Automatic Face Animation with Linear Model

Jixu Chen *
Rensselaer Polytechnic Institute, Troy, NY

Abstract

This report proposed an automatic face animation method. First, 28 features facial features are automatically extracted from the video-recorded face. Then, using a linear model, we can decompose the variation of the 28 facial features into the shape variation and the expression variation. Finally, the expression variation is used to control the animation of the target face. All the tracking and animation procedure are fully automatic.

Keywords: Face Animation, Linear Model

1 Introduction

3D facial animation remains a fundamental challenge in computer graphics. One of the earliest works on head models for animation is done by Parke in 1974 [Parke 1974]. The model was a mesh of 3D points controlled by a set of parameters. Parke divides the parameters into two groups: 1. Conformation parameters, e.g. the width and the height of face, mouth, eye, describe the variance among different people. 2. Expression parameters are "facial actions" that can be performed on face such as stretching lips or closing eyes.

Because of its efficiency, i.e. face can me modeled with a small set of parameters, the Parke's parameterized facial model is widely used so far. And the model is extended in various ways. Although there are some other methods, such as "key frame", are used in facial animation, we will only focus on the parameterized model in this report.

The idea of our method is still based on Parke's assumption. We also divide the facial parameters into conformation (shape) parameters, which is only related to the identity of the subject; and the expression parameters, which is related to the facial actions. Here, we assume that these two sets of parameters control the face variation through a linear model. Then, the shape and action parameters can be easily estimated by a Least Squares Fitting(LSF) procedure. Finally, the estimated parameters can be directly used to drive a 3D face. Experiment result shows that our method can automatically make the face animation from captured face video.

2 Related work

"There are two basic ideas involved in the creation of parameterized graphical models. The first is the underlying concept of parameterization and the development of appropriate parameter sets. The second is that of graphic image synthesizers that produce images based on some defined parameterization".(cited from [Parke 1982].) In this section I will briefly review the related work based on various parameterization methods.

*e-mail: chenj4@rpi.edu

One kind of face models are based on the face muscles, such as the very successful vector muscle model which was proposed by Waters [Waters 1982]. Figure 1 shows the Water's muscle model. A muscle definition includes the vector field direction, an origin, and an insertion point. In this method, the shape of the skin mesh is influenced by the muscles beneath the skin. This model is anatomically right, and can produce natural result facial movement. But no automatic way of tracking the muscle positions is reported.

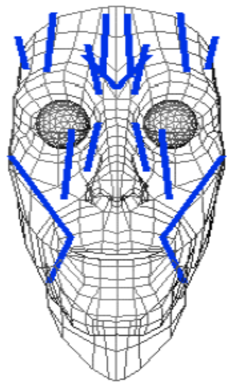


Figure 1: *Waters' linear muscles*

Another kind of very popular facial parameterization methods are based on Facial Action Units(AUs). The concept of Action Units was first described by the Swedish researcher Carl-Herman Hjortsjo in 1969, and later extended to FACS, the Facial Action Coding System ([Parke and Waters 1977]). Based on FACS, the facial behavior is decomposed into 46 action units (AUs), each of which is anatomically related to the contraction of a specific set of facial muscles. Some common AUs are shown in Figure 2

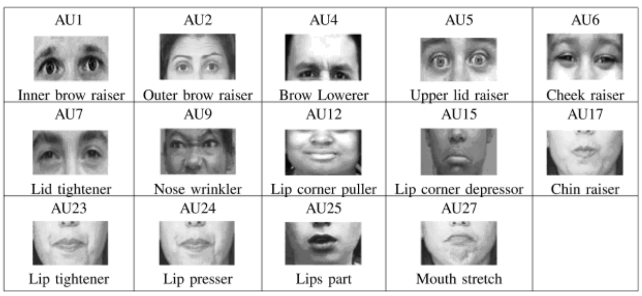


Figure 2: *List of AUs and their interpretations. Note that each AU has a discrete value, which has 5 levels*

FACS are used widely in both computer vision and computer graphics. For facial animation, the MPEG-4 Facial Animation [Pandzic and Forchheimer 2002] is an extension to FACS. It defines a 3D mesh , and the vertices of the 3D mesh are facial points(FPs). Then, 66 low-level Facial Animation Parameters (FAPs) and 2 high-level FAPs (expression, viseme) are defined. Each FAP describes which facial points it acts, in which direction and how much it moves. Most FAPs have their corresponding AUs.

In this report, we use the a simplified 3D face mesh: CANDIDE-3 ([Ahlberg 2001]). This model is defined on some shape and action parameters, each parameter describes the movement of a set of vertices. The final facial animation result is a linear combination of these parameters. Our method is also very similar with the multi-linear face transfer [Vlasic et al. 2005]. The difference is that, in our case, the identity in each sequence is fixed. So, the multi-linear model degenerates to a linear model which only has the expression mode.

3 Facial Feature Extraction

First, we employ the facial feature tracking algorithm in [Tong et al. 2007] to automatically extract feature points from video. This tracker first uses Haar wavelet feature and Adaboost classifier to automatically detect the face and the eye positions. And then the Gabor features are used to detect and track other facial features in realtime. All the features and Adaboost Classifiers are trained from a large face database which includes 500 images containing 200 persons from different races, ages, and facial expressions. The extracted features are shown in Figure 3. There are 10 points for the eyes, 8 points for the mouth, 5 points for the nose and 6 points for the eyebrows. We can see that there are no point on the cheek, because the features

Figure 3: *Automatic Facial Feature Tracking Result*

4 CANDIDE-3 Linear Model

4.1 CANDIDE model

CANDIDE [Ahlberg 2001] is a parameterised face mask specifically developed for model-based coding of human faces. Because its low number of polygons (113 for CANDIDE-3) allows fast reconstruction, it is suitable for realtime animation. (The other most important reason for using CANDIDE-3 is that it is publically available [Ahlberg 2000].)

The CANDIDE-3 model is shown in Figure 4. We also define the correspondences between the CANDIDE model and our tracked points. The 24 corresponding points are shown as circles.

The CANDIDE model can be seen as a $3N$-dimensional vectors $\bar{\mathbf{g}}$ (where $N = 113$ is the number of vertices) containing the (x, y, z) coordinates of the vertices. And the variation of the model is modeled as

$$\mathbf{g}(\sigma, \alpha) = \mathbf{R}s(\bar{\mathbf{g}} + \mathbf{S}\sigma + \mathbf{A}\alpha) + \mathbf{t} \quad (1)$$

where the resulting vector $\mathbf{g}$ contains the new vertex coordinates. $\mathbf{R}$,$\mathbf{t}$, s are the global rotation, translation and scale of the model. The column of $\mathbf{S}$ and $\mathbf{A}$ are the Shape and Action Units, and thus the vectors σ and α contain the shape and action parameters.

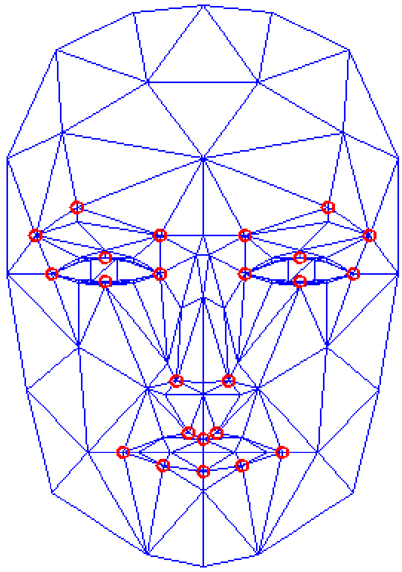


Figure 4: *CANDIDE-3 Model and tracked points*

The shape parameters describe the face shape which is different from person to person. In this project, we used 11 shape parameters. The effect of changing each shape parameter (adding 1 to the parameter value) is shown in Figure 5

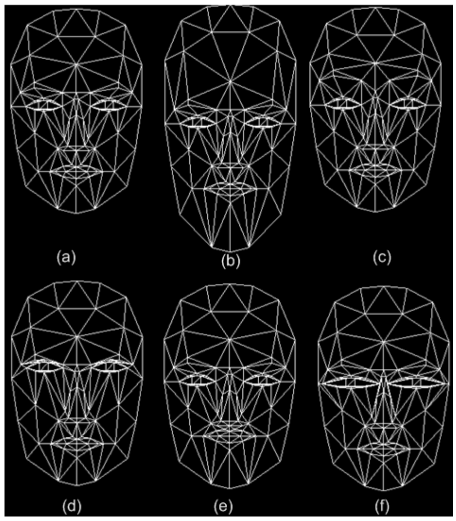


Figure 5: *The effect of changing shape parameters.(5 parameters are showed here.) (a) original CANDIDE-3 Model. (b) Head height. (c) Eyebrows vertical position (d) Eyes vertical position (e) Mouth vertical position (f) Eyes width.*

When the shape parameters are fixed, the action parameters can describe the facial expressions for this specific face. In this project, we used 6 action parameters. The effect of changing each shape parameter (adding 1 to the parameter value) is shown in Figure 6

4.2 Linear Model

Because of the rotation matrix $\mathbf{R}$, the Equation 1 is non-linear. In this project, we only track the frontal face, and we first rotate the face to a straight-up face as shown in Figure 7.

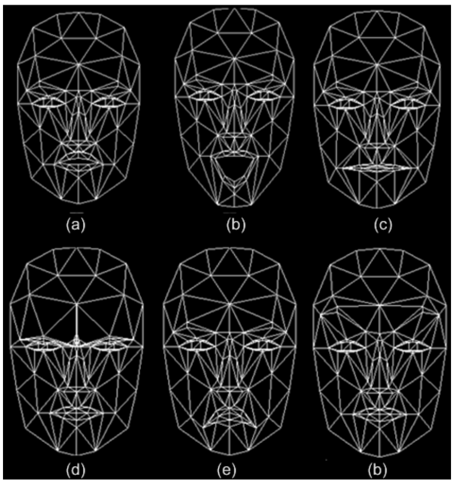
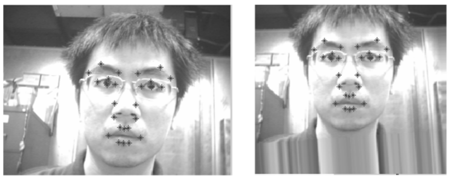


Figure 6: *The effect of changing 6 action parameters. (a) Upper lip raiser. (b) Jaw drop. (c) Lip stretcher (d) Brow lowerer (e) Lip corner depressor (f) Outer brow raiser.*



(a) Original tracked face (b) rotated face

Figure 7: *The tracked face is first rotated based on the eye positions.*

Then, the Equation 1 can be written as as linear model:

$$\mathbf{g}(\sigma, \alpha) = s\bar{\mathbf{g}} + s\mathbf{S}\sigma + s\mathbf{A}\alpha + \mathbf{t} \quad (2)$$

In this report, we only track the frontal face and ignore the z-coordinate. So the equation can be written as Equation 3. Here, $(x_{g1}, y_{g1}, \ldots, x_{gN}, y_{gN})$ are the x,y coordinates of the target face mesh. $(\bar{x}_{g1}, \bar{y}_{g1}, \ldots, \bar{x}_{gN}, \bar{y}_{gN})$ are the coordinates of the original CANDIDE-3 model. $S_{i,1\ldots 2N}, (i = 1\ldots 11)$ and $A_{j,1\ldots 2N}, (j = 1\ldots 6)$ are the shape unit vectors and action unit vectors. In our case, there are 24 facial points are known in both CANDIDE-3 model and the target face model, as shown in figure 4. So we can set up equations using these points. $(s, s\sigma_1, \ldots, s\sigma_1 1, s\alpha_1, \ldots, s\alpha_6, t_x, t_y)$ are the scale, translation, shape and action parameters which we need to solve.

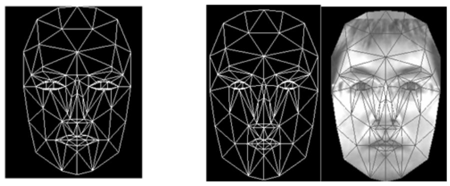
5 Face Animation

In practice, we can solve for Equation 3 in two steps. First we can adjust the shape parameters for each sequence, then fix the shape parameters and just solve for the action parameters for each frame.

5.1 Initialization - solve for shape parameters

To solve for the shape parameters, we first take a neutral face without expressions, as shown in Figure 7. So we can first set the action parameters to zeros, and just solve for the 11 shape parameters. The linear equation 3 has the form of $\mathbf{g} = \mathbf{Mx}$, we can solve for $\mathbf{x}$ conveniently using least square fitting to minimize $\|\mathbf{g} - \mathbf{Mx}\|^2$. The solution is $\mathbf{x} = (\mathbf{M'M})^{-1}\mathbf{M'g}$

Figure 8 shows the result mesh after shape adjustment, based on the tracked neutral face in Figure 7. We can see that, by just adjusting the shape parameters, the mesh can perfectly match to the target face.



(a) (b)

Figure 8: *Shape initialization. (a) Original CANDIDE-3 mesh. (b) The adjusted shape can map to the neutral face.*

5.2 Animation - solve for action parameters

Then, after the shape parameters are solved, we can fixed the shape parameters and use the some least square fitting method to adjust the action parameters for every frame in the sequence.

6 Result

Currently, this animation system works off-line. Given the tracked sequence, we first use one neutral frame from the sequence to adjust the shape parameters, then we apply the shape parameters to whole sequence and adjust the action parameters for each frame.

Finally, we can use the action parameters to animate the neutral face of the subject, or animate another neutral face. (Note that, in order to animate another face. We need to solve anther set of shape parameters based on that face.)

In our experiment, we used a 166 frames sequences to make the animation of two faces. (Jixu's face and James Bond's face). One sample result is shown in Figure 9

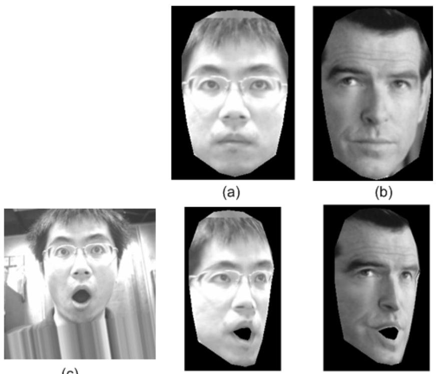


Figure 9: *Face animation result. (a) and (b) are Jixu and Bond's neutral faces. (c) is captured surprise face. (d) and (e) are profile faces of surprised Jixu and Bond.*

7 Conclusion

This report proposes a linear model for face animation. Through least square fitting, the proposed method can easily solve for the shape and action parameters from a tracked face sequence. Then, the action parameters can be used to make the face animation for different faces. This method is simple and efficient. But currently it can only work with frontal face. In the future work, we need to consider the face pose in the linear model.

$$\begin{pmatrix} x_{g1} \\ y_{g1} \\ \vdots \\ x_{gN} \\ y_{gN} \end{pmatrix} = \begin{pmatrix} \bar{x}_{g1} & S_{1,1} & \dots & S_{11,1} & A_{1,1} & \dots & A_{6,1} & 1 & 0 \\ \bar{y}_{g1} & S_{1,2} & \dots & S_{11,2} & A_{1,2} & \dots & A_{6,2} & 0 & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \bar{x}_{gN} & S_{1,2N-1} & \dots & S_{11,2N-1} & A_{1,2N-1} & \dots & A_{6,2N-1} & 1 & 0 \\ \bar{y}_{gN} & S_{1,2N} & \dots & S_{11,2N} & A_{1,2N} & \dots & A_{6,2N} & 0 & 1 \end{pmatrix} \begin{pmatrix} s \\ s\sigma_1 \\ \vdots \\ s\sigma_{11} \\ s\alpha_1 \\ \vdots \\ s\alpha_6 \\ t_x \\ t_y \end{pmatrix} \tag{3}$$

References

AHLBERG, J. 2000. http://www.icg.isy.liu.se/candide/.

AHLBERG, J. 2001. Candide-3 – an updated parameterized face.

PANDZIC, I. S., AND FORCHHEIMER, R. 2002. *MPEG-4 Facial Animation: The Standard, Implementation and Applications*. Wiley.

PARKE, F. I., AND WATERS, K. 1977. *Facial Action Coding System*. Consulting Psychologist Presss.

PARKE, F. I., AND WATERS, K. 1996. *Computer Facial Animation*. A. K. Peters.

PARKE, F. I. 1974. *A parametric model for human faces*. PhD thesis, University of Utah.

PARKE, F. I. 1982. Parameterized models for facial animation. *Computer Graphics* (Nov.).

TONG, Y., WANG, Y., ZHU, Z., AND JI, Q. 2007. Robust facial feature tracking under varying face pose and facial expression. *Pattern Recognition*.

VLASIC, D., BRAND, M., PFISTER, H., AND POPOVIC, J. 2005. Face transfer with multilinear models. 426–433.

WATERS, K. 1982. A muscle model for animation three-dimensional facial expression. *Computer Graphics 21*, 21.