

A provably robust and efficient algorithm for reconstruction using structured light

Lyon, France - Liris - **Internal Report**

B. Viguier*, É. Desseree†, J-M. Moreau‡

November 28, 2007

Abstract

This paper presents a robust solution to the problems underlying real-time movement tracking in the medical domain, and more precisely in the course of a radiotherapy treatment. However, a large number of the methods used to reconstruct and visualize movements (or acquire data pertaining to them), is inadequate for the medical environment considered in this paper. The general non-invasive scheme we have adopted is to project light dots on to a scene (*e.g.*, a patient lying on a table), filmed by two cameras. The resolution of correspondances between *source* and *observed* features is modeled using a *match graph* and implemented with an $O(n \log n)$ algorithm (n the number of dots), which only operates on geometric constraints (epipolar geometry), without making any assumption as to the continuity or homogeneity of the observed surface(s), or possible occlusions in the scene.

Keyword: Stereo-vision, epipolar geometry, matching, graph, reconstruction

1 Introduction

1.1 Context

Our long-term research work is focussed on allowing real-time tracking of the breathing movements of a patient lying immobilized on a possibly reclinable (but static) table, and to output morphometric data destined to be fed to simulators and tracking modules for moving organs [1, 2]. More and more methods attempt to answer this problem without resorting to too invasive devices for the patient. Also note that the environment of a radiation treatment room considerably limits the usage of *active* (magnetic, inertial, ...) sensors. Dynamic repositioning of patients has been the subject of studies based on *active vision* in the context of radiotherapy [3, 4]; we are

*benoit.viguier@liris.cnrs.fr

†elodie.desseree@liris.cnrs.fr

‡jean-michel.moreau@liris.cnrs.fr

trying to initiate new tracks for patient movements, with a strong *computer-aided vision* flavor, compatible with the medical context. The movements to be observed may be of various natures, depending on whether they pertain to the deformation of the rib cage (micro-displacements), or to the movement of an arm or even of the entire thorax (macro-displacements).

Rare are the solutions that manage to solve all types of problems; computer-aided vision methods are no exception to this, hence one must make a compromise between speed, precision and robustness, in relation to the resources available. Since recently, structured light has been acknowledged as a very powerful approach when great precision in the results is expected. One may divide methods related to structured light into two main classes:

- *multi-shot* methods, that reconstruct a static object by projecting various patterns over time, and
- *one-shot* methods, that use a single snapshot to reconstruct a scene in interactive time.

In both cases, the main objective is to *match* what is being observed (video cameras, webcams, photographic cameras, ...) with what is projected (video projectors, lasers, ...). The matching problem is common to the passive methods (those only using optical sensors) and the active ones (those using both emitting and capturing sensors); however, the latter methods allow efficient solutions by projecting *recognizable* light patterns.

As shown in several surveys (for instance [5, 6]), many approaches make it possible to solve the matching problem for structured light, and several apply perfectly well to the partial reconstruction of a human body [7, 8, 9]. The recurring idea is to detect characteristic features in the projected patterns or textures, and then to identify them. However, although these techniques allow robust sampling of the observed object, all of them present at least one of the following limitations, in our setting:

- the resolution process is interactive (at most several tens of seconds), but does not meet real-time requirements;
- the observed surface is assumed to be continuous and to generate no occlusion, to enable a correct identification of the various patterns;
- the color of the observed “scene” is assumed to be uniform, to allow projecting and discriminating colored patterns.

Instead, we consider as inevitable the presence of important occlusion phenomena — due to positioning conditions imposed by radiotherapists —, and also the non-uniform pigmentation of the patient’s skin (spots, blemishes, matte complexion, presence of clothes or bandages...). For this reason, we neither use colored patterns nor neighbourhood information in a pattern (which would both help in the *identification* process even in the case of occlusions or discontinuities). The advantage of including such “constraints” is that, because the resulting method suggested here is more robust than the already published ones, it will perform all the better in their more restricted setting (see Section 5.2).

1.2 Informal overview of our work

We use two cameras to film a fixed videoprojected pattern on a given scene. The pattern consists of several “spots” (or subpatterns), each being white, distinct and

indistinguishable from the others. Using three optical devices increases the number of viewpoints on the scene, and allows a greater robustness in the presence of occlusions. Because they are points, the projected subpatterns are less likely to be deformed by the discontinuities of the scene than larger ones would be; their white color allows simple threshold-based detection, even if the color of the scene is not homogeneous. Finally, as all these subpatterns are mutually independent, the partial occlusion of the global pattern does not hinder the detection of the rest of its partially or entirely visible subpatterns. In all the rest of the paper, we shall use the word “robust” when we mean that, if the geometry of the problem does not make it possible to correctly “interpret” the whole videoprojected pattern, we shall always be able to detect and analyze such a (theoretical) situation, and, most importantly, that such an impossibility will not stop the algorithm from recognizing all the theoretically recognizable subpatterns.

We shall progressively refine the matching problem in the various stages of the presentation. The first, informal, description could now be this: how can one match the videoprojected subpatterns and the observed ones, in a robust fashion with respect to the possible occlusions in the global pattern? By assuming the calibration of three optical devices, the resolution of this problem comes down to reconstructing sample points in 3D space. In this paper, we propose a model and a solution for this problem using geometric constraints only, as in three-dimensional particle tracking velocimetry [10, 11, 12]. To do this, we consider that the different optical devices are characterized by continuous pixel planes (*i.e.*, encompass an infinite number of pixels), and, hence, that videoprojected subpatterns are mathematical (dimensionless) points. As a result, the projection of a subpattern lights up a unique mathematical point in 3D space (no deformation nor magnifying effects). The only explicit assumption made is that each videoprojected subpattern is the source of at most one light point in 3D space, which excludes transparent or reflecting objects from the scene.

We shall start with a non exhaustive reminder on epipolar geometry notions before modelling the problem under the form of a *match graph*. We then propose a solution for the matching problem using such a graph, with running time $O(n \log n)$ (n the number of videoprojected points). To end, we present some results for this method, and some perspectives for our futur work.

2 Epipolar geometry

Notational convention: throughout the paper, i, j and k will denote three distinct integers in $\{0, 1, 2\}$ and will be used to identify the elements of the three optical devices described in the sequel. At one or two occasions, expressly mentioned, i and j will belong to $\{1, 2\}$.

Epipolar geometry allows to express a necessary condition for the matching of two points in the image planes of two distinct (emitting or receiving) optical devices; a more detailed presentation of this tool may be found in the book of Faugeras [13]. We formally define an optical system S_i by attaching to it its optical center O_i and its projection plane Img_i . Consider a point P in space, two optical systems S_i and S_j (with $O_i \neq O_j$), and also p_i and p_j the respective projections of P over Img_i and Img_j relative to O_i and O_j : P, p_i, p_j, O_i and O_j are coplanar, and the plane thus formed is qualified as *epipolar*. The reciprocal, named as *epipolar constraint*, implies that two points $p_i \in \text{Img}_i$ and $p_j \in \text{Img}_j$ cannot be the projection of the same space point if p_i, p_j, O_i et O_j are not coplanar (see Figure 1).

Let us decide to note $F_{i,j}$ the *fundamental matrix* that formalizes the epipolar

constraint between the optical systems S_i and S_j . If, and only if, the projective coordinates of image points p_i and p_j in the image planes satisfy $\mathbf{p}_i^T F_{i,j} \mathbf{p}_j = 0$, then p_i and p_j are said to be in *epipolar correspondance*. By noting that $F_{i,j} = F_{i,j}^T$, it is possible to put this equation under the form: $\mathbf{p}_j^T F_{j,i} \mathbf{p}_i = 0$. Furthermore, the image points $p_i \in \text{Img}_i$ (resp. $p_j \in \text{Img}_j$) that are in epipolar correspondance with a unique point $p_j \in \text{Img}_j$ (resp. $p_i \in \text{Img}_i$) form a projective straight line in Img_i (resp. Img_j), represented by $F_{i,j} \mathbf{p}_j$ (resp. $F_{j,i} \mathbf{p}_i$), and called *epipolar line*. The epipolar lines of Img_i et Img_j form line pencils (also called bundles), the origins of which are called *epipoles*; the epipolar condition establishes a one-to-one mapping between the two pencils of lines. Finally, the epipole of S_i relative to S_j will be denoted as $e_{[i,j]}$, and corresponds to the projection of O_j onto Img_i (refer to Figure 1 where epipole $e_{[i,j]}$ is the projection of O_j onto Img_i) (see Figure 1).

If one considers n space points projecting onto n distinct image points in Img_i and Img_j , and belonging to distinct epipolar lines, then the ‘‘pencil bijection’’ allows to compute the correspondances between the image points in both systems. However, if certain space points happen to lie on the same epipolar line, the epipolar condition does not suffice to perform matchings without further assumptions on the observed surface (order preservation, continuity, occlusions, ...).

By using a third optical device, each image point then belongs to two distinct epipolar lines in two separate pencils, and it is not necessary to impose restrictive assumptions to clear up the inevitable indeterminations pertaining to stereovision. Figure 2 illustrates this principle: three space points project onto the image planes of three distinct optical systems S_0 , S_1 and S_2 . In the left subfigure, the epipolar conditions are made explicit on the various image planes: epipolar line $\Delta_i[a]$ is in bijection with line $\Delta_j[a]$. If the relative positions of all optical devices are known, reasoning on those constraints only makes it possible to deduce the spatial location of the various *observed* space points (refer to the subfigure on the right-hand side, where the proportions with the left subfigure are not preserved, for ease of representation).

Although trifocal tensor [14, 15, 16] is a generalization of the fundamental matrix for three optical devices, both handling and solving the problem is more difficult with three optical devices. We now present a graph-based model for such situations, and then propose a robust solution to the matching problem, by exploiting the specific structure of such graphs as much as possible.

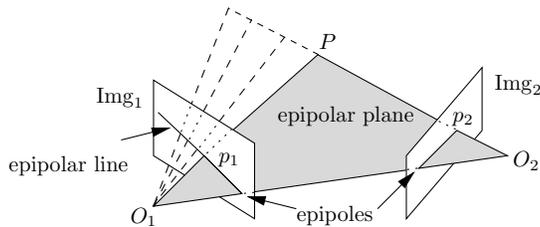


Figure 1: Epipolar conditions among two optical systems.

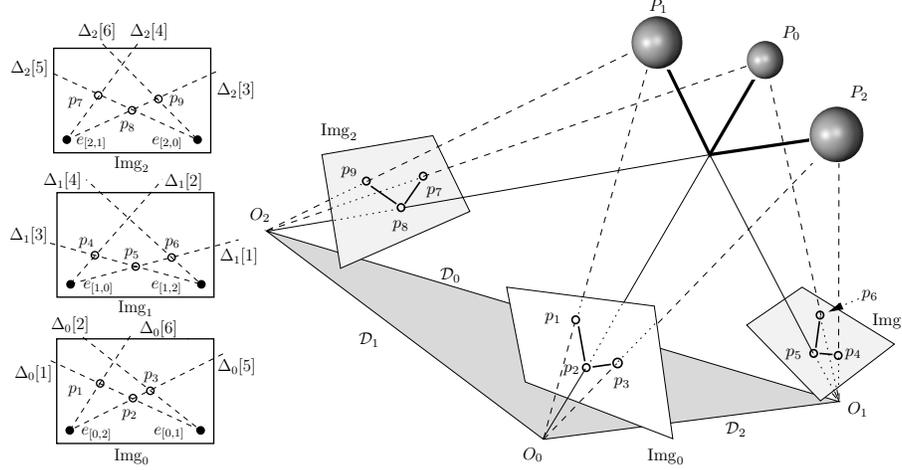


Figure 2: Epipolar conditions among three optical systems.

3 Match graphs

We propose to model epipolar conditions between all image points using a graph to allow the resolution of the matching problem described earlier. Graphs have already been used for similar purposes, and earlier research explain how to establish correspondances between various geometric elements (lines, curves, blobs, ...) from their relative positions [17, 18, 19, 20]. Let us consider three optical systems S_i , the optical centers of which are pairwise distinct, observing (or possibly emitting) n_i image points $p_i \in \text{Img}_i$. $F_{i,j}$ being the fundamental matrix between S_i and S_j , let us construct a *match graph* $G = (V, E)$ by considering each image point as a vertex in the graph of relation \mathcal{R} defined as follows:

$$\mathcal{R}(p_i, p_j) \Leftrightarrow \mathbf{p}_i^T F_{i,j} \mathbf{p}_j = 0.$$

In a match graph $G = (V, E)$, let us agree to note V_i the set of the n_i vertices corresponding to image points in Img_i , and $G_{i,j} = (V', E')$ the subgraph of G induced on vertices $V' = V_i \cap V_j$; note that $V = V_0 \uplus V_1 \cup V_2$ and $G_{i,j} = G_{j,i}$. Let us also define for each vertex $v \in V$ its i -neighborhood $V_i(v) = \{w \in V_i \mid \mathcal{R}(v, w)\}$, *i.e.*, the vertices in V_i that are adjacent to v in G .

The following properties are valid for any match graph $G = (V, E)$; all have a precise geometrical interpretation (in square brackets below), and all are proven in Appendix A.

- (P1) $\{V_0, V_1, V_2\}$ is a partition of V , and G is a tripartite graph over this partition. [No two image points within the same optical system may be matched.]
- (P2) For all i, j distinct in $\{0, 1, 2\}$, $G_{i,j}$ is a bipartite graph, and all its connected components are complete bipartite subgraphs (or bicliques). [All the image points on an epipolar line may be matched with all the image points on the dual line, and vice versa.]
- (P3) For i, j, k distinct in $\{0, 1, 2\}$, $\forall v, w \in V_i$: $(V_j(v) \cap V_j(w) \neq \emptyset) \Rightarrow (V_k(v) \cap V_k(w) = \emptyset)$. [Each image point in Img_i is represented exactly once in G ; hence,

two vertices may not share the same neighbors (see the handling of special cases in the proof)]

Match graphs model epipolar constraints in a given situation, and thus help solve those constraints to find a solution to the matching problem. However, all the graphs are not exploitable ($E = \emptyset$, for example), and all *configurations* of optical devices have their own specificities. In the sequel, a *configuration* $n_{\text{cam}}C/n_{\text{proj}}P$ will involve n_{cam} video cameras and n_{proj} videoprojectors ($n_{\text{cam}} + n_{\text{proj}} \geq 3$ in this paper). We first consider the characteristics of a match graph $G = (V, E)$ in a configuration with three cameras ($3C/0P$), and also the difficulties inherent to the matching problem in such a case. Next, we list the advantages of a $2C/1P$ configuration, which allows to solve the matching problem by exploiting in the best possible way the characteristics of the resulting graph in relation to the acquisition method.

3.1 $3C/0P$

3.1.1 Well-defined graph

In passive stereovision, the image points are defined as *interest points*, that is to say, image points with common graphical features (color, form, ...). To validate the assumption that these interest points correspond, in each optical system, to the projections of the same set of space points, they may be detected on particular epipolar lines or else may result from the detection of physical markers in the observed scene [21, 22, 13]. We shall suppose that this assumption is satisfied, and consider n space points, none of which occluded for all cameras. It is not necessary to impose $n = n_0 = n_1 = n_2$: two space points may be aligned with an optical center, and project onto the same image point — there is no loss of information, contrarily to what happens during an occlusion. Thus, $\forall i \in \{0, 1, 2\}, n \geq n_i$, and the assumption implies that all the space points are at the origin of three vertices of G , in pairwise epipolar correspondance, and hence forming a cycle of length three in G .

(P4) Two distinct cycles of length three in a match graph share at most one vertex.

(Def 1) The cycles of length three in a match graph will be denoted as *3-cycles*.

(Def 2) $G = (V, E)$ is a *well-defined* match graph if all its vertices belong to at least one 3-cycle.

The matching problem could hence be solved by identifying n 3-cycles in G ; unfortunately, the presence of a 3-cycle is not sufficient to infer the presence of a space point. The number of 3-cycles may be superior to n , which calls for the notion of *virtual points* (as opposed to *actual points*). For a set \mathcal{S} of space points and their associated match graph G , the virtual points relative to \mathcal{S} are the space points p such that the graphs of $\mathcal{S} - \{p\}$ and $\mathcal{S} \cup \{p\}$ are exactly equal to G . The image points associated to a virtual point are all the projection of at least two space points, which results in a 3-cycle, each vertex of which belongs to at least two 3-cycles. By extension, a *virtual vertex* will denote a vertex in G belonging to at least two 3-cycles, and a *virtual cycle* will denote a 3-cycle consisting of three virtual vertices. Hence, the actual vertices represent a set of matchings that are part of the solution, including no virtual space point. The graph on Figure 3(a) is built upon the data of Figure 2; the color of a vertex (arbitrarily) indicates whether it belongs to V_0, V_1 , or V_2 . Although $n_0 = n_1 = n_2 = 3$, it is possible to find four 3-cycles (solid edges), among which one *virtual* 3-cycle (center). The vertices of the latter correspond to image points p_2, p_5 et

p_8 on Figure 2, and the virtual 3-cycle that is formed corresponds to the space point located at the intersection of lines (O_0, P_0) and (O_1, P_1) ; whether or not this point is present makes no difference in the observations (and hence in the generated graph).

The virtual 3-cycles convey a certain form of indecision that prevents from telling whether they actually correspond to the observation of a space point or not: hence the matching problem may be stated as the detection of the at most n actual 3-cycles in G . However, note that there exist limit cases where all the 3-cycles are virtual ones, as shown on the match graph in Figure 3(b): $n_0 = n_1 = n_2 = 4$, but eight *virtual* 3-cycles may be found. The situation is nearly similar to that represented on Figure 2 with a new space point P_4 added as follows: let P' be the intersection of line (O_0, P_2) with (O_2, P_0) , and P'' the intersection of (O_0, P_1) and (O_1, P_0) ; P_4 is then the intersection of (O_1, P') with (O_2, P'') .

3.1.2 Incomplete graph

Let us consider a similar situation to the one in the previous subsection, but in which all the space points would not project onto image planes, for some reason (occlusion, reduced field of view, acquisition noise, ...). Such situations are all the more likely as the view points of the optical devices are different. Nevertheless, the associated graph $G = (V, E)$ may present great similarities with a well-defined match graph, since G would be well-defined were all these points visible.

(Def 3) $G = (V, E)$ is an *incomplete* match graph if there exists a well-defined match graph of which it is the subgraph induced by V .

Theoretically, an incomplete match graph G is the subgraph of an infinity of well-defined graphs, those being themselves incomplete graphs: it suffices to imagine that all the space points were visible for all the optical devices, except one point that would be invisible under any view point; the resulting graph is well-defined, but only partially represents the situation.

By *allowing* occlusions in a $3C/0P$ configuration, the detection of actual 3-cycles in G is no more a reliable solution to the matching problem (as opposed to the *well-defined* case); in the worst case, G may be a subgraph induced only on *virtual* vertices. For an example of this, consider the virtual 3-cycle on Figure 3(a). Such situations may not be detected on an incomplete graph, and a 3-cycle may thus not necessarily have a *physical reality*. These problems are inherent to all passive vision methods

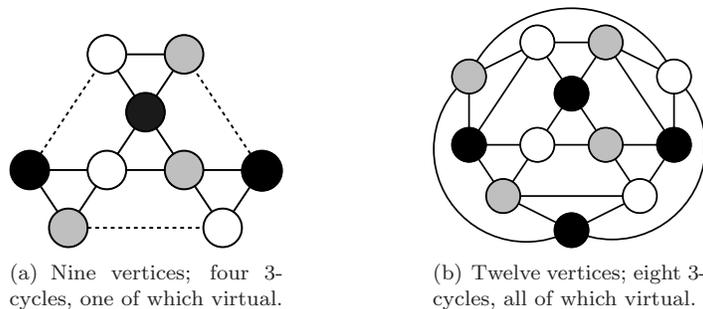


Figure 3: Match graphs.

on indistinguishable space points; when occlusions arrive, the lack of information on these points may lead to several interpretations, no one which being guaranteed to be justified.

3.2 $2C/1P$

3.2.1 The well-defined case

Structured light allows to reduce a certain number of ambiguities in the matching process by using, among the three optical devices, an *emitting* device (laser, videoprojector, ...) to create light points in the three-dimensional space (projection of image points onto space points). Let us consider that, among the three optical systems studied, S_0 is a videoprojector, that the set of image points V_0 is at the origin of $n_0 = n$ light space points, and that all of those are visible (or, in the worst case, aligned) for the two cameras observing the scene. The match graph $G = (V, E)$ associated with this situation is, by construction, a well-defined graph (in the sense $3C/0P$, see (Def 2)). However, using a videoprojector implies new properties that may be expressed by the following axioms:

Axiom 1: Each image point of the videoprojector is at the origin of at most one space light point (*projector axiom*).

Axiom 2: The image points of the cameras correspond to the observation of at most one space light point (*camera axiom*).

These physical considerations on the acquisition mode of image points (hence of a set of vertices in the graph) allow to specify a new class of match graphs that are well-defined in situations involving a videoprojector.

(Def 4) In a $2C/1P$ configuration, a match graph $G = (V, E)$ is *well-defined* if it respects (Def 2) and if, for any $i \in \{1, 2\}$, all the connected components $G' = (V'_0 \cup V'_i, E'_{0,i})$ of $G_{0,i} = (V_0 \cup V_i, E_{0,i})$ satisfy $\text{Card}(V'_0) \geq \text{Card}(V'_i)$ (with V'_0, V'_i and $E'_{0,i}$ respectively included in V_0, V_i and $E_{0,i}$), see Figure 4(a).

This new definition imposes that the number of image points on an epipolar line of the videoprojector is not smaller than the number of image points observed on the dual epipolar line of an image camera. This in turn implies that all *observed* image points are necessarily in epipolar relation with at least one point *emitted* by the videoprojector.

The matching problem for the $2C/1P$ case is identical to the one for a $3C/0P$ configuration, with the only supplementary constraint that each vertex of v_0 should belong to a unique 3-cycle (Axiom 1).

3.2.2 The incomplete case

Using a videoprojector allows to control the sampling of the observed scene, but one is still confronted to the potential occlusion of certain space light points for one or two cameras. The major difference is that, from a mathematical point of view, these space points always project onto the image plan of the videoprojector, whereas one camera may not see some of them. This aspect calls for a refinement of the notion of incomplete match graphs in the $2C/1P$ setting:

(Def 5) $G = (V, E)$ is an incomplete match graph (with V partitioned into V_0, V_1, V_2) if there exists a well-defined match graph $G' = (V'_0 \cup V'_1 \cup V'_2, E')$ of which G

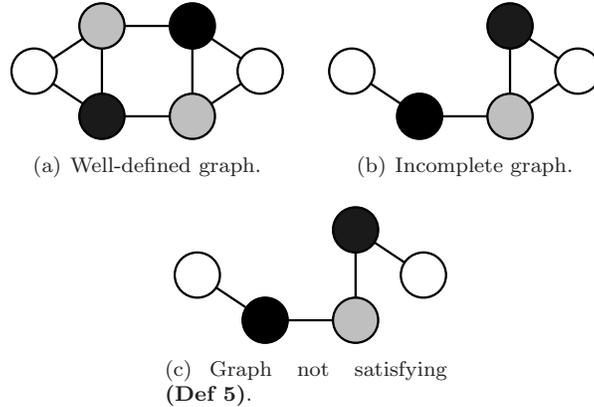


Figure 4: $2C/1P$ match graphs: the vertices of V_0 are shown in black.

is the subgraph induced by V , with $V'_1 \subseteq V_1, V'_2 \subseteq V_2$ et $V'_0 = V_0$ (see Figure 4(b)).

Contrarily to **(Def 3)**, this definition implies that an incomplete graph is the subgraph of only a finite number of well-defined $2C/1P$ graphs, as the vertices of V_0 are preserved. Solving the matching problem hence benefits from this situation:

- either because there exists a unique well-defined match graph G' , of which G is a subgraph — it is then equivalent to solve the problem on G' ,
- or because there exists at least two well-defined match graphs, all of which G is a subgraph — the problem may then only be *partially* solved on the vertices shared by these well-defined graphs.

Therefore, there will always be situations that cannot be solved for the matching of image points. Still, using an emitting optical device allows to reduce the field of theoretical solutions and to yield a better qualification of the potential solutions.

3.3 Other non-trivial configurations

The $2C/0P$ and $1C/1P$ configurations correspond, respectively, to active and passive stereovision methods, which may be assimilated to particular cases of configurations we analyzed earlier, by considering that one of the cameras does not see the scene. We then end up with an incomplete graph to which all the results obtained for previous situations may be applied. Although all situations may be formalized thanks to match graphs, the resolution mechanisms for the matching problem may vary; these perspectives are detailed in Section 5.2.

4 Matching

In this section, we propose a method for the resolution of the matching problem using (possibly incomplete) match graphs resulting from using a videoprojector and two cameras. Although the epipolar condition is not, *a priori*, sufficient to perform

matchings, **Axiom 2** allows nonetheless to exploit in the best possible way the physical properties of the camera points acquisition method — in particular, when an *observed* point is set in epipolar correspondance to only one *emitted* point. To formalize this, we need two new definitions on important elements in match graphs ($G = (V, E)$ is an incomplete match graph resulting from a $2C/1P$ configuration):

(Def 6) We define $V_{\text{pend}} = \{v \in V \mid \text{Card}(V_0(v)) = 1\}$ to be the set of *pending vertices* of G (i.e., vertices that will eventually be made to be match by the algorithm).

(Def 7) Let $G = (V, E)$ with $V_{\text{pend}} \neq \emptyset$; consider i, j distinct in $\{1, 2\}$ and v_0, v_i, v_j (belonging respectively to V_0, V_i, V_j) such that $v_i \in V_{\text{pend}}$, $V_0(v_i) = \{v_0\}$, and v_j the unique vertex closing 3-cycle $\{v_0, v_i, v_j\}$ (unicity is a consequence of **(P4)** subject to its existence, since G is incomplete); $m = \{v_0, v_i\}$ (or $m = \{v_0, v_i, v_j\}$ if v_j exists) is the *pending match* associated to v_0 and v_i .

Following these definitions, it is now possible to state the fundamental property of the pending matches, that the algorithm of this section will use iteratively to solve the general matching problem:

(P5) Let G be an incomplete match graph, and m a pending match in G . Then for each match graph G' of which G is an incomplete subgraph (according to **(Def 5)**), the two (resp. three) vertices of m belong to (resp. form) an actual 3-cycle in G' .

As stated earlier, virtual space points lead to some indecidability and several solutions could be *geometrically acceptable* for a given configuration. Pending matches are the right answer to the matching problem, as they represent all the correspondances that may be deduced from epipolar conditions only; it is thus impossible to perform any further matches without making more assumptions.

Removing the pending vertices makes it possible to reduce the quantity of information in G , to possibly create new matches, and hence to increase the number of feasible matches. Algorithm 1 below applies this principle; we give its content, a justification and then an analysis of its worst-case running time. We assume that the sets of all image points V_0, V_1 and V_2 observed or emitted by each optical device are available, and that the structures G (math graph), F (LIFO queue) and R (set for matches) are initially empty.

4.1 The algorithm

Note: The term *Crossed justification algorithm* (see Algorithm 1) refers to the fact that the algorithm is guided by the necessity to justify the physical presence of each detected point, and finds all the possible correspondances by successive double-checking on the information from the different systems.

4.2 Justification

The three following properties prove the correctness and the termination of the algorithm; their proofs are given in Appendix A.

(P6) Removing a pending match from G does not *remove* any other pending match from G .

(P7) Removing a pending match from G generates at most one new pending vertex.

Algorithm 1 *Crossed justification* algorithm

Require: G : graph, F : LIFO queue, V_i : image points for system i , ($i \in \{0, 1, 2\}$).

Ensure: R : the set of matches.

```
1: for all  $v_0 \in V_0$  do
2:   Insert  $v_0$  in  $G$ 
3: end for
4: for all  $v_i \in V_i$  ( $i \in \{1, 2\}$ ) do
5:   Insert  $v_i$  in  $G$ 
6:   if  $\text{Card}(V_0(v_i)) = 1$  then
7:     Append  $v_i$  in  $F$  {means  $v_i \in V_{\text{pend}}$ }
8:     Mark  $v_i$  as valid {means vertex is present in graph}
9:   end if
10: end for
11: while  $F \neq \emptyset$  do
12:   Extract pending vertex  $v_i$  ( $i \in \{1, 2\}$ ) at head of  $F$ 
13:   if  $v_i$  is valid then
14:     Find unique pending match  $m$  associated to  $v_i$ 
15:      $R \leftarrow R \cup m$ 
16:     Invalidate all vertices in  $m$ , and delete them from  $G$ 
17:     if there exists a vertex  $w$  in  $G$  not in  $F$  but in  $V_{\text{pend}}$  then
18:       Append  $w$  in  $F$ 
19:     end if
20:   end if
21: end while
22: return  $R$ 
```

(P8) Let $G = (V, E)$ be an incomplete match graph, m be a pending match in G , and $G' = (V', E')$ be the subgraph of G induced by $V' = V - m$; then G' is an incomplete match graph.

Using (P8), G remains a match graph at each stage, and (P6) confirms that the order in which the vertices with pending vertices are treated has no incidence on the result, which yields the correctness of the algorithm. That Algorithm 1 terminates is due to line 12 and (P7), that imply that the size of queue F may never increase: since G has a finite number of vertices, F will necessarily become empty.

Once the algorithm is finished, R contains the set of all matches that may be performed on G , using only epipolar constraints; however, this does not imply that G is empty. The *residual* graph of G , if not empty, represents an *unsolvable* situation with the set of assumptions made; it contains no more pending vertex that would otherwise be used to make new matches. In futur work, we would like to reduce the size of the *residual* graph, by using some other assumptions (see Section 5.2).

4.3 Running-time analysis

We detail the implementation model used to allow an $O(n \log n)$ running-time (with $n = n_0$, the number of projected subpatterns). Two important aspects are highlighted:

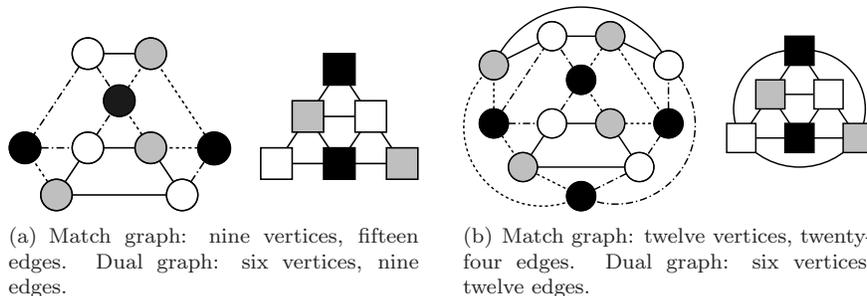


Figure 5: Primal and dual graphs (extracted from figures 3(a) and 3(b)).

an optimization of the structure used to lower the costs pertaining to representing and manipulating a match graph, and an optimization of the various search processes required by the algorithm at different stages.

4.3.1 Structural optimization

To reduce the number of edges in the graph, we use Property **(P2)** introducing the notion of *equivalence class* for sets of vertices, through the connected components of the various $G_{i,j}$ (i, j distinct in $\{0, 1, 2\}$). Property **(P3)** stresses indeed the fact that there is only one vertex $v_i \in V_i$ common to two connected components of $G_{i,j}$ and $G_{k,i}$. On this principle, let $G = (V, E)$ be the incomplete match graph of an observed situation; we build its dual graph, noted $G_{\text{dual}} = (V_{\text{dual}}, E_{\text{dual}})$ as follows: each vertex of G_{dual} corresponds to a connected component of a $G_{i,j}$ -type subgraph generated on G , and each edge of G_{dual} represents a unique vertex in G . By this mechanism, inserting a vertex in the match graph only requires the insertion of exactly one new edge and at most two vertices in the dual graph, as illustrated on Figure 5. Subfigures 5(a) and 5(b) represent: to the left, one match graph G and its connected components (solid, dotted or dash-dotted lines), and to the right its associated dual graph G_{dual} . In both cases, each vertex of G is in bijection with an edge of G_{dual} , and a vertex colored c (black, grey or white) in G_{dual} corresponds to one connected component of the subgraph induced by the vertices with a color *different from* c in G ; in comparison with example 5(a), three vertices are added to 5(b), which yields the creation of nine more edges in G , but only three edges and no extra vertex in G_{dual} .

4.3.2 Lowering the cost of searches

It is possible to optimize significantly the search operations in the algorithm by keeping the vertices of G_{dual} in some sorted order. Each edge in this graph corresponds to an image point in one of the optical systems; the two endvertices of any edge in S_i may be associated with two epipolar lines pertaining to the neighboring optical systems S_j and S_k . Since the epipolar lines form a pencil in the projective (image) plane, they may be put in bijection with the points of a projective line [13], which authorizes ordering epipolar lines and vertices in G_{dual} . Thus, in $G_{\text{dual}} = (V_{\text{dual}}, E_{\text{dual}})$, we find three types of nodes, according to whether they represent an epipolar line between the optical system pair $\{S_0, S_1\}$, $\{S_1, S_2\}$ or $\{S_2, S_0\}$. Each of these sets is sorted using a fixed parameterization of the epipolar lines in the appropriate pencil.

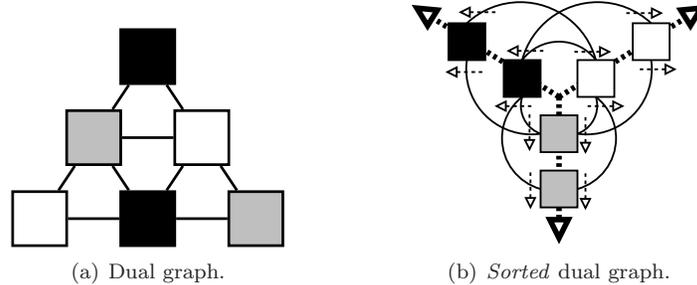


Figure 6: Optimization of search procedures in the dual graph reproduced from Figure 3(a)

On the same principle, it is possible to sort all the outgoing edges of each node in G_{dual} ; the endvertices of these edges belong to two distinct categories of sorted nodes, and are hence also sorted according to the order of their other endvertices. All these (sorted) sets are maintained as height-balanced search trees (AVL, [23]); graph G_{dual} is then optimally represented as a set of three AVL trees, the elements of which are nodes in G_{dual} that are themselves associated to two AVL trees containing the edges of G_{dual} . Figure 6(b) illustrates the reorganization of the dual graph 6(a) to optimize search; each arrow stands for a set of vertices or of edges sorted using an AVL tree.

4.3.3 Details of the running-time analysis

It is now possible to establish the running-times of the various stages in the algorithm, in function of the $n = n_0 = \text{Card}(V_0)$ videoprojected points.

Inserting a vertex: Inserting a vertex v_i in $G = (V, E)$ ($i \in \{0, 1, 2\}$) is equivalent to inserting an edge e'_i between some vertices v'_j and v'_k in $G_{\text{dual}} = (V_{\text{dual}}, E_{\text{dual}})$. The search and possible creation of v'_j and v'_k is performed in an AVL tree, hence in $O(\log n)$ time; similarly, edge e'_i is inserted in an AVL structure in $O(\log n)$ time, yielding an overall $O(\log n)$ for insertion.

Determining whether a vertex becomes pending: Since all the vertices of V_0 are initially inserted in a sequential manner (still in $O(\log n)$ time for each), it is possible to attach to every vertex in G_{dual} a counter indicating the number of edges associated with V_0 vertices in G . Updating this counter takes $O(1)$ time and does not affect the running-time of the insertion procedure. When a vertex v_i ($i \in \{1, 2\}$) is inserted in G , it is linked to at least one vertex in V_0 , and hence the resulting edge in G_{dual} is necessarily linked to a vertex with such a counter (camera Axiom); thus, it is possible to infer in $O(1)$ time from the counter whether the G vertex is pending or not, *i.e.*, whether it is linked to exactly one vertex of V_0 or not.

Finding a pending match: The pending vertices in G are stored in a LIFO queue; hence, retrieving the *next* pending vertex $v_i \in V_i$ ($i \in \{1, 2\}$) requires $O(1)$ time. To reconstitute the entire pending matching, one must find in G the unique vertex from V_0 linked to the pending vertex, and then determine, if it exists, a third vertex in G allowing to form a 3-cycle (Def 7). For the first task,

this unique vertex in G belonging to V_0 corresponds to an edge e'_0 adjacent to edge e'_i in G_{dual} , dual element of v_i . Since it is known at which endvertex of e'_i the search is to be performed, and since e'_0 is the unique edge in the associated AVL, finding e'_0 also takes $O(1)$ time. To end, finding vertex $v_j \in V_j$ (i, j distinct in $\{1, 2\}$) forming a 3-cycle in G with v_i et v_0 leads to finding one edge e'_j in G_{dual} forming a 3-cycle with e'_i and e'_0 . By construction, e'_j links the two endvertices, that are not shared by e'_i and e'_0 ; hence, the two *epipolar values* of the endvertices are known, and the existence of e'_j is checked by finding out the epipolar value of one of its endvertices in the AVL tree associated with the other endvertex. This last step takes $O(\log n)$, which also is the overall running-time for identifying any pending match.

Deleting a vertex: Similarly to the insertion case, removing a vertex in G comes down to deleting an edge from G_{dual} . Finding out the endvertices of the latter in AVL trees, and the effectively removing the edge (and possibly its endvertices) is also performed in AVL structures, so the overall running-time of the operation is $O(\log n)$. In the case where the vertex to be removed from G belongs to F (pending vertex), using a *valid / invalid* flag spares the search for it in the queue.

Discovering a new pending vertex: Proposition (P7) and its proof set forth the way to find out the vertex, if there is one, that is susceptible of becoming a pending vertex after the deletion operation of line 16 in the algorithm. In G_{dual} , this search comes down to determining whether the deletion of a “videoprojector edge” implies that of one of its endvertices, defined in the proof of (P7). Furthermore:

- if deleting this edge implies the deletion of this particular node, no node in G may become pending at this stage;
- else, if this particular node still exists, it is only linked to one edge in G_{dual} corresponding to a vertex in G not in V_0 ; this vertex becomes pending (see proof of (P7) in Appendix).

Both testing the existence of a node in G_{dual} that was not deleted in the previous removal operation, and accessing the unique edge linked to this node take $O(1)$ overall time.

The running-time of all these operations never exceeds $O(\log n)$ time; since all of them is liable to be performed for the at most $3 \times n = 3 \times n_0$ vertices in the match graph representation, the grand running-time for the entire algorithm is hence $O(n \log n)$. This is made possible by using the dual graph, balanced search tree structures, but mainly because match graphs being constructed on geometric observations, it is possible to maintain efficient orders on them at optimal cost, and to optimize several operations that would be more costly in a general graph (such as the detection of 3-cycles, for instance).

5 Conclusion

Developing this method had as a main objective to overcome several known limitations, and to make it possible to adapt to the more demanding medical environment.

- the overall running-time of our algorithm is very promising: the current application operates in interactive time for up to 200 projected points;

- occlusions do not hinder the robustness of the solution;
- the non-sophistication of the projected patterns make it possible to adapt to heterogeneous contexts.

The implementation of the match graph necessitated several important precautions: the fact that the subpatterns are not “mathematical points” may lead to undesirable situations. Unfortunately, it is always possible for a light point to split into two or more spots when located on one discontinuity in the scene; similarly, if the surface is tangential to a light ray, the corresponding subpattern will tend to get longer and lose its intensity.

It is also interesting to note that our work on matching may be compared to the resolution of a constraint satisfaction problem (*CSP* [24]). Each camera vertex is a variable to which one “value” (*i.e.*, one among the vertices in V_0) must be assigned; the constraints that must be reached are:

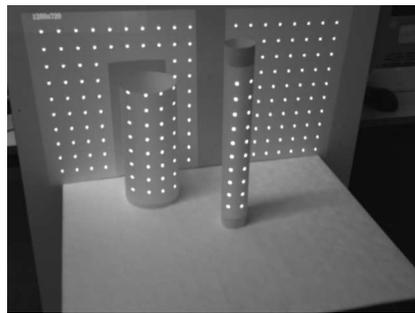
- if a variable is assigned a value, the corresponding nodes in the match graph are linked together;
- if two variables have the same value, the corresponding pairs of nodes in the match graph are linked together;

It is possible to show that making this CSP arc-consistent (*i.e.*, reducing the number of possible values for each variable, see [24]) yields a solution to the matching problem described in this paper: variables with only one possible value are the solution matches. Generic algorithms for arc-consistency have $\Omega(n^2)$ (worst-case) running-time. Note that the solution in this paper is better, due, as mentioned earlier, to the possibility of sorting the geometrical structures, which cannot be done in the general, abstract, setting of CSP.

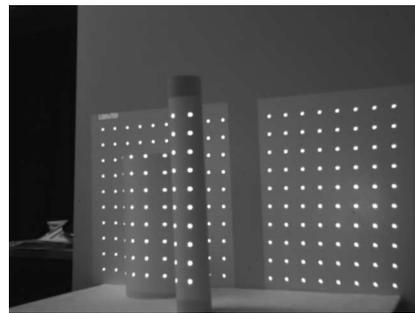
5.1 Results

This work has been implemented with two *firewire* color video cameras (resolution 640×480) and one videoprojector tri-LCD (resolution 1280×720); the code was developed on a machine with a AMD64 X2 3800 processor running at 2GHz, and with 1Gb RAM, under Linux Ubuntu. Communicating with, calibrating and synchronizing the cameras was done thanks to the OpenCV [25]; calibrating the videoprojector was also performed using this library, with the help of the previously calibrated cameras (as in [26]).

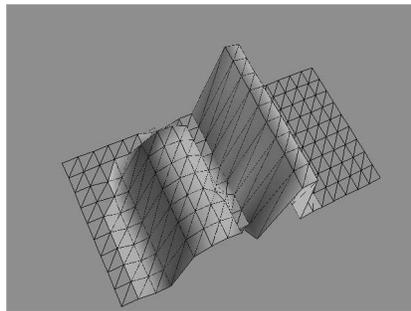
Results on Figures 7 and 8 are obtained with 200 and 150 videoprojecting points. On the videoprojector screen the subpatterns represent squares of 6×6 pixels. As for the light points observed by the cameras, they are, on the average, enclosed in an 8×8 pixels square. These images represent a screen capture of what is computed by our application at any given time; the reconstruction frequency varies as a function of the number of projected points, and is about 5 images per second. The first example illustrates how our model may bring a solution when occlusions occur, or when the *order* of the points is changed by a discontinuity in the scene. The second figure demonstrates the possibility to use such a process on skin, without any alteration to the identification process. The visualisation process was not our priority, so the mesh display is very basic, why the objects seem to be cover by some sort of a sheet. The precision in the results depends entirely on the precision of the calibration process, without forgetting that the calibration of the videoprojector is performed with the help of the calibrated cameras. The test we performed indicates an average error of about 4mm, but we wish to refine this measure in future work.



(a) Left view.

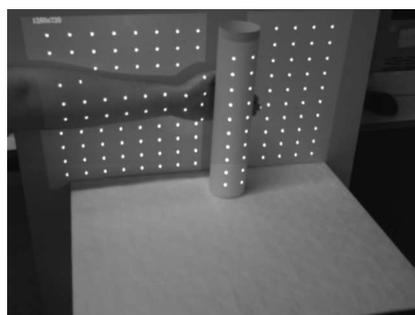


(b) Right view.

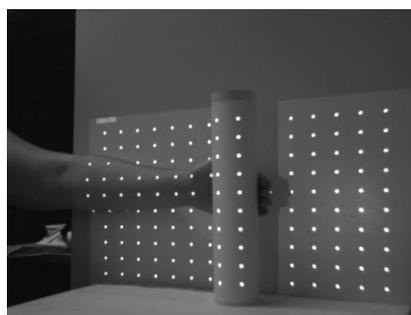


(c) Spatial reconstruction.

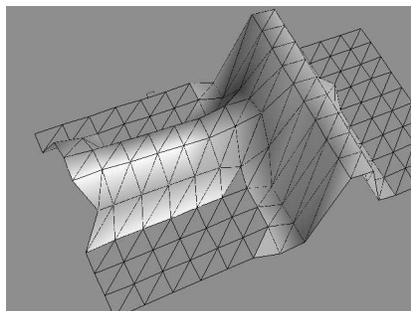
Figure 7: Projection and reconstruction de 200 subpatterns.



(a) Left view.



(b) Right view.



(c) Spatial reconstruction.

Figure 8: Projection and reconstruction of 150 subpatterns.

5.2 Perspectives

Using only epipolar geometry to solve the matching problem addressed in this paper allows a great robustness, since very few assumptions are made about what is being observed. Furthermore, it is very likely that the new acquisition at time $t+\delta t$ represents a situation quite similar to the previous one, at time t , and even more so if one has some knowledge of the *nature* of the observed scene. Our current work exploits the notion of *temporal continuity* in the displacements of light subpatterns by eliminating from the match graph all the points that have not moved more than a given threshold distance relative to the previous reconstruction; we don't exclude to use some theoretical bounds for the spot's displacements, as used by Dipanda and Woo [27]. We also should achieve greater density in the samples by hierachizing the pattern, as proposed by Wong, Peiyi and He [26]. We intend to exploit more thoroughly the interactivity allowed by the real-time monitoring of the videoprojector (as performed by Zhang, Curless and Seitz [8]); it should also be possible to modify the layout of subpatterns to improve the sampling in certain areas, or to clear up recurring ambiguities on a given subpattern. A genetic-type algorithm might, for instance, make it possible to find an optimal pattern, or even to refine the color setting for the detection of light points. Finally, our work does make it possible to use a arbitrary number of cameras or videoprojectors. Doing this requires a generalization of the theory presented in this paper; this is the research we are currently working on. Increasing the number of optical devices should help reduce the *conflicts* in the epipolar constraints [10, 11, 12], and improve the robustness as well as the density of sampling.

A Proofs

- (P1) $\{V_0, V_1, V_2\}$ is a partition of V , and G is a tripartite graph over this partition.

By definition, V_i is the set of image points of S_i and $\mathcal{P} = \{V_0, V_1, V_2\}$ is a partition of V in graph $G = (V, E)$. Furthermore, by relation \mathcal{R} , it is necessary for i and j to be distinct in $\{0, 1, 2\}$ so that $\mathcal{R}(v_i, v_j)$ with $v_i \in V_i$ et $v_j \in V_j$. The endvertices of each edge belong in different classes, so G is tripartite.

- (P2) For all i, j distinct in $\{0, 1, 2\}$, $G_{i,j}$ is a bipartite graph, and all its connected components are complete bipartite subgraphs (or bicliques).

Since G is a tripartite graph (P1), $G_{i,j}$ is bipartite for all i, j distinct in $\{0, 1, 2\}$. Let us now use a proof by contradiction: let $p_i, p'_i \in V_i$ and $p_j, p'_j \in V_j$ forming a complete, non bipartite, connected component of $G_{i,j}$. Neglecting the order, we have $\mathcal{R}(p_i, p_j), \mathcal{R}(p_j, p'_i)$ and $\mathcal{R}(p'_i, p'_j)$. By definition, the interpretation of those relations in epipolar geometry [13] first implies that p_i and p'_i lie on one epipolar line $d \in \text{Img}_i$, dual of the epipolar line containing p_j . Secondly, it implies that the epipolar line containing p'_j is only dual to the epipolar line containing p'_i (namely d), and not to the epipolar line containing p_i , that happens to also be d , a contradiction.

- (P3) For i, j, k distinct in $\{0, 1, 2\}$, $\forall v, w \in V_i$: $(V_j(v) \cap V_j(w) \neq \emptyset) \Rightarrow (V_k(v) \cap V_k(w) = \emptyset)$.

We give a geometrical proof based on epipolar geometry [13]. Suppose there exist $v_i, v'_i \in V_i$ such that $V_j(v_i) \cap V_j(v'_i) \neq \emptyset$ and $V_k(v_i) \cap V_k(v'_i) \neq \emptyset$. By (P2), $V_j(v_i) = V_j(v'_i)$ and $V_k(v_i) = V_k(v'_i)$; hence, v_i and v'_i are both at the intersection of two epipolar lines d_j et d_k relative to systems S_j and S_k . In the general case where $d_j \neq d_k$, this intersection is unique and the corresponding image points are identical; vertices v_i and v'_i represent the same information, but only one of the two is present in the graph. When $d_j = d_k$, let us agree to call *baseline* the line through the two epipoles of Img_i relative to S_j and S_k . The points on this line cannot be matched using only epipolar constraints; in practice, it is assumed that the optical devices are positioned in such a way that the various baselines are not *visible* on any screen.

- (P4) Two distinct cycles of length three in a match graph share at most one vertex.

Let us again use a proof by contradiction: suppose there exist $v_1 \in V_1, v_2 \in V_2$ and $v_0, v'_0 \in V_0$ such that vertices $\{v_0, v_1, v_2\}$ and $\{v'_0, v_1, v_2\}$ form two 3-cycles in G . In those conditions, $v_1 \in V_1(v_0) \cap V_1(v'_0) \neq \emptyset$ et $v_2 \in V_2(v_0) \cap V_2(v'_0) \neq \emptyset$, which is in contradiction with (P3).

- (P5) Let G' be an incomplete match graph, and m a pending match in G' . Then for each match graph G of which G' is an incomplete subgraph (according to (Def 5)), the two (resp. three) vertices of m belong to (resp. form) an actual 3-cycle in G .

Let $G = (V_0 \cup V_1 \cup V_2, E)$ be an incomplete match graph, $G' = (V_0 \cup V'_1 \cup V'_2, E')$ be a well-defined match graph of which G is a subgraph according to definition (Def 5), and let m be a pending match in G .

First consider the case where $\text{Card}(m) = 3$; then m is a 3-cycle in G , hence in G' , and the presence of a pending vertex in m implies that the 3-cycle is actual (not virtual).

Consider next that $\text{Card}(m) = 2$: one must find out whether these two vertices belong to the same 3-cycle in G' , if not in G . For this, note that as incomplete

2C/1P match graphs, G and G' have all the vertices of V_0 in common, by **(Def 5)**; hence, the pending vertex in m is linked to the very same vertex of V_0 in G and in G' . Since G' is, by construction, a well-defined match graph, this pending vertex belongs to a 3-cycle **(Def 4)**; therefore, the latter necessarily contains the two vertices in m and is not virtual due to the presence of the pending vertex in m .

- (P6)** *Removing a pending match from G does not remove any other pending match from G .*

The intersection of two pending matches is not necessarily empty. Let m_1 and m_2 be two pending matches in G such that $m_1 \cap m_2 \neq \emptyset$; by **(P4)** $\text{Card}(m_1 \cap m_2) \leq 1$, hence $m_1 \cap m_2 = \{v_i\}$; then by **(Def 6)** $i \in \{1, 2\}$ and $v_i \notin V_{\text{pend}}$. Consequently $\text{Card}(m_1) = \text{Card}(m_2) = 3$, and the deletion of (say) m_1 from G will yield the deletion of v_i in m_2 , but will not alter the *pending* status of the latter.

In practice, v_i is the projected image point of two light space points; using epipolar constraint only in an incomplete match graph G , it is not possible to know which light spot is *actually* seen. However, whether v_i is removed from G with m_1 or m_2 has no influence on the reconstruction, as a stereo matching is sufficient: although the *logical reconstruction* may differ, the *physical reconstruction* is the same. The algorithm presented in this paper may thus produce two different match sets according to the order in which pending matches are processed, but these match sets are equivalent as they correspond to the same set of space points.

- (P7)** *Removing a pending match from G generates at most one new pending vertex.*

Let m be a pending match in the incomplete graph G , and $v_0, v_i \in m$ with $v_0 \in V_0, v_i \in V_i, i \in \{1, 2\}, v_i \in V_{\text{pend}}$, then $V_0(v_i) = \{v_0\}$ and $V_i(v_0) = \{v_i\}$ (after **(Def 4)**).

Let us now assume that status of vertex $v_{i'} \in V_{i'}, i' \in \{1, 2\}, v_{i'} \notin V_{\text{pend}}$ changes to pending upon removing m from G . Since $v_{i'}$ is not yet pending but will become so when removing v_0 , $\text{Card}(V_0(v_{i'})) = 2, v_0 \in V_0(v_{i'})$ and $v_{i'} \in V_{i'}(v_0)$; hence $i' \neq i$ since $v_{i'} \neq v_i$. Furthermore, $\text{Card}(V_0(v_{i'})) = 2$ implies that $\text{Card}(V_{i'}(v_0)) \leq 2$ (by **(Def 4)**); if $\text{Card}(V_{i'}(v_0)) \leq 1$ then $v_{i'}$ is unique. If $\text{Card}(V_{i'}(v_0)) = 2$, then by **(Def 5)**: $\text{Card}(V_{i'}(v_0) \cap m) = 1$; however, as this vertex is about to be removed from G with m , there only remains one vertex liable to become pending.

- (P8)** *Let $G = (V, E)$ be an incomplete match graph, m be a pending match in G , and $G' = (V', E')$ be the subgraph of G induced by par $V' = V - m$; then G' is an incomplete match graph.*

Let $G'_A = (V'_A, E'_A)$ be an incomplete match graph based **(Def 5)** on a well-defined graph $G_A = (V_A, E_A)$ (with $\{V_{A0}, V_{A1}, V_{A2}\}$ a partition of V_A), m' a pending match in G'_A , m the pending match in G_A corresponding to m' ($m' \subseteq m$), $G_B = (V_B, E_B)$ the subgraph of G_A induced by $V_B = V_A - m$ (with $\{V_{B0}, V_{B1}, V_{B2}\}$ a partition of V_B), and finally let $G'_B = (V'_B, E'_B)$ be the subgraph of G'_A induced by par $V'_B = V'_A - m'$ (with $\{V'_{B0}, V'_{B1}, V'_{B2}\}$ a partition of V'_B): let us prove that G_B is an incomplete graph, and then that G'_B is also one.

If G_B is a *well-defined* match graph, then it is, by construction, incomplete, which proves the first point. If G_B does not satisfy this property, then there

is at least one vertex in G_B that does not belong to a 3-cycle; this comes from the fact that removing m from G_A suppresses the same number of vertices in V_{A0}, V_{A1} et V_{A2} . After **(P4)** and **(Def 6)**, all *broken* 3-cycles in G_B may be reconstructed by adding only one vertex: the non-pending vertex in m that does not belong to V_{A0} . Hence, according to **(Def 5)**, G_B is an incomplete graph.

By construction, G'_B is a subgraph of G_B induced over $V'_B, V'_{B1} \subset V_{B1}, V'_{B2} \subset V_{B2}$ and $V'_{B0} \subset V_{B0}$. Since G_B is itself an incomplete graph, G'_B is an incomplete match graph, by **(Def 5)**.

References

- [1] P.-F. Villard, M. Beuve, B. Shariat, V. Baudet, F. Jaillet, Simulation of lung behaviour with finite elements : Influence of Bio-Mechanical Parameters, Information Visualisation (2005) 9–14.
- [2] A.-L. Didier, P.-F. Villard, J.-Y. Bayle, M. Beuve, B. Shariat, Breathing thorax simulation based on pleura behaviour and rib kinematics, Information Visualisation (2007) 35–40.
- [3] F. Lilley, M. J. Lalor, D. R. Burton, Robust fringe analysis system for human body shape measurement, Optical Engineering 39 (2000) 187–195.
- [4] A. T. G. Baroni, M. Riboldi, R. Orecchia, G. Ferrigno, A. Pedotti, Evaluation of methods for opto-electronic body surface sensing applied to patient position control in breast radiation therapy, Medical and Biological Engineering and Computing 41 (6) (2003) 679–688.
- [5] J. Salvi, J. Pages, J. Batlle, Pattern codification strategies in structured light systems, Pattern recognition 37 (2004) 327–849.
- [6] J. Batlle, E. Mouaddib, J. Salvi, Recent progress in coded structured light as a technique to solve the correspondence problem: a survey, Pattern Recognition 31 (7) (1998) 963–982.
- [7] L. Zhang, B. Curless, S. M. Seitz, Rapid shape acquisition using color structured light and multi-pass dynamic programming, The 1st IEEE International Symposium on 3D Data Processing, Visualization, and Transmission (2002) 24–36.
- [8] L. Zhang, B. Curless, S. M. Seitz, Spacetime stereo: Shape recovery for dynamic scenes, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (2003) 367–374.
- [9] S. R. O. Hall-Holt, Stripe boundary codes for real-time structured-light range scanning of moving objects, The Eighth IEEE International Conference on Computer Vision (2001) 359–366.
- [10] H. Maas, A. Gruen, D. Papantoniou, Particle tracking velocimetry in three-dimensional flows, Experiments in Fluids 15 (1993) 133–146.
- [11] H. Maas, Complexity analysis for the determination of image correspondences in dense spatial target fields, International Archives of Photogrammetry and Remote Sensing 29 (1992) 102–107.
- [12] H. Maas, Robust automatic surface reconstruction with structured light, International Archives of Photogrammetry and Remote Sensing 29 (1992) 709–713.

- [13] O. Faugeras, *Three-Dimensional Computer Vision A geometric viewpoint*, MIT Press, 1993.
- [14] A. Shashua, M. Werman, On the trilinear tensor of three perspective views and its underlying geometry, *ICCV (1995)* 473–920.
- [15] R. Hartley, Lines and points in three views and the trifocal tensor, *Int. Journal of Computer Vision* 22 (1997) 125–140.
- [16] O. D. Faugeras, T. Papadopoulo, A nonlinear method for estimating the projective geometry of three views, *ICCV (1998)* 477–484.
- [17] R. P. Horaud, T. Skordas, Stereo correspondence through feature grouping and maximal cliques, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11 (11) (1989) 1168–1180.
- [18] N. M. Nasrabadi, Y. Liu, Stereo vision correspondence using a multichannel graph matching technique, *Image and Vision Computing* 7 (1989) 237–245.
- [19] T. Koninckx, I. Geys, T. Jaeggli, L. Van Gool, A graph cut based adaptive structured light approach for real-time range acquisition, *3D Data Processing, Visualization and Transmission (2004)* 413 – 421.
- [20] L. B. Ali Shokoufandeh, D. Macrini, M. F. Demirci, C. Jönsson, S. Dickinson, The representation and matching of categorical shape, *Computer Vision and Image Understanding* 103 (2006) 139–154.
- [21] L. Herda, P. Fua, R. Plänkers, R. Boulic, D. Thalmann, Skeleton-based motion capture for robust reconstruction of human motion, *CA'00: Proceedings of the Computer Animation (2000)* 77.
- [22] O. Ghita, J. Mallon, P. F. Whelan, Epipolar line extraction using feature matching, *Proceedings of the Irish Machine Vision and Image Processing Conference (IMVIP 01) (2001)* 87–95.
- [23] G. M. Adel'son-Vel'skii, E. M. Landis, An algorithm for the organization of information, *Soviet Mathematics Doklady* 3 (1962) 1259–1262.
- [24] E. Tsang, *Foundations of Constraint Satisfaction*, Academic Press, New York, 1993.
- [25] Intel, library for Computer Vision, freely available on <http://www.intel.com/technology/computing/opencv/>.
- [26] A. K. C. Wong, P. Niu, X. He, Fast acquisition of dense depth data by a new structured light scheme, *Computer Vision and Image Understanding* 98 (3) (2005) 398–422.
- [27] A. Dipanda, S. Woo, Towards a real-time 3d shape reconstruction using a structured light system, *Pattern Recognition* 38 (10) (2005) 1632–1650.