

Tools to Create Individual ECAs

Benjamin Dariouch - Nicolas Ech Chafai - Maurizio Mancini - Catherine Pelachaud
IUT de Montreuil
University of Paris 8

1. Introduction

We aim at creating Embodied Conversational Agents able to communicate with individualized behaviors rather than with some constant deterministic actions. More precisely, starting from an agent able to show some basic emotions through facial expression, we alter these expressions based on parameters that embed factors such as agent's personality and contextual environment. To undertake this work we first analyze the dynamism of human emotional behaviors, in particular, how facial muscles contract during emotional states. In this presentation we will present the analysis we conducted on motion capture data of actors. Then we will explain tools we have developed to allow one to create new facial MPEG4 compliant models into our agent environment called Greta.

2. Motion capture data analysis

We currently use a system based on key frame animation, where an expression is defined by 3 temporal parameters, namely onset, apex and offset. But such a specification does not allow one to capture the subtlety of facial expression dynamism. In order to improve these animations, we are studying real data of facial movements coming from motion capture sequences we have recorded using a Vicon system at the University of Rennes. Data acquisition has been made with the Oxford Metrics Vicon software. Our data are organized into 78 sequences performed by two actors, a man and a woman, each having 33 markers on the face, 21 of which correspond to FAPs (Facial Animation Parameter) locations. These sequences are simple basic movements, like raising eyebrows or smiling, and basic emotions such as anger, happiness, surprise. Finally we recorded two sequences of monologues in which extreme expressions of emotions were displayed.

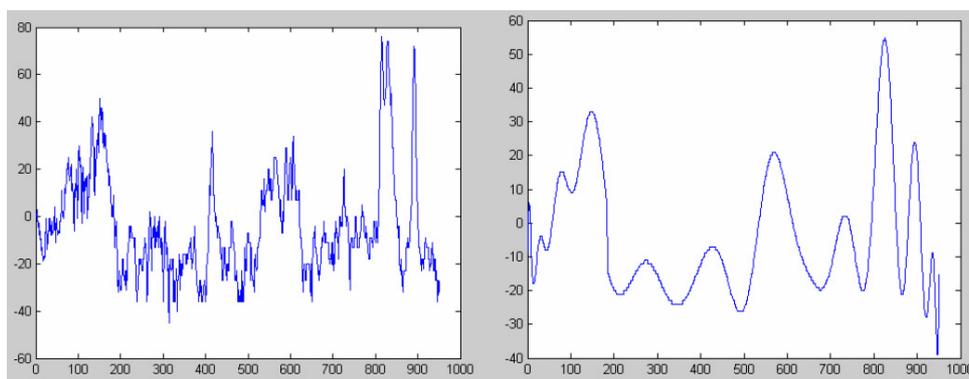


Figure 1 – unfiltered (left) and filtered (right) data

Raw captured data have been filtered at first using a Savitzky-Golay filter (see Figure 1 for comparison between non-filtered and filtered data). Then head movements have been taken out from markers displacement data using some referring markers placed on the top of the actors' heads,

so that all the markers data has been made relative to the origin of the head-system. After we analyzed the displacement of subsets of the markers, especially the ones relative to the eyebrows and the mouth regions. By normalizing markers movements using the actors' FAPUs (Facial Animation Parameter Units) we also generated the corresponding FAP animations that we tested on our agent. In general we noticed that FAPs from the same facial area, like the four ones describing the movement of an eyebrow, have usually the same movement with a proportional factor (see Figure 2).

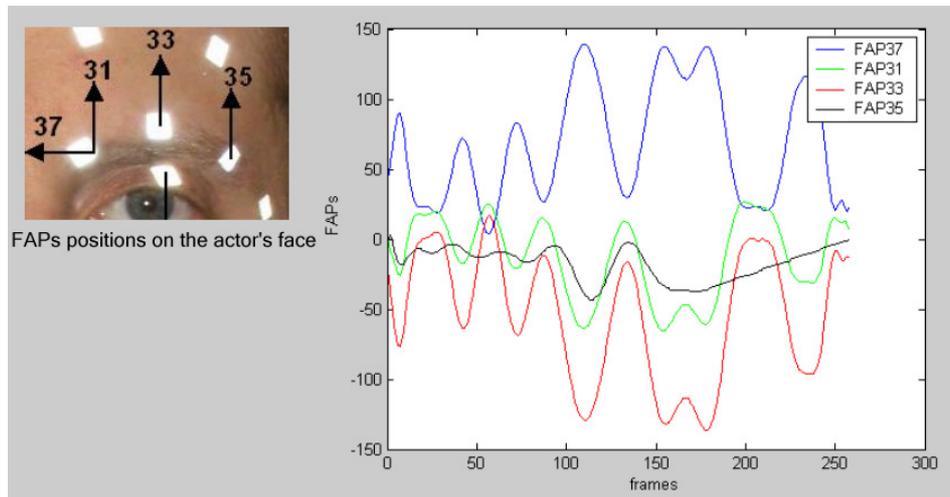


Figure 2 – captured data from the eyebrow region

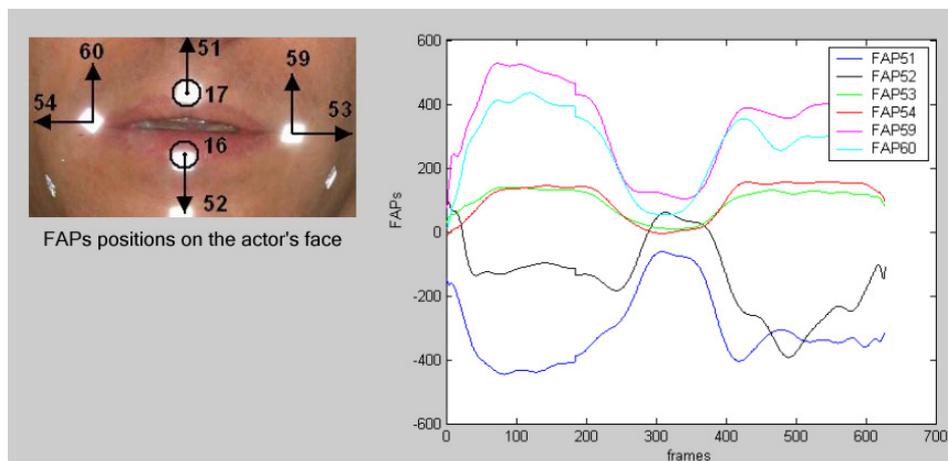


Figure 3 – captured data from the mouth region

Another example of captured data is shown in Figure 3. It reports the curves for the FAPs of the mouth region while the actor was performing a smile.

2.1 Expressions dynamism

Facial animation generally follows either the Onset-Apex-Offset model (see Figure 4 left) or the Attack-Decay-Sustain-Release model (see Figure 4 right). These models simulate how a facial expression raises (and decays) on the face. The ADSR model has the advantage to simulate more

dynamic behaviors. We have decided to use such a model in our system.

Our facial animation engine generates FAP curves using ADSR (Attack, Decay, Sustain, Release or an arbitrary sequence of them) based on Hermite interpolation. To generate the FAPs values, we determine the desired phases (A, D, S or R), an intensity and a duration value. The FAP values are then calculated with control points linked by a Piecewise cubic Hermite interpolation. The model can also be reduced to the simplest onset-apex-offset just by setting Decay=0. After this modifications we have conducted some analysis on data coming from both actors (to consider individual differences) comparing it to the animations generated by our engine.

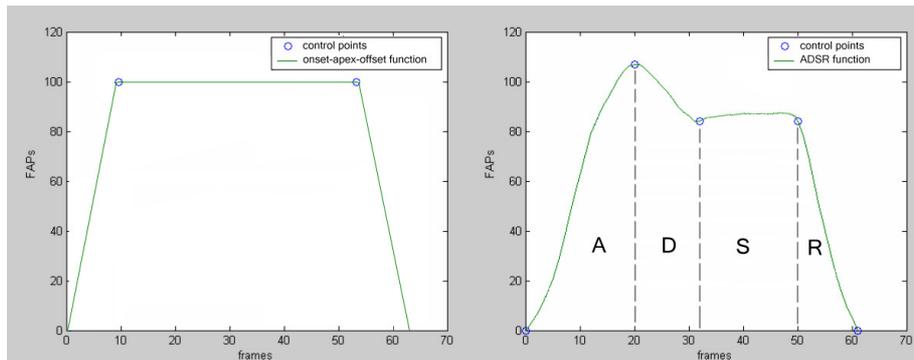


Figure 3 – onset apex offset (left) ADSR (right)

3. Tools to model MPEG4 faces

The face model of our Embodied Conversational Agent environment called Greta has been developed following the MPEG4/FAP standard directions. Until now the definition of the face was embedded in the code itself and it was not possible to easily modify it. To allow the creation and implementation of many different faces we developed a tool that is able to automatically import mesh models made under some commercial modelling programs (such as Poser) and then to generate MPEG-4 compliant facial models. After generation the models can be used in any MPEG-4 application.

Our tool has a graphical interface (see Figure 5) and most of the user input is made by mouse movement/selection. After the head models as been loaded from the 3D data file, there are two consecutive operations that have to be performed: first, one has to select the vertices corresponding to each FDP; second, the area of influence of each FDP has to be specified. Each area of influence corresponds to a sub-mesh of the face mesh (see Figure 6). These two steps are performed

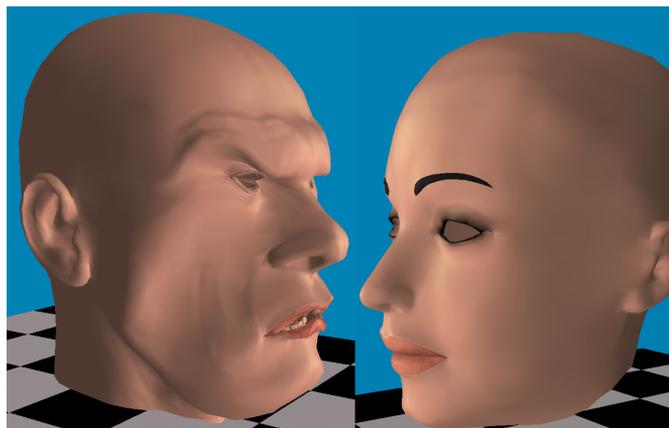


Figure 5 – different head models for our player

by selecting vertices and polygons on the 3D facial model using the mouse. To further enhance usability, two semi-automatic selection mechanisms have been implemented. The first one, given an approximative selection made by the user (that may contain holes, non closed areas) computes an "optimised" selection, that is the one the user "probably" wanted to do. This mechanism uses the "convex hull algorithm" which allows to calculate the smallest area containing all the selected faces. The second mechanism is based on MPEG-4 recommendations, so that the tool, given a specific region of influence, will always propose only the subset of FDPs that "have to" correspond to this region. For example, if the user wants to select the tongue-FDPs, only 6.1 to 6.4 are made available.

Finally we are extending MPEG4 specification to have a better reproduction of an animation on different facial models. MPEG-4 usually considers 5 FAPUs for the given facial model, so each FAP value is modulated by one of the FAPUs depending on the facial region the FAP acts on; in that way the same FAP values will give the same results on faces with different physical proportion. But these 5 FAPUs may be not always enough to take into account some important geometry variations in facial models. We are extending the original set of FAPUs introducing new units (always FAPU – like) that refine the specification of any new facial model. Examples of such units define the forehead orientation, lip thickness and so on.

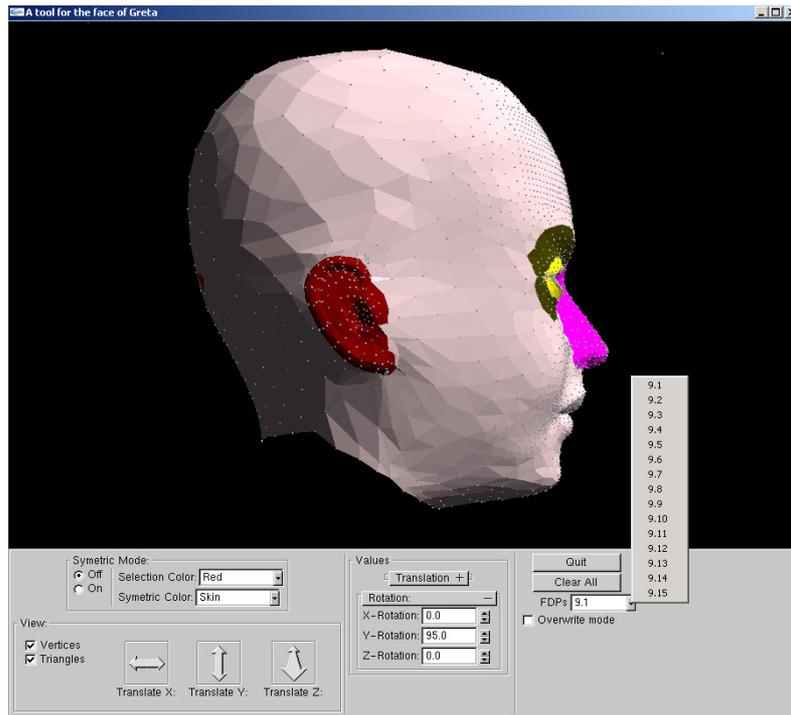


Figure 6 – the tool's graphical interface

4. Conclusions

Aiming at refining the behavior of conversational agents we started to conduct some captured data analysis. For this purpose, the WP4 knowledge base may be very useful. We believe that is possible to adapt techniques originally made to understand user's signals to the captured data analysis we are conducting, to get some improvements to our agent's behavior model.