Making Them Remember—Emotional Virtual Characters with Memory

Zerrin Kasap, Maher Ben Moussa, Parag Chaudhuri, and Nadia Magnenat-Thalmann ■ *University of Geneva*

The search for the perfect virtual character is on, but the moment users interact with characters, any illusion that we've found it is broken. Adding memory capabilities to models of human emotions, personality, and behavior traits is a step toward a more natural interaction style.

Research on virtual characters has been ongoing for the past 20 years. Early efforts focused mostly on making the characters move and speak—that is, on body and facial animation. Simultaneously, researchers worked on making characters look convincing by adding animation and rendering hair, clothes, and muscles. The next step was to increase artists' interactive control over characters so that it was easier to create convincing video games and cinema.

Today, research into user interactivity has come to the forefront. It's no longer sufficient for characters to simply look like imitations of humans. They must behave like humans, too. This fact drives research into emotional and conversational virtual characters, or embodied conversational agents. The goal is to create a virtual character that has a human-like personality and that can emotionally respond while conversing with a user. To this end, some researchers mathematically model emotions, behavior, mood, and personality for virtual characters. As we describe here, researchers can use these models to create an emotionally responsive character. However, such models lack the critical component of memory—a memory of not just events but also past emotional interaction.

We've developed a memory-based emotion model that uses the memory of past interactions to build long-term relationships between the virtual character and users. We combine this model with state-of-the-art animation blending to generate smooth animation for the character during the interaction. To make the interaction more natural, we also use face recognition techniques; the character can thus "remember" a user's face and automatically adjust the current interaction on the basis of its existing relationship with the user. Finally, to increase the user's immersion, we place a life-sized character in a real environment using marker-based augmented reality (AR) techniques. Our example application is Eva, a geography teacher who has multiple interactions with two student users.

Modeling Realistic Characters

To create realistic characters, we must create models based on three general aspects: emotion, mood and personality, and relationship.

Modeling Emotions

Emotions have proven effects on cognitive processes such as action selection, learning, memory, motivation, and planning. Our emotions both motivate our decisions and have impact on our actions. As such, they're a key mechanism for controlling virtual-character behavior by both creating characters' personality and automatically producing animations by simulating characters' internal dynamics.

Jonathan Gratch and Stacy Marsella define two methods for modeling emotion in lifelike characters: communicative-driven methods and simulation-based methods.[1] Communicative-driven methods treat emotional displays as a means of communication. These systems don't internally calculate emotion; instead, they select an emo-

Published by the IEEE Computer Society

tional display—which is typically encoded in a scripting language—based on the current interaction state. Communicative-driven methods typically use Paul Ekman's model of basic emotions, which has six universally accepted emotion labels: fear, disgust, anger, sadness, surprise, and joy.[2]

Simulation-based approaches attempt to model the impact of events on internal emotion dynamics, focusing on an emotion's cognitive function. This approach ties emotional displays to the virtual character's emotional state, rather than having them triggered by their communicative function. These systems model the appraisal of environmental events and their effects on the character's internal emotional state. The most popular example of this is the Ortony, Clore, and Collins (OCC) model.[3] The OCC model divides a character's concerns in an environment into goals (desired states of the world), standards (ideas about how people should act) and preferences (likes and dislikes). The model defines 22 emotion labels, although Ortony later found the 22 distinct emotion types to be too complex for simulating believable characters. He therefore decreased the number of labels to 12—six positive (joy, hope, relief, pride, gratitude, and love) and six negative (distress, fear, disappointment, remorse, anger, and hate).[4] As we discuss later, we use Ortony's 12 emotion labels and add four labels.

Modeling Mood and Personality

We can differentiate moods and emotions on the basis of three criteria: time, expression, and cause.[5] In terms of time, moods last longer than emotions and aren't associated with a specific event: emotions modulate actions, while moods modulate cognition. In terms of expression and cause, the relation between mood and emotions is two-way. Mood affects the appraisal of events and decides which emotion will be triggered and with what intensity. For example, when people feel anxious, they're more easily disappointed by bad events, and that disappointment's intensity is heightened. Emotions can also cause a particular mood to occur. For example, a bored person can change to a more positive mood after a positive emotional appraisal from the environment. Researchers typically represent moods using continuous dimensions rather than discrete labels. To model moods, we use Albert Mehrabian's pleasure-arousal-dominance (PAD) space (as we describe in detail later).

Personality influences how people perceive their environment and affects their behaviors and actions. Personality is constant; like mood, it's not specific to particular events. For example, people with stable personalities tend to behave less emotionally in difficult situations. Although there's no universally accepted theory of personality, the Five Factor, or "Ocean," model[6] is the most widely used for simulating virtual-character personality. According to this model, we can define a person's personality according to five traits:

- *Openness* (O). Open people are imaginative, intelligent, and creative. They like to experience new things.
- *Conscientiousness* (C). Conscientious people are responsible, reliable, and tidy. They think about all their behaviors' outputs before acting and take responsibility for their actions.
- *Extroversion* (E). Extroverts are outgoing, sociable, and assertive. They're energetic in achieving their goals.
- *Agreeableness* (A). Agreeable people are trustworthy, kind, and cooperative. They consider other people's goals and are ready to surrender their own goals.
- *Neuroticism* (N). Neurotic people are anxious, nervous, and prone to depression. They lack emotional stability.

Typically, a character is a combination of these traits, sometimes with an emphasis on one of them. Although this static, trait-based personality model doesn't truly reflect human behavior's complexity, it's widely used in computational models because of its simplicity.

Modeling Social Relationships

Another key factor shaping emotional reactions is a person's relationships with other people. This concept becomes especially important when conversational partners come together multiple times, forming a long-term interaction. Timothy Bickmore and Rosalind Picard mention five relationship models based on social psychology:[7]

- *Dyadic models* define relationship as the interdependency between two people such that a change in the state of one will produce a change in the state of the other.
- *Provision models* are based on what one person provides for the other.
- *Economic models*, such as social-exchange theory, model relationships in terms of costs and benefits.
- *Stage models* assume that relationships go through a fixed set of stages.
- *Dimensional models* attempt to abstract a given relationship's characteristics to a point in a small-dimensional Euclidean space.

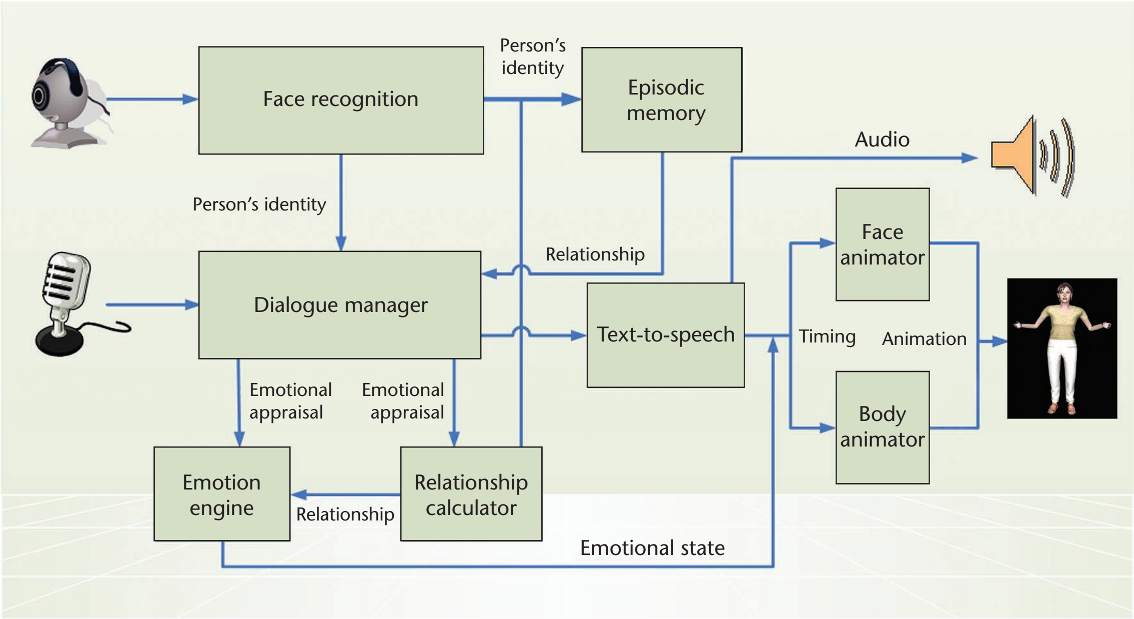


Figure 1. Our system's interaction architecture. The emotion engine uses an emotion model formed by the virtual character's personality, mood, and emotions.

Dimensional models have become the most common way to represent relationships. Michael Argyle's model of relationship, one of the most widely used models of relationship for constructing virtual characters, is based on the dimensions of dominance and friendliness.[8] In his model, dominance refers to one individual's ability to control the resources of another, and friendliness refers to the closeness and friendship level between two people.

Interaction Architecture

Figure 1 shows our system's interaction architecture. We use a layered emotion model, with the emotion engine formed by the virtual character's personality, mood, and emotions. We put mood in the center because we want to simulate the real internal dynamics of an emotional, lifelike character, and we believe that mood is the true representation of internal dynamics. Emotions, personality, and relationships all affect this internal state, on the basis of either outside environmental factors or internal factors of the character itself.

Our research is similar to A Layered Model of Affect (ALMA)[9] in that we use Mehrabian's model to express the relationship between the Ocean personality parameters and the character's three-dimensional PAD mood. However, we differ in that we integrate the interpersonal-relationship concept into our emotion model, in which emotion, mood, personality, and social relationships all affect each other.

We use the OCC model as an emotion model of the interaction between virtual characters and real users. The OCC model also includes agent-related emotions, which we can easily link with Argyle's friendliness-dominance relationship model.[8] We chose Argyle's model because our goal is to create an intelligent, autonomous character that can form relationships with the people it interacts with. The model also shares common dimensions with the PAD mood representation, so we can map the PAD model's pleasure and dominance values to the Argyle model's friendliness and dominance values.

How It Works

Currently, our system uses text-based dialogue input; owing to time constraints, rather than any technical hurdle, we've yet to implement speech input. We handle interactions in sessions. In each session, the system calculates the overall relationship with the user through emotional impulses. For example, if a user says something bad happened to him or her and the virtual character has positive impressions of the user, the resulting emotion will be "sorry-for." If the impressions are negative, the character will experience the "gloating" emotion. The system updates and stores the relationship level related to each user and recalls it in the next interaction session, affecting the virtual character's overall emotional state. A session's emotional effects also decay over time, decreasing the recall probability; the virtual character's traits also impact its ability to remember older emotional memories. For example, a neurotic character will hold on to bad memories longer than a stable character.

Figure 2 shows our system setup. As Figure 2a shows, users wear a head-mounted display (HMD) that includes a small webcam. When users look at the AR marker (see Figure 2b), the system displays the virtual character in front of the marker; the big screen on the wall beside the users reflects what they see (see Figure 2c). The camera above the marker recognizes the user's face using a standard OpenCV Computer Vision Library module.

This face recognition capability lets the virtual character interpret when users want to start or end an interaction session, making the communication more natural. The system stores information about each interaction session—such as the user who participated and any changes in the relationship—in episodic memory.

Episodic Memory

Long-term memory is a central concept in our interaction architecture because we want our virtual humans to store interaction information over time and retrieve it as needed. This long-term memory can be related either to facts (declarative) or skills (procedural). Procedural memory is related to learning skills such as riding a bike or playing a guitar. In our system, we focus more on declarative memory, which is important for natural-language communication.

Declarative memory is either episodic or semantic. Episodic memory represents our experiences as points on a timeline, where each memory entry is associated with a point in time. Endel Tulving defines episodic memory as a neurocognitive (brain/mind) system that is uniquely different from other memory systems that enable human beings to remember past experiences.[10] Semantic memory is derived from episodic memory and is a structured representation of learned facts and concepts.

For our emotion model, virtual characters must remember specific interaction sessions with people at specific times. So, we use episodic memory. For each record in the memory, we store this information:

- the person involved in the interaction, p_i;
- time passed since the session started, t_i;
- the relationship status at the session's start, R_i^s;
- the relationship status at the session's end, R_i^e;
- the relationship affect of the current session, R_i; and
- recall probability, P_i.

By updating the status of a character's mood, emotional state, and relationship, the system alters the character's affective state.

Affective-State Update

Affective-state updates let our virtual character interact with users in a more natural and dynamic way.

Emotional-State Update

The OCC model defines the emotions in relation to the events causing them. Cognitive-appraisal models of emotion explain the overall process of how emotions occur and affect our decision making, but it's also important to have a dynamic model of attitudes toward users to decide which emotion will occur. Typical agents don't include such modeling and thus can't consider emotions such as "happy-for" and "sorry-for." In our model, however, we added to the OCC model four user-related emotion types (happy-for, gloating, sorry-for, and resentment) to create a total of 16 emotional categories (see Table 1, next page).

At design time, during the *quantification phase*, we specify which events cause which emotions, along with the emotions' intensity. In our geography teacher example, the dialogue script has a desired emotional impulse associated with every exchange. So, for example, when the student interacts with the teacher, a polite answer elicits a positive reaction from the teacher, whereas a rude answer elicits a negative reaction. In addition, the virtual character's personality also alters her appraisal of the event.

Our emotion model differs from most models in two respects. First, we don't directly map between Ocean personality traits and OCC emotions but instead use the PAD mood space as an intermediate level. We do this because no existing theory relates OCC emotions to Ocean traits, whereas Mehrabian has described the relation between the PAD mood space and both Ocean traits and OCC emotions.[11] Second, we construct a relationship

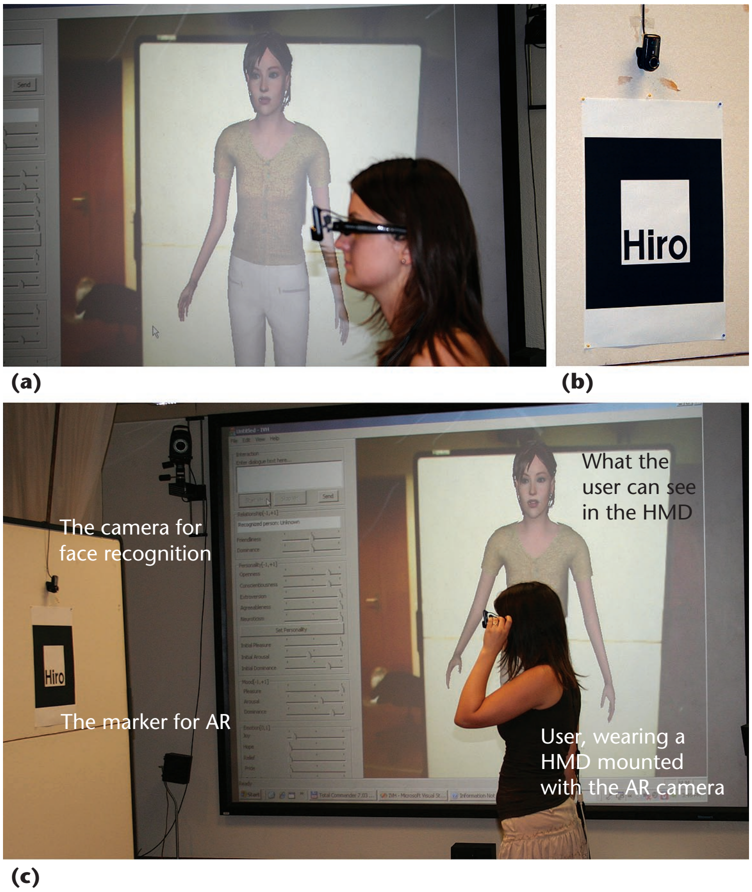


Figure 2. The system setup. (a) The user wears a head-mounted display (HMD) that includes the augmented reality (AR) camera. (b) The AR marker has a face recognition camera above it. The system's face recognition capability lets it interpret when users want to start or stop a session, making communication more natural. (c) The user's view displays on a large screen.

Table 1. The system's emotion labels.

Positive reactions	**Appraised events**
Joy	Because something good happened
Hope	About the possibility of something good happening
Relief	Because a feared bad thing didn't happen
Pride	About a self-initiated praiseworthy act
Gratitude	About an other-initiated praiseworthy act
Love	Because a person finds someone or something appealing
Happy-for	Because something good happened to a liked person
Gloating	Because something bad happened to a person who isn't liked
Negative reactions	**Appraised events**
Distress	Because something bad happened
Fear	About the possibility of something bad happening
Disappointment	Because a hoped-for good thing didn't happen
Remorse	About a self-initiated blameworthy act
Anger	About an other-initiated blameworthy act
Hate	Because a person finds someone or something unappealing
Sorry-for	Because something bad happened to a liked person
Resentment	Because something good happened to a person not liked

Table 2. Mehrabian Mood Types.

Trait combination*	**Mood type**
+P+A+D	Exuberant
-P-A-D	Bored
+P+A-D	Dependent
-P-A+D	Disdainful
+P-A+D	Relaxed
-P+A-D	Anxious
+P-A-D	Docile
-P+A+D	Hostile

**P = pleasure, A = arousal, and D = dominance*

model with each user, so the character's emotional state is affected dynamically by its interaction with different users.

Assuming that the input events are listed, we create an emotion vector with 16 emotions as an initial input to the emotion engine. If we represent the emotional state as E_s, appraised emotions as E_a, and the current mood as M_{cur}, then the emotion update function is

$$\begin{aligned}&\text{If } E_s, E_a = [e_1, \ldots, e_{16}]; e_i \in [0,1]\\ &\text{Then } E_s = E_s + filter(E_a, M_{\text{cur}})\end{aligned},$$

where

$$filter(E_a, M_{\text{cur}}) = E_a + \frac{\sum_{i=1}^{16}\sum_{j=1}^{3}\alpha_{ij} * m_j}{\sum_{i=1}^{16}\sum_{j=1}^{3}\alpha_{ij}}.$$

As we describe in more detail later, α_{ij} is the OCC-to-PAD conversion matrix. Also, e_i represents each emotion type in the emotional state vector, and m_j represents each mood dimension in the mood vector. In our model, mood directly affects the virtual character's emotional state at each emotion appraisal. Relationships with the users also change the affective state, but these changes are triggered during interaction with a new user or when a session ends.

Mood Update

As we mentioned before, we model moods using the pleasure (P), arousal (A), and dominance (D) traits. These traits are independent of each other and form a 3D space. The pleasure-displeasure level relates to the emotional state's positivity or negativity, arousal-nonarousal shows the level of physical activity and mental alertness, and dominance-submissiveness indicates the feeling (or lack) of control. These trait's values lie between the positive (+1.0) and negative (–1.0) ends of each dimension. As Table 2 shows, Mehrabian defines eight mood types based on combinations of negative (–) and positive (+) values for each dimension: pleasant (+P), unpleasant (–P); aroused (+A), unaroused (–A); and dominant (+D), submissive (–D).

Because Mehrabian also defines the relationship between the Ocean personality traits and the PAD space, we can translate the 5D personality vector (P) into a corresponding PAD space mood point:

$$\begin{aligned}&P = (O, C, E, A, N),\ O, C, E, A, N \in [-1, 1]\\ &M_{\text{base}} = (m_1, m_2, m_3),\ m_i \in [-1, 1]\\ &m_1 = 0.21 * E + 0.59 * A + 0.19 * N\\ &m_2 = 0.15 * O + 0.30 * A - 0.57 * N\\ &m_3 = 0.25 * O + 0.17 * C + 0.60 * E - 0.32 * A.\end{aligned}$$

Initially, $M_{\text{cur}} = M_{\text{base}}$; M_{base} is the base or starting mood. When the system updates the emotional state, the mood point shifts in the 3D PAD space. The change is based on which emotion is activated and the emotion's relationship with each mood dimension. Table 3 shows our mapping between OCC emotions and PAD space.[9,11]

We accomplish this emotion-to-PAD-space conversion using a 2D linear operator, OCCtoPAD, which is a 16×3 matrix in which each entry is an α_{ij}, where $i \in [1, 16]$ and $j \in [1, 3]$.

We update the mood at four points:

- at the beginning, when we initialize the character's mood offline with a particular personality;
- at the start of each interaction session when a user is recognized;

- at the end of each interaction session; and
- at each emotional impulse during the dialogue.

For a new session, we update mood as follows:

$$M_{\text{cur}} = (m_1 + r_f,\ m_2 + 0.2,\ m_3 + r_d),$$

where r_f and r_d are the friendliness and dominance dimensions of the relationship vector between the virtual character and the user.

At a session's end, we remove the effect of the relationship with the user from the mood as follows, before updating the current mood:

$$M_{\text{cur}} = (m_1 - r_f,\ m_2 - 0.2,\ m_3 - r_d).$$

Finally, if we've updated the emotional state on the basis of an emotion impulse from dialogue, we follow that update with a mood update:

$$M_{\text{cur}} = M_{\text{cur}} = UpdateMood(E_s, \alpha)$$
$$UpdateMood(E_s, \alpha) = \sum_{i=1}^{3} \sum_{j=1}^{16} e_j * \alpha_{ij} \quad .$$

Relationship Update

We update the relationship on the basis of the cumulative evaluation of the virtual character's overall interaction with the user. For each interaction, positive and negative impulses from the user help construct a relationship between the interacting parties.

We base our affect calculation for each session on the user's emotional appraisals. As we mentioned earlier, in the OCC model, these appraisals can occur owing to events in the environment, other agents, and the character's likes and dislikes. To determine whether the occurring emotion's source is another agent, we consider only the effect of specific emotions related to other agents: gratitude as positive, and anger as negative. During an interaction session, the system checks each emotional state to see whether it contains gratitude or anger and then calculates the session's overall affect as a sum of the values of these emotions. According to the OCC model, gratitude and anger are positive and relative versions of the other-agent-related emotions. Gratitude has a positive affect on friendliness and a negative effect on dominance, whereas anger has the opposite effect. For completeness, we also add general versions of good and bad emotions—"joy" and "distress"—as Table 4 shows.

A relationship **R** is a vector consisting of two values, (r_f, r_d). For each interaction session, we first calculate a recall probability to decide each session's effect on the current session. As time passes, the effect of earlier interaction sessions decreases; we model this temporal effect, which has a constant relation to the character's personality, using an exponential function.

If $P_i \in [0, 1]$, $t_i \geq 0$, then $P_i = e^{-t/A}$. Here, P_i is the recall probability associated with session i and t_i is the time passed since i started. Also, A is a constant, given by A = (N + 1)/2, A $\in$ [0, 1], and N is the neuroticism value from the character's 5D Ocean personality vector. The constant A decides the decay curve's shape.

For each user, n represents his or her total number of interaction sessions before the current session, which is n + 1. If the starting relationship for a session i is $\mathbf{R}_i^s$ and the ending relationship is $\mathbf{R}_i^e$, then the session's relationship is $\mathbf{R}_i = \mathbf{R}_i^e - \mathbf{R}_i^s$. For session n + 1, the starting relationship is

$$\mathbf{R}_{n+1}^s = \frac{\sum_{i=1}^{n} \mathbf{R}_i * P_i}{\sum_{i=1}^{n} P_i}$$

and the relationship at the end of the session is $\mathbf{R}_{n+1}^e = \left(r_f^s, r_d^s\right)$ and updated as

Table 3. Mapping from OCC* emotions to PAD space.

Emotion	Pleasure	Arousal	Dominance	Mood type
Joy	0.40	0.20	0.10	+P+A+D Exuberant
Hope	0.20	0.20	–0.10	+P+A–D Dependent
Relief	0.20	–0.30	0.40	+P–A+D Relaxed
Pride	0.40	0.30	0.30	+P+A+D Exuberant
Gratitude	0.40	0.20	–0.30	+P+A–D Dependent
Love	0.30	0.10	0.20	+P+A+D Exuberant
Happy–for	0.40	0.20	0.20	+P+A+D Exuberant
Gloating	0.30	–0.30	–0.10	+P–A–D Docile
Distress	–0.40	–0.20	–0.50	–P–A–D Bored
Fear	–0.64	0.60	–0.43	–P+A–D Anxious
Disappointment	–0.30	0.10	–0.40	–P+A–D Anxious
Remorse	–0.30	0.10	–0.60	–P+A–D Anxious
Anger	–0.51	0.59	0.25	–P+A+D Hostile
Hate	–0.60	0.60	0.30	–P+A+D Hostile
Sorry–for	–0.40	–0.20	–0.50	–P–A–D Bored
Resentment	–0.20	–0.30	–0.20	–P–A–D Bored

**Ortony, Clore, and Collins model*

Table 4. OCC other-agent-related emotions and their effect on relationship dimensions.

OCC other–agent related emotions	Friendliness	Dominance
Gratitude	↑	↓
Anger	↓	↑
Joy	↑	↑
Distress	↓	↓

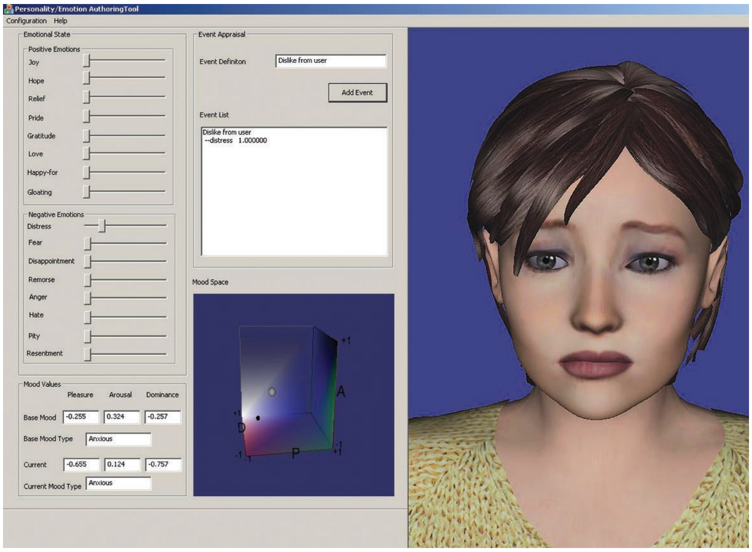


Figure 3. The offline visual simulator. During the quantification phase, we tested Eva's emotion, mood, and personality in this simulator, which let us observe and analyze Eva's responses to events.

$$r_f^e = r_f^s + \sum_{j=1}^{k}\left(E_S[\text{Gratitude}] * \alpha_{[5][1]}\right) + \sum_{j=1}^{l}\left(E_S[\text{Anger}] * \alpha_{[13][1]}\right) + \sum_{j=1}^{m}\left(E_S[\text{Joy}] * \alpha_{[1][1]}\right) + \sum_{j=1}^{n}\left(E_S[\text{Distress}] * \alpha_{[8][1]}\right)$$

$$r_d^e = r_d^s + \sum_{j=1}^{k}\left(E_S[\text{Gratitude}] * \alpha_{[5][3]}\right) + \sum_{j=1}^{l}\left(E_S[\text{Anger}] * \alpha_{[13][3]}\right) + \sum_{j=1}^{m}\left(E_S[\text{Joy}] * \alpha_{[1][3]}\right) + \sum_{j=1}^{n}\left(E_S[\text{Distress}] * \alpha_{[8][3]}\right)$$

where k, l, m, and n represent the number of emotional states with "gratitude," "anger," "joy," and "distress," respectively, during the current session.

Emotion and Mood Decay

Another important factor is the affective state's deterioration over time. Following an event, emotions return to their normal state in a relatively short period, and decay is based on both the emotion's intensity and the character's personality.

Personality defines the individual's control over his or her emotions and is related mainly to the neuroticism trait. For people who are more neurotic, positive emotions disappear more quickly and negative emotions disappear more slowly; the opposite is true for people with more stable personalities. We model the decay of emotions and mood with an exponential curve similar to the one we use to model the recall probability of interaction episodes in memory.

(a) **(b)**

Figure 4. Face captures made by the face recognition module. Even when users are wearing the head-mounted display (HDM), the system easily recognizes (a) Maher, the "good" student, and (b) Zerrin, the "bad" student, because the HMDs leave enough facial features visible for recognition.

Eva Teaches Geography

To implement Eva, we first tested her internal emotion, mood, and personality mechanism offline during the quantification phase. We did this testing in our visual simulator of Eva's mind, which lets us observe and analyze her emotional response when we send her preappraised events. Figure 3 shows a snapshot of this process. We then scripted the dialogues with their corresponding emotional impulses.

In this scenario, two users acted as students. One played a good student named Maher, and the other played a difficult student named Zerrin. The interactions began with Maher, who put on the HMD and stood before the face recognition camera. At this point, Maher could see a life-sized image of Eva in front of him. The face recognition module automatically detected that a new person had arrived, so Eva greeted the student. Eva then captured images of Maher's face to create an identity profile of him in her memory. As Figure 4 shows, the face recognition module can identify subjects easily—even when they're wearing the HMD.

Once Eva recognizes a user, the new interaction session begins. In our scenario, Eva tried to teach the students basic geography concepts. (As

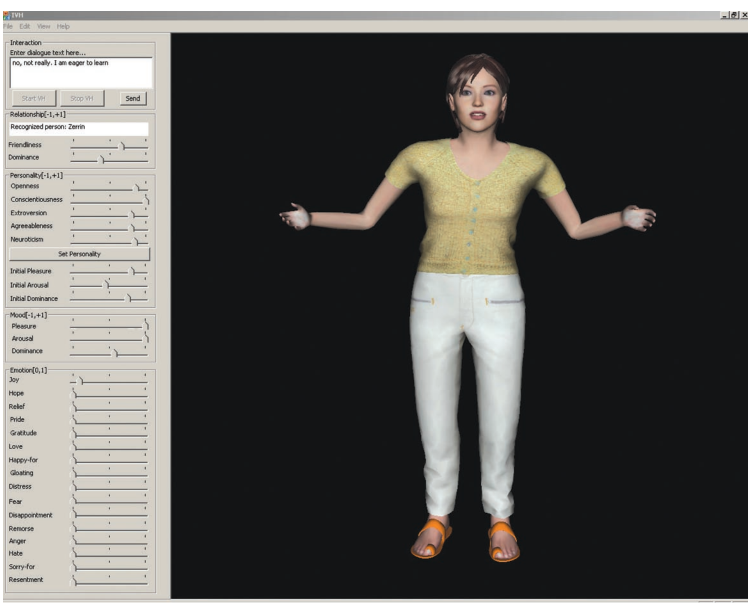


Figure 5. Eva's expressiveness. Eva can speak and show emotion using facial expressions and gestures in response to users. Although the current gesture database is quite small, transitions between gestures are smooth, increasing the interaction's naturalness.

we mentioned before, because our speech recognition isn't fully functional, a facilitator types in the user's answers.) As Figure 5 shows, Eva can speak and show emotion via her facial expressions and gestures. Our current gesture database is quite small. So, Eva's gestures can seem a bit repetitive, but they do transition smoothly from one to the next, enhancing the interaction's naturalness.

With Maher, Eva stayed calm and polite and responded accordingly. Zerrin was rude, so Eva was more aloof in her responses. Figure 6 shows a snippet from the two dialogues. Sessions automatically end when the face recognition camera no longer sees the user for an extended time period.

Some time later, the two students again interacted with Eva. The face recognition module automatically recognized both students when they appeared in front of the camera again. As the snippets in Figure 7 show, Eva remembered each student's attitude and responded accordingly.

Although Eva remembered the previous interaction, she wasn't rude to the difficult student because a good teacher wouldn't be rude. We have, however, modeled her personality suitably to bring out the differences between the two interactions more starkly. Developers can adjust such qualities during the authoring phase according to a system's intended applications—which are many and diverse within the serious game field—and they can use these adjustments to produce many interesting characters.

We videotaped the interaction sessions with both the good and difficult student. A video of the interaction sessions, along with a screen capture of the application, is available at http://personal.miralab.unige.ch:1312/z.kasap/IEEECG&A_SeriousGames/CGA_video.wmv.

Creating Expressiveness

We store Eva's visemes—that is, her facial movements—for each type of emotional response in a database and encode the expressions in an MPEG4 FAP (Facial Animation Parameters) format. The system reads the body gestures from Collada files containing the body animations; we blend the animations to allow smooth transition from one to the other. The blending is done efficiently based on dual quaternions.[12] The system then synchronizes the animation with the timing information generated by the Acapela's text-to-speech engine to coordinate Eva's expressions with her dialogues.

Moving Beyond AR

We placed Eva in an AR environment. Users look toward a marker that the system tracks using the ARToolkit library, and we place a life-sized model

First interaction with a good student

Eva: Nice to meet you Maher. I would like to start with a few simple concepts first. Do you know about different landforms on the Earth?

Student: No, not really. I am eager to learn.

Eva: Ok, today we will talk about three types of landforms: cape, strait, and gulf. Then we will have a small test to check if you understood the concepts. Is it ok for you?

Student: Yes, fine for me.

Eva: Good. Let's start with the cape. A cape is a pointed piece of land that sticks into the sea, ocean, lake, or river. A good example is the Cape of Good Hope to the south of Cape Town. Do you know what a strait is?

Student: No.

First interaction with a difficult student

Eva: Nice to meet you Zerrin. I will start with a few simple concepts first. Do you know about different landforms on the Earth?

Student: I don't know anything about them. Boring topics!

Eva: Ok, let me explain some of them to you. It is not that boring, trust me.

Student: If you say so.

Eva: Today we will talk about three types of landforms: cape, strait, and gulf. Then we will have a small test to check if you understood the concepts. Is it ok for you?

Student: You teachers always want to take tests.

Eva: It is important for you to learn these concepts. Let's start with the cape. A cape is a pointed piece of land that sticks into the sea, ocean, lake, or river. A good example is the Cape of Good Hope to the south of Cape Town. Do you know what a strait is?

Student: No.

Figure 6. Example dialogue from the first interaction. As these snippets show, Eva initially addressed both users in the same manner but adjusted her responses according to how the users responded.

Second interaction with a good student

Eva: Hi Maher, great to see you!

Student: Hi Eva!

Eva: You were very good in our last class.

Student: Thank you. You are a nice teacher too.

Eva: You really think so? Thank you. Are you ready to learn more things today?

Student: I don't feel well today. Maybe it is better if we continue later on.

Eva: Oh, I am really sorry to hear that. Ok. See you another time. Bye.

Student: Bye.

Second interaction with a difficult student

Eva: Hi Zerrin! You are back again? I thought you were not very interested in geography.

Student: Hi Eva! I am sorry for last time. I will try my best from now on.

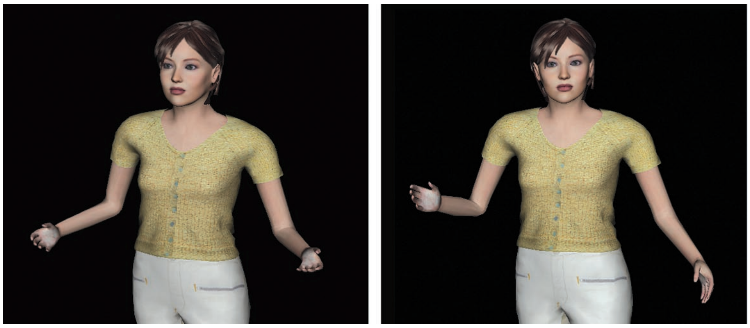
Eva: Well, I hope so. Ok, Lets start the lesson then.

Student: I don't feel well today. Can we continue later on?

Eva: I hope this is the truth. See you later then. Bye.

Eva: Bye.

Figure 7. Example dialogue from the second interaction. As these snippets show, Eva remembered the previous interaction with each student and greeted them accordingly, adjusting her responses to suit both the previous interaction and the students' current responses.



(a) (b)

Figure 8. The effect of self-adaptive animation. (a) Without self-adaptive animation, the character's position is unaffected by changes in the user's position. (b) With self-adaptive animation, Eva automatically adapts her position to changes in the user's position, making her seem more aware of the user and thus enhancing the interaction's naturalness.

of Eva at the tracked location. We go beyond ordinary AR, however, in that we've integrated user-perspective-based self-adaptive animation of our virtual character.[12] This enhances user interaction, in that the body animation automatically adapts itself to changes in the user's position. In our example, Eva always looks toward the user, and her body animation adapts by blending according to the changes in the user's position. These position changes typically aren't large; the user must stay in view of the face recognition camera. However, as Figure 8 shows, our approach does make Eva seem more aware of the user's presence.

As our scenarios indicate, our system is a step in the right direction in the fast-developing area of serious games. It has numerous potential applications, from better virtual trainers to more intelligent characters in video games and interactive media.

Adding a speech recognition module will make our system more complete. We'd also like to increase our gesture vocabulary to enrich the animation. In terms of Eva's internal mechanism, we'd like to give her more awareness about the environment and her role in it. For example, in this scenario, Eva has no notion of being a geography teacher; she's just playing out scripted dialogues. We'd like to augment this by developing a sense of self for the character. In addition, we want to explore using semantic memory to enhance her memory model.

Acknowledgments

We thank Nedjma-Cadi Yazli for creating the example animations and helping create the resulting videos that were crucial to this research. The European research project Indigo (IST-045388) funds our project.

References

1. J. Gratch and S. Marsella, "Lessons from Emotion Psychology for the Design of Lifelike Characters," *Applied Artificial Intelligence*, vol. 19, nos. 3–4, 2005, pp. 215–233.
2. P. Ekman, *Emotion in the Human Face,* Cambridge Univ. Press, 1982.
3. A. Ortony, G.L. Clore, and A. Collins, *The Cognitive Structure of Emotions,* Cambridge Univ. Press, 1988.
4. A. Ortony, "On Making Believable Emotional Agents Believable," *Emotions in Humans and Artifacts*, R. Trappl et al., eds., MIT Press, 2003, pp. 189–212.
5. A. Steed et al., "Building Expression into Virtual Characters," *Ann. Conf. European Assoc. for Computer Graphics, State of the Art Reports*, 2006; www.cs.ucl.ac.uk/staff/M.Gillies/expressivevirtualcharacters.pdf.
6. S. Hampson, "State of the Art: Personality," *The Psychologist*, vol. 12, no. 6, 1999, pp. 284–290.
7. T.W. Bickmore and R.W. Picard, "Establishing and Maintaining Long-Term Human–Computer Relationships," *ACM Trans. Computer–Human Interaction*, vol. 12, no. 2, 2005, pp. 293–327.
8. M. Argyle, *Bodily Communication,* 2nd ed., Methuen, 1998.
9. P. Gebhard, "ALMA—A Layered Model of Affect," *Proc. 4th Int'l Joint Conf. Autonomous Agents and Multi-Agent Systems* (AAMAS), ACM Press, 2005, pp. 29–36.
10. E. Tulving, "Episodic Memory: From Mind to Brain," *Ann. Dutch Rev. of Psychology*, vol. 53, 2002, pp. 1–25.
11. A. Mehrabian, "Analysis of the Big-Five Personality Factors in Terms of the PAD Temperament Model," *Australian J. of Psychology*, vol. 48, no. 2, 1996, pp. 86–92.
12. P. Chaudhuri, G. Papagiannakis, and N. Magnenat-Thalmann, "Self-Adaptive Animation Based on User Perspective," *The Visual Computer*, vol. 24, no. 7, 2008, pp. 525–533.

Zerrin Kasap *is a PhD candidate and research assistant at MIRALab, University of Geneva. Her research interests are in aspects of social interaction with intelligent agents, both virtual humans and robots, and specifically in emotion and personality modeling, memory simulation, and expressive facial animation. Kasap received her MSc in computer science and engineering from Dokuz Eylul University. Contact her at zerrin.kasap@miralab.unige.ch.*

Maher Ben Moussa *is a PhD candidate and research assistant at MIRALab, University of Geneva. His research interests are in interacting with conversational agents for both virtual humans and robots, focusing on facial expressions, gaze, dialogue man-*

agement, and emotion and personality. Ben Moussa received his MSc in computer science and engineering from the University of Twente. Contact him at maher.benmoussa@miralab.unige.ch.

Parag Chaudhuri *is a postdoctoral researcher at MIRALab, University of Geneva. His primary research interests include computer animation, rendering, motion capture, real-time computer graphics, and mixed reality. Chaudhuri received his PhD in computer science and engineering from IIT Delhi and received IBM's Outstanding PhD award. Contact him at parag.chaudhuri@miralab.unige.ch.*

Nadia Magnenat-Thalmann *is a professor at the University of Geneva and the director of MIRALab. She pioneered the virtual-humans field—particularly the facial-animation and emotion domain—as a partner in several European projects. Magnenat-Thalmann received her PhD in quantum physics from the University of Geneva. She's editor in chief of* The Visual Computer *and is an associate editor of* IEEE Transactions on Multimedia. *Contact her at nadia.thalmann@miralab.unige.ch.*