3D Object Reconstruction from a Single 2D Line Drawing without Hidden Lines

Liangliang Cao and Jianzhuang Liu
Department of Information Engineering
The Chinese University of Hong Kong
{llcao3,jzliu}@ie.cuhk.edu.hk

Xiaoou Tang
Microsoft Research Asia
Beijing, China
xitang@microsoft.com

Abstract

The human vision system can interpret a single 2D line drawing as a 3D object without much difficulty even if the hidden lines of the object are invisible. Several reconstruction approaches have tried to emulate this ability, but they cannot recover the complete object if the hidden lines of the object are not shown. This paper proposes a novel approach for reconstructing complete 3D objects from line drawings without hidden lines. First, we develop some constraints and properties for the inference of the topology of the invisible edges and vertices of an object. Then we present a reconstruction method based on perceptual symmetry and planarity of the object. We give a number of examples to demonstrate the ability of our approach.

1. Introduction

A line drawing is defined as a 2D projection of the edges and vertices of a 3D object in a generic view, with or without hidden lines visible. The human vision system has the ability to interpret 2D line drawings as 3D objects without difficulty. Emulating this ability is an interesting research topic for machine vision. The applications include providing 2D sketch query interface for 3D object retrieval from large databases or from the web [1], interactive generation of 3D models from images [2], [3], and flexible sketching interface for designers to sketch ideas in object design [4], [5].

Since the early stage of computer vision , a large amount of work called line labeling has been carried out for line drawing interpretation [6], [7]. Line labeling focuses on finding a set of consistent labels from a line drawing without hidden lines, and does not explicitly give the 3D structure represented by a line drawing. Another body of work on line drawing interpretation is related to judging the correctness of line drawings and give their possible reconstruction based on algebra test with linear equalities and inequalities [8], [9], [10]. The common problem of these methods is that the formulation is superstrict and not robust; an origi-nally correct line drawing will be judged as *impossible* after a little deviation of one or more vertices, causing a 3D reconstruction to fail [10].

Recently, some researchers formulated the 3D reconstruction as an optimization problem based on different objective functions. Marill proposed a criterion, minimizing the standard deviation of the angles (MSDA) in a reconstructed object to emulate human 3D perception of 2D line drawings [11]. This idea is followed by the researchers in [4], [5], [12], [13]. This approach is tolerant of freehand sketching errors, but cannot reconstruct complete 3D objects if their hidden lines are not drawn.

This paper proposes an approach to the 3D reconstruction from line drawings without hidden lines. Compared with the line drawings with hidden lines visible, these line drawings are easier and more natural to draw. In addition, if we want to recover the complete 3D shape of an object in an image, we usually do not have the invisible edges. Our approach mainly consists of two elements: inference of the topology of the hidden part of an object and reconstruction of the complete shape of the object. Fig. 1 shows the two steps with an example.
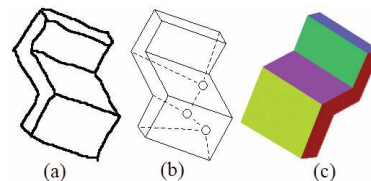

Fig. 1: (a) A line drawing without hidden lines. (b) The inferred hidden topological structure. (c) Reconstructed complete 3D object.

For easier understanding of the technical content of our approach, we first summarize the terms that will be used in the rest of the paper. Some of these terms are illustrated with Fig. 2.

- **Degree**. The degree $d(v)$ of a vertex v is the number of edges meeting at v in a line drawing.

- **Incomplete vertex**. An incomplete vertex v is a vertex of $d(v) = 2$.
- **Complete vertex**. A complete vertex v is a vertex of $d(v) = 3$.
- **Broken vertex**. A broken vertex v is a vertex of $d(v) = 1$.
- **Cycle**. A cycle is a closed trail in a line drawing where all its vertices except the end vertices are distinct.
- **Visible face**. A visible face is a face bounded by a cycle composed of all visible edges in a line drawing.
- **Rank**. The rank $R(e)$ of an edge e is the number of visible faces with edges passing through e.
- **Zero edge**. A zero edge e is an edge of $R(e) = 0$.
- **Boundary edge**. A boundary edge e is an edge of $R(e) = 1$.
- **Boundary cycle**. A boundary cycle is a cycle where all its edges are boundary edges.
- **Hidden cycle**. A hidden cycle is a cycle where all its edges and vertices are hidden.
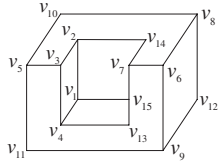- N_H. N_H denotes the number of hidden vertices.

Fig. 2: Illustration of some terms. Here v_{1-9} are complete vertices, v_{10-14} are incomplete vertices, v_{15} is a broken vertex, (v_{11}, v_9) is a boundary edge, (v_1, v_{15}) is a zero edge, and $(v_{10}, v_5, v_{11}, v_9, v_{12}, v_8, v_{10})$ is a boundary cycle.

2. Theoretical Inference of the Hidden Topological Structure

2.1. Assumptions

Given a line drawing without hidden lines, there are infinite possible structures that might be the hidden part of the object although human beings usually have the unique perception of the complete 3D shape. Therefore, we have to impose reasonable constraints for the inference of the hidden structure such that the final recovered 3D object is in accordance with our perception from the line drawing. In this first work to tackle the problem, we focus on a class of relatively simple solids and make the following assumptions.

Assumption 1 *The 3D objects are polyhedra with all the vertices met by three edges and all the edges passed through by two faces.*

Assumption 2 *A line drawing is the parallel or near-parallel projection of the visible edges and vertices of a single polyhedron defined above in a generic view.*

Assumption 3 *Every hidden vertex is connected with at least one visible vertex.*

2.2. Finding the Degrees and Ranks

Given a line drawing, there are three types of vertices as defined in Section 1. Incomplete vertices are easy to find, but to distinguish broken vertices from complete vertices is not obvious. The following theorem allows us to identify the broken vertices.
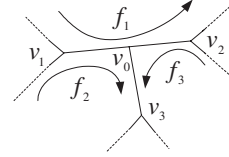

Fig. 3: Part of a line drawing where v_1, v_0 and v_2 are collinear.

Theorem 1 *If a vertex v_0 touches a straight line in a line drawing as shown in Fig. 3, then v_0 is a broken vertex.*

Proof. Suppose, to the contrary, that v_0 is not a broken vertex. Then it is a complete vertex. With the assumption that every edge of the object is passed through by two faces, there are three planar faces passing through v_0. Let them be $f_1 = (v_1, v_0, v_2, ..., v_1), f_2 = (v_1, v_0, v_3, ...v_1)$ and $f_3 = (v_2, v_0, v_3, ..., v_2)$. since v_1, v_0, v_2 are collinear, the straight line (v_1, v_2) and the vertex v_3 that is not on this line defines a plane, implying that the two faces f_2 and f_3 are coplanar. Thus the edge (v_3, v_0) should not exist. Therefore, if edge (v_3, v_0) is a visible edge of the object, v_0 is a broken vertex. □

Knowing all the types of vertices, we can find the degrees of the vertices directly. To find the ranks of the edges, we have to find the visible faces first. Face identification from a line drawing is not a trivial problem. Fortunately, there have been algorithms available for this purpose. We can use one of the algorithms published in [14] and [15] to find the visible faces. For example, the four visible faces found from the line drawing shown in Fig. 2 are $(v_1, v_2, v_3, v_4, v_1)$, $(v_6, v_9, v_{12}, v_8, v_6)$, $(v_2, v_3, v_5, v_{10}, v_8, v_6, v_7, v_{14}, v_2)$, and $(v_4, v_3, v_5, v_{11}, v_9, v_6, v_7, v_{13}, v_4)$. Since v_{15} is a broken vertex, we cannot find the visible face passing through (v_1, v_{15}) at this stage. Thus, $R(v_1, v_{15}) = 0$. Note that when v_{15} is found to be a broken vertex, the line from v_7 to v_{13} is one edge but not two.

2.3. Constraints for the Inference

The key to the inference of the invisible vertices and edges is to determine the number of hidden vertices N_H and the

connections among invisible, incomplete, and broken vertices. The following theorems gives constraints useful for the inference.

Theorem 2 *Let $\mathcal{V}_\mathcal{I}$ and $\mathcal{V}_\mathcal{B}$ be the set of incomplete and broken vertices of a given line drawing, respectively. Then we have*

$$N_H \leq |\mathcal{V}_\mathcal{I}| + |\mathcal{V}_\mathcal{B}|, \tag{1}$$

where $|\cdot|$ denotes the number of elements in a set.

Proof. A complete vertex does not connect to any hidden vertex. An incomplete vertex connects to one hidden vertex. A broken vertex connect to one hidden vertex too. From Assumption 3, the largest value of N_H appears when all the hidden vertices connect to different visible vertices, which implies the inequality in (1). $\square$

The following Lemma [16] is used to prove Theorem 3.

Lemma 1 *Let $\mathcal{G}$ be any graph, $\mathcal{E}$ be the set of edges and $\mathcal{V}$ be the set of vertices in $\mathcal{G}$. It holds that*

$$\sum_{v \in \mathcal{V}} d(v) = 2|\mathcal{E}|.$$

Theorem 3 *Given a line drawing, if $|\mathcal{V}_\mathcal{I}| + |\mathcal{V}_\mathcal{B}|$ is even (odd), N_H of the line drawing must be even (odd).*

Proof. Suppose there are N_H hidden vertices and l hidden edges in the object. If we construct a graph using all the hidden edges, hidden vertices, incomplete vertices, and broken vertices (without all visible edges), from Lemma 1 we have

$$\sum_{v \in \mathcal{V}_\mathcal{I}} 1 + \sum_{v \in \mathcal{V}_\mathcal{B}} 1 + \sum_{v \in \mathcal{V}_\mathcal{H}} 3 = 2l,$$

where $\mathcal{V}_\mathcal{I}, \mathcal{V}_\mathcal{B}$ and $\mathcal{V}_\mathcal{H}$ are the sets of incomplete, broken, and hidden vertices, respectively, and $|\mathcal{V}_\mathcal{H}| = N_H$. The above equation can be rewritten as $|\mathcal{V}_\mathcal{I}| + |\mathcal{V}_\mathcal{B}| + N_H = 2(l - N_H)$. Therefore, if $|\mathcal{V}_\mathcal{I}| + |\mathcal{V}_\mathcal{B}|$ is even (odd), N_H must be even (odd) too. $\square$

3. An Algorithm to Recover the Hidden Topological Structure

In this section, an algorithm based on the constraints and properties stated in the previous theorems is designed to infer the hidden vertices and edges in a line drawing.

3.1. Outline of the Algorithm

Step 1. Finding the degrees of all the vertices and the ranks of all the edges from a line drawing.

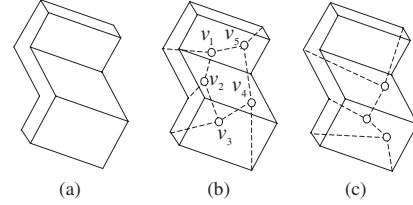**Step 2.** Constructing an initial hidden topological structure (see Fig. 4(b)).

Fig. 4: An example of inferring the hidden topology from line drawing. (a) A line drawing. (b) Initial hidden vertices (open circles). (c) Reduction of the initial hidden vertices.

Step 3. Reducing the initial hidden structure to one according to human visual perception of the 3D object (see Fig. 4(c)).

Step 1 has been described in Section 2.2. We discuss Steps 2 and 3 in the following two sections.

3.2. Constructing the Initial Hidden Structure

Theorem 2 lets us know that the largest $N_H = |\mathcal{V}_\mathcal{I}|$ since we consider the line drawings without zero edges (thus $\mathcal{V}_\mathcal{B} = \emptyset$). We set $|\mathcal{V}_\mathcal{I}|$ hidden vertices and connect each incomplete vertex to a different hidden vertex. Two hidden vertices are connected if their corresponding incomplete vertices are closest on the boundary cycle. One example is given in Fig.4(b), where the cycle $(v_1, v_2, v_3, v_4, v_5, v_1)$ is a hidden cycle.

3.3. Reducing Initial Hidden Structure

Beginning with the initial hidden structure, we design a procedure to search for other possible hidden structures with fewer hidden vertices. The procedure uses a strategy of cutting-and-merging of edges and vertices. *Cutting* one edge on a hidden cycle removes this edge from the cycle while keeping the two vertices of the edge. After the cutting, the two hidden vertices of the edge are met by only two hidden edges (see Fig. 5(b)). To maintain that every vertex is met by three edges, we *merge* the two vertices to their adjacent hidden vertices (see Fig. 5(c)). Since each cutting-and-merging reduces two of the hidden vertices, the resulting number of hidden vertices is even (odd) if the initial N_H is even (odd), which satisfies the constraint by Theorem 3.

At first, the cutting is applied to one hidden edge on the hidden cycle each time, resulting in different hidden structures. The cutting is also used to cut two or more such edges each time. It should be emphasized that not every cutting of two edges or more each time is valid. Fig. 5(d) and (f) show two examples of cutting two edges each time. The former is valid but the latter is not.

The cutting-and-merging procedure is always applied to the initial hidden structure, cutting one edge or multiple edges each time. The maximal edges that can be cut each
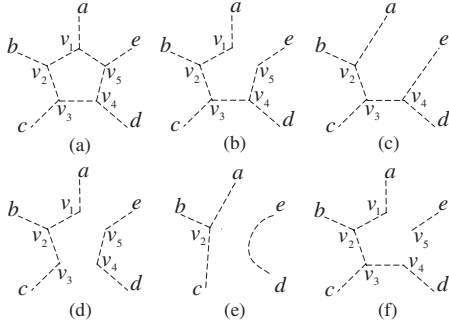
3

Fig. 5: Illustration of the cutting and merging procedure where a-e are incomplete vertices, v_{1-5} are hidden vertices, and $(v_1, v_2, v_3, v_4, v_5, v_1)$ is a hidden cycle. (a) The initial hidden structure. (b) Cutting one edge (v_1, v_5), (c) Merging v_1 to v_2 and v_5 to v_4. (d) Cutting two edges (v_1, v_5) and (v_3, v_4). (e) Merging v_1 and v_3 to v_2; v_5 and v_4 disappearing after being merged. (f) Cutting two edges (v_1, v_5) and (v_4, v_5), where v_5 has no place to be merged.

time is the largest integer $\leq N_H/2$ because removing one edge reduces two hidden vertices. All the hidden structures obtained from the procedure plus the initial structure are kept for the selection of the most plausible one. Fig. 6 shows one example.
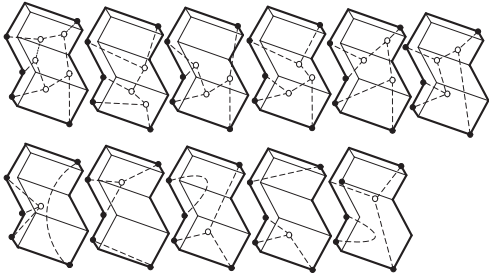

Fig. 6: The initial hidden structure and all others derived by the cutting-and-merging procedure.

3.4. Selecting the Most Plausible Structure

Given a set of possible hidden structures, the selection of the most plausible one is based on some visual psychological properties from Gestalt psychology, which is one of the most influential perception theories. It asserts that human being are innately driven to experience things as good a whole as possible. Here *good* can mean many things such as symmetry, simplicity, regularity, and order of an object [17], [18]. This nature leads us, when seeing, to strongly favoring certain shapes and configurations over others without high level identification.

The law of symmetry is one of the most important

Gestalt laws, which reveals that the human visual system is overwhelmed by symmetry and tends to interpret a figure in such a way as to produce an object that is as symmetrical as possible. When this law is applied to the inference of the hidden structure of a line drawing, each hidden face should correspond to a similar visible face. At this stage, since we discuss topological structures only, we say that two faces are similar if they have the same number of edges. These two similar faces are also called topologically same. Motivated by this law, the rule of selecting the most plausible hidden structure is: given a set of hidden structures from a line drawing, select the one having as many hidden faces similar to the visible faces as possible.

We define a non-symmetry measure NSM for the selection. Suppose there are n hidden structure, at first, we set $NSM_i = 0, 1 \leq i \leq n$. For every hidden face in the ith structure, we check if there is a corresponding topologically same visible face. If no, increase NSM_i by 1. For each structure, one visible face cannot be used more than once in the checking. The most plausible is the jth structure with $NSM_j = \min_{1 \leq i \leq n} \{NSM_i\}$.

With the method above , we can select the most plausible hidden structure for each line drawing. For the example shown in Fig. 6, the second hidden structure is selected.

4. Reconstruction of 3D Objects

In what follows, we call a line drawing with its recovered hidden structure a complete line drawing. After recovering a complete line drawing, the next important work is to reconstruct its 3D shape. Since we already know the face topology of the complete line drawing, we only need to derive the 3D coordinates of all the visible and hidden vertices. We consider a line drawing is a parallel (or near parallel) projection of a 3D object. The x- and y-coordinates of each visible vertex is thus already known, and only the z-coordinate (depth) has to be derived. However, all the x-, y- and z-coordinates have to be found for hidden vertices. This reconstruction problem is more difficult than those in the previous work [4], [5], [11], [12], where all the vertices and edges of a line drawing are given in the 2D projection. In addition, the previous methods cannot recover the complete 3D object from a line drawing without hidden vertices and edges. In the following, we develop a new objective function to recover the 3D position of both visible part and hidden part, which is in accordance with human visual perception.

Based on the spirit of the law of symmetry from Gestalt psychology, we consider a symmetry measure S for a closed planar figure first. It is defined as:

$$S = \frac{P^2}{A} \qquad (2)$$

4

where A and P are the area and perimeter of the figure, respectively.

It holds that $S \geq 4\pi$ for any closed planar figure, among which the most symmetrical one is a circle with $S = 4\pi$. For a polygon with m vertices, its most symmetrical figure has m equal-length sides. These facts indicate that (2) is a rather reasonable measure of symmetry.

An object consists of more than one face. We treat the recovered object as the integration of all its 3D planar faces. For an object with n faces, we define the whole symmetry measure of the object as

$$S_w = \sum_{i=1}^{n} \frac{P_i^2}{A_i} \qquad (3)$$

where A_i and P_i, $1 \leq i \leq n$, are the area and perimeter of face i, respectively. We expect that given a line drawing, minimizing S_w combined with other two criteria would provide us with the most plausible recovered 3D object. This measure S_w plays the main role in the reconstruction, with the other two involved to get better results.

Marill [11] presented his approach to 3D reconstruction based on a simple criterion: minimizing the standard deviation of all the angles (SDA) in the reconstructed object. SDA can be calculated by:

$$SDA = \operatorname*{Var}_{i,j}(\cos^{-1}(\mathbf{u}_i \cdot \mathbf{u}_j)) \qquad (4)$$

where $\mathbf{u}_i$ and $\mathbf{u}_j$ are the unit vectors of two lines meeting at a vertex of the 3D object. This criterion can be regarded as another form of symmetry constraint and is also incorporated into the objective function.

When we observe a line drawing representing a 3D polyhedron, we can clearly identify the cycles representing faces. This face information is very useful in helping our perception of the shape of the object [12]. We also enforce this planarity constraint in the object function.

Suppose that face i has m vertices (x_{ij}, y_{ij}, z_{ij}), $1 \leq j \leq m$. The plane that fits these vertices best can be expressed as:

$$a_i x + b_i y + c_i z - 1 = 0. \qquad (5)$$

Thus, we evaluate the deviation from planarity DP_i for face i, by the sum of all the distances of the m vertices from the plane:

$$DP_i = \frac{1}{\sqrt{a_i^2 + b_i^2 + c_i^2}} \sum_{j=1}^{m} |a_i x_{ij} + b_i y_{ij} + c_i z_{ij} - 1|. \quad (6)$$

The coefficients a_i, b_i, c_i of the plane can be obtained by the least square best-fit plane algorithm [19].

For an object with n faces, the total deviation from planarity DP is defined as $DP = \sum_{i=1}^{n} DP_i$.

Finally, the objective function to be minimized is defined as

$$f(z_1, z_2, ..., z_v, x_{h1}, y_{h1}, x_{h2}, y_{h2}, ..., x_{hu}, y_{hu})$$
$$= \lambda_1 S_w + \lambda_2 SDA + \lambda_3 DP \quad (7)$$

where λ_1, λ_2 and λ_3 are weighting factors, $z_1, z_2, ..., z_v$ are the depths of all the v visible and hidden vertices, and $(x_{h1}, y_{h1}), (x_{h2}, y_{h2}), ..., (x_{hu}, y_{hu})$ are the x- and y-coordinates of all the u hidden vertices. Minimizing f expresses our aim to construct a 3D object as symmetrical as possible under the constraint of planarity. We perform the minimization using a hill-climbing method [12].

5. Experimental Results

Fig. 7 shows a set of line drawings and their reconstruction results. The original line drawings are given in the first column. The results of the inference of the hidden structures are illustrated in the second column. The third and fourth columns show the recovered 3D shapes in two views each. From Fig. 7, we can see that the reconstructed objects quite accord with our visual perception. The algorithm implemented in C runs fast enough. It takes less than 1 second to do the inference and reconstruction for every line drawing.

Although the objects shown in Fig. 7 are not very complex, our work with these encouraging results is the first step towards the research on handling a wider range of objects.

6. Conclusions

We have proposed a novel approach to 3D reconstruction from single 2D line drawings without hidden lines. We first infer the hidden structure of a line drawing with the help of the constraints given in several theorems. Then we present an optimization-based method to recover the 3D shape from the complete line drawing. The objective function is developed based on perceptual symmetry and planarity. A number of encouraging results have been obtained, which demonstrate the success of the proposed approach. Our future work will focus on handling more complex polyhedra and curved objects.

Acknowledgments

References

[1] P. Min, J. Chen, and T. Funkhouser, "A 2D sketch interface for a 3D model search engine," *ACM SIGGRAPH'02 Technical Sketches*, p. 138, 2002.
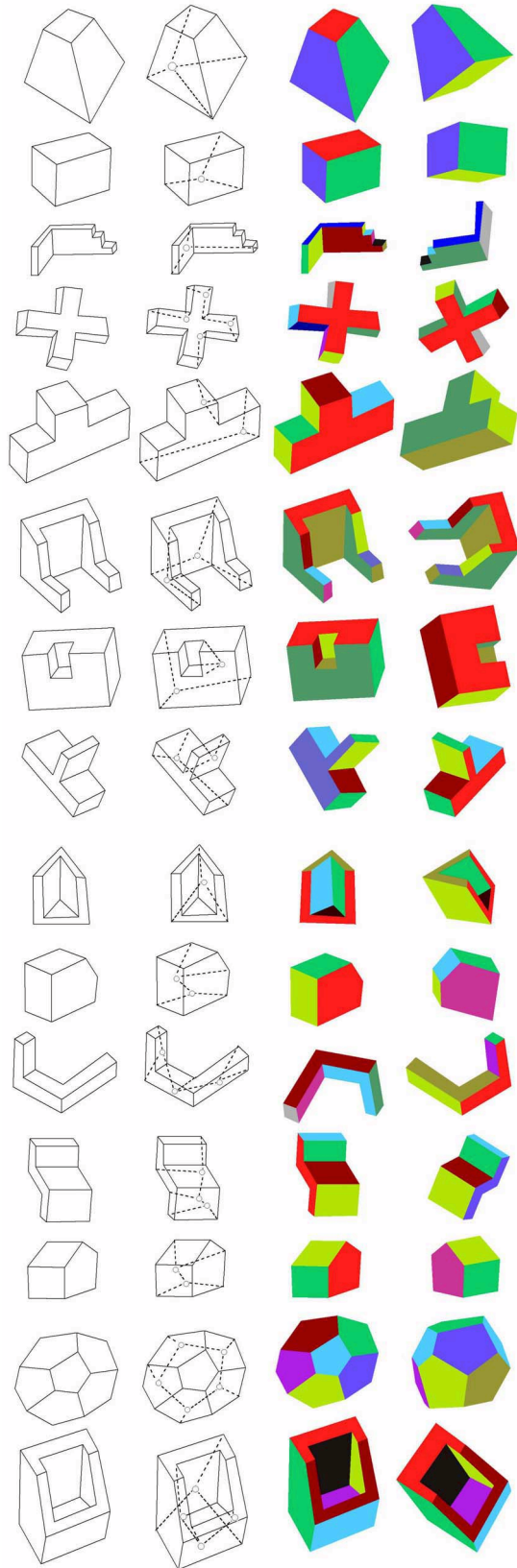
5

IEEE
COMPUTER
SOCIETY

Fig. 7: Results of the inference and reconstruction.

[2] A. Turner, D. Chapman, and A. Penn, "Sketching space," *Computers & Graphics*, vol. 24, pp. 869–879, 2000.

[3] P. Sturm and S. Maybank, "A method for interactive 3d reconstruction of piecewise planar objects from single images," *British Machine Vision Conf.*, pp. 265–274, 1999.

[4] H. Lipson and M. Shpitalni, "Optimization-based reconstruction of a 3D object from a single freehand line drawing," *Computer-Aided Design*, vol. 28, no. 8, pp. 651–663, 1996.

[5] A. Shesh and B. Chen, "Smartpaper: An interactive and user friendly sketching system," *Proc. EuroGraphics*, pp. 301–310, 2004.

[6] D. Huffman, "Impossible objects as nonsense sentences," *Machine Intelligence*, vol. 6, pp. 295–323, 1971.

[7] L. Shapiro and G. Stockman, *Computer Vision*. Prentice Hall, 2001.

[8] K. Sugihara, "A necessary and sufficient condition for a picture to represent a polyhedral scene," *IEEE Trans.Pattern Analysis and Machine Intelligence*, vol. 6, no. 5, pp. 578–586, 1984.

[9] P. A. C. Varley and R. R. Martin, "A system for constructing boundary representation solid models from a two-dimensional sketch," *IEEE Proc. Geometric Modeling and Processing*, pp. 13–30, 2000.

[10] L. Ros and F. Thomas, "Overcoming superstrictness in line drawing interpretation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 456–466, 2002.

[11] T. Marill, "Emulating the human interpretation of line-drawings as three-dimensional objects," *Int'l Journal of Computer Vision*, vol. 6, no. 2, pp. 147–161, 1991.

[12] Y. Leclerc and M. Fischler, "An optimization-based approach to the interpretation of single line drawings as 3D wire frames," *Int'l Journal of Computer Vision*, vol. 9, no. 2, pp. 113–136, 1992.

[13] K. Shoji, K. Kato, and F. Toyama, "3-D interpretation of single line drawings based on entropy minimization principle," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 2, pp. 90–95, 2001.

[14] M. Shpitalni and H. Lipson, "Identification of faces in a 2D line drawing projection of a wireframe object," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 10, pp. 1000–1012, 1996.

[15] J. Liu and Y. Lee, "A graph-based method for face identification from a single 2D line drawing," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 10, pp. 1106–1119, 2001.

[16] D. West, *Introduction to Graph Theory*. Pretice Hall, 1996.

[17] S. Palmer, *Vision Science: Photons to Phenomenology*. The MIT Press, 1999.

[18] K. Koffka, *Principles of Gestant Psychology*. Routledge & K. Paul, 1963.

[19] W.H. Press et. al, *Recipes in C++: The Art of Scientific Computing Numerical*. Cambridge University Press, 2002.