PERGAMON

Computers & Graphics 27 (2003) 581–592

COMPUTERS & GRAPHICS

www.elsevier.com/locate/cag

Technical section

Progressive reconstruction of 3D objects from a single free-hand line drawing

Beom-Soo Oh, Chang-Hun Kim*

Department of Computer Science and Engineering, Korea University, 1, 5-ka, Anam-Dong, SeungBuk-Ku, 136-701, Seoul, South Korea

Accepted 15 April 2003

Abstract

This paper presents a progressive algorithm by which we can narrow down the searching domain for face identification and reconstruct various 3D objects from a single freehand line sketch drawing in short time. The sketch drawings served as an input for the reconstruction process in this paper are obtained from an inaccurate freehand sketch of a 3D wireframe object, which is the edge–vertex graph without any removal of hidden lines. The algorithm is executed through two stages. First, at the face identification stage, every possible faces are generated from the sketch and they are classified into implausible faces, basis faces and minimal faces by means of the geometrical and topological constraints in order to reduce the searching space. The proposed algorithm searches minimal space of the faces so that the actual faces of the given object could be identified in short time. Second, at the object reconstruction stage, we calculate a 3D structure progressively by optimizing the vertices coordinates of the object in accordance with the sketching order of faces. This progressive method reconstructs the most plausible 3D object in a short time by applying the 3D constraints that are derived from the relationship between an object and its sketch drawing in the optimization process. In addition, this method makes it possible for designers to change their viewpoints while sketching. In this paper, we are describing the progressive reconstruction algorithm, and giving the results of the implementation.

1. Introduction

At the conceptual stage of designing products, designers usually draw their basic ideas of the mechanical parts on the paper with a pencil. This is a natural tendency because interfacing with a computer is not an appropriate method enough to convey their basic idea of products.

Lipson and Shpitalni [1] introduced a conceptual design based on the 3D object reconstruction from a single 2D sketch drawing, which is an inaccurate projection of a 3D wireframe model in the conceptual stage. It is useful in inputting geometrical information to represent 3D information by using a line drawing. Once the 3D model is obtained, it is possible to manipulate or modify it, and more details can be added in order to get a more precise model. This approach provides designers with the methods to convey their ideas to the CAD system.

However, the systems that process the sketches depicting 3D objects are very rare. The primary difficulty in reconstructing is that an inverse projection should be performed from the sketch drawing plane to the 3D space. This process is mathematically indeterminate. Also, the 2D sketch drawing is an inaccurate projection of the 3D wireframe model. In contrast, human beings are able to perform this process with little

*Corresponding author. Tel.: +82-23290-3199; fax: +82-2953-0771.

E-mail addresses: obs@cgvr.korea.ac.kr (B.-S. Oh), chkim@korea.ac.kr (C.-H. Kim).

0097-8493/03/$ - see front matter © 2003 Elsevier Ltd. All rights reserved.
doi:10.1016/S0097-8493(03)00100-6

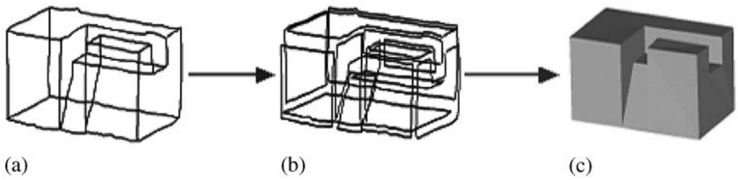

Fig. 1. Conceptual design with sketching: (a) Sketch drawing, (b) 2D identified faces, and (c) 3D object.

difficulty. It is exactly this ability that we intend to emulate in this paper.

There are two stages in reconstructing the 3D geometrical information from a 2D sketch drawing. The concept of the reconstruction system is illustrated in Fig. 1. At the face identification stage, it is required to generate the edge–vertex graph by analyzing strokes in the sketch drawing [2]. Then, we should identify the 2D actual faces (edge circuits) of the 3D object depicted as a 2D edge-vertex graph [3,4]. At the object reconstruction stage, it is needed to restore the 3D object by means of the geometrical/topological relationship among the identified 2D faces, edges and vertices [1]. It means that the depth values of the individual vertices in the 3D object should be restored.

In conventional methods, potential faces are classified into implausible faces and minimal ones so that the identification of actual faces could be done through the combinatorial search of minimal faces. However, they require a number of combinatorial searches of tremendous minimal faces because a single 2D sketch drawing corresponds to an infinite number of 3D objects. In addition, the existing methods reconstruct the 3D structure from the off-line sketch drawing by optimizing the objective function derived from the image regularities of the sketch drawing. Thus, those methods are difficult to be applied to the on-line sketch drawing which allows the change of viewpoints. Moreover, it takes much time to process them because they use the global relationship among all the 2D entities.

In this paper, we describe a progressive algorithm to identify the 2D faces in a short time by classifying potential faces as well as to reconstruct 3D objects in a short time in accordance with the sketch order of faces. Fig. 2 shows an overview of this progressive 3D reconstruction.

At the face identification stage, the algorithm (i) generates potential faces. (ii) It reduces the searching domain of minimal faces by classifying the potential faces into the basic, implausible and undetermined minimal faces. Basic faces can be determined to be actual faces without any searching process, and the implausible faces cannot be actual faces. (iii) It identifies the 2D actual faces of the given object by searching the reduced minimal faces only.

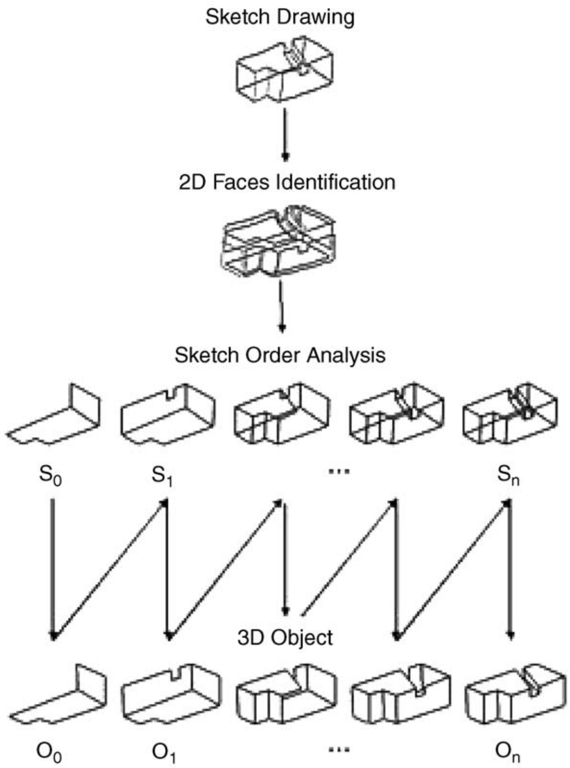

Fig. 2. Overview of the progressive 3D reconstruction.

At the object reconstruction stage, the algorithm (i) analyzes the optimal sketching order of faces in accordance with the way of human perception. (ii) derives 3D constraints from the relationship between partial object and sketch drawing. (iii) refines the geometrical information of the rough partial object into that of the detailed complete object by adding faces to the partial object.

- *Fast face identification*: The algorithm efficiently classifies the potential faces into the implausible, the basic, and the minimal faces with the constraints generated by the gab between the sketch drawing and the actual object. Then, it identifies the 2D actual faces of the object in a short time by reducing the searching domain of the minimal faces.

- *Fast object reconstruction*: In the off-line sketch drawing, the algorithm estimates the optimal sketching order of faces. Then, it progressively derives the objective function from the geometric relationship between the sketch drawing and the object. It reconstructs the 3D object in a short time by the progressive calculation of the 3D structure with the objective function.
- *Dynamic viewpoints*: The algorithm allows the change in viewpoints with no need to recalculate the relationship among the 2D entities on the sketch drawing because it only uses the 3D relationship between the sketch drawing and the object.

2. Related works

Since a sketch drawing does not have enough dimensions, additional methods should be used to extract the missing dimensions. Wang et al. [5] and Lipson [6] conducted studies in detail on the reconstruction of 3D object from a single projection.

Huffman [7] made qualitative studies on the single view scenes with hidden lines removed and others, e.g., Kanade [8] and Sugihara [9] made quantitative studies, resulting in various line labeling schemes based on the junction libraries. According to these techniques, all segments of the line in a drawing are labeled as a concave, a convex junction of faces or as an edge of the occluded face. A set of consistent line labels are searched in the library of possible junction configurations.

Marill [10] suggested an optimization-based reconstruction in which the depth value of vertices is calculated by means of the minimum standard deviation of angles (MSDA). His method expands the flat sketch into 3D objects by minimizing the target objective function derived from MSDA at junctions. However, this technique yields implausible reconstruction of non-orthogonal models in experiments.

Braid and Wang [11] improved Marill's approach by elevating the optimization procedure using conjugate gradients.

Leclerc and Fiscler [12] identified all the non-self-intersecting closed circuits of edges, but they cannot be applied to the concave faces as well as to the ambiguous sketch drawings. Moreover, in spite of their revision of Marill's method by using face planarity, their method limited the type of objects.

Lamb [13] implemented an interactive reconstruction system based on the line labeling. His algorithm enables users to avoid ambiguity, and it identifies the principal axis and symmetry of faces.

Shpitalni and Lipson [4] efficiently identified actual faces by using the maximum rank equation and the face adjacency theory. However, their method requires large searching domain of the minimal faces including a number of implausible faces.

Lipson and Shpitalni [1] reconstructed a 3D object with flat and cylindrical faces based on the optimization process that formalizes various image regularities. Experiments performed using their technique yielded a plausible reconstruction for various objects including manifold or non-manifold.

Although many methods have been proposed, it is still difficult to develop a practical reconstruction system for the following reasons: (i) the existing methods require large-scale combinatorial searches at the face identification stage, in that a 2D sketch drawing corresponds to multiple 3D objects and it contains a number of potential faces. (ii) the optimizing processing is slow at the object reconstruction stage, in that the existing methods optimize the objective function that is derived from the 2D image regularities. In addition, their methods are difficult to be applied to the drawings with changing viewpoints such as on-line sketch drawing.

3. Problem statements

A sketch drawing is a single 2D projection of a 3D object that is manifold or non-manifold. The goal of the sketch reconstruction is to restore the original 3D object using the information derived from the projection only.

3.1. Assumptions

- *Sketch drawing*: The input projection presents the wireframe model of a general object, and it consists of only a single 2D line drawing which is a graph of connected entities.
- *On-line/off-line sketch drawings*: The on-line sketch drawing has its own sketch order of faces as well as dynamic viewpoints. In contrast, the off-line sketch drawing which has a fixed viewpoint does not have any sketching order with a fixed viewpoint.
- *3D object*: It can be manifold or non-manifold object depicted in the sketch drawing. However, the reconstruction system does not provide any 3D information such as the type of 3D objects, viewpoints, or the relative position in the viewpoint.
- *General viewpoint*: The projection is drawn from a general viewpoint from which all edges and vertices are shown. In other words, any edge or vertex of an object should not coincide or accidentally join in the projection.

3.2. Requirements

- *The most plausible object*: A 2D sketch might project an infinite number of possible 3D objects. The reconstruction must produce the most plausible 3D object in a given projection, which human observers are most likely to select.

- *Sketch order analysis*: In the off-line sketch drawing, there can be many orders of drawing. The algorithm in this paper should analyze the optimal sketching order to reconstruct the most plausible object.
- *Dynamic viewpoint*: Once part of an object has been reconstructed, it is possible to change the visual point on the depicted object in three dimensions, and to continue to sketch more details from different viewpoints. It makes users to be able to sketch a model gradually and part by part from the viewpoint they prefer.

4. Face identification process

This section deals with the method to reduce the searching space at the face identification stage by classifying the potential faces based on the geometrical and topological constraints.

4.1. Basic algorithm for identifying actual faces

It is very important to identify edge circuits corresponding to the actual faces of a 3D object when reconstructing the most plausible 3D object. To identify faces of an object in sketch drawing is a matter of selection, i.e., the selection of k number of faces among m number of potential faces by means of 2^m number of the combinatorial searches.

Actual faces can be identified by optimizing Eq. (1). By reducing the number of minimal faces in the combinatorial search, actual faces can be identified in short time [3,4].

$$|R^{+}(e) - R(e)| + |R + (v) - R(v)|, \tag{1}$$

where, the ranks $R(v)$ and $R(e)$ are the number of faces whose boundary contains either the vertex v or the edge e; the upper bound of each rank is denoted as $R^{+}(v)$ and $R^{+}(e)$.

4.2. Classification of the plausible faces

There are many combinatorial searches because of a number of potential faces that correspond to the faces of an object depicted in a line drawing, and it is necessary to reduce the searching space of the face identification. In this paper, We describe several constraints to cut down the searching space by classifying the potential faces into the implausible, the minimal and the basic faces as shown in Table 1.

Table 1
Classification of potential faces

Face class	Description	Set symbol
Potential faces	All non-self intersect edge circuits	*PF*
Implausible faces	Edge circuits that cannot be used as actual faces	*IF*
Minimal faces	Edge circuits that may be actual faces	*MF*
Basis faces	Edge circuits that must be actual faces	*BF*

If the actual faces of an object are referred to as AF, Eqs. (2)–(4) will be derived, which means that actual faces can be identified by searching only the minimal faces

$$IF \cup MF \cup BF = PF, \tag{2}$$

$$IF \cap MF = MF \cap BF = BF \cap IF = \phi, \tag{3}$$

$$BF \subseteq AF \subseteq (BF \cup PF). \tag{4}$$

4.3. Identifying the class of faces

As mentioned above, the ranks $R(v)$ and $R(e)$ are the number of faces whose boundary contains either the vertex v or the edge e; the upper bound of each rank is denoted as $R^{+}(v)$ and $R^{+}(e)$ for each. Also, we define $RF(v)$ and $RF(e)$ as the sets of faces whose boundaries contain each entity.

There are 6 steps for the classification of potential faces.

- *Step* 1: Generate all potential faces using n edges, i.e., PF. Initially, $IF = MF = BF = \phi$

$$PF = makeface\{e_1, e_2, \ldots, e_{n-1}, e_n\}. \tag{5}$$

- *Step* 2: Find implausible faces, IF, contain internal edge(s)

$$\{f \backslash f \in PF, [f = (f_1 \cup f_2) - (f_1 \cap f_2), \text{ if } \forall e(f_1 \cap f_2), e \text{ is the internal edge of} f]\}. \tag{6}$$

- *Step* 3: Find basis faces, BF

$$\{f \backslash f \in (PF - IF) = F, [\exists e_1, \exists e_2, n(RF(e_1) \cap RF(e_2)) = 1, f \in RF(e_1) \cap RF(e_2)]\}, \tag{7}$$

where, e_1 and e_2 are connected edges.

- *Step* 4: Find implausible faces by using the “maximum rank”

$$\{f \backslash f \in (PF - BF - IF), [\exists e, RBF(e) = R^{+}(e), f \in (RF(e) - RBF(e))]\}. \tag{8}$$

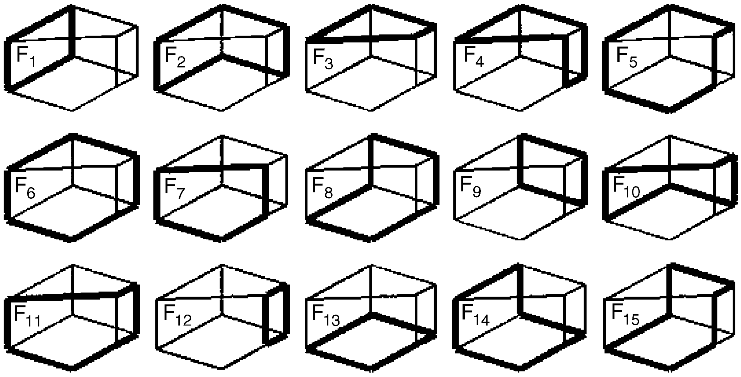

Fig. 3. All 15 potential faces that could be used in 3D reconstruction from 2D sketch drawing.

- *Step* 5: Recover over-reduced minimal faces

$$\{f \backslash f \in IF, F = (PF - IF), f \in makeface\{e|(R^{+}(e) - n(RF(e)) \geqslant 1\}\}. \quad (9)$$

- *Step* 6: Repeat *Step* 3–*Step* 5 until there is no change in the class of faces. All the faces in $(PF - IF - BF)$ are undetermined minimal faces.

For example, in *Step* 1, 15 potential faces can be generated from the 2D sketch as in Fig. 3.

In *Step* 2, 7 implausible faces, f_2, f_4, $f_{5,6}$, f_8, f_{11} and f_{14} can be found. By applying Eq. (6), 6 basic faces, f_1, f_3, f_7, f_9, f_{12} and f_{13} can be caught by applying Eq. (6).

However, according to the 'face adjacency theorem' [4], face f_7 and face f_{13} cannot coexist. Therefore, some constraints must be added in *Step* 3.

$$\{f \backslash f_1, f_2 \in BF, \forall e \in (f_1 \cap f_2) \text{ are smooth}\}. \quad (10)$$

By applying Eq. (10), two faces f_7 and f_{13} remain in potential faces. In *Step* 4, we find implausible faces f_{10} and f_{16} as shown in Fig. 4.

Step 5–*Step* 6 are not applied to this example. Finally, we derived 4 basic faces f_1,f_3,f_9 and f_{12}, and two minimal faces f_7 and f_{13}. In this paper, we can identify actual faces of an object in short time by searching minimal faces only.

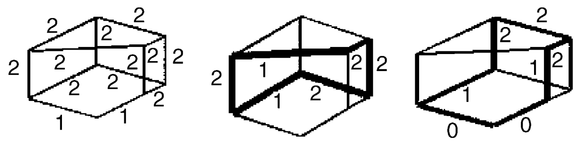

Fig. 4. Implausible faces complying with Eq. (8).

5. Object reconstruction process

5.1. Sketch order analysis

In this paper, we analyze the sketching order of faces in reconstructing a 3D object progressively from the off-line sketch drawing. There are a number of sketching orders in drawing an object. If there are n number of actual faces of an object, and the sketch is done face by face, there will be about (actually smaller than) $n!$ number of sketch orders in case of the face by face drawing. We analyze the initial sketch and the successive sketch orders based on the following constraints.

1. *Initial sketch* (S_0) *analysis*: It is general to draw orthogonal, adjacent faces of an object first. Also, it is usual to start sketching with large faces. In this paper, it is assumed that the initial sketch drawing (S_0) contains at least two adjacent faces and 6 points as shown in the leftmost part of Figs. 5b,c, which come from Fig. 5a.
2. *Successive sketch order* ($S_i, i > 0$) *analysis*: In general, human beings tend to draw the adjacent faces of the previous sketch drawing. To get ith sketching order (S_i), we select the face adjacent to those of $(i-1)$th sketching order (S_{i-1}) and that has the highest rate in the number of previously sketched edges as shown in Fig. 5b.

Our sketching order analysis is a natural one and coincides with the general order of drawing. In this respect, the sketching order of Fig. 5c is inappropriate to produce the most plausible object depicted in the sketch drawing.

5.2. Initial object reconstruction

We identify the prevailing axis system by analyzing the distribution of strokes by means of an angular

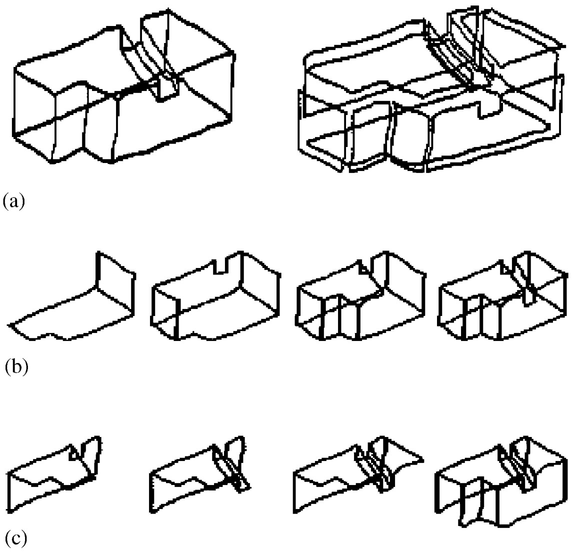

Fig. 5. Sketch order analysis: (a) Off-line sketch drawing and face identification, (b) Normal sketch order, and (c) Abnormal sketch order.

distribution graph (ADG) at first. ADG is constructed by sampling the angle of every entity in the sketch drawing. Then, we generate the objective function by weighting coefficients of various image regularities [1]. We optimize the objective function for the Z coordinates derived from points $P(X, Y)$ in the initial sketch drawing (S_0), and reconstruct the initial object (O_0). Brent minimization algorithm is used [14] to solve the full n-dimensional nonlinear optimization problem, where n refers to the number of points in the initial sketch drawing. ADG provides good initial estimates for most of typical engineering parts with some degree of orthogonality.

Once the initial object has been reconstructed, we can define the projection matrix T where the 3D vertex v, which is expressed in homogeneous coordinates, is transformed into the normalized homogeneous 2D point p on the sketch plane. In the experiment, it is found that the matrix T is useful to show the relationship between the sketch drawing and an object even though the sketch drawing itself is inaccurate. The matrix T is defined as [15]

$$\begin{pmatrix} p_u \\ p_v \\ 1 \end{pmatrix} = \begin{pmatrix} t_0 & t_1 & t_2 & t_3 \\ t_4 & t_5 & t_6 & t_7 \\ t_8 & t_9 & t_{10} & 1 \end{pmatrix} \begin{pmatrix} v_x \\ v_y \\ v_z \\ 1 \end{pmatrix}. \tag{11}$$

The projection matrix is calculated as follows. Supposing that there is a normalized 2D line $l(a, b, c)$ in a sketch drawing, we can define a 3D plane A projected into the 2D line l as

$$A = (a \quad b \quad c)T = \begin{pmatrix} at_0 + bt_4 + ct_8 \\ at_1 + bt_5 + ct_9 \\ at_2 + bt_6 + ct_{10} \\ at_3 + bt_7 + c \end{pmatrix}. \tag{12}$$

Let us refer to the two 2D lines which are orthogonal to 2D point $p(p_u, p_v)$ as $l_h(0, 1, -p_v)$ and $l_v(1, 0, -p_u)$ for each. Then, we can define the 3D planes A_h and A_v that are respectively projected into the 2D lines as Eq. (12). Eq. (13) can be derived from the plane A_h and vertex v because the 3D vertex v should lie on the plane A_h. Eq. (14) can be derived from the plane A_v and the vertex v in the same way

$$A_h v = (0 \quad 1 \quad -p_v)Tv = 0, \tag{13}$$

$$A_v v = (0 \quad 1 \quad -p_u)Tv = 0. \tag{14}$$

Therefore, each 2D point whose corresponding 3D positions are given provides two linear equations. To get 11 coefficients from ith projection matrix T_i, the sketch drawing should have at least 6 points. We can solve the linear equations such as Eqs. (13) and (14) by using singular value decomposition [16].

5.3. 3D constraints

To estimate the progressive positions for the inserted vertices, three constraints are derived from the relationship between an object and the sketch drawing. Each vertex should satisfy three constraints in the reconstruction process. Three constraints are as follows:

1. *Vertex constraint*: According to Eq. (11), vertex constraint means that each vertex v of a 3D object should be transformed into the corresponding 2D point p of the sketch drawing by the projection matrix T (Fig. 6). That is, each vertex v satisfies $Tv - p = 0$. Supposing that there are n vertices of a 3D object, the vertex constraint can be

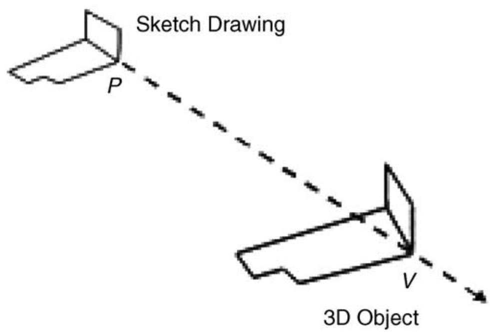

Fig. 6. Vertex constraint.

defined as

$$F_{vertex} = \sum_{i=0}^{n} \{(Tv_i - p_u^i)^2 + (Tv_i - p_v^i)^2\}, \qquad (15)$$

where (p_u^i, p_v^i) are the coordinates of the corresponding point p_i of ith vertex v_i; T denotes the projection matrix extracted at the previous reconstruction step.

2. *Edge constraint*: Given a 2D line l $(\overline{p_1p_2})$, a 3D plane A is generated by Eq. (12). The corresponding end vertices of a 3D line L $(\overline{v_1v_2})$ should lie on the plane A. The edge constraint can be defined by summing the distance from each vertex to the plane. However, there are infinite number of plausible lines that satisfy the vertex constraint and the edge constraint as shown in Fig. 7. Assuming that there are n number of edges that make a 3D object. Then, the edge constraint can be defined as

$$F_{edge} = \frac{\sum_{i=0}^{n-1} \{(l_i T v_1)^2 + (l_i T v_2)^2\}}{n}, \qquad (16)$$

where, v_1 and v_2 are the two end vertices of a 3D line L; l is the normalized 2D line corresponding to L; T refers to the projection matrix extracted at the previous reconstruction step.

3. *Face constraint*: Face constraint means that 3D vertex v should lie on the 3D plane (face). With the constraints on vertex, edge and face, we can narrow down the optimal 3D line (Fig. 8) among the infinite plausible lines (Fig. 7). Assuming that there are n number of faces of a 3D object, and each face has m number of vertices, face constraint can be defined as

$$F_{face} = \frac{\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} \{a_i x_i^j + b_i y_i^j + c_i z_i^j + d_i\}^2}{n}, \qquad (17)$$

where, a_i, b_i, c_i and d_i are the coefficients of ith plane equation and satisfy $\sqrt{a_i^2 + b_i^2 + c_i^2} = 1$; (x_i^j, y_i^j, z_i^j) are the coordinates of jth vertex of ith face.

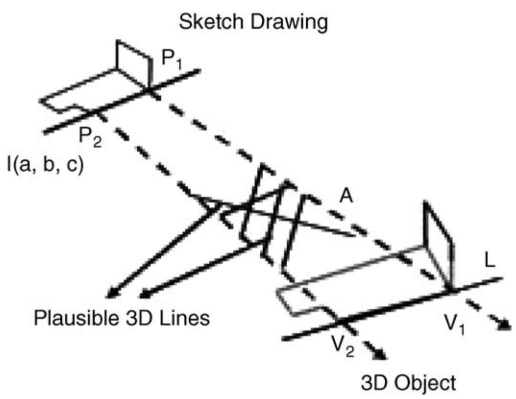

Fig. 7. Edge constraint.

5.4. Optimization

To estimate the progressive positions of the vertices as shown in Fig. 9(a), we generate the $(i+1)$th partial object (O^{i+1}) by optimizing the objective function derived from the previously reconstructed partial object (O^i), projection matrix (T^i), and current sketch drawing (S^{i+1}). The objective function must be optimized for the coordinates of vertices. In this paper, we define the objective function $(F_{progressive})$ as the sum of the three 3D constraints as

$$F_{progressive} = F_{vertex} + F_{edge} + F_{face}. \qquad (18)$$

A vector V containing the x, y, and z coordinates of the vertices can represent a 3D configuration of an object. An objective function $F_{progressive}(V)$ can be

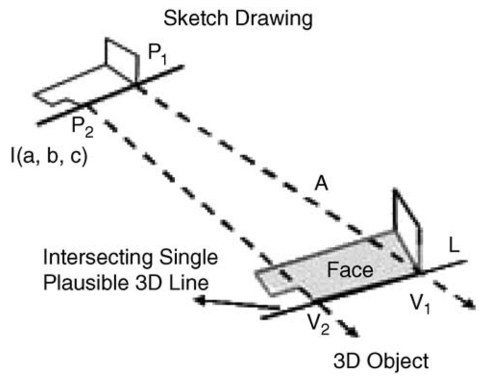

Fig. 8. Face constraint.

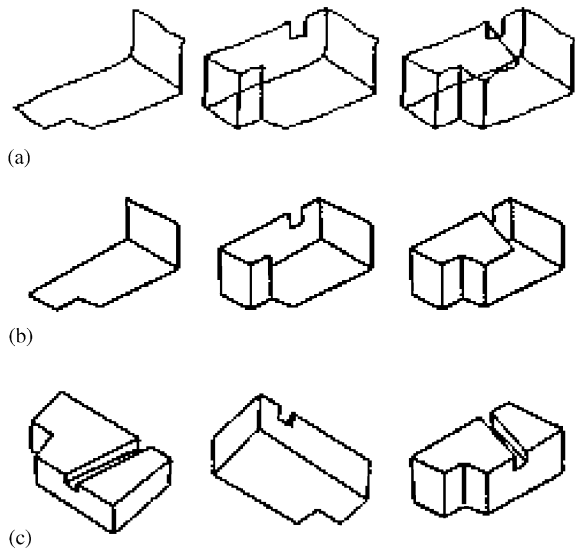

Fig. 9. Reconstruction results: (a) Progressive sketch drawings, (b) Progressive reconstructed partial objects, and (c) The final reconstructed complete object.

computed for any 3D configuration by evaluating the three constraints above. The manipulation process of the V to get the best reconstruction is a matter of a full $3 \times n$-dimensional nonlinear optimization, where n is the number of vertices of the current partial object. The Brent minimization algorithm [14] is used in this paper. Since the minimization actually needs much time, we reconstruct the 3D structure of an object by the unit of face.

Although the nonlinear problem is optimized by dividing it into smaller units, initial estimation of the solution is still the main issue. ADG is used as an initial estimation to reconstruct the initial object, and the $(i-1)$th partial solution as an initial estimation of the ith solution. This initial estimation is useful to restore the 3D structure of an object efficiently as shown in Fig. 9.

5.5. Dynamic viewpoints

On-line sketching enables its users to draw a model part by part at the viewpoint they prefer. Even after some parts of an object have been reconstructed, it is still possible to change the viewpoint. Then, the 3D vertices $V(X, Y, Z, 1)$ are transformed into $V'(X', Y', Z', 1)$ by means of the 4×4 transformation matrix M as in Eq. (19). That is, the 3D vertices V are projected into the points $P'(X', Y')$ in the new sketch drawing

$$V' = MV. \tag{19}$$

If we use the relationship between the 2D entities in the sketch drawing to reconstruct a 3D object, Z' can be used as the initial depth value in points P'. However, such values calculated from the previous viewpoint cannot be used as the weighted coefficients of the image regularities. Furthermore, the optimization of the depth values will be needed again because the coordinates of the object vary from time to time when the viewpoints are changed.

Fig. 10 shows the results of the reconstruction in which the progressive reconstruction algorithm is used. The reconstructed 3D object has its own original X, Y, and Z coordinates. It is possible to use all the information calculated from the previous viewpoint. We have only to recalculate the projection matrix T that establishes relationship between a partial object and a sketch drawing (new P'). The new T' can be calculated by matrix multiplication ($T' = TM$). As a result, the 3D object will be reconstructed from the sketch drawing with dynamic viewpoints in short time.

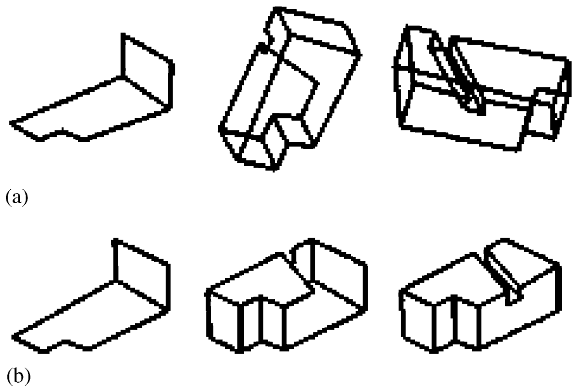

Fig. 10. Reconstruction from dynamic viewpoints: (a) On-line sketch drawing with dynamic viewpoints, and (b) Progressive reconstruction corresponding to the dynamic viewpoints.

6. Experimental results

6.1. Results

To estimate the efficiency of the proposed algorithm, it was applied to various sketch drawings. The experiment is done on a PC with Pentium III processor (600 MHz).

To show the effectiveness of the progressive reconstruction algorithm, the algorithm is applied to the synthetic sketch drawing which is acquired by a parallel projection of the synthetic objects as shown in Fig. 11a. Our algorithm can even be applied to an inaccurate sketch drawing (parallel freehand sketch drawing) as shown in Fig. 11b as well as to an accurate one. In the case of perspective projection (Fig. 11c,d), the algorithm generates somewhat slanted objects because the initial object reconstruction uses image regularities, which is assumed to be the parallel projection. In experiments, we assume that the initial object is given.

We illustrate a few examples that are too complex to be sketched as shown in Fig. 12. The projections of synthetic 3D objects is used as the sketch drawing here. The results are satisfactory in that the concept of products can be conveyed.

To determine the accuracy of reconstruction results, we showed the reconstructed 3D objects arbitrarily rotated with hidden-line removal or rendering. Normally, the more accurately we could reconstruct the 3D object, the more accurate the sketch drawings became. Table 2 shows the evaluation of the progressive algorithm for various projections and objects. It shows that the proposed algorithm reconstructs the most plausible objects from the sketch drawings in short time.

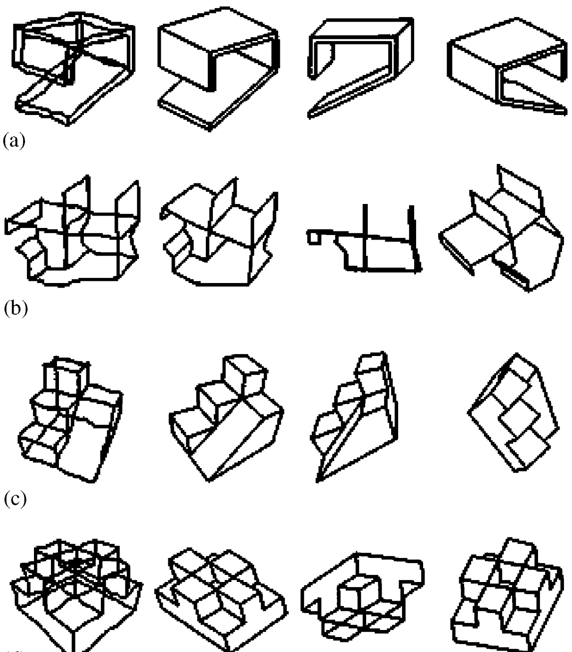

Fig. 11. Reconstruction results 1 (leftmost: sketch drawings, right 3: the reconstruction results in wireframe): (a) Manifold object (orthogonal), (b) Non-manifold object (parallel or freehand), (c) Manifold object (perspective), and (d) Object with hole (perspective).

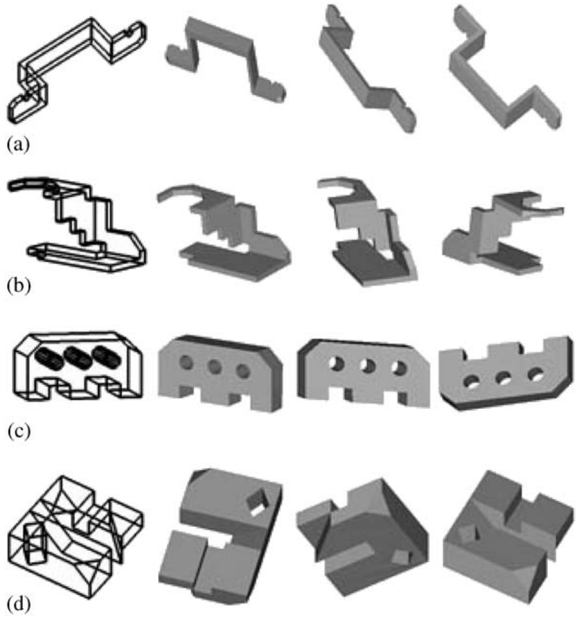

Fig. 12. Reconstruction results 2 (leftmost: sketch drawings, right 3: the reconstruction results in rendering): (a) Sheet metal, (b) General mechanical part, (c) Product with hole, and (d) General product with hole.

Table 2
Evaluation of the progressive algorithm

Sketch drawing	# of vertices	# of edges	# of faces	$F_{progressive}$ (V)	Time (s)
Fig. 11a	20	30	12	0.30161	7
Fig. 11b	28	39	11	1.41668	8
Fig. 11c	22	41	28	12.61546	5
Fig. 11d	32	56	26	0.35640	21
Fig. 12a	48	72	26	0.72953	30
Fig. 12b	68	100	34	2.14005	58
Fig. 12c	124	186	64	0.51826	74
Fig. 12d	46	68	24	0.36145	30

6.2. Discussions

6.2.1. The comparison of time in the reconstruction process

To estimate the efficiency of the proposed approach, it was applied to various objects as shown in Fig. 13.

Table 3 shows that the method efficiently narrows down the searching space of the face identification to a manageable size. Note that the number of the searching domains of minimal faces was zero at face identification stage. Therefore, the total time for face identification in most cases was dramatically reduced when the proposed method was used.

To evaluate our results at the object reconstruction stage, we compared them with the results from Lipson's algorithm, which are acknowledged as one of the best. Lipson's algorithm works well in the freehand sketch with parallel projections. It optimizes the objective function derived from image regularities for Z coordinates. Before comparing our results with his, it should be mentioned that our results are not absolute evaluation in that we compared the estimated value of the objected functions. We want to show that our reconstruction process is acceptable as shown in Fig. 13 although image regularities are not used in our objective function. In general, the results show low evaluation values through the objective function because each algorithm minimizes the objective function directly.

Table 4 shows the evaluated values of the objective function applied, F_{lipson} and $F_{progressive}$, respectively. It shows that the proposed algorithm reconstructs the most plausible objects from the sketch drawings in short time. In experiments, we discovered that the evaluated value of Lipson's algorithm F_{lipson} was merely the optimization value of the objective function. In other words, it reflects only the image regularities on

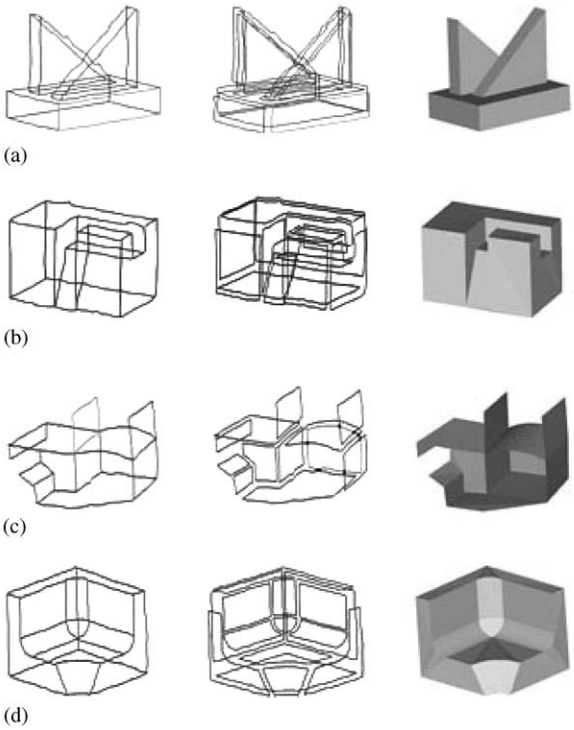

Fig. 13. Experimental results (left: input 2D sketches, center: identified 2D faces, right: reconstructed 3D object): (a) Polyhedral object, (b) Manifold polyhedral object, (c) Non-manifold curvilinear object (sheet metal), and (d) Curvilinear object with conical, spherical and cylindrical surfaces.

Table 3
Comparison of the face identification

Fig. 13	Method	# of *PF*	# of *IF*	# of *BF*	# of *MF*	SOL	Time (s)
(a)	A	33	14	0	19	1	142
	B	33	17	16	0	1	30
(b)	A	279	159	0	120	1	1200
	B	279	265	14	0	1	138
(c)	A	205	164	0	41	1	1091
	B	205	193	12	0	1	551
(d)	A	896	679	0	202	1	7283
	B	896	882	14	0	1	4420

A: previous method [4], B: proposed method.

the sketch plane to derive weight coefficients to minimize the objective function. In contrast, the evaluated values of the objective function applied in our algorithm, $F_{progressive}$ reflects the accuracy of the sketch drawings

Table 4
Comparison of the object reconstruction

Fig. 13	Method	F_{lipson} (V)	$F_{progressive}$ (V)	Time (s)	Total time (s)	B/A
(a)	A	6.37806	121.74585	44	186	0.21505
	B	8.44948	0.36059	10	40	
(b)	A	3.17720	5.20678	33	1233	0.11922
	B	3.69264	1.06106	9	127	
(c)	A	14.38779	174.27446	66	1157	0.48314
	B	23.06679	1.41668	8	559	
(d)	A	11.85102	121.76334	33	7316	0.60524
	B	16.48199	8.98607	8	4428	

A: previous method [1], B: proposed method.

as discussed above as well as the optimization values of the objective function. As shown in Fig. 13, the convergence of the objective function does not necessarily mean the reconstruction is the most plausible object. Experiments show that there is no absolute objective function that correctly simulates the way human beings perceive it.

Table 4 shows the total time which is spent on the reconstruction process including the time at the face identification stage and at the objective reconstruction stage. By searching for the reduced minimal faces only, we can identify the actual faces of an object in short time at the face identification stage. Also, the 3D structure can be restored by applying the progressive reconstruction method in accordance with the sketching order of faces step by step.

6.2.2. The comparison of accuracy in the reconstruction process

By examining the relationship between the 3D object and the sketch drawing, we suggested three constraints to reconstruct the most plausible object. Nevertheless, the reconstruction tends to produce a somewhat distorted 3D object due to the inherent inaccuracies in sketch drawing. Since the sketch drawing itself is generally inaccurate, we cannot determine whether the optimization will succeed or fail. All we can do is to start the optimization process from good initial estimates so as not to get into local minimum.

Currently, it is difficult to decide which reconstruction is more perceivable than others, because the sketch drawing itself is inaccurate mathematically. We can get different reconstruction using the objective function. For example, different objects might be obtained

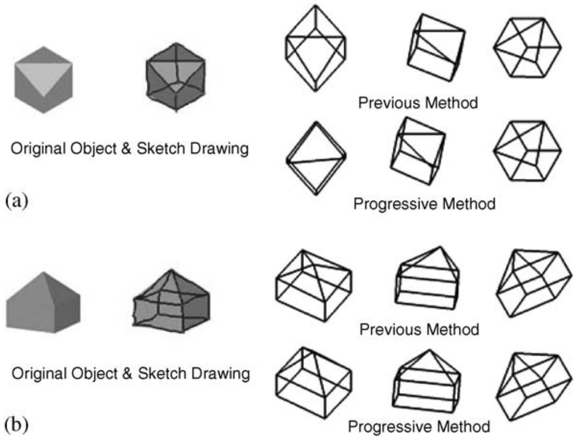

Fig. 14. Comparison results of the reconstruction error between previous method [8] and ours: (a) Object 1, and (b) Object 2.

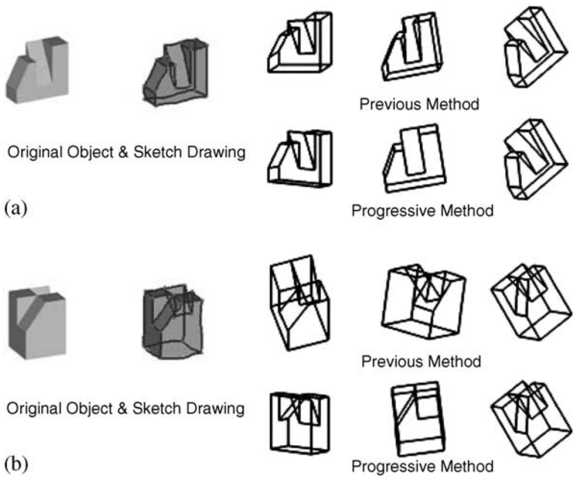

Fig. 15. The accuracy comparison of practical objects between Lipson [8] and ours: (a) Practical object 1, and (b) Practical object 2.

from one sketch drawing by applying different objective functions derived from different constraints as shown in Figs. 14 and 15 despite their similar appearances. Synthetic objects can be used to compare the accuracy of the reconstruction between Lipson's method and that proposed in this paper. We estimated the standard deviation of Z coordinates between the synthetic object's vertices and the reconstructed object's. Table 5 shows the results produced by applying the proposed method are more accurate than those produced by applying Lipson's method. The coordinates of vertices of an object could be converged into the configuration of the most plausible object by optimizing the objective function derived from the 3D constraints more accurately than Lipson's method in which only 2D image regularities could do.

Table 5
Comparison of the accuracy of the reconstruction

Figure	Method	Average Z error	Standard deviation of Z	Reconstruction time (s)
14(a)	A	87.89184	45.79811	2
	B	62.25519	31.96594	1
14(b)	A	34.45684	21.76027	2
	B	54.66039	27.53135	1
15(a)	A	26.07769	14.50979	19
	B	21.34016	6.13122	4
15(b)	A	93.22366	43.85853	11
	B	12.11686	7.63214	4

A: previous method [1], B: proposed method.

7. Conclusion

We have shown that the proposed algorithm effectively tackles the problems in reconstructing the 3D sketch in case where the depth information is difficult to establish by progressively restoring 3D structure. The proposed algorithm reconstructs the 3D object in accordance with the sketching order of faces to improve the efficiency.

As a result, we were able to develop a system that can reconstruct the complete model of mechanical components synthetically and automatically from the sketch drawing. By the progressive approach, a manifold/non-manifold object could be generated more rapidly than by the freehand/off-line/on-line sketch drawings with orthographic, parallel or perspective projections. In addition, the proposed algorithm could handle the dynamic viewpoints of users efficiently.

We can promote future studies as follows. First, we are interested in refining the sketch drawing. Second, we will use quadric models to extend our approach to the general modeling of mechanical components.

References

[1] Lipson H, Shpitalni M. Optimization based reconstruction of a 3d object from a single free-hand line drawing. Computer Aided Design 1996;28(8):651–63.

[2] Shpitalni M, Lipson H. Classification of sketch strokes and corner detection using conic sections and adaptive clustering. Transactions of the ASME Journal of Mechanical Design 1997;119(2):131–5.
[3] Oh BS, Kim CH. Fast reconstruction of 3d objects from single free-hand line drawing. Lecture Notes in Computer Science, vol. 2059. Berlin: Springer, 2001. p. 706–15.
[4] Shpitalni M, Lipson H. Identification of faces in a 2d line drawing projection of a wireframe object. IEEE Transactions on Pattern Analysis & Machine Intelligence 1996;18(10):1000–12.
[5] Wang W, Grinstein G. A survey of 3d solid reconstruction from 2d projection line drawings. Computer Graphics Forum 1993;12(2):137–58.
[6] Lipson H. Computer aided 3D sketching for conceptual design. Ph.D. thesis. Israel: Israel Institute of Technology; 1998.
[7] Huffman DA. Impossible objects as nonsence sentences. In: Meltzer, Michie D, Machine intelligence, vol. 6B. Edinburgh: Edinburgh University Press, 1971.
[8] Kanade T. Recovery of the three-dimensional shape of an object from a single view. Artificial Intelligence 1980;17:409–670.
[9] Sugihara K. Interpretation of line drawing. Cambridge, MA: MIT Press, 1996.
[10] Marill T. Emulating the human interpretation of line drawings as three-dimensional objects. International Journal of Computer Vision 1991;6(2):147–61.
[11] Braid L, Wang P. Three-dimensional object recognition using gradient descent and the universal three-dimensional array grammar. SPIE 1991;1607:711–8.
[12] Leclerc YG, Fiscler MA. An optimization based approach to the interpretation of single line drawings as 3d wire frames. International Journal of Computer Vision 1992;9(2):113–6.
[13] Lamb D, Bandopadhay A. Interpreting a 3d object from a rough 2d line drawing. In: Visualization '90 Proceedings, 1990.
[14] Brent RP. Algorithms for minimization without derivatives. Englewood Cliffs, NJ: Prentice-Hall, 1973.
[15] Rogers DF, Adams JA. Mathematical elements for computer graphics. New York: McGraw-Hill, 1976.
[16] Press WH, Teukolsky SA, Vetterling WT, Flannery BP. Numerical recipes in C: the art of scientific computing, 2nd ed. Cambridge: Cambridge University Press, 1992.