1998 International Computer Symposium
Workshop on Computer Graphics and Virtual Reality
December 17-19, 1998, N.C.K.U., Tainan, Taiwan, R.O.C.

The Design of a Syntax-Directed Facial Animation System

Dong-Hwa Lu[1], *Ming-Shing Su*[2, 3], *and Kuo-Young Cheng*[2, 3]

[1]Computer Service Center, SZE-HAI Institute of Technology and Commerce,
Tu-Cheng City, Taipei County, Taiwan
Email:dhlu@sitc.edu.tw
[2]Department of Computer Science and Information Engineering,
National Taiwan University, Taipei, Taiwan
[3]Institute of Information Science, Academic Sinica, Taipei, Taiwan
Email:{simon, kycheng}@iis.sinica.edu.tw

ABSTRACT

A syntax-directed approach is proposed to avoid the inflexibility problem occurred in most of the existing facial animation systems. Our system contains a region-based face model and provides a developing environment for users to describe the behaviors of action units and the needed facial expressions via writing sentences. The syntax and semantics of the language can be redefined interactively and the kernel of the proposed system is always the same. We demonstrate the feasibility of syntax-directed facial animation system on a window-based personal computer. With the function of texture mapping, our system can generate very realistic facial expressions. Experiment shows that our system is a very flexible and effective tool for the presentation of human facial animations.

1. INTRODUCTION

Developing a facial animation system is a very interesting and important research topic. There are two major problems needed to be considered during the process of generating facial animations: the representation of the face itself and the modeling of motion [21]. As an object to be animated, the face is more complex than what has been animated to date. It is composed of flexible and consistently organized regions whose reactions to change vary depending not only on the particular face being animated, but also what actions the face has already performed. This is an interdisciplinary project and a team is often needed to work it out. So far, most of the facial animation systems have been implemented on graphic-accelerated workstations or mainframes.

Since 1978, the Facial Action Coding System (FACS) [8] developed by psychologists P. Ekman and W. Friesen has been irreplaceable in the research community of facial animation. In FACS, facial expressions are represented in terms of physically controllable actions, the action units (AUs). Each action unit describes the actions about a group of muscles underneath the skin. In facial animation systems, using the concepts of FACS as a human-machine interface is an important milestone.

There are many research approaches in computer-generated facial animations. Among them, the main techniques used include parameterized model [7, 10, 20, 22, 24, 25, 26, 27], region-based model [2, 28, 29], and physics-based model [11, 31, 32, 33, 34, 35, 36, 37]. Some facial animation systems include texture-mapping function [3, 12, 39, 40] to enhance the fidelity of the generated facial animations.

Although these systems can generate good facial animations, they all have the inflexibility problem. The inflexibility problem occurs since the face model and the description of each graphic entity are implemented via almost fixed parameters and/or procedures, various actions are generated simply by changing parameter values. When the original definition of face model, parameter set, or action unit is not proper, the problem occurs since there is no easy way to modify or reuse these procedures and parameters. Besides, most of the existing facial animation systems provide fixed graphic entities for user to present expressions. The behavior of each graphic entity is almost fixed. These behaviors cannot be modified without rewriting corresponding procedures and this is a time-consuming task. Furthermore, when the designer decides to use different presentation technique in the facial animation system, it often needs to rewrite the whole animation system if possible.

Based upon our previous research experiences [4, 5, 6, 13, 14, 15, 16, 17, 18, 19, 38], we find that to improve the inflexibility problem requires the use of computer graphics techniques and high-level structural description of graphic entities. Therefore, we propose a syntax-directed facial animation system, which contains a region-based face model and provides an environment for users to describe the behaviors of action units and the needed facial expressions via writing sentences. The syntax and semantics of the language can be redefined interactively if the result is not satisfied and the kernel of the proposed system is always the same.

This paper presents the design methodology of the syntax-directed facial animation system. Section 2 describes the design concepts of the syntax-directed facial animation system. Section 3 shows the implementation results and Section 4 draws some conclusions.

2. THE SYNTAX-DIRECTED FACIAL ANIMATION SYSTEM

The system architecture of the syntax-directed facial animation system is shown in Figure 1. The users of this system can be divided into two categories: designer and general user. The designer construct abstract face model from one instance of three-dimensional face data via windowing user interface. The designer first interactively divides the three-dimensional face data into regions according to anatomy structure of human face in as much as the same way as the formalized specification of FACS proposed by Platt [29].

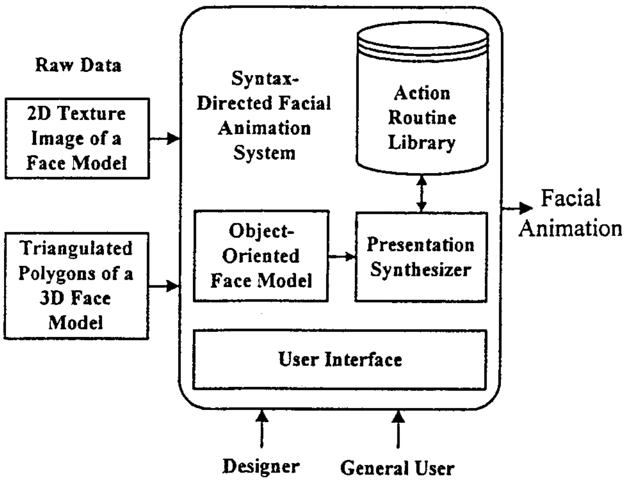

Figure 1. System architecture of the facial animation system.

Once the abstract face model has been constructed, an adapted physical face model can be obtained. The adapted physical face model contains the physical three-dimensional face image data as well as the position of each shape control point. The designer interactively uses an editor to specify exerting point and magnitudes of force that can cause appearance changes in corresponding face region and meet the requirements of AU.

Since FACS is for a human observer to analyze facial expressions of a mattered subject, the observer must have the background knowledge about the muscular basis of facial movements. It is very difficult to express the background knowledge via computer graphics techniques. Therefore, we use syntax-directed approach to provide a windowing user interface for the designer to define a language to describe the behaviors of action units and the texture mapping process.

The designer describes the behavior of each action unit via sentences, which conform to the predefined syntax and semantics rules. If the semantics cannot be expressed by way of existed action routines, the designer can rewrite his own action routines and specify them in the semantic descriptions via filling proper entries in the semantic table of the corresponding grammar rules.

Similarly, when the predefined grammar rules are not proper for the needed face presentation, the designer only needs to add some grammar rules or rewrite the whole grammar rules and the corresponding semantics. The language transducer generator will generate the new scan table and parsing table which cooperate with the corresponding drivers will become a new language transducer. After this, designer can use new sentential forms to describe the behavior of action units and texture mapping.

The general user must learn the concept of FACS before using our system. He/she can decompose the needed facial expression into a combination of action units and their respective intensities. Then, the general user can use sentences to state the combination.

The kernel of our facial animation system is a syntax-directed presentation synthesizer as shown in Figure 2. The syntax-directed presentation synthesizer consists of three major processes: a language transducer generator, a language transducer, and a windowing user interface. The windowing user interface provides both designer and general user a working place to state the necessary information for the face presentation. The language transducer generator can adapt to different vocabularies and presentation techniques. The grammar that accepted by the language transducer generator is a kind of L-attributed attribute grammar [9].

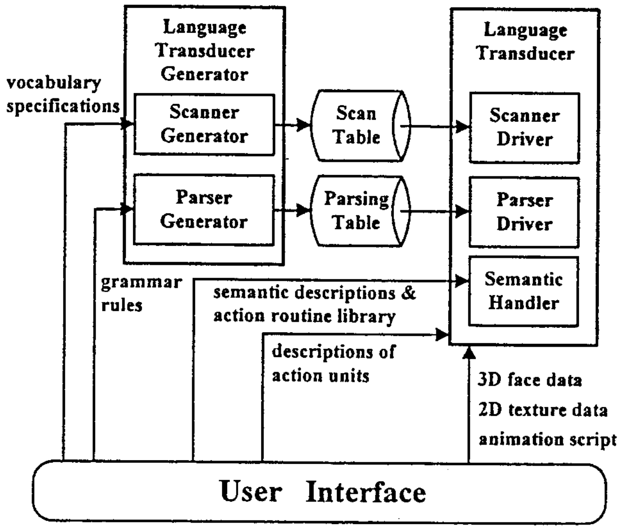

Figure 2. System architecture of the syntax-directed presentation synthesizer.

The designer determines the vocabulary specification for the description of FACS concepts and texture mapping. Then he/she defines grammar rules that will be used for syntax-directed translation of FACS concepts and texture mapping. In each grammar rule, the designer must determine the semantic routines to control the data flow. The designer can select needed action routines from action routine library or write action routines by himself/herself. The general user uses sentences to describe the needed facial expression in a manner like FACS.

The language transducer generator consists of a scanner generator and a LL(1) parser generator. The designer tells the scanner generator what tokens he/she wants recognized; producing a scanner that meets his/her specification is the generator's responsibility. Our scanner generator does not produce an entire scanner; rather, it produces tables that can be used with a standard driver program. The combination of generated tables and standard driver yields the desired custom scanner.

The language transducer consists of a scanner driver, a parser driver, and a semantic handler. No matter the designer defines different languages the presentation synthesizer is always the same. When the designer select new languages, only vocabulary specifications, grammar rules, and semantic routines are changed. The presentation synthesizer can generate new scan table, parsing table, and semantic table. These tables combine with the fixed scanner driver, parser driver, and semantic handler yield a new language transducer.

3. EXPERIMENTAL RESULTS

We implement the syntax-directed facial animation system on a personal computer equipped with Microsoft™ Windows 95 operating system. We use Microsoft™ Visual C++ and Borland™ C++ Builder as the programming environments. The rendering process and texture mapping are implemented by calling proper OpenGL graphic library routines [1, 23]. The kernel of the syntax-directed face presentation is implemented via C++ programming language [30]. To obtain real-time animation, a personal computer with Pentium 233 MMX CPU and 64MB SDRAM is used.

The generic face model in the proposed system contains 38 regions, 260 shape control points, 3712 polygons, and 1986 vertices. The surface of the generic face consists of thirty regions. The eyes are divided into six regions and the teeth are split into upper teeth region and lower teeth region.

The specification of grammar rules, which are used to parse the facial expression sentences, is listed as follows:

```
#Terminal
end
ClearAllFrame
KeepAllFrame
BeginKeyFrame
Animate
IntLit
RealLit
StringLit
ID
(
,
)
#Production
<START>      ::= <CLEAR> <ACTIONS> end
<ACTIONS>    ::= <ACTION> <ACTIONS>
<ACTIONS>    ::=
<ACTION>     ::= <KEY> <AULIST>
<ACTION>     ::= Animate <STEPS> <SHADING>
<AULIST>     ::= <AUHEAD> <AULIST>
<AULIST>     ::=
<AUHEAD>     ::= <AUID> <STRENGTH>
<AUID>       ::= IntLit
<STRENGTH>   ::= RealLit
<STEPS>      ::= IntLit
<CLEAR>      ::= ClearAllFrame
<CLEAR>      ::= KeepAllFrame
<KEY>        ::= BeginKeyFrame
<SHADING>    ::= StringLit
#End
```

The sentences to achieve the effect of AU 1 (Inner Brow Raiser) are written as follows:

```
AU 1
Above_Brow         6    0.00   10.00   0.00
Above_Brow         7    0.00   10.00   0.00
Brow               3    0.00   10.00   0.00
Brow               4    0.00   10.00   0.00
Brow               5    0.00   20.00   0.00
Brow               6    0.00   20.00   0.00
Brow               7    0.00   10.00   0.00
Brow               8    0.00   10.00   0.00
Below_Lower_Lid    0    0.00    5.00   0.00
Below_Lower_Lid    3    2.00    8.00   0.00
Below_Lower_Lid    5    1.00    7.00   0.00
Below_Lower_Lid    10  -1.00    7.00   0.00
Below_Lower_Lid    12  -2.00    8.00   0.00
Below_Lower_Lid    15   0.00    5.00   0.00
true
Animate 4 "SmoothShading"
Animate 4 "Texture"
end
```

Figure 3 shows the (a) initial frame, (b) final frame, (c) initial frame with texture, and (d) final frame with texture of AU 1.

The user can write sentences to describe the needed facial expression in a manner like FACS. The vocabulary used is based upon the description of action units defined by designer. By combining AU 2, 4, 5, 7, 15, 17, 20, 23, 26, and 42 with properly chosen intensities, we can obtain the facial expression anger. Figure 5 shows the animation sequence of expression anger and Figure 4 shows the same expression with texture. The sentences used are as follows:

```
ClearAllFrame
BeginKeyFrame
BeginKeyFrame
      2    1.0
      4    2.0
      5    1.2
      7    1.0
      15   1.2
      17   1.2
      20   0.8
      23   0.3
      26   0.4
      42   1.0
Animate 2 "SmoothShading"
Animate 2 "Texture"
end
```

Figure 6 to Figure 10 show the animation sequences with texture of expression disgust, fear, happiness, sadness, and surprise, respectively.

4. CONCLUSION

In this paper, we propose a syntax-directed facial animation system that contains two major components: an object-oriented system model and a presentation synthesizer, both based on the Facial Action Coding System (FACS). Some examples have been given to illustrate the design concepts of the proposed system.

Experiment shows that the syntax-directed facial animation system is a very flexible and effective tool for the design of human facial animations. With the syntax-directed kernel, we can improve the inflexibility problem occurred in most of the existing facial animation systems. The designer can express the behaviors of each action unit by defining proper grammar rules, action routines, and/or writing sentences. The general user can express the needed facial expression by specifying a combination of action units and their corresponding intensities. During these description processes, the kernel of our system is always the same.

5. REFERENCES

[1] Angel, E., Interactive Computer Graphics: A Top-Down Approach with OpenGL, Addison-Wesley, Reading, Massachusetts, 1997.

[2] Badler, N. I., "Human Body Models and Animation," IEEE Computer Graphics and Applications, Vol. 2, pp. 6-7, 1982.

[3] Chen, C. H., Rule-Based Image Warping Technique, Master Thesis, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, 1995.

[4] Cheng, K. Y., C. H. Chen, and D. H. Lu, "Rule-Based Image Warping Technique," Proceedings of 1995 Computer Graphics Workshop, Taipei, Taiwan, pp. 58-60, 1995.

[5] Cheng, K. Y., C. H. Hsu, and D. H. Lu, "A Visual Programming Tool for the Design of Syntax-Directed Pattern Description Languages," Proceedings of 1995 Computer Graphics Workshop, Taipei, Taiwan, pp. 50-53, 1995.

[6] Cheng, K. Y., S. N. Huang, and D. H. Lu, "Design of a Software Tool for Vocal Speech to Picture Synchronization," Proceedings of 1995 Computer Graphics Workshop, Taipei, Taiwan, pp. 38-40, 1995.

[7] DiPaola, S., "Extending the Range of Facial Types," The Journal of Visualization and Computer Animation, Vol. 2, pp. 129-131, 1991.

[8] Ekman, P. and W. V. Friesen, Manual for the Facial Action Coding System, Consulting Psychologists Press, Palo Alto, California, 1978.

[9] Fischer, C. and R. J. LeBlanc, Crafting a Compiler with C, The Benjamin/ Cummings, Redwood City, California, 1991.

[10] Kalra, P., A. Mangili, N. Magnenat-Thalmann, and D. Thalmann, "SMILE: A Multilayered Facial Animation System," SIGGRAPH'92 Course Notes #17: Advanced Techniques in Human Modeling, Animation, and Rendering, Chicago, Illinois, pp. 68-75, 1992.

[11] Lee, Y. C., D. Terzopoulos, and K. Waters, "Constructing Physics-Based Facial Models of Individuals," SIGGRAPH'93 Course Notes #80, Recent Techniques in Human Modeling, Animation, and Rendering, Anaheim, California, pp. 3-18~3-25, 1993.

[12] Lou, C. C., Texture Mapping for Facial Animation, Master Thesis, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, 1993

[13] Lu, D. H., Notes on Visual and Iconic Languages, Master Thesis, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, 1988.

[14] Lu, D. H., The Design of a Syntax-Directed Face Presentation System, Ph.D. Dissertation, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, 1998.

[15] Lu, D. H., H. L. Cheng, and K. Y. Cheng, "An Interactive Tool for the Construction of Physically Based Face Models," 1994 IPPR Conference on Computer Vision, Graphics and Image Processing, SunLinkSea, Nan-Tou County, Taiwan, pp. 377-381, 1994.

[16] Lu, D. H. and K. Y. Cheng, "The Design of an Object-Oriented Face Model," Proceedings of Pacific Graphics'94/CADDM'94, Beijing, China, Vol. 1, pp. 158-162, 1994.

[17] Lu, D. H. and K. Y. Cheng, "Design of a Syntax-Directed Face Presentation System," Images and Recognition, Taipei, Taiwan, Vol. 4, pp. 1-14, 1997.

[18] Lu, D. H., S. N. Huang, and K. Y. Cheng, "Design of a Software Tool for Vocal Tract Simulation Systems," Journal of Information Science and Engineering, Vol. 13, pp. 155-169, 1997.

[19] Lu, D. H., M. S. Su, and K. Y. Cheng, "The Design of a FACS-based Facial Animation System," Proceedings of the National Science Council, Part A: Physical Science and Engineering, Vol. 22, pp. 400-409, 1998.

[20] Magnenat-Thalmann, N., E. Primeau, and D. Thalmann, "Abstract Muscle Action Procedures for Human Face Animation," The Visual Computer, Vol. 3, pp. 290-297, 1988.

[21] Magnenat-Thalmann, N., "The Problematics of Facial Animation," in: N. Magnenat-Thalmann and D. Thalmann, State-of-the-art in Computer Animation, Springer, Tokyo, pp. 47-55, 1989.

[22] Magnenat-Thalmann, N., H. T. Minh, M. de Angelis, and D. Thalmann, "Design, Transformation and Animation of Human Faces," The Visual Computer, Vol. 5, pp. 32-39, 1989.

[23] Neider, J., T. Davis, and M. Woo, OpenGL Programming Guide, Release 1, Addison-Wesley, Reading, Massachusetts, 1993.

[24] Paouri, A., N. Magnenat-Thalmann, and D. Thalmann, "Creating Realistic Three-Dimensional Human Shape Characters for Computer-Generated Films," in: D Thalmann, SIGGRAPH'92 Course Notes #17: Advanced Techniques in Human Modeling, Animation, and Rendering, Chicago, Illinois, pp. 19-25, 1992.

[25] Parke, F. I., "A Model for Human Faces that Allows Speech Synchronized Animation," Journal of Computers & Graphics, Vol. 1, pp. 3-4, 1975.

[26] Parke, F. I., "Parametrized Models for Facial Animation," IEEE Computer Graphics and Applications, Vol. 2, pp. 61-68, 1982.

[27] Parke, F. I., "Parametrized Models for Facial Animation -- Revisited," SIGGRAPH'89 Course Notes #22: State of the Art in Facial Animation, Boston, Massachusetts, pp. 43-56, 1989.

[28] Platt, S. M. and N. I. Badler, "Animating Facial Expressions," Computer Graphics, Vol. 15, pp. 245-252, 1981.

[29] Platt, S. M., A Structural Model of the Human Face, Ph.D. Dissertation, Dept. of Computer and Information Science, University of Pennsylvania, Philadelphia, USA, 1985.

[30] Stroustrup, B., The C++ Programming Language, 2nd Ed., Addison-Wesley, Reading, Massachusetts, 1991.
[31] Terzopoulos, D. and K. Waters, "Physically-based Facial Modeling, Analysis, and Animation," Journal of Visualization and Computer Animation, Vol. 1, pp. 73-80, 1990.
[32] Terzopoulos, D. and K. Waters, "Sampling and Reconstruction with Adaptive Meshes," Proceedings of Computer Vision and Pattern Recognition Conference, Maui, Hawaii, pp. 70-75, 1991.
[33] Terzopoulos, D. and K. Waters, "Techniques for Realistic Facial Modeling and Animation," in: N. Magnenat Thalmann and D. Thalmann, Computer Animation'91, Springer, Tokyo, pp. 59-74, 1991.
[34] Waters, K., "A Muscle Model for Animating Three-Dimensional Facial Expression," Computer Graphics, Vol. 21, pp. 17-24, 1987.
[35] Waters, K., "Modeling three-dimensional Facial Expressions: Tutorial Note," SIGGRAPH'89 Course Notes #22: State of the Art in Facial Animation, Boston, Massachusetts, pp. 127-160, 1989.
[36] Waters, K. and D. Terzopoulos, "A Physical Model of Facial Tissue and Muscle Articulation," Proceedings of First Conference on Visualization in Biomedical Computing, Atlanta, Georgia, pp. 77-82, 1990.
[37] Waters, K. and D. Terzopoulos, "Modeling and Animating Faces using Scanned Data," The Journal of Visualization and Computer Animation, Vol. 2, pp. 123-128, 1991.
[38] Weng, W. K., D. H. Lu, and K. Y. Cheng, "A FACS-Oriented Facial Animation System," Proceedings of 1st Workshop on Computer Graphics, Nankang, Taipei, Taiwan, pp. 109-111, 1993.
[39] Williams, L., "Performance-Driven Facial Animation," Computer Graphics, Vol. 24, pp. 235-242, 1990.
[40] Yau, J. F. S. and N. D. Duffy, "3-D Facial Animation Using Image Samples," in N. Magnenat-Thalmann and D. Thalmann, New Trends in Computer Graphics, Springer, Heidelberg, pp. 64-73, 1988.

(a) (b) (c) (d)

Figure 3. The animation frames of AU 1.

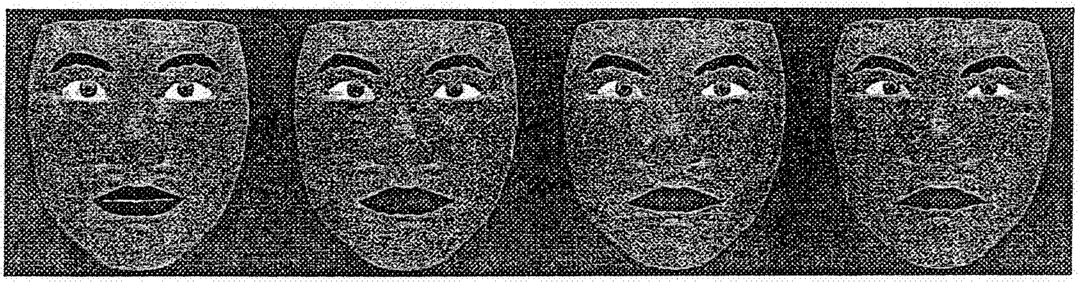

Figure 4. Animation sequence of expression Anger (without texture).

Figure 5. Animation sequence of expression Anger (with texture).