

Using skewed mirror symmetry for optimisation-based 3D line-drawing recognition

A. Piquer*, R.R. Martín⁺ and P. Company*

*Dept. of Technology, Universitat Jaume I, E-12004, Castellon, Spain

E-mail: pvicent@tec.uji.es, pcompany@tec.uji.es

⁺ Dept. of Computer Science, Cardiff University, PO Box 916, Cardiff, CF24 3XF, UK

E-mail: ralph@cs.cf.ac.uk

Abstract

We aim to reconstruct three-dimensional polyhedral solids from axonometric-like line sketches. A new approach is proposed to make use of planes of mirror symmetry detected in sketches. Taking into account mirror symmetry of such polyhedra can significantly improve the reconstruction process. Applying symmetry as a regularity in optimisation-based reconstruction is shown to be adequate by itself, without the need for other inflation techniques or regularities. Furthermore, we show how symmetry can be used to reduce the size of the reconstruction problem, leading to a reduction in computing time.

Keywords: Planes of Symmetry. Graphics Recognition. Graphics Interpretation. Perceptual Reasoning. Sketch Understanding. Geometric Modelling. 3D Reconstruction. Mirror Symmetry.

1 Introduction

Sketching-based geometric modellers have emerged during the last decade [1-5], providing a very intuitive and easy to use interface to 3D reconstruction engines. Optimisation has been used with some success in this sketch-based modelling approach. One of the main challenges in optimisation-based 3D reconstruction is the mathematical formulation of perceptual cues, also called artefacts, or regularities. At present, they lead to poorly defined objective functions, and, hence the success rate is below the expectations of users. In addition, optimisation processes are bottlenecks for an interactive session, with clearly unacceptable calculation times (typically up to several minutes).

Using symmetry improves optimisation performance, since it is a fundamental concept which human visual perception utilizes [6]. Furthermore, many man-made objects are symmetric, both because this makes them easier to interpret and manufacture, and because of functional and aesthetic requirements. Studies have shown that a large proportion of industrial components have some symmetry [7].

In a previous paper, we presented a novel method for determining the skewed planes of symmetry of polyhedral objects, starting with a two-dimensional axonometric-like view [8]. This algorithm is aimed at single view reconstruction: 3D reconstruction from multiple orthographic views requires a radically different approach. The nature of shapes that can be reconstructed with this optimisation-based approach extends, at present, to polyhedral blocks (manifolds), origami objects (sheet metal) and wire frame models (non-manifold objects).

In this paper, we present a new optimisation regularity based on a novel formulation of model symmetry. Its advantage in the success rate of optimisation approaches are discussed. Furthermore, we discuss the advantages of dealing with model symmetries as constraints, instead of as regularities, in order to significantly reduce the calculation time for optimisation-based 3D reconstruction.

2 Related work

Various research exists focused on using freehand drawings and sketches as a way to obtain 3D geometric models via input devices, like graphic tablets and tablet PCs (Figure 1). *Gestural modelling* systems are one approach. Predefined gestures encode a set of geometric manipulation operations [9-13]. *Reconstructive modelling* is an alternative, where geometric reconstruction techniques build the object's geometry from a sketch that represents a projection of the object [12, 14-19]. Such systems can be categorized as *projective* (when projection invariants are extensively used to recover as much information on the object as possible), or *perceptual* (based on the way human perception creates very rich three-dimensional scenes from retinal images). However, often both categories are used together. For instance, optimisation approaches, which use perceptual rules to interpret regularities, also use projective principles like *inflation* (Figure 1d), which is a kind of inverse projection process.

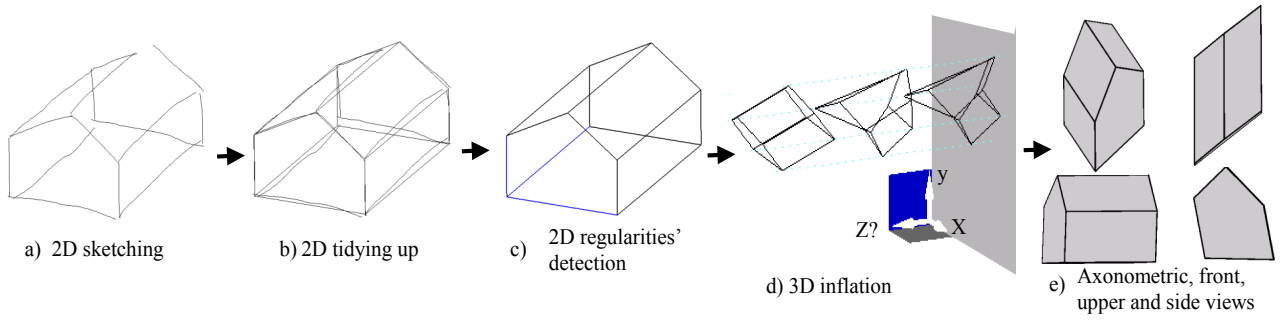


Figure 1. From sketch to 3D model

Optimisation-based 3D reconstruction is discussed in [20-27]. We simply note that optimisation-based reconstruction is a two-step strategy where an inflation process is used to transform 2-D line drawings into 3D models by choosing the appropriate z coordinate for every vertex through an *optimisation* process. In other words, coordinates are related to a so-called inflation-coordinate-system, where the xy coordinate plane is the drawing plane and the infinite set of three-dimensional objects whose orthogonal projection matches the given line drawing (i.e. the *orthographic extension*) is defined by the set of z coordinates of all model vertices: $\mathbf{z} = (z_1, z_2, \dots, z_n)$. Only a few or even just one of these models fits perceptual requirements, i.e. many of the models in the orthographic extension are *tangles* (Figure 1d), models that contain twisted, non-planar faces. The psychologically plausible model is selected by *optimising* a figure of merit, or objective function, defined as a weighted sum of contributions from perceptual cues, as suggested by Equation 1.

$$F(z) = \sum \alpha_j R_j(z) \quad \text{Equation 1}$$

Various researchers have studied such perceptual cues or *regularities* [22, 24, 28, 29]. Currently, inflation approaches succeed in 3D reconstruction whenever the input drawings are *geometrically correct*, i.e. they correspond to an orthogonal projection of a real shape. However, if input drawings are approximate sketches (see Figure 1a), with imprecise coordinates, inflation gives distorted or even tangled shapes. Sketching interfaces typically perform on-line conversion of sketched lines into straight segments, then join segments whose ends are close to each other, then ensure parallelism where needed, and so on (Figure 1b). A detailed description of one such *analyser* is given in [14]. Additionally, on-line capabilities may be complemented by off-line tidying of the complete sketch after the user finishes drawing. This can perform a more detailed analysis of the user's sketch, giving a better opportunity to discover those *design intents* that are difficult to detect on-line.

In this paper, we are concerned with the particular regularity of *model symmetry*, which is often an explicit design intent. The effects of its absence can be clearly seen in the front, side and upper views of the output 3-D model of Figure 1e. Previous work exists on the detection and application of symmetry to object modelling and recognition [30].

In our approach, a symmetry plane is obtained from a set of symmetry lines. Finding model symmetry planes requires the prior detection of axes of skewed facial symmetry, which, in turn, requires edges bounding faces to be previously recognised in the two-dimensional image [8]. We need to search for 2-D circuits representing intersections of the model with a symmetry plane. Each skewed plane of symmetry is made up of skewed axes of symmetry and edges of various faces. A closed sequence of axes and edges that meet each other must be found, forming a planar polygon. The plane that contains that *polygon of symmetry* represents a plane of symmetry (Figure 2). We perform our computation in 2D, so the resulting output comprises *skewed planes of symmetry*: real planes of symmetry viewed from some (unknown) viewing direction [8].

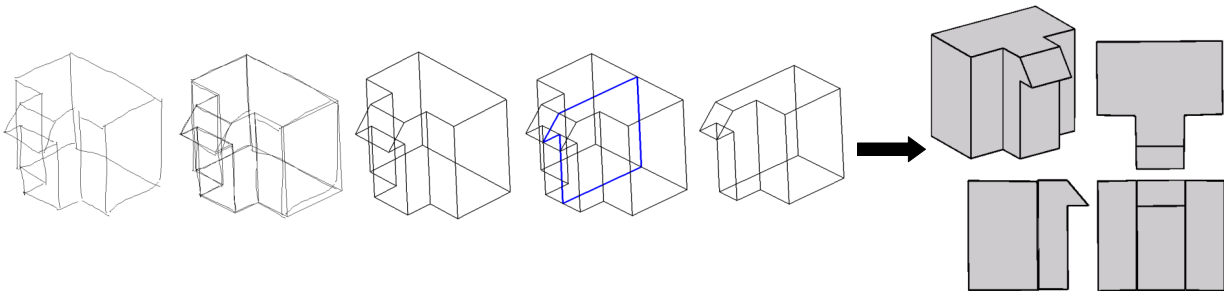


Figure 2. Input sketch, wire-frame line drawing and symmetry plane, represented as a 2-D circuit (left) and the final reconstructed 3-D model (right).

2.1 Previous Approaches

The simplest way to use symmetry planes in optimisation-based reconstruction is to consider them as special (virtual) *faces* of the final model. In such an approach, the symmetry planes are optimised using the same *face planarity* regularity which is applied to other faces.

Suppose the symmetry plane has coefficients A , B , C and D in the equation $Ax + By + Cz + D = 0$. In our implementation, the first three coefficients were estimated by computing the respective areas of the projection of the virtual face in the yz plane (for A), xz plane (for B), and xy plane (for C). D was estimated by simply substituting the coordinates of one of the vertices into the equation. Poor results were obtained. By using symmetry planes, calculation times increased, while the final model precision was not enhanced appreciably.

An alternative approach is to use as a regularity criterion the deviation from the theoretically equal distances each pair of symmetric vertices ought to have from the symmetry plane. The n pairs of signed distances of symmetric vertices should satisfy Equation 2:

$$\sum_{i=0}^{n-1} \text{distance}(v_i, \text{plane}) = 0 \quad \text{Equation 2}$$

However, this is too simplistic. Equation 2 by itself does not ensure that each pair of symmetric vertices is connected by a line orthogonal to the symmetry plane: *any* point on a given plane parallel to the symmetry plane has the same distance from it. Checking orthogonality of these lines requires identifying every pair of symmetrical vertices, and the calculation time becomes very high. (Actually, a better approach is to sum the *squares* of the distances, but this would not affect the above remarks.)

3 Overview of our new approach

Once the symmetry planes have been determined [8], in order to deal with symmetry, it is convenient to introduce a new coordinate system, a so-called *symmetry system*, as explained in the next Section. This allows the symmetry condition to be formulated in a compact and convenient way.

To do this, symmetric vertices have to be properly paired or *matched*. This is simple whenever a vertex in one half is connected to its pair through an edge that crosses the symmetry plane, with midpoint on the plane, as for example in e_1 , e_2 and e_3 in Figure 3(a), or e_4 and e_5 in Figure 3(b).

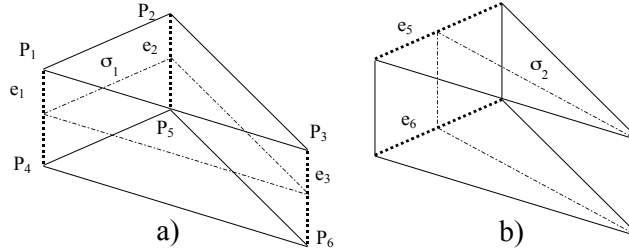


Figure 3. Determination of subsets of symmetrical vertices.

However, the general matching problem is more difficult, and needs to be solved for a practical formulation of *bilateral symmetry regularity*.

Note that the presence of a symmetry plane can be used to reduce the size of the inflation problem. Because of the symmetry, only one half of the object needs to be inflated. When multiple symmetry planes exist, further reductions of problem size can be obtained. Overall, a much smaller problem may be result.

4 Symmetry coordinate system

We introduce a different *symmetry coordinate system* for each symmetry plane, in which the symmetry plane is the xy plane. One such system $(O_{SI} X_{SI} Y_{SI} Z_{SI})$, together with the inflation coordinate system $(O_I X_I Y_I Z_I)$ and the model coordinate system $(O_M X_M Y_M Z_M)$, is illustrated in Figure 4. The origin is placed at the centroid of the planar symmetry polygon, which is invariant and so ensures that the origin of all symmetry systems is the same. X_{SI} and Y_{SI} are aligned with two model coordinate system axes, whenever such a system has already been determined (this problem is related to detecting main directions, and is beyond the scope of this paper [24]). Z_{SI} is chosen to give a right hand system.

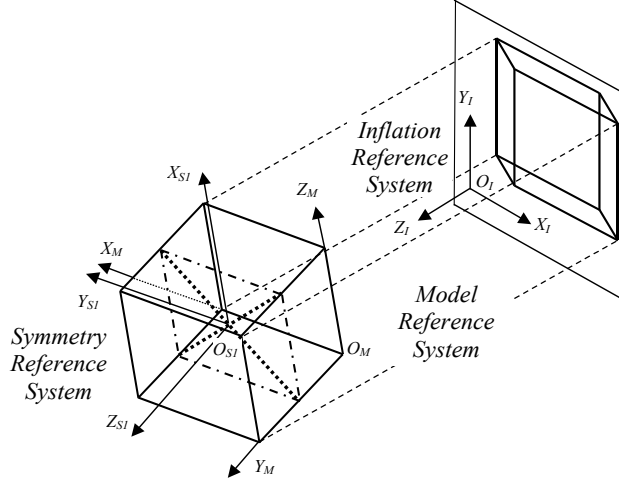


Figure 4. Inflation versus model and symmetry reference systems.

The best-fit plane defined by all vertices of the symmetry polygon is initially calculated using the method given at the start of Section 2.1.

Let (x_c, y_c, z_c) be the coordinates of the centroid of the symmetry polygon. Let (u_1, u_2, u_3) be a unit vector normal to the symmetry plane, and let R be defined by: $R = \sqrt{u_2^2 + u_3^2}$. We denote the coordinates of a vertex P in the model by (x_p, y_p, z_p) when they are in the inflation coordinate system, and by (x'_p, y'_p, z'_p) when they are in the symmetry coordinate system. Then, the homogeneous transformation between these systems is:

$$(x'_p, y'_p, z'_p, 1) = (x_p, y_p, z_p, 1) \cdot \begin{bmatrix} R & 0 & u_1 & 0 \\ -u_1 u_2 / R & u_3 / R & u_2 & 0 \\ -u_1 u_3 / R & -u_2 / R & u_3 & 0 \\ -x_c R + \frac{u_1 u_2}{R} y_c + \frac{u_1 u_3}{R} z_c & -\frac{u_3}{R} y_c + \frac{u_2}{R} z_c & -u_1 x_c - u_2 y_c - u_3 z_c & 1 \end{bmatrix} \quad \text{Equation 3}$$

5 Symmetry condition

The set of n vertices of a symmetric model can be renumbered to give three subsets satisfying:

$$\begin{aligned} (1, 2, \dots, n) &= (1, 2, \dots, m), & \text{with } z'_1, \dots, z'_m > 0 \\ & (m+1, m+2, \dots, 2m), & \text{with } z'_{m+1}, \dots, z'_{2m} < 0 \\ & (2m+1, 2m+2, \dots, n) & \text{with } z'_{2m+1}, \dots, z'_n = 0 \end{aligned}$$

Equation 4

In other words, m points lie on one side of the symmetry plane, another m points lie on the opposite side, and $n-2m$ points are contained in the symmetry plane (m may be 0). The points in the second subset are paired with the points in the first subset, so that point 1 corresponds to point $m+1$ and so on. This gives the following conditions:

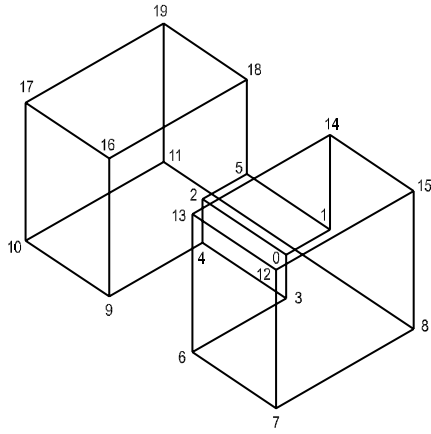
$$\begin{aligned} x'_1 &= x'_{m+1}, \dots, x'_m = x'_{2m} \\ y'_1 &= y'_{m+1}, \dots, y'_m = y'_{2m} \\ z'_1 &= -z'_{m+1}, \dots, z'_m = -z'_{2m} \end{aligned}$$

Equation 5

This is the simplest formulation of the symmetry condition for vertices, and is the basic formulation of our *bilateral symmetry regularity*.

6 Matching symmetric vertices

Once the symmetry plane has been detected, if a full symmetry formulation is desired, a matching vertex must be located for each vertex. Our algorithm fills the elements of a $2 \times n$ matrix V . In $V(l, i)$ the vertex symmetric to vertex i



Face No.	Vertices on faces
Face 0	0, 2, 5, 1
Face 1	2, 0, 3, 4
Face 2	18, 16, 17, 19
Face 3	6, 13, 12, 7
Face 4	7, 12, 15, 8
Face 5	0, 1, 14, 13, 6, 3
Face 6	12, 13, 14, 15
Face 7	17, 16, 9, 10
Face 8	5, 2, 4, 9, 16, 18
Face 9	17, 10, 11, 19
Face 10	15, 14, 1, 5, 18, 19, 11, 8
Face 11	11, 8, 7, 6, 3, 4, 9, 10

$$V = \begin{pmatrix} 2 & 5 & 0 & 4 & 3 & 1 & 9 & 10 & 11 & 6 & 7 & 8 & 17 & 16 & 18 & 19 & 13 & 12 & 14 & 15 \\ +1 & +1 & -1 & +1 & -1 & -1 & +1 & +1 & +1 & -1 & -1 & -1 & +1 & +1 & +1 & +1 & -1 & -1 & -1 & -1 \end{pmatrix}$$

Figure 5. Matching symmetrical vertices.

(its match) is stored. The second row $V(2,i)$ stores a label indicating which subset vertex i belongs to. The label +1 is assigned to one of the sides of the symmetry plane, label -1 is assigned to the other side, and label 0 corresponds to vertices on the symmetry plane.

6.1 Identification of the two sides of a symmetrical model

Identification starts at one of the faces divided by the symmetry plane. One of the sides of the symmetry plane is arbitrarily assigned +1 without loss of generality. Next, adjacent faces that are also cut by the symmetry plane are evaluated in turn. In this way, common vertices can be used to guarantee consistency in labelling propagation.

In a third step, the process is consistently extended to the rest of the faces. All vertices in such faces are assigned the same label as the one assigned to previously labelled vertices of the face.

Vertices belonging to the symmetry polygon are finally visited and labelled 0.

6.2 Matching vertices

Next, symmetry axes are considered in turn, and the vertices of the faces they belong to are explored. Their vertices are matched through a simultaneous clockwise-anticlockwise scan. For instance, in face 11 (Figure 6), if evaluation starts at the a axis, which cuts the mid-point of edge 11-8, it is easy to fix vertices 11 and 8 as a pair of symmetrical vertices. The vertices of one half of the face starting at this edge are scanned clockwise (8-7-6-3) and the other half anticlockwise (11-10-9-4), and the two resulting lists are paired to match all the vertices in the face. The same procedure is applied to each face sharing a line with the symmetry polygon. See Figure 6.

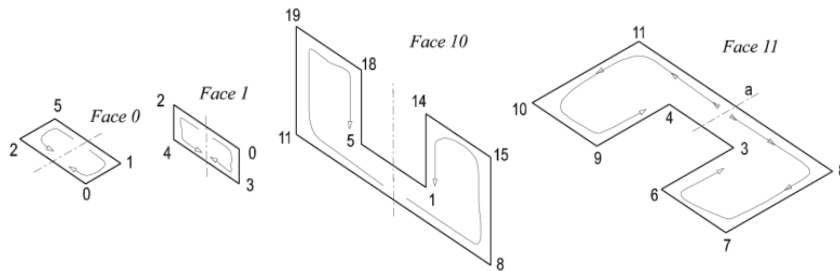


Figure 6. Faces with axis belonging to the polygon of symmetry.

Once all such faces have been evaluated, we continue with the remaining faces. In that case, symmetry between vertices involves vertices of different faces. Thus, the objective is to find symmetrical faces.

The list of such incomplete faces is scanned by searching for a face with at least three vertices already labelled ($v_i, v_j, v_k \subset F_m$); two of these (at least) must be at the ends of an edge shared with an already-processed face. Three non-collinear points determine a plane, so the vertices symmetric to (v_i, v_j, v_k) must determine a symmetric face ($v'_i, v'_j, v'_k \subset F'_m$). The correspondence between the vertices must still be found, and is done by considering the vertices at the ends of the edges shared with processed faces. (see Figure 7: for example 6-7 in *face 3*, and 9-10 in *face 7*). The relative ordering of vertices around the faces may be the same or opposite, and must be determined. For instance, the correspondence between *faces 3* and *7* has oppositely ordered vertices (9-10-17-16 and 6-7-12-13). However, *faces 9* and *4* have a correspondence (19-17-10-11 and 15-12-7-8) with vertices arranged in the same order.

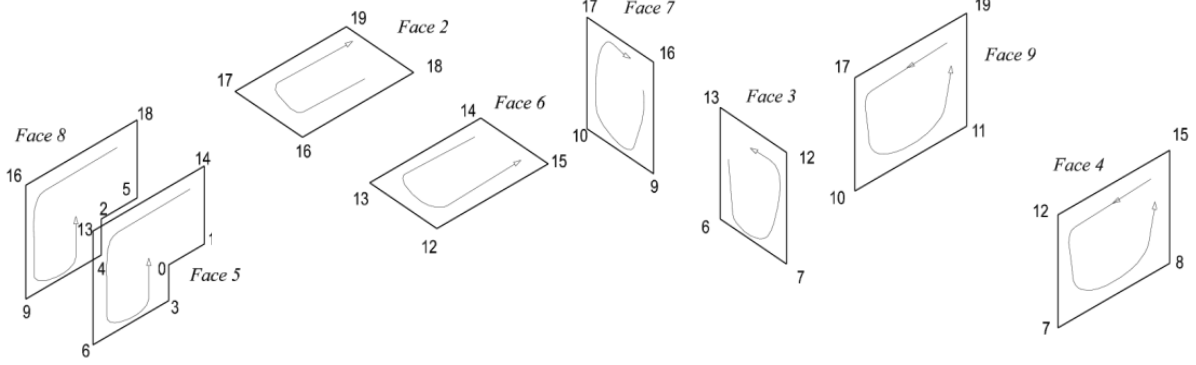


Figure 7. Faces without axis belonging to the polygon of symmetry

7 Symmetry as regularity

Once the symmetry polygon has been detected and matching of vertices completed, the symmetry regularity is easy to formulate. In reconstruction by optimisation, every regularity has a *cost*, and the objective is to minimize the sum of costs. A cost is chosen to be zero when the desired condition is achieved, and differs from zero as we go away from this ideal. Hence, the bilateral symmetry condition formulated in Equation 5, can be reformulated as the standard deviation (σ^2) of the difference of coordinates of all pairs of symmetrical vertices:

$$Cost_j = \sigma^2(dif_x') + \sigma^2(dif_y') + \sigma^2(sum_z') \quad \text{Equation 6}$$

where

$$dif_x' = (x'_i - x'_{m+i})$$

$$dif_y' = (y'_i - y'_{m+i}) \quad \text{for every vertex } i=1 \dots m.$$

$$sum_z' = (z'_i + z'_{m+i})$$

and

$$dif_x' = dif_y' = 0,$$

$$sum_z' = z'_i \quad \text{for every vertex } i=2m+1 \dots n$$

The standard deviations are calculated with an assumed mean equal to zero. Using the real mean does not improve reconstruction results and the objective of the function is distorted. The real objective is to find variables in which the mean of every summand is also zero. When every pair of vertices fulfils the condition, the cost of the regularity reaches the minimum.

The cost defined above is for a single plane of symmetry. The total cost for a model with r planes of symmetry is defined as:

$$Cost_of_Symmetry = \frac{1}{r} \sum_{j=1}^r Cost_j \quad \text{Equation 7}$$

8 Model rebuilt from the half part

Faster reconstruction can be based on the idea that, knowing the symmetric pairs of vertices, information about just one half of the model in the drawing can be used for reconstruction, and once a half model has been obtained, then symmetry can be applied to obtain the rebuilt complete model. This reduces the number of vertices, edges and faces in

the sketch graph, thus reducing calculation times for the reconstruction process. The half graph is easy to obtain after pairs of symmetrical vertices have already been found. This idea can be extended to multiple planes of symmetry.

Symmetry in 3-D space can be obtained by composition of transformations of the form:

$$S = \begin{bmatrix} 1 - 2u_1^2 & -2u_1u_2 & -2u_1u_3 & 0 \\ -2u_1u_2 & 1 - 2u_2^2 & -2u_2u_3 & 0 \\ -2u_1u_3 & -2u_2u_3 & 1 - 2u_3^2 & 0 \\ -2Du_1 & -2Du_2 & -2Du_3 & 1 \end{bmatrix} \quad \text{Equation 8}$$

where (u_1, u_2, u_3) is the unit normal vector of the symmetry plane $(Ax+By+Cz+D=0)$.

Having found a plane of symmetry and a half 3-D model, applying S to its vertices allows the complete 3-D model to be obtained. Actually, as the final model retains x and y vertex coordinates from the original drawing, only the computation of z' coordinates for reflected vertices is required. If (x, y, z) are the coordinates of a vertex in the reconstructed half, the z' for the reflected vertex is given by:

$$z' = -2u_1u_3 \cdot x - 2u_2u_3 \cdot y + (1 - 2u_3^2) \cdot z - 2Du_3 \quad \text{Equation 9}$$

Note that information is used from the whole object even when we eliminate half of the figure. The planes of symmetry [8] are found from points situated at the midpoints of edges (and vertices). Thus, 3-D object construction is based on information averaged from both halves.

9 Results

A sketch with a single symmetry plane is used to illustrate the results of the different approaches. Initially, a tidied line drawing and symmetry polygon were automatically obtained (see Figure 8).

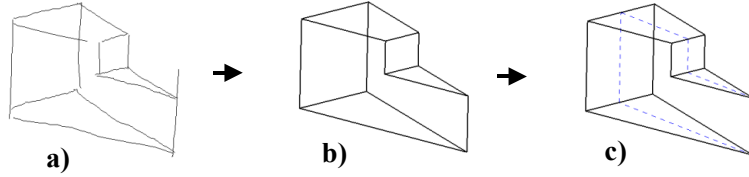


Figure 8. a) Input sketch, b) line drawing, c) symmetry polygon.

If the symmetry regularity is formulated only in terms of the planarity of symmetry plane (see Section 2.1), the result is a model with a nearly planar symmetry plane, but which is not the shape intended (see Figure 9: an axonometric-like image of the model is given, together with front, top and side views). Furthermore, the condition is also satisfied by a model in which all z coordinates are 0. Some other inflation method is needed to escape from this trivial solution. The same happens if orthogonality between the symmetry plane and edges-crossing-symmetry-plane is instead enforced (see Figure 10). When both conditions are enforced, the objective function is ill-conditioned, resulting in a clearly tangled model, corresponding to a local minimum in the objective function: see Figure 11.

In all three cases, a simple hill climbing optimisation process was used, and a random inflation was done in advance to avoid the trivial solution (all z coordinates 0). Random inflation consists of randomly choosing all z_i in the normalized interval $[0, \max(\Delta x, \Delta y)]$. Because of the initial random inflation, the initial figure to be reconstructed can vary, and depending on this, the final results can vary slightly, and are not always exactly the same.

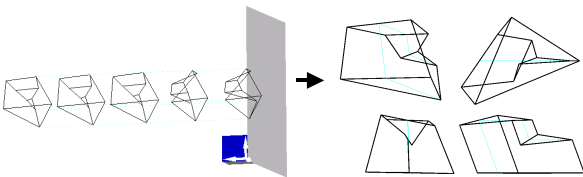


Figure 9. Model obtained using symmetry plane planarity as the only regularity.

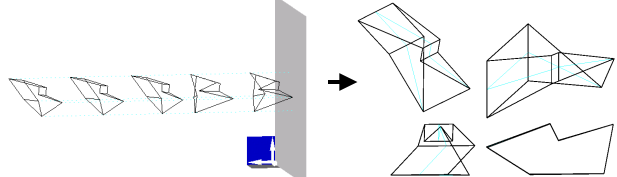


Figure 10. Model obtained using orthogonality among edges-crossing-symmetry-plane and the proper symmetry plane as the only regularity.

Finally, using the bilateral symmetry regularity described in Section 7, a much better model was obtained (see Figure 12). Additionally, no initial inflation method was required, since symmetric-model regularity is not trivially satisfied in the input image. The regularity is formulated as a quadric function (see Equations 7-8), which provides quicker convergence than a linear function.

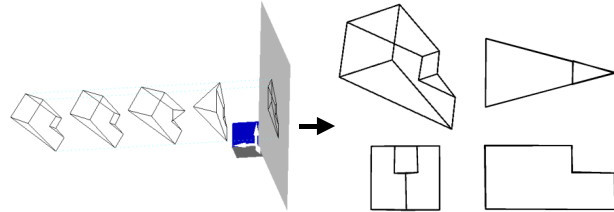
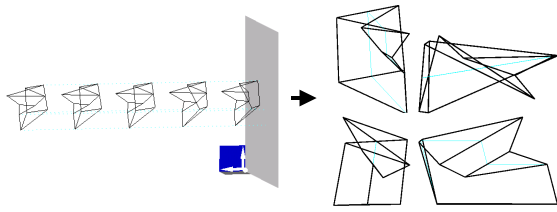


Figure 11. Model obtained using simultaneously symmetry plane planarity and orthogonality among edges-crossing-symmetry-plane and the proper symmetry plane as regularities.

Figure 12. Model obtained using the bilateral symmetry regularity.

Next, a table summarizes some examples that show the performance of our new approach. Four examples are shown as they appeared after the symmetry regularity was applied alone.

Line drawing	3D model	Process	Line drawing	3D model	Process
 19 edges 12 vertices		9 faces 1 plane of symmetry Inflation time: less than 1"	 33 edges 22 vertices		13 faces 1 plane of symmetry Inflation time 1"
 24 edges 16 vertices		10 faces 1 plane of symmetry Inflation time: less than 1"	 46 edges 30 vertices		18 faces 1 plane of symmetry Inflation time 2"

Table 1

This regularity is a *true regularity*. Its use only depends on the detection of characteristic clues in the input drawing. It can be applied alone for converting sketches to 3D models, without any other inflation method, and without the use of other, possibly conflicting, regularities. Thus, this regularity can be considered as a new inflation method by itself. It solves reconstruction for every symmetric sketch for which the planes of symmetry have been detected.

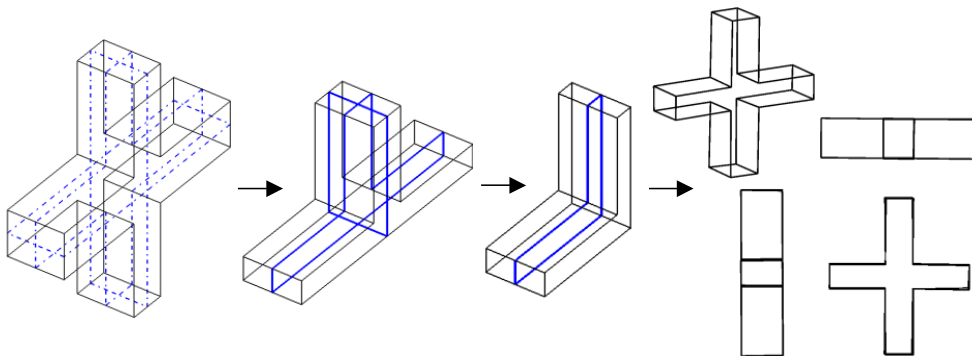


Figure 13. Model with three planes of symmetry

Finally, we give some examples of more complex shapes, to illustrate the benefits of half-part reduction. Figure 13 shows a shape with three planes of symmetry. Two of them are used to reduce the problem size, then the last one is used as a regularity to reconstruct a partial model. Finally, the complete model is restored. In this example, computation times are reduced from 3 seconds for reconstruction of the complete drawing to less than 1 second if the reduced model is used (on a Pentium III 1GHz, with 256 Mb RAM).

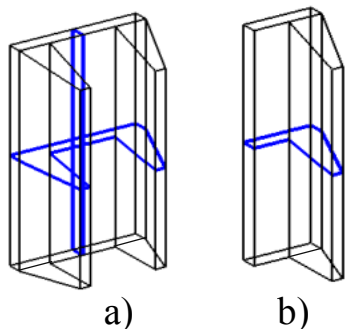


Figure 14. Figure with two planes of symmetry. Reconstruction from a) 3" and from b) less than 1

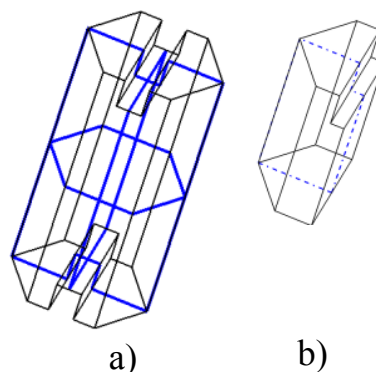


Figure 15. Figure with two planes of symmetry. Reconstruction from a) 6" and from b) less than 1"

Our new approach simplifies the process of reconstruction by diminishing the number of variables used in the main computation, leading to a reduction of time in the whole process. Extruded models are an exception to this observation. They have (at least) one plane of symmetry (orthogonal to the extrusion direction), but the half part formed from this has exactly the same number of vertices, edges and faces as the original figure (see Figure 16). Furthermore, after this "simplification", a new plane of symmetry with the same shape as the first is found. An additional test must be done for extrusions to prevent this false simplification.

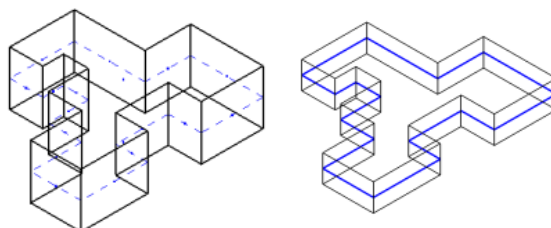


Figure 16. Detection of symmetry planes in extruded shapes.

10 Summary

This paper considers the three-dimensional reconstruction of polyhedral solids from planar sketches. A new approach has been proposed to make use of planes of mirror symmetry detected [8] in sketches. This depends on identification of the two symmetric sides in the 2D representation and the matching of vertices using methods given in this article.

Two novel ideas for reconstruction have been presented. The first is to apply symmetry as a regularity in optimisation-based reconstruction. In fact, symmetry is shown to be adequate by itself, without the need for other inflation techniques or regularities. Good qualitative results and acceptable computing times are provided by this method.

The second idea is to eliminate the duplicated information in symmetrical models before reconstruction. After elimination of half of the drawing, reconstruction of the simplified drawing can be made by any of the methods already known (and indeed with the first idea above). The aim of this technique is to decrease computing time. This objective is achieved in most examples, although not for extrusions.

Overall, use of symmetry is a powerful method, since it allows complex shapes to be inflated without introducing a large set of potentially conflicting heterogeneous regularities (face planarity, orthogonality, and so on), and it does not require initial estimates for inflation. Moreover, it has been successfully employed to recursively reduce the problem size.

11 Acknowledgments

This work was partially supported by *Fundació Caixa Castelló-Bancaixa* under the *Universitat Jaume I* program for Research Promotion (Project P1-1B2002-08, titled “From sketch to model: new user interfaces for CAD systems”), and by facilities provided by *Cardiff University Computer Science Department* during a visit by the first author.

References

- [1] Ullman D.G, Wood S., Craig D. “The Importance of Drawing in the Mechanical Design Process.” *Computers & Graphics*, Vol 14(2): 263-274, 1990.
- [2] Cugini U. “The problem of user interface in geometric modelling.” *Computers in Industry*, 17: 335-339, 1991.
- [3] Dori D. “From engineering drawings to 3D CAD models: are we ready now?” *Computer Aided Design*, vol. 27 (4): 243-254, 1995.
- [4] Lipson H. and Shpitalni M. A. “New Interface of Conceptual Design Based on Object Reconstruction From a Single Freehand Sketch.” *Ann CIRP*. 44 (1): 133-136, 1995.
- [5] Ullman D.G. “Toward the ideal mechanical engineering design support system.” *Research in Engineering Design*. 13: 55-64, 2002.
- [6] Palmer S. E. *Vision Science. Photons to Phenomenology*. The MIT Press. 1999.
- [7] Langbein, F.C.; Mills, B.I.; Marshall, A.D.; Martin, R.R. “Recognizing Geometric Patterns for Beautification of Reconstructed Solid Models” *Proc. of the International Conference on Shape Modelling and Applications*, Genova 2001.
- [8] Piquer, A., Company, P. y Martin, R.R. “Skewed mirror symmetry in the 3D reconstruction of polyhedral models” *Journal of WSCG*. 11 (3): 504-511, 2003.
- [9] Zeleznik, R.C., Herndon, K.P., Hughes, J.F. “SKETCH: An interface for sketching 3D scenes.” *SIGGRAPH’96 Conference Proceedings*, 1: 163-170, 1996.
- [10] Egli, L., Hsu, C.-Y., Bruederlin, B. D., Elber, G. “Inferring 3D Models from Freehand Sketches and Constraints.” *Computer-Aided Design*, 29 (2): 101-112, 1997.
- [11] Bloomenthal K., Zeleznik R.C. et al. “SKETCH-N-MAKE: Automated machining of CAD sketches.” *Proceedings of ASME DETC’98*, 1: 1-11, 1998.
- [12] Qin S.F., Wright D.K., and Jordano I.N. “From On-line Sketching to 2D and 3D Geometry: A System Based on Fuzzy Knowledge.” *Computer-Aided Design*, 32 (14): 851-866, 2000.
- [13] Pereira J., Jorge J., Branco V. y Nunes F. “Towards calligraphic interfaces: sketching 3D scenes with gestures and context icons.” *WSCG’2000. Conference Proceedings*, Prague, 2000.
- [14] Jenkins, D.L. and Martin, R.R. “Applying constraints to enforce users’ intentions in free-hand 2-D sketches.” *Intelligent Systems Engineering*, 1 (1):31-49, 1992.
- [15] Igarashi T., Matsuoka S., Kawachiya S. and Tanaka H. “Interactive Beautification: A Technique for Rapid Geometric Design.” *Proceedings of the ACM Symposium on User Interface Software and Technology UIST’97*, 1: 105-114, 1997.
- [16] Schweikardt E., Gross M.D. “Digital Clay: deriving digital models from freehand sketches.” *ACADIA ‘98*, Quebec City, Canada, 1: 202-211, 1998.
- [17] Oh B.S., Kim C.H. “Progressive 3D reconstruction from a sketch drawing.” *Proceedings of the 9th Pacific Conference on Computer Graphics and Applications*, 1: 108 -117, 2001.
- [18] Naya F., Jorge J., Conesa J., Contero M. and Gomis J.M. “Direct modeling: from sketches to 3D models.” *Proceedings of the 1st Ibero-American Symposium in Computer Graphics SIACG*, Guimarães, Portugal, 1: 109-117, 2002.
- [19] Naya F.; Conesa J.; Contero M.; Company P. and Jorge J. “Smart Sketch System for 3D Reconstruction Based Modeling”. *3rd International Symposium on Smart Graphics 2003*, (not yet published), 2003.
- [20] Marill, T. “Emulating the Human Interpretation of Line-Drawings as Three-Dimensional Objects.” *International Journal of Computer Vision*, 6 (2): 147-161, 1991.
- [21] Baird, L.; Wang, P. “3D object recognition using Gradient descent and the Universal 3-D array grammar.” *Proc. of SPIE. Conference on Intelligent Robots and Computer Vision X: Algorithms and Techniques*, 1607: 711-718, 1991.
- [22] Leclerc, Y. and Fischler M. “An Optimization-Based Approach to the Interpretation of Single Line Drawings as 3D Wire Frames.” *International Journal of Computer Vision*, 9(2): 113-136, 1992.

- [23] Baird, L.; Wang, P. "3D Object Perception Using Gradient Descent." *Int. Journal of Mathematical Imaging and Vision (IJMIV)*, 5: 111-117, 1995.
- [24] Lipson H. and Shpitalni M. "Optimization-Based Reconstruction of a 3D Object from a Single Freehand Line Drawing." *Computer-Aided Design*, 28(8): 651-663, 1996.
- [25] Brown, E.W. and Wang, P.S. "Three-dimensional object recovery from two-dimensional images: A new approach." *Proc. SPIE. Intelligent Robots And Computer Vision XV: Algorithms, Techniques, Active Vision, and Materials Handling*, 2904: 138-147, 1996.
- [26] Company, P., Gomis, J.M. and Contero, M. "Geometrical Reconstruction from Single Line Drawings Using Optimization-Based Approaches." *WSCG'99. Conference proceedings*, 2: 361-368, 1999.
- [27] Lipson, H.; Shpitalni, M. "Correlation-Based Reconstruction of a 3D Object From a Single Freehand Sketch" *AAAI Spring Symposium Series- Sketch Understanding*, 2002.
- [28] Grimstead, I.J. and Martin, R.R. "Creating solid models from single 2D sketches." *Proc 3rd Symposium on Solid Modeling and Applications, ACM Siggraph*, 1: 323-337, 1995.
- [29] Varley, P. A. C. and Martin, R. R. "Constructing Boundary Representation Solid Models from a Two-Dimensional Sketch - Frontal Geometry and Sketch Categorisation." *Proc. First UK-Korea Workshop on Geometric Modeling and Computer Graphics, Kyung Moon Publishers*, 1: 113-128, 2000.
- [30] Tate S.J. and Jared G.E.M. "Recognising symmetry in solid models." *Computer-Aided Design*, 35, pp 673-692, 2003.