

User's Gestural Exploration of Different Virtual Agents' Expressive Profiles

Matthieu Courgeon, Jean-Claude Martin & Christian Jacquemin

LIMSI-CNRS +33.1.69.85.81.04 BP 133, 91403 Orsay

France

`martin@limsi.fr; courgeon@gmail.com; Christian.Jacquemin@limsi.fr`

Designing affective user interfaces involving expressive virtual characters raises several research questions. From a computer science point of view, the character should be able to display facial expressions of complex emotions as dynamic and realtime reactions to user's inputs. From a cognitive point of view, designers of virtual characters need to know how the user will perceive the dynamics of these facial expressions in relation with her/his own input.

There has been already a long history of computational models of facial expressions in virtual characters as well as psychological studies showing the importance of the dynamics of facial expressions (Frank *et al.*, 1993). Several animation techniques were proposed for talking heads involving model-based and image-based approaches (Bailey *et al.*, 2003). In order to go beyond the display of individual basic emotions, models are defined for the facial display through so-called blends of emotion or nonarchetypal expressions (Ekman & Friesen 1975; Tsapatsoulis *et al.*, 2002; Albrecht *et al.*, 2005; Niewiadomski 2007). Interpolation algorithms have been proposed for generating facial expressions for a continuum from pure to mixed emotions of varying intensity (Albrecht *et al.*, 2005). MPEG-4 is a standard for facial animation (Pandzic & Forchheimer 2002) that researchers use to specify both archetypal facial expressions and facial expressions of intermediate emotions (Tsapatsoulis *et al.*, 2002; Malatesta *et al.*, 2007). Rather than interpolating two expressions, models were also proposed that combine different areas of the face to display blends of emotions (Ekman and Friesen 1975; Niewiadomski 2007). Experiments were conducted to study individual differences in users' perceptions of blended emotions from virtual characters expressions (Buisine *et al.*, 2006; Niewiadomski 2007). Layered models were defined for relating facial expressions of emotions on one hand, and on the other hand moods and personality traits using three different timescales (Kshirsagar 2002; Gebhard 2005).

A few affective computing studies have combined the interaction between the user's input and the display of expressive signals by the system. Gesture input via a tablet is suggested as a relevant modality for realtime and interactive control of expressive speech (D'Alessandro *et al.*, 2005). An experiment used an anthropomorphic tangible interface to dynamically control the animation of a 3D face (Jacquemin 2007) and shows that the tactile control of a face can be experienced as a unique affective communication medium. The Sentoy enables the user to express emotion using tactile input (Paiva *et al.*, 2002). All these studies suggest that gesture can be used as a means for exploring the space of facial expressions that a given virtual character can convey.

The PAD space (Pleasure, Arousal, Dominance) can be seen as a framework for the description and measurement of emotional states as well as temperamental dispositions to certain emotional responses (see Mehrabian (1996) for an overview). The three dimensions of this model are: Pleasure (*i.e.* positive versus negative affective state), Arousal (*i.e.* level of physical activation and/or mental alertness), and Dominance (*i.e.* feelings of control and influence over others and situations, versus feeling controlled and influenced by external circumstances). The PAD space is used both for experimental studies in Psychology (*i.e.* mapping of emotion terms onto the three dimensions (Mehrabian 1996)) and in computational models of emotions for virtual characters (Kshirsagar, 2002; Gebhard, 2005; Becker *et al.*, 2006).

The above mentioned studies did not focus on the fine-grained dynamics of interaction between the user and an expressive character. In this paper we present the design of a software platform that enables virtual characters to display blended facial expressions of emotions as realtime reactions to users' input. We aim at mixed realistic and artistic applications (Martin *et al.*, 2007) which require high quality rendering (detailed head model, wrinkles, and layered skin rendering techniques enabling the character to blush or to turn pale). We also use this platform to understand how users perceive and

react to blends of facial expressions of emotions that a given virtual character is able to display. We are interested in the gesture input modality since it seems to be a relevant media for low-level realtime interaction in mixed reality applications.

A PLATFORM FOR A REALTIME 3D INTERACTIVE AGENT

The purpose of this platform for real-time rendering of an interactive agent is to offer a cohesive and flexible framework that combines high quality and expressive visual rendering, and interactive control of facial expressions:

- *flexibility* results from the use of MPEG-4 visual animation encoding that enables various face models animation,
- *high quality visual rendering* is obtained through shader implementation of multi-layered skin model,
- *expressivity* is obtained by enhancing mesh animation with visual signs of emotions such as blushing and blemishing, frowning, and crows-foot wrinkles.

Animation is carried out through Virtual Choreographer (<http://virchor.sf.net>), an open source 3D engine. Animation is encoded in VirChor using MPEG4 that ensures compatibility between different face models and facilitates the reuse of animation tables. Only a minimal computation of blend coefficients is made in the CPU and passed to the graphic card in order to overcome the bottleneck of data transmission through the graphic bus.

GESTURE EXPLORATION OF EXPRESSIVE PROFILES

Our goal is to investigate how a user can perceive the expressive profile of a virtual character during realtime interaction using low-level gesture classical devices such as a joystick or similar devices that are well-known to users. Our purpose is to design applications in which the user should be able to perceive and react on implicit visual-only facial expressions of complex emotions (game, mediated communication, ...). Three dimensions enable to navigate a rich set of emotional expressions.

The PAD space (Mehrabian 1996) is relevant to our research goals for practical reasons since we want to study user's exploration of a continuous space of blended expressions of emotions, and its three dimensions can be easily mapped onto a 3D gesture device. In our experimental approach, it is the user who decides when and where to move the state of the agent in the PAD space. We selected 8 affective state labels to represent the corners of this 3D cube: *fear*, *distress*, *anger*, *reproach*, *joy*, *relief*, *satisfaction*, and *admiration*. We interpret a point in this 3D space as a blend of these 8 emotions. Via a joystick, the user moves a 3D point in the PAD space. The 3D coordinates (x, y, z) captured from the joystick are mapped onto 8 PAD activation values ($N1, \dots, N8$) for each of the corners of the PAD space. In order to test if the user is able to perceive different emotional expression profiles, we implemented different filters which transform these 8 coordinates into 8 values ($C1, \dots, C8$). These 8 modified activation values are sent to the VirChor graphical engine which maps them to FAPs, which are finally sent to the graphical processing unit for rendering.

Using the graphical platform described in the previous section, we defined static FAPs tables for each of the 8 expressions of emotion at the corners of the PAD cube. These tables were stored into XML tables which are loaded as weighting coefficients in GPU at compile time. We were willing to use low level gesture interaction in order to study the impact of realtime interactivity on the perception that the user has over the expressive profile of the agent. Through the three dimensions of a joystick (including joystick vertical rotation), the user can explore the PAD space of emotional expressions.

The initial position of the joystick is (0, 0, 0) and is associated with the neutral expression in the PAD space. The mapping of a 2D point M onto a 4D point N : ($N1, N2, N3, N4$) in the emotions space is defined by four vectors ($E1, E2, E3, E4$). In order to compute N , we use a function named *Act* for Activation.

The same principle applies when using a PAD 3D cube instead of a 2D square. We get 8 values from a three dimensional point. Based upon our virtual character that reacts in realtime to user's gestures on a joystick, we have evaluated how users perceive different expressive profiles. Our hypothesis is that such real time and dynamic interaction enables the user to perceive the expressive profile defined for the agent. An expressive profile defined for the agent constrains user's action in the PAD cube. Six expressive profiles are defined along three dimensions: 1) Expressivity (Low / High), 2) Speed (Slow / Fast), and 3) Valence (Negative / Positive). These six expressive profiles are selected because of their potential to be perceived via low-level gesture interaction.

We define an expressive profile as a set of attributes for each edge/emotion of the PAD cube: an increment rate for the attack period, a decrement rate for the decay, and a Bezier curve for computing the final activation of the expressed emotion as a modulation of user's action on the joystick. Thus, from the user's actions on the joystick, we compute a modulated target in the PAD space, and the activation of the expressed blend of emotional expressions moves toward this target, using the dynamics defined by the attack, decay and Beziers parameters. The expressive profile assigned to the agent modulates the activation of a facial expression as follows. Let N_i be the user-defined activation of an emotion I . Let the expressive profile of the current agent be defined using eight Bezier functions (one for each emotion) called B_i . The activation target of this emotion for this expressive profile is computed as: $T_i = B_i(N_i)$.

The current modulated activations of the 8 emotions are called C_i . C_i depends on the T_i target positions, and on the increase and decrease rates defining attack and decay. The increase rate INC , and decrease rate DEC have the same purpose, except that INC will be used if $C_i < T_i$, and DEC will be used if $C_i > T_i$. For example, if $C_i=0$, $T_i=1$, and $INC_i=p$, it will take $10/p$ frames for C_i to reach T_i . With a 100% increase rate, it will take at least 10 frames to stabilize C_i , and at 60 frames per second, 0.166 second, which is a very fast reaction. With a 1% ratio, it will take 1000 frames, which will last 16.66 second, which would be a slow reaction to user's action. Once computed, the 8 PAD activation values are sent to VirChor using UDP communication and VirChor computes the combined FAPs table for facial animation:

$$\text{Displayed FAPs table} = \sum_{i=1}^8 C_i \times (\text{FAPs Table})_i$$

Where C contains the PAD values and the FAPs tables are the displacement tables loaded in VirChor at startup.