Asian Journal of Information Technology 8 (2): 61-66, 2009
ISSN: 1682-3915
© Medwell Journals, 2009

Creating 3D Model by Area Deformation

[1, 2]Nanik Suciati and [1]Koichi Harada
[1]Graduate School of Engineering, Hiroshima University, Japan
[2]Department of Informatics, Faculty of Information Technology,
Sepuluh Nopember Institute of Technology, Indonesia

Abstract: The using of Computer Aided Design (CAD) in wide range of applications has evoked many researches in developing an easy and intuitive computer-based tool for creating both real and virtual 3D objects. This paper describes an algorithm to deform an area on surface inside the closed-curves defined by user, allowing easy creation of 3D models from scratch that begin with a simple surface as base surface and then modify the base surface using a sequence of closed-curves. We implement the algorithm of area deformation in the framework of wavelets-based multiresolution surface, so that other editing operations applicable to the multiresolution surface can be directly applied to the final 3D model. The experiment shows 3D model examples that can be easily created by the algorithm.

Key words: Computer aided design, surface deformation, multiresolution surface

INTRODUCTION

Although, CAD has been used in wide range of applications, including automotive, architectural, entertainment industries and many more, the tools currently available in commercial CAD systems that support the designing of 3D models are still not sufficiently suited (Cheutet *et al*., 2004). As a consequence, developing many types of computer-based 3D models editing tools which consider quality, cost, intuitiveness and aesthetic aspects has became an emerging research direction in recent years. The ultimate goal of researches in this area is to empower 3D modelers with direct control of a model's shape, freeing them from understanding the underlying machinery.

In computer-based modeling, sketching on study is often used in the early prototyping stages of a design, before the depicted design is manually converted into a 3D model by a trained 3D artist (Fig. 1). However, the interfaces in current commercial 3D modeling systems are mostly based on constructing geometry by selecting operations from menus, entering parameters in dialog boxes and moving control points, which are more natural to computer scientists and engineers than to artists (Olsen *et al*., 2005), who usually perform the 3D modeling by drawing or sketching. An alternative modeling system, which is based on sketch, ideally provides a more natural interface and enables the less technically artists to leverage their sketching and drawing skills for model creation.

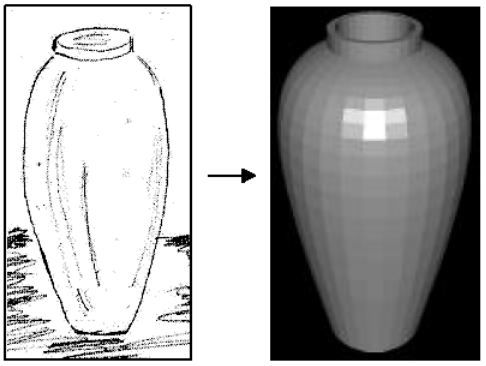

Fig. 1: a) Sketch of vase, b) before converted into 3D model of vase

SKETCH system of Zeleznik *et al*. (1996) and Teddy system of Igarashi *et al*. (1999) are examples of modeling system that create a mesh based purely on a user's two-dimensional sketch. There are also other works that begin with an existing mesh and then continued with sketch-based editing to manipulate the mesh. The research classified in the later category are: system of Lawrence *et al*. (2003), which allow user to manipulate the mesh with sketching a velocity on the surface of a mesh, system of Funkhouser *et al*. (2004) presenting a modeling by example that allows user to sketch portions of the base mesh for editing, as well as to cut out parts from other meshes to import into the base mesh and system of Zelinka *et al*. (2004) that also present a modeling by example, except their examples are not other meshes but

Corresponding Author: Nanik Suciati, Department of Informatics, Faculty of Information Technology, Sepuluh Nopember Institute of Technology, Indonesia

user-drawn curves that guide deformation along slices through the mesh. Free-form deformations (Sedenberg *et al*., 1986; Singh *et al*., 1998) modify a given surface shape through a suitable deformation, typically using a force field function. The advantage of these methods is that they are in principle, applicable to any surface modeling setting. However, due to smoothness constraint of the underlying surface, implementation of free-form deformation, which should allow deformation on arbitrary area of surface is not trivial, more over if the area border are not aligned with iso-parameter lines or patch boundaries of the surface.

The goal of the research is to describe an algorithm to deform an arbitrary area on surface inside the closed-curves defined by user, allowing easy creation of 3D models from scratch that begin with a simple surface and then modify the base surface using a sequence of closed-curves. We implement the algorithm of area deformation in the framework of wavelets-based multiresolution surface, so that other editing operations applicable to the multiresolution surface can be directly applied to the final 3D model. The surface representation we used is endpoints interpolated cubic B-spline surface, which have uniform knots distance except knots at surface endpoints which have multiplicity 4. This surface is defined using patch scheme, which is computed from tensor product of parametric basis function. The basis functions is polynomial functions with degree 3, have continuity C2 at joint (meeting point between two series of surface patches) and have uniform knots distance.

MATERIALS AND METHODS

Overview of the system for creating 3D models from scratch using simple multiresolution surface as the base surface and then, modify the base surface using area deformation defined by a sequence of user's closed-curves is shown in Fig. 2. There are three main processes in the system: process for creating a simple multiresolution base surface, process for drawing a closed-curve on the base surface, process for deforming an area on surface. The right column of Fig. 2 shows the design example after completing each process.

The first process creates a base surface using mesh of control points inputted by user. In the second process, user draws closed-curve on the base surface and inputs height of deformation. The third process deforms an area inside the closed-curve with positive or negative height specified by user. User can repeat the second and third process to draw a sequence of closed-curves and apply the area deformation until he gets the desired 3D model.

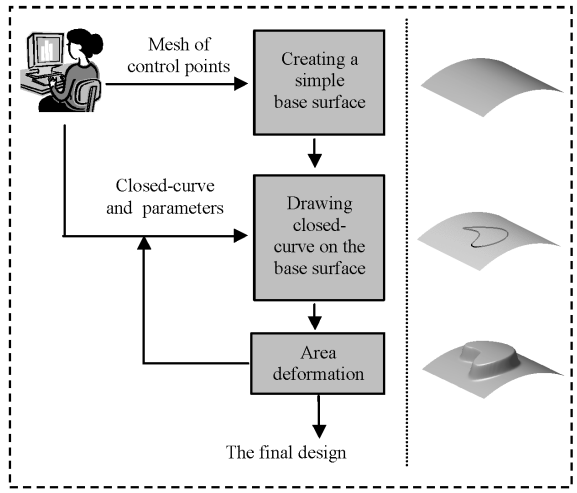

Fig. 2: Overview of the system for creating 3D model using area deformation

Creating a simple base surface: Since, we use wavelets-based multiresolution surface as the framework of the algorithm, the surface can be represented in different resolutions, allowing local editing as well as global editing while preserve the detail information of the surface. Let multiresolution of tensor product b-spline surface c (u, v) be defined as an element of a certain functional space V^n, computed by:

$$\begin{aligned} c(u,v) &= \sum_{i=1}^{2n+3}\sum_{j=1}^{2n+3}\left(x^n\right)\phi_i^n(u)\phi_j^n(v) \\ &= \left(x^n\right)\left(\Phi(u)^n\right)\left(\Phi(v)^n\right) \end{aligned} \tag{1}$$

where, x^n is a control points mesh with size m×m in R^2, $m = 2^n+3$. User can create a simple base surface by inputting mesh of control point's x^n, with size 4×4 at resolution 0.5×5 at resolution 1.7×7 at resolution 2, etc.

The multiresolution surface is extension to the research of Finkelstein *et al*. (1994) in multiresolution curve. Decomposition process that split the mesh of control points x^n into lower resolution mesh x^{n-1}, detail in row dx^{n-1} and detail in column dy^{n-1} is computed by using analysis filters A^n and B^n as follow:

$$\begin{aligned} xt^n_{(m\times m')} &= \left(A^n_{(m'\times m)}x^n_{(m\times m)}\right)^T \\ dx^{n-1}_{(m''\times m)} &= B^n_{(m''\times m)}x^n_{(m\times m)} \\ dy^{n-1}_{(m''\times m')} &= B^n_{(m''\times m)}xt^n_{(m\times m')} \\ x^{n-1}_{(m'\times m')} &= \left(A^n_{(m'\times m)}xt^n_{(m\times m')}\right)^T \end{aligned} \tag{2}$$

As inverse of the decomposition process, reconstruction process which allows recovering the original mesh o f control points x^n from x^{n-1}, dx^{n-1} and dy^{n-1} is computed by using synthesis filters P^n and Q^n as follow:

$$\begin{aligned} xt^{n-1}_{(m'xm)} &= \left(P^n_{(mxm')}x^{n-1}_{(m'xm')} + Q^n_{(mxm'')}dy^{n-1}_{(m''xm')}\right)^T \\ x^n_{(mxm)} &= P^n_{(mxm')}xt^{n-1}_{(m'xm)} + Q^n_{(mxm'')}dx^{n-1}_{(m''xm)} \end{aligned} \quad (3)$$

Drawing closed-curve on the base surface: We use uniform cubic B-spline curve to represent the closed-curve defined by user. Uniform cubic B-spline curve is a curve composed by cubic B-spline basis function with uniform knots distance. Cubic B-spline basis function is a parametric polynomial function with degree 3, piecewise and has continuity C2 at joint. Continuity C2 means that the basis function has continuity at position, first derivative and second derivative.

Figure 3a shows the example of curve composed using uniform cubic B-spline basis function. One segment of uniform cubic B-spline curve is defined using 4 control points. The curve approximates the 4 control points which have index from 0-3. Since each curve segment is defined using 4 basis functions, the curve requires 3 more basis functions and 3 more control points than the number of curve segments. Each basis function has non zero value at 4 parametric intervals. The most left basis function adds 3 extra intervals in the left of curve and the most right basis function adds 3 extra intervals in the right of curve. There are m+1 control points, m+1 basis functions, m-2 curve segments in the range of m-1 knots and m-1+3+3=m+5 knots in total. Curve is drawn with $\bar{u}$ parameter running from $\bar{u}_3$ to $\bar{u}_{m+1}$.

The curve example shown in Fig. 3a has the last index of control point m = 9. The curve consists of 7 segments. Its definition requires 10 basis functions and 10 control points at $\bar{u}_3 \le \bar{u} < \bar{u}_{10}$ with parameter space $\bar{u}_0 \le \bar{u} < \bar{u}_{13}$. The uniform cubic B-spline basis function used to compose the curve in Fig. 3a is shown in Fig. 3b.

In order to draw a curve, some control points V_i have to be defined first and then, the control points are used to draw curve Q using the following equation:

$$\begin{aligned} Q_i(u) &= V_{i+r}b_r(u) \\ &= V_{i-3}b_{-3}(u) + V_{i-2}b_{-2}(u) + \\ &\quad V_{i-1}b_{-1}(u) + V_{i-0}b_{-0}(u) \end{aligned} \quad (4)$$

The basis functions $b_r(u)$ for cubic B-spline curve are shown in Fig. 4 and have the following equation:

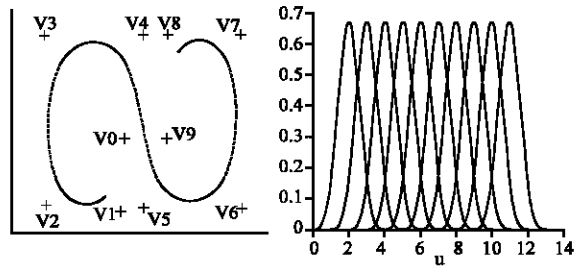

Fig. 3: a) the example of uniform B-spline curve, b) the basis functions of uniform cubic B-spline

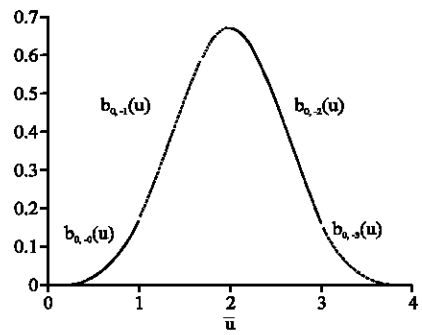

Fig. 4: Uniform cubic B-spline basis function $B_{0,4}$ (u)

$$\begin{aligned} b_{i,-0}(\bar{u}) &= \frac{1}{6}u^3 \\ b_{i,-1}(\bar{u}) &= \frac{1}{6}(1+3u+3u^2-3u^3) \\ b_{i,-2}(\bar{u}) &= \frac{1}{6}(4-6u^2+3u^3) \\ b_{i,-3}(\bar{u}) &= \frac{1}{6}(1-3u+3u^2-u^3) \end{aligned} \quad (5)$$

Area deformation: Surface deformation is carried out on the area bordered by the closed-curve defined by user. In this case, there are two problems. First, how to identify mesh of control points located inside the area of the closed-curve? Second, how to displace the mesh of control points, while maintaining the smoothness of the base surface? If the curve boundary is aligned with control points mesh or patch of surface, computing the displacement of control points around the curve boundary is relatively easy, but if the curve boundary is not aligned with control points mesh or patch of surface, due to smoothness constraint, computing the displacement of control points around the curve boundary is not trivial.

In the algorithm, the three main steps used to deform an arbitrary area on surface are increasing resolution of surface mesh in order to maintain smoothness of the based surface, classifying all mesh of control points in

4 different types related to their distance from the closed-curve and displacement the 3 types of the mesh of control points according to butterworth function profile.

Figure 5 shows the surface and mesh examples that depict overview of the three main steps in the algorithm. Figure 5b and c are input and output of the step 1, Fig. 5c and d are input and output of the step 2, Fig. 5d and e are input and output of the step 3. While, input and output of the area deformation algorithm are shown in Fig. 5a and f, respectively.

Since, we research in the framework of wavelets-multiresolution representation of B-spline surface, increasing and decreasing the resolution of surface mesh can be easily carried out. In the multiresolution representation, 0 is the coarsest resolution of the surface consisted of 4×4 mesh of control points for each X, Y and Z coordinates. Increasing the resolution of surface is carried out by computing Eq. 7 with mesh of control points at current resolution as x^{n-1}, 0 as detail in row dx^{n-1} and 0 as detail in column dy^{n-1}. Since the higher the resolution of the surface mesh, the more number of mesh control points used to represent the surface, representing surface in high resolution would maintain the shape of detail features and the smoothness of the surface in the augmentation process.

After increasing the resolution of mesh, each control point is classified into 4 different types related to its position from the closed-curve:

- Control points that locate on the closed-curve
- Control points that locate inside the closed-curve
- Control points that locate at distance ≤ radius from the closed-curve
- Control points that locate at distance > radius from the closed-curve
- Euclidian distance is used to compute the closest distance between control point and the closed-curve. The value of radius is stated as 1

Only control points belonged to the type 1-3, will be displaced into new position. All control points belonged to the type 2 and 3 are displaced in the surface normal direction with positive or negative height defined by user. In order to maintain smoothness of the surface, new position of all control points belonged to the type 1 is computed using butterworth, one of the most commonly used digital filters in image processing (Gonzales and Woods, 2002), as the force function. In its original application domain, butterworth filters are transition between two extreme filters, i.e., Ideal and Gaussian filters, which the shape can be easily modified to the others filter. The following equation is used to define butterworth function:

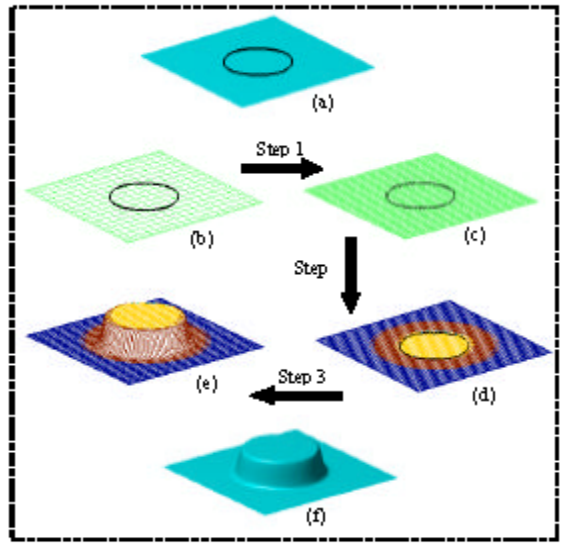

Fig. 5: Overview of area deformation

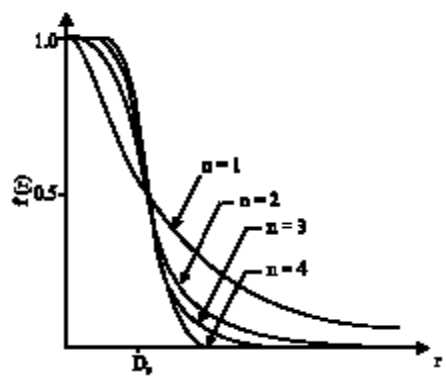

Fig. 6: The butterworth functions defined using different values of parameter n and $D_0 = 0.5$

$$f(r) = \frac{1}{1 + [r / D_0]^{2n}} \quad (6)$$

Figure 6 shows shapes of butterworth functions defined by different values of n, $D_0 = 0.5$ and r assumed as Euclidian distance of control points to the user-defined curve. The values of f (r) vary from 1 to 0, with cut-off frequency (the transition between f (r) =1 and f (r) = 0) D_0 and order of the function n. We use values n = 4 and $D_0 = 0.3$ in the algorithm of area deformation.

RESULTS AND DISCUSSION

The objective of experiment is to show that the system can be used to create simple 3D models from scratch easily. We show the designing process of two objects: hat and panel in step-by-step, to depict the

intuitiveness and easiness of the system. In designing process of the 2 models, user does not need to modify/edit the surface by moving one-by-one of control point mesh, since modifying the surface is specified by drawing the closed-curve on surface as the intended area of deformation. The experiments are carried out in Matlab 2008.

Hat model: The steps of creation hat model are shown in Fig. 7. First, user defines control points mesh of simple circle surface at resolution 0 as a base surface. The mesh of control points of the base surface shown in Fig. 7a are defined in X, Y and Z coordinates, each with size 4×4, as follows:

X

3	4	6	7
2	4	6	8
2	4	6	8
3	4	6	7

Y

7	8	8	7
6	6	6	6
4	4	4	4
3	2	2	3

Z

2	2	2	2
2	2.5	2.5	2
2	2.5	2.5	2
2	2	2	2

Figure 7 shows grid and direction of coordinate X, Y and Z, so that the reader could imagine the position of each control points in 3D space. All control points in Z is 2, except control points in the middle that have value 2.5. As shown in Fig. 7a, the circle surface is not flat, but a bit high at center. Using mesh of control points inputted by user, the system create a base surface by implementing Eq. 1.

Next, user specifies the area deformation by drawing closed-curve directly on the base surface, as shown in Fig. 7b and inputting the height parameter. In this example, the user inputs 0.5 as the value of height parameter. As the result, area inside the closed-curve on the base surface is deformed with height 0.5 (Fig. 7c). Again, user specifies another area deformation by drawing 2 closed-curves and 0.1 as height parameter, as shown in Fig. 7d. Figure 7e shows the final design of the hat model.

Panel model: The steps of creation panel model are shown in Fig. 8. First, user defines control points mesh of simple flat rectangular surface at resolution 0 as a base surface. The mesh of control points of the base surface shown in Fig. 8a are defined in X, Y and Z coordinates, each with size 4×4, as follows:

X

2	2	2	2
4	4	4	4
6	6	6	6
8	8	8	8

Y

2	7	12	15
2	7	12	15
2	7	12	15
2	7	12	15

Z

2	2	2	2
2	2	2	2
2	2	2	2
2	2	2	2

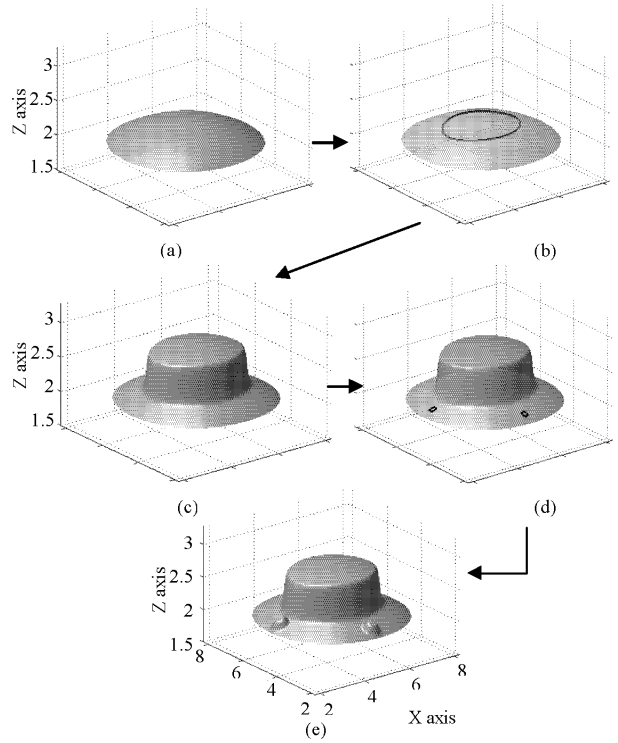

Fig. 7: Step-by-step of creating hat model

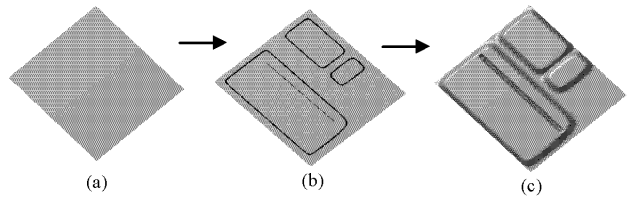

Fig. 8: Step-by-step of creating panel model

Next, on the base surface, user draws 3 closed-curves and 1 opened-curve, as shown in Fig. 8b. The final design of the panel model after deforming areas inside the closed-curve with positive height 0.1 is shown in Fig. 8c.

Since, we implement the algorithm of area deformation in the framework of multiresolution surface, at the step of creating base surface, user must input mesh of control points with size that follow the rule of resolution. For example, at the coarsest resolution 0, the size is 4×4 for each X, Y and Z coordinate, at resolution 1 the size is 5×5, etc.

In creation hat and panel model, user input mesh of control point with size 4×4 (resolution 0). During the process of area deformation, resolution of the base surface is increased to resolution 6, with size 67×67, in order to maintain smoothness of the surface. The running time of area deformation algorithm is sufficient fast, regarding the need of interactive respond of the system.

CONCLUSION

We have proposed an algorithm for deforming an area on surface inside the closed-curve defined by user and incorporate the algorithm to the system for creating 3D models from scratch. As shown in the experiment, the system allows easy and intuitive creation of 3D models. In order to modify, the surface shape, user just specifies the deformation by drawing a sequence of closed-curves on surface and inputting the height parameter. User does not need to modify the surface by moving one-by-one control points, which is known as a difficult and boring task. In the next study, we plan to improve the system by providing some simple surfaces as the base surface and/or allowing user to create the base surface by drawing a curve, that would avoid user to define control points mesh of the base surface manually in the very first step of creation 3D models.

REFERENCES

Cheutet, V., C.E. Catalano, J.P. Pernot, B. Falcidieno, F. Giannini and J.C. Leon, 2004. 3D sketching with fully free form deformation features (δ-F4) for aesthetic design. Proc. of Eurographics Workshop on SKETCH-Based Interfaces and Modeling, Eurographics, pp: 9-18.

Finkelstein, A. and D.H. Salesin, 1994. Multiresolution curves. Proc. of the 21st annual conference on Computer graphics and interactive techniques, ACM press, New York, pp: 261-268. DOI: 10.1145/192161.192223. http://portal.acm.org/citation.cfm?id=192223.

Funkhouser, T., M. Kazhdan, P. Shilane, P. Min, W. Kiefer, A. Tal, S. Rusinkiewicz and D. Dobkin, 2004. Modeling by example. ACM Trans. Graphics, 23 (3): 652-663. DOI: 10.1145/1015706.1015775. http://portal.acm.org/citation.cfm?id=1015706.1015775.

Gonzales, R.C. and R.E. Woods, 2002. Digital Image Processing. 2nd Edn. Prentice Hall, New Jersey, pp: 167-180. ISBN: 0-13-094650-8.

Igarashi, T., S. Matsuoka and H. Tanaka, 1999. Teddy: A sketching interface for 3D freeform design. Proc. the 26th Annual Conference on Computer Graphics and interactive techniques, ACM Press/Addison-Wesley Publishing Co., New York, pp: 409-416. DOI: 10.1145/311535.311602. http://portal.acm.org/citation.cfm?doid=311535.311602.

Lawrence, J. and T. Funkhouser, 2003. A painting interface for interactive surface deformation. Graphical Models, Special issue on Pacific Graphics, Academic Press Professional, Inc., San Diego, pp: 418-438. DOI: 10.1016/j.gmod.2004.05.008. http://portal.acm.org/citation.cfm?id=1055753.

Olsen, L., F.F. Samavati, M.C. Sousa and J.A. Jourge, 2005. SKETCH-based Mesh Augmentation. Proc. Eurographics Workshop on Sketch-Based Interfaces and Modeling, Eurographics.

Sedenberg, T. and S. Parry, 1986. Free-form deformation of solid geometric models. ACM SIGGRAPH Computer Graphics, 20 (4): 151-160. DOI: 10.1145/15886.15903. http://portal.acm.org/citation.cfm?id=15903.

Singh, K. and E. Fiume, 1998. Wires: A geometric deformation technique. Proc. the 25th annual Conference on Computer Graphics and Interactive Techniques, ACM Press, New York, pp: 405-414. DOI: 10.1145/280814.280946. http://portal.acm.org/citation.cfm?id=280946.

Zeleznik, R., K. Herndon and J. Hughes, 1996. Sketch: An interface for sketching 3D scenes. Proc. the 23rd Annual Conference on Computer Graphics and Interactive Techniques, ACM Press, New York, pp: 163-170.

Zelinka, S. and M. Garland, 2004. Mesh modeling with curve analogies. Proc. the Computer Graphics and Applications, 12th Pacific Conference, IEEE Computer Society, Washington DC, pp: 94-98. http://portal.acm.org/citation.cfm?id=1025128.1026032.