A Matrix-Based Approach to Reconstruction of 3D Objects from Three Orthographic Views

Shi-Xia Liu[a], Shi-Min Hu[a], Chiew-Lan Tai[b] and Jia-Guang Sun[a]
[a]*Department of Computer Science and Technology, Tsinghua University, Beijing 100084*
{liusx, shimin}@ncc.cs.tsinghua.edu.cn, sunjg@tsinghua.edu.cn
[b]*Department of Computer Science, Hong Kong University of Science and Technology, HongKong*
taicl@cs.ust.hk

Abstract

In this paper we present a matrix-based technique for reconstructing solids with quadric surface from three orthographic views. First, the relationship between a conic and its orthographic projections is developed using matrix theory. We then address the problem of finding the theoretical minimum number of views that are necessary for reconstructing an object with quadric surfaces. Next, we reconstruct the conic edges by finding their matrix representations in space. This effectively constructs a wire-frame model corresponding to the three orthographic views. Finally, volume information is searched within the wire-frame model to form the final solids. The novelty of our algorithm is in the use of the matrix representation of conics to assist in the 3D reconstruction, which increases both the efficiency and the reliability of the proposed approach.

Keywords: Engineering drawings, matrix representation, wire-frame, reconstruction

1. Introduction

Three-view engineering drawings have been widely used to illustrate product design; they play an essential role in traditional engineering. Most existing products are presented in the form of engineering drawings in blue print or in 2D CAD systems. To manufacture these products, the corresponding 3D objects have to be reconstructed from 2D engineering drawings.

A number of techniques have been proposed for reconstructing 3D objects from orthographic engineering drawings [12,18]. Among the various approaches [2, 10,11,14,16,17], the B-rep oriented methods are the most studied so far [14,16]. The B-rep oriented method is a bottom up approach. It consists of the following steps:

- Generate 3D candidate vertices from 2D vertices in each view.
- Generate 3D candidate edges from 3D candidate vertices.
- Construct 3D candidate faces from 3D candidate edges on the same surface.
- Construct 3D objects from candidate faces.

Wesley and Markowsky [10,18] provide a comprehensive algorithm to reconstruct polyhedral solids from three orthographic views. Later, Sakurai [15], Lequette [8], Shin [16], Dutta[3], and Liu[9] proposed algorithms that can handle limited quadric surface solids. However, researchers have also pointed out that the range of objects that can be reconstructed using the B-rep oriented method is limited [7,9]. To remedy this problem, Kuo [7] suggested applying a technique based on the fact that any five distinct points define a unique conic. However, a major problem remains—the high computation requirement. In addition, the axes of 2D conic sections are restricted to be parallel to one of the coordinate axes.

In this paper, we present a new approach for reconstructing 3D objects based on some important properties of the matrix representation of conics. The utilization of these properties leads to a simple and effective computation scheme for reconstruction. The proposed approach can handle a wider variety of manifold objects with curved surfaces than existing methods and imposes no restrictions on the axes of the conics. Our approach consists of the following three stages:

1. Preprocessing the engineering drawings to prepare for 3D reconstruction;
   The input data are checked for validity and each view is separated and identified from the input engineering drawing. Some aided information needed by the reconstruction process is added.
2. Analyzing the relationship between 3D conic edges and their projections using matrix theory to construct a wire-frame model;
   This stage presents details of constructing the wire-frame model from three orthographic views, which includes creating 3D candidate vertices and edges. Detailed techniques for generating 3D edges are

0-7695-0868-5/00 $10.00 © 2000 IEEE

explained.

3. Reconstructing the 3D object by searching for the faces and solids within the wire-frame model.
   At this stage, all candidate faces are created by searching all fundamental edge loops in the wire-frame. Depth information in the engineering drawings is completely utilized to remove pseudo edges and faces. At last, all true faces are assembled to form 3D objects.

Section 2 presents the matrix representation of conics. Section 3 discusses the number of 2D views required to generate a 3D object with quadric surface. Algorithms for reconstructing 3D objects are presented in Section 4. In section 5, we show several experimental examples that are generated using the reconstruction algorithm from section 4. Section 6 offers conclusion and remarks.

2. Matrix representation of conics

In this section, we develop the tie between conics and symmetric matrices, which leads to a simple linear relationship between them.

We begin with describing the general equation for a conic. A conic in a plane can be described by an algebraic expression

$$a_{11}x^2 + 2a_{12}xy + a_{22}y^2 + 2a_{13}x + 2a_{23}y + a_{33} = 0 \qquad (1)$$

Using matrix notation, we can write the equation as

$$f(\mathbf{u}) = \mathbf{u}^T\mathbf{A}\mathbf{u} = 0 \qquad (2)$$

where $\mathbf{u} = [x, y, 1]^T$, $\mathbf{u}^T$ is the transpose of $\mathbf{u}$, and $\mathbf{A}$ is the symmetric 3×3 matrix given by

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{12} & a_{22} & a_{23} \\ a_{13} & a_{23} & a_{33} \end{bmatrix}$$

The matrix $\mathbf{A}$ is required to be non-singular for $f(\mathbf{u})$ to be a non-degenerate conic.

This representation of a conic as a symmetric matrix provides us a simple characterization of its orthographic projection. For example, any linear transformation of the form $\mathbf{u} = \mathbf{P}\mathbf{u}'$ is reflected by the change of the corresponding representing matrix $\mathbf{A}$ to $\mathbf{A}_P$ given by

$$\mathbf{A}_P = \mathbf{P}^T\mathbf{A}\mathbf{P}$$

Therefore the relationship between a conic and its orthographic projection in this representation is linear, unlike the standard parameterization that is based on geometric parameters. For convenience, we will often not distinguish between the conic $f(\mathbf{u})$ and the matrix $\mathbf{A}$ that represents it in what follows.

3. Minimum number of views for reconstructing conics

In this section we will discuss the theoretical minimum number of views required for the reconstruction of 3D conics. This number is important since it limits the possibility of exactly reconstructing an object with quadric surfaces in theory.

We first define non-degenerate parallel projection which is useful for subsequent discussions.

Definition 1. Under a parallel projection, if the plane containing the space conic is not perpendicular to the projection plane, then the parallel projection is non-degenerate.

Under non-degenerate parallel projections, all conics are equivalent, i.e., conics are mapped to conics [5]. In addition, the class of conic curves is invariant under such projection. It follows that, under non-degenerate parallel projections, ellipses, parabolas, and hyperbolas in line drawings are projections of ellipses, parabolas, and hyperbolas, respectively, in space [13]. Therefore, if at least one of the projections of a planar curve in space is a conic, then the planar curve must also be a conic. Furthermore, we can determine the class of a space conic from the class of its projection curves by the above property.

Suppose a space conic lies on a plane p and that an object coordinate system c_p is defined such that its x_p and y_p axes lie on the plane p. Let c be a global coordinate system in space. Then the global representation $\mathbf{x} = [x \quad y \quad z \quad 1]^T$ in c of a point $\mathbf{x}_p = [x_p \quad y_p \quad z_p \quad 1]^T$ in c_p can be derived by applying a transformation of the form

$$\mathbf{x} = \mathbf{R}\mathbf{x}_p + \mathbf{t} \qquad (3)$$

where $\mathbf{R}$, $\mathbf{t}$ are rotation and translation, respectively; that is,

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} r_{00} & r_{01} & r_{02} \\ r_{10} & r_{11} & r_{12} \\ r_{20} & r_{21} & r_{22} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ z_p \\ 1 \end{bmatrix} + \begin{bmatrix} t_0 \\ t_1 \\ t_2 \\ 0 \end{bmatrix}$$

For any point on the plane p, we have $z_p = 0$, thus we obtain

$$\mathbf{x} = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} r_{00} & r_{01} & r_{02} \\ r_{10} & r_{11} & r_{12} \\ r_{20} & r_{21} & r_{22} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} t_0 \\ t_1 \\ t_2 \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} r_{00} & r_{01} & t_0 \\ r_{10} & r_{11} & t_1 \\ r_{20} & r_{21} & t_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = \mathbf{P}\mathbf{u}_p \qquad (4)$$

where $\mathbf{u}_p = [x_p, y_p, 1]^T$ and $\mathbf{P}$ is a 4×3 matrix given by

$$\mathbf{P} = \begin{bmatrix} r_{00} & r_{01} & t_0 \\ r_{10} & r_{11} & t_1 \\ r_{20} & r_{21} & t_2 \\ 0 & 0 & 1 \end{bmatrix}$$

The space geometry of the conic is represented in $\mathbf{P}$.

We now consider the relationship between a space conic and its orthographic projections onto some projection planes. Let $\mathbf{c}_i$ ($i = 1,2,\cdots,q$) denote the 2D local coordinate system associated with the ith projection plane, and $\mathbf{u}_i = [x_i, y_i, 1]^T$ denote the homogeneous coordinates of an arbitrary point in this local coordinate system. If $\mathbf{C}_i$ is a 3×4 matrix whose three columns form an orthogonal basis for this projection subspace, then the transformation from a point $\mathbf{x}$ in 3D space to point $\mathbf{u}_i$ in the projection plane is given by the relation

$$\mathbf{u}_i = \mathbf{C}_i\mathbf{x} \qquad (5)$$

Now, the matrix relating $\mathbf{u}_i$ and $\mathbf{u}_p$ is easily obtained from equation (4)

$$\mathbf{u}_i = \mathbf{C}_i\mathbf{P}\mathbf{u}_p = \mathbf{G}_i\mathbf{u}_p \qquad (6)$$

where $\mathbf{G}_i = \mathbf{C}_i\mathbf{P}$.

We are now ready to prove the main result in this section.

Theorem Three distinct orthographic projections are sufficient to uniquely recover a space conic.

Proof. First of all, suppose that none of three projections is degenerate. The degenerate case will be considered later.

Let $\mathbf{A}$ be a space conic that lies on a plane p

$$\mathbf{u}_p^T\mathbf{A}\mathbf{u}_p = 0 \qquad (7)$$

and its projection curves $\mathbf{A}_i$ are represented by

$$\mathbf{u}_i^T\mathbf{A}_i\mathbf{u}_i = 0 \qquad i = 1,2,\cdots,q \qquad (8)$$

Substituting the linear transformation of the form $\mathbf{u}_i = \mathbf{G}_i\mathbf{u}_p$ into equation (8), we obtain

$$\mathbf{u}_p^T\mathbf{G}_i^T\mathbf{A}_i\mathbf{G}_i\mathbf{u}_p = 0 \qquad (9)$$

From equation (7) and (9) it follows that

$$\mathbf{G}_i^T\mathbf{A}_i\mathbf{G}_i = \mathbf{P}^T\mathbf{C}_i^T\mathbf{A}_i\mathbf{C}_i\mathbf{P} = \mathbf{A} \quad i = 1,2,\cdots,q \qquad (10)$$

where $\mathbf{A}$ and $\mathbf{P}$ are unknown matrices. It follows that three orthographic projections yield 18 equations in 15 unknowns. By Bernstein's seminal theorem [1], we can derive that the system of polynomial equations (10) is solvable. Hence, three distinct views are sufficient to uniquely recover a conic if none of its projections is degenerate.

Further, consider the special case where at least one of the orthographic projections is degenerate. If an orthographic projection is degenerate, then the projection of a conic onto this projection plane is a straight line. By the definition of orthographic projections, we can determine the plane on which the conic lies, which is obtained by extruding the straight line along the degenerate projection direction. Since at least one of the projections of the conic is also a conic, we locate the center point (vertex) of the space conic by finding its corresponding points in the other two views. Accordingly, the matrix $\mathbf{P}$ is obtained. Without loss of generality, let the projection in the front view be a conic. To reconstruct the space conic, we then need to solve the equation $\mathbf{P}^T\mathbf{C}_f^T\mathbf{A}_f\mathbf{C}_f\mathbf{P} = \mathbf{A}$ for $\mathbf{A}$, where the subscript f indicates the front view. Hence, if at least one of the orthographic projections is degenerate, three distinct orthogonal projections are also sufficient to identify the space conic. This completes the proof. ■

4. Reconstruction

We propose a wire-frame oriented approach for recovering the 3D objects. The method builds a wire-frame model in preparation for automatic interpretation of faces and solids and uses a B-rep model to represent the reconstructed solid.

4.1. Preprocessing of engineering drawings for 3D reconstruction

The input data consists of three orthographic views of the object. By convention, they are the front view, top view, and side view. Each view is a line drawing in which features that are directly visible are represented by solid lines, while features that are not directly visible are shown as dashed lines. In practice, three-view engineering drawings can be input via an image scanner, and the 2D vertices, edges and line type are segmented by a low-level image processing stage [4]. We assume that such information of the three views is available in this work. Every view in the engineering drawing must be separated and identified. Views are separated by the fact that the projections of three views on the coordinate axes are two separated segments respectively. Therefore, we project each curve onto the two coordinate axes and get four different segments. Two segments are on x-axis, while the other two on y-axis. Next we draw a vertical line between two segments on x-axis and a horizontal line between the other two segments on y-axis. The view plane is divided into four parts by the vertical and horizontal line. Then we separate three views by collecting curves that belong to the same part and identify each view by its location on the view plane.

Each view of the drawing may be considered to be a graph of 2D curves and node points embedded in the plane of the drawings. Two 2D curves may meet only at a node, i.e., the graph is planar and no crossovers are allowed. If two 2D curves are found to intersect, a new node must be created at the intersection point and the 2D curves must be subdivided at that node. When two faces of the 3D object meet tangentially, the corresponding tangent lines must be added to the views. In addition, the center lines and axes are added to the three orthographic views to reconstruct conic edges.

4.2. Reconstruction of wire-frame

In this stage, we construct all the possible 3D vertices and 3D edges that constitutes the wire-frame model by establishing correspondences among the orthographic views. To reconstruct 3D vertices, we use the fundamental idea described in Kuo [7] and Liu [9]. A candidate vertex is created from the 2D vertices in three views. Let $\mathbf{N}_f = (N_f(x), N_f(z))$, $\mathbf{N}_t = (N_t(x), N_t(y))$, and $\mathbf{N}_s = (N_s(y), N_s(z))$ be 2D vertices in the front, top, and side view respectively. If

$$|N_f(x) - N_t(x)| < \varepsilon,\ |N_t(y) - N_s(y)| < \varepsilon,$$
$$|N_t(z) - N_s(z)| < \varepsilon$$

then we know that $\mathbf{N}_f, \mathbf{N}_t$, and $\mathbf{N}_s$ are the corresponding projections in each view of a 3D vertex. Where ε is a tolerance that allows for inexact match. Here, we propose a new method for generating the 3D candidate edges. To generate 3D candidate edges, we must determine the symmetric matrix $\mathbf{A}$ from the following equation

$$\mathbf{G}_i^T \mathbf{A}_i \mathbf{G}_i = \mathbf{P}^T \mathbf{C}_i^T \mathbf{A}_i \mathbf{C}_i \mathbf{P} = \mathbf{A} \qquad i = 1,2,3 \qquad (11)$$

Without loss of generality, suppose that the original three orthographic views are the top, front, and side views. Thus the respective projection matrices are

$$\mathbf{C}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{C}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
$$\mathbf{C}_3 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Thus, from $\mathbf{G}_i = \mathbf{C}_i \mathbf{P}$, we obtain

$$\mathbf{G}_1 = \begin{bmatrix} r_{00} & r_{01} & t_0 \\ r_{10} & r_{11} & t_1 \\ 0 & 0 & 1 \end{bmatrix} \quad \mathbf{G}_2 = \begin{bmatrix} r_{00} & r_{01} & t_0 \\ r_{20} & r_{21} & t_2 \\ 0 & 0 & 1 \end{bmatrix}$$
$$\mathbf{G}_3 = \begin{bmatrix} r_{10} & r_{11} & t_1 \\ r_{20} & r_{21} & t_2 \\ 0 & 0 & 1 \end{bmatrix}$$

In particular, the relationship between $\mathbf{A}_i$ and $\mathbf{A}$ is linear in the element of $\mathbf{A}$. This transformation gives us a simple way to characterize the solvability of the inverse problem of determining $\mathbf{A}$ from equation (11). There exist several methods for solving this system of polynomial equations [6, 21]. Hence, the conic $\mathbf{A}$ represented by its projection $\mathbf{A}_i$ ($i = 1,2,3$) can be described by an algebraic expression

$$a_{11}x^2 + 2a_{12}xy + a_{22}y^2 + a_{13}x + a_{23}y + a_{33} = 0$$

The type of the conic $\mathbf{A}$ is determined by the type of its projection curves since the class of conic curves is invariant under parallel projection [13].

In solid modeling, a common way of representing a conic is by its more intuitive geometric parameters. For example, a central conic is represented by its center coordinates (x_c, y_c), major and minor axes $\{a,b\}$, and orientation θ with respect to the coordinate axes; a parabola is described by the intersection point of the axis of symmetry with the parabola, the focal distance p, and the direction θ of the focal point from that intersection point [20]. It is convenient to convert the algebraic form of a central conic to the geometric representation. For the central conic, the center (x_c, y_c) is the solution of the following system of equations [22]:

$$\begin{cases} a_{11}x_c + a_{12}y_c + \dfrac{1}{2}a_{13} = 0 \\ a_{12}x_c + a_{22}y_c + \dfrac{1}{2}a_{23} = 0 \end{cases} \qquad (12)$$

Hence the geometric representation of the central conic is

$$x_c = \frac{a_{12}a_{23} - a_{22}a_{13}}{2(a_{11}a_{22} - a_{12}^2)}$$
$$y_c = \frac{a_{12}a_{13} - a_{11}a_{23}}{2(a_{11}a_{22} - a_{12}^2)}$$
$$\theta = \frac{1}{2}\tan^{-1}\frac{2a_{12}}{a_{11} - a_{22}} \qquad (13)$$
$$\{a,b\} = \begin{cases} \{\sqrt{-a_{33}'/a_{11}'}, \sqrt{-a_{33}'/a_{22}'}\} & \textit{for an ellipse} \\ \{\sqrt{-a_{33}'/a_{11}'}, \sqrt{a_{33}'/a_{22}'}\} & \textit{for a hyperbola} \end{cases}$$

where

$$\begin{cases} a_{11}' = \dfrac{a_{11} + a_{22}}{2} + \dfrac{a_{11} - a_{22}}{2}\cos 2\theta + a_{12}\sin 2\theta \\ a_{22}' = \dfrac{a_{11} + a_{22}}{2} - \dfrac{a_{11} - a_{22}}{2}\cos 2\theta - a_{12}\sin 2\theta \\ a_{33}' = \dfrac{1}{2}a_{13}x_c + \dfrac{1}{2}a_{23}y_c + a_{33} \end{cases}$$

Moreover, for a parabola, we have

$$I_2 = a_{11}a_{22} - a_{12}^2 = 0 \qquad (14)$$

From equation (14), we can conclude that a_{11} and a_{22} have the same sign. Without loss of generality, suppose

$a_{11} > 0$ and $a_{22} > 0$. Let $\alpha^2 = a_{11}$ and $\beta^2 = a_{22}$, thus $a_{12} = \alpha\beta$. The signs of α and β are determined by a_{12}. If $a_{12} > 0$, then $\alpha > 0$ and $\beta > 0$; otherwise, let $\alpha > 0$ and $\beta < 0$. With this notation, we may represent our original equation (1) equivalently as:

$$(\alpha x + \beta y)^2 + a_{13}x + a_{23}y + a_{33} = 0 \tag{15}$$

To reduce this equation to its standard form, rotate the coordinate axes through θ about the origin, where θ satisfies

$$\tan\theta = -\alpha/\beta \tag{16}$$

Accordingly,

$$\sin\theta = -\frac{\alpha}{\sqrt{\alpha^2+\beta^2}}, \cos\theta = \frac{\beta}{\sqrt{\alpha^2+\beta^2}} \tag{17}$$

The rotation transformation is therefore defined by

$$\begin{cases} x = x'\cos\theta - y'\sin\theta = \dfrac{\beta x' + \alpha y'}{\sqrt{\alpha^2+\beta^2}} \\ y = x'\sin\theta + y'\cos\theta = \dfrac{-\alpha x' + \beta y'}{\sqrt{\alpha^2+\beta^2}} \end{cases} \tag{18}$$

Thus, the transformation in (18) maps the representation (15) to

$$(\alpha^2+\beta^2)y'^2 - \frac{2(\beta a_{13} - \alpha a_{23})}{\sqrt{\alpha^2+\beta^2}}x' - \frac{2(\alpha a_{13} + \beta a_{23})}{\sqrt{\alpha^2+\beta^2}}y' + a_{33} = 0 \tag{19}$$

From the above derivation, we then obtain the geometric representation of the parabola

$$\begin{aligned} x_c &= \frac{\beta a_{13} + \alpha a_{23}}{2(\alpha^2+\beta)^{3/2}} - \frac{a_{33}\sqrt{\alpha^2+\beta^2}}{2(\beta a_{13} + \alpha a_{23})} \\ y_c &= -\frac{\beta a_{13} + \alpha a_{23}}{(\alpha^2+\beta)^{3/2}} \\ \theta &= \tan^{-1}(-\alpha/\beta) \\ p &= \frac{\alpha a_{23} - \beta a_{13}}{(\alpha^2+\beta)^{3/2}} \end{aligned} \tag{20}$$

Finally, we transform every geometric parameter of the reconstructed conic into the global space by left multiplying it by a 4×3 matrix $\mathbf{P}$.

4.3. Reconstructing 3D solids

This stage begins with the pseudo wire-frame model generated in the previous stage. First, all candidate faces are found within the wire-frame, using a *maximum turning angle* method. Next, pseudo faces that could be generated from back projection are detected and deleted using the definition of manifold and the Moebius rule. Finally, all true faces are assembled to form an oriented 3D object. We describe each of these steps in the following subsections.

4.3.1. Generation of surfaces and faces. We now find the surfaces that contain the faces of the solid. We use the term surface to refer to an unbounded face, and the term face to refer to the region of a surface that is bounded by an edge loop. All possible surfaces can be constructed from the information of two edges sharing a common vertex. The type of these two edges and the relationship between them determine the type and the features of the generated surface. For example, a planar surface can be built from two coplanar edges sharing a candidate vertex; a line (conic) and a conic that are not coplanar with each other constitute a quadric surface. Once a surface is identified, our algorithm searches all the edges on the surface for tracing the corresponding face. This exploration is a depth first search, which begins with the vertices on the surface. For each vertex on the surface, all the edges connected to the vertex are checked; those that are on the surface are recorded, and their vertices are added to a stack of vertices for deeper explorations.

For each surface sought in the previous step, we must find all its edge loops from the wire-frame model to form the corresponding faces. The method we use is called maximum turning method. Before describing the maximum turning method, we introduce the definition of turning angle, which are useful for the following discussions. The turning angle is the angle by which the tangent vector of a curved edge is to be rotated about the common vertex to go from the tangent vector of the current curved edge to the tangent vector of its adjacent curved edge. Counterclockwise rotations are assumed to be positive.

To find all the fundamental edge loops within the wire-frame, we first sort the edges at each vertex in descending order with respect to the turning angle from the selected edge, then traverse the resulting graph of oriented edges in this order to find all the circuits on the surface.

4.3.2. Searching for solids. The last step is to find all the solids that can be constructed from the set of faces and to compare them with the original projections. Among all the candidate faces that are generated in the previous step, some really lie on the boundary of a solid; these are real faces, the others are called pseudo faces. Hence, the challenge in this last step is to find a group of faces that make up of a real solid and resolve ambiguities.

The determination of these sets of faces is based on the Moebius rule and the definition of manifolds. The Moebius rule says that each edge in a manifold solid belongs to two faces and the orientation of the edge is inverted by each face [8]. We implement this idea by traversing the whole graph of possible face configurations, backtracking when necessary, to find all possible solids.

Initially, we use depth information including solid curves and dashed curves to remove some false faces. Faces that obstruct solid edges are pathological and should be removed. On the other hand, if the projection of a 3D candidate edges is a dashed curve in one view, there should exist at least one face that obstructs the edges in the projecting direction. Otherwise the 3D candidate edge is pseudo and must be deleted. Then, the remaining pseudo faces in the wire-frame model are completely removed by a divide-and-conquer approach based on the Moebius rule and the definition of manifold. The final solid is made up of a consistent set of real faces, which can be determined and oriented with respect to one another without violating the Moebius rule.

In the searching process, we compute an integer for each 2D curve in the views, which is the number of 3D edges that project to that 2D curve. When the solid is built, some edges may be removed from the wire-frame model. If so, the number of edges will decrease. In order to match the original views, we must keep the integer associated with each 2D curve positive.

5. Implementation

In this section we demonstrate the use of our algorithm by reconstructing three objects. The first two examples are engineering drawings that correspond to unique objects, while the third example involves an ambiguous case that has multiple solutions.

Planar, conical and cylindrical surfaces play an important role in the description of manufactured mechanical parts. A reconstructed mechanical part with planar, cylindrical, and conical faces is shown in Figure 1. Figure 2 shows a unique solution constructed from a three-view engineering drawing with straight lines, circular arcs, hyperbola, and elliptical arcs. An ambiguous case that has two possible solutions is shown in Figure 3.

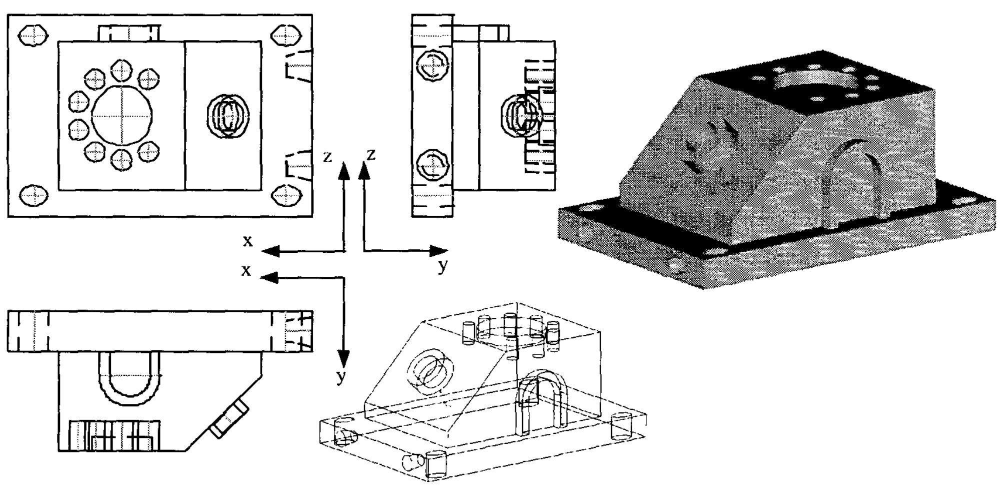

Figure 1. A mechanical part reconstructed from a three-view engineering drawing, with planar, cylindrical, and conical faces.

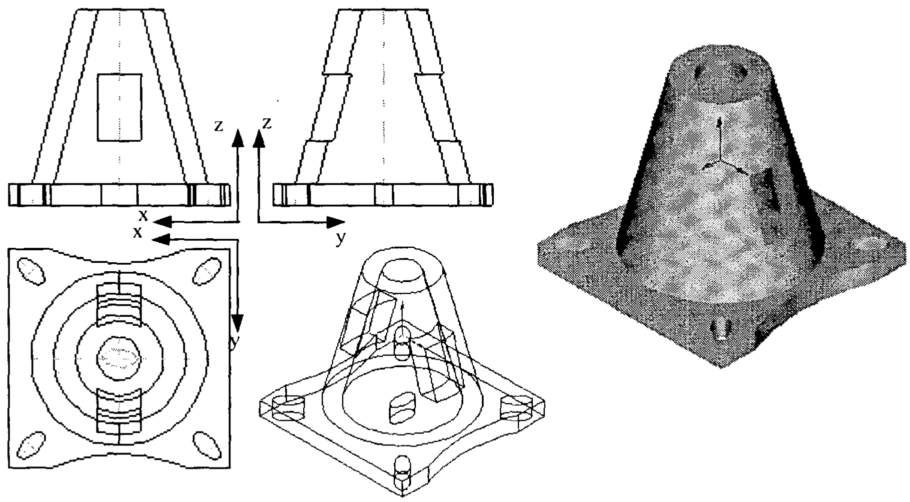

Figure 2. A mechanical part reconstructed from a three-view engineering drawing with straight lines, circular, hyperbola, and elliptical arcs.

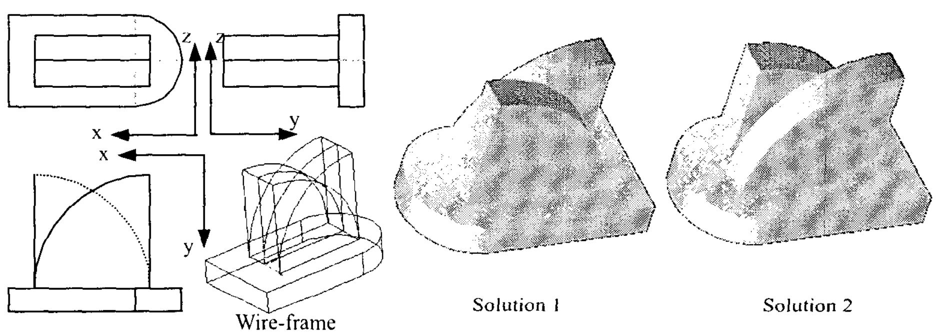

Figure 3. An ambiguous drawing and the reconstructed wire-frame model (left) and two solutions (right)

6. Conclusion

Reconstructing solid objects from orthographic projections is an important research topic in CAD/CAM. In this paper, we described an approach for generating solids with quadric surface. The algorithm takes advantage of the matrix representation of conic sections, which is efficient because no 2D curves need to be segmented for generating conics in space as it is required in Kuo's method [7]. In addition, the proposed scheme has no restriction on the axis of the conics and quadric surfaces.

Future work includes increasing the efficiency of the reconstruction process by developing more systematic and reliable techniques to cut down the search space and the processing time. In addition, we will make full use of the annotation in engineering drawings to extend the range of objects to be constructed.

Acknowledgements

This work was supported by the National Nature Science Foundation of China (Project Number 69902004), National '863' R&D high-tech Program (Project Number 863-511-842-004) of China, and Basic Research Foundation of Information School at Tsinghua University

(Project Number 985-14-14). Special thanks go to the K.C. Wong Education Foundation for its financial support. We would like to thank Professor Yu-Jian Chen and Professor Ge-Zhi Zhao for their constant encouragement and advice. We are grateful to Hui Zhang for her careful reading and constructive criticism.

References

[1] DN Bernstein, The number of Roots of a System of Equations, Functional Analysis and its Applications, 1975, 9(3), pp. 1-4.

[2] Z Chen and DB Perng, Automatic reconstruction of 3D solid objects from 2D orthographic views, Pattern Recognition, 1988, 21(5), pp. 439-449.

[3] D Dutta and Y L Srinivas, Reconstruction of curved solids from two polygonal orhtographic views, computer-aided design, 1992, 24(3), pp. 149-159.

[4] D.Dori and K Tombre, From engineering drawings to 3D CAD models: are we ready now?. Computer Aided Design, 1995, 27(4), pp. 243-254.

[5] G Farin, Curves and Surfaces, Computer Aided Geometric Design, 4rd edn. Academic Press, New York, 1997.

[6] J.C. Faugeere, P. Gianni, D. Lazard, and T. Mora, Efficient Computation of Zero-Dimensional Grobner Bases by Change of Ordering, Journal of Symbolic Computation, 1996, 16, pp. 329 -344.

[7] MH Kuo, Reconstruction of quadric surface solids from three-view engineering drawings, Computer Aided Design, 1998, 30(7), pp. 517-527.

[8] R Lequette, Automatic construction of curvilinear solids from wireframe views, Computer Aided Design, 1988, 20(4), pp. 171-179.

[9] SX Liu, SM Hu, and JG Sun. Reconstruction of Curvilinear Objects from Three Orthographic Views. In CAD/CG'99, , 1999, pp. 1040-1044.

[10] G Markowsky, MA, Wesley, Fleshing Out Wire Frames, IBM J. RES.DEVELOP, 1980, 24(5), pp. 582-597.

[11] S Meeran and MJ Pratt, Automated feature recognition from 2D drawings. Computer-Aided design, 1993, 25(1), pp. 7-17.

[12] IV Nagendra and UG Gujar, 3-D objects from 2-D orthographic views-a survey, Computer & Graphics, 1988, 12(1), pp. 111-114.

[13] VS Nalwa, Line-Drawing Interpretation: Straight Lines and Conic Sections. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1988, 10(4), pp. 514-529.

[14] BS Oh and CH Kim, Systematic reconstruction of 3D curvilinear objects from two-view drawings, Computers & Graphics, 1999, 23(3), pp. 343-352.

[15] H Sakurai and DC Gossard, Solid model input through orthographic views. In SIGGRAPH 83 proceedings, 1983, pp. 243-252.

[16] B Shin and Y Shin, Fast 3D solid model reconstruction from orthographic views. Computer Aided Design, 1998, 30(1), pp. 63-76.

[17] SP Shum, WS Lau, MF Yuen, and KM Yu, Solid reconstruction from orthographic opaque views using incremental extrusion, Computer & Graphics, 1997, 21(6), pp. 787-800.

[18] W Wang and GG Grinstein, A survey of 3D solid reconstruction from 2D projection line drawings. Computer Graphics forum, 1993, 12(2), pp. 137-158.

[19] MA Wesley, G Markowsky, Fleshing Out Projection. IBM J. RES.DEVELOP, 1981, 25(6), pp. 934-954.

[20] PR Wilson, Conic Representations for Shape Description. IEEE CG&A, 1986,7(4), pp. 23-30.

[21] W. Wu, Basic Principles of Mechanical Theorem Proving in Elementary Geometries. Journal of Automated Reasoning, 1986, 2, pp. 221-252.

[22] Dani Daniel Zwillinger, Standard mathematical tables and formulae, CAC Press, New York, 1996.