INTERFACE toolkit: a new tool for building IVAs

Piero Cosi, Carlo Drioli, Fabio Tesser, Graziano Tisato

Istituto di Scienze e Tecnologie della Cognizione
Sezione di Padova "Fonetica e Dialettologia"
Consiglio Nazionale delle Ricerche
Via G Anginoni, 10 - 35121 Padova, ITALY
{ cosi, drioli, tesser, tisato }@pd.istc.cnr.it
http://www.pd.istc.cnr.it

Abstract. INTERFACE is an integrated software implemented in Matlab© and created to speed-up the procedure for building an emotive/expressive talking head. Various processing tools, working on dynamic articulatory data physically extracted by an optotracking 3D movement analyzer called ELITE, were implemented to build the animation engine and also to create the correct WAV and FAP files needed for the animation. By the use of INTERFACE, LUCIA, our animated MPEG-4 talking face, can copy a real human by reproducing the movements of passive markers positioned on his face and recorded by an optoelectronic device, or can be directly driven by an emotional XML tagged input text, thus realizing a true audio/visual emotive/expressive synthesis. LUCIA's voice is based on an Italian version of FESTIVAL - MBROLA packages, modified for expressive/emotive synthesis by means of an appropriate APML/VSML tagged language.

1 Introduction

Emotions are quite important in human interpersonal relations and individual development. Linguistic, paralinguistic and emotional transmission are inherently multimodal, and different types of information in the acoustic channel integrate with information from various other channels facilitating communicative processes. The transmission of emotions in speech communication is a topic that has recently received considerable attention, and automatic speech recognition (ASR) and multimodal or audio-visual (AV) speech synthesis are examples of fields, in which the processing of emotions can have a great impact and can improve the effectiveness and naturalness of man-machine interaction.

Viewing the face improves significantly the intelligibility of both natural and synthetic speech, especially under degraded acoustic conditions. Facial expressions signal emotions, add emphasis to the speech and facilitate the interaction in a dialogue situation. From these considerations, it is evident that, in order to create more natural talking heads, it is essential that their capability comprises the emotional behavior.

In our TTS (text-to-speech) framework, AV speech synthesis, that is the automatic generation of voice and facial animation from arbitrary text, is based on parametric descriptions of both the acoustic and visual speech modalities. The visual speech synthesis uses 3D polygon models, that are parametrically articulated and deformed, while the acoustic speech synthesis uses an Italian version of the FESTIVAL diphone TTS synthesizer [1] now modified with emotive/expressive capabilities.

Various applications can be conceived by the use of animated characters, spanning from research on human communication and perception, via tools for the hearing impaired, to spoken and multimodal agent-based user interfaces.

The aim of this work was that of implementing INTERFACE a flexible architecture that allows us to easily develop and test a new animated face speaking in Italian.

2 A/V Acquisition Environment

INTERFACE is an integrated software designed and implemented in Matlab© in order to simplify and automate many of the operation needed for building-up a talking head. INTERFACE is mainly focused on articulatory data collected by ELITE, a fully automatic movement analyzer for 3D kinematics data acquisition [2].

ELITE provides for 3D coordinate reconstruction (see Fig. 1), starting from 2D perspective projections, by means of a stereophotogrammetric procedure which allows a free positioning of the TV cameras.

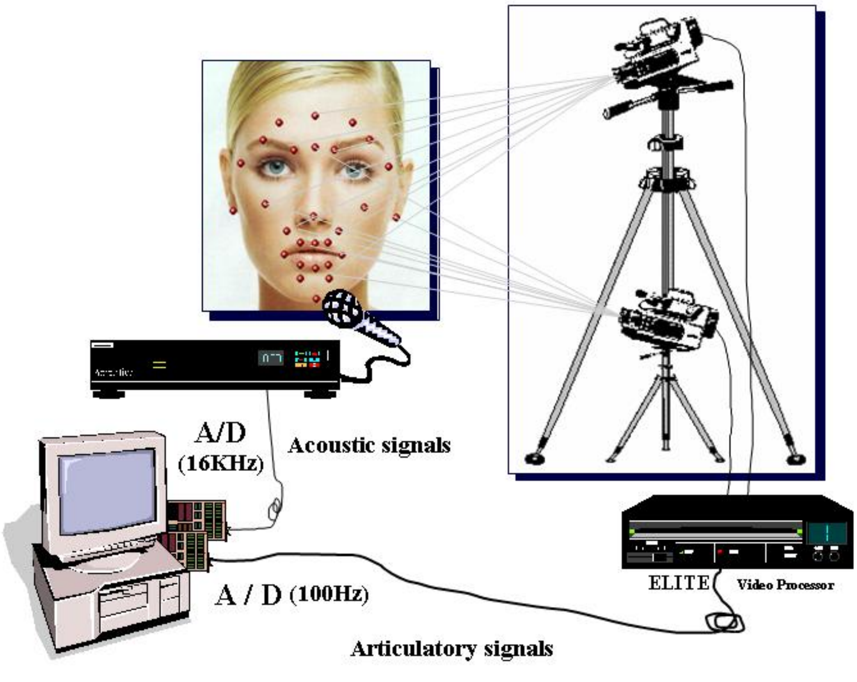


Fig. 1. A/V acquisition environment.

The 3D data dynamic coordinates of passive markers such as those illustrated in Fig.2 are then used to create our lips articulatory model and to drive directly, copying human facial movements, our talking face.

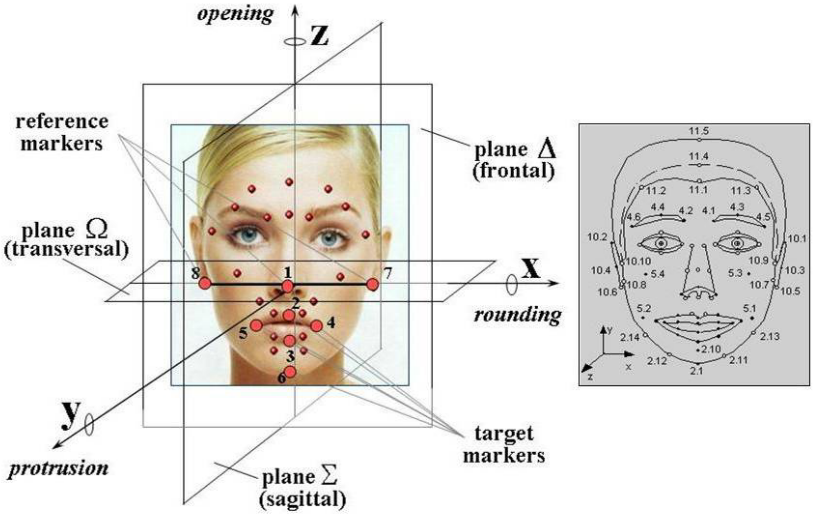


Fig. 2. Position of reflecting markers and reference planes for the articulatory movement data collection (on the left), and the MPEG-4 standard facial reference points (on the right).

Two different configurations have been adopted for articulatory data collection: the first one, specifically designed for the analysis of labial movements, considers a simple scheme with only 8 reflecting markers (bigger grey markers in Fig. 2) while the second, adapted to the analysis of expressive and emotive speech, utilizes the full and complete set of 28 markers. All the movements of the 8 or 28 markers, depending on the adopted acquisition pattern, are recorded and collected, together with their velocity and acceleration, simultaneously with the co-produced speech which is usually segmented and analyzed by means of PRAAT [3], that computes also intensity, duration, spectrograms, formants, pitch synchronous F0, and various voice quality parameters in the case of emotive and expressive speech [4-5].

3 INTERFACE

INTERFACE, whose block diagram is given in Fig. 3, was created mainly to develop LUCIA [6] our graphic MPEG-4 [7] compatible Facial Animation Engine (FAE). In MPEG-4 FDPs (Facial Definition Parameters) define the shape of the model, while FAPs (Facial Animation Parameters), define the facial actions [8]. In our case, the model uses a pseudo-muscular approach, in which muscle contractions are obtained through the deformation of the polygonal mesh around feature points that correspond to skin muscle attachments. A particular facial action sequence is generated by deforming the face model, in its neutral state, according to the specified FAP values, indicating the magnitude of the corresponding action, for the corresponding time instant.

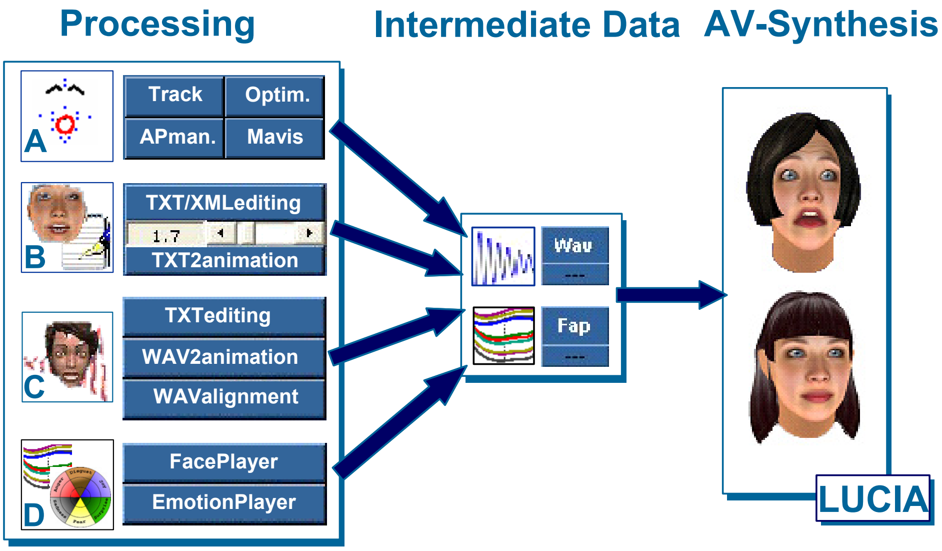


Fig. 3. Block diagram of INTERFACE (see text for details).

For a complete description of all the features and characteristics of INTERFACE, a full detailed PDF manual is being prepared and it is available at the official LUCIA web site: http://www.pd.istc.cnr.it/LUCIA/docs/InterFace20.pdf .
INTERFACE handles four types of input data from which the corresponding MPEG-4 compliant FAP-stream could be created:

- **Articulatory data**, represented by the marker trajectories captured by ELITE; these data are processed by 4 programs:
  - "*Track*", which defines the pattern utilized for acquisition and implements a new 3D trajectories reconstruction procedure;
  - "*Optimize*", that trains the modified coarticulation model [9] utilized to move the lips of LUCIA, our current talking head under development;
  - "*APmanager*", that allows the definition of the articulatory parameters in relation with marker positions, and that is also a DB manager for all the files used in the optimization stages;
  - "*Mavis*" (Multiple Articulator VISualizer, written by Mark Tiede of ATR Research Laboratories [10]) that allows different visualizations of articulatory signals;
- **Symbolic high-level TXT/XML text data**, processed by:
  - "*TXT/XMLediting*", an emotional specific XML editor for emotion tagged text to be used in TTS and Facial Animation output;
  - "*TXT2animation*", the main core animation tool that transforms the tagged input text into corresponding WAV and FAP files, where the first are synthesized by emotive/expressive FESTIVAL module, and the last, which are

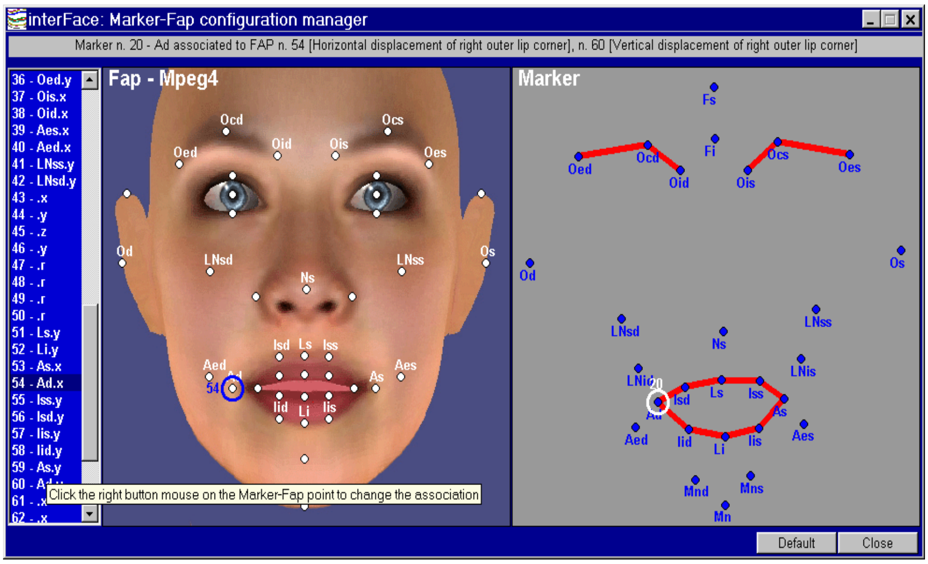
needed to animate MPEG-4 engines such as LUCIA, by the optimized animation model (designed by the use of Optimize);

  - "TXTediting", a simple text editor for unemotional text to be used in TTS and Facial Animation output;

- **WAV data**, processed by:
  - "*WAV2animation*", a tool that builds animations on the basis of input wav files after automatically segmenting them by an automatic ASR alignment system [11];
  - "*WAValignment*", a simple segmentation editor to manipulate segmentation boundaries created by WAV2animation;
- **manual graphic** low-level data , created by:
  - "*FacePlayer*", a direct low-level manual/graphic control of a single (or group of) FAP parameter; in other words, *FacePlayer* renders LUCIA's animation, while acting on MPEG-4 FAP points, for a useful immediate feedback;
  - "*EmotionPlayer*", a direct low-level manual/graphic control of multi level emotional facial configurations for a useful immediate feedback.

3.1 "Track"

MatLab© *Track* was developed with the aim of avoiding marker tracking errors that force a long manual post-processing stage and also a compulsory stage of marker identification in the initial frame for each used camera. *Track* is quite effective in terms of trajectories reconstruction and processing speed, obtaining a very high score in marker identification and reconstruction by means of a reliable adaptive processing. Moreover only a single manual intervention for creating the reference tracking model (pattern of markers) is needed for all the files acquired in the same working session. *Track*, in fact, tries to guess the possible target pattern of markers and the user must only accept a proposed association or modify a wrong one if needed, then it runs automatically on all files acquired in the same session. Moreover, we let the user the possibility to independently configure the markers and also the FAP-MPEG correspondence. The actual configuration of the FAPs is described in an initialization file and can be easily changed. The markers assignment to the MPEG standard points is realized with a context menu as illustrated in Fig. 4. By *Track* the articulatory movements can also be separated from the head roto-translation, thus allowing to realize a correct data driven articulatory synthesis.



interFace: Marker-Fap configuration manager

Marker n. 20 - Ad associated to FAP n. 54 [Horizontal displacement of right outer lip corner], n. 60 [Vertical displacement of right outer lip corner]

36 - Oed.y
37 - Ois.x
38 - Oid.x
39 - Aes.x
40 - Aed.x
41 - LNss.y
42 - LNsd.y
43 - .x
44 - .y
45 - .z
46 - .y
47 - .r
48 - .r
49 - .r
50 - .r
51 - Ls.y
52 - Li.y
53 - As.x
54 - Ad.x
55 - Iss.y
56 - Isd.y
57 - Iis.y
58 - Iid.y
59 - As.y
60 - Ad.y
61 - .x
62 - .x

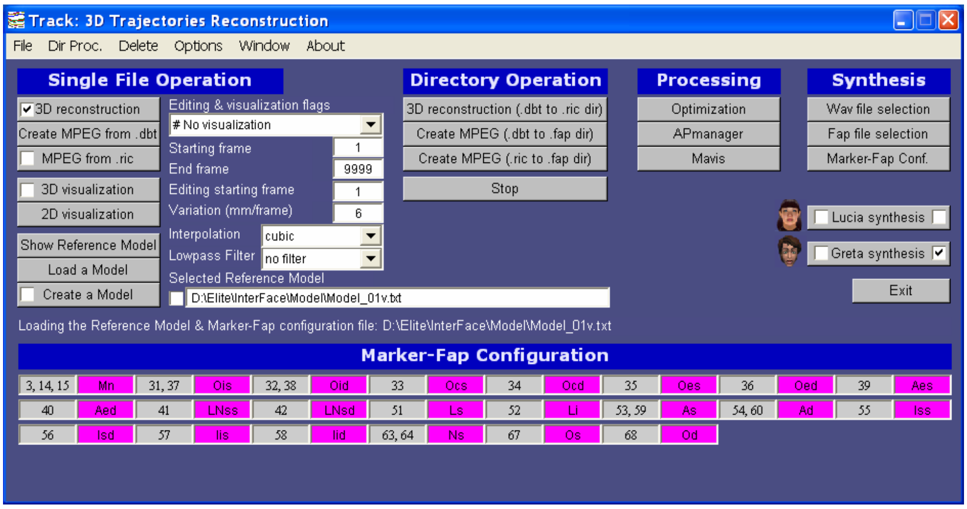
Click the right button mouse on the Marker-Fap point to change the association

Fig. 4. Marker MPEG-FAP association with the *Track*'s reference model. The MPEG reference points (on the left) are associated with the *Track*'s marker positions (on the right).

The main innovations introduced with the *Track* software can be here summarized:

- the reference model (see Fig. 4) remains the same for the entire working session, that is for all the acquisition files for which the configuration model is not modified; in other words, once the valid mask for a particular session is defined, the process of tracking the trajectories could be automatically started for the whole set of files;
- the marker identification and the reference space deformation problem have been exceeded with an algorithm based on the Singular Value Decomposition (SVD)[12]; this procedure has the intrinsic advantage to operate an error minimization while calculating the roto-translation, even independently from using a perfect undeformable reference space;
- almost all the processing stages have been automated; *Track* can work on a single file or on an entire directory, without manual intervention; a manual error correction phase can always be obviously set at the end of the processing;
- in the generation of the necessary FAP-stream for the animation, the correspondence between the acquisition marker points and standard MPEG-4 points is completely reconfigurable, this implying the possibility to adopt whichever other protocol to be used for the animation;
- the produced FAP-stream takes into account the roto-translation and the scale factors of the head that has to be animated, thus allowing a correct data-driven synthesis of whichever MPEG-4 compatible agent.

The *Track* interface is illustrated in Fig. 5. The area on the left regards operations on single files that is the 3D reconstruction, the MPEG-4 compatible data conversion, the visualization and editing of 2D and 3d marker trajectories, and the setting of the reference model (see Fig. 4).



Editing & visualization flags: # No visualization
Starting frame: 1
End frame: 9999
Editing starting frame: 1
Variation (mm/frame): 6
Interpolation: cubic
Lowpass Filter: no filter
Selected Reference Model: □ D:\Elite\InterFace\Model\Model_01v.txt

Loading the Reference Model & Marker-Fap configuration file: D:\Elite\InterFace\Model\Model_01v.txt

Marker-Fap Configuration

3, 14, 15	Mn	31, 37	Ois	32, 38	Oid	33	Ocs	34	Ocd	35	Oes	36	Oed	39	Aes
40	Aed	41	LNss	42	LNsd	51	Ls	52	Li	53, 59	As	54, 60	Ad	55	Iss
56	Isd	57	Iis	58	Iid	63, 64	Ns	67	Os	68	Od				

Fig. 5. “Track” interface.

The central *Directory Operation* buttons do the same processing with directories instead of single files, while the bottom area shows the correspondence between the FAP animation parameters and the trajectories of the markers that are currently under control. The presence of more than a single identification number for each FAP means that the control can be executed along the three different Cartesian axes. As an example, the first cell on the left (Mn), relative to the chin, contains the movements on all the three reference axes. With a simple click on the relative push-buttons it is possible to redefine the marker-FAP correspondence. Processing and Synthesis buttons refers to other INTERFACE programs which can be directly called within T*rack* itself other than within the main interface.

In summary, as illustrated in the examples shown in Fig. 6, for LUCIA, *Track* allows 3D real data driven animation of a talking face, converting the ELITE trajectories into standard MPEG-4 data and eventually it allows, if necessary, an easy editing of bad trajectories. Different MPEG-4 Facial Animation Engines (FAEs) could obviously be animated with the same FAP-stream allowing for an interesting comparison among their different renderings.

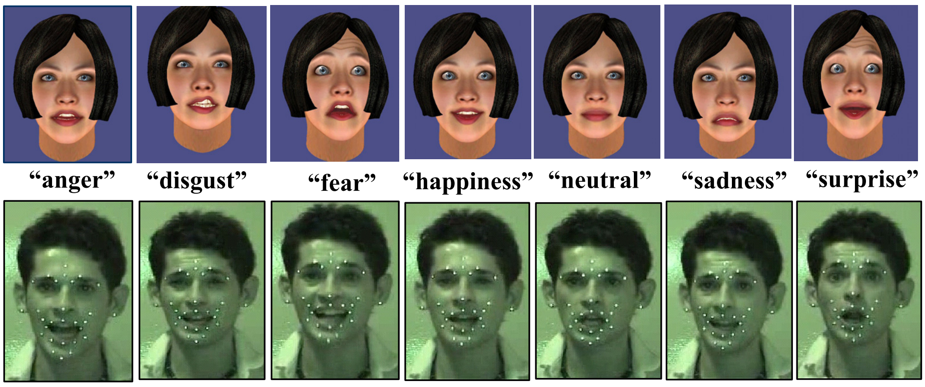


Fig. 6. Examples of a single-frame LUCIA's emotive expressions. These were obtained by acquiring real human movements with ELITE, by automatically tracking and reconstructing them with "Track", and by reproducing them with LUCIA.

3.2 *"Optimize"*

The *Optimize* module implements the parameter estimation procedure for LUCIA's lip articulation model. For generating realistic facial animation is necessary to reproduce the contextual variability due to the reciprocal influence of articulatory movements for the production of following phonemes. This phenomenon, defined coarticulation is extremely complex and difficult to model. A modified version of the Cohen-Massaro coarticulation model [6] has been adopted for LUCIA and a semi-automatic minimization technique, working on real cinematic data acquired by the ELITE opto-electronic system [2], was used for training the dynamic characteristics of the model, in order to be more accurate in reproducing the true human lip movements .
This procedure is based on a least squared phoneme-oriented error minimization scheme with a strong convergence property, between real articulatory data Y(n) and modeled curves F(n) for the whole set of R stimuli belonging to the same phoneme set:

$$e = \sum_{r=1}^{R}\left(\sum_{n=1}^{N}\left(Y_r(n) - F_r(n)\right)^2\right) \quad \textbf{(1)}$$

where F(n) is generated by a modified version of the Cohen-Massaro coarticulation model [6] as introduced in [13-14]. Even if the number of parameters to be optimized is rather high, the size of the data corpus is large enough to allow a meaningful estimation, but, due to the presence of several local minima, the optimization process has to be manually controlled in order to assist the algorithm convergence. The mean total error between real and simulated trajectories for the whole set of parameters is lower than 0.3 mm in the case of bilabial and labiodental consonants in the /a/ and /i/ contexts [15, p. 63]. At the end of the optimization stage the lip movements of our

MPEG-4 LUCIA can be obtained simply starting from a wav file and its corresponding phoneme segmentation information.

3.3 "TXT/XMLediting"

This is an emotional specific XML editor explicitly designed for emotional tagged text such as that shown in Fig.7.

```
<?xml version="1.0"  encoding="iso-8859-1"?>
<!DOCTYPE APML SYSTEM "apml.dtd">
<apml>
Ciao sono LUCIA.
<affective type="anger"> Sono proprio arrabbiata.</affective>
<affective type="fear"> Ma anche molto impaurita.</affective>
<affective type="sadness"> Sono molto triste,</affective>
</apml>
```

Fig. 7. Example of a text tagged with APML mark-up language extensions for emotive audio/visual synthesis.

The APML mark up language [16] for behavior specification permits to specify how to markup the verbal part of a dialog move so as to add to it the "meanings" that the graphical and the speech generation components of an animated agent need, to produce the required expressions (see Fig. 8).

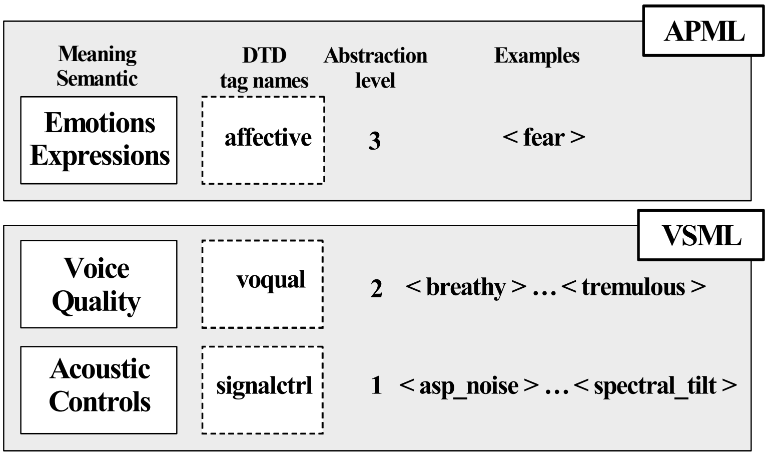


Fig. 8. APML/VSML mark-up language extensions for emotive audio/visual synthesis.

So far, the language defines the components that may be useful to drive a face animation through the facial description language (FAP) and facial display functions. The extension of such language is intended to support voice specific controls. An extended version of the APML language has been included in the FESTIVAL speech synthesis environment, allowing the automatic generation of the extended ".pho" file from an APML tagged text with emotive tags. This module implements a three-level hierarchy in which the affective high level attributes (e.g. <anger>, <joy>, <fear>, etc.) are described in terms of medium-level voice quality attributes defining the phonation type (e.g., <modal>, <soft>, <pressed>, <breathy>, <whispery>, <creaky>, etc.). These medium-level attributes are in turn described by a set of low-level acoustic attributes defining the perceptual correlates of the sound (e.g., <spectral tilt>, <shimmer>, <jitter>, etc.). The low-level acoustic attributes correspond to the acoustic controls that the extended MROLA synthesizer can render through the sound processing procedure described above. This descriptive scheme has been implemented within FESTIVAL as a set of mappings between high-level and low-level descriptors. The implementation includes the use of envelope generators to produce time curves of each parameter.

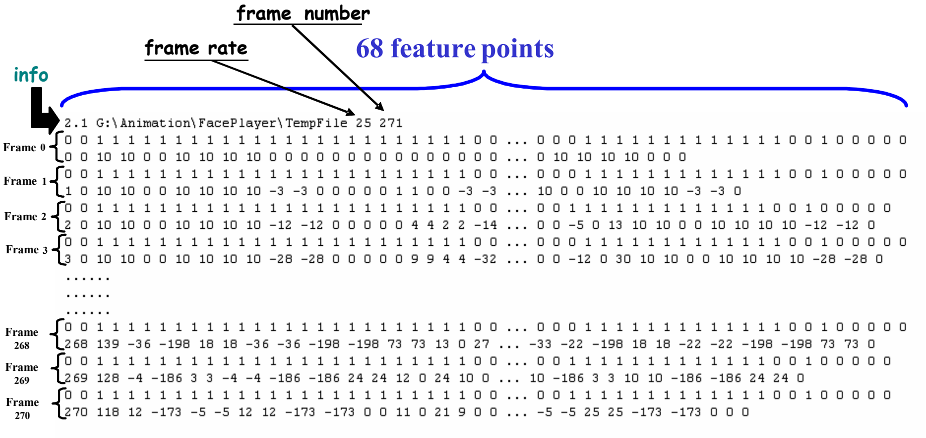
3.4 *"TXT2animation" ("AVengine")*

This represents the main core animation module. *TXT2animation* (also called "*AVengine*") transforms the emotional tagged input text into corresponding WAV and FAP files, where the first are synthesized by the Italian emotive version of FESTIVAL, and the last by the optimized coarticulation model, as for the lip movements, and by specific facial action sequences obtained for each emotion by knowledge-based rules.

Anger, for example, can be activated using knowledge-based rules acting on action units AU2 + AU4 + AU5 + AU10 + AU20 + AU24, where Action Units correspond to various facial action (i.e. AU1: "inner brow raiser", AU2: "outer brow raiser", etc.) [8].

In summary, a particular facial action sequence is generated by deforming the face model, in its neutral state, according to the specified FAP values, indicating the magnitude of the corresponding action, for the corresponding time instant. In MPEG-4, FDPs (Facial Definition Parameters) define the shape of the model while FAPs (Facial Animation Parameters), define the facial actions deforming a face model in its neutral state. Given the shape of the model, the animation is obtained by specifying the FAP-stream that is for each frame the values of FAPs (see Fig. 9). In a FAP-stream, each frame has two lines of parameters. In the first line the activation of a particular marker is indicated (0, 1) while in the second, the target values, in terms of differences from the previous ones, are stored. In our case, the model uses a pseudo-muscular approach, in which muscle contrac-tions are obtained through the deformation of the polygonal mesh around feature points that correspond to skin muscle attachments.

Each feature point follows MPEG4 specifications where a FAP corresponds to a minimal facial action. When a FAP is activated (i.e. when its intensity is not null) the feature point on which the FAP acts is moved in the direction signaled by the FAP itself (up, down, left, right, etc).


frame number
frame rate
68 feature points
info
Frame 0
Frame 1
Frame 2
Frame 3
Frame 268
Frame 269
Frame 270

```
2.1 G:\Animation\FacePlayer\TempFile 25 271
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 ... 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0 0 0 0 0
0 0 10 10 0 0 10 10 10 10 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ... 0 10 10 10 10 0 0 0
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 ... 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0 0 0 0 0
1 0 10 10 0 0 10 10 10 10 -3 -3 0 0 0 0 1 1 0 0 -3 -3 ... 10 0 0 10 10 10 10 -3 -3 0
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 ... 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0 0 0 0 0
2 0 10 10 0 0 10 10 10 10 -12 -12 0 0 0 0 0 4 4 2 2 -14 ... 0 0 -5 0 13 10 10 0 0 10 10 10 10 -12 -12 0
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 ... 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0 0 0 0 0
3 0 10 10 0 0 10 10 10 10 -28 -28 0 0 0 0 0 9 9 4 4 -32 ... 0 0 -12 0 30 10 10 0 0 10 10 10 10 -28 -28 0
......
......
......
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 ... 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0 0 0 0 0
268 139 -36 -198 18 18 -36 -36 -198 -198 73 73 13 0 27 ... -33 -22 -198 18 18 -22 -22 -198 -198 73 73 0
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 ... 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0 0 0 0 0
269 128 -4 -186 3 3 -4 -4 -186 -186 24 24 12 0 24 10 0 ... 10 -186 3 3 10 10 -186 -186 24 24 0
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 ... 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0 0 0 0 0
270 118 12 -173 -5 -5 12 12 -173 -173 0 0 11 0 21 9 0 0 ... -5 -5 25 25 -173 -173 0 0 0
```

Fig. 9. Example of a FAP stream.

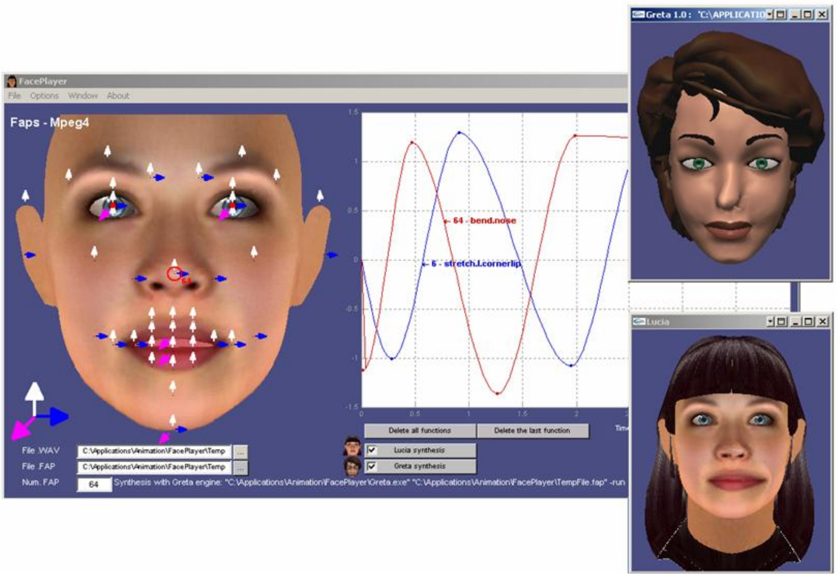
Using the pseudo-muscular approach, the facial model's points within the region of this particular feature point get deformed. A facial expression is characterized not only by the muscular contraction that gives rise to it, but also by an intensity and a duration. The intensity factor is rendered by specifying an intensity for every FAP. The temporal factor is modeled by three parameters: onset, apex and offset [8].
The FAP-stream needed to animate a FAE (Facial Animation Engine) could be completely synthesized by using a specific animation model, such as the coarticulation one used in LUCIA, or it could be reconstructed on the basis of real data captured by an optotracking hardware, such as ELITE.

3.5. *"WAV2animation" and "WAVsegmentation"*

WAV2animation is essentially similar to the previous TXT2animation module, but in this case an audio/visual animation is obtained starting from a WAV file instead that from a text file. An automatic segmentation algorithm based on a very effective Italian ASR system [11] extracts the phoneme boundaries. These data could be also verified and edited by the use of the *WAVsegmentation* module, and finally processed by the final visual only animation module of TXT2animation. At the present time the animation is neutral because the data do not correspond to a tagged emotional text, but in the future this option will be made available.

3.6. *"FacePlayer" and "EmotionPlayer"*

The first module *FacePlayer* (see Fig. 10) lets the user verify immediately through the use of a direct low-level manual/graphic control of a single (or group of) FAP (acting on MPEG4 FAP points) how LUCIA or another FAP Player renders the corresponding animation for a useful immediate feedback.



File .WAV C:\Applications\Animation\FacePlayer\Temp

File .FAP C:\Applications\Animation\FacePlayer\Temp

Num. FAP 64

Synthesis with Greta engine: "C:\Applications\Animation\FacePlayer\Greta.exe" "C:\Applications\Animation\FacePlayer\TempFile.fap" -run

Fig. 10. *FacePlayer*. A simple graphic tool for the facial rendering of a FAP Player such as LUCIA or GRETA [17] by the dynamic manipulation of single markers.

EmotionPlayer, which was strongly inspired by the EmotionDisc of Zofia Rutkay [18]), is instead a direct low-level manual/graphic control of multi level emotional facial configurations for a useful immediate feedback, as exemplified in Fig. 11.

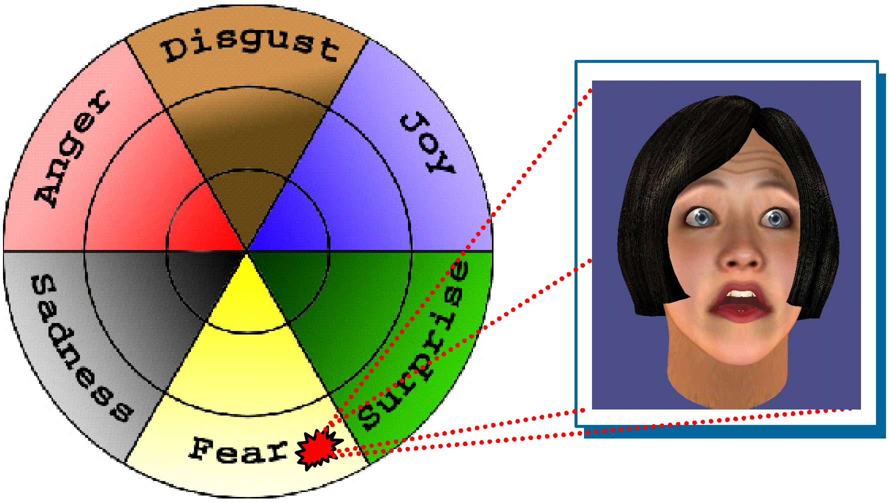


Fig. 11. Emotion Player. Clicking on 3-level intensity (low, mid, high) emotional disc [18], an emotional configuration (i.e. high -fear) is activated.

4 Conclusions

With the use of INTERFACE, the development of Facial Animation Engines and in general of expressive and emotive Talking Agents could be made, and indeed it was for LUCIA, much more friendly. Evaluation tools will be included in the future such, as for example, perceptual tests for comparing human and talking head animations, thus giving us the possibility to get some insights about where and how the animation engine could be improved.

5 Acknowledgements

Part of this work has been sponsored by PF-STAR (Preparing Future multiSensorial inTerAction Research, European Project IST- 2001-37599, http://pfstar.itc.it) and TICCA (Tecnologie cognitive per l'Interazione e la Cooperazione Con Agenti artificiali, joint "CNR - Provincia Autonoma Trentina" Project).

References

1. Cosi P., Tesser F., Gretter R., Avesani, C., "Festival Speaks Italian!", Proc. Eurospeech 2001, Aalborg, Denmark, September 3-7, 509-512, 2001.
2. Ferrigno G., Pedotti A., "ELITE: A Digital Dedicated Hardware System for Movement Analysis via Real-Time TV Signal Processing", IEEE Trans. on Biomedical Engineering, BME-32, 943-950, 1985.
3. Boersma P., "PRAAT, a system for doing phonetics by computer", Glot International, 5 (9/10), 341-345, 1996.
4. Magno Caldognetto E., Cosi P., Drioli C., Tisato G., Cavicchio F., "Coproduction of Speech and Emotions: Visual and Acoustic Modifications of Some Phonetic Labial Targets", Proc. AVSP 2003, Audio Visual Speech Processing, ISCA Workshop, St Jorioz, France, September 4-7, 209-214, 2003.
5. Drioli C., Tisato G., Cosi P., Tesser F., "Emotions and Voice Quality: Experiments with Sinusoidal Modeling", Proceedings of Voqual 2003, Voice Quality: Functions, Analysis and Synthesis, ISCA Workshop, Geneva, Switzerland, August 27-29, 127-132, 2003.
6. Cosi P., Fusaro A., Tisato G., "LUCIA a New Italian Talking-Head Based on a Modified Cohen-Massaro's Labial Coarticulation Model", Proc. Eurospeech 2003, Geneva, Switzerland, 127-132, 2003.
7. MPEG-4 standard. Home page: http://www.chiariglione.org/mpeg/index.htm
8. Ekman P. and Friesen W., Facial Action Coding System, Consulting Psychologist Press Inc., Palo Alto (CA) (USA), 1978.
9. Cohen M., Massaro D., "Modeling Coarticulation in Synthetic Visual Speech", in Magnenat-Thalmann N., Thalmann D. (Editors), Models and Techniques in Computer Animation, Springer Verlag, Tokyo, 139-156, 1993.
10. Tiede, M.K., Vatikiotis-Bateson, E., Hoole, P. and Yehia, H, "Magnetometer data acquisition and analysis software for speech production research", ATR Technical Report TRH 1999, ATR Human Information Processing Labs, Japan, 1999.

11. Cosi P. and Hosom J.P., “High Performance ‘General Purpose’ Phonetic Recognition for Italian”, Proc. of ICSLP 2000, Beijing, Cina, Vol. II, 527-530, 2000.
12. Soderkvist I. and Wedin P., Determining the movements of the skeleton using well-configured markers, Journal of Biomechanics, 26:1473-1477, 1993.
13. Pelachaud C., Magno Caldognetto E., Zmarich C., Cosi P., “Modelling an Italian Talking Head”, Proc. AVSP 2001, Aalborg, Denmark, September 7-9, 2001, 72-77.
14. Cosi P., Magno Caldognetto E., Perin G., Zmarich C., “Labial Coarticulation Modeling for Realistic Facial Animation”, Proc. 4th IEEE International Conference on Multimodal Interfaces ICMI 2002, Pittsburgh, PA, USA, 505-510, 2000.
15. Perin G., Facce parlanti: sviluppo di un modello coarticolatorio labiale per un sistema di sintesi bimodale, MThesis, Univ. of Padova, Italy, 2000-1.
16. De Carolis, B., Pelachaud, C., Poggi I., and Steedman M., “APML, a Mark-up Language for Believable Behavior Generation”, in Prendinger H., Ishizuka M. (eds.), Life-Like Characters, Springer, 65-85, 2004.
17. Pasquariello S., Pelachaud C., “Greta: A Simple Facial Animation Engine”, 6th Online World Conference on Soft Computing in Industrial Appications, Session on Soft Computing for Intelligent 3D Agents, September, 2001.
18. Ruttkay Zs., Noot H., ten Hagen P., “Emotion Disc and Emotion Squares: tools to explore the facial expression space”, Computer Graphics Forum, 22(1), 49-53, 2003.