LUCIA-WebGL: A Web Based Italian MPEG-4 Talking Head

G.Riccardo Leone, Piero Cosi

Institute of Cognitive Sciences and Technologies - National Research Council of Italy
riccardo.leone@pd.istc.cnr.it, piero.cosi@pd.istc.cnr.it

Abstract
In this work we present the reviewing of the activities focused on the development of the WebGL software version of LUCIA talking head, an open source facial animation framework developed at ISTC-CNR of Padua. LUCIA works on standard MPEG-4 Facial Animation Parameters and speaks with the Italian version of FESTIVAL TTS. LUCIA is totally based on true real human data collected by the use of ELITE, a fully automatic movement analyzer for 3D kinematics data acquisition. These informations are used to create lips articulatory model and to drive directly the talking face, generating human facial movements. We are exploiting the use of LUCIA WebGL as a virtual guide in the Wikimemo.it project: The portal of Italian Language and Culture. The easy integration of this technology in websites offers promising future uses for the WebGL Avatars: on-line personal assistant, storyteller for web-books, digital tutor for hearing impaired are only few examples.

Index Terms: WebGL, talking head, facial animation, mpeg4, 3D avatar, virtual agent, TTS, LUCIA, FESTIVAL

1. Introduction

Computers are becoming an integral part of all activities in our daily lives and the simply natural interaction with them is one of the most vibrant research in the last decade. Life-like characters [1] are embodied agents living on the screens of computational devices that invite us to communicate with them in familiar, expressive/emotive multi-modal ways. Face to face communication is the main element of human-human interaction because both acoustic and visual signal simultaneously convey linguistic, extra linguistic and paralinguistic informations. This is the reason why a realistic audiovisual synthesis is very important for virtual agents. This is a research topic since the early 70’s and many different principles, models and animations have been proposed over years [2]. In the late 90’s a specification for efficient coding of shape and animation of human face was included in the MPEG-4 international standard [3]. The focus was extended from traditional audio and video coding to other multimedia context including images, text, graphics, 3D scenes, animation and synthetic audio. Concerning Facial Animation MPEG-4 standard defines the shape of the model (FDP) and a set of actions (FAP); the animation is obtained by specifying a stream of numbers that is for each frame the values of the Facial Animation Parameters. Many implementations of this standard were born in the last decade [4] as stand alone applications built for research purpose. At ISTC-CNR of Padua we developed LUCIA talking head, an open source facial animation framework [5]. The recent introduction of WebGL [6], which is 3D graphics in web browsers, opens the possibility to bring all these applications to the home computers of a very large number of people and offers the new possibility to burst this natural way of interaction with the machines. WebGL extends the capability of the JavaScript programming language to generate interactive 3D graphics within any compatible web browser. The WebGL Working group (including Mozilla, Google, Apple and Opera) started in early 2009; two years after, on February 2011, they released the version 1.0 of WebGL. On May 2011 Google Chrome and Mozilla Firefox support WebGL and so will be in short time also Apple Safari and Opera browsers. WebGL is based on OpenGL Embedded System 2.0, the Graphic Library developed for mobile devices. This means that it does not have all the functionalities of the latest OpenGL for desktop, but on the other hand it is more easy that WebGL websites will run (in the near future) on smart-phone and other mobile devices. This is a real revolution because it brings the power of 3D graphics directly into web-browser without installing any plug-ins or customized and maybe dangerous software. Any application can run in all the platforms that support the new standard. The easy integration of this technology in any website offers promising future uses for WebGL Avatars: on-line personal assistant, storyteller for web-book, digital tutor for hearing impaired are only few examples. In this paper we will present the activities focused on the development of the WebGL software version of LUCIA talking head. We will depict the overall client-server architecture.
although only the 3D graphic animation is almost completed, which is the part we will go into details.

Figure 2: The 3D graphic animation is almost completed.

**2. The WebGL Client**

Lucia-WebGL follows the common client-server paradigm. The client (a web browser) connects to the server opening a webpage. The server answers with an HTML5 web-page where many new tag and multimedia contents have been introduced. One of this is the Canvas 3D, which is the place where all our 3D graphics lives. The typical WebGL application is composed by three parts: the standard html code for user interaction, the main JavaScript program and the Shading language section which is the code that is compiled and copied on the memory of the Video Card and is executed on the Graphic Processing Unit.

The server answers with an HTML5 web-page where many new tag and multimedia contents have been introduced. One of this is the Canvas 3D, which is the place where all our 3D graphics lives. The typical WebGL application is composed by three parts: the standard html code for user interaction, the main JavaScript program and the Shading language section which is the code that is compiled and copied on the memory of the Video Card and is executed on the Graphic Processing Unit. The server answers with an HTML5 web-page where many new tag and multimedia contents have been introduced. One of this is the Canvas 3D, which is the place where all our 3D graphics lives. The typical WebGL application is composed by three parts: the standard html code for user interaction, the main JavaScript program and the Shading language section which is the code that is compiled and copied on the memory of the Video Card and is executed on the Graphic Processing Unit.

Moving only the FDPs is not sufficient to smoothly move the whole 3D model; thus, each “feature point” is related to a particular influence zone constituted by an ellipses that represents a zone of the reticule where the movement of the vertexes is strictly connected. We consider two zone in the influence area: the points of area A that are affected by muscular contraction will be deformed by the muscular displacement function, while the points of area B (area of the bulge / furrow) will be moved outward to simulate the skin accumulation and bulging (fig. 5). We use the same “raised cosine” function of [8] to define the two areas. Finally, after having established the relationship for the whole set of FDPs and the whole set of vertices, all the points of the 3D model can be simultaneously moved with a graded strength following a raised-cosine function rule associated to each FDP. Each feature point follows MPEG4 specifications where a FAP corresponds to a minimal facial animation. 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). Using the
Fig. 3: The new client-server architecture: WebGL allows any system, even smart-phone and P.D.A, to interact with LUCIA via standard web browsers. At the beginning of the connection the model-parts of Lucia are fetched from the server in the JSON format. After that every communication involves only FAPs and audio streams with a very low bandwidth consumption.

 pseudo-muscular approach, the facial model 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 [9]. Only the reticule of polygons corresponding to the skin is directly driven by the pseudo-muscles and it constitutes a continuous and unitary element, while the other anatomical components move themselves independently and in a rigid way, following translations and rotations (for example the eyes rotate around their center). According to this strategy the polygons are distributed in such a way that the resulting visual effect is quite smooth with no rigid jumps over the entire 3D model. We are currently working on the audio streaming over the internet. The synchronization of the lips movements with the audio will be achieved due to the frame number information in the FAPs stream. The FAPs stream and the audio stream are the necessary information to animate the MPEG-4 based synthetic talking face (see fig. 3). These streams are generated on the server side by the Audio Video Engine.

3. Audio Video Engine Server

Audio Video speech synthesis, that is the automatic generation of voice and facial animation parameters from arbitrary text, is based on parametric descriptions of both the acoustic and visual speech modalities. The acoustic speech synthesis uses an Italian version of the FESTIVAL di-phone TTS synthesizer [10] modified with emotive/expressive capabilities: the APML/VSM mark up language [11] 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. For the visual speech synthesis a data-driven procedure was utilized: visual data are physically extracted by an automatic opto-tracking movement analyzer for 3D kinematics data acquisition called ELITE [12]. The 3D data coordinates of some reflecting markers positioned on the model subject face are recorded and collected, together with their velocity and acceleration, simultaneously with the co-produced speech. Using PRAAT [13], we obtain parameters that are quite significant in characterizing emotive/expressive speech [14]. In order to simplify and automates many of the operation needed for building-up the 3D avatar from the motion-captured data we developed INTERFACE [15], an integrated software designed and implemented in Matlab. To reproduce realistic facial animation in presence of co-articulation [16], a modified version of the Cohen-Massaro co-articulation model [17] has been adopted for LUCIA [18] and a semi-automatic minimization technique,
working on the real cinematic data, was used for training the dy-
namic characteristics of the model, in order to be more accurate
in reproducing the true human lip movements.

Figure 6: The FAPs stream is the necessary information to an-
imate an MPEG-4 based synthetic face. It is a very low band-
width transmission very good also with a slow connection

4. Conclusions and future work

LUCIA-WebGL is an MPEG-4 standard FAPs driven facial an-
timation talking head implementing a decoder compatible with
the "Predictable Facial Animation Object Profile" (very early
result in fig. 7). It has a high quality 3D model and a fine co-
articulation model, which is automatically trained by real data,
used to animate the face. The modified co-articulatory model is
able to reproduce quite precisely the true cinematic movements
of the articulatory parameters. The WebGL client-server ar-
chitecture separates the visualization/interaction process (on the
WebGL client) from the generation of the audiovisual streams
necessary for the animation of the talking head (on the server side).
These streams are very low bit-rate and can function very
well also with slow connections. This is a work in progress ac-
tivity and we currently finished the 3D graphic animation part.
Next steps are the synchronization of audio and 3D animation
(handling streams over the internet) and the implementation of
the emotive/expressive functionalities based on APML/VSMIL
markup language. Future development will include the per-
sonalization of the model using as textures some photos of a
real face taken from different views. The first use of LUCIA
WebGL will be a virtual guide in the Wikimemo.it project: The
portal of Italian Language and Culture. However the easy in-
tegration of this technology in common websites allows many
other roles for LUCIA: we want to test it as a storyteller for
web-books and a digital tutor for the hearing impaired persons.

5. Acknowledgements

This work has been sponsored by WIKIMEMO.IT (The Portal
of Italian Language and Culture, FIRB Project, RBNE078K93,
Italian Ministry of University and Scientific Research)

6. References

4.htm.
tion: The Standard, Implementation and Applications. New
[8] C. Pelachaud and S. Pasqualiello, “Greta, an italian talking face,”
italian!” in Proceedings of Eurospeech 2001, Aalborg, Denmark:
a mark-up language for believable behavior generation,” Life-Like
system for movement analysis via real-time tv signal processing,”
IEEE Transactions on Biomedical Engineering, pp. 943–
quality: Experiments with sinusoidal modeling,” in Proceedings
for building emotive/expressive talking heads,” in Proceedings
of INTERSPEECH 2005. Lisbon, Portugal: INTERSPEECH, 2005,
pp. 781–784.
[16] E. Farnetani and D. Recasens, “Coarticulation models in recent
speech production theories,” Coarticulation in Speech Production,
1999.
[17] P. Cosi and G. Perin, “Labial coarticulation modeling for realis-
tic facial animation,” in Proceedings of ICMI 2002. Pittsburgh,
[18] P. Cosi, A. Fusaro, and G. Tisato, “Lucia a new italian talking-
head based on a modified cohen-massaros labial coarticulation