Synthesizing facial expressions for signing avatars using MPEG4 feature points

Conference Paper · October 2013

DOI: 10.1109/ICTA.2013.6815304

Yosra Bouzid
Université de Tunis

Mohamed Jemni
Arab League Educational, Cultural and Scien…

Synthesizing Facial Expressions for Signing Avatars using MPEG4 Feature Points

Yosra Bouzid[#1], Oussama El Ghoul[#2], Mohamed Jemni[#3]

[#] *Research Laboratory LaTiCE, University of Tunis*
High School of sciences and techniques of Tunis

[1] yosrabouzid@hotmail.fr

[2] oussma.elghoul@rnu.tn

[3] Mohamed.jemni@fst.rnu.tn

Abstract— **Thanks to the advances in virtual reality and human modeling techniques, signing avatars have become increasingly used in a wide variety of applications like the automatic translation of web pages, interactive e-learning environments and mobile phone services, with a view to improving the ability of hearing impaired people to access information and communicate with others. But, to truly understand and correctly interpret the signed utterances, the virtual characters should be capable of addressing all aspects of sign formation including facial features which play an important and crucial role in the communication of emotions and conveying specific meanings. In this context, we present in this paper a simple yet effective method to generate facial expressions for signing avatars basing on the physics-based muscle model introduced by Keith Waters. The main focus of our work is to automate the task of the muscle mapping on the face model in the correct anatomical positions as well as the detection of the jaw part by using a small set of MPEG-4 Feature Points of the given mesh**

Keywords— **Sign language, signing avatar, facial expressions, MPEG-4, features points, physics-based approach**

I. Introduction

The past few years have seen a growing interest in the use of 3D signing avatars because of the important role they play in overcoming the communication problems faced by deaf and hearing impaired people in some aspects of their daily lives. To bring the information on signing as closer to the visual nature of sign language, these digital humanoids are required to perform not just broad arm and hand movements, but also many subtle clues and features like face movements and expressions, which must be clearly seen in order to understand the meaning. The inclusion of these non manual gestures can alter the meaning of a sign and its absence can render a sign meaningless.

In fact, there are several techniques that can be used to create facial animations, but most require a significant effort and time-consuming to adjust animation parameters. For example, the process of rigging requires many hours of manual work to set up the bone structure for an entire face. Even simple method like shapes blending needs the intervention of an artist to create the desired key shapes. On the other hand, the majority of existing techniques involve a high computational complexity to generate expressive and plausible animations, which makes them inappropriate for real time and interactive applications.

In this context, we present in this paper a simple yet effective method capable of deforming a 3D facial mesh of an arbitrary synthetic human face to generate emotional expressions without considerable amount of manual intervention and artistic skill. This approach relies on the physics-based muscle model proposed by Keith Waters to emulate the contraction of the muscle onto the skin surface. Our contribution consists essentially on the automatic construction of mimetic muscles as well as the jaw mesh detection using only MPEG4 feature points of the given mesh.

II. Related and Previous Work

Since the pioneering work of Frederic I. Parke in 1972 [1], different approaches have been reported for modeling the geometry of faces and for animating them such as the Blend shapes, direct parameterizations and muscle based modeling.

The blend shape animation method, also known as morph target animation or shape interpolation is the most intuitive and commonly used technique in the field since it is quite straightforward and easy to accomplish. The fundamental of this method is that during the animation, the interpolated facial model is created from a specific set of key facial poses called shapes, through interpolation over a normalized time interval. Typically, a blend shape model is the linear weighted sum of a number of topologically conforming shape primitives. Varying the weights of this linear combination allows the representation of facial motions with little computation. However, it is important to note here that the generation of a significant range of highly detailed expressions usually implies the creation of large libraries of blend shapes which can be very time-consuming. Moreover, if the topology of the model needs to be changed, all the shapes must be redone [2].

Direct parameterization models could offer an alternative approach to these limitations by directly manipulating a few selected parameters on the face geometry, for instance the vertices of polygon mesh. Probably the most famous frameworks which can be included in this category are the Facial Action Coding System [3] and the MPEG-4 Facial Animation standard. The advantage of these parameterizations is that once control parameters are determined, they provide a detailed control over the face. But determining this is hard: complexity of creating an animation with these control parameters is related to the number of control parameters, as is the possible range of expressions [4]. Furthermore, the animation does not seem to respond to basic physical deformations of human faces since direct parameterizations make no attempt to represent the detailed anatomical structure. They model only the changes visible on the skin surface.

Physics-based muscle methods have the potential to produce natural 3D animations by precisely simulating the real effects of the facial muscular tissues. They can generally fall into three different categories: mass spring systems, layered spring meshes and vector representations. Mass-spring systems [5] are designed to propagate muscle forces in an elastic spring mesh that models the elastic properties of the skin, while, layered spring meshes [6] extends the mass spring structure into three connected mesh layers to simulate the anatomical aspects of the face more faithfully. In vector representations, the actions of muscles upon skin are modelled using motion fields in delineated regions of influence. A very successful muscle model was proposed by Waters [7] in 1987. It includes essentially two types of muscles: linear muscles that pull and sphincter muscles that squeeze. Another complete anatomically based face model is provided by Kähler et al.[8] who propose a muscle structure composed by quadric segments. But, although these techniques are the most scientifically based, they are also among the most difficult to implement. The construction of the anatomical facial structure is an extremely tedious task which requires artistic skills and a massive computation.

III. MPEG-4

One of the more revolutionary parts of the recently released MPEG-4 International Standard is the Face and Body Animation, or FBA: the specification for efficient coding of shape and animation of human faces and bodies [9]. MPEG-4 FBA can specify a face model in its neutral state with three sets of parameters:

- The Facial Animation Parameters (FAPs) are used to direct control the face movements. They are based on the study of minimal perceptible actions (MPA) and are closely related to muscle actions, such as movements of lips, jaw, cheeks and eyebrows. They make up a complete set of basic facial actions that represent the most natural facial expressions. The FAP set contains two high level FAPs for selecting facial expressions and visemes, and 66 low level FAPs for denoting small facial motion.
- The Facial Definition Parameters (FDPs) are needed for the calibration of a synthetic face. These parameters are scalable; they can define the shape, texture or even the whole facial polygon mesh.
- The feature Points (FPs) are used to describe and define the shape of a standard face. There are a total of 84 feature points in a head model. They are subdivided in groups, mainly depending on the particular region of the face to which they belong. Each of them is labelled with a number identifying the particular group to which it belongs and with a progressive index identifying them within the group. A subset of these points can be affected by the facial animation parameters (FAPs) to control the animation.

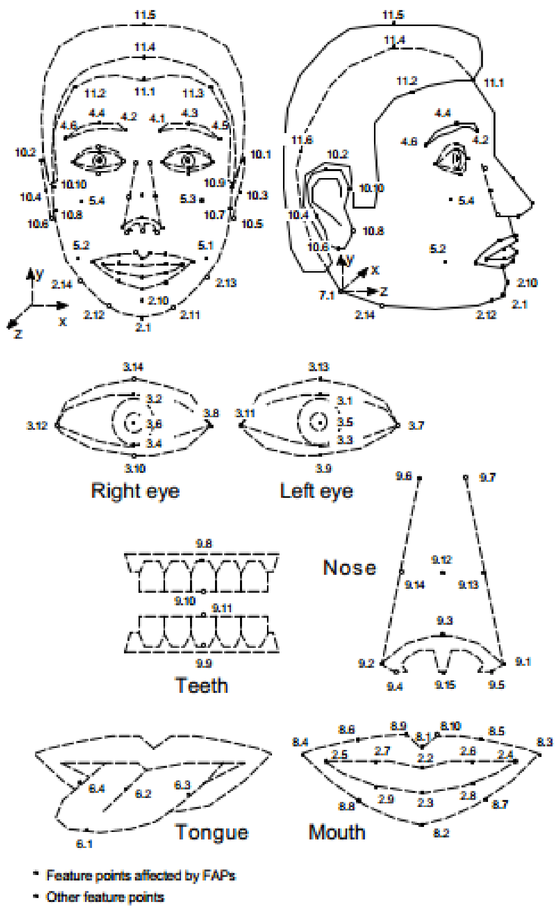

Fig. 1 The location of the 84 FPs on a neutral face as defined by the MPEG-4 standard.

IV. Approach

Our approach aims to perform realistic facial animation automatically by contracting the mimetic muscles. The face model is a single layered mesh with no skeleton, it is expected to be triangular or quadratic polygons, high-poly or low-poly mesh. To elaborate the facial musculature, we have followed the physics-based muscle modeling approach proposed by Waters. Our contribution consists on specifying a mechanism to place the muscle vectors in the anatomically correct positions by using the MPEG-4 Feature Points.

A. *Muscle Modeling.*

To emulate the behaviour of muscles upon skin, Waters presents one of the most popular and complete parametric muscle models that are based on the human facial anatomy. This model is computationally cheap and easy to implement, it defines two types of muscles based on their natural actions: linear and sphincter. These muscles were embedded in the surface and were independent of the bone structure, so it was possible to transfer them to face models with different topologies [7]. An individual muscle can be defined by two key nodes, an area of influence which presents a skin portion affected by the contraction, and a deformation formula for all influenced vertices.

The linear muscle is modeled as a vector from a bony attachment point that remains static, to an insertion point which is embedded in the soft tissue of the skin. Its influence area is represented by a cone shape (Fig.2). For example the Zygomaticus Major which acts to draw the angle of the mouth up and back to smile or laugh, is a linear muscle.

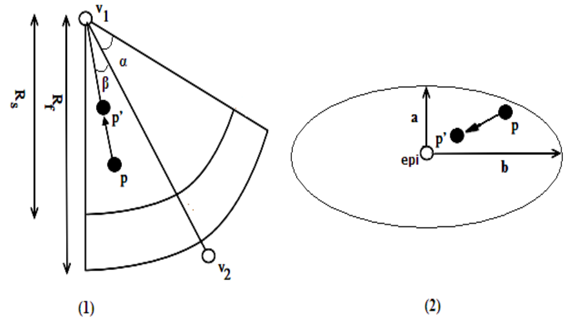

Fig. 2 Representation of Waters muscle model: (1) linear muscle (2) sphincter muscle

When a linear muscle contracts, all points in its influence area are displaced towards its point of attachment. The displacement of a point p affected by the muscular action is given by the Equation 1 (k is a fixed constant representing the elasticity of the skin).

$$p' = p + a\,r\,k\;\overrightarrow{pv_1} \quad (1)$$

$$a = \frac{(\cos\beta - \cos\alpha)}{(1 - \cos\alpha)}$$

$$r = \begin{cases} \cos\left(\dfrac{1 - \|v_1 p\|}{R_s} \cdot \dfrac{\pi}{2}\right), \|v_1 p\| < R_s \\ \cos\left(\dfrac{\|v_1 p\| - R_s}{R_f - R_s} \cdot \dfrac{\pi}{2}\right), \quad else \end{cases}$$

For sphincter muscles, we can identify only two instances in a human face: the Orbicularis Oculi muscle around each eye and the Orbicularis Oris which circles the mouth. This kind of muscle attaches to the skin both at the origin and at the insertion. Its influence area has an elliptical shape defined by a virtual center and two semi-axes (Fig. 2). When a sphincter muscle contracts, the points in its influence area are displaced towards the center of the spheroid. The displacement of a point p affected by the action of muscle is given by the Equation 2 (k is a fixed constant representing the elasticity of the skin).

$$p' = p + r\,k\;\overrightarrow{po} \quad (2)$$

$$r = cos\left(\left(1 - \frac{\sqrt{{p_x}^2 b^2 + {p_y}^2 a^2}}{ab}\right) \cdot \frac{\pi}{2}\right)$$

It is important to note that Waters combined these muscle actions sequentially by applying the displacements caused by them on a vertex one by one. Nevertheless, this process can produce undesirable effects especially when a mesh vertex is under the influence of multiple muscle actions: the vertex will shifted out of the influence area of adjoining muscle vectors.

So, to avoid the undesired effects, we have used the Wang approach [10] which summarizes the displacements and then applies it the vertex.

B. *Muscle Construction*

The main objective of our work is to automate the challenging task of correct muscle mapping on the face model. The use of MPEG-4 features points as key nodes of each muscle will certainly reduce the amount of work that must be done manually by animators. The key nodes will be identified according to the anatomy that characterizes muscle properties. Thereby, the central muscle fiber, the influence area as well as the minimum and the maximum displacement of influenced vertices will be automatically detected for each input mesh.

Our face model comprises essentially 31 muscles including three sphincter muscles that are used to represent the orbicularis oris and orbicularis oculi, and 11 pairs of linear muscles that are placed symmetrically through the face to accomplish the major face movements. The remaining linear muscles, namely Cheek Sup, Cheek Center and Cheek Inf, are located on each cheek and don't exist in a real human face. They have been added to our model to simulate specialized expressions in sign languages like cheek movements. The complete facial muscle structure is shown in Fig.3.

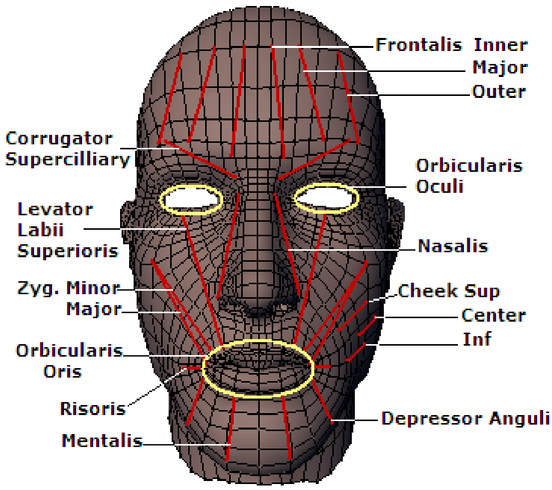

Fig.3 The Facial musculature

The construction of the proposed musculature involves two basic steps. At first, the anatomical positions of the muscle control points will be defined with the suitable MPEG-4 Feature Points. Second, the set of vertices that belong to each influence area will be detected.

For a linear muscle, three points are needed to define its location on the input face mesh: an attachment point AP, an insertion point IP and a reference point RP which was not used in Waters model, we have added it in our method to facilitate the determination of the mesh part affected by the muscle action. The obtained properties of muscle vectors are given in Table 1. For instance, the Left Nasalis which depresses the cartilaginous part of the nose is characterized by:

- an AP which coincides with the feature point 9.7 located in the left upper edge of nose bone
- an IP which coincides with the feature point 9.1 located in the left nostril border.
- an RP which coincides with the feature point 9.3 located in the nose tip (Fig. 4).

Similarly, each sphincter muscle is defined by three key points: the epicenter of the spheroid EP, a semi-major axis SJ and a semi-minor axis SN. The obtained properties of sphincter muscles are shown in Table 2. For instance, the Orbicularis Oculi Left which closes the eyelids of the left eye is characterized by (Fig. 5):

- an EP which coincides with the midpoint of the line segment formed by the two points 3.7 and 3.11
- SJ is equal to the length of the line segment formed by the two points 3.7 and EP
- SN is equal to the length of the line segment formed by the two points 3.9 and EP.

TABLE I
LINEAR MUSCLE PROPERTIES (RIGHT AND LEFT)

Muscle name	**Attachment point**	**Insertion point**	**Reference point**
Frontalis Inner	$11.1+\frac{1}{6}(11.2-11.1)$	4.2	$4.2+\frac{1}{2}(4.4-4.2)$
	$11.1+\frac{1}{6}(11.3-11.1)$	4.1	$4.1+\frac{1}{2}(4.3-4.1)$
Frontalis Major	$11.1+\frac{2}{3}(11.2-11.1)$	$4.2+\frac{1}{2}(4.6-4.2)$	4.6
	$11.1+\frac{2}{3}(11.3-11.1)$	$4.1+\frac{1}{2}(4.5-4.1)$	4.5
Frontalis Outer	11.2	4.6	4.4
	11.3	4.5	4.3
C.S	$4.2+\frac{1}{2}(4.2-4.4)$	4.4	$4.2+\frac{1}{2}(4.2-3.8)$
	$4.1+\frac{1}{2}(4.1-4.3)$	4.3	$4.1+\frac{1}{2}(4.1-3.11)$
Nasalis	9.6	9.2	9.3
	9.7	9.1	9.3
Levator Labii Superioris	3.10	2.7	$8.9+\frac{1}{3}(8.6-8.9)$
	3.9	2.6	$8.10+\frac{1}{3}(8.5-8.10)$
Z. Minor	9.2	2.7	8.4
	9.1	2.6	8.3
Z. Major	5.4	2.9	9.15
	5.3	2.8	9.15
Risoris	5.2	2.5	$2.5+\frac{1}{3}(2.5-9.2)$
	5.1	2.4	$2.4+\frac{1}{3}(2.4-9.1)$
Depressor Anguli Oris	$8.4+(8.4-8.6)$	8.6	8.8
	$8.3+(8.3-8.5)$	8.5	8.7
Mentalis	$2.1+\frac{1}{2}(2.1-2.12)$	2.9	8.2
	$2.1+\frac{1}{2}(2.1-2.11)$	2.8	8.2
Cheek Inf	$K+\frac{1}{2}(K-2.11)$	$K=5.1+(5.1-8.2)$	5.1
Cheek Center	$K+\frac{1}{2}(K-2.13)$	$K=5.1+(5.1-8.2)$	5.1
Cheek Sup	$K+\frac{1}{2}(K-2.11)$	$K=5.1+(5.1-8.3)$	5.3

TABLE II
SPHINCTER MUSCLE PROPERTIES

Muscle name	**Semi-major axis**	**Semi-minor axis**	**Epicenter**
Orbicularis Oculi	3.12	3.10	$\frac{1}{2}(3.8+3.12)$
	3.7	3.9	$\frac{1}{2}(3.7+3.11)$
Orbicularis Oris	8.3	8.2	$\frac{1}{2}(8.3+8.4)$

Once the positions of muscle control points are computed and mapped on the facial mesh, we can determine then the set of vertices which will be influenced by the muscle contraction. For a linear muscle, all influenced vertices should verify the following conditions (see Fig. 2):

$$\|\overrightarrow{pv_1}\| > 0,\ \|\overrightarrow{pv_1}\| \le Rf,$$
$$\cos\beta \ge \cos\alpha$$

Similarly, all influenced vertices of a circular muscle should be within its spheroid (see Fig. 2).

$$\left(\frac{p_x - epi_x}{a}\right)^2 + \left(\frac{p_y - epi_y}{b}\right)^2 + \left(\frac{p_z - epi_z}{\sqrt{a^2 - b^2}}\right)^2 < 1$$

Fig. 4 shows the obtained result after applying the proposed algorithm on Nasalis muscle. The face in the left, illustrates the set of vertices having a distance from AP less than the length of muscle fiber. The second face illustrates the set of vertices that will be displaced when muscle contracts.

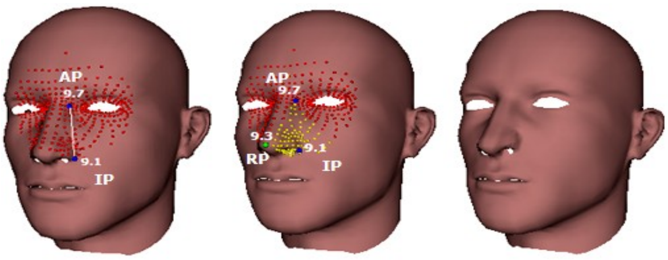

Fig.4 The set of vertices that belong to the influence area of Nasalis Left is colored in yellow

C. *Jaw Articulation*

As we mentioned above, the face model does not have a skull, so the jaw is not a particular mesh. It will be rather detected automatically from the initial mesh. To do so, we have based on some features points for approximating the set of vertices that are affected by the jaw rotation. However, when the jaw is opened, the vertices of the chin and the lower lip are rotated to give the effect of mouth opening

1) *Jaw Detection*: To define the chin vertices, we have used the following procedure: first, we project the facial mesh on the plane P passing through the midpoint of the segment joining the FPs 10.8 and 10.7, with a normal vector defined by this midpoint and the vertex 10.8 in order to get a profile view.

Second, the projections of 2.14, 10.8 and 8.3 respectively p1, p2 and p3 are marked on the projected mesh. The vertices whose projects are inside the angular sector $(p1, \widehat{p2}, p3)$ are considered to be in the chin influence. (see Fig. 5*)*

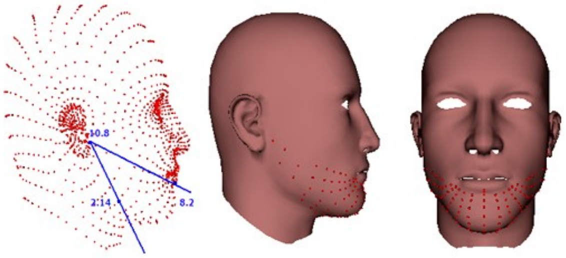

Fig. 5 The detection of chin vertices

The process is about the same for the lower lip but by using other feature points. Indeed, the vertices located on the inner contour of the lower lip should also be taken into account since the projection is incapable to detect them. The extraction of these vertices is done as follows: the algorithm will browse all the edges of the mesh to find those that belong only to one surface. The selected edges depict the contours of the facial model such as the openings of the eyes, nose as well as the space between the two lips. Using this set of edges and some MPEG-4 FPs, we can distinguish the inner contour of the lower lip. All we have to do is finding the closest edge to the point 2.3, and then its neighboring segments. For each new segment found, we perform the same process until finding edges closest to the points 8.3 and 8.4 which define the corners of the lips.

Fig. 6 The extraction of the inner contour of lower lip

2) *Jaw Rotation:* The jaw rotation is done by rotating the vertices of the chin and the lower lip around a line passing through the feature point 10.8 and parallel to the X axis. Thus, their final positions are calculated by the following equation:

$$\begin{pmatrix} x \\ y' + y1 \\ z' + z1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & -\sin\varphi \\ 0 & \sin\varphi & \cos\varphi \end{pmatrix} * \begin{pmatrix} x \\ y - y1 \\ z - z1 \end{pmatrix} \quad (3)$$

where Φ represents the degrees of rotation and (x1, y1, z1) the coordinates of 10.8.

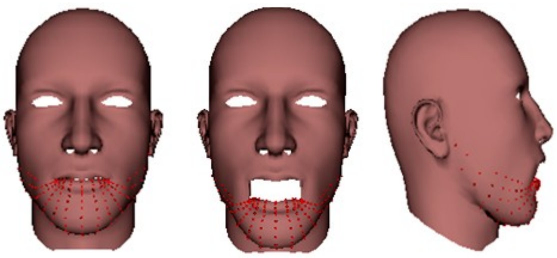

Fig. 7 The jaw rotation

V. Evaluation

The proposed approach aims to generate basically the expression of six emotions: happiness, sadness, anger, fear, disgust and surprise. The production of these emotions is the result of a contraction or relaxation of one or more facial muscles. Fig. 8 shows some examples of basic facial expressions on different face models, while Fig. 9 illustrates the contraction of Cheek Sup, Cheek Center and Cheek Inf which are used to emulate the tongue motion on right cheek.

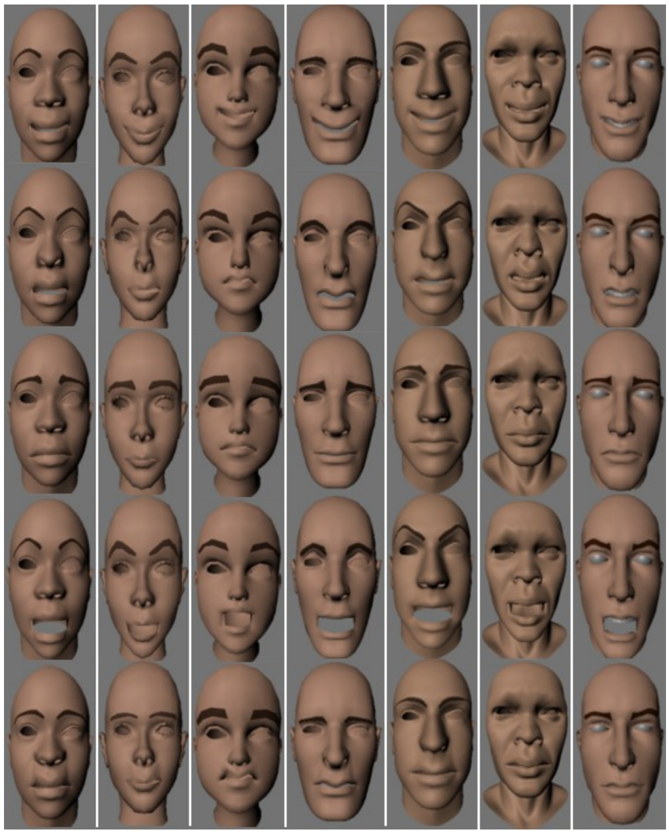

Fig. 8 The simulation of some basic facial expressions on different face models: happiness, anger, sadness, surprise and disgust

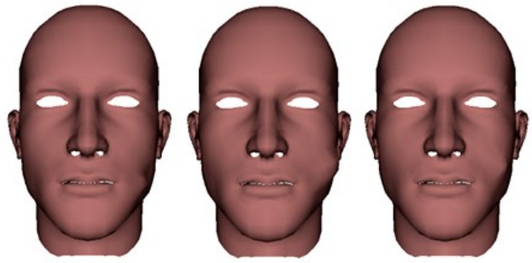

Fig. 9 The contraction of cheek Sup, Cheek Center and Cheek Inf

In order to validate our approach, we have tested the performance of facial muscles on different face models. The goal of this evaluation is to check the choice accuracy of features points in the definition of muscle key nodes as well as the fitting of those features with anatomically correct positions on the face. Fig. 10 shows the recognition rate of each muscle motion is calculated with 50 models.

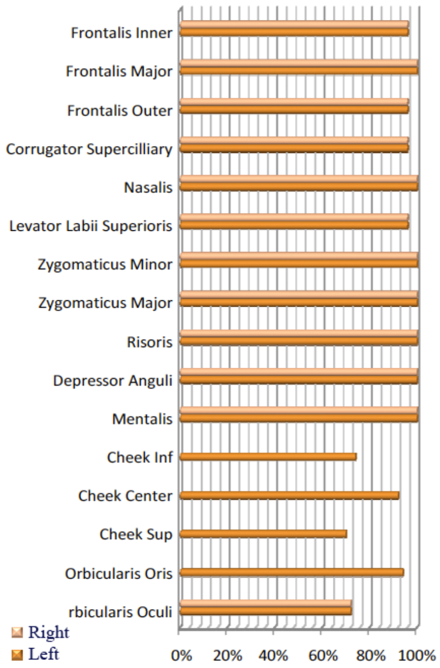

Fig. 10 The recognition rates of muscle motion on 50 face models

We can notice that the contraction effects of the vast majority of muscles have been simulated. For the Frontalis Major, Nasalis, Zygomaticus Major, Zygomaticus Minor and Risoris, our approach achieves 100% recognition rate, whereas, the Cheek Inf, Cheek Sup and Orbicularis oculi achieve recognition rates ranging from 70% to 75%. This is can be explained by the fact that muscular activities do not give the desired animations for some meshes.

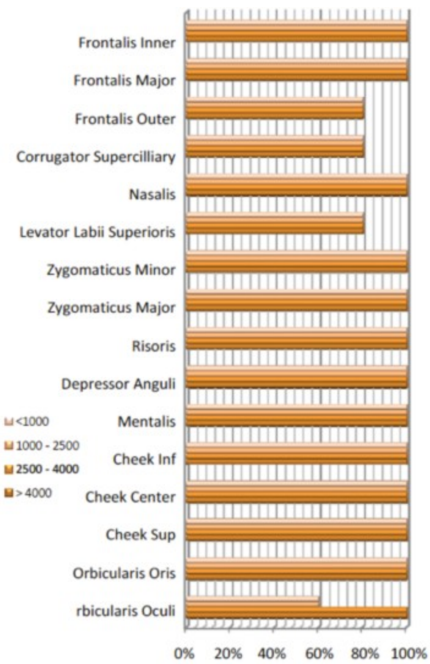

Fig. 11 The recognition rates of muscle motion on low and high poly meshes

In order to study the sensitivity of muscle performance on high-poly and low-poly meshes, we have used three sets of face models that have the same appearance, the same polygonal resolution, but with different number of vertices: less than 1000 vertices, between 1000 and 4000 vertices and over than 4000 vertices. The obtained result is drawn in Fig.11. It is clear that most muscles are insensitive to the changes in the number of vertices, with the exception of Orbicularis Oculi. This is can be explained by the fact that in low-poly meshes, the eyelids and eyebrows have common polygons.

VI. Conclusion

We have presented in this paper an automatic facial animation method based on Waters vector model and some MPEG-4 feature points. The experimentation shows that more than 90% of tested faces can be animated without any human intervention. However, for some low-poly meshes we need to adjust the influenced area of sphincter muscles. To this end, we aim in our future work to modify the sphincter muscle model by ameliorating the algorithm used to detect its influence area. It should be noted that the proposed method depicts one module of the WebSign project [12][13] which renders sign language animations in real time using a virtual avatar, form a writing text or a SignWriting notation [14].

References

[1] F.I. Parke. Computer generated animation of faces. In Proceedings of the ACM annual conference-Volume 1, pages 451-457, USA, 1972.

[2] H. Y. Ping, L. N. Abdullah, P. S. Sulaiman, and A. Abdul Halin, “Computer Facial Animation: A Review,” in International Journal of Computer Theory and Engineering, Vol. 5, No. 4, August 2013

[3] P. Ekman and W. Friesen. Manual for the Facial Action Coding System. Consulting Psychologist, Palo Alto, California, 1977

[4] M.D. Ilie, C. Negrescu, and D. Stanomir, “Energy minimization tool for generating composite facial expressions in 3D facial animations”, International Journal of Computer Theory and Engineering, 2012

[5] S.M. Platt and N.I. Badler. Animating facial expressions. In Proceedings of the 8th annual conference on computer graphics and interactive techniques, pages 245{252. ACM New York, NY, USA, 1981 D.

[6] D. Terzopoulos and K. Waters. Physically-based facial modeling, analysis, and animation. Journal of visualization and Computer Animation, 1990

[7] K. Waters. A muscle model for animation three-dimensional facial expression. ACM SIGGRAPH Computer Graphics, 1987.

[8] K. Kähler. 3D Facial animation-Recreating human heads with virtual skin, bones, and muscles. Verlag, October, 2007

[9] I.S. Pandzic and R. Forchheimer. MPEG-4 facial animation: the standard, implementation and applications, New York, 2002.

[10] CL. Wang.: A hierarchical spline based facial animation system with simulated muscles. Master's thesis, University of Calgary, 1993

[11] M. Fratarcangeli. Physically based synthesis of animatable face models. Workshop on Virtual Reality Interaction and Physical Simulation, 2005

[12] A. Othmen, O. Elghoul and M. Jemni. An automatic approach for facial feature points extraction from 3D head. In GRAPP/IVAPP 2012, pages 369-372, Italy, 2012.

[13] O. Elghoul and M. Jemni. Automatic generation of sign language's facial expression. In proceeding of ICTA 2009, pages 143-146, Tunisia

[14] Y. Bouzid and M. Jemni, An avatar based approach for automatically interpreting a sign language notation. The 13th IEEE International Conference on Advanced Learning Technologies, China, July, 2013