www.cadanda.com Computer-Aided Design AND APPLICATIONS

Geometric Definition of the Hidden Part of a Line Drawing in a Sketch-to-Solid Methodology

Sofia Kyratzi[1] and Philip Azariadis[2]

[1]University of the Aegean, skiratzi@aegean.gr
[2]University of the Aegean, azar@aegean.gr

ABSTRACT

The method presented in this paper is part of a 3-step reconstruction methodology for the extraction of a polyhedron from a single view natural sketch. In particular, the current paper focuses on the 2D geometric definition of the hidden part in a topologically reconstructed line drawing (intermediate wireframe sketch) as this is generated from the given natural sketch. The proposed method is based on the different topologic relations that exist in the hidden part of the intermediate sketch. Contrarily to other approaches it employees the cross-section criterion to ensure that the final fully-determined wireframe sketch is realizable. The proposed method is successfully applied to indicative line drawings with various combinations of visible and hidden elements.

Keywords: computer-aided sketching, sketch-to-solid, wireframe sketch.

1. INTRODUCTION

A sketch is considered as a primary tool for the communication of ideas especially in the conceptual phase of design. The growing evolution and the consistent efforts to properly found the new arising Computer Aided Sketching (CAS) systems [5] establish the importance of 2D sketches in design oriented disciplines. The ability of humans to realize the 3D shape of a sketched object, even if the latter appears ambiguous or contains geometric errors motivate the researchers of multiple fields, such as Computer Aided Design and Sketching, Geometric Modeling, and Artificial Intelligence, to search for efficient methods for the reconstruction of a 3D model from a given single sketch.

The subject of this paper is related to the automatic construction of a "Polyhedron from a Single Natural Sketch". In natural sketches, the topologic and geometric determination of their hidden part underline the main focus of the methods proposed in [2], [4], [6], [8–10], [13]. In most works, a two-step reconstruction methodology is followed, where, firstly, the topologic reconstruction of a wireframe sketch is achieved on the basis of the 2D sketch information, and, secondly, a 3D polyhedron is determined from this topologically reconstructed sketch. In particular, the 3D geometry of the polyhedron is derived by exploiting the sketch topology according to an optimization procedure with soft constrains [2], or through the generation of a partial 3D object [6], [13]. In most studies, the final 3D model is the result of an optimization–based procedure that weights the plausibility of each image regularity between the sketch and the inferred 3D model [3], [11]. The 2D geometry of the hidden part of a wireframe sketch is not considered as a distinct problem and is only indirectly specified through the reconstruction of the 3D model.

On the contrary, our "Sketch-to-Polyhedron" methodology for the reconstruction of a 3D model from a natural sketch proceeds in three distinct steps [9]. In the first step, a topological valid *intermediate wireframe sketch* is produced from the given natural sketch [8]. In the intermediate wireframe sketch the geometry and topology of its visible junctions, lines, and regions are precisely defined, while for its hidden elements only the topological relations are known. In the second step, the 2D geometry of the hidden elements in the intermediate wireframe sketch is specified. The second step results in a topologically and geometrically defined wireframe sketch, whose visible part is identical to the input natural sketch. In the third step, a polyhedron is created on the basis of the output wireframe sketch of the second step.

This paper focuses on the second step of the aforementioned 3D reconstruction methodology and presents a method for the definition of the 2D

Taylor & Francis
Taylor & Francis Group

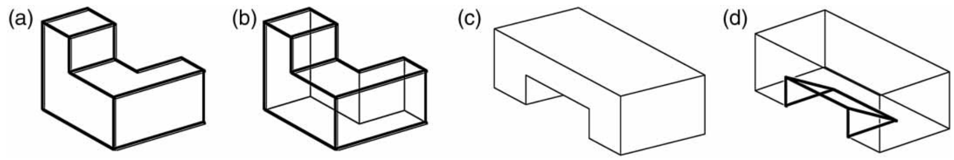

Fig. 1: (a) A natural sketch, and (b) its corresponding wireframe sketch, (c-d) the minimal completion strategy results to the sketch in (d).

geometry of the hidden junctions and lines in an intermediate wireframe sketch. The proposed method is semi-automatic and is based on the different topologic relations that exist in the hidden part of a sketch. Contrarily to existing methods, by utilizing the cross-section criterion [13] we are able to assert the realizability of the produced wireframe sketch (i.e., whether a sketch identifies with the projection of a valid polyhedron). When multiple geometric solutions exist for the hidden part, as a result of strict ambiguities in the visible part, a constrained interactive approach is employed in order to provide a unique realizable solution.

The proposed method is based on the algebraic representation of the cross-section criterion that is thoroughly described in [1] and [7]. In these works a cross-section algebraic method is developed for testing the realizability of a given wireframe sketch. In this paper, the algebraic model is extended in order to encompass wireframe sketches with incomplete geometry (i.e., intermediate wireframe sketches). Following this new algebraic representation, the geometry of the hidden part is defined by verifying that the produced wireframe sketch identifies with the projection of a valid polyhedron. This approach allows for the development of more robust 3D reconstruction algorithms for the third part of the aforementioned methodology.

The structure of the paper is as follows: Section 2 briefly discusses the topologic relations of a sketch and Section 3 describes the cross-section criterion and its corresponding algebraic model. Section 4 introduces the geometric and algebraic framework of the hidden-geometry problem and presents an algorithmic implementation for its solution. The proposed algorithm is tested with sketches with various number of visible and hidden elements and its performance is studied for sketches with perfect and imperfect geometry (i.e., sketches with small errors in the visible junction coordinates).

2. GEOMETRIC AND TOPOLOGICAL DESCRIPTION OF SKETCHES

A sketch is a set of straight lines on a plane that intersect at junctions. Non self-intersecting loops of lines and junctions form the regions of a sketch. Lines, junctions and regions are called "*elements*" of a sketch. The sketch is considered to be the orthographic projection, on plane $\Phi : Z = 0$, of a trihedral solid (i.e., each vertex of it belongs to exactly three faces) with planar faces. The solid is considered to be in "general position" with respect to the projection plane, i.e., no face or edge of the solid is perpendicular/parallel to Φ and the adjacent faces (edges) lie on distinct planes. A "one-to-one correspondence" exists between the lines (L), junctions (J) and regions (R) of a sketch and the edges (E), vertices (V) and faces (F) of a polyhedron.

A *natural sketch* (1(a)) is a sketch without hidden elements and is considered to be drawn in the *most informative view*, i.e., there is nothing at the "back of the sketch" that cannot be directly inferred from its visible part [13], [8]. A *wireframe sketch* (Fig. 1(b)) includes both visible and hidden elements. On the basis of trihedral polyhedral properties, a wireframe sketch is defined as follows [7]:

Definition 1. A *wireframe sketch* is a connected graph that:

1. Every junction is adjacent to three lines (i.e., the degree of each junction j is $d(j) = 3$).
2. Every line is adjacent to two regions.
3. Two adjacent regions of the sketch share exactly one line or two-or-more collinear lines. □

In a natural sketch, the visible lines and junctions are categorized as *internal* or *boundary*. Internal lines (ℓ_{int}) and junctions (j_{int}) follow the properties of Definition 1 (Fig. 2(a)). Boundary lines (ℓ_b) and junctions are visible elements that are directly associated with the hidden part of a wireframe sketch (Fig. 2(a)). Boundary junctions are classified according to the number of their adjacent visible lines, in complete junctions j_{bc} with degree 3, in L-junctions j_L with degree 2, and in T-junctions j_T with degree 1. In particular, T-junctions are visible junctions whose actual position lies on the hidden part of the sketch (Fig. 2(b)) and thus, they are considered to be adjacent to only one visible line. A boundary line that is adjacent to two L-junctions is called an "L-chain" line ℓ_L (Fig. 2(c)).

A wireframe sketch is produced from a given natural sketch, when to the latter's boundary elements appropriate hidden lines (ℓ_h) and junctions (j_h) are added (Fig. 2(a - b)). Following a "*minimal completion*" strategy [8], [9] (1(c - d)) the topologic description for the hidden part results in a wireframe sketch, with all

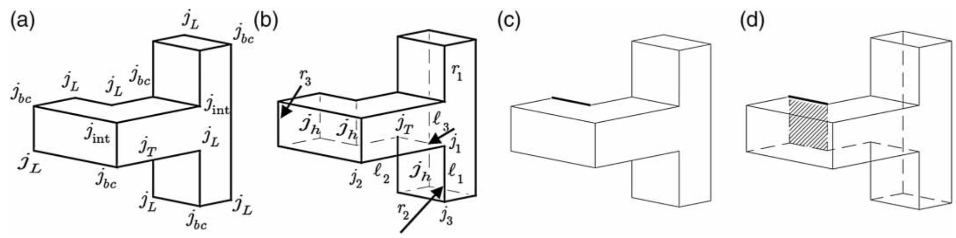

Fig. 2: (a) The four categories of visible junctions in a natural sketch, and (b) the associated hidden junctions in the corresponding wireframe sketch, (c) an "L-chain" boundary line and (d) the adjacent "L-chain" hidden region.

Wireframe Sketch	ℓ_{int}:○○	ℓ_b:●○	ℓ_{ph}:●○	ℓ_h:●●	Symbols
j_{int}	○○ ○				○: visible regions
j_L	○	○	●		●: hidden regions
		○●		●	
			●		
j_{bc}	○	●○			For the line columns ( ): junctions can be joined together to form a line
j_T		●	○	(●)	
	○		○●		
j_h				●●●	
			● ○	(●)	

Tab. 1: For each junction type (j), the number of dots in the corresponding line column (ℓ) indicate the number of the adjacent line types. The number of black/white dots indicates the number of hidden/visible regions associated with the corresponding junction. The line column (ℓ) headers indicate the dot colors that describe each line type. For each column all the possible combinations of dots (in pairs) form a line (with the corresponding line type). A pair of dots in a cell indicates that two junctions of this type can produce a line of the corresponding type.

hidden regions (r_h) adjacent to at least one boundary line and thus to one visible region (r_v) (Fig. 2(b)). A hidden region that is adjacent to an L-chain boundary line ℓ_L is called an L-chain hidden region r_L (Fig. 2(d)) and is the only hidden region r_L that is adjacent to exactly one visible line. T-junctions of a natural sketch are considered as hidden junctions in the wireframe sketch (Fig. 2(b)). Consequently, the hidden lines that are adjacent to a T-junction and belong in one visible and one hidden region are called as *partially hidden lines* (ℓ_{ph}). An "*intermediate wireframe sketch*" is a minimal wireframe sketch, where (a) its visible elements are topologically and geometrically defined, and (b) its hidden part is only topologically determined.

The topologic relations in a wireframe sketch produced from a natural sketch are presented in Tab. 1. For example, the j_1 junction in (Fig. 2(b) is an L-junction in the corresponding natural sketch (Fig. 2(a)). Junction j_1 is adjacent to the internal line ℓ_1, to the boundary line ℓ_2 and to the partially hidden line ℓ_3. Moreover it is adjacent to two visible regions r_1 and r_2, and to one hidden region r_3. With respect to Tab. 1, j_1 corresponds to the first case of the j_L row, where an internal line (ℓ_{int} column) can be formed by a combination of j_{int}-j_{int}, j_{int}-j_L, j_L-j_{bc} or j_{int}-j_T junctions. Indeed, line ℓ_1 is an internal line that belongs to the case of j_{int}-j_L (resp. j_3-j_1). On the same manner, the boundary line ℓ_2 belongs to the case j_L-j_{bc} (resp. j_1-j_2) of the ℓ_b column and the partially hidden line ℓ_3 in the case j_L-j_T (resp. j_1-j_T) of the ℓ_{ph} column.

2.1. Notations

A line between two adjacent regions r_i and r_j is denoted as ℓ_{ij}. A line ℓ_{ij} with terminal junctions $v_p(x_p, y_p)$ and $v_q(x_q, y_q)$ is written as $\ell_{ij} : k_{ij}x + m_{ij}y + n_{ij}$, with $k_{ij} = y_p - y_q$, $m_{ij} = x_q - x_p$, and $n_{ij} = x_p y_q - x_q y_p$. The number of visible/hidden lines, regions, and junctions in the sketch is respectively denoted as L_v/L_h, R_v/R_h, and J_v/J_h.

3. THE CROSS-SECTION CRITERION

The cross-section criterion is a sketch realizability criterion, i.e., it asserts whether a sketch identifies with the projection of a valid polyhedron. Given a wireframe sketch S (Fig. 3(a)) with L lines, J junctions and R regions, a *cross-section* of S is an arrangement of lines $\{L_{f_k} : k = 1, ..., R\}$ that represents the regions $\{R_k : k = 1, ..., R\}$ of S. If every pair of *cross-section* lines

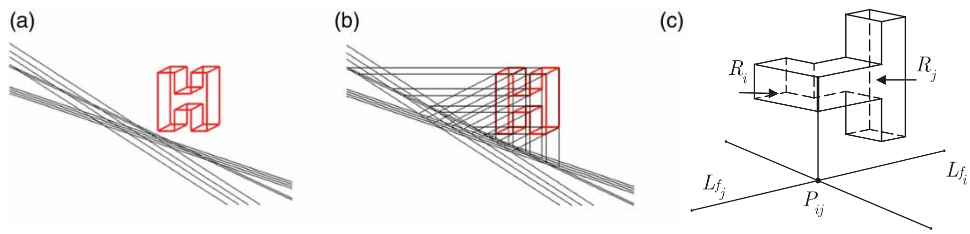

Fig. 3: (a) A cross-section generated from a sketch. (b) A compatible with sketch cross-section. (c) Cross-section compatibility for two regions R_i and R_j and the corresponding cross-section lines.

L_{f_i} and L_{f_j} (Fig. 3(c)), that correspond to adjacent regions R_i and R_j, intersect at a point P_{ij} on (the extension of) the common line ℓ_{ij} of R_i and R_j, the cross-section is called *compatible with S* (Fig. 3(b-c)). A detailed description of the Cross-section criterion can be found in [1], [7], [12].

Whiteley [14] was the first to establish the realizability of a wireframe sketch. The authors of [12] rewrote Whiteley's theorem as follows:

Theorem 1. *(Cross-Section Criterion)* A wireframe sketch is *realizable* if and only if it has a *compatible cross-section*, where the cross-section lines L_{f_i} and L_{f_j} of the adjacent regions R_i and $R_j$are not identical. □

On the basis of the cross section Theorem 1 the following two Corollaries can be derived.

Corollary 1. Given a polyhedron, two non-adjacent parallel faces F_i and F_j of it correspond to two parallel cross-section lines L_{f_i} and L_{f_j}. □

Corollary 1 is a necessary condition for the parallelism of two faces, but it can also robustly indicate which faces intersect.

Corollary 2. [9]: Let S be a wireframe sketch and CS/CS' two compatible cross-sections of it. The intersection point P_{ij}, with $i, j \in \{1, ..., R\}$ of two cross-section lines L_{f_i} and L_{f_j} in CS, and the intersection point $P_{ij}{}'$, of the corresponding cross-section lines L'_{f_i} and L'_{f_j} in CS', are both on (the extension of) the same line ℓ_{ij}. □

3.1. Algebraic Model of the Cross-Section Criterion

This section describes an algebraic model of the geometric cross-section criterion, combined with an algebraic representation of all the constraints that assert sketch's realizability (i.e., cross-section compatibility with a sketch) [1]. This algebraic representation serves as a tool for an efficient computational implementation of the cross-section criterion, and it forms the basis upon which an algorithmic method is developed for the generation of a cross-section from a given sketch.

Definition 2. *(Cross-Section Compatibility)* Let S be a wireframe sketch with R regions, L lines and J junctions (Figure 4(b)). A *compatible with S cross-section* is a set of lines $\{L_{f_i}\}$ such that:

(A) Each cross-section line L_{f_i} that corresponds to region R_i, is written in the form: $b_i x + a_i y + c_i = 0$, with $i = 0, ..., R-1$.
(B) The cross-section lines L_{f_i} and L_{f_j} of two adjacent regions R_i and R_j are not identical.
(C) For each region R_i, its adjacent sketch lines ℓ_{ij} intersect the cross-section line L_{f_i}.
(D) Each line ℓ_{ij} of S that is adjacent to regions R_i and R_j, and the corresponding to these regions cross-section lines L_{f_i} and L_{f_j} intersect at a point P_{ij}. □

Properties (B) and (C) are introduced, with the following Theorem 2, as constraints for the generation of a cross-section from a sketch [1].

Theorem 2. A set of lines$\{L_{f_i}\}$ satisfies the properties (B), (C) of Definition 2 *if and only if* the following hold true:

(C1) $a_i b_j \neq a_j b_i$,
(C2) $k_{ij} a_i - m_{ij} b_i \neq 0$,

where $i, j \in [0, ..., R-1]$ correspond to the two regions R_i and R_j that are adjacent to the sketch line $\ell_{ij} : k_{ij} x + m_{ij} y + n_{ij}$. □

Property (D) is related to the compatibility of the cross-section with a sketch and is employed for the construction of the following cross-section system.

$$\begin{aligned}
\ell_{st} \leftrightarrow eq_{st} &: (c_t a_s - c_s a_t) k_{st} \\
&\quad + (c_s b_t - c_t b_s) m_{st} + (a_t b_s - a_s b_t) n_{st} = 0 \\
\ell_{ps} \leftrightarrow eq_{ps} &: (c_p a_s - c_s a_p) k_{ps} \\
&\quad + (c_s b_p - c_p b_s) m_{ps} + (a_p b_s - a_s b_p) n_{ps} = 0 \\
\ell_{pt} \leftrightarrow eq_{pt} &: (c_p a_t - c_t a_p) k_{pt} \\
&\quad + (c_t b_p - c_p b_t) m_{pt} + (a_p b_t - a_t b_p) n_{pt} = 0 \\
&\cdots\cdots\cdots
\end{aligned}$$

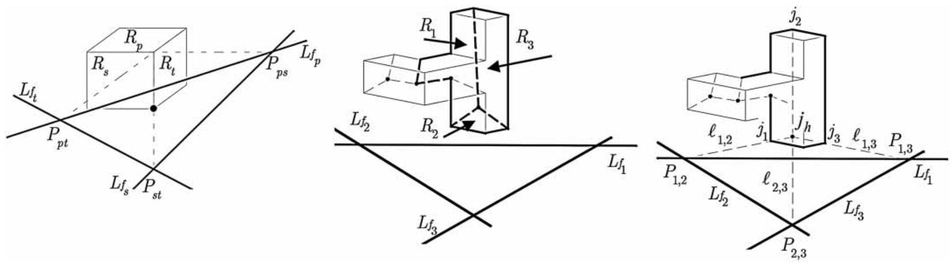

Fig. 4: (a) The generation of cross-section line L_{f_p} on the basis of lines L_{f_s} and L_{f_t}, (b-c) the cross-section lines of the bold marked hidden regions R_1, R_2 and R_3 can be used for the determination of regions' common hidden junction.

$$\begin{aligned}
\ell_{pq} \leftrightarrow eq_{pq} &: (c_p a_q - c_q a_p) k_{pq} \\
&+ (c_q b_p - c_p b_q) m_{pq} + (a_p b_q - a_q b_p) n_{pq} = 0 \\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
\ell_{km} \leftrightarrow eq_{km} &: (c_k a_m - c_m a_k) k_{km} \\
&+ (c_m b_k - c_k b_m) m_{km} + (a_k b_m - a_m b_k) n_{km} = 0 \\
\ell_{dm} \leftrightarrow eq_{dm} &: (c_m a_d - c_d a_m) k_{dm} \\
&+ (c_d b_m - c_m b_d) m_{dm} + (a_m b_d - a_d b_m) n_{dm} = 0
\end{aligned} \tag{3.1}$$

The unknowns of system (3.1) are the coefficients (b_i, a_i, c_i) of the cross-section lines L_{f_i}, $i = 0, ..., R-1$. Each equation eq_{ij} corresponds to a line ℓ_{ij} of the sketch, where $i, j \in [0, ..., R-1]$. Thus, system (3.1) includes $3 * R$ unknowns and L equations. Combined with the constraints (C1) and (C2) of Theorem 2 the Cross-Section Problem (CSP) is defined. In [6] a Cross-Section Calculation Algorithm (CSCA) has been presented to obtain the unknown coefficients $(b_i, a_i, c_i = 1)$. In this paper, CSCA is expanded to solve CSP in terms of the unknown triplet (b_i, a_i, c_i). An overview of the CSCA is presented below.

Let R_s, R_t and R_p be three adjacent regions of a sketch S (Fig. 4(a)). The coefficients of the corresponding cross-section lines L_{f_s}, L_{f_t} and L_{f_p} are included in the equations of system (3.1). The calculation of two unknown triplets (b_s, a_s, c_s) and (b_t, a_t, c_t), fixes the position of the cross-section lines L_{f_s} and L_{f_t} and enables the calculation of (b_p, a_p, c_p) as follows. According to Definition 2, the cross-section line L_{f_t}: $b_t x + a_t y + c_t = 0$ intersects with sketch line ℓ_{pt} : $k_{pt} x + m_{pt} y + n_{pt}$ at point

$$P_{pt} = \left(\frac{a_t n_{pt} - c_t m_{pt}}{b_t m_{pt} - k_{pt} a_t}, \frac{c_t k_{pt} - b_t n_{pt}}{b_t m_{pt} - k_{pt} a_t} \right) \tag{3.2}$$

Similarly, lines L_{f_s}:$b_s x + a_s y + c_s = 0$ and ℓ_{ps} : $k_{ps} x + m_{ps} y + n_{ps}$intersect at point

$$P_{ps} = \left(\frac{a_s n_{ps} - c_s m_{ps}}{b_s m_{ps} - k_{ps} a_s}, \frac{c_s k_{ps} - b_s n_{ps}}{b_s m_{ps} - k_{ps} a_s} \right) \tag{3.3}$$

Points P_{pt} and P_{ps} define the cross-section line L_{f_p} of region R_p, and the coefficients (b_p, a_p, c_p) of L_{f_p} can be calculated in terms of point coordinates

$$\begin{cases} b_p = y_{pt} - y_{ps} \\ a_p = x_{ps} - x_{pt} \\ c_p = x_{pt} y_{ps} - x_{ps} y_{pt} \end{cases}$$

Thus, for the calculation of cross-section line L_{f_p}, on the basis of Eqns. (3.2)-(3.4), only two of the sketch lines that are adjacent to region R_p are used, i.e., lines ℓ_{pt} and ℓ_{ps}. The connected graph of a wireframe sketch allows for the development of an incremental procedure that calculates all unknowns of system (3.1) with respect to initial values provided for two unknown triplets as a starting point [7] (i.e., initialized triplets are chosen to correspond to adjacent regions and thus are associated with a single equation of (3.1)). For the determination of each cross-section line L_{f_i}, CSCA employs only two of the equations that include triplet (b_i, a_i, c_i), leaving in total $U_{eq} = L/3 - 1$ "unused" equations of system (3.1). Indeed, the six initially determined values reduce the number of equations (to be used) to $L-1$ and the number of unknowns to $3R-6$. For the latter to be calculated, only $2R-4$ equations are employed. Combining Euler Formula $(R - L + J = 2)$ with relation $L = 3J/2$ that associates junctions and lines in a graph of degree three, one can easily prove that the number of unused equations equals to $U_{eq} = L - 1 - (2R - 4) = L/3 - 1$.

4. GEOMETRIC DEFINITION OF THE HIDDEN JUNCTIONS

This section focuses on the calculation of the coordinates of the hidden junctions j_h and T-junctions j_T located in the hidden part of the intermediate wireframe sketch. The number of hidden junctions J_h equals to the number of j_h and j_T in a sketch. The *Hidden Junctions Geometry (HJG)* method that is presented here is based on the cross-section criterion (Definition 2), Theorem 1 and on Corollaries 1&2. The input of the method is the intermediate wireframe sketch, i.e., the geometry and the topology of the visible part of the sketch and the topologic relations of its hidden part. The output of the method is a geometrically defined position for each hidden junction together with a cross-section that verifies the realizability of the produced wireframe sketch.

The HJG method proceeds in two steps. The first step focuses on the construction of a cross-section from the input intermediate wireframe sketch (see Section 4.1). In particular, a cross-section line is generated for each visible and hidden region of the sketch except for the L-chain hidden regions r_L. For a hidden region r_h that is adjacent to at least two visible lines, the cross-section line L_{f_h} can be defined on the basis of sketch's visible geometry. On the contrary, for a hidden region r_L, its corresponding cross-section line L_{f_L} cannot be directly determined, since r_L is adjacent to only one boundary line ℓ_L. Thus, the cross-section line of each r_L is determined at the second step along with the geometric definition of each hidden junction (see Section 4.2). The input of the second step is the intermediate wireframe sketch and a compatible cross-section associated with it.

4.1. Hidden Junctions Geometry: Cross-Section Generation

The cross-section criterion is applied to the intermediate wireframe sketch. In complete analogy to system (3.1), a cross-section system (4.1) is generated from the sketch on the basis of the latter's visible lines (Tab. 1). The boundary lines ℓ_L are excluded from system (4.1) equations. The visible lines that are adjacent to a T-junction are included in (4.1), where the coordinates of the visible T-junction are used in the corresponding equations.

$$
\begin{aligned}
\ell_{st} \leftrightarrow eq_{st} &: (c_t a_s - c_s a_t)k_{st} \\
&\quad + (c_s b_t - c_t b_s)m_{st} + (a_t b_s - a_s b_t)n_{st} = 0 \\
\ell_{ps} \leftrightarrow eq_{ps} &: (c_p a_s - c_s a_p)k_{ps} \\
&\quad + (c_s b_p - c_p b_s)m_{ps} + (a_p b_s - a_s b_p)n_{ps} = 0 \\
\ell_{pt} \leftrightarrow eq_{pt} &: (c_p a_t - c_t a_p)k_{pt} \\
&\quad + (c_t b_p - c_p b_t)m_{pt} + (a_p b_t - a_t b_p)n_{pt} = 0 \\
&\cdots\cdots\cdots\cdots\cdots\cdots \\
\ell_{pq} \leftrightarrow eq_{pq} &: (c_p a_q - c_q a_p)k_{pq} \\
&\quad + (c_q b_p - c_p b_q)m_{pq} + (a_p b_q - a_q b_p)n_{pq} = 0 \\
\ell_{dm} \leftrightarrow eq_{dm} &: (c_m a_d - c_d a_m)k_{dm} \\
&\quad + (c_d b_m - c_m b_d)m_{dm} + (a_m b_d - a_d b_m)n_{dm} = 0
\end{aligned}
\tag{4.1}
$$

System (4.1) includes $L_v - L_L$ equations, where L_L is the number of "L-chain" lines. The equations of system (4.1) are a subset of the L equations that appear in system (3.1), since system (4.1) refers to the visible part of the wireframe sketch that is represented by (3.1). The unknowns of system (4.1) are the triplets (b_i, a_i, c_i), with $i \in [0, ..., R-1] - \{r_L\}$. The number of the unknowns is $3(R - R_L)$, where R_L is the number of r_L regions. In particular, excluding lines ℓ_L from system (4.1), the triplets (b_L, a_L, c_L) that correspond to hidden regions r_L are also excluded from the set of unknowns. Thus, system (4.1) includes only unknown triplets that appear to at least two equations, establishing the existence of a solution on the basis of CSCA. A cross-section is generated from the intermediate wireframe sketch and its compatibility with the sketch is evaluated with respect to a predefined accuracy level.

According to property (D) of Definition 2 the intersection point P_{ij} of cross-section lines L_{f_i}: $b_i x + a_i y + c_i = 0$ and L_{f_j}: $b_j x + a_j y + c_j = 0$, should lie on line $\ell_{ij} : k_{ij}x + m_{ij}y + n_{ij}$ and satisfy:

$$
\left| \left(\frac{c_i a_j - c_j a_i}{a_i b_j - a_j b_i} \right) k_{ij} + \left(\frac{c_j b_i - c_i b_j}{a_i b_j - a_j b_i} \right) m_{ij} + n_{ij} \right| \le \mu \tag{4.2}
$$

For perfect sketches that contain no error in their junctions positions it holds $\mu = 0$. However, small round-off errors can appear in junctions coordinates $(x + E, y + E)$ during computer-based sketching under finite precision or due to numerical processes that apply at a sketch's digitization stage. In that case, an acceptable level of accuracy must be set for testing the cross-section compatibility with the sketch. For numerical precision errors $E \le 5 * 10^{-6}$, we set an upper limit for μ on the order of $O(10^{-4})$ (see related discussion in [6]). If the cross-section is found compatible, the geometric definition of the hidden part proceeds with the second step of the method, otherwise the sketch is considered as non-realizable and the process terminates.

4.2. Hidden Junctions Geometry: Hidden Junctions Position Calculation

The second step of the proposed methodology commences with the initial intermediate wireframe sketch and the cross-section produced during the first step. The output of this step consists of the geometric definition of the sketch hidden part together with a cross-section that verifies its realizability. This step is presented in three phases: Firstly, the geometric framework of the proposed method is detailed, followed by an algebraic representation of the geometric concept in the second phase. The presentation of the second step concludes with the algorithmic implementation of the theoretical scheme (HJCalc Algorithm) and its application to different test cases.

4.2.1. Geometric framework

For the definition of the position of a hidden junction j_h (or j_T) the following procedure is applied. Let R_1, R_2, and R_3 be three adjacent hidden regions (Fig. 4(b))

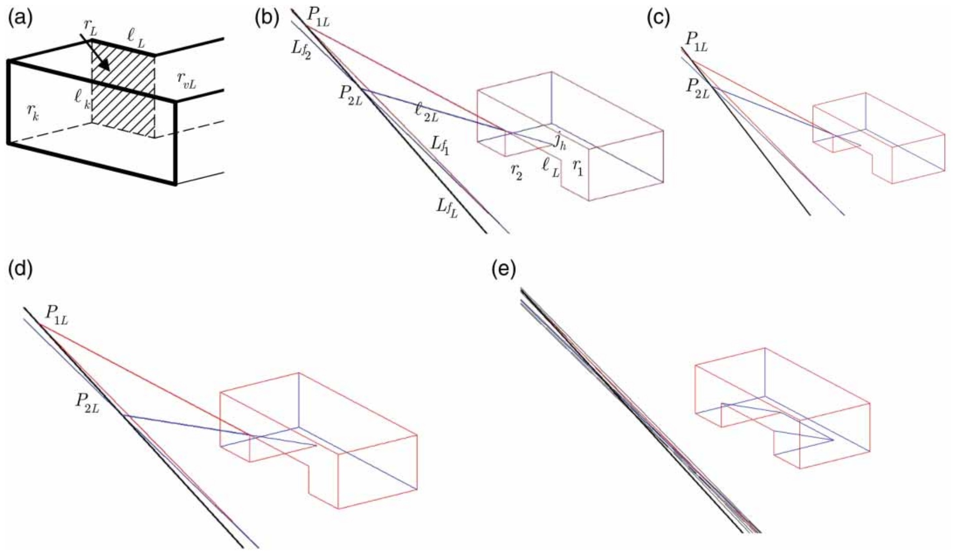

Fig. 5: (a) Make Parallel Procedure: regions r_k and r_L correspond to parallel faces F_k and F_L, (b) UDPosition Procedure: the cross-section line $L_{f_L}$is determined with respect to fixed point $P_{1,L}$ and the multiple constraint positions of $P_{2,L}$, (c - e) different solutions for the hidden part as a result of the L_{f_L}'s different slopes.

that include a junction j_h and let $\ell_{1,2}$, $\ell_{1,3}$, and $\ell_{2,3}$ be the three adjacent to j_h hidden lines (Fig. 4(b-c)). For the sake of clarity, let lines $\ell_{1,2}$,$\ell_{2,3}$, and $\ell_{1,3}$ be adjacent respectively to an already known junction j_1, j_2, and j_3 (note: more complicated arrangements of lines are considered below). Given the cross-section lines L_{f_1}, L_{f_2}, and L_{f_3} of the above regions, Corollary 1 establishes that each pair of these lines will intersect, respectively at points $P_{1,2}$, $P_{2,3}$, and $P_{1,3}$ (Fig. 4(c)). Properties (C) and (D) of Definition 2 indicate that these points must be on the extension of the corresponding sketch lines $\ell_{1,2}$, $\ell_{2,3}$, and $\ell_{1,3}$. Thus, line $\ell_{1,2}$(resp. $\ell_{2,3}/\ell_{1,3}$) is defined by the point $P_{1,2}$ (resp. $P_{2,3}/P_{1,3}$) and the junction j_1 (j_2/j_3) (Fig. 4 (b)). Hidden junction j_h is the intersection point of $\ell_{1,2}$, $\ell_{2,3}$, and $\ell_{1,3}$. Corollary 2 ensures that the position of a hidden junction is the same for any cross-section generated by the sketch.

For the generation of a cross-section line L_{f_L} of an L-chain hidden region r_L, we study two different approaches an *automatic approach* (*Make Parallel Procedure*) and an *interactive approach* (*UDPosition Procedure*). The automatic approach is based on Corollary 1 and on the fact that a sketch includes parallel lines that imply parallel faces of the corresponding 3D model (Theorem 2 in [8]). Specifically, given the geometry and topology of an intermediate wireframe sketch, the key idea is to force cross-section line L_{f_L} to become parallel to a line L_{f_k}, where L_{f_k} is the cross-section line of a visible region r_k, which along with r_L may correspond to parallel faces F_k and F_L. The region r_k is determined as follows (Fig. 5(a)): Firstly, the visible region r_{vL} that is adjacent to ℓ_L is identified. Afterwards, all visible lines are searched in order to find a line ℓ_k parallel to ℓ_L that belongs to at least one region r_k different to r_{vL}. If such a line is detected, then r_k and L_{f_k} are selected for the definition of L_{f_L}. Otherwise, L_{f_L} is determined through the alternative interactive approach.

According to the interactive approach, the user controls the slope of the cross-section line L_{f_L} and different alternative solutions for the corresponding hidden part are obtained (Fig. 5(b-e)). However, in order to produce feasible solutions this procedure is constrained. In particular, the slope of L_{f_L} is constraint by the elements that are associated with r_L and the known cross-section lines that should intersect with L_{f_L} (Fig. 5(b)). In the example of Fig. 5(b) the slope of L_{f_L} is constraint by (i) the point $P_{1,L}$, which is the intersection point of ℓ_L and L_{f_1} (that is already known), must also be on line L_{f_L}, and (ii) the point $P_{2,L}$ which is on the extension of the hidden line $\ell_{2,L}$. In particular, $\ell_{2,L}$ and the adjacent not-specified hidden junction j_h belong in region r_2 and r_L, thus $P_{2,L}$ must be on both L_{f_2} and L_{f_L}. Since, the line L_{f_2} is already known, the different slopes of L_{f_L} are obtained by sliding the point $P_{2,L}$ along line L_{f_2} (Fig. 5(b-d)). These different slopes of L_{f_L} result in solutions for the hidden part associated to the junction j_h.

4.2.2. *Algebraic representation of geometric framework*

Following the solution of system (4.1), a cross-section system (4.3) is generated from the intermediate wireframe sketch on the basis of the hidden lines (Tab. 1). System (4.3) includes equations that correspond to (i) the boundary lines ℓ_L that were excluded from system (4.1), and (ii) each visible line that is adjacent to a T-junction. In this system, T-junctions are considered as hidden junctions and their true position in

Eq. Types / Unknowns	Type 0	Type 1	Type 2	Type 3	Type 4	Type 5	Type 6
(x_h, y_h)or(x_{Th}, y_{Th})	0	1	1	2	2	0	1
(b_L, a_L, c_L)	0	0	1	0	1	1	2
Total Number of Unknowns	0	2	5	4	7	3	8

Tab. 2: The equation types of system (4.3) are generated on the basis of the system's unknowns. The table lines show the number of unknown pairs (x_h, y_h)or(x_{Th}, y_{Th}) or unknown triplets (b_L, a_L, c_L) in each equation type and the total number of unknowns in the equation.

the wireframe sketch is to be calculated:

$$
\begin{aligned}
\ell_{qn} \leftrightarrow eq_{qn} &: (c_n a_q - c_q a_n)(y_{h0} - y_0) + (c_q b_n - c_n b_q) \\
&\quad \times (x_0 - x_{h0}) + (a_n b_q - a_q b_n)(x_{h0} y_0 - y_{h0} x_0) = 0 \\
\ell_{qd} \leftrightarrow eq_{qd} &: (c_d a_q - c_q a_d)(y_{h0} - y_1) + (c_q b_d - c_d b_q) \\
&\quad \times (x_1 - x_{h0}) + (a_d b_q - a_q b_d)(x_{h0} y_1 - y_{h0} x_1) = 0 \\
\ell_{dn} \leftrightarrow eq_{dn} &: (c_n a_d - c_d a_n)(y_{h0} - y_3) + (c_d b_n - c_n b_d) \\
&\quad \times (x_3 - x_{h0}) + (a_n b_d - a_d b_n)(x_{h0} y_2 - y_{h0} x_2) = 0 \\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
\ell_{td} \leftrightarrow eq_{td} &: (c_d a_t - c_t a_d)(y_{Thi} - y_{r-1}) \\
&\quad + (c_t b_d - c_d b_t)(x_{r-1} - x_{Thi}) + (a_d b_t - a_t b_d) \\
&\quad \times (x_{Thi} y_{r-1} - y_{Thi} x_{r-1}) = 0 \\
\ell_{me} \leftrightarrow eq_{me} &: (c_e a_m - c_m a_e)(y_{hi} - y_r) \\
&\quad + (c_m b_e - c_e b_m)(x_r - x_{hi}) + (a_e b_m - a_m b_e) \\
&\quad \times (x_{hi} y_r - y_{hi} x_r) = 0 \\
\ell_{mk} \leftrightarrow eq_{mk} &: (c_k a_m - c_m a_k)(y_{hi} - y_{r+1}) \\
&\quad + (c_m b_k - c_k b_m)(x_{r+1} - x_{hi}) + (a_k b_m - a_m b_k) \\
&\quad \times (x_{hi} y_{r+1} - y_{hi} x_{r+1}) = 0 \\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
\ell_{ke} \leftrightarrow eq_{ke} &: (c_e a_k - c_k a_e)(y_{hi} - y_{hj}) \\
&\quad + (c_k b_e - c_e b_k)(x_{hj} - x_{hi}) + (a_e b_k - a_k b_e) \\
&\quad \times (x_{hi} y_{hj} - y_{hi} x_{hj}) = 0 \qquad (4.3)
\end{aligned}
$$

The unknowns of system (4.3) are the coordinates of each hidden junction j_h and j_T, and all triplets (b_L, a_L, c_L) that were excluded from system (4.1). Thus, system (4.3) includes $L_h + L_L + J_T$ equations and $2(J_h + J_T) + 3 * R_L$ unknowns, where J_T is the number of T-junctions in the sketch. For the solution of system (4.3), the equations of the system are analyzed and classified into seven (7) different types (Tab. 2) with respect to the number and the categories of the included unknowns.

Equation Type 0 occurs when all the unknowns of an equation are calculated. This type is used to assert that a solution is achieved. Equations of *Type 1* and *Type 3* are the simplest equations, since they are generated from a hidden line that is adjacent to two non "L-chain" hidden regions. Equation of Type 1 is associated to a hidden line that is adjacent to one visible and one hidden junction (Fig. 6(a)), while the line that generates a Type 3 equation is adjacent to two hidden junctions (Fig. 6(c)). A *Type 2* equation is associated with a hidden line which is adjacent to one visible junction and belongs to one r_L region (Fig. 6(b)). The lines that generate the equations of *Type 4* and 5 belong also in r_L region. Type 4 corresponds to a hidden line that is adjacent to two hidden junctions (Fig. 6(d)), while Type 5 corresponds to a boundary line ℓ_L (Fig. 6(e)). An equation of *Type 6* corresponds to a hidden line that is adjacent to one visible junction and to two r_L hidden regions (Fig. 6(f)).

Similarly, the coordinates of each hidden junction (unknowns (x_h, y_h) or (x_{Th}, y_{Th})) are classified into 4 Types. More specifically, an unknown pair that belongs to 2 or 3 equations of Type 1 is of *Type A* (Fig. 6(g)), the unknowns that belong to one equation of Type 1 is of *Type B,* while the rest unknowns are classified as *Type C* (Fig. 6(h)). An unknown of Type B also belongs to an equation of Type 2, 4, or 6 and it is associated with three triplets (b_i, a_i, c_i). More specifically, in the case of a minimal intermediate wireframe sketch, a Type B unknown is included in at least one equation of Type 2. Finally, the coordinates of a junction are considered of *Type E* when they are calculated.

4.2.3. *Hidden junction calculation algorithm*

This section presents the *Hidden Junction Calculation Algorithm (HJCALC)* that is developed for the solution of system (4.3). The input of the algorithm is system (4.3), the intermediate wireframe sketch and a corresponding compatible cross-section. The outputs are the coordinates of each hidden and T- junction that completely define the final wireframe sketch, and the values of all triplets (b_L, a_L, c_L) that define the corresponding cross-section lines. HJCALC proceeds as follows:

Hidden Junction Calculation Algorithm (HJCALC)

STEP 1: Identify Equation Types; Identify Unknown Types;

STEP 2: Check for equations of Type$\neq$0. *IF* exist continue with STEP 3, *ELSE* exit with a solution.

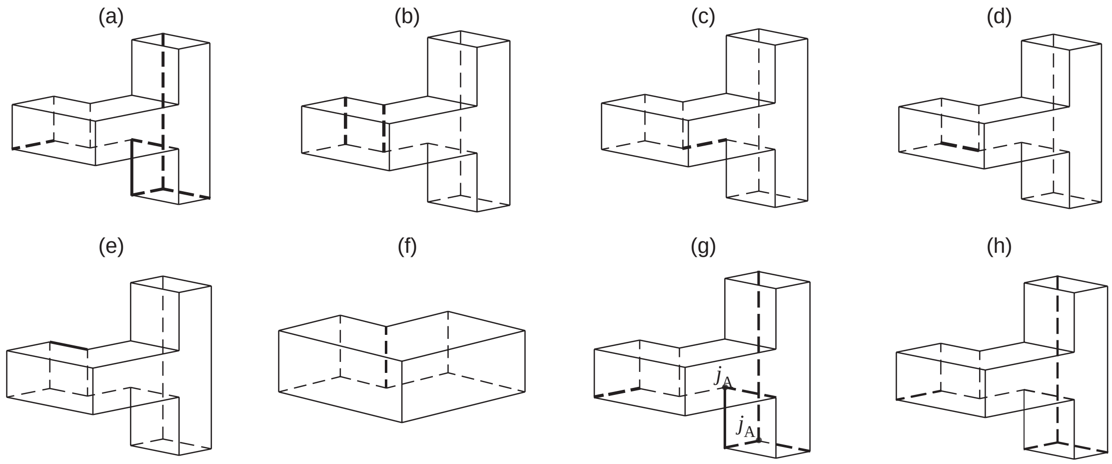

Fig. 6: (a) – (f) The bold marked lines indicate the sketch lines that generate an equation respectively of (a) Type 1, (b) Type 2, (c) Type 3, (d) Type 4, (e) Type 5, and (f) Type 6, (g) junctions that their unknown coordinates are of Type A, and (h) unknown pairs of Type B and C correspond to junctions that are adjacent to "L-chain" hidden regions.

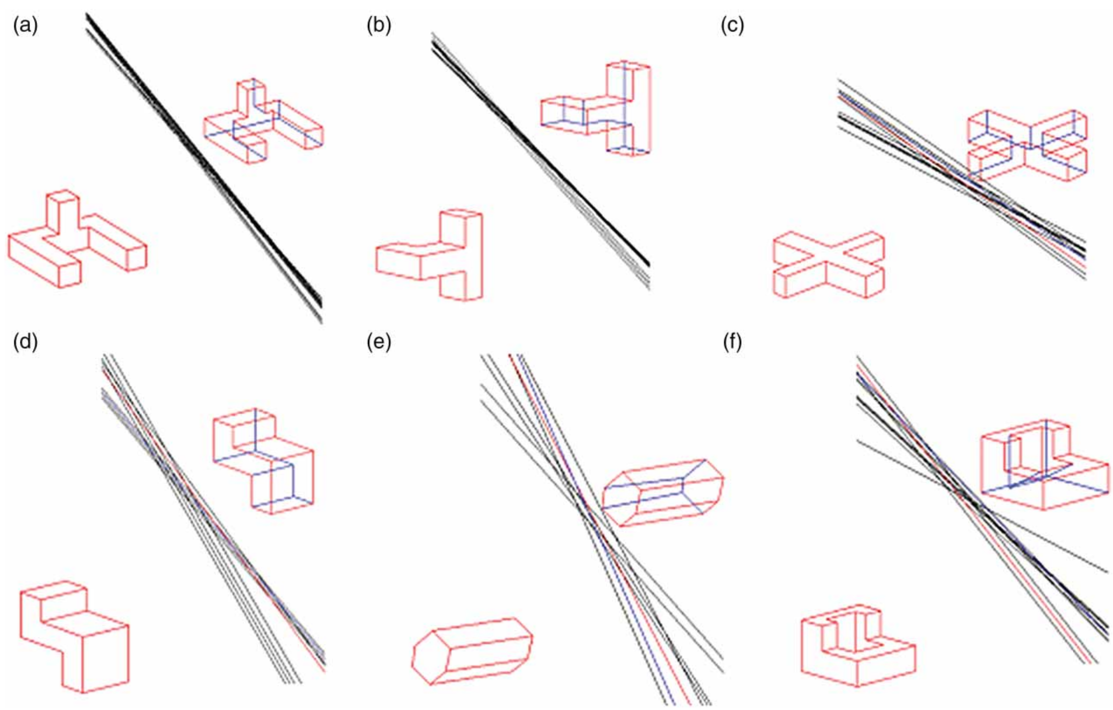

Fig. 7: (a) A simple case with unknowns of Type A, (b-d) the derived system (4.3) for these sketches includes unknowns of all types and the Make Parallel procedure is employed for the determination of the included L-chain region(s), (e) system (4.3) of this sketch does not include an unknown of Type A, and the user – interaction was necessary for the determination of the hidden part, and (f) the derived system includes unknowns of all types, but for the determination of the L-chain hidden region the UDPosition was employed.

STEP 3: Type_A = Find_TypeA $((x_{hi}, y_{hi}))$; *IF* Type_A = *true*, Find equations of Type 1 that include (x_{hi}, y_{hi}); Solve linear system with respect to (x_{hi}, y_{hi}); Return to STEP 1. *IF* Type_A = *false*, Continue to STEP 4.

STEP 4: Type_B = Find_TypeB $((x_{hi}, y_{hi}))$; *IF* Type_B = *true*, Find the equation of Type 2 that include (x_{hi}, y_{hi}) and (b_{Lj}, a_{Lj}, c_{Lj}). *IF* (Calculate = Make_Parallel $((b_{Lj}, a_{Lj}, c_{Lj}))$) = *true*, Return to STEP 1, *ELSE* Calculate = UDPosition$((b_{Lj}, a_{Lj}, c_{Lj}))$; Return to STEP 1. □

HJCALC Algorithm is an iterative procedure that terminates when all unknowns in system (4.3) are determined. During the first iteration, all unknowns of Type A are calculated. In each iteration, the types of equations and the types of unknowns change on the basis of the calculated unknown variables. In particular, when a pair (x_{hi}, y_{hi}) of Type A is calculated the corresponding equations change from Type 1 to Type 0 or from Type 3 to Type 1. When a triplet (b_{Lj}, a_{Lj}, c_{Lj}) is calculated the corresponding equations change from Type 2 to Type 1 or from Type 4 to Type 3 or from Type 5 to Type 0 or from Type 6 to Type 2. In turn, the associated unknown pairs (x_{hi}, y_{hi}) change from Type B to Type A, and from Type C to Type B. Thus, each time an unknown is calculated and at least one hidden junction becomes a junction of Type A or B. The algorithm is applied to selected sketches that include all the studied unknown and equation types as well as different combinations of them. The test case results are presented in Fig. 7.

5. CONCLUSIONS

This paper presented a framework under which the geometry of the hidden part in a wireframe sketch can be determined. Contrarily to other approaches, the employment of the cross-section criterion establishes that, for a given realizable natural sketch, the produced wireframe sketch will correspond to a valid trihedral solid model whose orthographic projection is the given sketch. The proposed system (4.3) is an algebraic representation of the geometric constraints that should be satisfied in the hidden part of a minimal and realizable wireframe sketch. For natural sketches with strong ambiguities, the hidden geometry problem is under-constrained and more algebraic, geometric, or heuristic constraints must be defined with respect to system (4.3). The "Make Parallel" and "UDPosition" procedure are two examples of such constraints. The first one is a fully-automated procedure while the second one is based on the user interaction to address ambiguities in the hidden geometry. Although the final decision for the hidden geometry relies on the user selections, all possible alternatives are constrained to conclude to a valid and realizable solution. In future work, more constraints are to be studied and employed to facilitate the hidden geometry determination process and include possible richer structures.

ACKNOWLEDGMENTS

The research of Ph. Azariadis has been co-financed by the European Union (European Social Fund - ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: THALES (MIS 379516), Investing in knowledge society through the European Social Fund.

REFERENCES

[1] Azariadis, P.; Kyratzi, S.; Sapidis S.N.: A hybrid-optimization method for assessing the realizability of wireframe sketches, 3D Research, 4(1), 2013, Article: 3, http://dx.doi.org/10.1007/3DRes.01(2013)3

[2] Cao, L.; Liu, J.; Tang, X.: What the Back of the Object Looks Like: 3D Reconstruction from Line Drawings without Hidden Lines, IEEE Transactions on Pattern Analysis and Machine Intelligence, 30(3), 2008, 507–517, http://dx.doi.org/10.1109/TPAMI.2007.1185

[3] Company, P.; Contero, M.; Conesa, J.; Piquer, A.: An optimisation-based reconstruction engine for 3D modeling by sketching, Computers & Graphics, 28(6), 2004, 955–979, http://dx.doi.org/10.1016/j.cag.2004.08.007

[4] Company, P.; Piquer, A.; Contero, M.; Naya, F.: A survey on geometrical reconstruction as a core technology to sketch-based modeling, Computer and Graphics, 29(6), 2005, 892–904, http://dx.doi.org/10.1016/j.cag.2005.09.007

[5] Contero, M.; Varley, P.; Aleixos, N.; Naya F.: Computer-aided sketching as a tool to promote innovation in the new product development process, Computers in Industry, 60(8), 2009, 592–603, http://dx.doi.org/10.1016/j.compind.2009.05.018

[6] Grimstead, I. J.; Martin, R. R.: Creating solid models from single 2D sketches, Proceedings of the third ACM symposium on Solid modeling and applications (SMA '95), 1995, pp. 323–337, http://dx.doi.org/10.1145/218013.218082

[7] Kyratzi S.; Azariadis P.; Sapidis N.S: Realizability of a Sketch: An Algorithmic Implementation of the Cross-Section Criterion, Computer Aided Design and Applications, 8(5), 2011, 665–679, http://dx.doi.org/10.3722/cadaps.2011.665-679

[8] Kyratzi, S.; Sapidis, N.: Extracting a polyhedron from a single view sketch: Topological construction of a wireframe sketch with minimal hidden elements, Computers and Graphics, 33(3), 2009, 270–279, http://dx.doi.org/10.1016/j.cag.2009.03.001

[9] Kyratzi, S.; Sapidis, N.: 3D Object modeling using sketches, Information Resources Management Journal, 24(4), 2011, 27–29, http://dx.doi.org/10.4018/irmj.2011100102

[10] Lee, T. Y; Fang, F.: A new hybrid method for 3D object recovery from 2D drawings and its validation against the cubic corner method and the optimisation-based method, Computer-Aided Design, 44(11), 2012, 1090–1102, http://dx.doi.org/10.1016/j.cad.2012.06.001

[11] Lipson, H.; Shpitalni, M.: Optimization-based reconstruction of a 3D object from a single freehand line drawing, Computer Aided

Design, 28(8), 1996, 651–663, http://dx.doi.org/10.1145/1281500.1281556

[12] Ros L.; Thomas F.: Geometric Methods for Shape Recovery from Line Drawings of Polyhedra, Journal of Mathematical Imaging and Vision, 22(1), 2005, 5–18, http://dx.doi.org/10.1007/s10851-005-4779-4

[13] Varley, P.: Automatic Creation of Boundary-Representation Models from Single Line Drawings. Ph.D. Thesis, Cardiff University (2002).

[14] Whiteley W.: Weavings, Sections and Projections of Spherical Polyhedra, Discrete Applied Mathematics, 32(3), 1991, 275–294, http://dx.doi.org/10.1016/0166-218X(91)90004-G