Honda Research Institute Japan. Figure 5 shows a picture of Asimo's presentation.

The humanoid robots here are assumed to have pre-defined actions which are specified and activated by MPML-HR. Like MPML-VR, the <MOVE> tag in MPML-HR makes the biped humanoid robot move on the horizontal floor; here the position of the robot needs to be calibrated initially. A new unique tag introduced in MPML-HR is <POINT>, which makes the humanoid robot perform a pointing action in the direction of the information screen. The x-y values in the <MOVE> tag of MPML-HR denote the horizontal floor position; on the other hand, the x-y values in the <POINT> tag indicate the vertical screen position. Hence we separated these tags in the design of MPML-HR.

For example, if

```
<POINT id="Asimo" x="300" y="150" />
```

is written, then the robot moves to one side (right or left side) of the screen, and points to a position of the screen with his left or right hand. The x-y screen

coordinate values in the arguments will be approximated into one of six areas in the current implementation. i.e., the high, middle or low area of the right or left half of the screen display. In this case, if the pointing position is on the right half of the screen, the robot moves to the right side of the screen and points with his right hand. When pointing, the robot turns his body 20 degrees to the screen.

In order to increase the lifelikeness of the humanoid robot, an autonomous head motion occurs when the robot speaks with no action commands. Also, other autonomous idle behaviors are evoked when there are no action instructions for a certain period of time.

Although the control of humanoid robots has been very hard or impossible for non-professionals to date, MPML-HR enables them to easily generate certain behaviors of robots and to produce multimodal contents using the robots in the physical world. Although the presentation setting with a humanoid robot is not always available like the ordinary displays on which the character agents perform, the presentations by the humanoid robot are proving to be able to give a stronger impression than those by the character agents.

§5 Emotion-related Functions

5.1 *SCREAM*: An Artificial Emotion Module

Emotion is one important factor for making the agent lifelike, believable, friendly and empathic. The description and expression of the agent's emotion is one of the features of MPML. Specifying emotion by using the `<EMOTION>` tag whenever needed is, however, cumbersome. Thus, an artificial emotion module called SCREAM (Scripting Emotion-based Agent Minds)[45~50] has been developed, which works as an external module of MPML (and other programs), and decides the emotional state (a type of emotion and its intensity) and external emotion expression of the agent based on the current interaction situation. Parameters for its decision are derived from the agent's mental model (goal, belief and attitude), social relations holding among the interacting parties (both virtual and human), and features of interlocutor(s) or the user. In the case of MPML, SCREAM is called via the `<CONSULT>` tag. Here, information regarding the interaction situation is fed into SCREAM in the form of communicative acts written as `com_act (S, H, Concept, Sit)`, where `S` is the speaker, `H` the addressee (interlocutor), and `Concept` refers to information conveyed by `S` to `H` in situation `Sit`. Other parameters are pre-defined or maintained in SCREAM for each agent.

The core part of SCREAM is basically a rule-based system implemented in a Java-based Prolog system. It includes four sub-modules, i.e., appraisal, emotion resolution, emotion maintenance, and emotion regulation sub-modules. Figure 6 shows the structure of SCREAM.

In the appraisal sub-module, an event is evaluated as to its emotional significance for the agent, based on emotion-eliciting conditions formulated in the OCC emotion model[43], which defines twenty-two types of emotions such

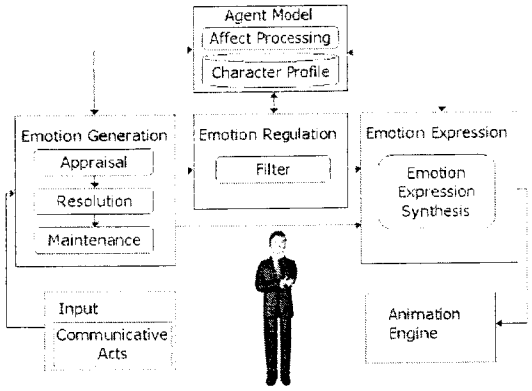

Fig. 6 Structure of SCREAM

as "joy", "happy for", "distress", "pride", "satisfaction", "anger", "love", and "hate", among others. The personality of the agent, if it is defined, is taken into account in this computation.

More than two active emotions in an agent can be generated with their intensities in this process; they are resolved in the emotion resolution sub-module. In the simple case that there exists one dominant emotion with sufficiently high intensity, it becomes the output of the appraisal sub-module. On the other hand, when there is no such a dominant emotion but are several low-intensity active emotions, then all the active emotions are taken into account in the computation. In this case, the active emotions are classified into 'positive' and 'negative' emotions. Examples of positive emotions are "joy", "happy for", "pride" and "satisfaction", whereas "distress", "anger" and "hate" are negative emotions. Then the dominant mood results by comparing the overall intensity values associated with the positive and negative emotion sets. The 'winning' emotional state is decided by comparing the intensities for the active emotions and the mood. Thereby, the emotion resolution sub-module can account for situations where, for example, an agent has a joyful experience but is still influenced by its overall negative mood towards another agent.

The emotion maintenance module handles the decay process of emotions. Depending on their type and intensity, emotions may remain active in the agent for a certain time during the interaction. The decay rate is determined by the agent's personality; an agreeable agent's decay rate for negative emotions is faster than that for positive ones.

In addition to the above three sub-modules, there is an emotion regulation sub-module attached, which decides whether a generated emotion is expressed or suppressed considering the social relation with the interlocutor or the user. When the agents interact, they do not only exchange information but also establish and maintain social relationships. Hence it is important that the agent avoids introducing disharmony into a conversation. Two parameters, social power (relative ranking such as a boss-subordinate relation) and social

distance (the closeness between two agents), are taken into account in emotion regulation. If the social distance is close, then the agent may express its emotion freely. If the social power of the interlocutor is higher, then the emotion regulation module makes the agent suppress its undesirable emotional expression against the interlocutor. In this way, this sub-module enables a social filtering function with respect to emotion expression.

When the author of the contents directly specifies the emotional state of an agent in MPML using the <EMOTION> tag, unnatural emotion transitions may occur; for example, a happy agent may abruptly change into a negative emotional state. The use of SCREAM is effective to avoid this problem by smoothening transitions between emotions of opposite valence (positive, negative), and also by keeping a dynamically updated record of the agent's attitude towards it interlocutor. However, an issue regarding current SCREAM is that it is not always easy to provide its inputs as appropriate communicative acts.

Content demonstrating the integration of MPML and SCREAM has been produced, where an advisor agent endowed with a SCREAM-based mind plays an advisor role in a virtual Black Jack casino. Here, thanks to the sophisticated modulation of emotional expression and attitude change, the advisor agent can achieve a high level of naturalness and believability.

5.2 Physiological Sensing and Eye Tracking

In order to achieve affective communication with a user, it is important for the interacting agent to infer the mental (and emotional) state of the user. There have been researches of using facial expression or voice tone for sensing human emotion. Although these sensing methods have the advantage that the user does not need to wear any sensing devices, they are not necessarily sufficiently reliable. On the other hand, physiological sensors measuring skin conductivity (SC), blood volume pulse (BVP), from which heart rate can be computed, electromyography (EMG), and others are now widely used in affective computing.[72] The development of portable sensing devices is now under way for the purpose of affective human-computer interfaces. Although they still have problems in the performance of emotion sensing, they can provide more reliable results than those obtained from facial expression or voice tone. Moreover, since these biosignals are mostly involuntary, they allow us to detect the user's cognitive and affective state free from the user's conscious regulation such as 'social masking' (for instance, suppressing the display of an "angry" face).

These physiological sensors are recently becoming employed in affective gaming in which the user's emotion is detected and used to control the scenario of the game. Likewise, we can utilize them to achieve 'physiologically perceptive lifelike characters' as a new generation of interface agents.[52,56~58] Towards this end, we have constructed some systems, which use SC and BVP sensors for detecting the user's emotional state. Skin conductivity (SC) correlates with a user's arousal level, and the blood volume pulse (BVP) rate with the valence (positive or negative) of a user's emotion. Thus, in this case, we can conveniently apply Lang's two-dimensional emotion model[23] shown in Fig. 7 for the

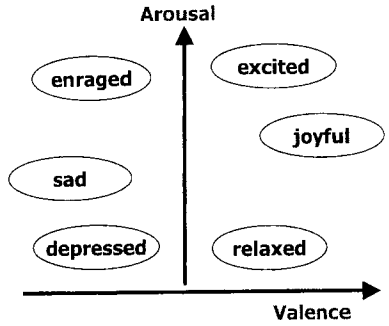

Fig. 7 Lang's Two-dimensional Emotion Model

recognition of the emotional state of the user.

One system constructed for a demonstration is a virtual job-interview system, where an agent character plays the interviewer of a company and the interviewee is a user.[55] The system is written in MPML receiving an input from the physiological sensors. While it is desirable for the interviewee to maintain his/her mind calm during the interview, some negative emotions, such as enraged-ness, fear and disappointment, might occur with him/her, for example, when the interviewer touches upon a weak point of the interviewee. An avatar agent of the user appears in a display window called "emotion mirror", and expresses strong emotions detected through the physiological sensors in real time. In this way, the user can have a practice interview while being notified about the emotional dynamics of his/her mind. This function thereby helps the user to have some ideas of controlling his/her mind in the job interview.

Another system that has been developed is a kind of affective gaming. It is an interactive cards game where a user plays the "Skip-Bo" game against an opponent agent.[7] Here, the affective behavior of the character agent is contingent on the user's physiological responses during game play. Two versions of the agent were implemented, each with its own type of empathetic attitude towards the user. In a positive empathetic version, the agent displays happiness if the user is detected to be in a happy or relaxed affective state. In the negative empathetic version, on the other hand, the agent will display gloating joy if the user is recognized to be negatively aroused. In both cases, the agent character will also display self-centered emotions, such as being happy about its own successful game move. One result of this study indicates that the absence of negative empathy is conceived as stressful (derived from the sensor of the galvanic skin conductivity), as it might also be experienced when playing against a human player. A complementary result is that the negative empathetic behavior of the agent character induced negatively valenced emotions in the user (derived from the electromyography sensor), which indicates a form of reciprocity in human response to the agent's behavior regarding the valence of emotions.

Physiological sensing of the user's mind and affective state also proved valuable in quantitatively evaluating how a part of a multimodal content is

perceived by the user.[34,51,53,55] Based on the results of those empirical studies, we can improve the friendliness and entertainment value of the content.

The eye tracker is another possible means offering useful information for affective interactions. We can infer the user's focus of attention and also the shift of the user's interest from eye information. Since most current devices are not very compact and convenient, we also started to employ a latest generation non-contact video based eye tracker in order to develop non-intrusive adaptive interfaces and affective interactions. Previously, we have used a head-mounted eye tracker for generating multimodal educational contents that is adaptive and affective by responding to the eye tracking results. The contents here employ the agent characters as virtual tutors or advisors, and are written basically in MPML. We also used the eye tracker to evaluate the effectiveness of agent characters and other modalities such as text and speech. Specifically, we evaluated the degree of the user's attention to the visual appearance of the character agent and the effectiveness of deictic hand and facial gestures of the agent for directing the user's attention to a particular point on the display.[26]

5.3 *SmArt Agent*: A Facial-emotion-rich Agent Character

While the emotion description and expression are an important feature of MPML, most full-bodied characters such as Microsoft Agent characters have poor facial emotion expressivity due to their small facial area. We therefore developed our original agent character system named SmArt,[6,14] which is a 'talking head' with rich facial emotion expressivity as shown in Fig. 8. The facial expressions of SmArt Agent have been synthesized by manipulating the action units (AU) proposed by Ekman.[15] The head is, of course, equipped with a lip-synchronization function for speech output. The SmArt Agent characters can be used through the same interface as that of Microsoft Agent characters.

Recently, other talking head systems with rich facial expressions similar to our SmArt Agent are appearing; some of them are designed to conform to MPEG-4 FAP (Facial Animation Parameters).

§6 More Autonomy of the Agents

The basic style of content creation in MPML is direct scripting, which is the most practical way of producing acceptable contents accommodating most people's needs at present. However, it is cumbersome for the content author to write every detail of a presentation or interaction scenario including multimodal expressions of the agent. An ideal situation might be such that the author gives a presentation text (or a presentation concept) and its associated materials such as graphs, pictures, etc., and then the system can let the embodied agent present the content in a friendly, affective and unique manner with appropriate use of multiple modalities, just like human presenters do. The presentation here may include interaction with the audience. In other words, we need to endow or enhance the autonomy of the presenter agent. There are some approaches being undertaken in the research area of lifelike agents towards this goal. In this section, we introduce three of our approaches.

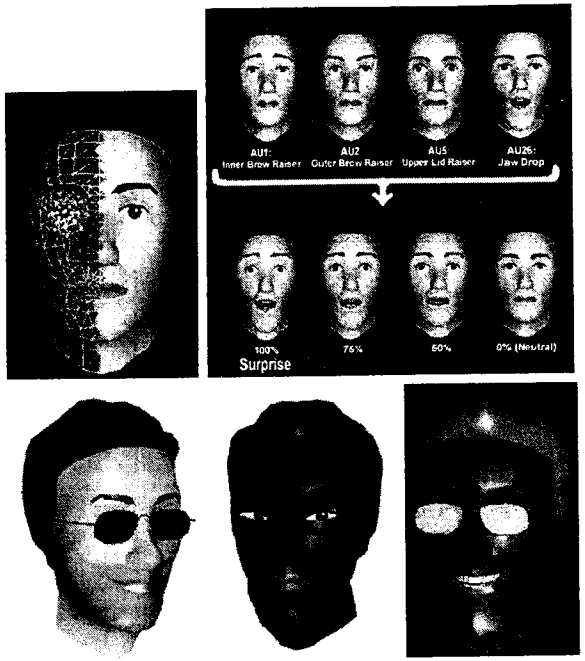

Fig. 8 SmArt Agents

6.1 Incorporation of Chatterbot for Flexible Conversation

Speech synthesis (or text-to-speech) technology is already at a practical level while researches are still going on towards making the speech output more natural and/or affective. On the other hand, speech recognition, especially in free conversation and noisy environments, is still problematic in many practical cases. A lot of research efforts are devoted to this area nowadays, and we can expect its performance getting better gradually.

In order to allow for flexible conversation with a user, the agent should have an intelligent dialogue management module as well as sufficient knowledge. For adaptive dialogue management, some advanced management models are proposed and studied,[1] such as the frame-based model, the plan-based model, and so on. The most practical one at present, however, is the finite-state script model, where responses are generated using a set of conceivable response patterns defined in each state and the state moves to a next state to process the user's new input. In all of the above models, since they are based on pre-defined knowledge or patterns for user's conceivable speech (or text) inputs, it is difficult to deal with unrestricted and unpredictable dialogue adequately. Thus the dialogue cannot proceed smoothly and the user often gets frustrated.

The chatterbot technology, which has its origin in Weizenbaum's ELIZA in 1966, has been progressing well in last decade; the Loebner-prize contest has been contributing as the central initiative of this progress. Chatterbots are able

to generate natural responses for almost any input from an interlocutor (a user). The dialogue generated by the chatterbot, however, is almost reactive in nature and does not convey any purposeful meanings.

We combined the chatterbot technology with the MPML script description of presentation scenarios for enabling flexible dialogues.[35] Here, a scenario-based presentation including some predictable dialogue parts is written in MPML basically conforming to the finite-state script model, whereas the chatterbot deals with unpredictable free conversation. We call the topics of the dialogue covered by the described script in MPML 'in-domain', and the topics not covered but taken care by the chatterbot 'out-of-domain'. In order to achieve a smooth transition between these two domains, two intermediate states, i.e., "reluctant" and "concede" states, are introduced in the dialogue management module.

During an in-domain dialogue, once the user says something out of the specified in-domain topics, the dialogue module enters into the reluctant state, where the agent engaged in the dialogue suggests that he/she is unable to understand the user, and therefore reluctant to pursue the direction proposed by the user, and would like to return to the in-domain topic. If the user persists in talking about out-of-domain topics, the dialogue module enters into the concede-state where it eventually allows the user to take control of the conversational topics in the out-of-domain. Although the dialogue module stays in the out-of-domain while the user persists in talking about the out-of-domain topic, it occasionally reminds the user of the in-domain topic of the current presentation, conveying its intention to move back to its original in-domain state. In this way, the agent is able to direct the conversation to proceed along the scripted scenario of the multimodal presentation while accepting free conversation on any topics with the chatterbot.

The chatterbot we are using is ALICE,[2] which is a two-time winner of the Loebner prize and contains more than 20,000 response patterns (rules). Our current system can accept only text input from the user because of the low reliability of speech input. We consider the use of the chatterbot technology as a promising approach towards flexible conversation.

6.2 Textual Emotion Estimation

One difficulty of authoring attractive multimodal contents with lifelike agents is that the author has to designate appropriate gestures or behaviors of the agent in addition to providing speech texts. It would be more convenient if these gestures or behaviors were generated automatically from the speech texts, just as in the case of human presenters. There have been some researches in this direction.

In BEAT,[10] some gestures, e.g., iconic gestures associated with particular verbs, are generated from a speech text for the agent. A similar approach is taken in Nakano et al.[37] for Japanese texts. In Kipp[20] the transcriptions of gestures and speech shown in a TV talk show were analyzed to empirically find associations between lexicons in the speech and gestures; the resultant association is used to generate the gestures of the agent from its speech texts. Hand

gestures of the agent are automatically generated in Stone et al.[38] based on the analysis of recorded human performances. Furthermore, in the NECA project, automatic gesture generation for an agent was studied primarily relying on the prosodic information available from speech synthesis.[21] Recently, in Smid et al.[35] facial gestures are generated from speech texts, based on the analysis of the transcripts of human newscasters' facial gestures.

We took a different approach, i.e., an approach of estimating the emotional state from textual information in order to automatically generate emotional behaviors of the agent.[27] Here, we assume that other behaviors such as pointing, moving to some screen location, and so on, are described in the MPML script by the author, since full automatic generation of the behaviors is still hard in terms of the practical content quality. The emotional behaviors of the agent contribute to improving the lifelikeness and believability of the agent. This function is also fit to the MPML specification, such that the emotion tags are automatically inserted into the MPML description through the analysis of the speech texts.

Recent good research on textual affect or emotion sensing is described in Liu et al.[25] where a sentence-level analysis (as opposed to word-level analysis) is conducted using a corpus of real-world sentences to build a textual affect model. By contrast, our approach described in Ma et al.[27] can basically be subsumed to the category of keyword spotting, but it is augmented with a synonym database and sentence-level processing. The objective of this method is to estimate the six basic emotions from a sentence, i.e., happiness, sadness, anger, fear, surprise and disgust. We first employ the WordNet-Affect Database[70] to identify affective words corresponding to the six emotions. Referring to WordNet1.6,[16] the synonyms of these affective words are additionally used for this purpose.

The above affective-word spotting method is too simple to deal with sentences such as "I think that he is happy" or "the lake near my city was very beautiful, but now it is polluted", in which the speakers are not necessarily happy. Hence we perform the following sentence-level processing.

We first eliminate non-emotional phrases such as (i) questions, and (ii) clause phrases beginning with 'when', 'after', 'before' or 'if'. In the second phase, we derive the syntactic subject and verb of a top-level sentence, and then recognize the following structure patterns of the verb phrase: (a) verb + adjective phrase, (b) verb + noun phrase, and (c) verb + clause sentence.

In (a), if the subject is referred to by the first person pronoun, or with noun terms that are related to the speaker, we calculate the emotion of the adjective phrase using the aforementioned word-level method. Otherwise, we treat the sentence as non-emotional according to this pattern. For example, in the case of "The book I bought yesterday is very good", since "The book" is related to the speaker due to the phrase of "I bought yesterday", the emotion estimation of the adjective phrase "very good" is attributed to the emotion of the speaker.

In (b), if the verb is included in the affective word and synonym databases, we analyze the noun phrase. If the noun term in the noun phrase is related to the speaker, we apply the emotion estimation of the verb to the speaker. If

the verb is not directly included in the affective word or synonym databases, we search another knowledge database extracted from the OMCS (Open Mind Common Sense) knowledge base,[33] which contains affective rules such that "buy something" is a way to "get something" and one of the effects of "get good thing" is "being happy". Thereby we can classify the sentence "I bought a very good book yesterday" as expressing happiness.

In (c), we apply the above methods (a) and (b) to the clause sentence, and obtain the emotion estimation of the clause sentence. For example, in the case of "I think he is good in that", "he is good in that" is considered to be non-emotional, so that through the analysis of "I think + non-emotional clause", the computed result is 'non-emotional'.

In addition to the above sentence-level processing methods, we need to handle negation in sentences. Since negatively prefixed words such as "unhappy" are already included in the affective word database, we do not need to treat them in a special way. On the other hand, if we detect negative verb forms such as "have not", "was not", "did not", and so on, then we flip the polarity of the emotion words in the sentence.

In order to generate lifelike emotional behaviors of the agent based on the result of textual emotion estimation, we need a more detailed emotion categorization than the six basic emotions and we have to prepare the agent's gestures associated with these emotions. We are currently working towards this aim.

6.3 *Auto-Presentation*: Automatic Content Creation from the Web

In parallel to the research on multimodal media and interfaces, we have also been working on Web intelligence functions, which include Web information mining and multidocument text summarization.[8,19,40,41] Thus we tried to combine the two researches to automatically create multimodal contents for the agent from Web sources. The first resultant system is Auto-Presentation,[64] in which technologies of Web information extraction and multidocument text summarization are employed as well as MPML.

The Auto-Presentation system generates a multimodal explanation or presentation from available Web sources in response to a given query word or sentence corresponding to the user's request. The system first understands the user's request for presenting a topic. Then it searches related information in a Web encyclopedia, i.e., Wikipedia in this case, and through search engines, i.e., Google, Yahoo! and AltaVista search engines. If the search topic is found successfully in Wikipedia, the retrieved information from this source can be considered as well structured, and hence the outline of the presentation is instantiated by its given data. If the request cannot be matched with information found in Wikipedia, the system tries to extract essential information segments for the presentation using template-based data mining. The template used in the present system is structured as having the following items: (i) What/Who is the [topic] (about us, about [topic], introduction, mission, and objective), (ii) Whereabout [topic] (contact us, profile, location, and services), (iii) Why [topic] (the text snippets returned by the search engines), (iv) How [topic] (the snippets

returned by the search engines), and (v) The short texts not already inserted in the present template and found with emphasizing tags like <h1>, ···, <h4>, <b>, , <big>, and <dt>, where "( )"s following the items indicate key/cue phrases to mine the text contents for those items.

Multidocument text summarization is performed on the set of extracted text segments to produce a compact and meaningful explanation or presentation for each item. Furthermore, images related to the topic are retrieved by an image search engine, i.e., Google Image Search in this case, for producing the presentation background.

Using the presentation materials (itemized texts and images) thus prepared for the topic, the system then constructs an MPML presentation script according to a pre-defined presentation template, where more than two agents present the produced content with the background consisting of the retrieved texts and images. Figure 9 shows a screenshot of this kind of presentation, where the topic is 'big bang'.

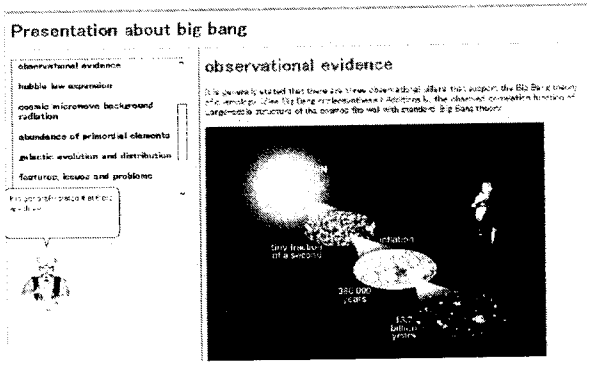

Fig. 9 A Screenshot of AutoPresentation, where the Topic is 'Big Bang'

While Auto-Presentation can generate multimodal presentations with good or acceptable quality when Wikipedia information is available, its content quality otherwise is not always satisfactory or below an acceptable level at present. As we have been working on methods for sentence ordering among extracted important sentences to generate multidocument text summarization with high readability,[8,41] we are now applying these techniques to improve the content quality of Auto-Presentation. This work can be said to constitute a part of a storytelling system that organizes comprehensible and readable stories from the extracted fragmental pieces of information.

§7 Concluding Remarks

We have described an overview of our research and developments on the multimodal media and contents with embodied lifelike agents, particularly the research centered on our MPML (Multimodal Presentation Markup Language).

While the effectiveness of lifelike agents has been shown in research and developments of the last decade, and we sometimes see multimodal contents utilizing character agents on the Web and elsewhere nowadays, their popularity still remains low. One major reason for this, we think, is the lack of a standard (de facto or de jure) content description language for easy authoring and the universally wide distribution of multimodal contents. In the course of the research on MPML, we have been wishing to contributing to the establishment of such a standard, and thus to raise its popularity as a new style of media contents and interfaces. Beside our efforts, there have been various efforts in the world in the same direction. Yet, the road to standardization is still unclear at present. Like HTML for Web contents, it is of utmost importance to have a standardized description language, which is easy for everyone to understand and to write, in order to generate multimodal contents with lifelike agents.

To demonstrate the advantage and usability of MPML in various environments, we have developed several versions of MPML while keeping its basic format. The basic MPML version is for the Web contents to be played on a display screen. Other versions are MPML-VR for 3D VHML space, MPML-Mobile for mobile phones, and MPML-HR for humanoid robots in the physical world. In order to have lifelike agents be liked and accepted by people as an attractive and friendly media style, lifelike agents should be affective and empathetic as well as natural. Hence the functionalities concerning emotion of agents have been emphasized in the research on MPML, and have become an essential feature of MPML. One important issue of MPML and lifelike agent research in general is to endow and enhance the autonomy of the agents in order to alleviate the authoring workload for producing contents for lifelike agents. We have shown some of our approaches towards that goal in this paper. As for general key qualities regarding the lifelikeness or believability of agents, it is said that agents should seem conversational, intelligent, individual, social, empathic, variable, and coherent.[17] These are current and future issues of lifelike agent research.

Acknowledgements

We would like to thank the MPML members who contributed to the researches reported here. The collaborations with Hottolink Inc. and Honda Research Institute Japan are also acknowledged regarding MPML-Mobile and MPML-HR, respectively. Our research reported here was supported by the Research Grant (FY1999 - FY2003) of the Future Program ("Mirai Kaitaku") from the Japan Society for the Promotion of Science (JSPS).

References

1) Allen, J., et al., "Towards Conversational Human-Computer Interaction," *AI Magazine, Vol. 22, No. 4*, pp. 27-38, 2001.

2) A.L.I.C.E. Artificial Intelligence Foundation, URL: http://www.alicebot.org/.

3) Arafa, Y., et al., "Two Approaches to Scripting Character Animation," in *Proc. AAMAS-02 Workshop on ECA - Let's Specify and Evaluate Them!*, Bologna, Italy.

4) Badler, N.I., et al., "Parameterized Action Representation for Virtual Human Agents," in *Embodied Conversational Agents* (Cassell, J., et al. (eds.)), pp. 256-284. The MIT Press, 2000.

5) Ball, G. and Breese, J., "Emotion and Personality in a Conversational Agent," in *Embodied Conversational Agents* (J. Cassell, et al. (eds.)), pp. 189-219. The MIT Press, 2000.

6) Barakonyi, I. and Ishizuka, M., "A 3D Agent with Synthetic Face and Semiautonomous Behavior for Multimodal Presentations," *Proc. Multimedia Technology and Applications Conference (MTAC'01, IEEE Computer Soc.)*, pp. 21-25, Irvine, California, USA, 2001.

7) Becker, C., Prendinger, H., Ishizuka, M. and Wachsmuth, I., "Evaluating Affective Feedback of the 3D Agent Max in a Competitive Cards Game," in *Proc. First Int'l Conf. on Affective Computing and Intelligent Interaction (ACII'05)* (Tao, J., Tan, T. and Picard, R.W. eds.), *LNCS 3784*, Springer, Beijing, China, pp. 466-473, 2005.

8) Bollegala, D., Okazaki, N. and Ishizuka, M., "A Machine Learning Approach to Sentence Ordering for Multidocument Summarization and its Evaluation," in *Proc. of 2nd Int'l Joint Conf. on Natural Language Processing (IJCNLP'05)* (Dale, R., Wong, K.-F., Su, J. and Kwong, O.Y. (eds.)), *LNAI 3651*, Springer, Jeju Island, Korea, pp. 624-635 , 2005.

9) Cassell, J., Sullivan, J., Prevost, S. and Churchill, E. (eds.), *Embodied Conversational Agents*, The MIT Press, 2000.

10) Cassell, J., Vilhjalmsson, H. and Bickmore, T., "BEAT: The Behavior Expression Animation Toolkit," in *Proc. SIGGRAPH-01*, pp. 477-486, 2001.

11) DeCarolis, B., Carofiglio, V., Bilvi, M. and Pelachaud, C., "APML: a Mark-up Language for Believable Behavior Generation," in *Proc. AAMAS'02 Workshop on ECA - Let's Specify and Evaluate Them!*, Bologna, Italy, 2002.

12) Descamps, S. and Ishizuka, M., "Bringing Affective Behavior to Presentation Agents," in *Proc. 3rd Int'l Workshop on Multimedia Network Systems (MNS2001)* (IEEE Computer Soc.), pp. 332-336, Mesa, Arizona, 2001.

13) Descamps, S., Prendinger, H. and Ishizuka, M., "A Multimodal Presentation Mark-up Language for Enhanced Affective Presentation," in *Advances in Education Technologies: Multimedia, WWW and Distant Education*, in *Proc. Int'l Conf. on Intelligent Multimedia and Distant Learning (ICIMADE'01)*, pp. 9-16, Fargo, North Dakota, USA, 2001.

14) Descamps, S., Barakonyi, I. and Ishizuka, M., "Making the Web Emotional: Authoring Multimodal Presentations Using a Synthetic 3D Agent," in *Proc. OZCHI'01 (Computer-Human Interaction, SIG of Australia)*, pp. 25-30, Perth, Australia, 2001.

15) Ekman, P., Friesen, W.V. and Hager, J.C., *The Facial Action Coding System, 2nd ed.*, Weidenfeld & Nicolson, London, 2002.

16) Fellbaum, C., *WordNet: An Electronic Lexical Database*, The MIT Press, 1982.

17) Hayes-Roth, B., "What Makes Characters Seem Life-like?," in *Life-Like Characters* (Prendinger, H. and Ishizuka, M. (eds.)), pp. 447-462, Springer-Verlag, 2004.

18) Huang, Z., Eliens, A. and Visser, C., "STEP: a Scripting Language for Embodied Agent," in *Proc. PRICAI'02 Workshop on Lifelike Animated Agent - Tools, Affective Functions and Applications*, Tokyo, 2002.

19) Jatowt, A. and Ishizuka, M., "Summarization of Dynamic Content in Web Collections," in *Proc. 8th European Conf. on Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD'04), Lecture Notes in Artificial Intelligence, LNAI 3202*, pp. 245-254, Springer, Pisa, Italy, (2004)

20) Kipp, M., "From Human Gesture to Synthetic Action," in *Proc. 5th Int'l Conf. on Autonomous Agents*, pp. 9-14, Montreal, 2001.

21) Krenn, B. and Pirker, H., "Defining the Gesticon: Language and Gesture Coordination for Interacting Embodied Agents," in *Proc. AISB'04 Symp. on Language, Speech and Gesture for Expressive Characters*, pp. 107-115, Univ. of Leeds, UK, 2004.

22) Kushida, K., Nishimura, Y., Dohi, H., Ishizuka., M., Takeuchi, J. and Tsujino, H., "Humanoid Robot Presentation through Multimodal Presentation Markup Language MPML-HR," *Proc. AAMAS'05 Workshop 13, Creating Bonds with Humanoids*, pp. 23-29, Utrecht, The Netherlands, 2005.

23) Lang, P.J., "The Emotion Probe: Studies of Motivation and Attention," *American Psychologist, Vol. 50, No. 5*, pp. 372-385, 1995.

24) Lester, J., et al., "The Persona Effect: Affective Impact of Animated Pedagogical Agents," in *Proc. CHI-97*, pp. 359-666, Atlanta, Georgia, 1997.

25) Liu, H., Lieberman, H. and Selker, T., "A Model of Textual Affect Sensing Using Real-World Knowledge," *Proc. Int'l Conf. on Intelligent User Interfaces (IUI'03)*, pp. 125-132, Miami, Florida, 2003.

26) Ma, C., Prendinger, H. and Ishizuka, M., "Eye Movement as an Indicator of Users' Involvement with Embodied Interfaces at the Low Level," *Proc. Symposium on Conversational Informatics for Supporting Social Intelligence & Interaction: Situational and Environmental Information Enforcing Involvement in Conversation (AISB'05)*, pp. 136-143, Hatfield, UK, 2005.

27) Ma, C., Prendinger, H. and Ishizuka, M., "Emotion Estimation and Reasoning Based on Affective Textual Interaction," in *Proc. First Int'l Conf. on Affective Computing and Intelligent Interaction, First Int'l Conf. ACII '05* (J. Tao, T. Tan and R. W. Picard (eds.)), *LNCS 3784*, Springer, pp. 622-628, Beijing, China, 2005.

28) Marriott, A. and Stallo, J., "VHML - Uncertainties and Problems: A Discussion," in *Proc. AAMAS'02 Workshop on ECA - Let's Specify and Evaluate Them!*, Bologna, Italy, 2002.

29) Masum, S.M.A., Ishizuka, M. and Islam, Md.T., "'Auto-Presentation': A Multi-Agent System for Building Automatic Multi-Modal Presentation of a Topic from World Wide Web Information," *Proc. 2005 IEEE/WIC/ACM Int'l Conf. on Intelligent Agent Technology (WI/IAT2005)*, pp. 246-249, Compiegne, France, 2005.

30) McCrae, R.R. and John, O.P., "An Introduction to the Five Factor Model and its Applications," *Journ. of Personality, Vol. 60*, pp. 175-215, 1992.

31) Mehrabian, A., *Nonverval Communication*, Aldin-Atherton, Chicago, 1971.

32) Microsoft: *Developing for Microsoft Agent*, Microsoft Press, 1998.

33) MIT Media Lab. "Open Mind Common Sense." http://commonsense.media.mit.edu/, 2005.

34) Mori, J., Prendinger, H. and Ishizuka, M., "Evaluation of an Embodied Conversational Agent with Affective Behavior," in *Proc. AAMAS'03 Workshop (W10) - Embodied Conversational Characters as Individuals*, pp. 58-61, Melbourne, Australia, 2003.

35) Mori, K., Jatowt, A. and Ishizuka, M., "Enhancing Conversational Flexibility in Multimodal Interactions with Embodied Lifelike Agents," in *Proc. Int'l Conf. on Intelligent User Interfaces (IUI'03)*, pp. 270-272, ACM Press, Miami, Florida, USA, 2003.

36) Murray, I.R. and Arnott, J.L., "Implementation and Testing of a System for Producing Emotion-by-Rule in Synthetic Speech," *Speech Communication, Vol. 16*, pp. 369-390, 1995.

37) Nakano, Y.I., et al., "Converting Text into Agent Animation: Assigning Gestures to Text," in *Proc. Human Language Tech. Conf. of North America Chapter of ACL (HLTNAACL'04)*, pp. 153-156, 2004.

38) Nozawa, Y., Dohi, H., Iba, H. and Ishizuka, M., "Humanoid Robot Presentation Controlled by Multimodal Presentation Markup Language MPML," in *Proc. 13th IEEE Int'l Workshop on Robot and Human Interactive Communication. (RO-MAN'04), No. 026*, Kurashiki, Japan, 2004.

39) Okazaki, N., Aya, S., Saeyor, S. and Ishizuka, M., "A Multimodal Presentation Markup Language MPML-VR for a 3D Virtual Space," *in Workshop Proc. (CD-ROM) on Virtual Conversational Characters: Applications, Methods, and Research Challenges (in conjunction with HF'02 and OZCHI'02)*, 4 pages, Melbourne, Australia, 2002.

40) Okazaki, N., Matsuo, Y. and Ishizuka, M., "TISS: An Integrated Summarization System for TSC-3," in *Working Notes of the Fourth NTCIR Workshop Meeting (NTCIR-4)*, pp. 436-443, Tokyo, Japan, 2004.

41) Okazaki, N., Matsuo, Y. and Ishizuka, M., "Improving Chronological Sentence Ordering by Precedence Relation," in *Proc. 20th Int'l Conf. on Computational Linguistics (COLING'04)*, pp. 750-756, Geneva, Swiss, 2004.

42) Okazaki, N., Saeyor, S., Dohi, H. and Ishizuka, M., "An Extention of the Multimodal Presentation Markup Language (MPML) to a Tree-dimensional VRML Space," *Systems and Computers in Japan, Vol. 36, No. 14*, pp. 69-80, Wiley Periodicals Inc., 2005.

43) Ortony, A., Clore, G.L. and Collins, A., *The Cognitive Structure of Emotions*, Cambridge Univ. Press, 1988.

44) Piwek, P., et al., "RRL: A Rich Representation Language for the Description of Agent Behavior in NECA," in *Proc. AAMAS-02 Workshop on ECA - Let's Specify and Evaluate Them!*, Bologna, Italy, 2002.

45) Prendinger, H. and Ishizuka, M., "Social Role Awareness in Animated Agents," in *Proc. 5th International Conf. on Autonomous Agents (Agents'01)*, pp. 270-277, Montreal, Canada, 2001.

46) Prendinger, H. and Ishizuka, M. "Agents That Talk Back (Sometimes): Filter Programs for Affective Communication," in *Proc. 2nd Workshop on Attitude, Personality and Emotions in User-Adapted Interaction, in conjunction with User Modeling 2001*, 6 pages, Sonthofen, Germany, 2001.

47) Prendinger, H. and Ishizuka, M., "Let's Talk! Socially Intelligent Agents for Language Conversation Training," *IEEE Trans. on System, Man and Cybernetics, Part A, Vol.31, Issue 5*, pp.465-471, 2001.

48) Prendinger, H. and Ishizuka, M., "SCREAM: Scripting Emotion-based Agent Minds," in *Proc. 1st Int'l Joint Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS-02)*, pp.350-351, Bologna, Italy, 2002.

49) Prendinger, H., Descamps, S. and Ishizuka, M., "Scripting the Bodies and Minds of Life-like Characters," in *PRICAI 2002: Trends in Artificial Intelligence (Proc. 7th Pacific Rim Int'l Conf. on AI, Tokyo)* (Ishizuka, M., Sattar, A. (eds.)), *LNAI 2417*, pp.571-580, Springer, 2002.

50) Prendinger, H., Descamps, S. and Ishizuka, M., "Scripting Affective Communication with Life-like Characters in Web-based Interaction Systems," *Applied Artificial Intelligence, Vol.16, No.7-8*, pp.519-553, 2002.

51) Prendinger, H., Mayer, S., Mori J. and Ishizuka, M., "Persona Effect Revisited: Using Bio-signals to Measure and Reflect the Impact of Character-based Interfaces," in *Proc. 4th Int'l Working Conf. on Intelligent Virtual Agents (IVA-03)*, pp.283-291, Kloster Irsee, Germany, (Springer, Berlin Heidelberg 2003), 2003.

52) Prendinger, H. and Ishizuka, (eds.), *Life-Like Characters – Tools, Affective Functions and Applications*, Cognitive Technologies Series, Springer-Verlag, 2004.

53) Prendinger, H., Mori, J., Saeyor, S., Mori, K., Okazaki, N., Juli, Y., Mayer, S., Dohi, H. and Ishizuka, M., "Scripting and Evaluating Affective Interactions with Embodied Conversational Agents," *KI Zeitschrift (German Journal of Artificial Intelligence), Vol.1*, pp.4-10, 2004.

54) Prendinger, H., Descamps S, and Ishizuka, M., "MPML: A Markup Language for Controlling the Behavior of Life-like Characters," *Journal of Visual Languages and Computing, Vol.15, No.2*, pp.183-203, 2004.

55) Prendinger, H., Dohi, H., Wang, H., Mayer S. and Ishizuka, M. ,"Empathic Embodied Interfaces: Addressing Users' Affective State," in *Proc. Tutorial and Research Workshop on Affective Dialogue Systems (ADS'04)*, pp.53-64, Kloster Irsee, Germany, 2004.

56) Prendinger, H., Mori, J. and Ishizuka, M., "Using Human Physiology to Evaluate Subtle Expressivity of a Virtual Quizmaster in a Mathematical Game," *Int'l Journal of Human-Computer Studies, Vol.62*, pp.231-245, 2005.

57) Prendinger, H. and Ishizuka, M., "The Empathic Companion: A Character-based Interface that Addresses User's Affective States," *Int'l Journal of Applied Artificial Intelligence, Vol.19, No.3-4*, pp.267-285, 2005.

58) Prendinger, H. and Ishizuka, M., "Human Physiology as a Basis for Designing and Evaluating Affective Communication with Life-Like Characters (Invited Paper)," *IEICE Trans. on Information and Systems, (Special Section on Life-like Agent and its Communication), Vol.E88-D, No.11*, pp.2453-2460, 2005.

59) Reeves, B. and Nass, C., *Media Equation: How People Treat Computers, Television, and New Media like Real People and Places*, Univ. of Chicago Press, 1996.

60) Ruttkay Z., and Pelachaud C., (eds.), *From Brows to Trust – Evaluating Embodied Conversational Agents*, Kluwer Academic Pub, 2004.

61) Saeyor, S., Binda, H. and Ishizuka, M., "Visual Authoring Tool for Presentation Agent Based on Multimodal Presentation Markup Language," in *Proc. Information Visualization (IV'01)*, pp. 563-567, London, England, 2001.

62) Saeyor, S., Uchiyama, K. and Ishizuka, M., "Multimodal Presentation Markup Language on Mobile Phones," in *Proc. AAMAS'03 Workshop (W10) - Embodied Conversational Characters as Individuals*, pp.68-71, Melbourne, Australia, 2003.

63) Saeyor, S., Mukherjee, S., Uchiyama, K. and Ishizuka, M., "A Scripting Language for Multimodal Presentation on Mobile Phones," in *Intelligent Virtual Agents* (Rist, T., Aylett, R., Ballin, D., Rickel, J. (eds.)), (in *Proc. 4th Int'l Workshop, IVA'03, Kloster Irsee, Germany*), 2003.

64) Shaikh, M., Ishizuka, M. and Islam, Md.T., "'Auto-Presentation': A Multi-Agent System for Building Automatic Multi-Modal Presentation of a Topic from World Wide Web Information," in *Proc. 2005 IEEE/WIC/ACM Int'l Conf. on Intelligent Agent Technology (WI/IAT'05)*, pp.246-249, Compiegne, France, 2005

65) Smid, K., Pandzic, I.S. and Radman, V., "Automatic Content Production for an Autonomous Speaker Agent," in *Proc. AISB'05 Symp. on Conversational Informatics for Supporting Social Intelligence*, pp.103-112, Hatfield, UK, 2005.

66) Synchronized Multimedia Integration Language, URL: http://www.w3.org/AudioVideo.

67) Stock, O. and Zancanao, M. (eds.), *Multimodal Intelligent Information Presentation*, Springer, 2005.

68) Stone, M., et al., "Speaking with Hand: Creating Automated Animated Conversational Characters from Recordings of Human Performance," *ACM Trans. Graphics (SIGGRAPH), Vol.23, No.3*, 2004.

69) Tsutsui, T., Saeyor, S. and Ishizuka, M., "MPML: A Multimodal Presentation Markup Language with Character Agent Control Functions," in *Proc. (CD-ROM) WebNet 2000 World Conf. on the WWW and Internet, San Antonio, Texas*, USA, 2000.

70) Valitutti, A., Strapparava, C. and Stock, O., "Developing Affective Lexical Resources," *Psychology Jour., Vol.2, No.1*, pp.61-83, 2004. 71.

71) Zong, Y., Dohi, H. and Ishizuka, M., "Multimodal Presentation Markup Language supporting Emotion Expression," in *Proc. (CD-ROM) Workshop on Multimedia Computing on the World Wide Web/ (MCWWW2000)*, Seattle, 2000.

72) Picard, R., *Affective Computing*, The MIT Press, 2000.

Mitsuru Ishizuka, Ph.D.: He is a professor at the Graduate School of Information Science and Technology, Univ. of Tokyo. Previously, he worked at NTT Yokosuka Laboratory and Institute of Industrial Sceince, Univ. of Tokyo. During 1980-81, he was a visiting assoc. professor at Purdue University. He received his B.S., M.S. and Ph.D. degrees in electronic engineering from the Univ. of Tokyo. His research interests are in the areas of artificial intelligence, multimodal media with lifelike agents, and intelligent WWW information space. He is a member of IEEE, AAAI, Japanese Society for AI (currently, president), IPS Japan, IEICE Japan, etc.

Helmut Prendinger, Ph.D.: He is associate professor at the National Institute of Informatics. Previously, he held positions as a research associate and JSPS post-doctoral fellow at the Univ. of Tokyo. Earlier he worked as a junior specialist at the Univ. of California, Irvine. He received his M.A. and Ph.D. degrees from the Univ. of Salzburg, Dept. of Logic and Philosophy of Science and Dept. of Computer Science. His research interests include artificial intelligence, affective computing, and human-computer interaction, in which areas he has published more than 65 papers in international journals and conferences. He is a co-editor (with Mitsuru Ishizuka) of a book on Life-Like Characters that appeared in the Cognitive Technologies series of Springer.