The Visual Computer

Direct manipulation of generalized cylinders based on B-spline motion

Tae-Ick Chang, Joo-Haeng Lee, Myung-Soo Kim, Sung Je Hong

Department of Computer Science, POSTECH,
Pohang 790-784, South Korea
E-mail: tichang,jhlee,mskim,sjhong@postech.ac.kr

We present a direct manipulation technique that allows interactive control of the shape of generalized cylinders. We interpret the generalized cylinder as the sweep surface of a planar cross-sectional B-spline curve under B-spline motion. The generated surface is a NURBS surface that interpolates a given sequence of cross-sectional curves along a skeleton curve. Directly manipulating a surface point on the generalized cylinder modifies the cross-sectional shape and its motion and deforms the generalized cylinder to interpolate the exact point location specified by the user. The surface is deformed by a target tracking procedure.

Key words: Generalized cylinder – B-spline motion – Sweep surface – NURBS – Direct manipulation – Target tracking

Correspondence to: M-S Kim

1 Introduction

Advanced techniques for modeling realistic and complex 3D shapes are important in computer graphics and animation. There has been much research effort in developing such techniques. As a result, commercial systems such as 3Design provide various handy tools for shape design [25]. Many of these tools are based on sweeping techniques. In fact, even simple sweeping techniques, such as linear extrusion and rotational sweeping, are among the tools most frequently used for designing 3D shapes.

Generalized cylinders provide a useful tool for the modeling and animation of flexible objects with tubular shape [14]. Using effective shape control techniques for generalized cylinders, the user can design natural shapes such as trees, arms, legs, and bodies with great ease. This paper presents such a technique. It allows the user to interactively control the shape of generalized cylinders by directly manipulating surface points on the generalized cylinders.

Given a 2D cross-sectional curve and a 3D skeleton curve, the generalized cylinder is defined as the sweep surface of the cross-sectional curve moving along the skeleton curve. The cross-sectional curve may change its shape dynamically. However, in most conventional methods, the cross-sectional plane is restricted to be orthogonal to the tangent direction of the skeleton curve. Because of this constraint, direct manipulation of generalized cylinders is a nontrivial task. It is very difficult to manipulate the skeleton curve while keeping it orthogonal to each cross-sectional plane. In conventional methods [4, 6, 14, 24], the Frenet moving frame is used to define the local coordinates on each cross-sectional plane. That is to say, the unit normal and binormal vectors are used to form the basis of each cross-sectional plane. These unit vectors are nonrational in general. Consequently, the generalized cylinders based on these vectors are also nonrational, even if the cross-sectional and skeleton curves are both rational.

In this paper, we eliminate these difficulties by relaxing the definition of generalized cylinders so that an arbitrary moving frame can be used in defining the local coordinates of each cross-sectional plane. We interpret the generalized cylinder as a smooth deformation of a cylinder, in which each cross-sectional curve is obtained by a smooth deformation of the circle. The deformation of cross-sectional curves can be represented as a gen-

The Visual Computer (1998) 14:228–239

eralized linear extrusion in which each cross-sectional curve changes its shape dynamically while moving along the direction of extrusion. Each cross-section is then placed along the skeleton curve by an appropriate euclidean motion (i.e., with translation and rotation). In our definition of the generalized cylinder, a cross-sectional plane may not necessarily be orthogonal to the skeleton curve. The shape, position, and orientation of each cross-sectional curve can be controlled independently, which makes a direct manipulation of generalized cylinders significantly easier.

Jüttler and Wagner [13] introduced the B-spline motion that represents a 3D euclidean motion by a 4×4 matrix (in homogeneous coordinates), each element of which is given as a B-spline function. An arbitrary point under a B-spline motion generates a rational B-spline curve. Moreover, the cross-sectional B-spline curve under a B-spline deformation, as well as a B-spline motion, generates a tensor-product rational B-spline sweep surface. By defining all underlying geometric, kinematic, and deformation components with B-spline basis functions, we can represent the generalized cylinder as a NURBS surface that can be supported in conventional CAD systems. In most previous methods, the generalized cylinders have nonrational forms; thus it is necessary to approximate them with many polygons or B-spline surface patches for CAD data exchange.

When the user directly manipulates a surface point on the generalized cylinder, the positional change of the surface point is converted into the corresponding changes in the shape, position, and orientation of a cross-sectional curve. Given a cross-sectional curve, its deformation and spatial motion determine the surface shape of a generalized cylinder. Each surface point is thus given as the result of a transformation that maps the deformation and motion parameters (of the cross-sectional curve) into the surface point. As the user moves the surface point into a new location, the underlying deformation and motion parameters (of the cross-sectional curve) must be updated so that the new surface point is interpolated by a deformed generalized cylinder. This is based on a numerical iteration procedure similar to those used in target tracking [15]. This technique has been implemented on a Pentium II PC (266 MHz). The implemented system gives real-time performance while supporting the interactive design of generalized cylinders. Some illustrative examples are given in this paper.

The rest of this paper is organized as follows. In Sec. 2, we briefly review previous work. In Sect. 3, we present some mathematical preliminaries for B-spline motion. Section 4 explains our mathematical model of generalized cylinders based on the B-spline motion. Section 5 presents a technique for direct manipulation of generalized cylinders. Some experimental results and illustrative examples are also demonstrated in this section. Finally, in Sect. 6, we conclude this paper.

2 Previous work

The generalized cylinder was first introduced in computer vision for the purpose of modeling and recognizing 3D shapes (such as airplanes, snakes, horses, and dolls) based on a few simple procedural rules [1]. Shani and Ballard [20] present a good survey on the early development of generalized cylinders in computer vision. Compared with other explicit object-representation schemes such as boundary representation (B-rep), procedural descriptions make the shape recognition task significantly easier since shape features can be described better with procedural rules.

The procedural way of modeling generalized cylinders also provides similar advantages in computer graphics and geometric modeling. We can describe complex 3D shapes using a few simple procedural rules in a compact way; thus one can reduce the size of the object database. Unfortunately, most conventional methods generate generalized cylinders as nonrational surfaces. Consequently, one needs to develop special algorithms for the display and geometric processing of generalized cylinders. Bronsvoort [2] and Bronsvoort and Klok [3] develop display algorithms for generalized cylinders. Note that the display algorithms essentially compute the intersection of a generalized cylinder with a line (in ray tracing) or a plane (in surface scanning). Thus, it is relatively easier to develop display algorithms than many other geometric algorithms (e.g., surface/surface intersection) for generalized cylinders.

It is unreasonable to develop special algorithms for every single geometric problem applied to generalized cylinders. A more reasonable approach would be either to approximate the generalized cylinders with rational surfaces or to redefine them as rational surfaces. Then we can apply a variety of geometric algorithms that have been developed for rational surfaces such as NURBS surfaces. The first approach was taken previously by many authors such as Bronsvoort and Waarts [4], Coquillart [6], and Kim et al. [14]. Recently, Johnstone and Williams [11] and Jüttler [12] use rational motions to design rational sweep surfaces (in which the moving curve has a fixed shape). It is quite straightforward to extend these results to include a smooth rational deformation of the cross-sectional curve in the formulation of these sweep surfaces. In this paper, we assume the definition of generalized cylinders based on this extension of rational sweep surfaces.

Given a sequence of cross-sectional curves C_i lying on the planes $z=z_i$ (parallel to the xy-plane), Shinagawa and Kunii [22] suggest a homotopy-based method that interpolates these curves by blending each pair of two consecutive cross-sections. The resulting surface is a generalization of linear extrusion in which the 2D cross-section changes its shape dynamically while the extrusion is carried out along the z direction. By mapping each 2D cross-section (of the generalized linear extrusion) to the normal plane of a skeleton curve, Tai et al. [24] present a homotopy sweep method that generates a generalized cylinder interpolating a given sequence of cross-sectional curves with arbitrary shape.

In this paper, we assume that the cross-sectional curves are given as rational curves of the same degree and knot sequence (by applying degree elevation and knot insertion if necessary). These curves can be interpolated by a rational skinning surface [17]. Each cross-section of this skinning surface represents the shape deformation of the cross-sectional curve. The placement of each cross-sectional curve along the skeleton curve is given by a B-spline motion that is totally independent of the skeleton curve. In some sense, our representation scheme of generalized cylinders is closely related to the modeling techniques of Shinagawa and Kunii [22] and Tai et al. [24]. The main difference, however, is that our approach is motivated by rational representation of generalized cylinders based on a recent development of B-spline motion technique [13].

Direct manipulation enhanced with appropriate 3D widgets provides an effective user interface for manipulating 3D objects [23]. Grimm et al. [9] develop various interface tools for directly manipulating free-form 3D objects, in particular, those for simple sweep, warp, and blend operations. However, they have not included a functionality that can directly manipulate surface points on a sweep surface. Post and Klok [18] provide a deformation method for sweep-defined polyhedral surfaces. The shape control is restricted to the cross-sectional planes given at some discrete locations along the skeleton curve. Therefore, it is a challenging task to develop a technique that can automatically modify the shape of a sweep surface (or a generalized cylinder) when the user moves an arbitrary surface point to a new location.

Hsu et al. [10] suggest a direct manipulation technique for free-form objects defined by free-form deformation (FFD). An FFD method deforms an enclosing space of an object so as to apply a local/global shape change to the object [19]. The FFD involves a mapping (from R^3 to R^3) that is represented by a trivariate tensor product free-form volume. A direct manipulation of FFD objects is based on a nonlinear inversion of this mapping. Lee at al. [16] apply a similar technique (based on a multilevel FFD) to generate a smooth transition (morphing) of an image to a target image.

The main difference between the direct manipulation scheme based on FFDs and our approach in this paper is in the nonlinear mapping involved in the shape transformation of an object. In our motion-based approach, the mapping is from a configuration space (of the sweep motion) to the workspace R^3. To realize a direct manipulation of a sweep surface, the positional change of a surface point must be inverted to the corresponding change of the sweep motion (represented as a curve in the configuration space). A sweep motion consists of three basic components: translation, rotation, and scaling. Because of the noneuclidean structure of 3D rotations, the nonlinear inversion is more complex than the case of FFD-based methods. Nevertheless, the basic approach is similar.

There are some other previous works related to our approach. Gleicher and Witkin [7] suggest a technique called *differential manipulation* as a general solution method for directly manipulating geometric objects under constraints. This technique can be applied to the direct manipulation of generalized cylinders with appropriate nonlinear constraints. Gleicher and Witkin [8] show that, using a similar technique, the virtual camera can be controlled by directly manipulating image points on the display screen. Recently, Kyung et al. [15] suggest a simpler method for virtual camera control based on a target tracking procedure. In this paper, we apply a similar technique for the direct manipulation of generalized cylinders.

3 Mathematical preliminaries

In this section, we briefly review some mathematical preliminaries for B-spline motion. A B-spline motion consists of its translational and rotational components, both of which are represented in B-spline forms. Let $\mathbf{v}(t)=\frac{1}{v_w(t)}\left(v_x(t), v_y(t), v_z(t)\right)\in\mathbf{R}^3$ be a rational B-spline curve of degree d_1, and $\mathbf{r}(t)=(r_w(t), r_x(t), r_y(t), r_z(t))\in\mathbf{R}^4$ be a polynomial B-spline curve of degree d_2, where $v_w(t)$, $v_x(t)$, $v_y(t)$, $v_z(t)$, $r_w(t)$, $r_x(t)$, $r_y(t)$, $r_z(t)$ are all B-spline functions. The 3D curve $\mathbf{v}(t)$ represents the translational component of the motion. Moreover, the unit quaternion curve $\mathbf{q}(t)=\mathbf{r}(t)/\|\mathbf{r}(t)\|\in\mathbf{S}^3$ represents the 3D rotational component [21]. Using $\mathbf{v}(t)$ and $\mathbf{r}(t)$, we can formulate a B-spline motion as a 4×4 matrix (in homogeneous coordinates), each component of which is of degree d_1+2 d_2:

$$M(t)=\left(\begin{array}{c|c} v_w(t)\,\delta(t) & 0\quad 0\quad 0 \\ \hline v_x(t)\,\delta(t) & \\ v_y(t)\,\delta(t) & v_w(t)\,D(t) \\ v_z(t)\,\delta(t) & \end{array}\right), \qquad (1)$$

where

$$D(t)=\begin{pmatrix} r_w^2+r_x^2-r_y^2-r_z^2 & 2\left(r_xr_y-r_wr_z\right) & 2\left(r_xr_z+r_wr_y\right) \\ 2\left(r_xr_y+r_wr_z\right) & r_w^2-r_x^2+r_y^2-r_z^2 & 2\left(r_yr_z-r_wr_x\right) \\ 2\left(r_xr_z-r_wr_y\right) & 2\left(r_yr_z+r_wr_x\right) & r_w^2-r_x^2-r_y^2+r_z^2 \end{pmatrix},$$

$$\delta(t)=r_w^2+r_x^2+r_y^2+r_z^2.$$

Since each element of $M(t)$ is a B-spline polynomial, we can obtain the control matrices $\{A_i\}$ of $M(t)$ as follows:

$$M(t)=\sum_{i=0}^{n}N_i^d(t)A_i, \qquad (2)$$

where $N_i^d(t)$ is the ith rational B-spline basis function of degree d [17]. The matrix A_i represents an affine transformation, in general, even though the matrix $M(t)$ represents a rigid 3D euclidean motion. Note that the trace of an arbitrary point $\mathbf{p}$ under a B-spline motion $M(t)$ generates a rational B-spline curve

$\mathbf{p}(t)=M(t)\cdot\mathbf{p}$.

The first column of $M(t)$ has $\delta(t)$ as a common factor. When we divide each element of $M(t)$ by $v_w(t)\ \delta(t)$, we realize that the translational and rotational components of the motion $M(t)$ are independent. Jüttler and Wagner [13] formulate the matrix representation of B-spline motion somewhat differently, so that the overall degree can be reduced. We skip the details of constructing the B-spline motion matrix from the B-spline representations of $\mathbf{v}(t)$ and $\mathbf{r}(t)$. See Chang [5] and Jüttler and Wagner [13] for more details.

4 The generalized cylinder based on B-spline motion

This section briefly explains our mathematical model of generalized cylinders. The shape of each cross-sectional curve is determined (with a smooth deformation) and the cross-sectional plane is then placed at appropriate position and orientation in a B-spline motion (see Eq. 2):

$$M(t)=\sum_{i=0}^{n}N_i^d(t)A_i.$$

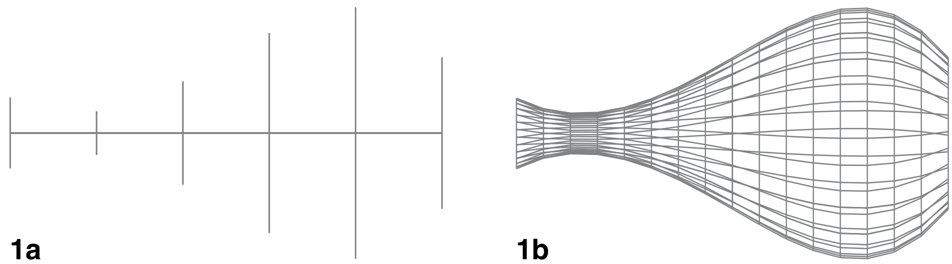
The sweep of a rational B-spline curve $C(u)$ under the B-spline motion $M(t)$ generates a tensor product rational B-spline surface:

$S(u, t)=M(t)\cdot C(u)$.

When we take a 2D cross-sectional closed curve $C(u)$ and apply a B-spline affine transformation $A(t)$ to $C(u)$, we get a deformed cross-sectional B-spline curve:

$C_t(u)=A(t)\cdot C(u)$.

A simple special case of $A(t)$ includes the xy scaling that can be represented by a diagonal matrix. Note that, in conventional methods, the xy scaling is the most popular deformation for cross-sectional curves. In the most general case, we may apply a



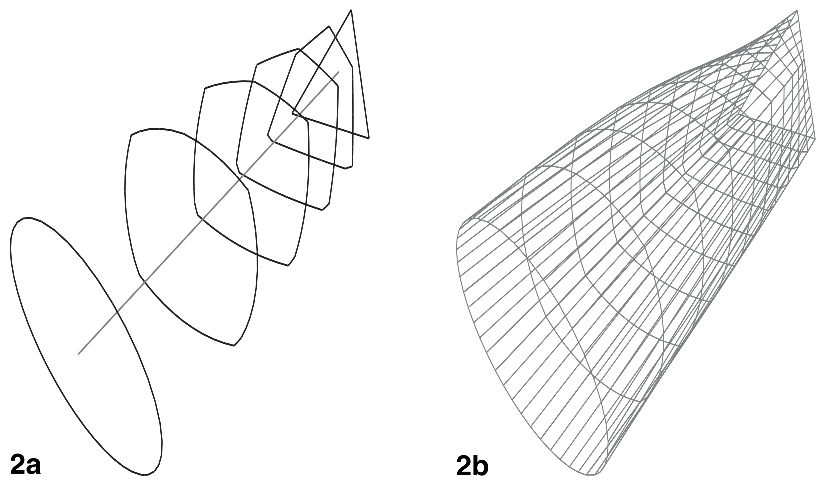


Fig. 1a, b. Generalized cylinder with cross-sectional circles

Fig. 2a, b. Generalized cylinder with variable cross-sections

general deformation $D(t)$ to $C(u)$ and get a free-form closed B-spline curve:

$C_t(u)=D(t)\cdot C(u)$.

When we sweep the 2D curve $C_t(u)=(x_t(u), y_t(u))$ along the t-axis (orthogonal to the xy-plane), we get a B-spline surface of generalized linear extrusion in the xyt space:

$C(u, t)=\{(x_t(u), y_t(u), t) | 0\le u, t\le 1\}$.

The axis of this surface is

$l(t)=\{(0, 0, t) | 0\le t\le 1\}$.

Under the B-spline motion $M(t)$, this linear axis $l(t)$ will be transformed into the skeleton (rational) B-spline curve $\mathbf{v}(t)$ of the generalized cylinder:

$\mathbf{v}(t)=M(t)\cdot(1, 0, 0, 0)^t$,

where $(1, 0, 0, 0)^t$ is the coordinate of the origin $(0, 0, 0)^t$ in homogeneous coordinates. The generalized cylinder itself is given as a tensor product rational B-spline sweep surface:

$S(u, t)=M(t)\cdot C_t(u)$.

Figure 1 shows a surface of revolution. The cross-sectional curves are circles of different radii. Uniform scaling is applied to the deformation of each cross-sectional curve. In Fig. 2, each cross-sectional curve has a different shape. Using a skinning surface that interpolates a given sequence of cross-sectional curves, we can generate a generalized linear extrusion. When we deform this surface so that its main axis transforms into an arbitrary skeleton curve and each cross-sectional plane is placed at appropriate position and orientation along the skeleton curve, we can generate a generalized cylinder.

5 Shape control of generalized cylinders

In this section, we describe how to control the shape of a generalized cylinder by directly manipulating an arbitrary surface point. As the user moves a surface point into a new position, the change in the point location is transformed into the corresponding changes in the shape of the cross-sectional curve and the position and orientation of the cross-sectional plane.

When the user selects a surface point $\mathbf{p}$ on the generalized cylinder $S(u, t)$, the point $\mathbf{p}$ is on a cross-sectional curve $C_t(u)$. As we move the point $\mathbf{p}$ into a new location $\hat{\mathbf{p}}$, the shape, position, and orientation of the cross-sectional curve $C_t(u)$ must be modified so that the new surface $\hat{S}(u, t)$ interpolates the point $\hat{\mathbf{p}}$.

The shape of $C_t(u)$ can be modified by changing (1) an affine transformation $A(t)$ of $C_t(u){=}A(t)\cdot C(u)$, and/or (2) the shape of $C(u)$ itself or, in the most general case, by changing (3) the generalized linear extrusion $C(u, t)$. The position and orientation of $C_t(u)$ can be modified by changing the B-spline motion $M(t)$.

To construct a generalized cylinder $S(u, t)$, we need basic components such as $\mathbf{v}(t)$, $\mathbf{r}(t)$, $A(t)$, $C(u)$, and/or $C(u, t)$. When a surface point $\mathbf{p}{=}S(u_0, t_0)$ is relocated to $\hat{\mathbf{p}}$, the corresponding points $\mathbf{v}(t_0)$, $\mathbf{r}(t_0)$, $A(t_0)$, $C(u_0)$, and $C(u_0, t_0)$ must be changed to new locations $\hat{\mathbf{v}}(t_0)$,$\hat{\mathbf{r}}(t_0)$, $\hat{A}(t_0)$, $\hat{C}(u_0)$, and $\hat{C}(u_0, t_0)$ so that the modified surface $\hat{S}(u, t)$ can be constructed from the deformed basic components $\hat{\mathbf{v}}(t)$, $\hat{\mathbf{r}}(t)$, $\hat{A}(t)$, $\hat{C}(u)$, and $\hat{C}(u, t)$. Figure 3 illustrates the deformation procedure of a generalized cylinder through the deformation of each basic component.

The relationship between the generalized cylinder and its basic components is highly nonlinear. Therefore, it is nontrivial to get a closed form solution for the required deformation of each basic component. In this paper, we apply a numerical iteration procedure that is based on target tracking. In each iteration, we modify one basic component so that the surface point $S(u_0, t_0)$ moves a little closer to the target point $\hat{\mathbf{p}}$. The basic component, say $\mathbf{v}(t)$, is deformed so that it interpolates the new location of $\mathbf{v}(t_0)$. The deformation of a basic component can be done by relocating some of its B-spline control points [17].

Note that the modification of one basic component at a time requires solving a simple linear equation. When a surface point moves from $\mathbf{p}$ to $\hat{\mathbf{p}}$, one can interpolate the new surface location by moving $\mathbf{v}(t_0)$ to $\hat{\mathbf{v}}(t_0) = \mathbf{v}(t_0) + \hat{\mathbf{p}} - \mathbf{p}$. In the case of modifying the basic component $A(t)$ or $C(u)$, we need to project the point $\hat{\mathbf{p}}$ into the cross-sectional plane at time t_0 under the assumption that the deformation is carried out on the cross-sectional plane only.

The case of rotation curve $\mathbf{r}(t)$ is slightly more difficult to deal with. We first project $\hat{\mathbf{p}}$ into a point $\tilde{\mathbf{p}}$ on the sphere with center at $\mathbf{v}(t_0)$ and radius $\|\mathbf{p}{-}\mathbf{v}(t_0)\|$. There is a well-defined rotation of this sphere that transforms $\mathbf{p}$ to $\tilde{\mathbf{p}}$. This rotation can be represented by a unit quaternion $\mathbf{q}$. By applying this rotation to $\mathbf{r}(t_0)$, we get a new location $\hat{\mathbf{r}}(t_0)$ of the rotation curve. (See Chang [5] for more details.)

Figure 4 shows the effect of changing the position of $C_t(u)$. As the user moves a surface point to a new location, the translational component $\mathbf{v}(t)$ of the B-spline motion $M(t)$ is modified appropriately. In Fig. 5, the shape and position of $C_t(u)$ are fixed, and only the orientation of $C_t(u)$ changes as the user manipulates a surface point on the curve $C_t(u)$. Figure 6 shows the case of $A(t)$ being a uniform scaling for each cross-sectional curve $C_t(u)$; that is, we have

$$C_t(u){=}A(t)\cdot C(u){=}\alpha(t)\ C(u)$$

for some scalar function $\alpha(t)$. As the user manipulates a surface point, the scaling function $\alpha(t)$ is modified appropriately. In Fig. 7, the deformation is applied to the initial cross-sectional curve $C(u)$ itself rather than to the affine transformation $A(t)$ of the cross-sectional curve $C_t(u){=}A(t)\cdot C(u)$. Note that the shape change in one cross-sectional curve affects all cross-sectional curves. Figure 8 shows the case of restricting the shape change of $C(u)$ to a limited range of t. The result is similar to an interactive surface editing of the generalized linear extrusion $C(u, t)$.

These control schemes can be combined; e.g., we can combine the changes in the scaling and rotation of $C_t(u)$ in the same mode of direct manipulation. In general, we can define the total degrees of freedom (DOF) for the deformation and motion of $C_t(u)$. Given a surface point movement from $\mathbf{p}$ to $\hat{\mathbf{p}}$, we subdivide the shape control in n steps; i.e., from $\mathbf{p}_{i-1} = \mathbf{p} + \frac{i-1}{n}(\hat{\mathbf{p}} - \mathbf{p})$ to $\mathbf{p}_i = \mathbf{p} + \frac{i}{n}(\hat{\mathbf{p}} - \mathbf{p})$,

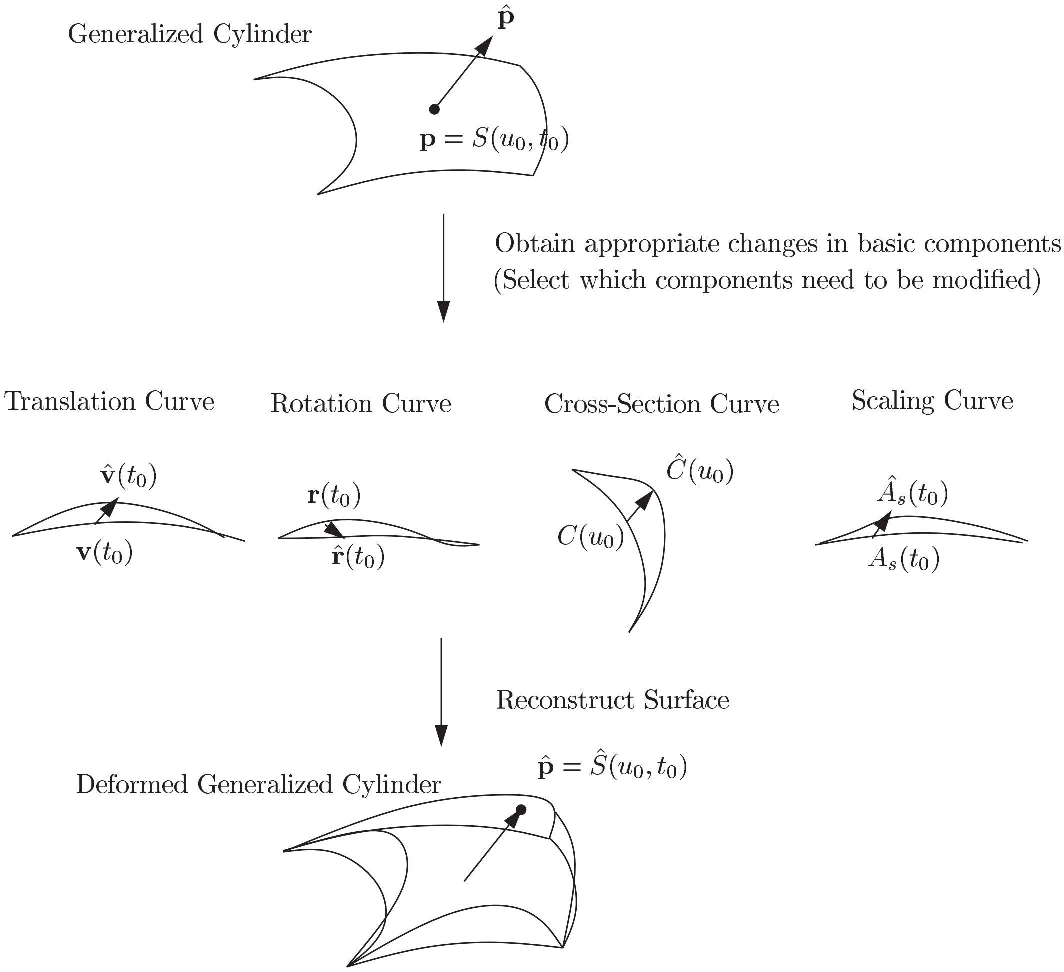
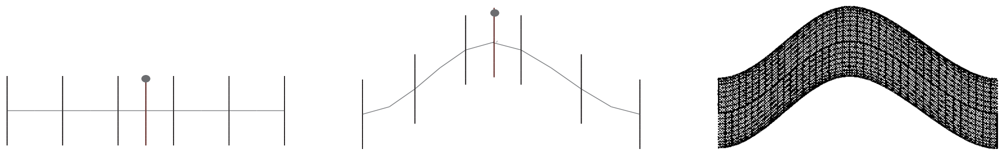
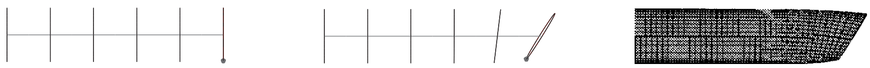


Fig. 3. Deformation procedure for a generalized cylinder

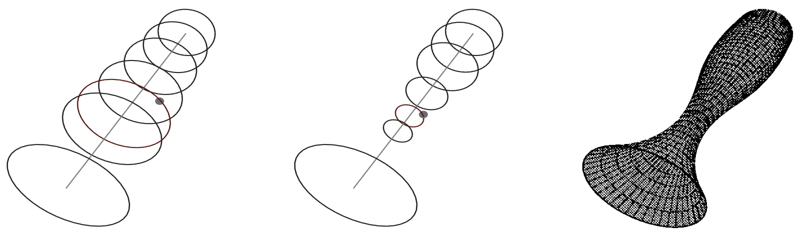
for i=1, …. , n. At each step, we may apply the shape deformation, translation, and rotation of $C_t(u)$ consecutively. The overall numerical procedure is a target tracking in which the surface point $S(u, t)$ keeps track of its target point $\hat{\mathbf{p}}$ that the user manipulates. (Kyung et al. [15] presented a similar target tracking procedure for the control of a virtual camera.) After the shape, position, and orientation of $C_t(u)$ has been modified, the overall shape of $C(u, t)$ and the euclidean motion $M(t)$ are also modified so that the underlying B-spline representations for $C(u, t)$ and $M(t)$ smoothly interpolate the new cross-sectional curve $C_t(u)$. Figure 9 shows a sequence of smooth deformation of a generalized cylinder that demonstrates a target tracking procedure. Figure 10 presents some examples of direct manipulation for a toroidal surface. Figure 11 shows an example of a fish-shaped 3D object created with our system. This object consists of several generalized cylinders represented with trimmed NURBS surfaces. The trimming was done by a NURBS surface editor. The whole design process took only a couple of minutes, starting with circular cylinders and deforming and trimming them to the final fish-shaped object.



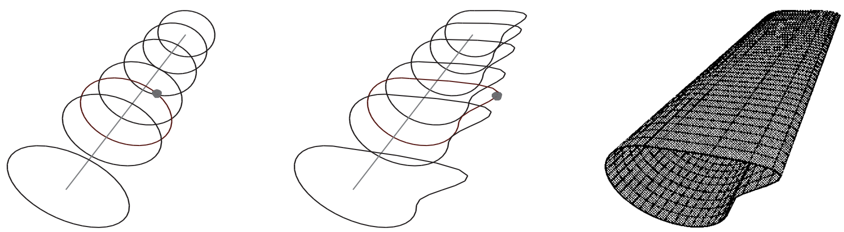
4



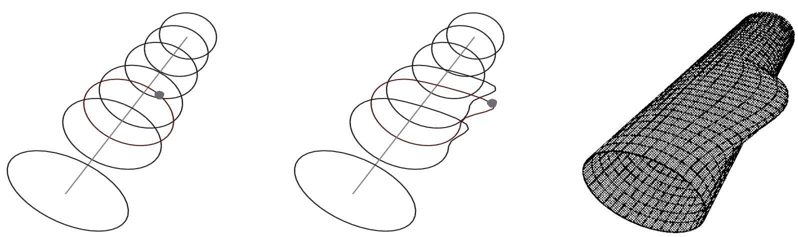
5



6



7



8

Fig. 4. Translation curve deformation

Fig. 5. Rotation curve deformation

Fig. 6. Scale curve deformation

Fig. 7. Cross-sectional curve deformation

Fig. 8. Variable cross-section surface deformation

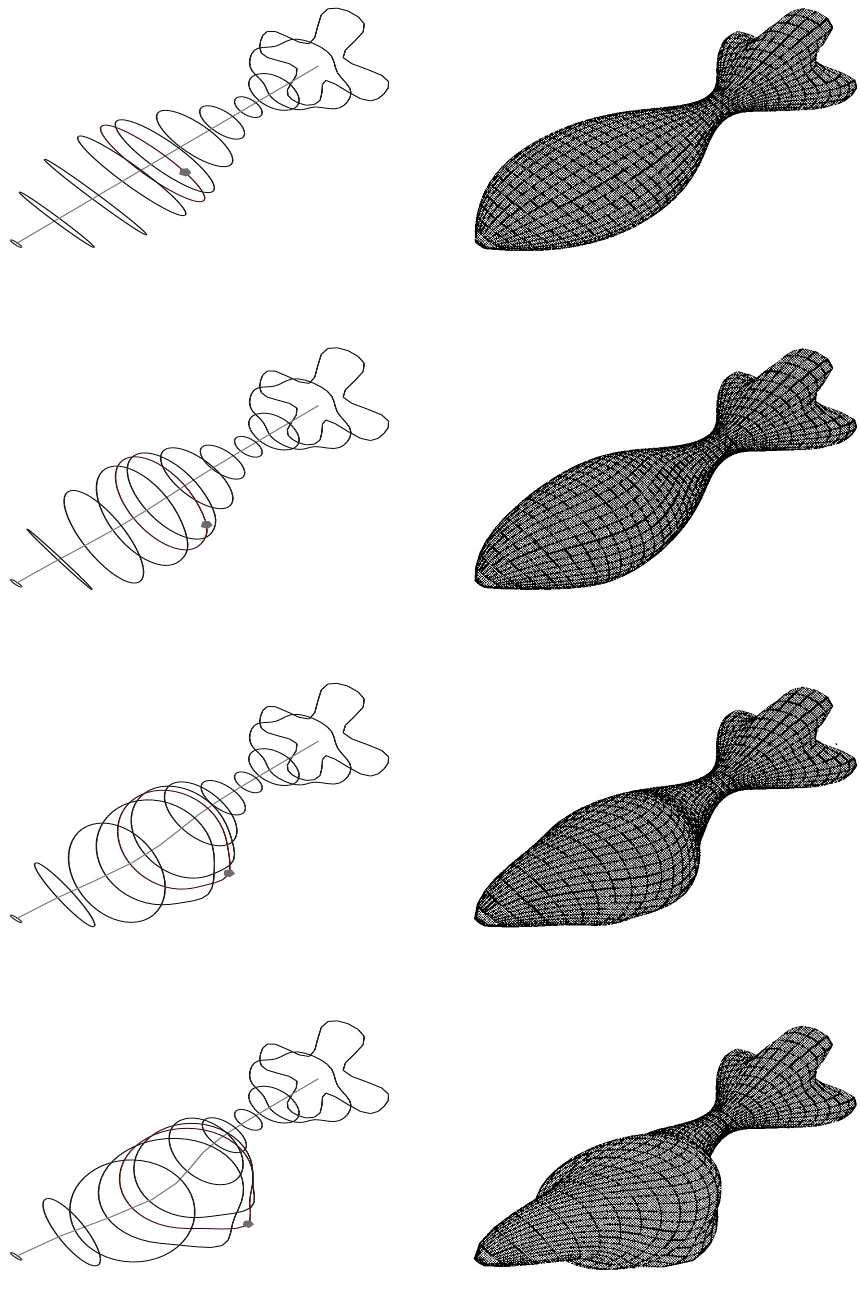
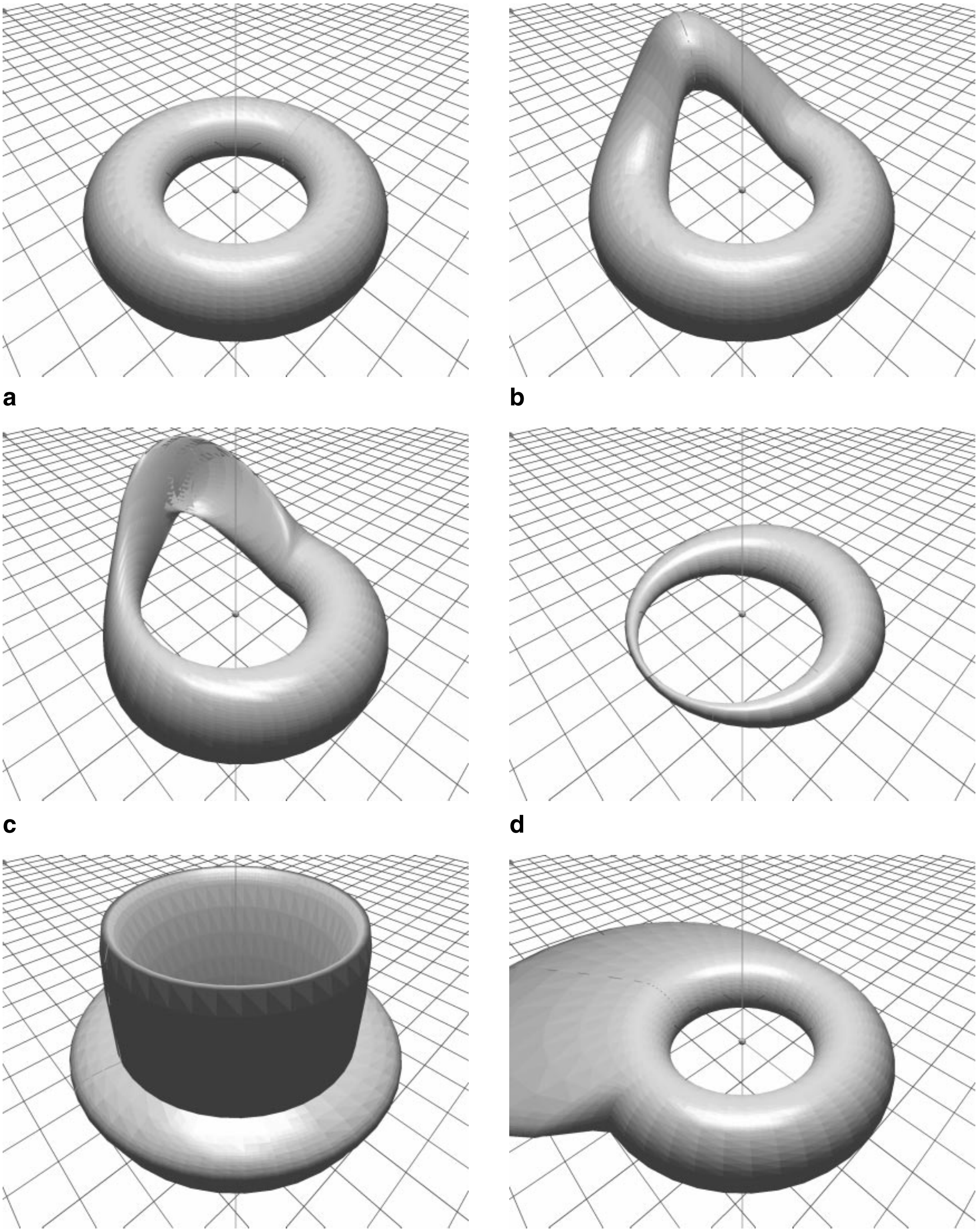


Fig. 9. Smooth deformation based on target tracking



a

b

c

d

e

f

Fig. 10a–f. Direct manipulation of a toroidal surface: **a** an initial torus; **b** translation control; **c** translation and rotation control; **d** scaling control; **e** changing the shape of $C(u)$; **f** changing variable cross-sections

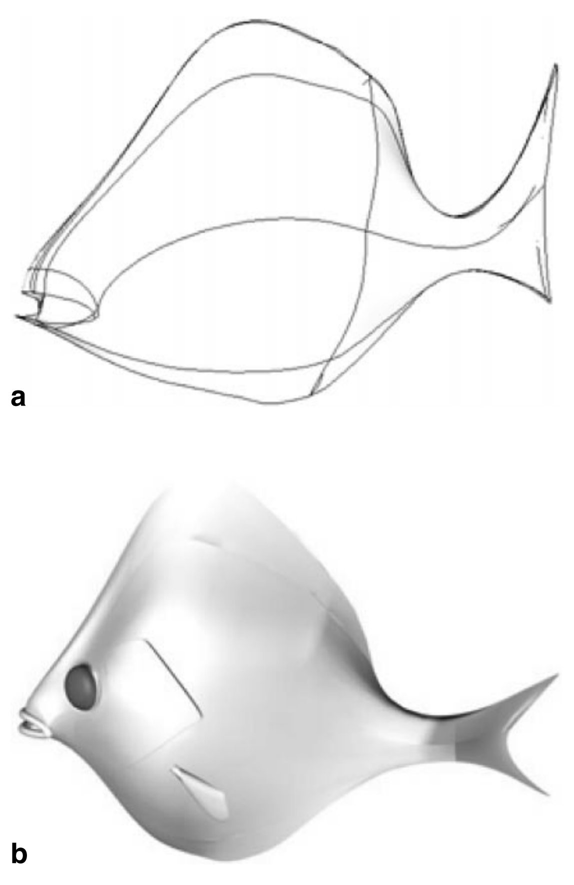


Fig. 11a, b. Designing a fish-shaped 3D object using generalized cylinders: **a** the wireframe of its prototype; **b** a shaded image of the object decorated with extra polygons

6 Conclusion

In this paper, we presented a direct manipulation technique for generalized cylinders. It was non trivial to develop such a technique with conventional methods due to the constraint of restricting the cross-sectional plane to the normal plane of the skeleton curve. To eliminate this difficulty, we redefined the generalized cylinder as a smooth transformation of a cylindrical surface to a sweep surface with tubular shape. When the user directly manipulates a surface point on the generalized cylinder, the basic components are updated so that a reconstructed generalized cylinder interpolates the surface point at a new location. For this purpose, we use an iterative numerical procedure based on a target tracking technique. Our implementation on a Pentium II PC (266 MHz) provides real-time performance.

Acknowledgements. This research was supported in part by ETRI under Grant 1997-318.

References

1. Agin G, Binford T (1976) Representation and description of curved objects. IEEE Trans Comput 25:439–449
2. Bronsvoort W (1992) A surface-scanning algorithm for displaying generalized cylinders. Visual Comput 8:162–170
3. Bronsvoort W, Klok F (1985) Ray tracing for generalized cylinders. ACM Trans Graph 4:291–303
4. Bronsvoort W, Waarts J (1992) A method for converting the surface of a generalized cylinder into B-spline surface. Comput Graph 16:175–178
5. Chang T-I (1998) Generalized cylinder modeling with rational B-spline motion. MS Thesis, Deptartment of Computer Science, POSTECH, South Korea
6. Coquillart S (1987) A control-point-based sweeping technique. IEEE Comput Graph Appl 7:36–45
7. Gleicher M, Witkin A (1991) Differential manipulation. Proceedings of Graphics Interface'91, June 3–7, 1991, Calgary, Alberta, Canada, Morgan Kaufmann, Palo Alto, pp 61–67
8. Gleicher M, Witkin A (1992) Through-the-lens camera control. Comput Graph (Proceedings of SIGGRAPH'92) 26: 331–340
9. Grimm C, Pugmire D, Bloomenthal M, Hughes J, Cohen E (1995) Visual interface for solids modeling. Proceedings of UIST'95, Nov 14–17, 1995, Pittsburgh, Pennsylvania, ACM, New York, pp 51–60
10. Hsu W, Hughes J, Kaufman H (1992) Direct manipulation of free-form deformations. Comput Graph (Proceedings of SIGGRAPH'92) 26:177–184
11. Johnstone J, Williams J (1995) A rational model of the surface swept by a curve. Comput Graph Forum 14:77–88
12. Jüttler B (1995) Spatial rational motions and their application in computer aided geometric design. Math Meth Curves Surfaces. In: Dæhlen M, Lyche T, Schumaker LL (eds), Vanderbilt University Press, La Vergne, Tenn., pp 271–280
13. Jüttler B, Wagner M (1996) Computer-aided design with spatial rational B-spline motions. ASME J Mechan Des 118:193–201
14. Kim M-S, Park E-J, Lee H-Y (1994) Modelling and animation of generalized cylinders with variable radius offset space curves. J Visualization Comput Anim 5:189–207
15. Kyung M-H, Kim M-S, Hong SJ (1996) A new approach to through-the-lens camera control. Graph Models Image Processing 58:262–285
16. Lee S, Wolberg G, Chwa K-Y, Shin S (1996) Image metamorphosis with scattered feature constraints. IEEE Trans Visualization Comput Graph 2:337–354
17. Piegl L, Tiller W (1995) The NURBS book. Springer, Berlin
18. Post F, Klok F (1986) Deformations of sweep objects in solid modeling. Proceedings of EUROGRAPHICS'86, August 19–25, 1986, Lisbon, Portugal, Publisher: North-Holland, Amsterdam, pp 103–114

19. Sederberg T, Parry S (1986) Free-form deformation of solid geometric models. Comput Graph (Proceedings of SIGGRAPH'86) 20:151–160
20. Shani U, Ballard D (1984) Splines as embeddings for generalized cylinders. Comput Vision Graph Image Processing 27:129–156
21. Shoemake K (1985) Animating rotation with quaternion curves. Comput Graph (Proceedings of SIGGRAPH'85) 19:245–254
22. Shinagawa Y, Kunii T (1991) The homotopy model: a generalized method for smooth surface generation from cross sectional data. Visual Comput 7:72–86
23. Snibbe S, Herndon K, Robbins D, Zeleznik R, Conner D, Dam A van (1992) Using deformations to explore 3D widget design. Comput Graph (Proceedings of SIGGRAPH'92) 26:351–352
24. Tai C, Loe K,. Kunii T (1992) Integrated homotopy sweep technique for computer aided geometric design. In: Kunii T (ed) Visual Computing Springer, Tokyo, pp 583–595
25. Wavefront Technologies (1994) 3Design user guide, version 4, November 1994, Santa Barbara

JOO-HAENG LEE is a PhD student in Computer Science at POSTECH, Korea, where he received a BS and an MS in 1994 and 1996, respectively. His research interests include computer graphics and animation.

MYUNG-SOO KIM is an Associate Professor in Computer Science at POSTECH, Korea. He received a BS and an MS in Mathematics from Seoul National University, Korea, in 1980 and 1982, respectively. He also received MS degrees in Applied Mathematics (1985) and Computer Science (1987) from Purdue University, where he completed his PhD in Computer Science in 1988. His research interests include geometric modeling, computer graphics, and computer animation.

TAE-ICK CHANG is a PhD student in Computer Science at POSTECH, Korea, where he received a BS and an MS in 1996 and 1998, respectively. His research interests include computer graphics and animation._

SUNG JE HONG is the Head of Computer Science at POSTECH, Korea. He received a BS in Electronics Engineering from Seoul National University, Korea, in 1973. He also received an MS in Computer Science from Iowa State University in 1979, and a PhD in Computer Science from the University of Illinois at Urbana-Champaign in 1983. From 1983 to 1989, he was a Research Staff Member at Corporate Research and Development, General Electric. His research interests include VLSI architectures, parallel processing, and computer graphics.