12

An Interface for 3D Conceptual Design Based on Freehand Sketching

H. Lipson, M. Shpitalni
Laboratory for Computer Graphics and CAD
Faculty of Mechanical Engineering
Technion - Israel Institute of Technology
Haifa, Israel 32000
Tel: 972-4-829-3155
Fax: 972-4-832-4533
e-mail: shefi@hitech.technion.ac.il

Abstract

This research proposes and implements a new user interface, based on freehand sketching, for three-dimensional conceptual design. There are several difficulties in implementing this kind of interface. The primary difficulty is that the reconstruction is mathematically indeterminate because the two-dimensional sketch lacks three-dimensional depth information. Inaccuracies inherent in the freehand sketch intensify the difficulty. The proposed reconstruction algorithm operates in several stages. First, the raw sketch is analyzed, and basic geometrical entities (lines, elliptic arcs and corners) are identified and smoothed. These entities are then linked to form an edge-vertex graph representing the two-dimensional topology of the object's *projection*. The three-dimensional object is then reconstructed using implicit spatial cues in the sketch plane. These cues originate from three main sources: (a) geometrical regularities, (b) sketch topology, and (c) statistical distribution of entities in the sketch plane. This paper proposes a sketch-based interface, discusses the underlying algorithms and provides examples from a working implementation.

Keywords

Line drawing interpretation, CAD interface, 3D object reconstruction.

1. INTRODUCTION

Sketching appears to be a natural communication language enabling faster conveyance of qualitative information while not impeding the creativity of the user or disrupting the flow of ideas. Despite these indications, current CAD/CAM software has not meaningfully implemented user interfaces based on freehand sketching. In this paper, we propose and demonstrate a new approach to a user interface for three-dimensional conceptual design, based on freehand sketching. Using this interface, a designer is able to construct a three-dimensional model by freehand sketching. Later specification of accurate dimensions will be possible.

Implementation of this type of interface faces several major hurdles. The first relates to the input stage, in which the sketch is inherently inaccurate in many respects and requires smoothing, classification and linking. The next hurdle is that a three-dimensional object must

M. J. Pratt et al. (eds.), *Product Modeling for Computer Integrated Design and Manufacture*
© IFIP International Federation for Information Processing 1997

be reconstructed from the single two-dimensional sketch. Although human observers usually have no difficulty understanding spatial information depicted by a flat drawing, this step is indeterminate mathematically. Since the two-dimensional sketch does not contain depth information, the three-dimensional object must be reconstructed using implicit spatial cues in the sketch plane. The information originates mainly from geometrical regularities, face identification, and statistical distribution of entities in the sketch plane. Once this stage is complete, the system can proceed to confirm the correctness of the solution and perhaps can then further refine the sketch using optimization.

This paper discusses the concept of sketching as means of conveying conceptual geometrical information and provides a brief outline of a working implementation. It is the belief of the authors that the merger of natural sketching with CAD capabilities can bring CAD to the early stages of design where pencil and paper still rule today.

2. BACKGROUND

Recent studies (Ullman et al, 1990; Jenkins and Martin, 1992; Puttré, 1993) of mechanical design process and methodology emphasize the importance of sketching, especially at the conceptual stage. Ullman et al (1990) provide strong evidence that drawing is the preferred method of external representation and that sketching is an important form of representation in mechanical design. As a natural descriptive language, sketching allows faster conveyance of qualitative information; designers often use this form of communication to convey technical information among themselves.

When using a CAD system, the user is forced to describe ideas explicitly in terms of primitives and tools supported by the software. This requirement calls for a cognitive conversion process, where abstract information is to be translated into an accurate descriptive language. This additional cognitive load is detrimental to the design process. In addition, the modeled idea may still require some additional refinement especially at the conceptual design stages, and it is not necessary that the description be as refined and accurate as demanded by current CAD systems. Another aspect of sketching is speed; during the conceptual design stage the designer employs short term memory which is fast but limited in capacity. A CAD tool for this stage must therefore be appropriately fast. Menus, toolbars and selections which are not part of natural sketching slow down the drawing action to a point where they impede creative flow and impair the cognitive problem-solving process taking place at that instant.

Ideally, a conceptual design system involving sketching should feature the following capabilities:

- Interpreting freehand sketches as three-dimensional models, allowing the user to gradually sketch a model, part by part, from a convenient viewpoint each time. The outcome of this phase is a conceptual (inaccurate) 3D model.
- Allowing designers to communicate among themselves using interpreted 3D sketching. This is especially useful for communicating spatial concepts among remote designers in a concurrent engineering setting.
- Supplying a natural interface for converting a conceptual 3D inaccurate model into an accurate 3D solid model by supplying the necessary dimensions. This capability should be achieved using variational geometry techniques combined with automatic inference of implicit relationships within the conceptual model.
- Allowing solid editing and feature specification by sketching on top of existing solids.

The above applications require the following two basic capabilities: (a) reconstruction of a conceptual three-dimensional object from a single sketch and (b) inference and imposition of constraints on the conceptual model. The structure of the proposed system is depicted in Figure 1.

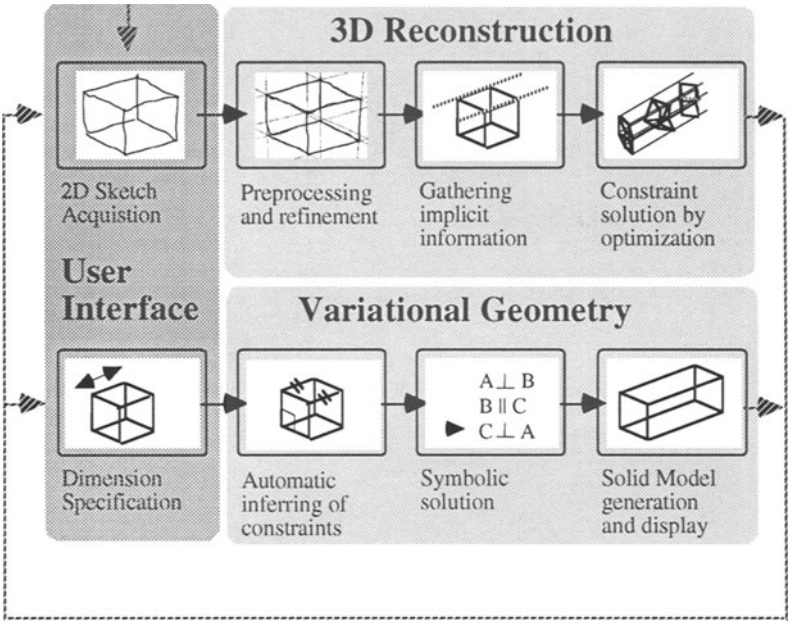


Figure 1 Overview of system structure.

This paper focuses on the first stage, which is the process of reconstructing a three-dimensional object from a single freehand sketch. The algorithm developed to perform the reconstruction consists of several stages: In the first stage, the raw sketch is analyzed and basic geometrical entities (lines, elliptic arcs and corners) are identified and smoothed. These entities are then linked to form an edge-vertex graph representing the two-dimensional topology of the *projection* of the object. In the second stage, implicit information regarding the three-dimensional model is gathered from the sketch, allowing formulation of a set of soft constraints. The reconstruction is then sought by optimizing the shape of the object until most of the constraints are met. The reconstruction can then be confirmed and refined.

Following is a brief literature review of the various subjects involved in reconstruction. The algorithm is then described phase by phase, and examples of a working implementation are provided. A more elaborate description of the implementation can be found in articles by Lipson/Shpitalni (1995).

3. LITERATURE REVIEW

Sketch reconstruction began with sketch stroke analysis. Many methods have been used to analyze line drawings and convert image data into more meaningful geometrical information (Smith, 1987). However, analysis of image data is fundamentally different from analysis of on-line sketching strokes. Strokes have been classified into entity types using curvature analysis (Jenkins and Martin, 1992), inspection of local angles at points along the stroke (Kato et al, 1982), correlation with predefined templates (Spur and Jansen, 1984), and artificial

neural nets (Koo and Fernandez, 1993). In some sketchers, the user can or must explicitly indicate which type of entity was intended (Eggli et al, 1995). Classification should be distinguished from the pure smoothing employed by "sketch-pad" applications. Smoothing is typically accomplished by averaging, convoluting, or fitting to Bezier, B-Spline and conic curves (Bookstein, 1979).

Work done on reconstruction of a 3D object from three (and sometimes two) orthogonal projections is surveyed by Nagendra and Gujar (1988). Another survey dealing with 3D reconstruction methods from both single views and multiple views is provided by Wang and Grinstein (1993).

Single view scenes with hidden lines *removed* have been studied qualitatively by Huffman (1971) and Clowes (1971) and quantitatively by others (Kanade, 1980; Sugihara, 1986). These studies have resulted in various line labeling schemes based on junction libraries. An optimization-based approach for reconstruction from an accurate view has been suggested by Marill (1991). In his work, the depth of vertices of a body represented in a line drawing projection is optimized until the minimum-standard-deviation-of-angles (MSDA) between line pairs at junctions is reached. This approach has been amended by Leclerc and Fiscler (1992) to consider face planarity as well; however, these two criteria alone appear to be sufficient only for reconstructing a limited set of object types. An interactive reconstruction system based on line labeling has been implemented by Lamb and Bandopadhay (1990). Marti et al (1993) have dealt with interpretation of sketches of origami-world objects with hidden lines visible. A recent work by Grimstead and Martin (1995) constructs a trihedral object from an orthographic view with hidden lines removed.

The proposed algorithm attempts to provide a basis for reconstructing a larger variety of object types depicted in inaccurate sketches with hidden lines visible, as is often the case in mechanical engineering drawings.

4. PROPOSED METHOD

4.1 Sketch preprocessor

In the classification stage, the sketcher is required to accept sketch strokes and classify them into various well-defined geometrical entities. Line drawings of manmade objects often exhibit instances of straight lines, circular arcs and ellipses, all of which are conic sections. Therefore, a fit to a conic section equation is used for smoothing and classification. A fit to a hyperbola allows for corner detection.

The conic equation fit is obtained using linear regression on the six coefficients of the conic-section equation. (See Shpitalni and Lipson (1995a) for the mathematical details of the regression.) In Figure 2, some examples of conic sections (dashed) fitted to strokes (noisy solid) are shown.

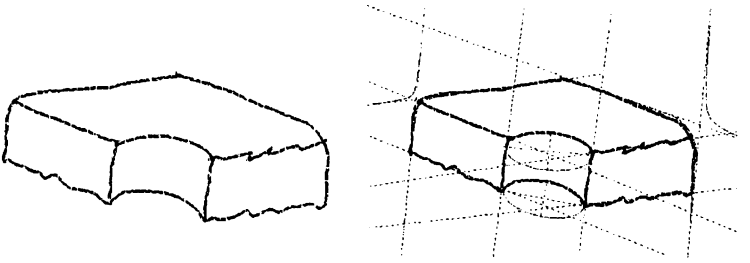


Figure 2 Some examples of conic fitting to strokes.

4.2 Reconstruction using implicit 3D information

The processed sketch, now a two-dimensional edge-vertex graph, is assumed to represent a projection of a general object, in wireframe representation, as seen from some arbitrary point of view. Since the 2D line drawing lacks depth information, the reconstruction process is non-deterministic. A simple line drawing can represent the projections of an infinite number of possible 3D objects (Figure 3). However, implicit three-dimensional information can be extracted from the line drawing, thus making it possible to reconstruct the object most likely represented by the input drawing, as a human observer would understand it.

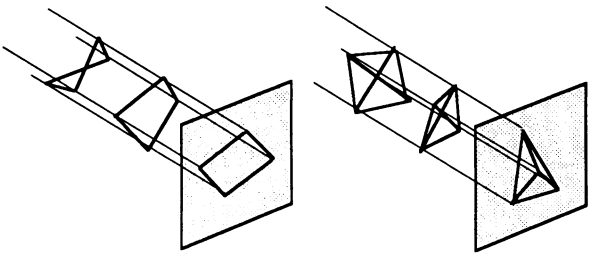


Figure 3 A single 2D sketch can be a projection of an infinite number of 3D objects.

The missing information is the depth coordinate of each vertex of the object. Implicit information in the sketch is used to formulate constraints between these coordinates. The constraints are gathered, and the depth coordinates of the object are then optimized until they comply with most of the constraints. The implicit three-dimensional information residing in the two-dimensional sketch can be classified into the following three categories:

- Image regularities. These are various geometrical regularities observable in the sketch. These regularities are not accidental, but usually reflect some real spatial relationship. The derived relationship can be used to formulate constraints between vertices of the sought object.
- Face identification. An important "clue" to the understanding of a drawing of a three-dimensional object is the identification of edge circuits corresponding to the object's faces. These face arrangements can be used to formulate constraints between vertices of the object.
- Angular distribution of lines. A statistical analysis of the distribution of lines in the sketch can provide some additional information regarding the general shape and orientation of the object. Again, this information can be used to formulate constraints.

The basic assumptions and requirements are:

- The input to the system consists only of the single two-dimensional line drawing which is entered as a graph of connected entities. Each edge of the graph corresponds to exactly one edge of the depicted object. Each vertex of the graph corresponds to exactly one vertex of the depicted object.
- The input projection represents a wireframe model of a general object that may be manifold, non-manifold or an assembly of both types of objects. No information is provided to the system about the three-dimensional object itself, its type, or its position relative to the viewpoint.
- The projection is drawn from a general - non-accidental - viewpoint that reveals all edges and vertices. That is, none of the object's edges or vertices coincide accidentally, and none of them accidentally appear to be joined in the projection.

- All drawn lines and curves in the projection represent real edges, silhouette curves or intersections of faces in the 3D object. No shadow lines or surface marks are allowed.

Image regularities

Primary implicit three-dimensional information resides in special entity configurations, termed *image regularities* (Kanade, 1980). Image regularities are special geometrical relationships between individual entities or within groups of entities. Image regularities usually do not appear in the drawing accidentally, but rather correspond to some real geometrical regularities existing in the 3D object. An example of a typical regularity is *parallelism*. The rule for the *parallelism* regularity is that if two lines are parallel in the sketch plane, they *probably* represent parallel lines in the 3D object, although mathematically this is not necessarily the case. It should be emphasized that interpretation of image regularities is not heuristic but is based on a profound statistical analysis; two parallel lines in space appear parallel when viewed from *any* viewpoint, whereas two crossing lines can appear parallel only when viewed from a limited subset of viewpoints. It is therefore much more probable for a two-parallel-line projection to represent parallel lines in space. In fact, humans make this assumption so naturally that images defying such relationships appear strange (excellently demonstrated by M. C. Escher's drawings).

Image regularities can be used to create a set of equations which are then solved to obtain the 3D object from an accurate drawing. This approach, however, has been found inappropriate for practical use when dealing with inaccurate drawing sources. The basic problems with regularities in such cases are: (a) they do not *necessarily* represent a 3D relationship because they might be accidental, and (b) in an inaccurate drawing where parallel lines, for instance, are not exactly parallel in the drawing, it may be difficult to detect these regularities with certainty. Therefore, the information available from image regularities is used as soft constraints which do not necessarily require full compliance.

In the proposed algorithm, a set of soft constraints has been constructed by checking all regularities over all the combinations of entities in the topological graph. Each constraint is weighted according to the probability that the regularity detected in the sketch indeed represents a 3D geometrical relationship. For a proposed 3D reconstruction, the regularities are evaluated using mathematical terms, multiplied by their weight-coefficient and summed to produce an overall compliance function. This compliance function estimates how well the specific 3D reconstruction conforms with the regularities identified in its 2D projection. A 3D configuration of vertices is then searched for by optimizing the compliance function. This approach tolerates inaccuracies while arriving at a reconstruction that will approximately satisfy *most* of the regularities. The success of this procedure depends on the variety of regularities used, their mathematical formulation, and the optimization procedure itself.

Following is a brief list of regularities used. Refer to Lipson and Shpitalni (1995) for more details.

- **Face planarity.** A face contour consisting entirely of straight lines reflects a planar face in 3D.
- **Line parallelism.** A parallel pair of lines in the sketch plane reflects parallelism in space.
- **Line verticality.** A line that is vertical in the sketch plane (parallel to the y-axis of the drawing page) is vertical in space.
- **Isometry.** Lengths of entities in the 3D model are uniformly proportional to their lengths in the sketch plane.
- **Corner orthogonality.** A junction of three lines that mathematically qualifies as a projection of a 3D orthogonal corner is orthogonal in space.
- **Skewed facial orthogonality.** A face contour that shows skewed orthogonality is probably orthogonal in space.
- **Skewed facial symmetry.** A face showing skewed symmetry in 2D denotes a truly symmetrical face in 3D.
- **Line orthogonality.** All line pairs in a junction *except those that are collinear* are perpendicular in 3D.

- **Face perpendicularity.** All adjacent faces must be perpendicular. This criterion serves to initially "inflate" the flat projection to a convex shape in 3D space from which optimization is more easily achieved.
- **Prismatic face.** A face joining two parallel elliptic arcs in the projection plane is prismatic in space. Being prismatic, it can resolve both cylindricality and planarity.
- **Line collinearity.** Lines collinear in the sketch plane are collinear in space.
- **Planarity of skewed chains.** When a *chain* of entities is found to exhibit skewed symmetry of skewed orthogonality, both *planarity* and *symmetry* or *orthogonality* are required.

Face identification

An important key to understanding a line drawing and accelerating the reconstruction process is the initial phase of analyzing the 2D projection and identifying the circuits corresponding to the actual faces comprising the 3D object. Those circuits corresponding to actual faces should be identified and distinguished from the many other circuits existing in the drawing. This identification phase is complex when the object is displayed in wireframe representation so that all of its edges appear in the projection. Such cases are common in the initial conceptual design stage of mechanical parts. Identification becomes even more complex when general objects including non-manifolds are considered. In Figure 4, the original input consisting of two-dimensional edge-vertex graphs appears on the left. On the right, these graphs have been broken down into the edge-circuits corresponding to the actual faces of the depicted 3D object. It is important to note that the graphs on the left and those on the right are all two-dimensional and that no information is available regarding the three-dimensional objects they represent.

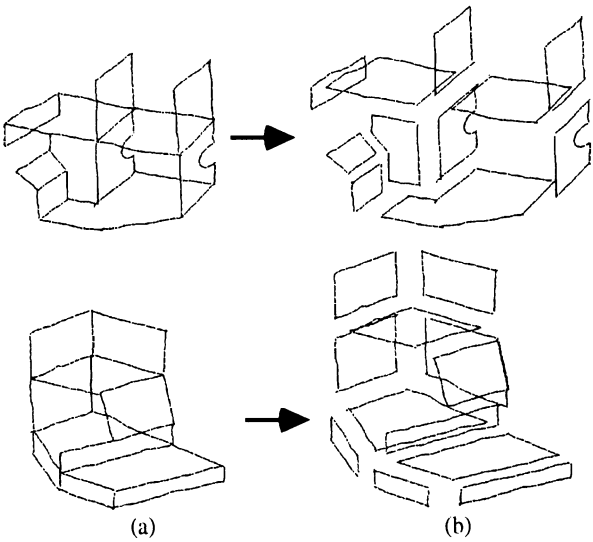


Figure 4 (a) The given 2D projection of some object, and (b) the identified face circuits in the 2D projection plane.

Identification of faces can be simplified and solved analytically if the object represented by the sketch is known a priori to be a manifold object of genus zero. In this case, faces can be identified using a planar configuration of the graph. The correspondence between the planar

graph and the topology of the 3D object is termed a *plane model*, a labeled plane figure whose edges and vertices correspond to those of the real 3D object. A formal definition of plane models is presented by Mantyla (1988). The given edge-vertex graph is, in fact, a plane model of the object it represents. Faces are identifiable when the graph is redrawn with no intersection between edges, i.e., when the input drawing is embedded as a *plane graph*. At this stage, all the faces of the graph (minimal, empty and closed circuits of edges) correspond to faces of the object represented by the current plane model. In addition, the *exterior* face of the plane graph defined by the outermost circuit of entities also corresponds to a face of the object. If the graph is planar but not 3-vertex-connected, several different embeddings corresponding to different face configurations can be found. The number of such embeddings is exponentially proportional to the number of components in the decomposition of the given graph into 3-vertex-connected components that contain circuits. Image regularities can be used to select between multiple solutions. For more details and proofs of the above, refer to Shpitalni and Lipson (1995).

Face identification in general objects is far more difficult. The term *general object* refers to an object of an unknown type. This object can be either manifold or non-manifold and is composed of solids, surfaces and skeletal structures with or without holes and perhaps with several detached components. This generality precludes a unique mathematical solution. The goal is to find the most plausible solution that would have been identified by a human observer. Heuristics must therefore be introduced into the solution method to account for psychological reasoning in picture understanding.

A general object with n vertices has $O(n)$ edges and $O(n)$ faces. Its projection has m non-self-intersecting circuits of edge-entities, where m is an exponential function of n. These circuits are called *potential faces* because the faces of the object are a subset of the m circuits. Although there are exactly 2^m possible combinations of *potential faces*, human observers seem to be able to easily identify this subset. In essence, the identification procedure proceeds as follows. First, the upper bounds of the edge and vertex *ranks* are computed using basic topological relationships. (A *rank* of a vertex or edge is defined as the number of face boundaries passing through it.) Assignment of a set of potential faces complying with the estimated ranks is then sought. A face adjacency theorem (Shpitalni and Lipson, 1995) is used to reduce the search domain to a manageable size. As in the previous case, image regularities can be used to select the right solution from multiple solutions. For more details and proofs of the above refer to Shpitalni and Lipson (1995).

Angular distribution of lines

The distribution of strokes is analyzed by means of an angular distribution graph (ADG). The ADG is constructed by sampling the angle of every entity in the sketch and plotting it on an angular distribution histogram. When an angle is added to the graph, a Gaussian distribution curve with $\sigma=7$ is superpositioned onto the graph to account for the inaccuracy of the sketch. After all the curves have been superpositioned and combined, the graph is normalized with its maximum at 1.0. The resulting ADG qualitatively shows the prevailing angles in the sketch.

Strictly orthogonal objects have clear three-hump ADG's. Other objects exhibiting some orthogonality still show a few noticeable humps surrounded by angular "noise." Often it is possible to identify the principal axis system of the intended object from this graph. To do so, an axis system must be found at a spatial orientation such that it will be projected onto the sketch plane with angles corresponding to maxima in the ADG.

Once the principal axis system is identified, lines in the sketch are associated with one of the axes, and additional constraints may be formulated. A more detailed description of the use of ADG is given in Lipson and Shpitalni (1995).

5. EXAMPLES

Figure 5 provides two examples of results obtained using an implementation of the algorithm proposed in the previous sections. The figure shows the original sketch and two arbitrary views of the interpreted three-dimensional model. To illustrate correct reconstruction, hidden lines have been removed.

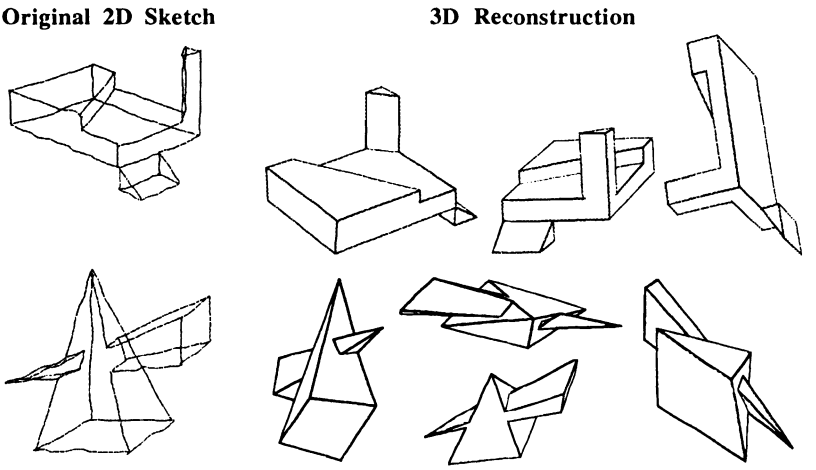


Figure 5 Examples of input sketches and their interpretations. Note that the reconstruction was performed from the sketch alone, without any prior knowledge about the 3D object represented or its type.

6. SUMMARY

This paper proposes a new type of user interface for three-dimensional mechanical/architectural CAD based on interpreting, reconstructing and refining objects drawn as freehand sketches. An algorithm was proposed for reconstruction of a 3D inaccurate model from a single freehand 2D sketch. The results achieved so far have attempted to combine image understanding, variational geometry and principles of artificial intelligence to achieve what we humans so easily do when reading a drawing. Additional work towards the dimensioning of the sketch is still needed before the system becomes practical. It is the belief of the authors that the process of sketching and subsequent placing of dimensions more closely corresponds to the human way of thinking and visualizing at the early stages of conceptual design. Although research in this area is not mature enough for incorporation into commercial software, it definitely constitutes a step towards improving CAD's weakest frontier, the user interface.

7. ACKNOWLEDGMENT

This work was carried out in partial fulfillment of the requirements for the degree of D.Sc. in Mechanical Engineering at the Technion [Hod Lipson]. This research has been supported in part by the Fund for the Promotion of Research at the Technion, Research No. 033-028.

8. REFERENCES

Bookstein, F. L. (1979) Fitting conic sections to scattered data. *Computer Graphics and Image Processing*, **9**, 56-71.

Clowes, M. B. (1971) On seeing things. *Artificial Intelligence*, **2**, 79-116.

Eggli, L., Brüderlin, B. P., and Elber, G. (1995) Sketching as a Solid Modeler Tool, Third Symposium on Solid Modeling and Applications, *ACM SIGGRAPH*, 313-321.

Grimstead, I. J. and Martin, R. R. (1995) Creating solid models from single 2D Sketches, *Proceedings of the Third Symposium on Solid Modeling Applications, ACM Siggraph*, 323-337.

Huffman, D. A. (1971) Impossible objects as nonsense sentences. *Machine Intelligence*, Edinburgh Univ Press, 295-323.

Jenkins, D. L. and Martin, R. R. (1992) Applying constraints to enforce user's intentions in free-hand 2-d sketches. *Intelligent Systems Engineering*, Autumn, 31-49.

Kanade, T. (1980) Recovery of the three dimensional shape of an object from a single view. *Artificial Intelligence*, **17**, 409-460.

Kato, O., Iwase, H., Yoshida, M., and Tanahshi, J. (1982) Interactive hand-drawn diagram input system. *Proc. IEEE Conference on Pattern Recognition and Image Processing (PRIP 82)*, Las Vegas, Nevada, 544-549.

Koo, J. C. and Fernandez, B. (1993) Geometrical error correction using hierarchical hybrid artificial neural systems. *IEEE International Conference on Neural Networks*, 232-237.

Lamb, D. and Bandopadhay, A. (1990) Interpreting a 3d object from a rough 2d line drawing, *Proc First IEEE Conf on Visualization 90*, 59-66.

Leclerc, Y. G. and Fiscler, M. A. (1992) An optimization based approach to the interpretation of single line drawings as 3d wire frames. *Int. J. of Computer Vision*, **9**(2), 113-136.

Lipson, H. and Shpitalni, M. (1995) Optimization-based reconstruction of a 3d object from a single freehand line drawing. *Computer Aided Design*, **8**, 651-663.

Lipson, H. and Shpitalni, M. (1995a) A new interface for conceptual design based on object reconstruction from a single freehand sketch, *Annals of the CIRP*, **45/1**, 133-136.

Mantyla, M. (1988) *An Introduction to Solid Modeling*, Computer Science Press.

Marill, T. (1991) Emulating the human interpretation of line drawings as three dimensional objects. *Int. J. of Computer Vision*, **6**(2), 147-161.

Marti, E., Regomcós, J., López-Krahe, J. and Villanueva, J. J. (1993) Hand line drawing interpretation as three dimensional object,. *Signal Processing*, **32**, 91-110.

Nagendra, I. V. and Gujar, U. G. (1988) 3D objects from 2D orthographic views - a survey. *Computer Graphics*, **12**(1), 111-114.

Putré, M. (1993) Gearing up for conceptual design. *Mechanical Engineering*, 46-50.

Spur, G. and Jansen, H. (1984) Automatic recognition of hand-drawn contours for CAD applications. *16th CIRP Int. Seminar on Manufacturing Systems*, 63-72.

Shpitalni, M. and Lipson, H. (1995) Identification of faces in a 2D line drawing projection of a wireframe object. Submitted (revised) to *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Shpitalni, M. and Lipson, H. (1995a) Classification of sketch strokes and corner detection using conic sections and adaptive clustering. Submitted to *ASME Journal of Mechanical Design*.

Smith, R. W. (1987) Computer processing of line images: a survey. *Pattern Recognition*, **20**(1), 7-15.

Sugihara, K. (1986) *Machine Interpretation of Line Drawings*, MIT Press.

Ullman, D. G., Wood, S., and Craig, D. (1990) The importance of drawing in the mechanical design process. *Computers & Graphics*, **14**(2), 263-274.

Wang, W. and Grinstein, G. (1993) A survey of 3D solid reconstruction from 2D projection line drawings. *Computer Graphics Forum*, **12**, 137-158.