Advanced Techniques of Functionally Based Shape Modeling with Applications in Computer Art

Pasko A., **Savchenko V.**

Shape Modeling Lab.
University of Aizu,
Aizu-Wakamatsu, Fukushima
Japan 965-8580
{pasko, savchen}@u-aizu.ac.jp

Sourin A.

School of Applied Science
Nanyang Technological University
Nanyang Avenue, Singapore 639798
assourin@ntu.edu.sg

ABSTRACT

We describe several advanced shape modeling techniques such as polygon-to-function conversion, pattern dependent interpolation of scattered data, and reconstruction from medial axis. These techniques are applied to model complex shapes for computer art works. Procedural and interactive approaches to synthetic carving are described. All introduced techniques are united under the shapes representation by real-valued functions.

KEYWORDS

Real function, shape modeling, F-rep, polygon, interpolation, medial axis, carving.

1 INTRODUCTION

This paper aims to illustrate our approach to provide complex shape modeling tools for computer art unified on the basis of a model called the *function representation* (or *F-rep*). This representation incorporates such different geometric models as set-theoretic objects, voxel data, swept objects, medial axis and skeleton based implicits [7,1,2]. In fact, any object that is defined with an inequality $f(x,y,z) \geq 0$ can be included in the model. In this paper, we describe several advanced modeling techniques and illustrate applications of them in computer-aided synthetic carving.

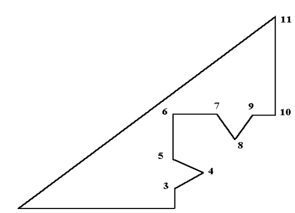
In the next Section, we describe algorithms of conversion of different input data to continuous functions of two and three variables. The polygon-to-function conversion and pattern dependent interpolation generate functions of two variables from input polygons and scattered scalar data respectively. The reconstruction from medial axis transform generates a function of three variables from the given skeletal curve and inscribed sphere radius data. In Section 3, we describe synthetic carving based on the proposed modeling techniques. Section 4 presents an example including the described models.

2 MODELING TECHNIQUES

2.1 Polygon-to-function conversion

Let a two-dimensional simple polygon be defined by a finite set of segments. The segments are the edges and their extremes are the vertices of the polygon. A polygon is simple if there is no pair of nonadjacent edges sharing a point and convex if its interior is a convex set. We consider the problem of representation of polygons by continuous real functions $F(x,y)$ taking zero values at polygon edges.

(a)



(b)

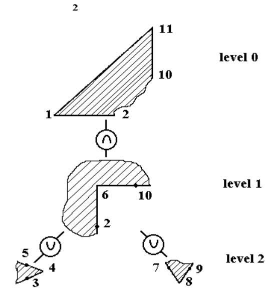


Fig. 1 A concave polygon (a) and a tree (b) representing its monotone set-theoretic formula.

The algorithm should satisfy the following requirements:

- It should provide exact polygon description as a zero set of a real function;
- No points with zero function value should be inside or outside a polygon;
- It should allow processing a simple arbitrary polygon without any additional information.

Rvachev [12] and Peterson [10] independently proposed to represent a concave polygon by a set-theoretic formula where each of the supporting half-planes appears exactly once and no additional half-plane is used. This approach does not generate any internal or external zeroes when applying R-functions. The monotone formula for the defining function of the polygon in Figure 1a results in the tree structure shown in Figure 1b. To evaluate the defining function value at the given point, the algorithm traces the tree from the leaves to the root, evaluates defining functions of half-planes and applies corresponding R-function to them. The details of the algorithm are given in [8]. An application of this algorithm is described in 3.2.

2.2 Pattern dependent interpolation

Another way to construct a function of two variables is a pattern dependent scheme for interpolation of scattered data with the help of a finite element method (FEM) [15]. It is used for smooth approximation of scattered data for the input image or any other scalar field. The method of interpolation is based on the minimum-energy properties. The input 2D data represent a set of man-made drawing points with arbitrary coordinates (x,y) and a scalar value assigned to each point, for instance, zero value for black and 255 for white colors respectively. The straightforward approach such as applying blurring operations cannot provide enough smooth resulting data. We must stress that the FEM approximation of scattered image points gives some additional possibilities to control the result by using a smoothness weight. The algorithm consists of the following steps:

1) Generation of original scattered data.
2) Sorting the data. The presented technique thus far assumed that the input image array is smaller than the image array being generated.
3) Numerical assembly. This step provides calculating the values of all elements of the FEM matrix according to the logical structure.
4) Cholesky factorization. The problem of approximating the data comes to the solution of the system of linear algebraic equations: $Ax = b$, where A is a band matrix of coefficients, x is a vector of unknown node values and b is a vector of right parts.
5) Calculating the right part for the resulting linear algebraic system, solving lower and upper triangular matrices.
6) Linear interpolation for each output array pixel.

The measured processing time on the ordinary workstation for the above mentioned steps is about one second. Applications of this algorithm are given in Sections 3 and 4.

2.3 Reconstruction from medial axis

We describe here a case study of the analytical reconstruction of an F-rep object from the medial axis transformation. A one-dimensional skeleton consisting of line segments is given in 3D space. For every point of the skeleton the radius of the maximal inscribable sphere is defined. The problem is to obtain a real function of three variables defining the original object. Let the skeleton consist of line segments with the variable sphere radius on it. Then, for the object defined by j-th segment of the skeleton with endpoints (x_{1j}, y_{1j}, z_{1j}) and (x_{2j}, y_{2j}, z_{2j}) we get the description:

$$f_{1j}(x,y,z,t) = R_j^2(t) - (x - x_{0j}(t))^2 - (y - y_{0j}(t))^2 - (z - z_{0j}(t))^2$$

where $x_{0j}(t) = x_{1j} + l_{xj}t$, $y_{0j}(t) = y_{1j} + l_{yj}t$, $z_{0j}(t) = z_{1j} + l_{zj}t$ and $l_{xj} = x_{2j} - x_{1j}$, $l_{yj} = y_{2j} - y_{1j}$, $l_{zj} = z_{2j} - z_{1j}$. To eliminate t variable, we have to find the maximal value of the function $f_{1j}(x,y,z,t)$ for the given 3D point (x,y,z) [9]. Assuming the linear function $R_j(t) = R_{1j}(1-t) + R_{2j}t$, it can be shown that the following parameter value corresponds to the function f_{1j} maximum:

$$t_{0j} = \frac{l_{xj}(x - x_{1j}) + l_{yj}(y - y_{1j}) + l_{zj}(z - z_{1j}) - R_{1j}(R_{2j} - R_{1j})}{l_{xj}^2 + l_{yj}^2 + l_{zj}^2 + (R_{2j} - R_{1j})^2}$$

Then, for the whole reconstructed 3D object the defining function is

$$f_2(x,y,z) = max_j(f_{1j}(x,y,z,t_{0j}))\ .$$

3 SYNTHETIC CARVING

Designing complex three-dimensional shapes is one of the challenging problems of computer art. Sculpting techniques have become recently one of the central themes in computer graphics literature [5,11]. Keskeys [5] postulates that “a system which will allow the reshaping of illusory computer material by carving into it, adding material to it or modeling it would open new and exciting creative opportunities”.

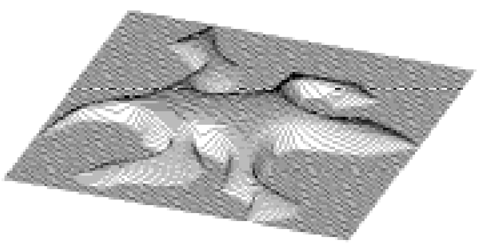
3.1 Other works

We deal with computer-aided carving in two directions:

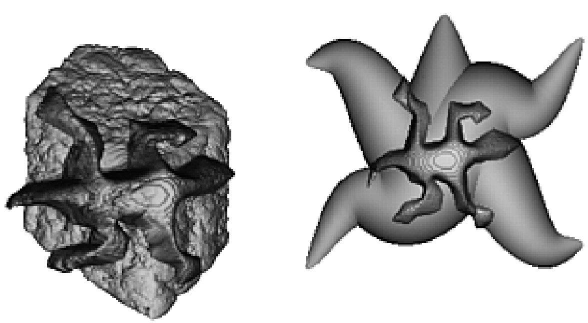
1) Definition of a shape by depth (2.5D) data with the following projection on the basic model for carving. Existing three-dimensional painting systems provide means for the depth data preparation [16,6]. Generated depth data can be used to appropriately elevate nodes of the polygonal mesh to model the carved relief [3].

2) Local modifications of the basic model using a modeled carving tool or chisel and set-theoretic operations. This approach was implemented in the voxel-based interactive modeling [4].

We consider carving as human activity aimed to produce new shapes by cutting or engraving initial shapes. There are different types of carved shapes: relief carvings, sculptures and engravings made with chisels. We propose procedural models for relief carvings, and interactive tools for carving with chisels.



(a)



(b) (c)

Fig. 2 Relief carving:
a) depth data generated by the polygon-to-function conversion procedure;
b) a reptile carved on a stone using offsetting along the normal controlled by depth data.
(c) a reptile carved on an object reconstructed from medial axis.

3.2 Relief carving

In the case of relief carving, we start from a 2D image (line drawing or greyscale raster) and generate depth data. Relief carving is modeled with the "offsetting along the normal" operation where the offset value is defined by generated depth data. The offsetting operation can be applied to an arbitrary 3D object with the normal defined at any surface point. For the functionally defined objects, the offsetting along the normal is reformulated in terms of the gradient of a defining function. We use here two above mentioned procedures of depth data generation: polygon-to-function conversion and pattern dependent interpolation.

A monotone set-theoretic expression with R-functions serves for the conversion of a 2D polygon to a function of two variables (see 2.1). Then, positive values of the function inside the polygon are used as depth data (Fig. 2a). Offsetting along the normal modulated by the depth data is applied to an arbitrary solid model to carve the relief (Figs. 2 b-c). Note that the initial object in Fig.2c is reconstructed from its medial axis as described in 2.3. This shows the advantage of having all operations closed on the same representation (F-rep).

Fig. 3 Relief carving with depth data generated by the pattern dependent interpolation

Figure 3 shows relief carving with depth data generated from a line drawing using the interpolation scheme presented in 2.2. A line drawing was digitized on the grid of 100x100 nodes and interpolated to the grid of 300x300 nodes. The obtained values in the grid nodes are used as depth data to carve the image on the initial surface.

3.3 Interactive carving and chasing

Another artistic application of functionally based solid modeling is interactive carving and chasing (see Figure 4). The artist operates with virtual cutters and dies choosing from the set of predefined shapes and sizes. If it is needed, the custom made tools can be defined and used varying geometric parameters of the cutter models. Doing this synthetic carving and chasing, the artist acts in the same way as he/she used to do it in the real world.

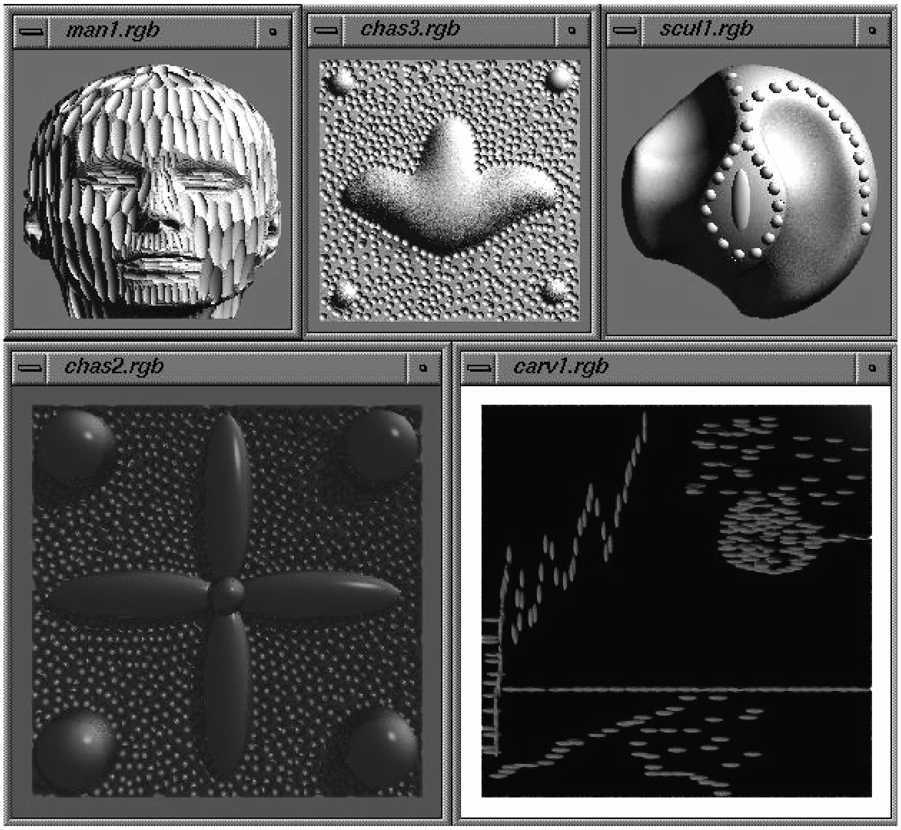


Fig. 4. Interactive carving and chasing
Upper-left: Volumetric head is carved with the elliptical cutters of different size.
Upper-middle: Synthetic chasing is done with interactive ray tracing. Subtractions and additions of spherical cutters are used. The central object is created by blending several hemispheres that simulates the real chasing when the relief is lift up from behind of the metal foil.
Upper-right: A sphere is cut with blending that simulates cutting followed by filing with sand paper. The result is unified with geometric shapes.
Lower-left: Photo realistic ray tracing of the chasing model is done with POV-Ray. Copper texture is applied to simulate the metal surface.
Lower-right: Carving on wooden board is done with elliptical cutters. Ray tracing is done with POV-Ray.

The artist places the virtual workpiece in front of him/her, chooses the current tool, defines where and how to apply it, and observes the result of the application. If the result is not acceptable, the multilevel undo operation is available. The program is implemented as a kind of interactive solid modeling tool where cutters are subsequently subtracted from and/or unified with another objects. Internally, all the tools and the workpiece itself are defined functionally and the final object is functionally defined as well. All the operations are implemented as subtraction, union and blending of solid objects. The final object can be represented in the data structure as a binary tree where each node is a set-theoretic operation and leaves are cutters.

For the visualization, an interactive ray tracing is used. Assuming that a typical model may contain about 1000 solid objects subtracted and unified with the workpiece, the ray tracing of it may take quite a long time since it assumes function evaluation for each ray cast from the virtual observer position towards the object through each pixel of the visualization window. To accelerate this process, only those parts which were affected by the most recent carving or chasing are redrawn. To estimate the size of the area affected and to detect which cutter instances are involved, the bounding boxes for the cutters are defined in the data structure for each leaf. The size of these boxes is equal to the size of the cutters if simple subtraction or union is applied and it is larger if union and subtraction with blending are used. This method ensures reasonable redrawing times of the affected areas not exceeding 0.5 sec for complex models and thus provides both photo-realism and interactivity. The final or interim model of the object can be also saved in de-facto standard POV-Ray graphics data format for further high quality ray tracing or for using together with another models. We use an extension of POV-Ray that allows us to visualize any object represented with a real function [14,13].

Fig. 5 “Geometric Mentality”:
Synthetic carving is used to model the table leg carved with depth data converted from a polygon; background results from the pattern dependent interpolation; the candle and the comet are reconstructed from medial axes.

4 EXAMPLE

The implemented carving techniques can be applied as design tools of complex three-dimensional shapes and scenes in computer art. The image presented in Fig. 5 was created using our models including synthetic carving and the modification of the POV-Ray ray tracing program that is able to render implicitly defined surfaces (isosurfaces) [14,13].

Let us explain the variety of models used to generate Fig. 5. The central object of the picture, that is a head with a drawer filled with typical solid primitives, is created by interpolation on the volume data and set-theoretic operations applied to the head and primitive solids. The hair strands are modeled as functionally defined generalized cylinders. To make a hairstyle, we applied set-theoretic operations and non-linear transformations to the cylinders. The candle holder is modeled as a patterned lattice using sweeping and set-theoretic operations. The candle is modeled as a cylinder with the top part sculpted by applying a splitting operation to remove its undesirable part. The candle is then unified with the “tears” reconstructed from the medial axis model (see 2.3). The table and the plate are solid objects defined functionally. The “reptiles” sitting on the leg of the table are created with the relief carving model (Fig. 2). The background with the clouds and the stylized comet is created with scattered data interpolation (see 2.2) and one-dimensional medial axis respectively. Because of the shapes and scene complexity, ray tracing with extended POV-Ray took several hours on a Silicon Graphics workstation.

5 CONCLUSION

We presented new techniques to model patterned lattices and relief carvings on arbitrary surfaces. An interactive system for carving with chisels was developed. All described different tools of synthetic carving are unified on the platform of the function representation. The authors believe that this representation and the described carving techniques will allow computer artists to discover a new source of shapes and operations.

The current output of our modeling system is restricted by halftone rendering and polygonization. The final model output to rapid prototyping equipment is a subject of the future work.

REFERENCES

[1] V. Adzhiev, A. Pasko, V. Savchenko, A. Sourin Shape modeling and computer graphics with real functions, Proceedings of GraphiCon’96, vol.1, GRAFO, 1996, pp. 208-213.

[2] В. Аджиев, А. Пасько, В. Савченко, А. Сурин Моделирование форм с использованием вещественных функций, Открытые Системы, 5 (19), 1996, стр.14-18.

[3] C.K. Chua, R. Gay, W. Hoheisel Computer aided decoration of ceramic tableware. Part I: 3-D decoration, *Computers and Graphics*, vol. 21, 5: 641-653, 1997.

[4] T. Galyean, J. Hughes Sculpting: an interactive volumetric modeling technique, *SIGGRAPH ‘91 proceedings*: 138-148.

[5] D.J. Keskeys Computer sculpture - new horizons, *Computer Graphics*, vol. 28, 4:255-257, 1994.

[6] C.W.A.M. van Overveld Painting gradients: free-form surface design using shading patterns, *Graphics Interface '96*, Canadian Human-Computer Communication Society, 1996: 151-158.

[7] A. Pasko, V. Adzhiev, A. Sourin, V. Savchenko Function representation in geometric modeling: concepts, implementation and applications, *The Visual Computer*, vol.11, 8:429-446, 1995.
http://www.u-aizu.ac.jp/public/www/labs/sw-sm/FrepWWW/F-rep.html

[8] A.A. Pasko, A.V. Savchenko, V.V. Savchenko. Polygon-to-function conversion for sweeping, *Eurographics/SIGGRAPH Workshop on Implicit Surfaces*, Eindhoven, The Netherlands, 1996: 163-171.

[9] Pasko A., Savchenko V. "Projection operation for multidimensional geometric modeling with real functions", Geometric Modeling: Theory and Practice, W. Strasser, R. Klein, R. Rau (Eds.), Springer-Verlag, Berlin/Heidelberg, 1997, pp. 197-205.

[10] D. Peterson, *Halfspace representation of extrusions, solids of revolution, and pyramids,* SANDIA Report SAND84-0572, Sandia National Laboratories, Albuquerque, NM, 1984.

[11] Special issue on interactive sculpting, J. Rossignac (Ed.), *ACM Transactions on Graphics*, vol. 13, 2, 1994.

[12] В.Л. Рвачев *Методы алгебры логики в математической физике*, Наукова Думка, Киев, 1974.

[13] А.И. Сурин Трассировка лучей с помощью Pov-Ray, Открытые Системы, 6, 1997, стр. 38-42.

[14] R. Suzuki POV-Ray 3 isosurface patch, http://www.public.usit.net/rsuzuki/e/povray/iso/

[15] В.А. Василенко *Сплайн-функции: теория, алгоритмы, программы*,,Наука,,Новосибирск,,1983.

[16] L. Williams 3D paint, *Computer Graphics*, vol.24, 2: 225-233, 1990..