A Geometric Modeling Tool for Stereo-Matching and Reconstruction of a Model of 3D-Scene

L. Sommellier, E. Tosan, and D. Vandorpe

Laboratoire d'Informatique Graphique Image et Modélisation (L.I.G.I.M.)
Université Claude Bernard Lyon I,
bâtiment 710, 43 Bld du 11 Novembre 1918,
69622 Villeurbanne CEDEX, FRANCE

Abstract. In this paper, we present how a geometrical modeling tool enables to accelerate the matching of stereo-images and to obtain a 3D-model of the reconstructed scene. In the first time, we represent the topology of images of segments using combinatorial maps. This representation enables to efficiently match the images and to construct, during the matching process, the topological model of the matched scene. A boundary representation of the 3D-scene is obtained by embedding this topological model in $\mathbb{R}^3$.

1 Introduction

Computer vision does not consist any more to only find the three-dimensional (3D) information. Many applications need to reconstruct a model of the scene, for example to use it in a CAD modeler or in a virtual world. The analysis and the synthesis domain must therefore cooperate [6,16]. In this article, we propose to introduce a geometric modeling tool within a stereo-vision process. This will enable to accelerate the matching of the stereoscopic views and to easily obtain a B-rep model of the 3D-scene.

The matching of the stereoscopic views is an important stage of the stereo-vision process : it enables to reconstruct the scene by triangulation. The matching consists to find couples of image primitives which are the projections of the same 3D-entity. It remains difficult due to its combinatorial aspect : given a geometric primitive in an image, what is its equivalent in the another image? Two broad classes of methods are used to reduce the extent of the search for matches :

- methods based on *geometric constraints* [1,4,10] (epipolar, disparity, order...). These constraints being local, they are used by combinatorial methods which verify the coherence of the local matches (dynamic programming [14], neural networks [15], integration of matching and surface interpolation [9]...).
- methods based on *topological information* [5] (graphs [1], grouping [11], neighbouring information [7]). These methods use this information to find the matches (graph isomorphism, clique search...). These methods are also combinatorial because the topological information is local and is used like a primitive attribute or an additional constraint.

After the matching process, the matches are reconstructed by triangulation. Then the set of 3D-points can be modelize. This step is difficult too.

We have approached this problem of matching in a context of modeling for the CAD. Our goal is, at last, to automatically construct the CAD model of the viewed object. Therefore it seems natural to use modeling tools during the matching process. We propose to construct geometrical models of the images using topological informations with the double-aim of reducing the combinatorial and of obtaining the 3D-model of the scene.

2 Principle of our approach

The principle of our approach [18,19] lies in the fact that the topological structure (faces, edges, vertices) of a polyhedral object can partly be found in the topological structure of the images of this object (regions, segments, junctions). For example (figure 1), the faces *face1* and *face2* are neighbors. These two faces
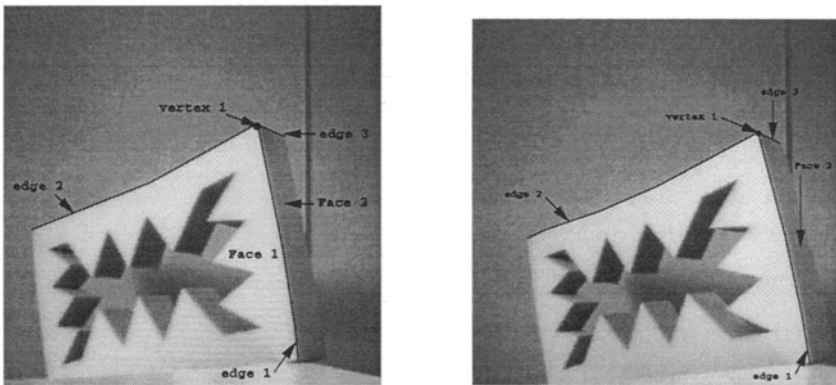

Fig. 1: Topological information about the 3D-object in the images

are the images of two 3D-faces of the viewed object, which are neighbors too. We propose to code this topological information concerning the 3D-object to reconstruct. They will be very useful :

- to structure the matched primitives : for example, we know that the 3D-faces reconstructed using *face1* and *face2* are 3D-neighbors.
- during the matching process :
  - to compare the neighborhoods of two primitives to match.
  - to move coherently in the two images : to go over a face, to follow a contour, etc. Thus, the research of matches becomes a traversal of the topological structures. For example, if the edges *edge1* match, then, by turning around *vertex1*, it is possible to verify that *edge2* and *edge3* match in the two images.

The principle of our approach can be resumed as follows : the perspective projection projects a 3D-structure faces/edges/vertices on two images. Matching the images structures enables, after triangulation, to embed again this structure in $\mathbb{R}^3$ and to find again a partial object model.

We choose to code this topological information with a well known tool in geometric modeling and in CAD : the *boundary representation* (B-rep model) [17].

3 Topological structures

The boundary representation has the particularity of representing surfaces according to their face, edge and vertex subdivision. This surface subdivision is represented by distinguishing the topological structure and the geometrical data. The topological structure [2][20] corresponds to the way in which the faces, edges and vertices are joined in the 3D-model, and the regions, segments and junctions are joined in the image model. The geometrical data correspond to the coordinates or equations of the vertices, edges and faces in the 3D-model, and to the attributes of the primitives (segments and junctions[1]) in the images.

In this paper, we use *combinatorial maps* [13] to code the topological aspect of the boundary representation. When applied to the primitives the map will give informations on their relative positions. It is therefore a tool for primitives grouping.

3.1 Combinatorial maps

Definition 1 - Combinatorial map [12] [3]
A *combinatorial map* $\mathcal{C} = (B, \alpha, \sigma)$ is given by :

- B, a finite set of darts
- α, an involution without fixed points acting on B : $\forall b \in B$, $\alpha(b) \neq b$ and $\alpha^2(b) = b$. α allows the passing from one dart to its opposite in an edge.
- σ, a permutation acting on B : $\sigma(b_{out}) = b_{next}$ with b_{out} the outgoing dart at a vertex and b_{next} the next outgoing dart at this vertex. σ allows the passing from one outgoing dart to the next at a vertex.

A permutation acting on B, ϕ, can also be used and is defined by : $\phi : \begin{vmatrix} B \to B \\ b \mapsto \sigma(\alpha(b)) \end{vmatrix}$

ϕ allows the passing from one dart to the next in a face.
The data of ϕ or σ are equivalent since $\phi = \sigma \circ \alpha \iff \sigma = \phi \circ \alpha$.

Example : in figure 2, edges are $\alpha(b_i^+) = b_i^-$, $\alpha(b_i^-) = b_i^+$; one vertex is $\sigma(b_1^+) = b_2^-$, $\sigma(b_2^-) = b_7^-$, $\sigma(b_7^-) = b_1^+$; one face is $\phi(b_1^-) = b_2^-$, $\phi(b_2^-) = b_3^-$, $\phi(b_3^-) = b_4^+$, $\phi(b_4^+) = b_1^-$.

---

[1] An n-branch junction [8] is the intersection of n segments.
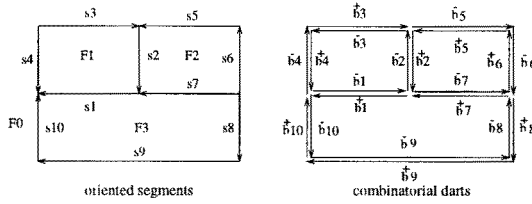
Fig. 2: Embedding of a map in an image of segments

Embedded combinatorial map The combinatorial map $\mathcal{C}$ previously defined is a set of linked darts : it is a purely combinatorial structure. To use it to represent a surface subdivision, it is necessary to link the darts of $\mathcal{C}$ to the geometrical components of the subdivision which they represent. The association of a geometrical element to each dart of the map defines the *embedding* of the combinatorial map.

In our approach, we use combinatorial maps to code the topology of contours images. This contours are approximated by segments. So, we will embed the maps in images of segments.

The embedding of the darts of $\mathcal{C}$ consists to create bijective links between the set of darts and the set of segments. First we link each edge to a segment. As each edge splits up into two darts, we *orient the segments* to associate to each dart a segment and a direction (direct or inverse). This orientation is an *arbitrary* orientation.

Definition 2 - Embedded combinatorial map [19]
The embedding of the map $\mathcal{C}$ in an image of oriented segments is defined by a bijection Q between B and $S \times \{-1, +1\}$ given by :

$$Q : \begin{vmatrix} B \to S \times \{-1, +1\} \\ b \mapsto (s, \pm 1) \end{vmatrix} \qquad Q^{-1} : \begin{vmatrix} S \times \{-1, +1\} \to B \\ (s, +1) \mapsto b \\ (s, -1) \mapsto \alpha(b) \end{vmatrix}$$

Example : in figure 2, the links between the darts and the segments are :
$Q(b_i^+) = (s_i, +1)$, $Q(b_i^-) = (s_i, -1)$, $Q^{-1}(s_i, +1) = b_i^+$, $Q^{-1}(s_i, -1) = b_i^-$.

This notion of embedded combinatorial map is used to create :

- *topological image models* which represent the topological structure of the primitives.
- a *topological matching model* which represent the structure of the matched scene.
- a *topological matched scene model* which is the matched scene map embedded in $\mathbb{R}^3$.

4 Topological modeling of an image of segments

The stereo images are segmented to obtain images of segments. Then, we construct a topological image model by embedding a combinatorial map in each image. This is done in three stages (more details can be found in [19]).

- **Creation of the embedded edges :** each segment is linked to an edge $\{b, \alpha(b)\}$ with Q and Q^{-1}.
- **Detection of the junctions :** after the previous stage, we obtain a set of independent edges. To create vertices, we must find topological relations between the segments. So, we search junctions of segments. We only search 2- and 3-branch junctions because most junctions in the images are such junctions. Junctions with more than 3 branches will be treated in the topological matching model (§5.2).
- **Creation of the junctions in the map :** each segment junction enables to create an edge junction in the map. We link the edges with the permutation ϕ using the Möbius rule to obtain coherently oriented faces.

5 Construction of the topological scene model

In the first paragraph, we only give the *principle* of the algorithm for junction matching. A detailed description of this algorithm can be found in [19,18]. Then we show how to construct a *topological matching model* during the junction matching.

5.1 Principle of matching

Let $\mathcal{C}$ and $\mathcal{C}'$ be two embedded combinatorial maps corresponding to the left and right image. Let's assume that a couple of 3-branch junctions (j_1, j_1') is matched by a geometrical method. The principle of matching is to move, stemming from two matched darts, to the neighbouring couple of 3-branch junctions (j_2, j_2'). Then we verify that (j_2, j_2') match geometrically.

5.2 Construction of the topological matching model

The topological matching model is constructed using the topological image models and the traversal moves of the matching process in these models. This is a new map $\mathcal{C}_m$ created to code the topology of the matched scene. A basic topological matching model is constructed during the 3-branch junction matching. Then we improve it during the matching of 2-branch junctions and we close some of its faces.

During the 3-branch junctions matching We suppose that each 3-branch junction match corresponds to a 3D-junction in the scene. So, for each new match, we create a junction in the map $\mathcal{C}_m$. This new junction is linked to the previously created junction by using the moves in the maps (cf. figure 3). This corresponds to the principle given in §2.

Each vertex of $\mathcal{C}_m$ corresponds to a couple of matched junctions. The edges are embedded in the contours between the matched junctions.
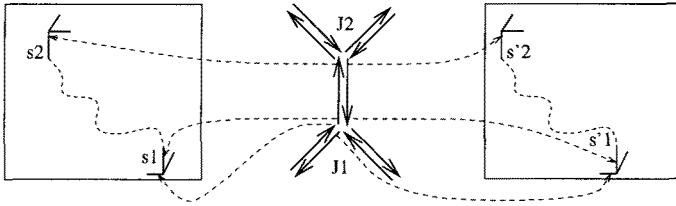
Fig. 3: Embedding of the scene model

Updating during the 2-branch junction matching We update the topological matching model during the 2-branch junction matching. The most robust matches are undertaken first : the matches of contours (broken lines) between two matched junctions. There is an other type of broken lines which match *a*
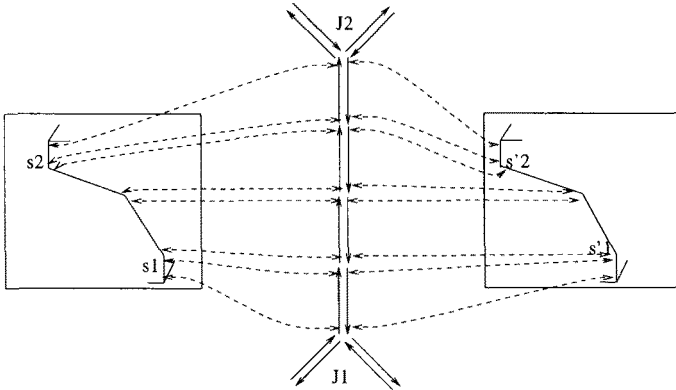

Fig. 4: Embedding of a broken line

priori : the edges branching out from a vertex in the scene model. To match them, we proceed as to match the edges situated between two matched vertices. The edges which are not connected to a matched 3-branch junction are matched as the 3-branch junctions (cf. §5.1).

Closing faces in the topological matching model By closing opened faces of this model, we create new vertices in $\mathcal{C}_m$ which correspond to new matches of junctions in the images. A face of the model will be closed if the corresponding faces in the images are closed at this junction and if the geometrical constraints on the junctions (epipolarity, position, type) are validated.

The closure of the faces in the map $\mathcal{C}_m$ enables :

- the 2-branch/3-branch matching : this case is important when the third branch of a junction was not detected or when a branch is in fact a noise segment.
- the creation of n-branch ($n > 3$) junctions in the model $\mathcal{C}_m$: therefore the detection of only 2- and 3-branch junctions has not impoverished the model of the scene to be reconstructed.

The evolution of the scene model during its construction is shown on a simple example[2] (cf. § "Some experimental results").

5.3   The topological matched scene model

After the matching stage, we have obtained a topological model of the matched scene. This model is embedded on couples of segments. This enables to calculate, by triangulation, the 3D-coordinates of each vertex of the model. After the triangulation, the model is embedded in $\mathbb{R}^3$ (figure 5). It is a boundary repre-
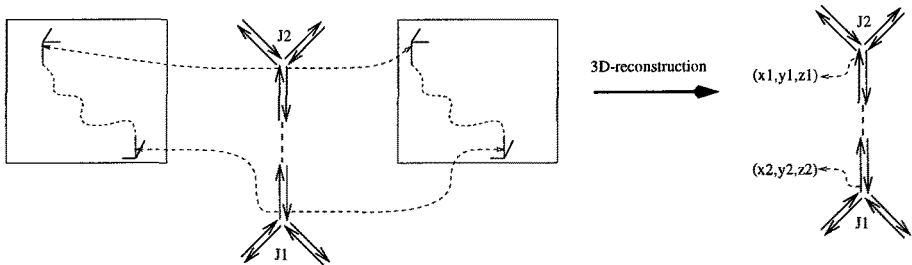

Fig. 5: Embedding of the scene model in $\mathbb{R}^3$

sentation which topological structure is represented by the combinatorial map $\mathcal{C}_m$ and the geometrical data by the 3D-coordinates of the vertices.

5.4   Results and conclusions

We have presented an algorithm for the stereo-matching and the 3D-reconstruction based on the use of geometrical models of the stereoscopic images. These models enable to code the topological informations contained in the stereo-images using combinatorial maps. This approach enables :

---

[2]Examples of stereo matching can be found in [19].

- a substantial reduction of the combinatorial of the matching :
  - The traversal of the models enables to find potential matches and guides the geometrical matching method.
  - The use of the topological matching model enables to eliminate all combinatorial during the matching of the 2-branch junctions linked to previously matched 3-branch junctions. It is just a going over of an edge.

  To assess the reduction in number of matching tests obtained with this cooperative approach, we have compared it to the geometrical matching method used alone. The tests, carried out on real images, lead to a reduction in the number of matching tests varying between 20% and 50% [19,18].
- the construction of the B-rep model of the 3D matched scene during the matching of the images. We have shown how to use the two geometric image models and the traversal moves of the matching process in these models. This enables us to easily obtain the B-rep model of the scene.

Today, we only have a model of the visible part of the object. A possible direction for future researches is to compute the complete CAD model of an object from multiple stereo views by turning around the object.
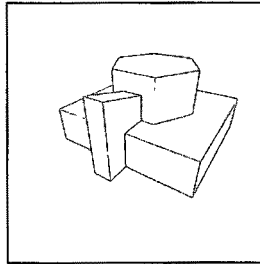
References

1. N. Ayache. *Artificial vision for mobile robots: stereo vision and multisensory perception.* MIT Press, 1991.
2. Baumgart. A polyhedron representation for computer vision. In *National Computer Conference*, pages 589–596, Arlintgton, 1975. AFIPS Press.
3. R. Cori. Un code pour les graphes planaires et ses applications. *Astérisque*, 27, 1975.
4. O. Faugeras. *Three-dimensional computer vision.* Artificial Intelligence Series. MIT Press, 1993.
5. M.M. Fleck. A topological stereo matcher. *International journal of computer vision*, 6(3):197–226, 1991.
6. A. Gagalowicz. Towards a vision system for a domestic robot. In *Journée analyse/synthèse d'images*, pages 63–82, 1994.
7. A. Gagalowicz and L. Vinet. Regions matching for stereo pairs. In *Sixth scandinavian conference on image analysis*, pages 63–70, Juin 1989.
8. M. Herman and T. Kanade. Incremental reconstruction of 3D scenes from multiple complex images. *Artificial intelligence*, 30:289–341, 1986.
9. W. Hoff and N. Ahuja. Surfaces from stereo: integrating feature matching, disparity estimation, and contour detection. *IEEE Transactions on pattern analysis and machine intelligence*, 11(2), Février 1989.
10. R. Horaud and O. Monga. *Vision par ordinateur.* série informatique. Traité des Nouvelles technologies, 1993.
11. R. Horaud and T. Skordas. Stereo correspondence through feature grouping and maximal cliques. *IEEE Transactions on pattern analysis and machine intelligence*, 11(11):1168–1180, Novembre 1989.
12. A. Jacques. Constellations et graphes topologiques. In *Combinatorial theory and applications*, pages 657–673, 1970.

14. P. Lienhardt. Topological models for boundary representation: a comparison with n-dimensional generalised maps. *Computer-aided design*, 23(1):59–82, Février 1991.
15. G. Medioni and R. Nevatia. Segment-based stereo matching. *Computer vision, graphics, and image processing*, 31:2–18, 1985.
16. R. Mohan. Constraint satisfaction networks for vision. Technical report, Exploratory computer vision group IBM Thomas J. Watson research center Yorktown, 1989.
17. Pun and Blake. Relationship between image synthesis and analysis : towards unification? *Computer Graphics Forum*, 9(2):149–163, 1990.
18. A. Requicha. Representations for rigid solids : theory, methods and systems. *Computing surveys*, 12(4):437–464, 1980.
19. L. Sommellier, E. Tosan, S. Bouakaz, and D. Vandorpe. Using combinatorial maps for stereo correspondence and reconstruction. In A. Behrooz, editor, *Proc. Int'l Conf. on Knowledge Transfer Visualization & Graphics'96*, pages 540–546, London, England, Juillet 1996.
20. K. Weiler. Edge-based data structure for solid modeling in curved-surface environments. *IEEE computer graphics and applications*, 5(1), 1985.
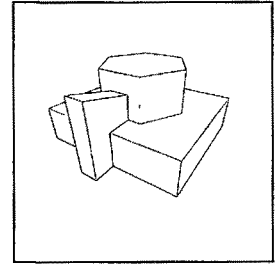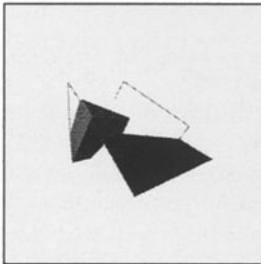
Some experimental results
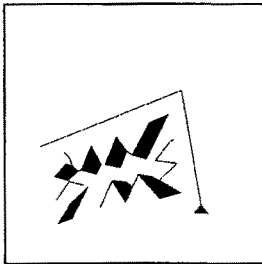
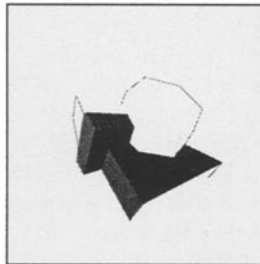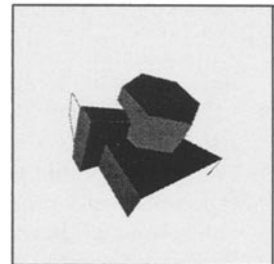A reconstructed object   Left image of contours   Right image of contours

Construction of the 3D matched scene :

After the 3-branch   After the 2-branch   After closing some faces
junction matching   junction matching