Computers & Graphics 59 (2016) 1–12

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Technical Section

Detecting mirror symmetry in single-view wireframe sketches of polyhedral shapes ☆

Raquel Plumed [a,*], Pedro Company [b], Peter A.C. Varley [b]

[a] Departament of Mechanical Engineering and Construction, Universitat Jaume I, Spain
[b] Institute of New Imaging Technology, Universitat Jaume I, Spain

ARTICLE INFO

Article history:
Received 21 January 2016
Received in revised form
4 May 2016
Accepted 7 May 2016
Available online 17 May 2016

Keywords:
Sketch-based modelling
Artificial perception
Bilateral symmetry

ABSTRACT

As part of a strategy for creating 3D models of engineering objects from a sketched input, we attempt to identify mirror symmetry planes early in the process. Our input is a 2D line-drawing derived from a single view sketch of a polyhedral shape, and suitable cues are used to identify mirror symmetry planes algorithmically. Previous attempts to simplify the search space use incomplete criteria. In this paper, we further simplify the search space by scoring valid symmetry planes by their likelihood, i.e., we do not assert "this is a symmetry plane" but "this is more/less likely to be a symmetry plane".

1. Introduction

The computer interpretation of sketched line-drawings is a classic problem in Computer Graphics. It is a complex area that includes a significant number of open problems [1]. One such problem is to provide computational support for sketching activities in design. Since sketches are quick depictions to facilitate visual thinking [2], they are inherently incomplete and imperfect. There is a general agreement that producing 3D models from sketches is not just a geometrical problem, as sketches convey shapes and relationships without necessarily attending to geometrical constraints or projective laws. This is due to the so-called "Semantic Gap" [3], the inability of algorithms to extract all the information which visual data conveys to a human. Therefore, artificial perception is increasingly used to detect cues that can guide depth perception. Cues, also called regularities, refer to the properties of a 2D image that humans perceive as properties of a 3D model [4,5].

Different cues have been used to reconstruct 3D objects from single line-drawings [6–8]. However, most of the strategies to detect cues require further improvements. Symmetry is one of the cues that require improvements.

Two important unaddressed deficiencies remain unanswered. First, existing approaches strongly rely on heuristic assumptions that are incomplete and sometimes incorrect, which results in a detection process with an excessive ratio of both false positives and false negatives. Second, most of the approaches were developed for perfect or nearly perfect line-drawings—they are not sufficiently tolerant of small geometric imperfections, and become ineffective for imperfect sketches. At best, they include thresholds with valid/invalid limits.

In this paper, we describe an approach to determine bilateral symmetry planes of polyhedral shapes sketched as single-view wireframes. Our approach takes advantage of the method to find skewed facial symmetry described by Piquer et al. [9], and improved by Zou and Lee [10]. It builds on two key points: (1) it improves the selection of candidate symmetry planes to further reduce the search space and 2) the symmetry planes are scored by their likelihood, as we do not assert "this is a symmetry plane" but "this is more/less likely to be a symmetry plane."

2. Terminology

The output of our Sketch-Based Modelling (SBM) system are 3D models delimited by *faces*, *edges* (where two faces meet) and *vertices* (where edges meet). A model is polyhedral if all faces are planar. The set of these elements bounds the solid model.

The input to our SBM system are the *strokes* (time-ordered sequences of 2D points) generated by the user as she moves the pointer to create the sketch. The *sketch* is the entire set of strokes.

A vectorization module fits each stroke into a straight line, thus producing a 2D line-drawing. Each stroke may not correspond to a single line, so a segmentation module breaks strokes that depict multiple chained segments into individual straight lines.

☆This article was recommended for publication by Y. Tsui Lee.
* Corresponding author.
E-mail address: plumed@uji.es (R. Plumed).

http://dx.doi.org/10.1016/j.cag.2016.05.013
0097-8493/

Vertex 0: 621.543928, -564.303734
Vertex 1: 874.500000, -145.250000
Vertex 2: 319.500000, -266.750000
Vertex 3: 911.050287, -411.275916
Line 0: V0 to V1
Line 1: V1 to V2
Line 2: V0 to V3
Line 3: V3 to V1
Line 4: V2 to V3
Line 5: V0 to V2
Face 0: 3 L3 L2 L0
Face 1: 3 L4 L5 L2
Face 2: 3 L0 L5 L1
Face 3: 3 L1 L4 L3

Fig. 1. Input 2D line-drawing plus faces of the tetrahedron shown in Fig. 13.

A line-drawing is a 2D pictorial representation of an object. *Lines* in the drawing represent the object's edges. Since lines are 2D views of 3D edges, we call them 2D-lines. 2D-lines intersect at *junctions* (or 2D vertices), which represent the model vertices. When a set of edges delimit the boundary of a face of the model, their corresponding 2D-lines define *face-loops* that subdivide the drawing into regions.

A natural line-drawing is a line-drawing where only the object's visible edges and parts of edges are shown. Alternatively, a wireframe shows all the edges of the object. The distinctions between junctions and vertices, 2D-lines and edges, and face-loops and faces are important as they clearly separate 2D from 3D worlds. However, the distinctions are sometimes omitted for brevity when the context provides the required disambiguation. For example, in Fig. 1 the input file (which implies a 2D line-drawing) of the tetrahedron from Fig. 13, where a mixed naming is used, is shown.

3. Related work

There is ample evidence that humans can quickly discriminate between symmetrical and asymmetrical shapes [11]. Detecting symmetry in 3D models is a classic problem in automatic feature recognition, where authors Tate and Jared gave an early insight observing that human perception of symmetry is considerably more robust than any computational method in existence and therefore the cognitive processes involved in human symmetry detection should be considered [12]. Jiang et al. [13] made a recent contribution to detecting global symmetry, while Li et al. [14] addressed local symmetry. In this paper, we focus on the *early* detection of bilateral symmetry, when the only available information is a flat sketch.

In addition to mirror or bilateral symmetry, other types of symmetries are common and useful in engineering design. Wolter et al. [15] compiled optimal approaches for symmetry detection, and showed that the problem of finding approximate symmetries in point sets was out of the scope of those approaches. More recently, Zou and Lee [16] studied the particular case of detecting rotational or radial symmetry in 2D line-drawings. In our paper, only bilateral symmetry is considered.

Techniques for the detection of symmetries usually distinguish between polyhedral and free-form shapes, as their unique inputs require different approaches (mainly because of the skew ambiguous problem derived from the existence of multiple symmetry axes). In this paper, only symmetries in polyhedral shapes are studied. Recent advances with free-form mirror-symmetric shapes can be found in the work by Cordier et al. [17,18], where curve-cusps were detected and paired with generator lines, thus adapting the skew symmetry detection approach defined by Kanade [19], described by Friedberg [20], and developed by Posch [21].

The problem of finding bilateral symmetry in 2D line-drawings of polyhedral shapes was already considered by Piquer et al. [9], who also gave an approach to use symmetry planes for producing 3D models from 2D line-drawings [22], a strategy that was improved by Zou and Lee [10]. Varley et al. [23] used a similar approach to detect bilateral symmetry planes and create 3D models from sketches. Varley et al. assumed that the axis-aligned planes where a vertex lies can be deduced. Then, once the planes are ordered, the symmetry plane must lie in the middle. Varley [24] introduced another approach to finding faces in natural line-drawings, which begins by looking for possible symmetry planes through single faces, and then propagating them from one face to the next. The strategy was not effective with partial faces, where only part of a face is visible in the drawing. However, this is not the case with wireframe drawings.

4. Finding symmetry planes

Our input for finding bilateral symmetry planes requires 2D line-drawings. Although further improvements are required, conversion from wireframe sketches into 2D line-drawings is a well-known stage in Sketch-Based Modelling (see “recognition” in Johnson et al. [2]).

Faces must be detected in advance. Symmetry detection before finding faces is unnecessary, as there are algorithms where symmetry is not used. The approach we use for finding faces in 2D wireframe line-drawings of polyhedral objects has been described in [25]. Other approaches can be found in the literature [26,27].

In this paper, we assume a graph-like wireframe input containing a list of junctions, a list of 2D-lines, and a list of face-loops. Junctions are defined by their x and y coordinates. 2D-lines are defined by the two junctions they connect. Face-loops are defined as closed loops of 2D-lines.

The proposed procedure for detection of bilateral symmetry planes can be summarized as follows:

1. The axes of skewed symmetry for each face-loop are detected.
2. To reduce the search space, a set of rules are applied to remove the (sides)—symmetry axes plus 2D-lines—that cannot belong to any true global symmetry plane.
3. We search for circuits of sides lying in a single plane of symmetry.
4. Finally, merits are assigned depending on whether the candidate symmetry circuits accomplish those properties of symmetry planes that can be indirectly measured in a flat-drawing.

4.1. Symmetry circuits

Symmetry planes cannot be found directly in 2D drawings. However, they can be determined by one indirect cue that we call *circuit of symmetry*.

Circuits of symmetry exist because intersections between planes and polyhedrons always produce one or more polygons whose *sides* are the segments that result from intersecting the plane with each face, and whose *corners*—we use this term to prevent confusion with polyhedron vertices—are the points that result from intersecting the plane with each edge. Two particular cases may happen: the intersection plane may contain edges (where it simultaneously intersects both the faces that share the