

An Evolutionary Approach to Determining Hidden Lines from a Natural Sketch

Alexandra Bonnici and Kenneth Camilleri
Department of Systems and Control Engineering,
Faculty of Engineering,
University of Malta, Malta
Email: alexandra.bonnici@um.edu.mt

Abstract—This paper focuses on the identification of hidden lines and junctions from natural sketches of drawings that exhibit an extended-trihedral geometry. Identification of hidden lines and junctions is essential in the creation of a complete 3D model of the sketched object, allowing the interpretation algorithms to infer what the unsketched back of the object should look like. This approach first labels the sketched visible edges of the object with a geometric edge label, obtaining a labelled junction at each of the visible junctions of the object. Using a dictionary of junctions with visible and hidden edges, these labelled visible junctions are then used to deduce the edge interpretation and orientation of some of the hidden edges. A genetic algorithm is used to combine these hidden edges into hidden junctions, evolving the representation of the hidden edges and junctions until a feasible hidden view representation of the object is obtained.

I. INTRODUCTION

Freehand sketches are an effective communication tool in engineering design [1], [2], allowing for externalisation of form concepts without the constraints imposed by computer-aided design tools [3], [4]. Such tools however allow for the creation of virtual 3D models of the design tools and are equally essential in the design process. Thus, sketches are typically redrawn using the CAD tools. The goal of our research is to bypass this last step, automatically creating the 3D model from the sketched drawing. Using the pipeline shown in Figure 1, the sketch is first pre-processed with vectorisation algorithms such as [5]–[7] after which, edge labelling algorithms such as [8], [9] are used, assigning a 3D geometric interpretation to the edge. From this, it is possible to determine the number of visible junctions and edges in the sketched drawing and hence the number of visible regions in the sketch. Together with the 3D geometric edge labels, an initial 3D inflation of the drawing may be obtained. However, in order to obtain a full 3D model of the drawing, the unsketched hidden edges and junctions must be deduced from the information held by the visible part of the drawing, that is, obtain a wireframe drawing from the sketch representation of the object.

This paper focuses on this step in the sketch-to-3D interpretation pipeline, using the visible part of the drawing to deduce the number of hidden junctions and edges required to create a valid wireframe representation of the object [10]. In addition, edge interpretation labels will be used to constrain the interpretation and orientation of the hidden edges at the

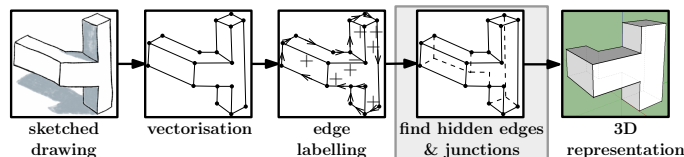


Fig. 1. The sketch-to-3D interpretation pipeline. This paper focuses on the detection of hidden lines and junctions (highlighted), using the edge label information at the visible edges.

visible junctions, thus reducing the number of combinations from which hidden edges can be selected to form the hidden junctions. However, these constraints are not always sufficient to identify the geometry of the hidden view. Thus, we propose to use a genetic algorithm framework, allowing the evolution mechanism to guide the selection of the best fitting representation of the back of the object.

The rest of this paper is organised as follows: Section II gives a review of existing literature on hidden view identification, Section III describes the proposed evolutionary approach, Section V presents the results obtained by the proposed algorithm while Section VI concludes the paper.

II. RELATED RESEARCH

The reconstruction of a 3D model from a single freehand sketch may be resolved in two different ways. One approach is to create a system of equations which represents each plane in the drawing, solving the equations to obtain the 3D co-ordinates of each vertex [11]. Alternatively, optimisation based approaches, driven by heuristics derived from the principals of human understanding of drawings, can be used to determine the optimal 3D co-ordinates of the vertices [12]–[14]. These methods however assume that a wireframe sketch representation of the object is readily available. However, sketching the wireframe of an object may be hard for non-expert sketchers. Moreover, our visual understanding of sketches allows us to infer the hidden parts of the drawing without too much effort [15]. Thus, we, like [10], [15], [16] are interested in creating the hidden topology of the sketch.

Identification of hidden sketch topology typically starts from the geometric information held within the visible, sketched parts. In general, a number of plausible connections between the existing, visible vertices in the drawing are created

to obtain a reasonable, initial wireframe representation of the drawing. This initial representation is then modified by breaking links, introducing new vertex nodes to merge two existing edge branches, or introducing new edge branches to link two otherwise disconnected vertexes [15], [16]. These modifications are carried out in such a way that the final hidden topology satisfies some heuristics, mainly based on human perception principals, such as similarity between the hidden faces and visible faces [15], retaining collinear and parallel relationships, and minimising the number of vertexes in the topology [10]. Exhaustive exploration of all the possibilities with which the visible vertices can be combined to form the hidden topology remains a problem. This can be solved by means of graph theory techniques which allow for multiple hypothesis of the hidden topology to exist in the branches of the tree structure [16].

III. THE SKETCH GEOMETRY

In this work, we will assume that the sketches represent trihedral, solid objects which can therefore be characterised by four junction geometries, commonly referred to as L, W, Y and T junctions. The solid object may be represented by the sketched visible regions R_v consisting of visible lines L_v and visible junctions J_v , and the un-sketched hidden regions R_h with hidden lines L_h and hidden junctions J_h . Since the objects are assumed to be trihedral, all junctions are of order $\mathcal{O}(j) = 3$. However, since the sketch only shows the visible part of the drawing, visible junctions J_v will appear to have different orders which we denote by $\mathcal{O}_v(j)$. The T junction can however be considered as a special case since although the junction appears to be an order $\mathcal{O}_v = 3$ junction, this is a result of a depth discontinuity and the true position of the junction is hidden from view as shown in Figure 2(a). Thus, the visible order of the T junction is $\mathcal{O}_v(j) = 1$ [10]. In this junction geometry, edges may be labelled with a geometric edge label $\lambda \in \{+, -, \rightarrow\}$ indicating convex, concave and occluding edge geometries [8], [17].

We note that individual objects can be arranged into more complex object arrangement as shown in Figure 2(b) - 2(c), giving rise to new junction geometries consisting of coinciding but separable junctions. Such junctions will appear to have visible junction orders of $\mathcal{O}_v(j) = 4, 5, 6$, but variants of the T and Y junctions are also possible. Such junctions are easily identifiable from the geometric edge labels at the junction, where each coinciding edge may be labelled with the edge label $\lambda \in \{\vec{+}, \vec{+}, \vec{+}\}$ [8]. The separable edge label at the junction edges determines the number of hidden edges at the junction. Thus, we introduce three new edge label identifiers namely Y_{s1} which has one hidden edge, Y_{s2} which has two hidden edges and T_s which has one hidden edge.

In order to assign the edge label to the edges at junctions, junction-edge dictionaries consisting of the junction geometry and corresponding edge labels which are possible at the junction are created [8], [17]. We note that these dictionaries can be extended to include the orientation and geometric interpretation attributes of the hidden edge. In this manner,

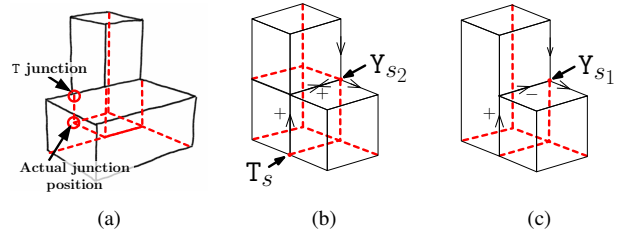


Fig. 2. (a) Showing the sketched T junction and its actual position in the wireframe drawing (b), (c) Drawings exhibiting an extended trihedral geometry, showing the separable edge labels and the Separable Y and T junctions.

knowing the orientations and geometrical edge interpretation of the edges at a junction will allow us to deduce the missing information about the hidden edges. Thus, the junction dictionary Γ can be used to constrain the interpretation of a hidden edge to a limited subset of the possible edge interpretations and orientations. We note however, that this is applicable only to those hidden edges that are members of a visible junction. The drawing may contain any number of additional, totally hidden junctions whose edges are not in contact with any visible junction. With the junction dictionary alone, such junctions would remain undetected. Additional measures are therefore required to deduce the total number of hidden edges and junctions and to group these hidden edges into meaningful hidden junctions in order to obtain the full wireframe representation of the sketched object.

A. Calculating the number of hidden edges in the sketch

The number of lines in a wireframe drawing can be expressed by $N_L = \frac{1}{2} \sum_{j=1}^{N_J} \mathcal{O}(j)$ where N_J is the total number of visible and hidden junctions in the drawing [10]. As shown in Figure 3(a), the wireframe representation of the object may be divided into a visible part and hidden part such that the order of each junction may be expressed as $\mathcal{O}(j) = \mathcal{O}_v(j) + \mathcal{O}_h(j)$ where $\mathcal{O}_v(j)$ is the junction order obtained from the visible component while $\mathcal{O}_h(j)$ is the junction order obtained from the hidden component of the wireframe drawing. Similarly, $N_L = N_{L_v} + N_{L_h}$ where N_{L_v} is the number of visible lines and N_{L_h} is the number of hidden lines; $N_J = N_{J_v} + N_{J_h}$ where N_{J_v} and N_{J_h} are the number of visible and hidden junctions respectively; and $N_R = N_{R_v} + N_{R_h}$ where N_{R_v} and N_{R_h} are the number of visible and hidden regions respectively while N_R is the total number of regions in the wireframe drawing.

The task at hand is therefore to deduce N_{J_h} , N_{L_h} and the junction order $\mathcal{O}_h(j)$ at each $j = 1, \dots, N_{J_h}$. As noted from Figure 3(a), the visible drawing has junctions in common with the wireframe drawing such that using the junction dictionary Γ , and labelled edges in the visible sketch, we may deduce the number of hidden edges at the junctions common to the visible and wireframe drawings. Thus, L, T_s and Y_{s1} junctions each introduce a single hidden edge and a single hidden junction of order $\mathcal{O}_h(j) = 1$. Likewise, Y_{s2} and depth-discontinuous T junctions introduce two hidden edges and a single hidden

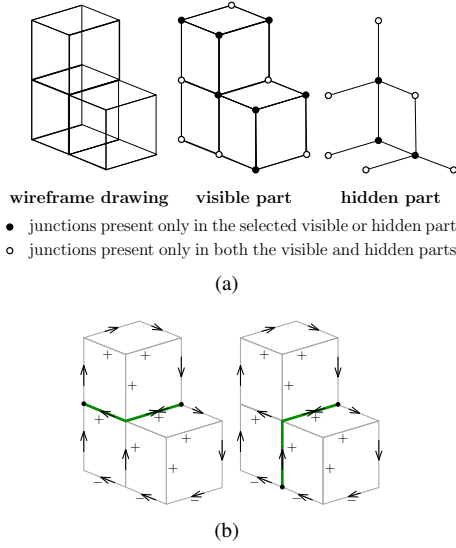


Fig. 3. (a) A wireframe drawing can be divided into a visible part and a hidden part. (b) Two SP paths in the drawing are shown.

junction of order $\mathcal{O}_h(j) = 2$ in the hidden drawing. Thus, the problem is reduced to determining those hidden junctions with orders $\mathcal{O}_v(j) = 0$. Since this work assumes drawings of trihedral geometry, the hidden junctions will be of either order $\mathcal{O}_h(j) = 3$ or $\mathcal{O}_h(j) = 4$. Moreover, order $\mathcal{O}_h(j) = 4$ junctions are due to separable trihedral objects and will only occur when the visible edges are labelled with one of the separable edge labels $\{\vec{+}, \vec{-}\}$. This implies that we can use the edge labels to determine the number of $\mathcal{O}_h(j) = 4$ junctions in the drawing.

Let us define a *separable path* SP as one consisting of visible edges with separable edge label $\{\vec{+}, \vec{-}\}$ where at least one of the terminal junctions is a Y_{s_1} or T_{s_2} junction as shown in Figure 3(b). The number N_{SP} of such paths gives the number of hidden, order $\mathcal{O}_h(j) = 4$ junctions. The number of hidden lines may therefore be expressed as $N_{L_h} = 1/2((N_{Y_{s_1}} + N_{T_s}) + 2(N_{Y_{s_2}} + N_T) + 3N_3 + 4N_{SP})$ where $N_{Y_{s_1}}$, N_{T_s} , $N_{Y_{s_2}}$, N_T are the number of Y_{s_1} , T_s , Y_{s_2} and T junctions respectively and N_3 is the unknown number of junctions with order $\mathcal{O}_h(j) = 3$.

Furthermore, the Euler formula for the wireframe representation of trihedral solids states that $N_R - N_L + N_J = 2$ [10]. Expressing this in terms of the hidden and visible components of the drawing, we obtain $J_h - L_h = 2 - J_v - R_v - R_h + L_v$. The regions in the hidden component of the drawing overlap sufficiently with the visible sketch such that the number of hidden regions R_h may be deduced from the visible sketch [10]. By observation, we note that this may be expressed as $R_h = N_{LL} + N_{LT_d} + N_{Y_s Y_s} + N_{T_s T_s} + N_{T_s Y_s}$ where $N_{Y_s Y_s}$, $N_{T_s T_s}$, $N_{T_s Y_s}$ are the number of $Y_s - Y_s$, $T_s - T_s$ and $T_s - Y_s$ paths respectively. Note that for the scope of determining the number of hidden regions, we make no distinction between Y_{s_1} and Y_{s_2} junctions which are represented collectively as Y_s . Thus, N_3 and N_{L_h} may be obtained.

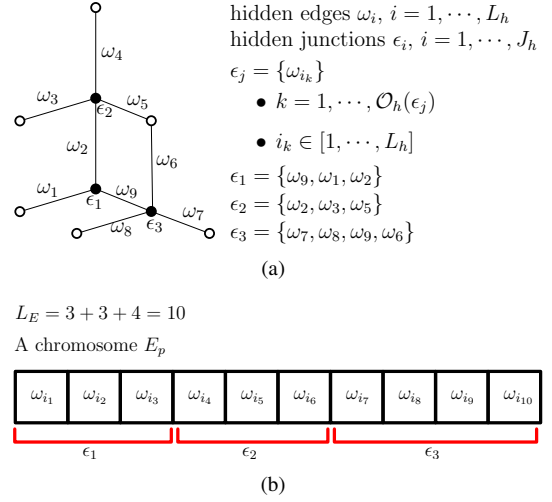


Fig. 4. The representation of the hidden drawing as a chromosome.

IV. GROUPING HIDDEN EDGES INTO HIDDEN JUNCTIONS

Once all hidden edges and junctions are identified, it is then necessary to group the hidden edges such that these form a geometrically fit representation of the drawing. Since an exhaustive search of all possible combinations of edges at each hidden junction can be time consuming [16], we look at different approaches for determining the grouping of edges into hidden junctions and consider a genetic algorithm approach since through the evolutionary mechanisms of crossover, mutation and survival of the fittest, it is possible to start with a random sampling of the search space and evolve the population until this converges to the desired solution [18], [19]. In order to apply the genetic algorithm framework to this problem, we must therefore specify the structure of the chromosome as well as the fitness function which will evaluate the population.

A. The chromosome structure

We will now use a genetic algorithm to determine the grouping of the hidden edges to form the hidden junctions. Let us assign an edge identification number $i = 1, \dots, N_{L_h}$ to each edge ω in the hidden drawing as shown in Figure 4(a). Let us also assign a junction identification number $j = 1, \dots, J_h$ to each hidden junction ϵ as shown in Figure 4(a). The junction ϵ_j is formed by a set of edges such that $\epsilon_j = \{\omega_{i_k}\}$, $k = 1, \dots, \mathcal{O}_h(\epsilon_j)$ where $\mathcal{O}_h(\epsilon_j)$ is the order of ϵ_j and $i_k \in [1, \dots, N_{L_h}]$ as shown in Figure 4(a).

Let the chromosome E represent the hidden edges at each of the hidden junctions such that the chromosome may be partitioned into the respective junctions and expressed as $E = [[\omega_{i_k}]_{j \in I_J}]$, $I_J = 1, \dots, N_{J_h}$ and thus, each gene in the chromosome is an edge of some hidden junction as shown in Figure 4(b). The length of the chromosome can then be defined as $L_E = \sum_{j=1}^{J_h} \mathcal{O}_h(\epsilon_j)$. The population of the genetic algorithm will therefore contain E_p such chromosomes, $p = 1, \dots, P$ where P is the selected population size.

Thus each chromosome in the population represents a possible combination of edges forming the hidden junctions.

Recall that junctions in the visible sketch are labelled with an edge interpretation label, obtained from the use of a junction-interpretation dictionary Γ [7]. Note however that the visible sketch has junctions which have hidden edges and that we can augment the dictionary Γ so that this also lists the possible edge interpretations λ_{ω_i} and edge orientations θ_{ω_i} of these hidden edges. Thus, the labelled visible edges effectively constrain the edge interpretation and line orientation of the hidden edges at the junctions which are shared between the visible and hidden parts of the drawing. The junction dictionary may be further augmented to include the edge interpretation at the hidden junction, such that the dictionary can be partitioned into $\Gamma = \{\Gamma_v, \Gamma_c, \Gamma_h\}$ where Γ_v is associated with junctions whose edges are all visible, Γ_c with junctions that have visible and hidden edges and Γ_h with junctions containing hidden edges only. Genes representing the edges at the junctions shared among the visible and hidden views are constrained such that the edge attributes satisfy $\theta_{\omega_k}, \lambda_{\omega_k} \in \Gamma_c$. The hidden junctions can therefore be grouped into the constrained group Ω_c and the unconstrained group Ω_u and rather than assigning the edges to the junctions in a random manner, we can pick the edges according to the line orientations and edge labels that satisfy these geometric constraints whenever these are known thus:

$$\epsilon_j(k) = \begin{cases} \omega_k \in \Omega_c, & \text{if } (\theta_{\omega_k}, \lambda_{\omega_k}) \in \Gamma_h(\mathcal{O}(\epsilon_j)) \\ \omega_k \in \Omega_u, & \text{otherwise} \end{cases} \quad (1)$$

The chromosome fitness can then assess the validity of this selection and evolve this initial chromosome population accordingly.

B. Chromosome fitness function

Edges $\omega_{i_n}, \omega_{i_m} \in \Omega_c$ assigned to the hidden junction that originate from two distinct visible junctions must intersect at a common point which lies within the visible sketch and which defines the location of the hidden junction. This hidden junction point may be expressed as $\mathbf{x} = \mathbf{v}_{\epsilon_j(m)} + d_m \Theta_m = \mathbf{v}_{\epsilon_j(n)} + d_n \Theta_n$, where d_m and d_n is the unknown length of the edge, $\mathbf{v}_{\epsilon_j(\cdot)}$ is the visible junction position and Θ is a unit vector in the direction of the edge. Thus, we can define the fitness function for the chromosome as

$$\chi(n, m) = \begin{cases} 1 & \text{if } \exists(\mathbf{x}) \text{ and } B(\mathbf{x}) = 1 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

$$Fitness = \frac{1}{J_h} \sum_{j=1}^{J_h} \frac{1}{\mathcal{O}(\epsilon_j)} \sum_n^{\mathcal{O}(\epsilon_j)} \sum_m^{\mathcal{O}(\epsilon_j)} \chi(\epsilon_j(m), \epsilon_j(n)) \quad (3)$$

where $\exists(\cdot)$ is the existential quantifier and B is a binary image containing the bounding box of the sketched object, such that for any pixel within the bounding box, $B(\mathbf{x}) = 1$.

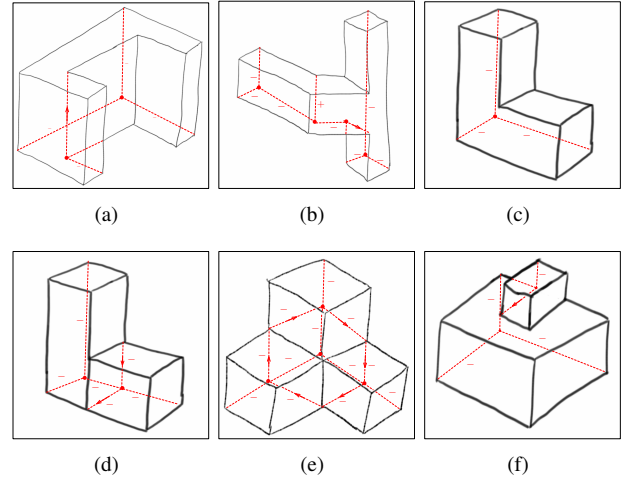


Fig. 5. Results obtained by the proposed method. The detected hidden drawing is shown as dashed lines superimposed on the original sketch.

V. RESULTS

Figure 5 shows the results obtained for a sample of sketched drawings selected from different research papers and replicated for the purpose of this study. The results were obtained using a population size of 100, a mutation rate of 0.03 and a cross-over rate of 0.9, using the half-uniform crossover method whereby half of the differing genes in two parent chromosomes were selected for cross-over. [20]. Stochastic uniform sampling was used as a selection mechanism and the population was allowed to evolve for at most 50 generations. In all cases, the hidden edges and junctions determined by the algorithm resulted in a wireframe representation of the drawing that is geometrically valid in terms. Moreover, the hidden edge interpretations found by the algorithm are also valid interpretations at these edges, thus obtaining a labelled wireframe representation of the drawing.

VI. CONCLUSION

In this paper we use the edge interpretation label information from the visible drawing to constraint the geometric interpretation of the hidden edges that are members of the visible junctions. This focuses the search of the hidden topology. Through this approach, we could relax the drawing assumptions, allowing us to cater for drawings which exhibit up to an extended trihedral geometry, giving a greater flexibility in the creation of more complex sketches. The results obtained compare well with the expected hidden geometry of the object and with similar results documented in elsewhere.

This work is proposed as part of a pipe-line to obtain a 3D representation from a single, offline sketched drawing. The drawing vectorisation and edge-labelling are processing steps carried out in earlier steps in the pipe-line as illustrated in Figure 1. In this work, we assume that these present no errors to the hidden topology algorithms. However, this will not always be the case, and thus, future work will investigate the performance of the hidden topology algorithms under erroneous information about the visible sketch.

REFERENCES

- [1] L. Cruz and L. Velho, "A sketch on sketch-based interfaces and modeling," in *Graphics, Patterns and Images Tutorials (SIBGRAPI-T), 2010 23rd SIBGRAPI Conference on*, 2010, pp. 22–33.
- [2] L. Fu and L. Kara, "From engineering diagrams to engineering models: Visual recognition and applications," *Computer-Aided Design*, vol. 43, no. 3, pp. 278–292, 2011.
- [3] M. Cook and A. Agah, "A survey of sketch-based 3-d modeling techniques," *Interacting with computers*, vol. 21, pp. 201–211, 2009.
- [4] E. Schweikardt and M. D. Gross, "Digital clay: deriving digital models from freehand sketches," *Automation in Construction*, vol. 9, no. 1, pp. 107 – 115, 2000.
- [5] G. Noris, A. Hornung, R. W. Sumner, M. Simmons, and M. Gross, "Topology-driven vectorization of clean line drawings," *ACM Trans. Graph.*, vol. 32, no. 1, pp. 4:1–4:11, Feb. 2013.
- [6] J. Chen, Q. Lei, Y. Miao, and Q. Peng, "Vectorization of line drawing image based on junction analysis," *Science China Information Sciences*, vol. 58, no. 7, pp. 1–14, 2015.
- [7] A. Bonnici and K. P. Camilleri, "Vectorisation of sketched drawings using co-occurring sample circles," in *Computer Analysis of Images and Patterns*, ser. Lecture Notes in Computer Science, G. Azzopardi and N. Petkov, Eds. Springer International Publishing, 2015, vol. 9256, pp. 690–701.
- [8] D. Waltz, *The Psychology of Computer Vision*. McGraw-Hill, 1975, ch. 2: Understanding line drawings of scenes with shadows, pp. 19–91.
- [9] P. A. Varley and R. R. Martin, "The junction catalogue for labelling line drawings of polyhedra with tetrahedral vertices," *International Journal of Shape Modeling*, vol. 7, no. 1, pp. 23–44, 2001.
- [10] S. Kyrtzi and N. Sapidis, "Extracting a polyhedron from a single-view sketch: Topological construction of a wireframe sketch with minimal hidden elements," *Computers & Graphics*, vol. 33, no. 3, pp. 270 – 279, 2009, {IEEE} International Conference on Shape Modelling and Applications 2009. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S009784930900020X>
- [11] L. Ros and F. Thomas, "Overcoming superstrictness in line drawing interpretation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 456–466, April 2002.
- [12] H. Lipson and M. Shpitalni, "Correlation-based reconstruction of a 3d object from a single freehand sketch," in *SIGGRAPH '07: ACM SIGGRAPH 2007 courses*. New York, NY, USA: ACM, 2007, p. 44.
- [13] J. Liu, L. Cao, Z. Li, and X. Tang, "Plane-based optimization for 3d object reconstruction from single line drawings," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 315–327, 2008.
- [14] J. Liu, Y. Chen, and X. Tang, "Decomposition of complex line drawings with hidden lines for 3d planar-faced manifold object reconstruction," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 1, pp. 3–15, 2011.
- [15] L. Cao, J. Liu, and X. Tang, "What the back of the object looks like: 3d reconstruction from line drawings without hidden lines," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 3, pp. 507 –517, march 2008.
- [16] P. Varley, "The use of neighbourhood matching in constructing hidden object topology," in *Proceedings of the World Congress on Engineering*, vol. 1, 2009.
- [17] M. Cooper, *Line Drawing Interpretation*. Springer-Verlag, 2008.
- [18] R. Myers and E. R. Hancock, "Genetic algorithms for ambiguous labelling problems," in *Energy Minimization Methods in Computer Vision and Pattern Recognition*, 1997, pp. 345–360.
- [19] A. Bonnici and K. P. Camilleri, "Exploiting artistic cues to obtain line labels for free-hand sketches," in *Proceedings of the International Symposium on Sketch-Based Interfaces and Modeling*, ser. SBIM '12. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2012, pp. 77–86.
- [20] A. E. Eiben and J. E. Smith, *Introduction to Evolutionary Computing*, G. Rozenberg, T. Back, J. N. Kok, H. Spaink, and A. Eiben, Eds. Springer, 2003.