JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 18, 459-466 (2002)

Short Paper

A Fuzzy-based Approach to Mesh Simplification*

CHIN-CHEN CHANG+, SHU-KAI YANG, DING-ZHOU DUAN
AND MING-FEN LIN
+*Computer and Communications Research Laboratories*
Industrial Technology Research Institute
Hsinchu, 310 Taiwan
E-mail: chinchen@itri.org.tw
Opto-Electronics and System Laboratories
Industrial Technology Research Institute
Hsinchu, 310 Taiwan

In this paper, a novel method, called the fuzzy-based approach, for mesh simplification is presented. We first propose the fuzzy position uncertainty function and the fuzzy curvature uncertainty function for, respectively, measuring the variation of the surface position and the variation of the surface curvature while meshes are simplified. We then utilize the TSK fuzzy inference model, which integrates and balances the fuzzy position uncertainty and the fuzzy curvature uncertainty, to determine the cost as a criterion for removing a portion of a mesh in the simplification process. Experimental results show that our approach can produce good approximations that preserve the features of a model.

Keywords: multi-resolution modeling, progressive mesh, mesh simplification, fuzzy set, fuzzy inference model

1. INTRODUCTION

The mesh simplification technique provides a way to generate simpler versions of a model. Naturally, we would like to take a highly detailed polygonal model as input and automatically generate a good low-polygon approximation of the original as output. To achieve this goal, mesh simplification is necessary.

Mesh simplification has been intensively researched in recent years. Hoppe et al. [5, 6] proposed the edge collapse technique for forming progressive meshes and presented an approach to minimizing an *energy function* that measures the competing desires of conciseness of representation and fidelity to the data. Algorri and Schmitt [1] presented an approach that evaluates the length of an edge as the cost and selecting the shortest edge for edge collapsing at each step. Garland and Heckbert [3] presented quadric error metrics for surface simplification to maintain surface error approximations and used iterative contractions of vertex pairs to simplify models. Schroeder et al. [7]

Received December 18, 2000; revised April 25 & September 5, 2001; accepted October 8, 2001.
Communicated by Yung-Nien Sun.

*A preliminary version of this paper has been presented at International Workshop on Advanced Technology 2001, February, Taejon, Korea, 2001.

were first to propose an approach that iteratively selects a vertex for removal, removes all adjacent faces, and re-triangulates the resulting hole.

The key to producing good approximations is to produce a simplification that will cause the least variation of the model. If variations of surface attributes are poorly measured and integrated, the features of the model, such as the legs in the spider model, may disappear at high-polygon approximations. This will lead to the destruction of features at low-polygon approximations or restriction of simplifying models further. In this paper, we propose a fuzzy inference model for simplifying meshes since 1) fuzzy set theory [4, 8-10] can model non-statistical imprecision and 2) the fuzzy inference model can integrate the concepts of fuzzy sets, fuzzy if-then rules and fuzzy reasoning.

The rest of this paper is organized as follows. In section 2, we briefly describe the TSK fuzzy inference model. In section 3, we present the fuzzy-based approach to mesh simplification. Results are given in section 4. Finally, conclusions are drawn in section 5.

2. FUZZY INFERENCE MODEL

A fuzzy set *A* is usually defined as

$$\mu_A(x): X \rightarrow [0,1], \tag{1}$$

where μ_A is the membership function indicating the membership grade of an element *x* in the universal set *X*.

The TSK fuzzy inference model, as proposed by Takagi, Sugeno and Kang (TSK) [8, 9], uses a typical fuzzy if-then rule in the following form:

$$\text{if } x \text{ is } A \text{ and } y \text{ is } B \text{ then } z = f(x, y), \tag{2}$$

where *x* and *y* are input variables; *A* and *B* are fuzzy sets in the antecedent part; and *z* is the output, which is a function of the input variables in the consequent part. Generally, $f(x, y)$ is a polynomial function of input variables *x* and *y*. If multiple fuzzy if-then rules are considered, then the final output is taken as the weighted average of each rule's output. The weight of each fuzzy if-then rule is computed based on the minimum of the membership grades of the input variables in the antecedent part.

Fig. 1 depicts an example of a two-rule two-input fuzzy inference model. The fuzzy if-then rules are defined as follows:

Rule 1: if x is A_1 and y is B_1 then z_1 is $f_1(x, y)$;

Rule 2: if x is A_2 and y is B_2 then z_2 is $f_2(x, y)$.

Here x and y are two input variables, and A_1, B_1, A_2 and B_2 are fuzzy sets. z_1 and z_2 are outputs of the two fuzzy if-then rules, respectively. $f_1(x, y)$ and $f_2(x, y)$ are functions of input variables x and y. The weights w_1 and w_2 of the two fuzzy if-then rules are calculated based on the minimum of the membership grades of input variables x and y in each antecedent part, respectively. The overall output z is taken as the weighted average of each rule's output.

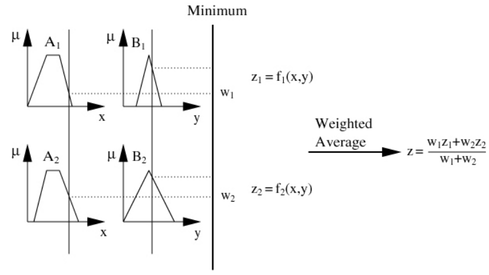


Fig. 1. An example of a TSK fuzzy inference model with two rules and two inputs.

3. MESH SIMPLIFICATION USING FUZZY-BASED APPROACHES

Without loss of generality, we assume that the polygonal models consist of triangles only. Basically, our approach is based on edge collapsing [5, 6]. Note that the proposed approach can also be applied to other mesh simplifications algorithms that make local, complexity-reducing simplifications, such as vertex decimation [7]. The basic idea of edge collapsing is to simplify models through repeated use of the simple edge collapse operation, depicted in Fig. 2. In this operation, two vertices v_s and v_t (the edge v_sv_t) are selected, and v_s is collapsed onto v_t. In each step, the method first evaluates the cost of each edge for collapsing and then selects the minimal cost edge for collapsing.

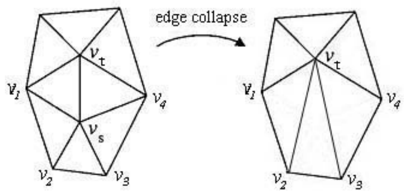


Fig. 2. Edge collapse operation.

We first use a fuzzy position uncertainty function and a fuzzy curvature uncertainty function for, respectively, measuring the variation of the surface position and the variance of the surface curvature [2] of an edge before and after the edge collapse operation is performed. We then utilize the TSK fuzzy inference model, which integrates and balances the fuzzy position uncertainty and the fuzzy curvature uncertainty, to determine the cost of collapsing the edge.

We measure the variation of the surface position while meshes are simplified by calculating the maximal value between the absolute values of the length of an edge adjacent to v_s before and after the edge collapse operation is performed. This maximal value is called the fuzzy position uncertainty of the edge v_sv_t when meshes are simplified. The fuzzy position uncertainty function is defined as

$$\mu_{pos}(v_sv_t) = \max_{v_j \in V(v_s)} \left\{ \left| \left\| v_s \cdot_{position} - v_j \cdot_{position} \right\| - \left\| v_t \cdot_{position} - v_j \cdot_{position} \right\| \right| \right\}, \quad (3)$$

where $V(v_s)$ of a vertex v_s is a set of vertices adjacent to v_s not including v_s itself. For example, see Fig. 2: here, the $V(v_s)$ contains the vertices v_1, v_2, v_3, v_4 and v_t. For simplicity, all computed fuzzy position uncertainties are normalized to [0, 1] in the edge collapse algorithm in each step.

Let the vertices adjacent to v_s be listed as a *vertex chain* C $[v_t, v_1, v_2, \ldots, v_n]$ in counterclockwise order (n > 1); see Fig. 2, which illustrates the case where $n = 4$. We evaluate the variation of the surface curvature while models are simplified by computing the absolute value of the difference between the minimal values of dot products of normal vectors of triangles adjacent to v_s before and after the collapse operation is performed. This absolute value is called the fuzzy curvature uncertainty of the edge v_sv_t when meshes are simplified. The fuzzy curvature uncertainty function is defined as

$$\mu_{cur}(v_sv_t) = \left| \min_{f_i, f_j \in T(v_s)} \{ f_i \cdot_{normal} \cdot f_j \cdot_{normal} \} - \min_{f_i, f_j \in T(C)} \{ f_i \cdot_{normal} \cdot f_j \cdot_{normal} \} \right|, \tag{4}$$

where the set $T(v_s)$ of v_s is a set of triangles adjacent to v_s and the set $T(C)$ of C is a set of triangles formed by $\Delta\, v_tv_1v_2$, $\Delta\, v_tv_2v_3, \ldots$, and $\Delta\, v_tv_{n-1}v_n$. An example is shown in Fig. 2. The set $T(v_s)$ contains a set of triangles $\Delta\, v_sv_1v_2$, $\Delta\, v_sv_2v_3$, $\Delta\, v_sv_3v_4$, $\Delta\, v_sv_4v_t$ and $\Delta\, v_sv_tv_1$. The set $T(C)$ consists of a set of triangles $\Delta\, v_tv_1v_2$, $\Delta\, v_tv_2v_3$ and $\Delta\, v_tv_3v_4$. For simplicity, all computed fuzzy curvature uncertainties are normalized to [0, 1] in the edge collapse algorithm in each step.

Let *LARGE* and *SMALL* be two fuzzy sets on [0, 1] used to characterize the fuzzy position uncertainty, and let *ROUGH* and *SMOOTH* be two fuzzy sets on [0, 1] used to characterize the fuzzy curvature uncertainty. For fuzzy-based approaches to mesh simplification, the TSK fuzzy if-then rules are defined as follows:

Rule 1: if x is *LARGE* and y is *ROUGH* then z_1 is $x^{p_1}y^{q_1}$;

Rule 2: if x is *LARGE* and y is *SMOOTH* then z_2 is $x^{p_2}y^{q_2}$;

Rule 3: if x is *SMALL* and y is *ROUGH* then z_3 is $x^{p_3}y^{q_3}$;

Rule 4: if x is *SMALL* and y is *SMOOTH* then z_4 is $x^{p_4}y^{q_4}$.

Here x is the fuzzy position uncertainty and y is the fuzzy curvature uncertainty. The functions z_1, z_2, z_3 and z_4 are outputs of the four fuzzy if-then rules, respectively, representing the cost of collapsing an edge in each step. p_i and q_j are parameters for determining the cost of collapsing an edge.

The weights w_1, w_2, w_3 and w_4 of the four fuzzy if-then rules are calculated based on the minimum of the membership grades of input variables x and y in each antecedent part, respectively. The overall system output z used as the cost of collapsing an edge is computed using the weighted average of each rule's output as follows:

$$z = \frac{w_1z_1 + w_2z_2 + w_3z_3 + w_4z_4}{w_1 + w_2 + w_3 + w_4}. \tag{5}$$

Thereafter, it is compared with other outputs of other edges for carrying out edge collapsing in each step. That is, the algorithm sorts all edges according to their costs and selects the edge with the minimal collapsing cost in each step.

4. RESULTS

Our experimental platform is a PC with an Intel® Pentium® III 733MHz CPU and 256 Mbytes of memory running the Microsoft® Windows 2000 Professional OS.

The *S*-function [4] is a valuable tool for defining a fuzzy set which encodes words like "tall" and "large". We apply the *S*-function to define the fuzzy sets *LARGE*, *SMALL*, *ROUGH* and *SMOOTH*. The membership functions of *LARGE* and *ROUGH* are defined as $S(x; 0,1)$, and the membership functions of *SMALL* and *SMOOTH* are defined as $1-S(x; 0, 1)$, respectively. The functions $S(x; 0, 1)$ and $1-S(x; 0, 1)$ are shown in Fig. 3.

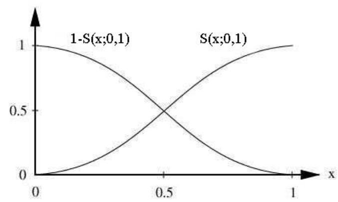


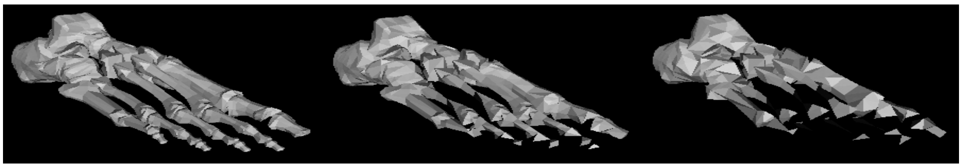
Fig. 3. The functions $S(x;0,1)$ and $1-S(x;0,1)$.

It is difficult to determine the best parameters p_i and q_j. Usually, they can be obtained by using heuristics as long as they can appropriately describe the output result. Since x (the fuzzy position uncertainty) and y (the fuzzy curvature uncertainty) are normalized to [0, 1], the smaller the values of p_i and q_j, the higher the cost of edge collapsing; and the larger the values of p_i and q_j, the lower the cost of edge collapsing. The parameters p_1 *and* q_1 are, respectively, assigned to be 0.2 and 0.2, indicating that the cost of edge collapsing is high; the parameters p_2 *and* q_2 are, respectively, assigned to be 2 and 2, indicating that the cost of edge collapsing is slightly less; the parameters p_3 *and* q_3 are, respectively, assigned to be 0.5 and 0.5, indicating that the cost of edge collapsing is slightly higher; the parameters p_4 *and* q_4 are, respectively, assigned to be 5 and 5, indicating that the cost of edge collapsing is low.

We apply our approaches to the original porsche model with 10474 polygons. The calculations take about 3.87 seconds. Figs. 4(a)-4(d) show the resulting images of a sample sequence of approximations generated by our approach for the porsche model. Fig. 4(a) shows the original porsche model, and Figs. 4(b)-4(d) show the simplifications with 5226, 2594 and 1286 polygons, respectively. Fig. 4(b) shows one simplification (50%) of the original porsche model. In this approximation, the major details of the tires and the antenna have been preserved. Fig. 4(c) is another approximation (25%) of the original porsche model. In this case, the details of the tires have been preserved, but the details of the antenna have disappeared. Fig. 4(d) is another simplification (12%) of the original porsche model. In this simplification, the details of the tires and the antenna of the porsche model have disappeared, but the basic structure of the porsche model is still preserved. We also test the foot bone model to compare the visual quality obtained using our approach with that obtained using the previous method, evaluating the length of an edge as the cost and selecting the shortest edge for collapsing at each step when meshes are simplified. Fig. 5(a) shows the original foot bones model with 4204 polygons. Figs. 5(b) and 5(c) show the resulting simplifications with 976 polygons generated using our approach and the previous method, respectively. In this case, using our approach, features such as the bone segments are preserved better than they are using the previous method.

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

Fig. 4(a). The original porsche model has 10474 polygons. Figs. 4(b)-4(d). The approximations have 5226, 2594, and 1286 polygons, respectively.



(a) (b) (c)

Fig. 5(a). The original foot bone model has 4204 polygons. Figs. 5(b) and 5(c). The approximations with 976 polygons produced by our approach and the previous method, respectively.

5. CONCLUSIONS

A fuzzy-based mesh simplification approach has been studied in this work. With the proposed approach, we can produce good approximations that can preserve the features of the original model. The proposed approach is not limited to two surface attributes and can be extended to m ($m > 2$) surface attributes, such as surface color [2]. Also, the characterization of the variation of each attribute while models are simplified is not limited to two fuzzy sets and can be extended to k ($k > 2$) fuzzy sets.

ACKNOWLEDGEMENTS

This work is a partial result of project number 903XS18F4, conducted by ITRI under the sponsorship of the Ministry of Economic Affairs, R.O.C. The authors also thank the referees for many constructive suggestions for revision of this paper.

REFERENCES

1. M. M. Algorri and F. Schmitt, "Mesh simplification," in *Proceedings of EUROGRAPHICS'96*, 1996, pp. 77-86.
2. J. Cohen, M. Olano and D. Manocha, "Appearance-preserving simplification," in *Proceedings of SIGGRAPH'98*, 1998, pp. 115-122.
3. M. Garland and P. S. Heckbert, "Surface simplification using quadric error metrics," in *Proceedings of SIGGRAPH'97*, 1997, pp. 209-224.
4. J. Giarratano and G. Riley, *Expert Systems: Principles and Programming*. PWS-KENT Publishing Company, 1989.
5. H. Hoppe, "Progressive meshes," in *Proceedings of SIGGRAPH'96*, 1996, pp. 99-108.
6. H. Hoppe, T. DeRose, T. Duchamp, J. McDonald and W. Stuetzle, "Mesh optimization," in *Proceedings of SIGGRAPH'93*, 1993, pp. 19-26.
7. W. J. Schroeder, J. A. Zarge and W. E. Lorensen, "Decimation of triangle meshes," in *Proceedings of SIGGRAPH'92*, 1992, pp. 65-69.
8. M. Sugeno and G. T. Kang, "Structure identification of fuzzy model," *Fuzzy Sets and Systems*, Vol. 28, 1988, pp. 15-33.
9. T. Takagi and M. Sugeno, "Fuzzy identification of system and its applications to modeling and control," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 3, 1985, pp. 116-132.
10. L. A. Zadeh, "Fuzzy sets," *Information and Control*, Vol. 8, 1965, pp. 338-353.

Chin-Chen Chang (張勤振) was born in 1969 in Hsinchu, Taiwan, Republic of China. He received his B.S. degree and M.S. degree in Applied Mathematics from National Chiao Tung University in Hsinchu, Taiwan, in 1992 and 1994, respectively, and his Ph.D. degree in Computer Science from National Chiao Tung University in Hsinchu, Taiwan, in 1998. Now he is an engineer in the Computer & Communications Research Laboratories at Industrial Technology Research Institute in Hsinchu, Taiwan. His current research interests include computer graphics, virtual reality and human-computer interface.

Shu-Kai Yang (楊舒凱) was born in 1975 in Tainan, Taiwan, Republic of China. He received his B.S. degree in Computer Science from National Chung Cheng University in 1997 and M.S. degree in Computer Science from National Chiao Tung University in Hsinchu, Taiwan, Republic of China, in 1999. Now he is an engineer in the Opto-Electronics Systems Laboratories at Industrial Technology Research Institute in

Hsinchu, Taiwan, Republic of China. His current research interests are in the area of real-time rendering technology in Computer Graphics.

Ding-Zhou Duan (段鼎洲) was born in 1975 in Kaohsiung, Taiwan, Republic of China. He received his B.S. degree and M.S. degree in Electronics Engineering from National Chiao Tung University in Hsinchu, Taiwan, Republic of China, in 1997 and 1999, respectively. Now he is an engineer in the Opto-electronics & Systems Laboratories at Industrial Technology Research Institute in Hsinchu, Taiwan, Republic of China. His current research interests include computer graphics, image processing, virtual reality, 3D transmission and human interface.

Ming-Fen Lin (林明芬) was born in 1969 in Taipei, Taiwan, Republic of China. She received her B.S. degree in Mechanical Engineering from National Taiwan University in Taipei, Taiwan, in 1991, and her M.S. degree in Electrical Engineering from University of Maryland, College Park in College Park, Maryland, United States of America, in 1993. Now she is an engineer in the Opto-Electronics & Systems Laboratories at Industrial Technology Research Institute in Hsinchu, Taiwan. Her current research interests are in the area of 3D computer graphics, virtual reality, and 3D compression.