A NURBS-based Vector Muscle Model for Generating Human Facial Expressions

Sy-sen Tang\(^1\), Hong Yan\(^1,2\), Alan Wee-Chung Liew\(^1\), et al.

\(^1\)Department of Computer Engineering and Information Technology
City University of Hong Kong, 83 Tat Chee Avenue, Kowloon, Hong Kong
\(^2\)School of Electrical and Information Engineering
University of Sydney, NSW 2006, Australia

Abstract

In this paper, a novel method based on non-uniform rational B-spline (NURBS) curves is proposed to simulate a physics-based muscle model for facial expression animation. In our system, non-uniform rational B-spline curves are used to construct facial muscles based on the anatomical knowledge. By changing the weights of the NURBS curves, we can form a motion vector to control facial muscle movements. With different number of control points, muscles are more controllable so that more detailed facial expression and mouth shapes can be simulated. Our approach can simulate different types of muscles including linear and sphincter muscles, as well as the non-linear part of the face like fatty tissue.

1. Introduction

Facial animation has been widely used in different areas such as entertainment and teleconferencing [1, 2]. Facial modeling and animation by geometric manipulations can be roughly classified into five categories [3], i.e., interpolation-based, parameterization-based, pseudo-muscle modeling, performance-based and physics-based modeling. Interpolation techniques use in-between method to generate the frames between two key-frames [4]. It is fast and relatively easy to generate primitive facial animation but difficult to create a wide range of realistic facial configurations. Parameterization techniques use a set of independent parameter values to specify any possible facial expressions [5, 6]. However, there might be conflict between parameters when they affect the same vertices. The pseudo-muscle modeling method applies geometric deformation like free form deformation (FFD) animation [2]. As it does not follow the actual facial anatomy, it is not suitable for facial animation that requires high fidelity. The performance-based method captures real human facial movements and uses that data for animation. The physics-based muscle modeling method tries to simulate real human facial muscles to perform the animation. As it is based on human anatomy, it is the closest to realistic human facial animation.

The physics-based muscle approach can be further divided into three categories [3]: mass spring systems, vector muscles and layered spring meshes. The mass spring method [7] uses the spring network to simulate forces applied on face. The layered spring mesh [8] extends it into three connected mesh layers. The vector muscle method simply defines a vector field to act as muscle to attract the mesh vertex and consumes less computation power than the mass spring and layered mass spring systems. However, it only considers the muscle effect on skin and cannot simulate the fatty tissue. Recently, Huang and Yan [9] have presented an NURBS curves based method which separates the human face into five facial units and uses the NURBS curves to control them. It uses fuzzy sets to associate the vertices with the NURBS curves. However, it is very hard to locate a control polygon on the mesh model as the control polygons of the NURBS curves only roughly follow the face mesh.

In this paper, a realistic facial expression animation system is developed by using an NURBS-based vector muscle system. The proposed system can simulate linear muscle, sphincter muscle and the non-linear part of the face like fatty tissue. The system allows more control on linear muscles to simulate a particular expression more realistically by modifying the weights of different control points. As it uses NURBS to simulate muscles, the control points can be put on the surface of face mesh based on the facial anatomy. This makes it easier to locate facial muscles. Through the control points, the curve can be formed under the mesh like the muscle under the face. By changing the weights of the control points, the knots will form a motion vector to control the movement of the mesh vertices within certain region. The number of control points can be determined by the complexity of different parts of face. The basic principles of NURBS curves and Waters' vector muscle are briefly reviewed in Section 2. Section 3 describes the proposed NURBS-based system for facial expression animation. Simulation results are presented in section 4, followed by conclusions in Section 5.

2. Background

2.1 Properties of NURBS

A degree n NURBS curve [10] is defined as,
where $e(u)$ is the knot of the curve that will be used for calculate the muscle motion vector as described in 3.1, $\omega_i$ are the weights, $P_i$ are the control points and $B_{i,n}(u)$ are the blending functions defined as follow,

$$B_{i,n}(u) = \frac{n!}{i!(n-i)!} u^i (1-u)^{n-i}\quad (2)$$

One important property of the NURBS curve is the convex hull property, i.e.

$$\sum_{i=0}^{n} B_{i,n}(u) \omega_i = 1\quad (3)$$

This property ensures that the polynomial can smoothly follows the control points without erratic oscillations. Another important property is endpoint interpolation, which means that the curve always passes through the first and last control points [10, 11]. Therefore, it is easy to form a NURBS muscle.

### 2.2 Waters' Vector Muscle

Waters [12] proposed a physics-based muscle model using the vector method. In his model, muscles are described with vector direction and magnitude. A linear muscle, as illustrated in Fig.1, is defined by two points: the original point $O$ and the insert point $I$. The vector field is $\overrightarrow{OT}$.

![Fig.1. Linear Muscle Model](image)

As shown in Fig. 1, points $I$ and $O$ do not move when the muscle is contracted. The vertex $V$ on the mesh will be influenced by the following rules.

$$V' = V + k \alpha r \times \frac{\overrightarrow{OV}}{||\overrightarrow{OV}||}\quad (4)$$

where $k$ is the muscle spring constant, $\alpha$ is an angular displacement parameter and is defined as follows.

### 3. Proposed NURBS-based Facial Muscle

The NURBS-based muscle can achieve animation by changing the weights of the control points. Fig. 3 shows a NURBS curve. Modification of control point's weight can force the knots of NURBS curve to move, which will then influences the nodes on the mesh. Since NURBS curve can have several control points, it can control the face skin and tissue to move to any desired position.

![Fig.3. A NURBS curve](image)
3.1 NURBS Linear Muscle and NURBS Sphincter Muscle

The control points of the NURBS model are classified into reference control point and current control point. Reference control point is used to relate the knot vector and the node of mesh inside the influenced region. Current control point is the control point whose weight is currently being modified. For the NURBS linear muscle, the current and reference control points are the same as illustrated in Fig. 4.

**Fig. 4. NURBS linear muscle model**

Fig. 5 shows the relationship between the reference point, the knot and the vertex. B is the average of the knots' position before movement and B' is average of the knots' position after movement. \( \overline{BB'} \) is a vector formed by the knot movement and can be obtained by

\[
\overline{BB'} = \left( \frac{\sum_{i=0}^{n} (e_i(u') - e_i(u))}{n+1} \right)
\]

where \( e_i(u) \) is the node before movement, \( e_i(u') \) is the node after the movement. C is the vertex of the mesh which is within the influence region. If C is repositioned to C', it will form a vector \( \overline{CC'} \). We can use vector \( \overline{BB'} \) to find out vector \( \overline{CC'} \) by the following rules,

\[
\angle BAC = \angle B'AC'
\]
\[
\angle ABC = \angle AB'C'
\]

\( \overline{CC'} \) is the vector which will be used to calculate the new position of vertex C. The displacement of the vertex, i.e., \( C'' \), can be calculated by adding a displacement vector,

\[
C'' = C + D \cdot R \cdot K\overline{CC'}
\]

where \( K \) is contraction factor, \( R \) is an angular displacement factor

\[
R = \frac{\cos \beta}{\cos \alpha} - 1
\]

and \( D \) is a radial displacement factor given by

\[
D = 1 - \left( \frac{AC}{d1} \right) \sin \left( \frac{AC}{d1} \frac{\pi}{2} \right)
\]

**Fig. 5. Relationship between control point, knot and vertex**

In our formulation, we use a sine function for radial displacement because the sine function can produce smoother result than the cosine function in a density face mesh as illustrated in Fig. 6.

**Fig. 6. (a) Model using cosine function; (b) model using sine function**

**Fig. 7. NURBS sphincter muscle model**

Since the sphincter muscle does not need the angular displacement, it can be formed by multiplying two radial displacement parameters as shown in Fig. 7.
where $a$ is the length between knot and vertex, $b$ is the length between control point and vertex.

3.2 Other Usage of NUBS Muscle

The sphincter muscle can be used for the deformation of the mouth region. However, it cannot form all phoneme shapes. To control the mouth to form more phoneme shapes, two more NURBS curves have to be added. Fig. 8 illustrates how the NURBS curves can control the mouth shape. The reference control point on the curve is fixed to ensure that the mesh would not form some unexpected mouth shape. The current control can be any one of the control points. With such arrangement, the muscle can control the lip to form a desired phoneme shape.

4. Experimental Result and Discussion

 Twenty-two muscles are implemented based on the facial action coding system (FACS) [13]. FACS use Action Unit (AU) to describe the basic actions of human face. The actions are based on the facial anatomy and each action represents a single muscle. Through different combinations of AU, we can create different facial expressions. Fig. 10 illustrates the mesh with NURBS muscles.

Since human face has fatty tissue under the skin, when we use muscle to control mesh skin we have to consider the effect on fatty tissue. To create a more realistic expression, our system provides a way to simulate the fatty tissue on a face as shown in Fig. 9. This is achieved by adding additional control points between two end control points. The control points will drag the mesh slightly up, simulating the fatty tissue. As the control point is placed on the surface of a face, the curve will follow the shape of the face.

Fig. 8. NURBS muscles with same reference point but different control points

Fig. 9. Fatty tissue simulation

Fig. 10. A set of muscles

Fig. 11. The six basic expressions: (a) happiness, (b) anger, (c) surprise, (d) sadness, (e) fear, (f) disgust
A NURBS facial modeling system is implemented to test the described method. The modeling system is developed by VC++/OpenGL. It runs on a P4 1.9GHz PC with 3D graphics interface. We create a variety of expressions and mouth shapes using NURBS-based linear and sphincter muscles. Each expression simulation is according to the Action Unit of the FACS. Fig. 11 shows the six basic facial expressions, in which only linear muscles are used in the expressions.

The muscles around the mouth are responsible for changing the mouth shapes. The fatty tissues surrounding the cheek and the mouth may make the expression look more realistic especially when animating the model. Fig. 12 shows a more realistic simulation of linear muscle with fatty tissues. Fig. 13 shows two more expressions.

![Fig. 12. (a) happiness, (b) sadness, with fatty tissue modeling](image)

![Fig. 13. (a) bite lip, (b) close mouth](image)

5. Conclusion

This paper presents a novel method for facial expression animation, which uses NURBS curves to create linear muscle, sphincter muscle, and control the mouth shape. Our NURBS-based muscle model can simulate more realistic facial expressions by incorporating fatty tissue simulation. The flexibility of NURBS shape makes the face more controllable. As the control points are attached to the surface of the mesh, the muscles position can be easily located.

Acknowledgment

The project reuses part of code source by Keith Waters (waters@crl.dec.com)

References


