# Greta: A Simple Facial Animation Engine

**Article** · January 2002

DOI: 10.1007/978-1-4471-0123-9_43



Catherine Pelachaud
French National Centre for Scientific Research

# Greta: A Simple Facial Animation Engine

Stefano Pasquariello[1], Catherine Pelachaud[2]

[1]Kyneste S.p.A, Rome, Italy
[2]Department of Computer and Information Science University of Rome "La Sapienza", Rome, Italy

**Abstract.** In this paper, we present a 3D facial model compliant with MPEG-4 specifications; our aim was the realization of an animated model able to simulate in a rapid and believable manner the dynamics aspect of the human face.

We have realized a Simple Facial Animation Engine (SFAE) where the 3D proprietary facial model has the look of a young woman: "Greta". Greta is the core of an MPEG-4 decoder and is compliant with the "Simple Facial Animation Object Profile" of the standard. Greta is able to generate the structure of a proprietary 3D model, to animate it, and, finally, to render it in real-time.

Our model uses a pseudo-muscular approach to emulate the behaviour of face tissues and also includes particular features such as wrinkles and furrow to enhance its realism. In particular, the wrinkles have been implemented using bump mapping technique that allows to have a high quality 3D facial model with a relative small polygonal complexity.

## 1 Introduction

MPEG-4 is an ISO/IEC standard developed by MPEG (Moving Picture Experts Group), the committee that also developed the well known standards MPEG-1 and MPEG-2.
MPEG-4, whose formal ISO/IEC designation is ISO/IEC 14496, was finalized in October 1998 and became an International Standard in the first months of 1999. The fully backward compatible extensions under the title of MPEG-4 Version 2 were frozen at the end of 1999, to acquire the formal International Standard Status early 2000. Some work, on extensions in specific domains, is still in progress.
MPEG-4 provides the standardized technological elements enabling the integration of the production, distribution and content access paradigms of: digital television, interactive graphics applications (synthetic content) and interactive multimedia (World Wide Web, distribution of and access to content).

Among other items, MPEG-4 defines specifications for the animation of face and body models within a MPEG-4 terminal [4, 9]. We realized a Simple Facial Animation Engine (SFAE) that has the look of a young woman: "Greta", it is the core of a MPEG-4 decoder and it is compliant with the "Simple Facial Animation Object Profile" defined by the standard.

# 2 Facial Animation Coding in MPEG-4 Standard

According to the MPEG-4 standard, the face is a node in a scene graph that include face geometry ready for rendering. The shape, texture and expressions of the face are generally controlled by the bitstream containing instances of Facial Definition Parameter (FDP) sets and Facial Animation Parameter (FAP) sets.
Upon initial or baseline construction, the Face Object contains a generic face with a neutral expression: the "neutral face". This face is already capable of being rendered. It is also immediately capable of receiving the FAPs from the bitstream, which will produce animation of the face: expressions, speech etc. If FDPs are received, they are used to transform the generic face into a particular face determined by its shape and (optionally) texture. Optionally, a complete face model can be downloaded via the FDP set as a scene graph for insertion in the face node.

## 2.1 The Object Profiles

MPEG-4 defines three profiles for facial animation object that allow different levels of configuration of the decoder:

- **Simple Facial Animation Object Profile**: The decoder has its own proprietary model that is animated by a coded FAP stream.
- **Calibration Facial Animation Object Profile**: This profile includes the simple profile. The encoder transmits to the decoder calibration data for some or all of the predefined feature points. The decoder adapts its proprietary face model such that it aligns with the position of these feature points. This allows for customization of the model.
- **Predictable Facial Animation Object Profile:** This profile includes the calibration profile. MPEG-4 also provides a mechanism for downloading a model to the decoder according to Section 2.3 and animating this model.

Actually, Greta, presented in this paper, is fully compliant with MPEG-4 specifications of the Simple Facial Animation Object Profile.

## 2.2 The Neutral Face

At the beginning of an animated sequence, by convention, the face is assumed to be in a neutral position. All the animation parameters are expressed as displacements from the position defined in the "neutral face" (see Fig. 1).

**Fig. 1.** The “neutral face” of Greta.

### 2.3 The Features Points

In order to define face animation parameters for arbitrary face models, MPEG-4 specifies 84 “feature points” located in relevant somatic places of the human face (see Fig. 2). They are subdivided in groups and labeled with a number, depending on the particular region of the face to which they belong. Therefore, some of the feature points (black labeled in the figure) are affected by the FAPs.

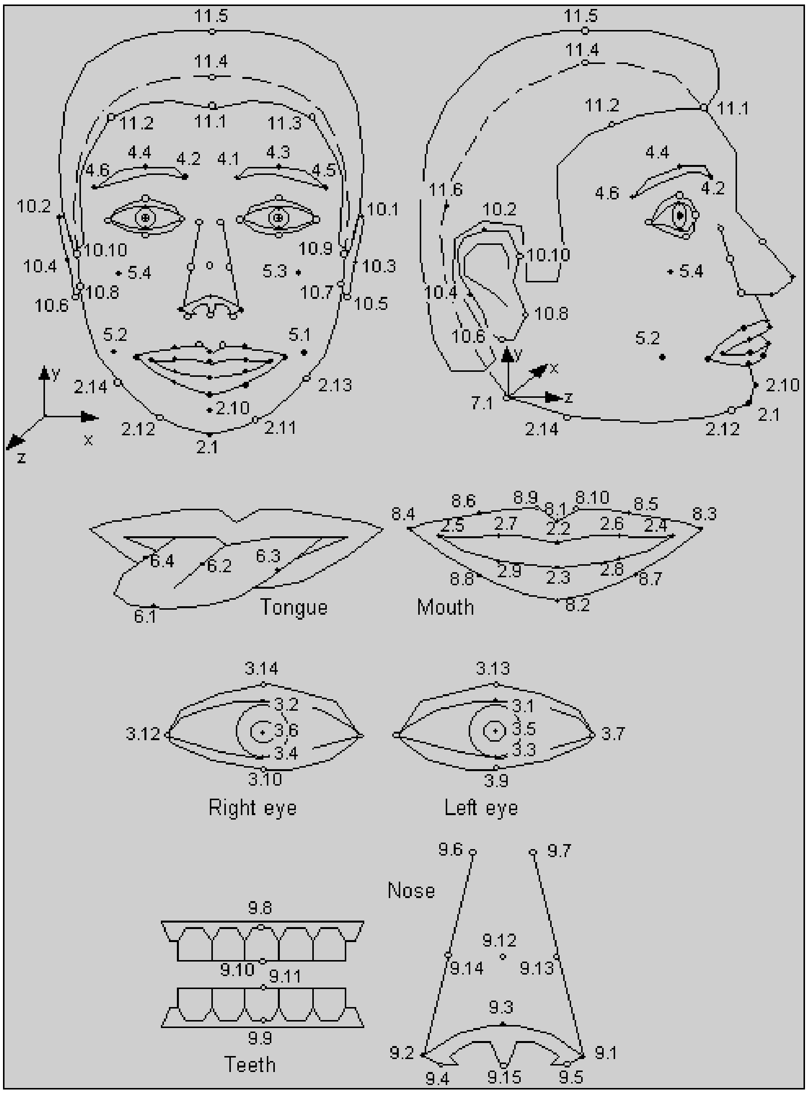


**Fig. 2.** The “feature points” (the black labeled ones are affected by FAPs).

## 2.4 The Facial Animation Parameters (FAPs)

The FAPs are based on the study of minimal perceptible actions and are closely related to muscle action. The 68 parameters are categorized into 10 groups related to parts of the face (see Table 1).

**Table 1.** FAP groups.

| Groups | Number of FAPs in the Group |
|---|---|
| Visemes and expression | 2 |
| Jaw, chin, inner lowerlip, cornerlips, midlip | 16 |
| Eyeballs, pupils, eyelids | 12 |
| Eyebrow | 8 |
| Cheeks | 4 |
| Tongue | 5 |
| Head rotation | 3 |
| Outer lip positions | 10 |
| Nose | 4 |
| Ears | 4 |

FAPs represent a complete set of basic facial actions including head motion, tongue, eye, and mouth control. They allow the representation of natural facial expressions. They can also be used to define facial action units [5].

Technically, the FAPs define the displacements of the feature points in relation to their positions in the neutral face. In particular, except that some parameters encode the rotation of the whole head or of the eyeballs, a FAP encodes the magnitude of the feature point displacement along one of the three Cartesian axes (see Table 2).

**Table 2.** FAP description table.

| # | FAP Name | FAP Description | Units | Uni /Bidir | Pos Motion | Grp | FDP Sub Grp Num |
|---|---|---|---|---|---|---|---|
| … | … | … | … | … | … | … | … |
| 3 | open_jaw | Vertical jaw displacement (does not affect mouth opening) | MNS | U | down | 2 | 1 |
| 4 | lower_t_midlip | Vertical top middle inner lip displacement | MNS | B | down | 2 | 2 |
| 5 | raise_b_midlip | Vertical bottom middle inner lip displacement | MNS | B | up | 2 | 3 |
| 6 | stretch_l_cornerlip | Horizontal displacement of left inner lip corner | MW | B | left | 2 | 4 |
| 7 | stretch_r_cornerlip | Horizontal displacement of right inner lip corner | MW | B | right | 2 | 5 |
| 8 | lower_t_lip_lm | Vertical displacement of midpoint between left corner and middle of top inner lip | MNS | B | down | 2 | 6 |
| … | … | … | … | … | … | … | … |

The magnitude of the displacement is expressed by means of specific measure units, called FAPU (Facial Animation Parameter Unit). Except for FAPU used to measure rotations, each FAPU represents a fraction of a specific distance on the human face; so, is possible to express FAP in a general way by a normalized range of values that can be extracted or reproduced by any model.

### 2.5 The Facial Definition Parameters (FDPs)

The FDPs are responsible for defining the appearance of the 3D face model. These parameters can be used in two ways:

- To modify the shape and appearing of a face model already available at the decoder.
- To encode the information necessary to transmit a complete model and the rules that must be applied to animate it.

In both cases the animation of the model is described only by the FAPs. FDPs, on the contrary, are typically used only at the beginning of a new session. However, in our work is not necessary to use the FDPs because, actually, Greta is compliant with the Simple Facial Animation Object Profile.

## 3 The 3D Model of Greta

Greta is a Simple Facial Animation Engine (SFAE) compliant with the Simple Facial Animation Object Profile of the MPEG-4 standard [7]; so, our decoder has a proprietary model that has been directly animated by the FAPs.

### 3.1 The Polygonal Structure of the 3D Model

After a study of the best solution for the surface representation of the 3D facial model [10], we decided to use polygonal surfaces because modern graphic workstation are adept at displaying them and can update complex facial polygonal models in near real time. In particular we used OpenGL technologies to develop, display and animate our original 3D model (see Fig. 3-5).

**Fig. 3.** Greta 3D model (front view).

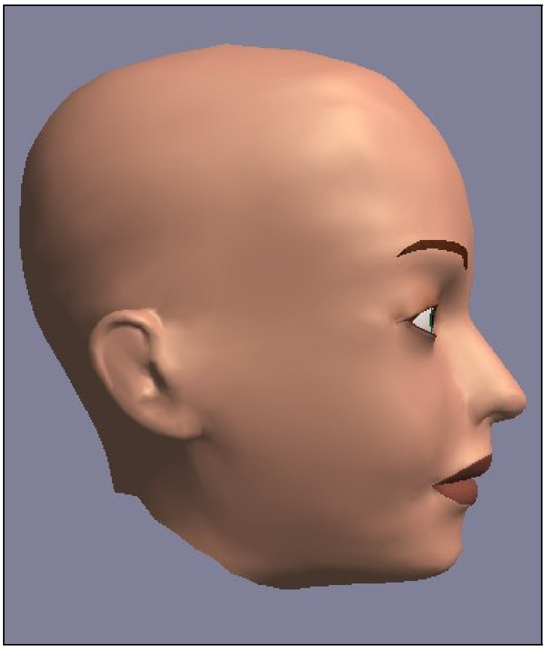

**Fig. 4.** Greta 3D model (side view).

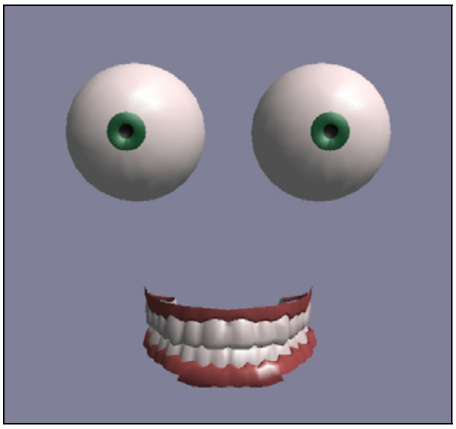

**Fig. 5.** The internal anatomic components.

In the human face there are specific regions that are dedicated to the communication of the information and to express the emotions, so they need to be well defined. We concentrated our efforts on giving a great level of detail in the most expressive regions of facial model:

The **mouth**: is the most mobile region of the human face; it takes part in the composition of all the face expressions and the lips generate visemes during the speaking action [1]. The polygonal structure (see Fig. 6) of this region should be conformable to the muscles fibers (in particular the *orbicularis oris*, see Fig. 7) around the mouth that make possible a great number of expressions and visemes [6].

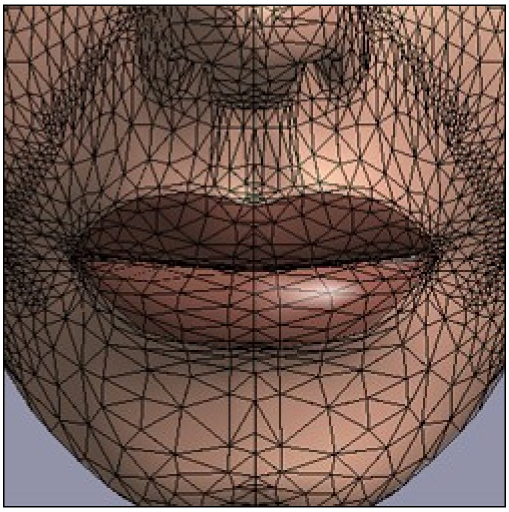

**Fig. 6.** The mouth of Greta (wireframe shade visualization).

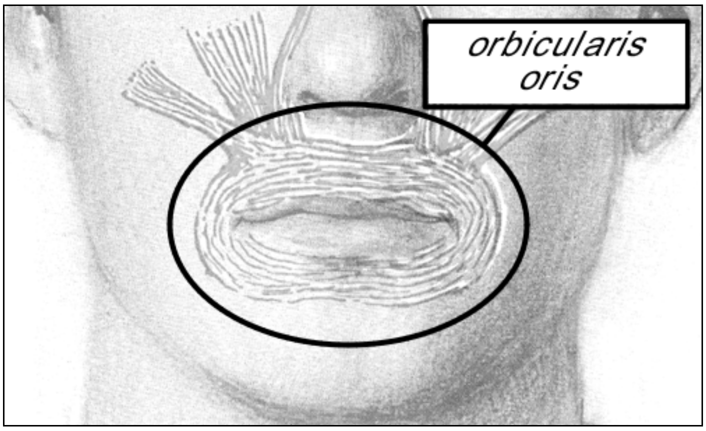


**Fig. 7.** The muscles that compose the lips.

The **eyes**: with brows are easily the most magnetic and compelling part of the human face. The polygonal structure (see Fig. 8) of this region should make easy coordinated linguistic and gaze communicative acts, blink, eyebrow and eyelid (upper and lower) moves.

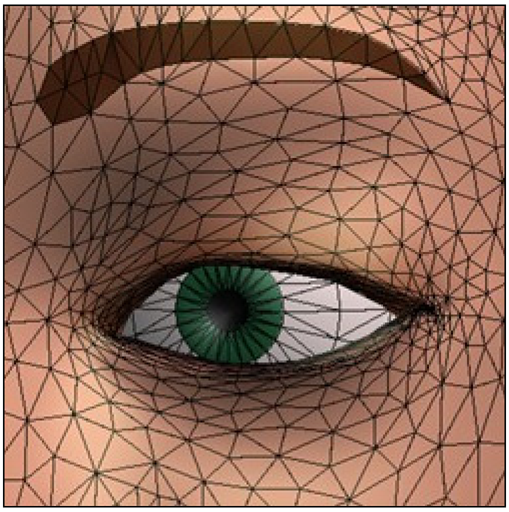

**Fig. 8.**The eye of Greta (wireframe shade visualization).

### 3.2 Critical Regions of the 3D Model

In order to simulate the complex behaviour of the skin like the generation of wrinkles and furrows during the muscular action, we dedicated particular attention to the polygonal structure of two further regions of the facial model:

- The **forehead**: is a plane region of the face in the rest and neutral position, but, during communicative acts, the raising of the eyebrows generates on it regular horizontal wrinkles, that are perpendicular to the vertical pull of the forehead muscles. So we needed a regular horizontal structure for the polygonal lattice of the forehead (see Fig. 9).

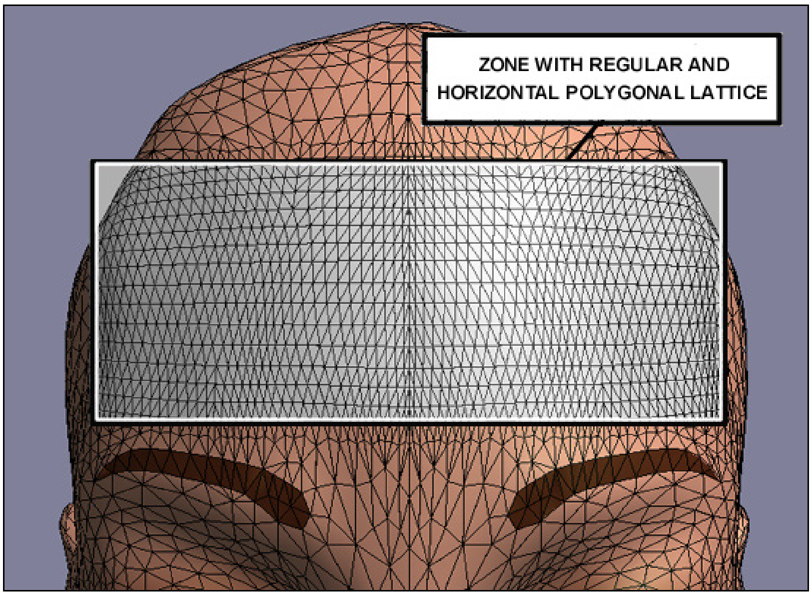


**Fig. 9.** The forehead of Greta with its polygonal structure.

- The **nasolabial furrow**: is typically of the contraction of the *zygomatic mayor* muscle. For example during a smile, there is a generation of a clear furrow between the stretched skin near the mouth an the skin in the cheeks that, pressed up from below, bulges out like a tiny balloon being squeezed. We needed, in this part of the model, a well defined line of separation between the two skin regions, with a considerable polygonal density (see Fig. 10-11).

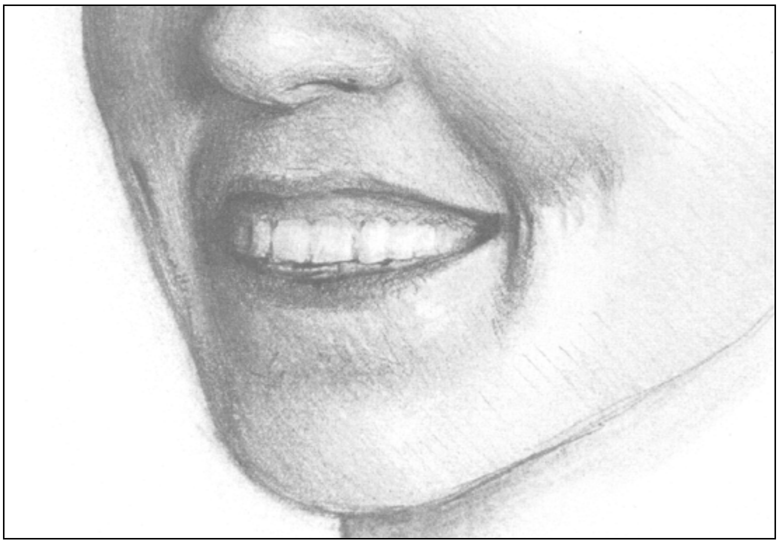

**Fig. 10.** Example of nasolabial furrow.

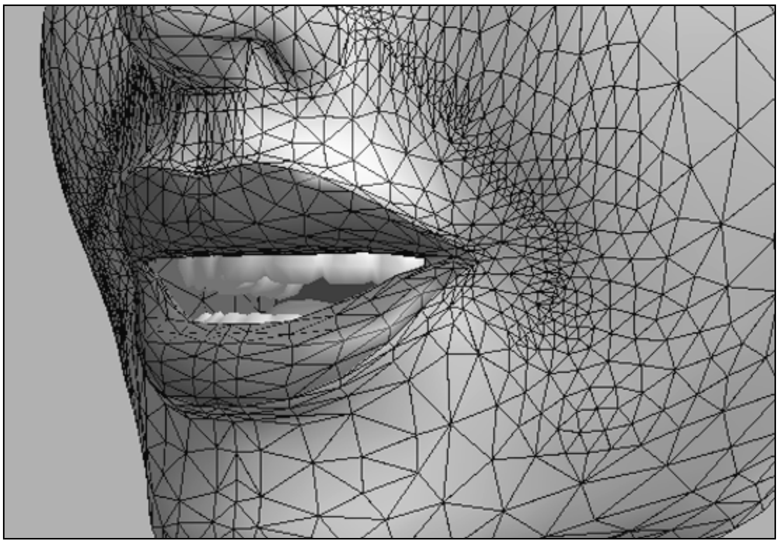

**Fig. 11.** The polygonal structure in the nasolabial furrow in Greta.

### 3.3 Subdivision of the 3D Model in "Specific Areas"

In order to have more control on the polygonal lattice we subdivided the 3D model surface in "specific areas" that corresponds to the feature points affected by the FAPs. This subdivision was necessary to circumscribe and control the displacements of polygonal vertexes induced by the FAPs applied in the various feature points. For example, in the nasolabial furrow, the two skin zones on the opposing sides of the furrow belong to distinct specific areas. All the specific areas are classified and can be seen in Fig. 12-13.

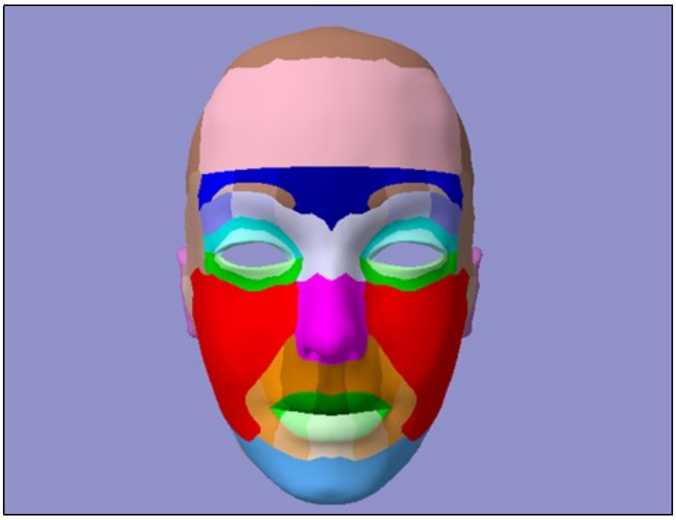

**Fig. 12.** Subdivision of Greta in "specific areas" (front view).

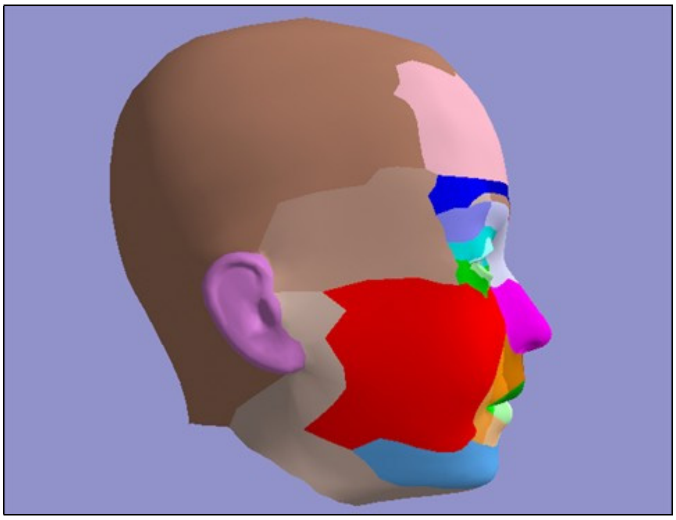

**Fig. 13.** Subdivision of Greta in "specific areas" (side view).

# 4 Simulation of the Skin Behaviour

Greta uses the FAPs to animate the skin of the 3D model in a realistic way, but it doesn't make a physical simulation of the muscles of the face and of the viscous-elastic behaviour of the skin, because the need of real-time calculation of the animation is a great restriction for this approach. So the FAPs activate functions that deform the 3D polygonal lattice during the time and make an "emulation" of the real behaviour of the face tissues [12].

## 4.1 Definition of "Area of Influence"

We have to define the "area of influence" of the feature point. The deformation, due to the FAP for a single feature point, is performed in a zone of influence that has an ellipsoid shape whose centroid is the feature point. The "area of influence" is the zone of the polygonal lattice that is within the ellipsoid; so we have a precise number of vertex that are affected by the displacement due to the FAP (see Fig. 14). It must be noticed how the area of influence of each feature points can overlap one another and, so, a vertex of the 3D model can be under the influence of various feature points and their FAPs.

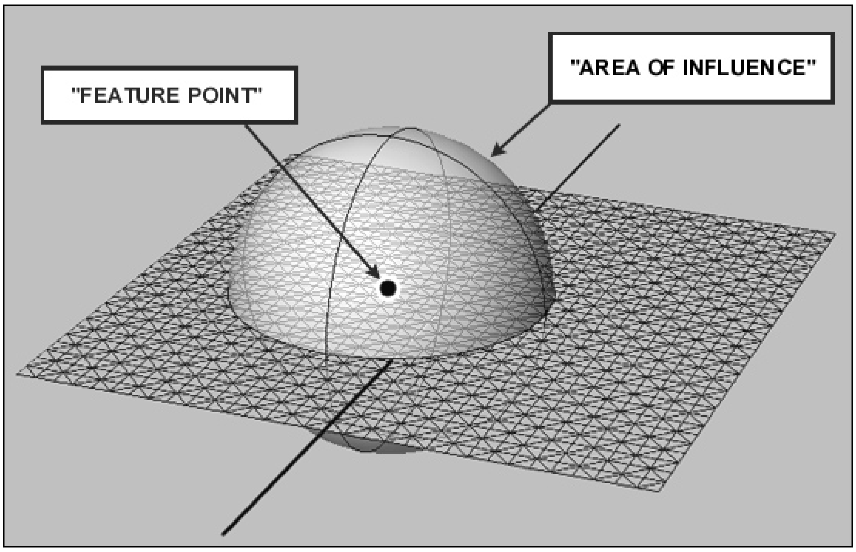


**Fig. 14.** The ellipsoid and the "area of influence".

## 4.2 Definition of "Displacement Function"

The FAP is like a muscular action applied in the key point [5, 8, 13]; so, the displacement of the vertexes of the polygonal lattice into the area of influence has to emulate the behaviour of  skin under a traction applied just in the feature point.

The displacement $\Delta x_j$ of a vertex $P_j$ $(x_j, y_j, z_j)$ due to a FAP operating in the $x$ direction $FAP_x$ is computed (without considering now the "specific area") as:

$$\Delta x_j = W_j * FAP_x \tag{1}$$

The weight $W_j$ is based on the distance of the vertex from the feature point and it is computed by a function that can be seen in Fig. 15, where $d''_j$ is a normalized distance. Therefore, in the sequence of Fig. 16-18 is possible to see effect of the displacement function centered in the white point on a plane polygonal lattice, with a growing intensity of displacement.

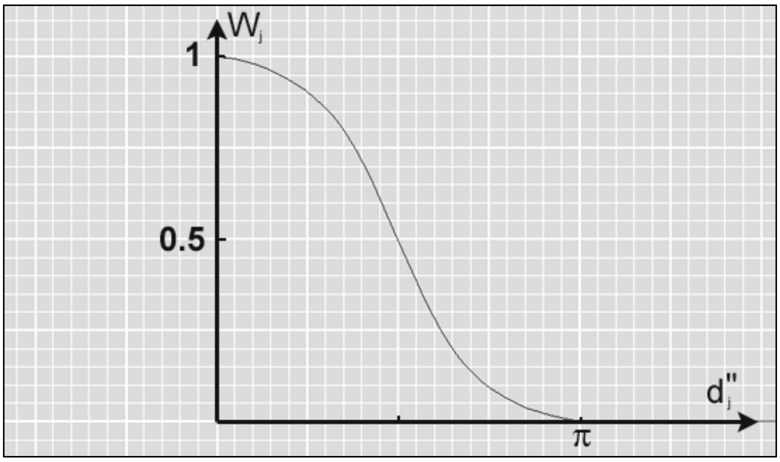


**Fig. 15.** The function of the weight $W_j$.

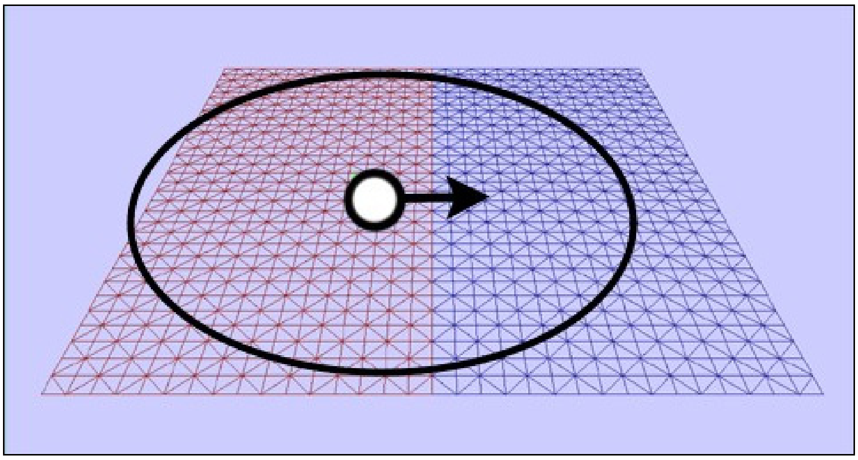

**Fig. 16.** Deformation within an area of influence (null intensity).

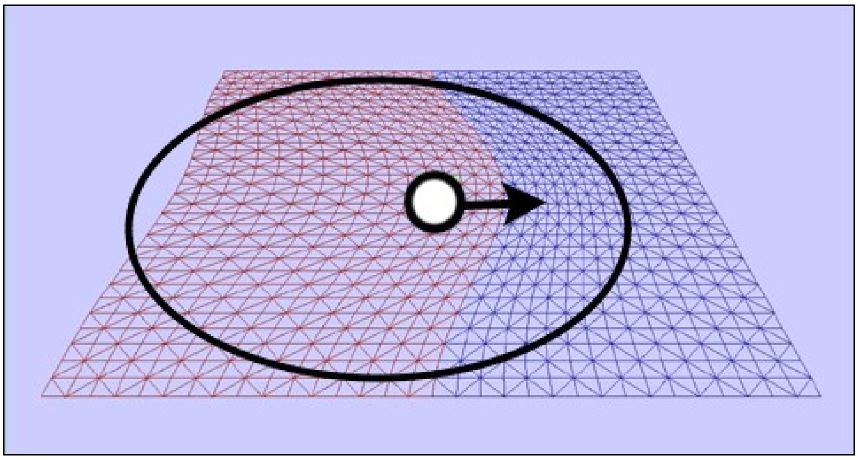

**Fig. 17.** Deformation within an area of influence (medium intensity)

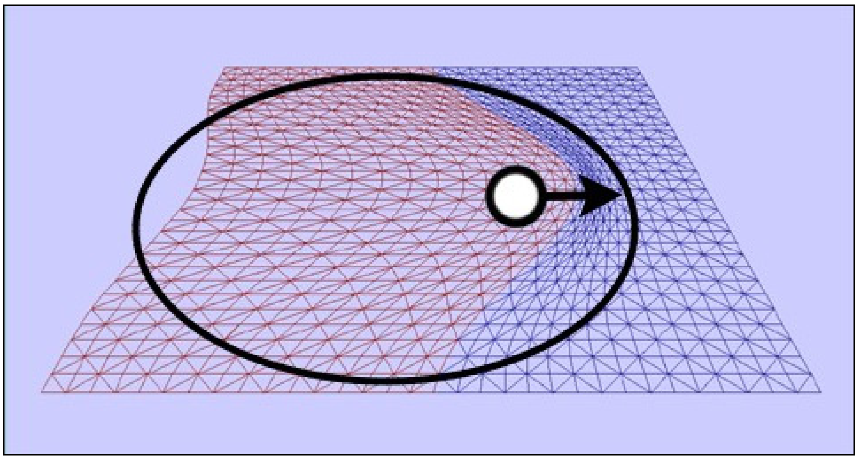

**Fig. 18.** Deformation within an area of influence (maximum intensity)

### 4.3 Simulation of Furrows and Bulges

Greta has the possibility to emulate the complex behaviour of the skin like furrow and bulges (i.e. the nasolabial furrow in Fig. 10 that occurs during the contraction of the *zygomatic mayor* muscle) [11, 12, 13]. The mechanism that generates the furrow in the skin is explained in Fig. 19; there is the traction of the mimic muscle that operates only onto a limited linear part of the skin and there are two distinct zones:

- The left zone, that is formed by the elastic skin stretched by the muscle.

- The right zone, that is formed by the accumulation of the skin due to the compression.

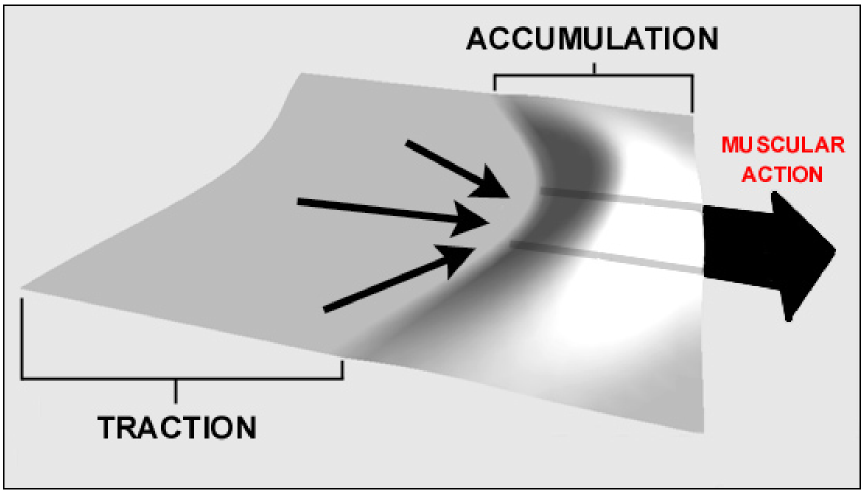


**Fig. 19.** The mechanism that generates the skin furrow.

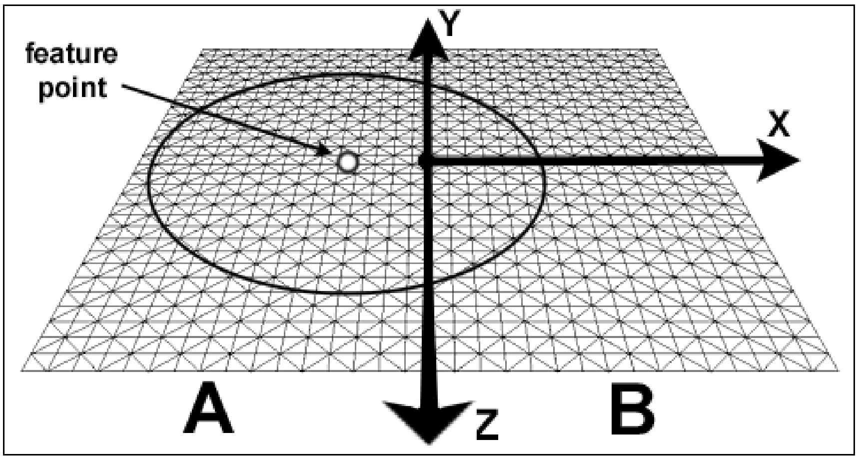


**Fig. 20.** The two zones of skin and the feature point.

If we have a simple situation like the Fig. 20 and we want a furrow along the z axis, the emulation of this behaviour of the skin is obtained by the sum of the horizontal displacement of the vertexes ($\Delta x_i$ of a vertex $P_i$ $(x_i, y_i, z_i)$) in the A and B zones (according to the laws previously explained and illustrated in Fig. 16-18) and the vertical displacement ($\Delta y_i$ of a vertex $P_i$ $(x_i, y_i, z_i)$) of the vertexes in the B zone, according to the following law:

$$\Delta y_i = \Delta x_i * K_1 * (0.5 * (1 + \cos(d_i))) * (1 - \exp(-d'_i / K_2)) \qquad (2)$$

where $d_i$ is the distance of the vertex $P_i$ from the feature point, $d'_i$ is the distance between the vertex and the z axis, and $K_1$ and $K_2$ are constants. Therefore, in the sequence of Fig. 21-23 is possible to see the effect of the displacement functions (along x and y axis) on a plane polygonal lattice when a FAP is applied in the feature point (green labeled), with a growing intensity.

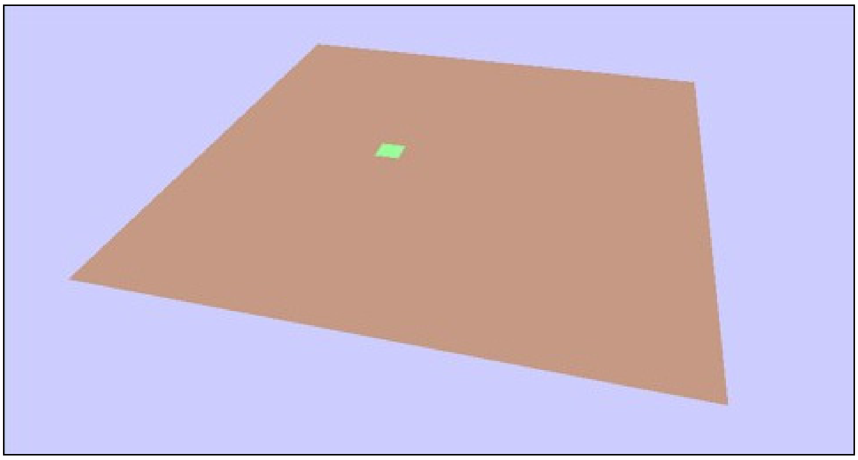

**Fig. 21.** Generation of the furrow and the bulge (null intensity of the FAP).

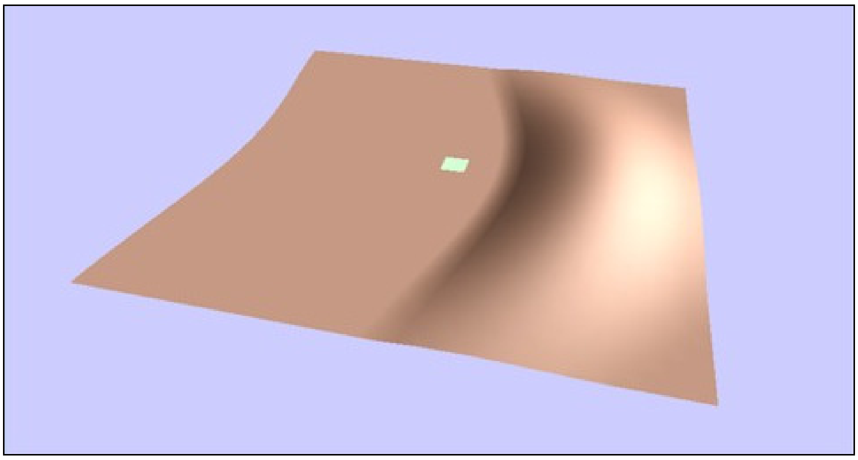

**Fig. 22.** Generation of the furrow and the bulge (medium intensity of the FAP).

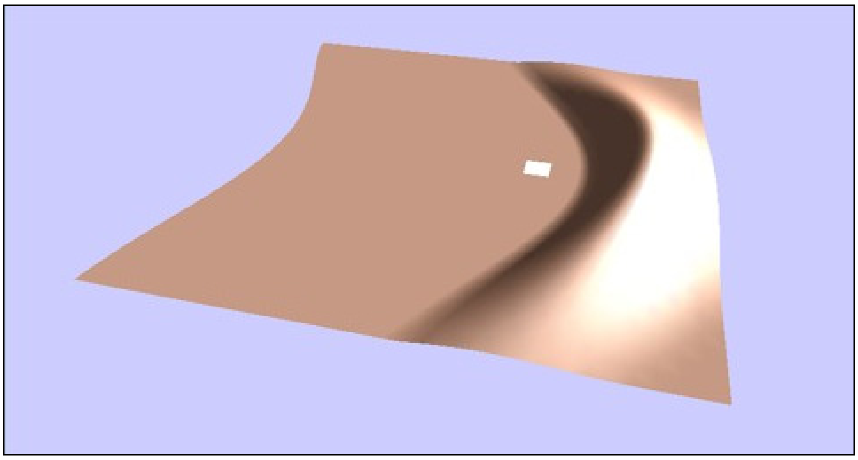

**Fig. 23.** Generation of the furrow and the bulge (maximum intensity of the FAP).

Finally, in the Fig. 24 is possible to see the effect of the displacement functions on the Greta face during the action of the *zygomatic major*.

**Fig. 24.** A "smile" of Greta.

## 4.4 Simulation of Wrinkles

Greta can also emulate the behaviour of the skin in the zone of the forehead where, during the raising of the eyebrows, there is the generation of many horizontal wrinkles. The mechanism is similar to that seen in the previous section, in fact there is an accumulation of skin due to the compression of the raising eyebrows but there is a series of little horizontal creases because the substrate of the forehead is different from the substrate of the cheeks.

Greta uses a bump mapping technique based on per-vertex shading of the OpenGl visualization; it is an innovative technique that let to have an high quality of the animated facial model with a *small polygonal complexity*. In fact, in OpenGl based systems, the Gouraud algorithm, that makes possible the shading rendering mode, uses the normals applied in the vertexes to compute a smooth lightning on the surfaces [2]. So, using the regular disposition of polygons in the forehead region of the 3D model, is possible to perform a perturbation of the normals, *without moving the vertexes*, and have the bump mapping, that emulates the forehead wrinkles. This is possible, because the Gouraud algorithm performs an interpolation of colors between horizontal darkest and lightest zones, determined by the normal orientations that are controlled by our perturbation function. The function is rather complicated and, based on the regular disposition of vertexes in the forehead region, uses many periodic functions; but more explicatory are the Fig. 25-27 that describe a bulge (created by the techniques of the previous section) with realistic wrinkles and the Fig. 28 that shows the wrinkles on the forehead of Greta during the raising of the eyebrows.

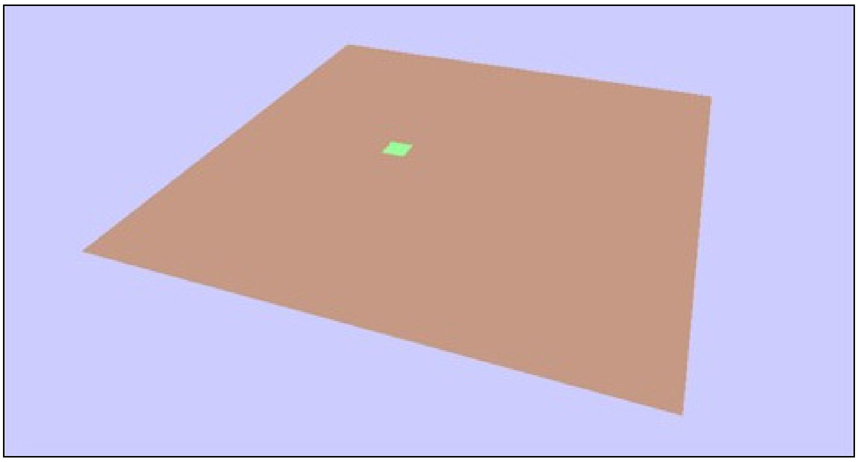

**Fig. 25.** Generation of a bulge with wrinkles (null intensity of the FAP).

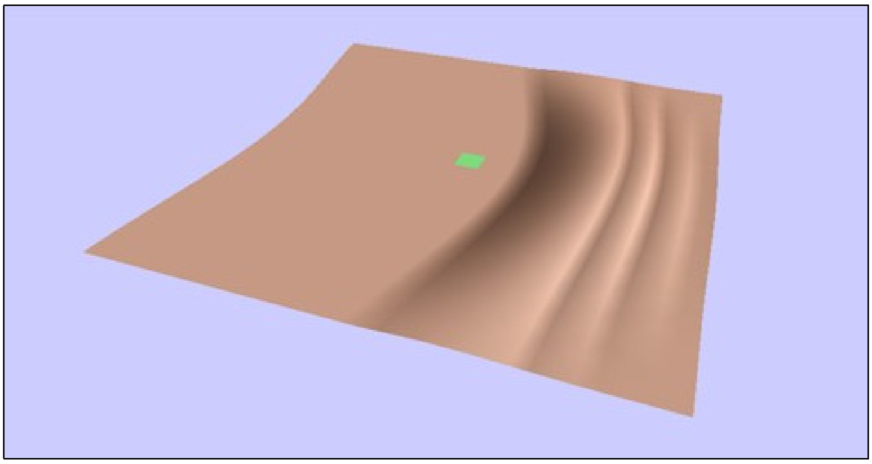

**Fig. 26.** Generation of a bulge with wrinkles (medium intensity of the FAP).

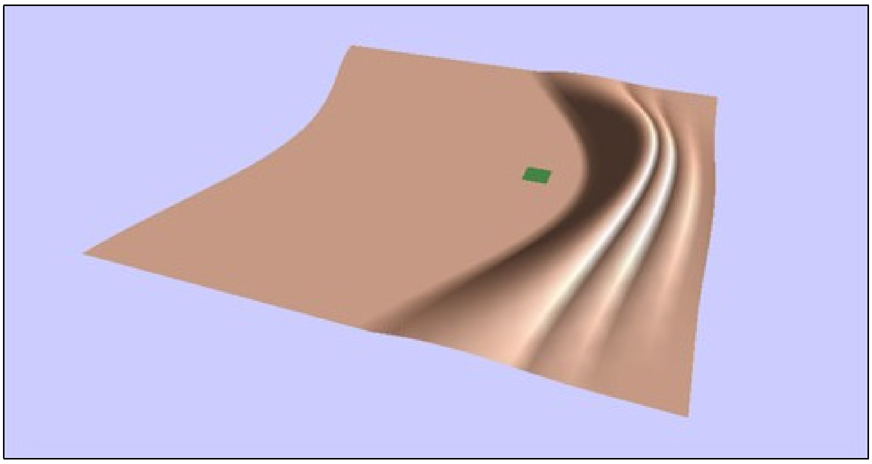

**Fig. 27.** Generation of a bulge with wrinkles (maximum intensity of the FAP).

**Fig. 28.** Generation of the wrinkles on the Greta forehead.

# 5 FAPs in Action

The action of the FAPs on the whole 3D model is obtained by the application of the previous explained techniques; the extension from the plane polygonal lattice to a complex 3D polygonal face is not difficult. After computing the displacement that all the FAPs produce for each vertex, is straightforward to calculate the new vertex coordinates and displaying the deformed Greta face using the OpenGl Gouraud shading. This procedure is performed for each frame of the animation.

## 5.1 Application of FAPs on the 3D Model

Generally the displacements $\Delta x_j$ (we have a similar situation for $\Delta y_j$ and $\Delta z_j$) of a vertex $P_j$ $(x_j, y_j, z_j)$ due to a single FAP operating in the $x$ direction is computed using three kind of information:

- The intensity of the FAP in the related feature point.
- The position of the vertex in the "area of influence" of the feature point.
- The position within a "specific area".

The displacement $\Delta x_j$ of a vertex $P_j$ $(x_j, y_j, z_j)$ in the *k-th* specific area due to a FAP operating in the $x$ direction $FAP_x$ is computed as:

$$\Delta x_j = W_j * W'_k * FAP_x \tag{3}$$

The weight $W_j$ is based on the distance of the vertex from the feature point and it is computed by the function seen in Fig. 15, the weight $W'_k$ is defined by the designer after a calibration of the animation and describe how the intensity of the FAP influences the displacement of the vertexes of the *k-th* "specific area". For example, if we want that the vertexes of the eyelids don't be affected by the FAPs that control eyebrows is possible to set $W_k = 0$ during the computation of the $\Delta x_j$ when the above mentioned FAPs operate; this is possible because the vertexes of the eyebrows and eyelids belong to distinct specific areas (see Fig. 12-13) and, so, the movements of the vertexes of the eyebrows don't affect the vertexes of the eyelids. This approach gives a great control on the displacement of the vertexes and allows a natural animation of the 3D face without visual artifacts during the rendering.

### 5.2 Application of auxiliary deformations

Greta uses auxiliary deformations to increase the realism of the animation in specific parts of the model. These auxiliary deformations perform a displacement of vertexes in the same manner of the FAPs, the only difference is that the intensity of the displacement is not controlled by the FAP stream but is a function of various weighted FAPs. These auxiliary deformations are used, during specific actions, to increase the realism of the behaviour of the face skin in two zones:

- The zone between the eyebrows during the act of frowning (see Fig. 29).
- The zone of the cheeks that swells out during the act of smiling.

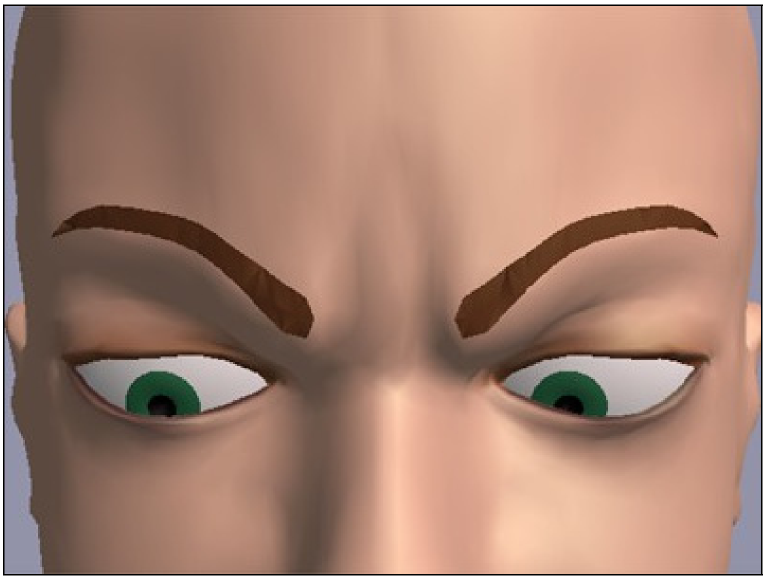

**Fig. 29.** The frown of Greta.

## 5.3 Example of multiple displacement

The zone of the eyebrow is a good example of multiple displacement of the vertexes due to the FAPs and to the auxiliary deformations combined with the wrinkles generation.

Setting increasing values on FAP 31 (raise_l_i_eyebrow), FAP 33 (raise_l_m_eyebrow) and FAP 35 (raise_l_o_eyebrow) we can notice the generation of wrinkles on the forehead of Greta (see Fig. 30-32). This is a simulation of the action of the *frontalis* muscle.

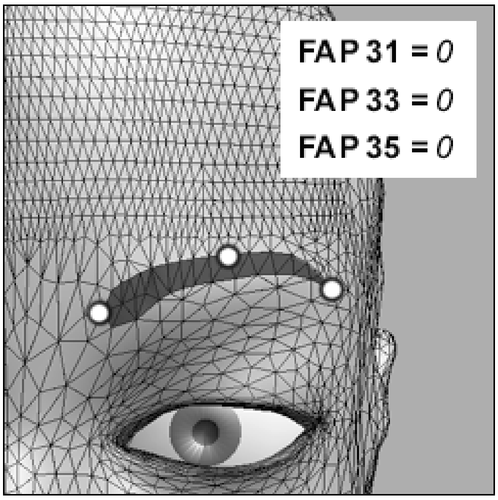


**Fig. 30.** The action of the *frontalis* muscle (1).

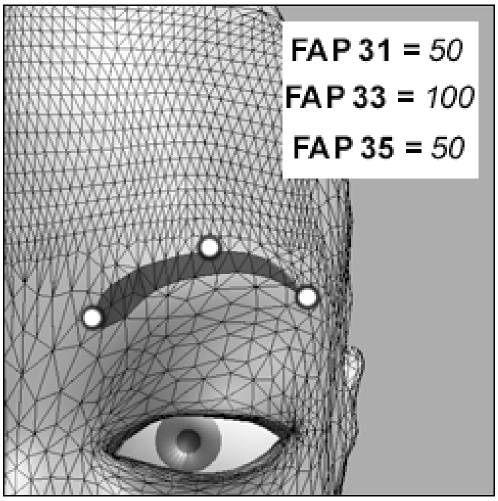


**Fig. 31.** The action of the *frontalis* muscle (2).

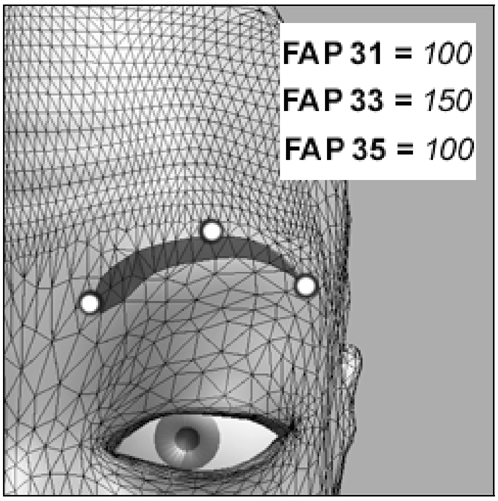


**Fig. 32.** The action of the *frontalis* muscle (3).

Setting increasing values on FAP 37 (squeeze_l_eyebrow) we can notice the frowning of Greta (see Fig. 32-34). This is a simulation of the action of the *corrugator* muscle.

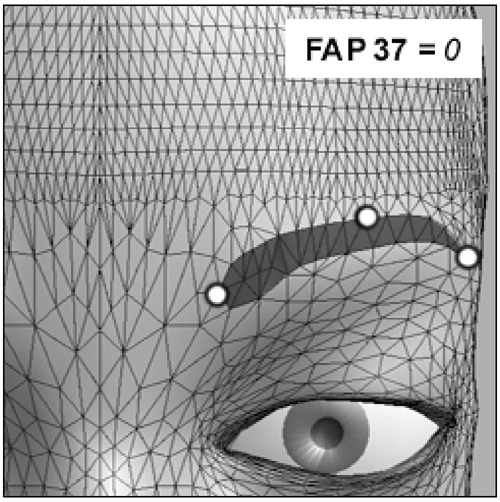


**Fig. 33.** The action of the *corrugator* muscle (1).

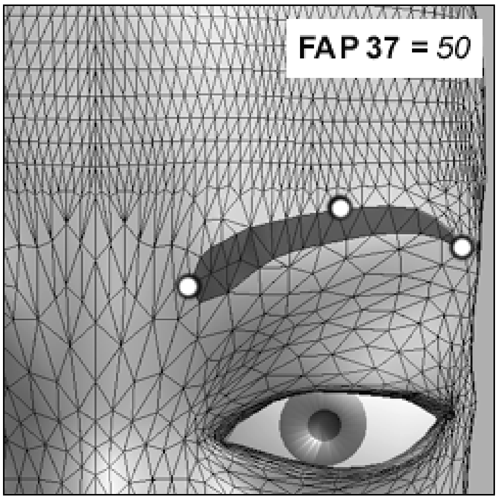


**Fig. 34.** The action of the *corrugator* muscle (2).

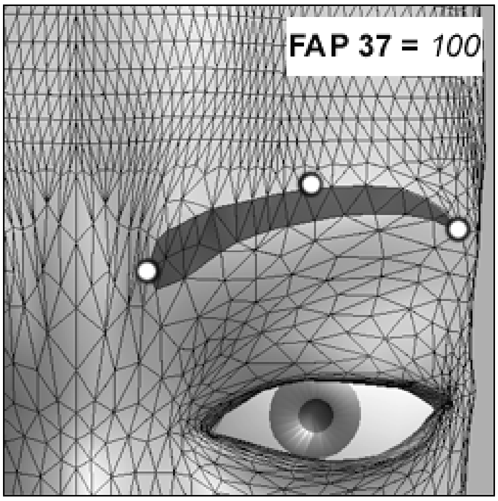


**Fig. 35.** The action of the *corrugator* muscle (3).

## 6 Conclusions

The facial model of Greta is made of 15,000 triangles. The high number of polygons allows us to maintain a great level of detail in the most expressive regions of human face: the eyes and the mouth. Indeed, these regions play an important role in the communication process of human-human conversation and in expressing emotions (see Fig 36-37).

The graphic engine of Greta has a conception similar to others MPEG based projects that were previously realized [7]. The novelty of Greta is the high quality of the 3D model, and the generation in real-time of wrinkles, bulges and furrows, enhancing the realism and the expressive look to the animated face. This work is part of an ongoing research on the creation of an embodied conversational agent, that is an agent able to communicate with other agent(s) or user(s) as well as to express its emotion [3].

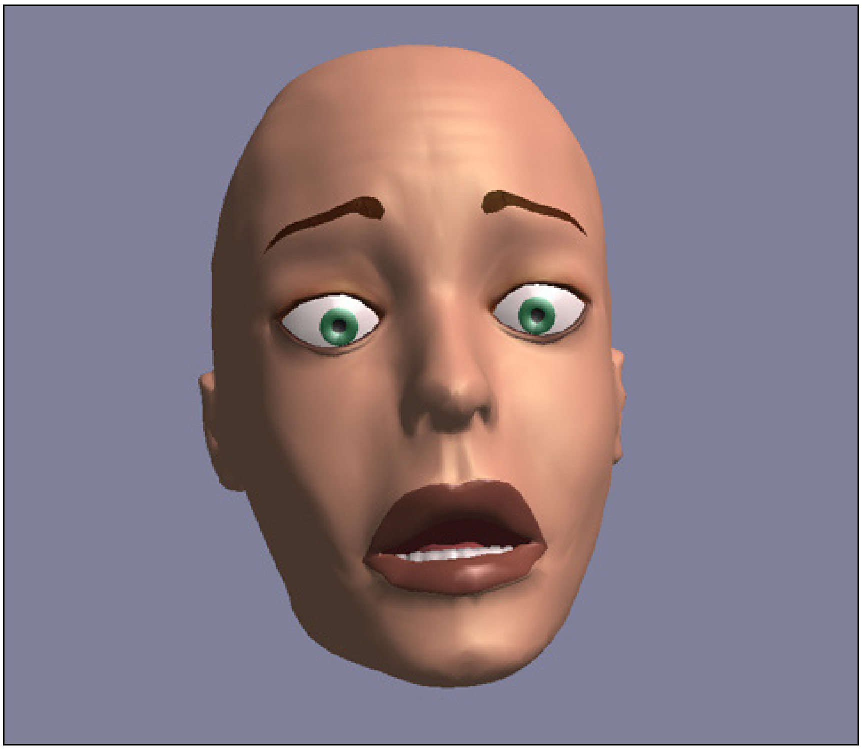

**Fig. 36.** The fear of Greta.

**Fig. 37.** The joy of Greta.

## References


1. Cohen MM, Massaro DW (1993) Modeling coarticulation in synthetic visual speech. In M. Magnenat-Thalmann and D. Thalmann, editors, Models and Techniques in Computer Animation, Tokyo, Springer-Verlag
2. Davis N, Davis J, Davis T, Woo M (1993) OpenGL Programming Guide: The Official Guide to Learning OpenGL, Release 1 ("Red Book"). Reading, MA: Addison-Wesley ISBN 0-201-63274-8
3. De Carolis N, Pelachaud C, Poggi I, De Rosis F (2001) Behavior planning for a reflexive agent. In IJCAI'01, Seattle, USA, August 2001
4. Doenges P, Lavagetto F, Ostermann J, Pandzic IS, Petajan E (1997) MPEG-4: Audio/video and synthetic graphics/audio for mixed media. Image Communications Journal, 5(4), May 1997
5. Ekman P, Friesen WV (1978) Manual for the Facial Action Coding System. Consulting Psychologists Press, Inc., Palo Alto
6. Faigin G (1990) The Artist's Complete Guide to Facial Expression. Watson-Guptill, New York

7. Lavagetto F, Pockaj R (1999) The facial animation engine: towards a high-level interface for the design of MPEG-4 compliant animated faces. IEEE Trans. on Circuits and Systems for Video Technology, 9(2):277-289, March 1999
8. Magnenat-Thalmann N, Primeau NE, Thalmann D (1988) Abstract muscle actions procedures for human face animation. Visual Computer, 3(5):290-297
9. Ostermann J (1998) Animation of synthetic faces in MPEG-4. In Computer Animation'98, pages 49-51, Philadelphia, USA, June 1998
10. Parke FI, Waters K (1996) Computer Facial Animation. A. K. Peters, Wellesley, MA
11. Reeves B (1990) Simple and complex facial animation: Case studies. In Vol 26: State of the Art in Facial Animation, pages 90-106. ACM Siggraph'90 Course Notes
12. Sederberg TW, Parry SR (1986) Free-form deformation of solid geometry models. Computer Graphics (SIGGRAPH '86), 20(4):151-160
13. Waters K (1987) A muscle model for animating three-dimensional facial expressions. Computer Graphics (SIGGRAPH '87), 21(4):17-24, July