ICDAR '97

Algorithms for the Hydrographic Network Names Association on Geographic Maps

R. Mariani+, M.P. Deseilligny+, J. Labiche++ and R. Mullot*

+ Institut Géographique National, DT/SR/MATIS, 2 Avenue Pasteur 94160 Saint-Mandé, FRANCE.

++ LACP, Faculté des Sciences, Université de Caen, 14000 Caen, FRANCE.

La3i-LACIS, UFR des Sciences et Techniques, Université de Rouen, 76000 Mont Saint Aignan, FRANCE.

Email: mariani@matis.ign.fr, tel: 01 43 98 81 28, Fax: 01 43 98 85 81.

Algorithms for the Hydrographic Network Names Association on Geographic Maps

Abstract: In this paper, we describe an algorithm which associates the toponyms to the hydrographic network represented in the French geographic maps at scale 1/25000. This work, realized at the French Institut Géographique National (IGN), is included in a project of automated map interpretation for Geographical Information System (GIS) data capture. Strings were already extracted and proved to be syntactically and phonetically correct, and the hydrographic network was already reconstructed, hierarchically attributed, and represented as an attributed non directed graph. The problem of associating a set of words to a piece of the graph is addressed as a minimum weighting bipartite matching, implemented with the Hungarian algorithm. The bipartite graph is constructed using cartographic and syntactic rules.

Keywords: geographic map understanding, image analysis, model, minimum weighting bipartite matching, attribute computation, graph theory.

1 Introduction

We describe a high-level method to compute the association between the toponyms and the hydrographic network represented on the French geographic maps. This method is included in a project of automated map interpretation for GIS data capture started several years ago [1]. Maps' scale is 1/25000 and they are scanned at 500 dpi. As IGN is a map producer, we directly work on the blue plane, used to represent only hydrographic data, and so we avoid the color separation problem encountered for the paper-based applications [2, 3]. The hydrographic network, represented with dashed lines, interrupted solid lines and fragmented areas, was already extracted, reconstructed [4] and hierarchically attributed [5], in order to obtain an attributed non directed graph, which is geometrically and topologically correct. In the same way, strings were previously extracted [1], and proved to be syntactically and phonetically correct.

Then, starting with an hydrographic network graph (HNG) and all the extracted words, the purpose is to find the elements of HNG indicated by a name, a name being an ordered set of words. This problem is addressed as a minimum weighting bipartite matching (MWBM) and is implemented with the Hungarian algorithm [6]. The bipartite graph is constructed using, as a priori knowledges, the cartographic and syntactic rules defined for the toponyms.

In the following section, we situate this work among the literature. Sections 2 and 3 gives respectively the cartographic and syntactic rules an ordered set of words must verify to be grouped in a single name. The section 4 describes the algorithm used to identify the likely elements of HNG indicated by a single word. The formalism of MWBM for the names creation process and the attribution to the HNG is presented section 5, and section 6 shows the bipartite graphe construction. Results are given section 7, before to conclude.

2 Related works

The problem of identifying an element designated by a name is difficult. Many works on cadastral maps [7, 8, 9], engineering drawings [10, 11], layered maps [12], geographic maps [13], have treated this problem. Generally, this attribution has been accomplished in a small geometrical neighborhood of a toponym. For example, the number of parcel is attributed to a parcel only by testing the inclusion of this number in the polygon symbolizing the parcel. The principal difference between those works and the proposed one, is that a toponym is often composed by several **distant** words, and the geometrical neighborhood of a toponym may be very large.

Here, the problem concerns the names or toponyms used in geographics maps to identify the hydrographic network. Those names are generally composed by a set of words which are placed according to the shape of the network. Then, an important step is to find those names using the geometry of the network, the cartographic and syntactic rules. There are two major problems: 1) a name may be very long; 2) geographic maps are very dense, then cartographers may not respect the drawing rules to place the toponyms.

3 Cartographic Rules

We describe the cartographic rules used by the cartographers to represent the names of the HNG. We consider a name as an ordered set of words S. Let P the piece of the HNG indicated by S. S is said to be "cartographically correct", if the following constraints are verified, for each couple of two successive words of S.

1. They have the same police (style, font).
2. They have the same size.
3. The distance between them is less or equal to 20 times their body.
4. They are close to P.
5. They are placed at the same side of P.
6. The words of S are placed according to the shape of P.
7. P may keep the same nature.
8. If P is long, S may be repeated.

The following figure 1 illustrates some definitions. The image (a) defines the body and the size of a character, and the image (b) describes S, P and HNG.

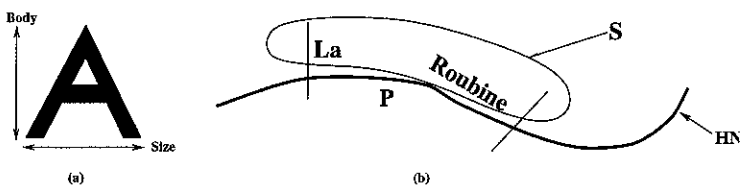


Figure 1: (a) Body and size of a A. (b) S,P and HNG

The **distance** between two words w_1 and w_2 (3) is defined by the distance of the last character of w_1 to the first character of w_2. To take into account the property (5), we have to consider that P is oriented, then for a P's edge, we have the **left** and the **right side**. This constraint implies that if the name of P is S, then S cannot intersect P, and if S intersects P, then S cannot be the name of P. The property (6) is a special case of the properties (4) and (5). (7) means that if P changes of hierarchical level then the name which indicates P will change too (there are very few exceptions). Finally, (8) will not be implemented here. Generally, a water course is represented on several maps, each map describing a piece of it. Then, the connectivity [14] of the natural (or real) network graph is not kept by the cartographic network, due to the map windowing which creates several connected components. Whereas, this connectivity property is very important to implement this last property.

The definition of **distance**, the way P and S are computed, and the method which calculates the matching are described later. The next section defines the syntactic rules used for the French maps, in order to decide if S is syntactically correct.

4 Syntactic Rules

In our case, we have a very specific problem because among all the hydrographic toponyms, the majority are proper names, with few adjectives. We can only build a parser using the definite articles (le, la, les, l'), the prepositions (du, de, des, d', à), the conjunction (et), the parenthesis and the words beginning with a vowel or consonant. In fact, a sentence S, represented as an ordered set $\{w_1, ...w_n\}$ of words will be considered as syntactically correct, only if the set $S^* = \{w_0, w_1, ..., w_n, w_{n+1}\}$ is correct. Here, w_0 and w_{n+1} are respectively the starting point and the ending point of S^*, and their value are $\emptyset_1$ and $\emptyset_2$. S^* is correct if:

- each couple (w_i, w_{i+1}), $i = 0..n$ of S^* is correct.
- All the parenthesis of S^* are matched.

Now, we can describe the table which indicates if a couple (a, b) of words is correct. The following table shows vertically the possible values of a and horizontally the possible values of b. The "x" symbol means that the transition from a to b is possible ((a, b) is correct). W (resp. C) indicates that the proper name starts with a vowel (resp. consonant):

	$\emptyset_1$	le	la	les	l'	du	de	des	d'	à	et	(	)	W	C	$\emptyset_2$
$\emptyset_1$		x	x	x	x									x	x	
le															x	
la															x	
les														x	x	
l'														x		
du															x	
de			x		x										x	
des														x	x	
d'														x		
à			x		x										x	
et		x	x	x	x	x	x	x	x	x				x	x	
(		x	x	x	x									x	x	
)														x	x	x
W						x	x	x	x	x	x	x	x	x	x	x
C						x	x	x	x	x	x	x	x	x	x	x

With the cartographic and syntactic rules, we are able to formalize our problem. In a first time we describe a method to find the likely pieces of HNG indicated by a single word. Then, we present the formalism of the MWBM for the names creation and association to the network. Finally, we show the way to construct the bipartite graph according to the predefined rules, and we develop the Hungarian algorithm which encodes the MWBM.

5 Identification of HNG's elements indicated by a word

This step is realized taking into account the cartographic rules defined before (cf. § 3). The result of this operation will provide two kinds of hypothesis for a word: the left and the right hypothesis.

The following paragraph recalls some definitions on the graphs. After we describe the geometric features associated to a character, in order to calculate the **principal vector** characterizing a word. Then, using this vector which is oriented in the reading direction, we will determine all the HNG's elements this vector may designate. This last operation is realized with a **visibility algorithm**.

5.1 Some definitions

Let $G = (X, U)$ a non directed graph. Let $C = \{s_0, s_1, ..., s_n\}$, a set of vertices, where $s_i \in X \ \forall i \in 0...n$, and $(s_i, s_{i+1}) \in U \ \forall i \in 0..n-1$. Let $deg(s)$ the degree of the vertex s.

C is an **elementary path** of G if $deg(s_i) = 2 \quad \forall i \in 0..n$, and C is a **chain** of G if $\{s_1, .., s_{n-1}\}$ is an elementary path and $deg(s_0) \neq 2$ and $deg(s_n) \neq 2$.

We call $hlevel(u)$ the hierarchical level of the edge u [5] (e.g. dashed line, solid line or textured area). We define $hlevel(E)$ the hierarchical level of an elementary path E, evaluated as the level which appears the most frequently in the set of edges E.

Finally, $G' = (X', U')$ is a **subgraph** of G if $X' \subseteq X$ and $U' \subseteq U$.

5.2 Features associated to a character

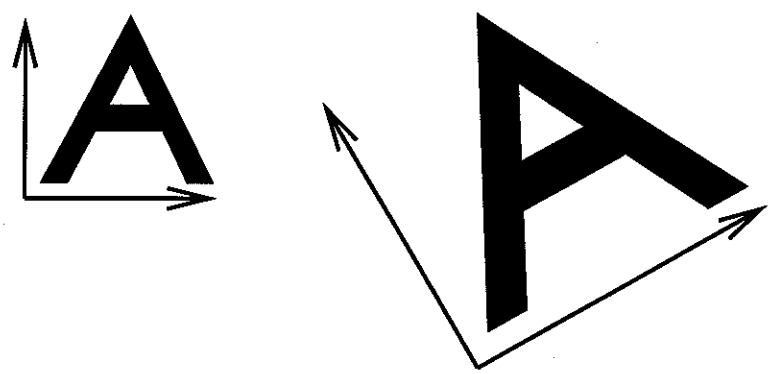


Figure 2: references associated to A

The first image defines the reference associated to the prototype character A, stored in the learned database. The second image shows the new reference, computed during the OCR step [15] to recognize this skewed A. Briefly, this is done by starting from the initial reference and after by estimating the coefficient for scale, rotation and translation. Then, for each character, we have its police, its orientation, its size and its ascii code.

5.3 Computation of the principal vector of a word

We define the center of a character (CC) (cf. figure [3]) for a fixed police as the center of its bounding box. This is stored in the prototypes used to recognize the characters in the map [1]. Let w a word and L the set composed by the CC of each character of w. The principal vector $\overrightarrow{P_w}$ of w is computed using the first (centroïd coordinates) and second order moments (axes of the inertial ellipse) of L [16]. $\overrightarrow{P_w}$ is oriented in the reading direction.

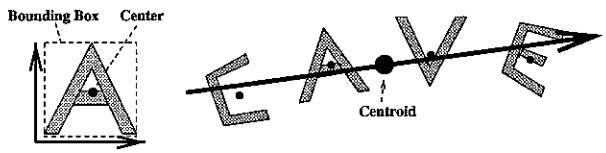


Figure 3: (a) Bounding box and center of a character (b) Principal Vector of "CAVE"

5.4 Identification of HNG's elementary pathes using the principal vector

The aim of this operation, illustrated below (cf. figure [4]), is to identify the HNG's elements which are likely designated by a toponym w. We have a box constructed with $\overrightarrow{P_w}$ and its corresponding normal vector. This box defines the **visibility domain** of w. This visibility domain contains some edges of the HNG, obtained by clipping the graph with the box. The problem is to compute the HNG's elementary pathes visible by w and to orient them in the reading direction.

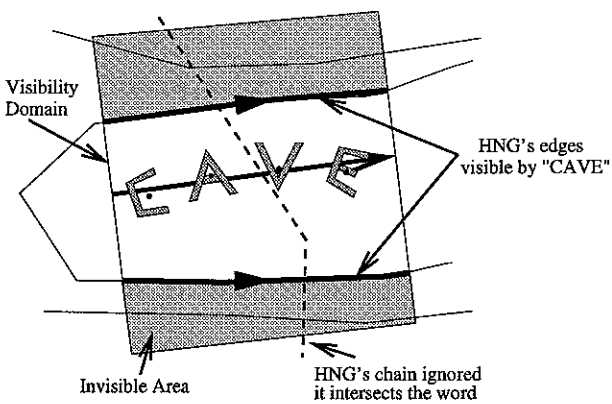


Figure 4: (a) Visibility Algorithm

We consider that a word w cannot generate more than two hypothesis: one **left** and one **right** hypothesis. This is due

to the fact that cartographers place the names to make easier as possible the reading of the map, avoiding confusion (e.g multiple possible name association). The left hypothesis (resp. right) of w is the hypothesis present at the left side (resp. right) of $\overrightarrow{P_w}$. Before to describe the algorithm, we give some definitions.

At the beginning, we determine all the edges which intersect $\overrightarrow{P_w}$, and according to the rule (5) (cf. § 3), we do not consider their corresponding chains for the identification process. After, we use a *visibility algorithm* [17] in order to obtain the "visible" parts of the chains of HNG. Those parts are elementary pathes of G.

Visibility is calculated with the Watkins algorithm [18] depicted (cf. figure [5] (b)). This algorithm has been used in image synthesis, for computing the hidden part of a three dimensional scene. The principle is to consider that the user is at an infinite distance of the screen. Thus, for each screen line L, we create an orthogonal plane to the screen and we cut the three dimensional objects with this plane (cf. figure [5] (a)). Finally, we apply a two dimensional algorithm to display the visible parts of edges (cf. figure [5] (b)) at screen line L.

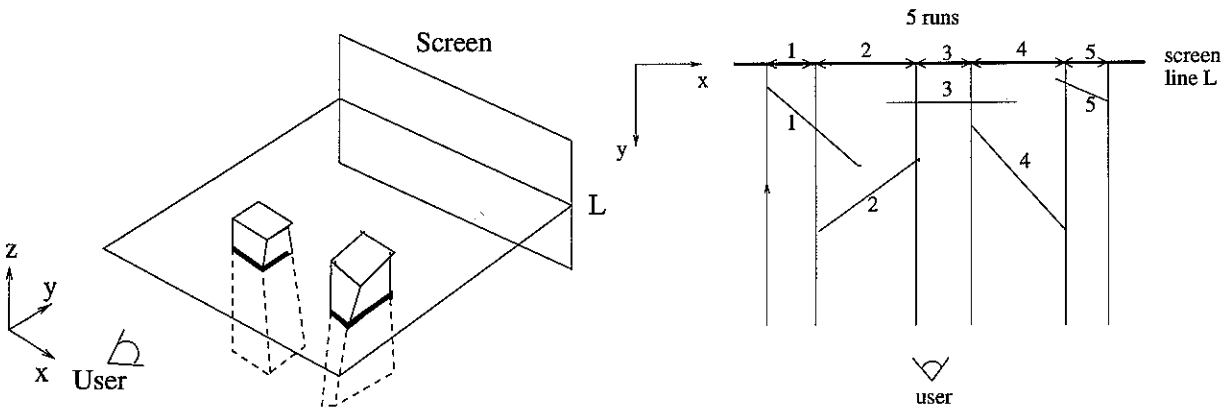


Figure 5: (a) Watkins Algorithm (b) two dimensional solution

We have used this two dimensional algorithm, which is efficient and very easy to implement. Here is its principle. Let S be a set of numbered edges to be displayed. A pixel p of L has a coordinate $(x, 0)$, where x varies from 0 to the L's length. Let K, the line passing through p orthogonal to L. For each L's pixel p, we firstly compute the intersections between K and S. Then, for the intersection point which has the highest y ordinate (e.g. closest to the user), we record its edge's number and y. At the end, we obtain a one-dimensional image L, which contains **runs**. Each run characterizing an edge E, we compute the visible part of E using the first and the last point of the run and their respective attributes, namely $(x_1, 0)$, $(x_2, 0)$, $(number, y_1)$ and $(number, y_2)$, and we obtain $[(x_1, y_1), (x_2, y_2)]$.

We calculate separately two sets of visible edges. The set that contains the edges present at the left side of $\overrightarrow{P_w}$, and those at the right side. This subdivision is realized to take into account the constraint (5) (cf. § 3). For the left side (resp. right), the upper edge (resp. lower) of the box, parallel to $\overrightarrow{P_w}$, symbolizes the screen (cf. figure [4]). The edges are oriented in the same direction than $\overrightarrow{P_W}$. For each side, we group the obtained set of edges in the elementary path which links all the vertices of the edges. This is calculated with the shortest path algorithm in HNG between them [19].

The following images (500x500 pixels, zoom 1) shows the identification process for a set of words.

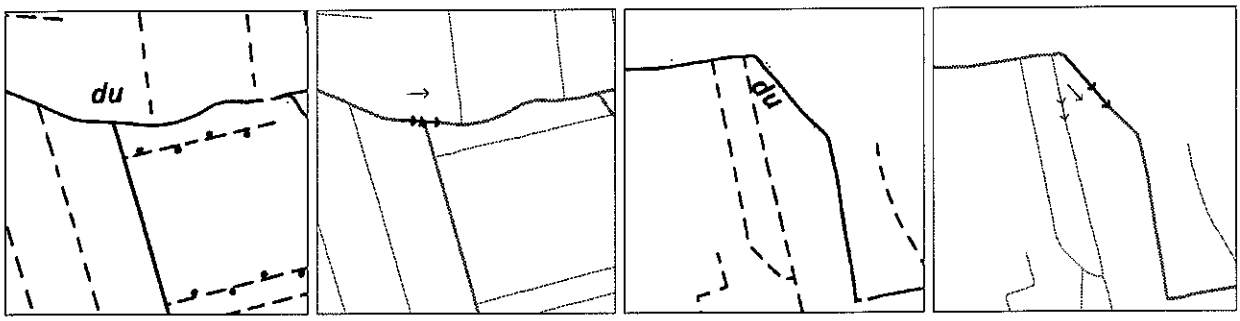


Figure 6: Identification process for a word

In the following, we describe the names construction process. This operation is formalized as MWBM and takes into account the cartographic and syntactic rules between two words.

6 Names construction

Let $W = \{w_0, w_1, ..., w_n\}$ the set of all the words on the map indicating only the hydrographic network. For each word $w_i \in W$, we compute w_i^- and w_i^+, respectively the left and the right hypothesis of w_i, an hypothesis being an oriented elementary path of HNG. We obtain then $W^- = \{(w_0, w_0^-), (w_1, w_1^-), ..., (w_n, w_n^-)\}$ and $W^+ = \{(w_0, w_0^+), (w_1, w_1^+), ..., (w_n, w_n^+)\}$.

Because it is impossible to envisage a link between a word of W^+ and another of W^-, (rule (5) (cf. § 3)), the names construction problem concerns separately those two sets. In the following, we present the reasons why we choose the MWBM formalism, how the corresponding bipartite weighted graph is built, and we finish by recalling the Hungarian algorithm and its application.

6.1 Formalism

Let $V = (W, Z)$ a weighted directed graph, where W is the set of words and Z is the set of the associations between two words of W. Each association is weighted according to syntactic and cartographic rules. Less the weight of an edge (w_1, w_2) is, more likely the transition from w_1 to w_2 is. A toponym being represented as an ordered set of words, the names construction problem is formulated in the following way:

- A word must have only 0 or 1 successor
- A word must have only 0 or 1 predecessor
- The solution set symbolizing the names must have the minimum weight

This problem can be solved using MWBM algorithm [6]. Thus, we construct the weighted bipartite graph and after we implement the resolution problem with the Hungarian Algorithm.

6.2 Bipartite graph construction

This operation is realized in two steps: 1) vertices and edges creation; 2) computing the weight of the edges. Let $W^+ = \{(w_0, w_0^+)..., (w_n, w_n^+)\}$ (or W^-) a set of couple of words and hypothesis. We have $HNG = (X, U)$ where X is the set of vertices and U the set of edges. Let $w_i^+ = \{s_i^0, s_i^1, ..., s_i^n\}$ and $w_j^+ = \{s_j^0, ..., s_j^m\}$, the directed elementary pathes in HNG associated respectively to w_i and w_j, $(s_i^k, s_i^{k+1}) \in U \ \forall k \in 0..n-1$ and $(s_j^k, s_j^{k+1}) \in U \ \forall k \in 0..m-1$.

6.2.1 Vertices and edges creation

We construct a bipartite weighted graph in two steps:

- Create two vertices w_i^{begin} and w_i^{end}, $\forall w_i : (w_i, w_i^+) \in W^+$
- Create the edges (w_i^{begin}, w_j^{end}) $\forall (i,j) \in [0..n]^2$ and weight them.

This is a bipartite graph because its set of vertices is composed with two partitions of vertices: the words labeled *begin* and those labeled *end*.

6.2.2 Edges' weight computation

Let $SHNG = (X', U')$ a subgraph of HNG, where $X' \subset X$ and $U' \subset U$. For two words w_i and w_j, $SHNG$ is constructed in the following way: it does not contains the HNG's edges which are included in w_i^+ or in w_j^+, or which have a hierarchical level different from the hierarchical level of w_i and w_j (cf. figure [7] (b)):

- $X' = X - \{s_i^k : k = 1...n-1\} - \{s_j^k : k = 1...m-1\}$
- $U' = U - \{u \in U : (hlevel(u) \neq hlevel(w_i)) \wedge (hlevel(u) \neq hlevel(w_j))\}$

We call the **distance** $d(w_i, w_j)$ between w_i and w_j, the length of the shortest path [19] linking s_i^n to s_j^0 in $SHNG$. When this path does not exist, the distance is assumed to be infinite.

Another important criterion is the **continuation** between w_i and w_j. It is used to reject as much as possible the transition between two words which have differents hierarchical levels (rule (7) (cf. § 3)). This test is involved if there is a **cross-section vertex** cx in the shortest path which links them in $SHNG$. It simply means that if there is, at the same hierarchical level than $hlevel(w_i)$, a chain of $SHNG$ which starts from cx and which is better aligned with w_i^+ than w_j^+, then the transition between w_i and w_j is impossible. We have the same reasoning for a chain which arrives at cx and which is better aligned with w_j^+ than w_i^+.

The weight wgh_{ij} between two words w_i^{begin} and w_j^{end} is computed according to the syntactic and cartographic rules, as described below.

- $wgh_{ij} = \infty$ if one of those condition is verified:
  - $i = j$
  - The couple (w_i^{begin}, w_j^{end}) is syntactically incorrect
  - They have different police or size
  - The distance between w_i^{begin} and w_j^{end} does not respect the rule (3) (cf. § 3).
  - There is a better continuation starting from w_i or arriving to w_j
- $wgh_{ij} = d(w_i, w_j)$

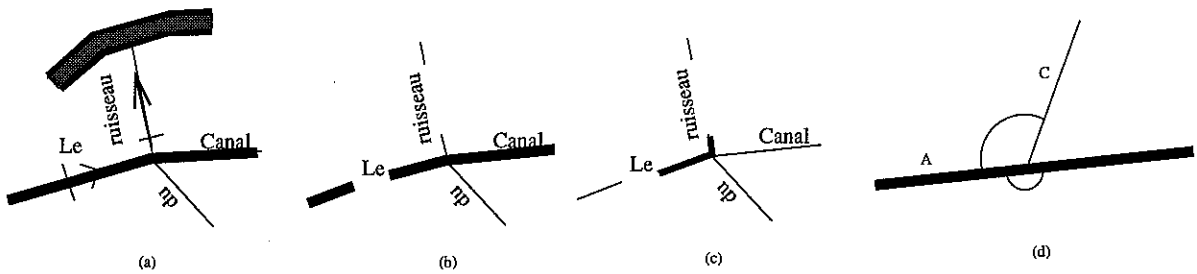


Figure 7: (a) Computation of the transition from "Le" to "Ruisseau" and HNG (b) SHNG considered (c) shortest path (d) continuation

The figure (cf. figure [7]) shows (a) HNG and its three hierarchical levels, and the transition between "Le" and "Ruisseau" is evaluated; (b) shows $SHNG$ concerning those two words; (c) illustrates the shortest path computed; (d) describes the continuation problem; here, the transition between "A" and "C" is rejected because they have differents hierarchical levels and "A" is better aligned with a chain starting from the cross-section vertex, than with "C".

6.3 Hungarian Algorithm and its application

A **matching** of a graph $G = (X, U)$ is defined as a subset $K \subset U$ where they are not two different edges of K which are **adjacents.**

This algorithm was originally used to model the problem of tasks allocation, where there are n machines and n tasks. For each machine i, we know the costs γ_{ij} for processing each task j. If a machine i cannot realize a task j the processing cost is infinite. The problem is to associate to each task a single machine and to have the minimal total processing cost. Then, we look for a permutation $\sigma = \{1, 2, ..., n\}$ having a minimal total cost $\sum_{i=1..n} \gamma_{i,\sigma(i)}$, among all the permutations set. This problem is known to be a perfect minimum weighted matching in a bipartite graph [6]. In our case, we have constructed a bipartite graph (§ 6.2.1) where a word w_i^{begin} represents a task i, w_j^{end} a machine j and where all the tasks are linked to the machines by a weighted edge.

Recalling the formalism described before (cf. § 6.1), we transform the conditions on the name's word in order to apply this algorithm.

- A word must have only 1 predecessor
- A word must have only 1 successor
- the solution set must have a minimum total weight

The two first conditions are equivalent to the fact that there are no adjacents edges in the solution set. At the end the solution set $N = \{\exists(i,j) \in [0..n]^2 : (w_i^{begin}, w_j^{end})\}$ contains n edges, and has a minimum weight. We remove from N, the edges which have an infinite cost (impossible transition), and thus the initial conditions upon the successors and predecessors (0 or 1) (cf. § 6.1) are satisfied.

6.4 Recovering and parsing the Names

Because the cartographic and syntactic constraints are too strong between two words, and because the hydrographic network is very sparse, the probability to have, in a constructed set of words CSW, two consecutive words which belong to different "real" names is very weak. But, a kind of error which may occur is a word omission in CSW, generally due to a bad network reconstruction. In fact, the distance (cf. § 6.2.1) between two words may be infinite, because the network is badly reconstructed.

Thus, our purpose is not look for all the subsets of CSW which are correct. If there is an error, we **alert** the user showing the region in the map where the trouble is. In a future, we will perhaps provide a partially automated network (and so name too) correction. But, when CSW is correct, we store it as attribute on the concerned edges of HNG (w_i^+ or w_i^-, for a word w_i of CSW).

Starting from the minimum weighted set of edges $N = \{\exists(i,j) \in [0..n]^2 : (w_i^{begin}, w_j^{end})\}$, the aim of this algorithm is to find the names contained in N and to test if they are syntactically correct. The first step is realized by computing the connected components in N, and after, for each connected component, by creating an ordered set of words. The second step is achieved using the predefined syntactic parser.

6.4.1 Recovering

The two operations of the recovering can be grouped in only one. Here is the algorithm which realizes it:

- $Names := \emptyset$
- While $N \neq \emptyset$
  - Find a vertex w_i^{begin} which has no predecessor ($\nexists j : (w_j^{begin}, w_i^{end}) \in N$)
  - If such w_i^{begin} does not exists consider the first vertex as w_i^{begin} (cyclic name)
  - $one-name := \{w_i\}$; $w := w_i^{begin}$
  - While $N \neq \emptyset \ \wedge \ \exists j : \ (w, w_j^{end}) \in N$ do

* $N := N - (w, w_j^{end})$
* $one - name := one - name \ \cup \ w_j$
* $w := w_j^{begin}$

- $Names := Names \ \cup \ \{one - name\}$

The variable $Names$ is the set of all the names the algorithm has found in N, while $name$ denotes a single name present in N. There is a special case, denoted as cyclic name, when the first vertex of this ordered set has one predecessor. Now, we have to confirm if a constructed name $NM = \{v_0, v_1, .., v_p\}$ is syntactically correct or not. For this, we consider two differents case, that is the NM is a cyclic set or not.

6.4.2 Parsing NM which is not a cyclic set

Due to the construction process, all the transition (v_i, v_{i+1}) $i \in [0..p]$ are correct. Then, we have three tests to check if NM is correct or not, using the parsing table (cf. § 4):

- $(\emptyset_1, v_0)$ is correct;
- $(v_p, \emptyset_2)$ is correct;
- There is no parenthesis mismatch.

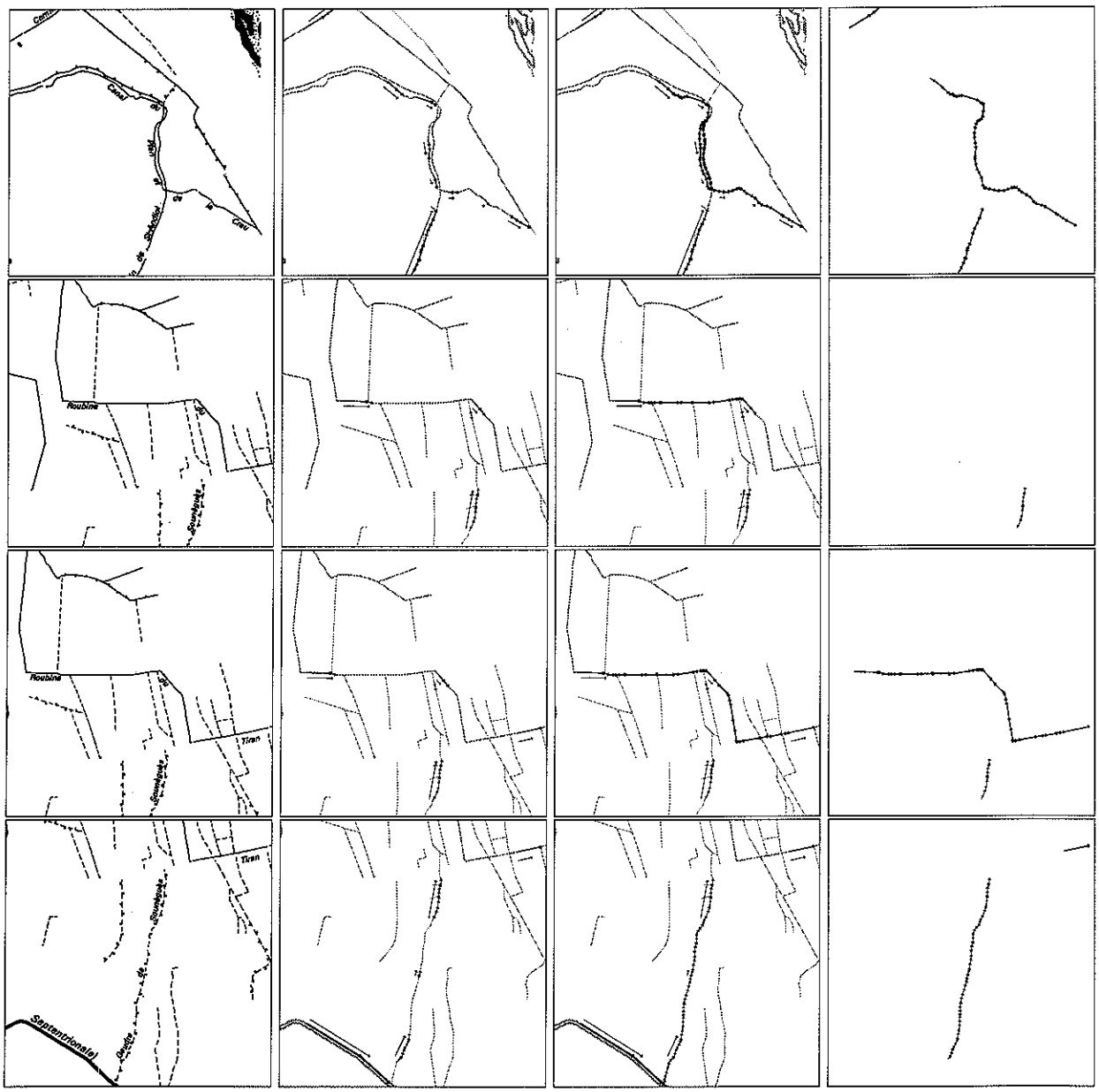
6.4.3 Parsing NM which is a cyclic set

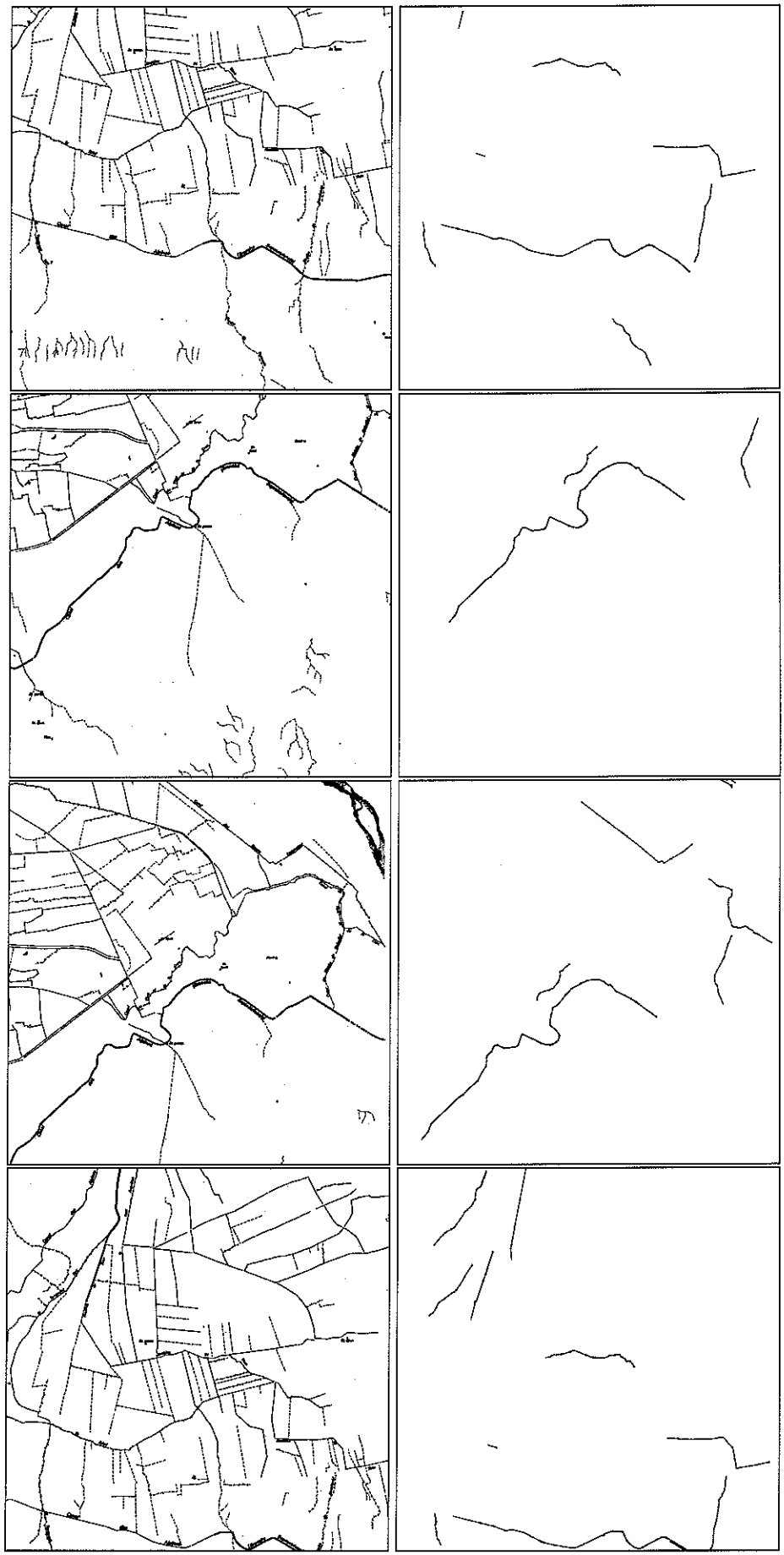
Starting form $NM = \{v_0, .., v_p\}$, we create the set $NM^* = \{\{v_0, ..., v_p\}, \{v_1, ..., v_p, v_0\}, ..., \{v_p, v_0, ..., v_{p-1}\}\}$, and we sort it by crescent costs. Then, we parse all the elements of NM^* until one name is correct. If all the names are incorrect, we alert the user.

7 Results

We present three kinds of results obtained for different pieces of a map (images), a scanned map representing 12800x9000 pixels: 1) images containing 1500x1500 pixels; 2) 5000x5000 pixels; 3) the entire map. The computation times are very small, less then 20 seconds for the entire map. For the first tests, we show from left to right, the four steps of the processing: 1) initial image; 2) the hierarchical graph, principal vectors of the words and their projection; 3) the names hypothesis; 4) network attributed by correct toponyms; Due to the space, we do not show the second and third step for other images.

There was only one error on the entire map, due to a **bad network reconstruction**. This error can be located, and the user will be alerted. In a future, we will perhaps provide an automated network correction.


Roubine
Roubine
Tiran
Tiran
Septentrional



8 Conclusion

We have described a high-level method to compute the association between the toponyms and the hydrographic network represented on the French geographic maps. Maps' scale is 1/25000 and they are scanned at 500 dpi. The hydrographic network, represented with dashed lines, interrupted solid lines and fragmented areas, was already extracted, reconstructed [4] and hierarchically attributed [5], in order to obtain an attributed non directed graph, which is geometrically and topologically correct. In the same way, strings were previously extracted [1], and proved to be syntactically and phonetically correct.

Then, starting with an hydrographic network graph (HNG) and all the extracted words, the purpose was to find the elements of HNG indicated by a name, a name being an ordered set of words. This problem is addressed as a minimum weighting bipartite matching (MWBM) and is implemented with the Hungarian algorithm [6]. The bipartite graph is constructed using, as a priori knowledges, the cartographic and syntactic rules defined for the toponyms.

The process contains three steps: 1) identification of the elements indicated by a word; 2) toponyms-hypothesis creation by grouping several words into an ordered set; 3) validation of the correctness of the toponyms, and network attribution. The main step is the second one addressed as a minimum weighted bipartite graph matching. Experiments were performed on several maps. The principal causes of failure is a bad network reconstruction. Thus, computing the association of toponyms to the network may provide an error's reconstruction identification process for the network validation.

References

[1] M. P. Desseilligny, *Lecture Automatique de Cartes.* PhD thesis, Université René Descartes, Centre Universitaire des Saints-Pères, UFR de Mathématiques et Informatique, Paris V, Oct. 1994.

[2] A. Khotanzad and E. Zink, "Color Paper Map Segmentation Using Eigenvector Line Fitting," in *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation, San Antonio, TEXAS (USA)*, pp. 190–194, Apr. 1996.

[3] L. Lefrère, *Contribution au Développement d'Outils pour l'Analyse Automatique de Documents Cartographiques.* PhD thesis, Université de Rouen, Rouen, Oct. 1993.

[4] R. Mariani, F. Lecourt, M. P. Deseilligny, J. Labiche, and Y. Lecourtier, "Interprétation de Cartes Géographiques: Algorithmes de Reconstruction des Réseaux Hydrographiques et Routiers," in *Actes 10ème Congrès AFCET Reconnaissance des Formes et Intelligence Artificielle, Rennes*, pp. 34–42, Jan. 1996.

[5] R. Mariani, M. P. Deseilligny, J. Labiche, and Y. Lecourtier, "Geographic Map Understanding. Attributes Computation for Hydrographic Network," in *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation, San Antonio, TEXAS (USA)*, pp. 252–257, Apr. 1996.

[6] M. Gondran and M. Minoux, *Graphes et Algorithmes*, pp. 157–158. Vol. 1 of de la Direction des Etudes et Recherches d'Electricité de France [20], 1985.

[7] J. Ogier, *Contribution à l'Analyse Automatique de Documents Cartographiques: Interpétation de Données Cadastrales.* PhD thesis, Université de Rouen, Rouen, Jan. 1994.

[8] L.Boatto and al., "An Interpretation System for Land Register Map," *IEEE Computer Society Press*, vol. 25, pp. 25–33, July 1992.

[9] J. D. Hartog, T. T. Kate, and J. Gerbrands, "Knowledge-Based Segmentation for Automatic Map Interpretation," in *Proceedings of IAPR International Workshop on Graphics Recognition, Penn State Scanticon (USA)*, pp. 71–80, Aug. 1995.

[10] C. Lai and R. Kasturi, "Detection of Dimension Sets in Engineering Drawings," *IEEE Transactions on PAMI*, pp. 848–855, Aug. 1994.

[11] D. Dori, Y. Velkovitch, and L. Wenyin, "Object-Process Based Segmentation and Recognition of ANSI and ISO Standard Dimensioning Texts," in *Lecture Notes in Computer Sciences*, vol. 1072, pp. 212–232, 1996.

[12] M. Kim, M. Park, O. Kwon, and Y. Kwon, "Automatic Region Labeling of the Layered Map," in *Lecture Notes in Computer Sciences*, vol. 1072, pp. 179–189, 1996.

[13] G. Myers, P. Mulgaonkar, C. Chen, J. DeCurtins, and E. Chen, "Verification-Based Approach for Automated Text and Feature Extraction From Raster-Scanned Maps," in *Proceedings of IAPR International Workshop on Graphics Recognition, Penn State Scanticon (USA)*, pp. 90–99, Aug. 1995.

[14] M. Gondran and M. Minoux, *Graphes et Algorithmes*, pp. 1–33. Vol. 1 of de la Direction des Etudes et Recherches d'Electricité de France [20], 1985.

[15] M. P. Deseilligny, H. L. Men, and G. Stamon, "Lecture Automatique des Ecritures sur Cartes Scannées," *Revue de Traitement du Signal - Spécial CNED'96*, vol. 12, no. 6, 1995.

[16] M. Coster and J. Chermant, *Précis d'Analyse d'Images*, pp. 82–86. Paris: Presses du CNRS, 1989.

[17] R. Mariani, F. Lecourt, M. P. Deseilligny, J. Labiche, and Y. Lecourtier, "Interprétation de Cartes Géographiques: Algorithmes de Reconstruction des Réseaux Hydrographiques et Routiers," *Revue de Traitement du Signal - Spécial RFIA'96 - A paraitre en Décembre 1996*, vol. 12, no. 1, 1996.

[18] Y. Gardan, *Elements de CAO. Matériels et Logiciels de Base*, vol. 1, pp. 179–197. Paris, Londres, Lausanne: Hermès, 1988.

[19] M. Gondran and M. Minoux, *Graphes et Algorithmes*, pp. 35–68. Vol. 1 of de la Direction des Etudes et Recherches d'Electricité de France [20], 1985.

[20] M. Gondran and M. Minoux, *Graphes et Algorithmes.* Collection de la Direction des Etudes et Recherches d'Electricité de France, 1985.

RFIA '96

Interprétation de cartes géographiques. Algorithmes de reconstruction des réseaux hydrographiques et routiers.

R. MARIANI*. F. LECOURT *. M.P. DESEILLIGNY*. J. LABICHE+. Y. LECOURTIER**.
*IGN/DT/SR , Laboratoire MATIS. 2 Av. Pasteur, 94160 Saint-Mandé.
Email: {mariani, lecourt, pierrot}@ign.fr.
fax: (16) 43.98.85.81
+LACP, Université de Caen. Faculté des sciences. 14000 Caen.
**La3i-Lacis. Université de Rouen.
UFR des sciences et techniques 76821 Mont-Saint-Aignan

Résumé- Cet article décrit une méthode générale de haut niveau pour la reconstruction automatique des graphes des réseaux représentés sur les cartes géographiques. Elle a été appliquée aux réseaux hydrographiques et routiers qui sont essentiellement composés de lignes tiretées, de traits pleins interrompus et d'objets surfaciques interrompus. L'objectif de cette méthode est d'obtenir, à partir d'un graphe de réseau interrompu (obtenu par vectorisation) et à l'aide de contraintes de reconstruction, un graphe géométriquement et topologiquement correct. Le processus de reconstruction est décomposé en quatre étapes successives: 1) reconnaissance et extraction des autres thèmes, puis vectorisation de l'image restante; 2) transformation des lignes tiretées en traits pleins; 3) connexion des traits pleins entre eux; 4) connexion des traits pleins aux surfaces. Cet article insiste sur les étapes 2) et 3), modélisées comme un problème d'arbre recouvrant de poids minimum, où les connexions à ajouter au graphe sont pondérées par des fonctions de poids spécifiques à chaque type de tracé. L'étape de connexion des traits pleins entre eux se termine par une fermeture de cycles qui utilise une propriété de connexité locale. Afin de maintenir la planarité du graphe reconstruit, nous utilisons une méthode de visibilité qui permet de réfuter les arêtes qui créent des intersections.

Mots-clés-- interprétation de cartes géographiques; analyse d'image; modélisation; arbre recouvrant de poids minimum; reconstruction de réseau; découpage polygonal; théorie des graphes;

1 Introduction

Cet article décrit une méthode générale de haut niveau pour la reconstruction automatique des graphes de réseaux représentés sur les cartes géographiques. Nous l'avons appliquée aux réseaux hydrographiques et routiers. Ce travail, mené à l'Institut Géographique National (IGN), s'inscrit dans le cadre d'un projet de lecture automatique de cartes géographiques [3] dont l'objectif est d'alimenter les Systèmes d'Informations Géographiques (SIG) en données de haut niveau. Ces cartes sont à l'échelle 1/25000 et sont scannées à 50 microns. Les cartes sont imprimées à partir de quatre planches correspondant chacune à une couleur d'impression de la carte: la planche des rouges, des bleus, des verts et des noirs. L'IGN étant un producteur de cartes, nous travaillons directement sur ces planches et évitons ainsi le problème de séparation des couleurs qui apparaît lors du scannage en RVB (rouge vert bleu) d'une carte papier. D'importants travaux ont débutés sur la planche des noirs [1][3], et nous travaillons actuellement sur les quatre planches. Cet article s'intéresse à la planche des bleus représentant le thème hydrographique et à la planche des rouges représentant une partie du réseau routier. L'objectif est de construire, pour chaque réseau, un graphe géométriquement et topologiquement correct.

Nous exposons d'abord les caractéristiques des thèmes hydrographiques et routiers. Ensuite, nous décrivons le processus général de reconstruction des réseaux et le formalisme utilisé. Puis, nous détaillons son application aux différentes planches et les résultats obtenus. Enfin, nous présentons les conclusions et les perspectives de nos travaux.

2 Caractéristiques des thèmes hydrographiques et routiers

Pour chaque réseau, nous distinguons deux types de connaissances: celles relatives au réseau réel, et celles relatives à sa représentation graphique. Dans ce qui suit, nous énonçons les règles qui régissent les propriétés générales des réseaux, les exceptions à ces règles et les conséquences pour la reconstruction.

2.1 Propriétés des réseaux réels

Le graphe du réseau hydrographique réel contient peu de composantes connexes et très peu de cycles (autant que d'îles). En fait, il peut être assimilé à une forêt, c'est à dire un ensemble d'arbres. Il y a peu de composantes connexes par rapport à la taille du réseau et donc le nombre d'arbres est limité.

Le graphe du réseau routier est un graphe connexe (il y a très peu d'exceptions) et il contient à grande échelle un nombre important de cycles.

2.2 Exceptions

Il y a deux exceptions fréquentes à ces règles:

- La présence des canaux, dans le réseau hydrographique, entraîne une apparition fréquente de cycles.
- A cause du fenêtrage lié à la carte papier, nous n'étudions qu'une partie du réseau réel, celle limitée par le rectangle (ou fenêtre) de la carte. Le fenêtrage crée des composantes connexes (et ouvre des cycles).

Ces exceptions affaiblissent les règles topologiques fortes sur les graphes des réseaux réels. Les graphes des réseaux cartographiques contiennent donc peu de composantes connexes et peu de cycles.

2.3 Caractéristiques graphiques des réseaux

La représentation graphique du réseau hydrographique est structurée en trois classes: les lignes tiretées, les traits pleins et les objets surfaciques texturés. Pour chaque classe, il existe des règles sémantiques et graphiques

différentes. Ainsi, les lignes tiretées représentent des cours d'eau temporaires, des canaux souterrains, des aqueducs souterrains, des limites de zones etc., les traits pleins représentent des cours d'eau bordés d'arbres, des aqueducs au sol, des canaux d'alimentation de faible largeur, et les objets surfaciques texturés symbolisent les fleuves, les rivières, les lacs.

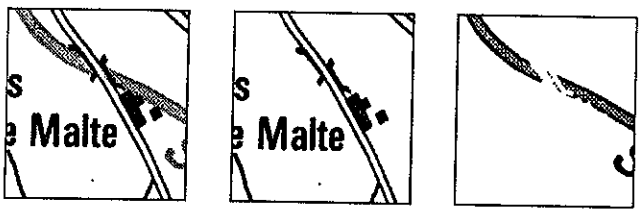
Pour le réseau routier, il existe deux textures: l'aplat pour les routes principales et le tramé pour les routes secondaires. Pour chacune d'elles, existe deux modes de représentation: le trait plein et la ligne tiretée, qui caractérisent le degré de viabilité d'une route. Ceci définit donc quatre types d'objets.

2.4 Conséquences pour la reconstruction

Les objets cartographiques étant fréquemment interrompus (lignes tiretées, occlusions par d'autres objets ...), une des étapes importantes de la tâche de reconstruction sera, après vectorisation, de rajouter les arêtes vraisemblables, ceci afin d'obtenir des réseaux cohérents vis-à-vis des connaissances topologiques a priori.

Les règles cartographiques utilisées pour représenter les lignes tiretées, les traits pleins et les objets surfaciques sont différentes. Afin d'exploiter ces connaissances propres à chaque thème, nous avons décomposé le processus de reconstruction du graphe d'un réseau en quatre étapes: 1) transformer les lignes tiretées en traits pleins; 2) reconnecter les traits pleins entre eux; 3) reconstruire les objets surfaciques fragmentés; 4) connecter les traits pleins aux objets surfaciques.

La transformation des lignes tiretées en traits pleins est réalisée en utilisant des règles intra-planches. La connexion des traits pleins entre eux utilise des règles inter-planches: nous devons, par exemple, prendre en compte les règles de superposition d'objets appartenant à des planches différentes et qui créent des interruptions de tracé des traits pleins. Ainsi, comme le montre les images 1a), 1b) et 1c), lorsqu'une route, dessinée sur la planche des noirs, passe au-dessus d'une rivière, le trait plein représentant la rivière sur la planche des bleus est interrompu à leur intersection.



Images 1a) Carte initiale; 1b) Planche des noirs; 1c) Planche des bleus. (500x500 pixels, grossissement de 1).

Dans cet article, nous décrivons les étapes 1), 2) et 3) du processus de reconstruction du réseau, (l'étape 4) est en cours de réalisation). Les étapes 1) et 2) sont communes aux deux réseaux. Les étapes 3) et 4) sont, ici, spécifiques au réseau hydrographique.

Cette analyse menée sur les réseaux réels et sur leur représentation graphique, nous a permis d'obtenir une décomposition du processus général de reconstruction. Nous présentons ce dernier dans la section suivante.

3 Description du processus

Le processus général (cf. figure 1) se décompose en deux grandes étapes: 1) prétraitements de l'image initiale; 2) reconstruction géométrique et topologique du réseau.

La phase des prétraitements consiste principalement à reconnaître et à supprimer les objets qui ne font pas partie du réseau linéaire (réseau formé des tirets et des traits pleins), à vectoriser l'image restante et à séparer les traits pleins des tirets.

La reconstruction géométrique et topologique du réseau linéaire s'effectue d'abord par la transformation des lignes tiretées en traits pleins puis par la reconnexion des traits pleins entre eux. Parallèlement, nous effectuons la reconstruction des objets surfaciques fragmentés. La reconstruction du réseau complet se termine par la connexion des traits pleins aux objets surfaciques.

Schéma général.

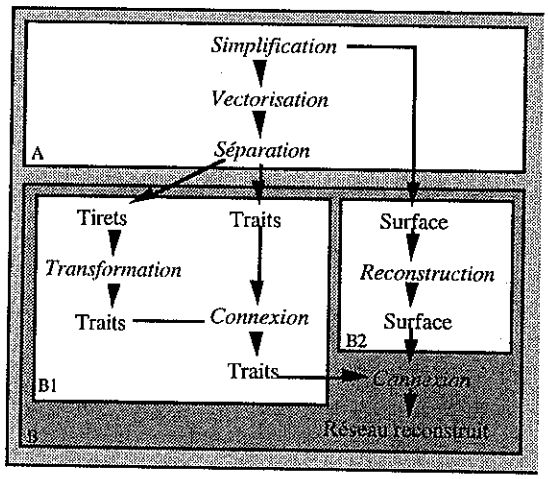


Figure 1: processus général de reconstruction de réseau.

La partie A correspond à la phase des prétraitements de l'image initiale. La partie B correspond à la reconstruction du réseau. Celle-ci comporte la phase de reconstruction du réseau linéaire (B1) et celle de reconstruction des surfaces (B2).

Nous décrivons dans les sections suivantes les éléments qui constituent ce processus, en commençant par la phase des prétraitements de l'image initiale.

4 Prétraitements de l'image initiale

Cette étape se décompose en deux parties: 1) simplification de l'image initiale; 2) vectorisation de l'image résultante par squelettisation et approximation polygonale [3];

4.1 Simplification de l'image

La simplification de l'image consiste, d'une part à reconnaître et à supprimer les objets qui n'intéressent pas la reconstruction du réseau, d'autre part à obtenir une représentation standardisée des objets intéressants (par exemple, la transformation des objets tramés du réseau routier en objets aplats pour la vectorisation).

Les objets préalablement extraits lors de cette phase sont conservés et seront plus tard utilisés pour l'interprétation globale de la carte.

Nous décrivons brièvement cette étape pour les deux planches considérées.

4.1.1 Simplification de la planche des bleus

Le but de cette opération est de supprimer tous les objets qui ne font pas partie du réseau linéaire. L'image résultante ne sera composée que de tirets et de traits pleins (Image 2c).

Les objets qui perturbent la reconstruction du réseau hydrographique sont les suivants: les caractères (surtout ceux connectés au réseau), les disques connectés aux branches du réseau représentant les arbres qui bordent les ruisseaux, et les objets surfaciques texturés symbolisant les lacs, les rivières, les fleuves etc...

L'extraction des chaînes de caractères, connectées ou non, est décrite dans [4].

L'extraction des objets surfaciques texturés commence par la création d'une image en quatre couleurs par l'application d'une table de conversion sur l'image des niveaux de gris. Nous obtenons ainsi, le fond, les contours et la texture, cette dernière étant un assemblage de ces quatre couleurs. Une particularité de cette image est que, contrairement au fond et aux contours, la texture est composée de petites composantes connexes (4-connexité), une composante connexe étant formée de pixels voisins de même couleur. Pour finir, nous attribuons à l'ensemble de ces petites composantes connexes une couleur unique. Cette opération bas niveau est suffisante pour la simplification de la carte (image 2b).

L'extraction des disques connectés est effectuée en deux étapes. Tout d'abord, le centre du disque est calculé par optimisation d'une fonction de corrélation entre le gradient théorique d'un disque connecté, et le gradient observé de l'élément traité. Puis, nous analysons le gradient observé afin de rejeter le disque hypothétique ou de déterminer les points d'attache du disque au réseau. Une fois ces points d'attache déterminés, nous extrayons le disque (image 2a).

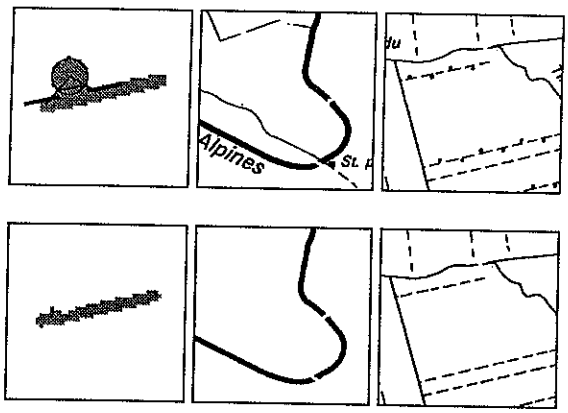


Image 2a Image 2b Image 2c

Images 2a) Extraction des disques connectés (500x500 pixels, grossissement de 10); 2b) Extraction des objets texturés (500x500 pixels, grossissement de 1); 2c) Image du réseau linéaire avant et après simplification (500x500 pixels, grossissement de 1).

4.1.2 Simplification de la planche des rouges

Afin de pouvoir squelettiser toute l'image, nous transformons la texture en aplat, à l'aide d'outils de la morphologie mathématique [19] (ouverture et fermeture).

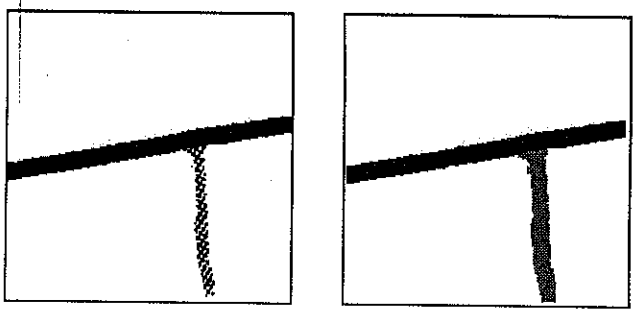


Image 3: simplification de la planche des rouges. (400x400 pixels, grossissement de 3)

4.2 Vectorisation

La vectorisation repose sur une méthode de squelettisation et d'approximation polygonale déjà décrite dans [3]. A l'issue de cette étape, nous obtenons un graphe. L'essentiel des étapes suivantes sera de reconstruire un graphe connexe correct d'un point de vue géométrique et topologique.

5 Positionnement par rapport aux autres travaux

Puisque nous travaillons avec une représentation vecteur du réseau linéaire, nous n'utilisons pas les méthodes qui travaillent directement sur l'image, quelle soit en niveaux de gris ou binaire, afin de détecter, par exemple, les lignes tiretées. Dans cette catégorie de méthodes, nous pouvons citer la transformation de Hough [7][8], l'appariement de lignes droites, d'arcs de cercle et de splines [8], et enfin celles qui utilisent des opérateurs directionnels de morphologie mathématique [9][10].

D'autres travaux concernant l'extraction des textures linéaires ont été proposés [5][6]. Ils offrent un cadre général de travail pour la détection de lignes présentes dans les documents, sans connaissances a priori.

Notre démarche se rapproche de celles qui procèdent par simplification de la carte pour faciliter l'extraction de certains thèmes [11], et qui utilisent des connaissances a priori sur les objets de la carte afin d'en améliorer la vectorisation [12][13][17]. Pour le réseau linéaire, nous manipulons un graphe qui détient des propriétés topologiques utilisées directement comme connaissances a priori dans le processus de reconstruction.

6 Reconstruction du réseau linéaire

Le processus de reconstruction du réseau linéaire est divisé en trois étapes: 1) séparation des tirets et des traits pleins; 2) transformation des lignes tiretées en traits pleins; 3) reconnexion des traits pleins entre eux. Après la description de l'algorithme de séparation, nous donnons pour les deux dernières étapes, les connaissances cartographiques spécifiques au tracé de chaque thème (tirets et traits pleins), le formalisme utilisé et les algorithmes développés.

6.1 Séparation

Cette étape est justifiée par les raisons suivantes: 1) les tirets et les traits pleins constituent deux thèmes cartographiques différents et dans un contexte d'interprétation de carte géographique, il faut les détecter; 2) cette séparation permet de simplifier le processus et

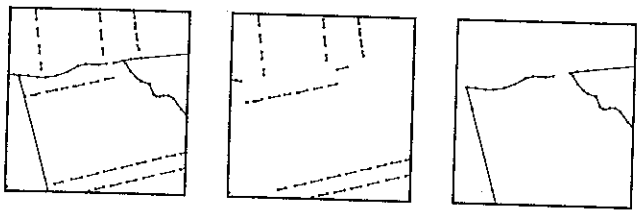
d'utiliser les règles cartographiques, propres à chacun des thèmes, pour leur reconnexion.

Nous considérons ici, l'image simplifiée (cf. Image 4a). Elle est composée exclusivement de tirets et de traits pleins. Cette image a été vectorisée par squelettisation et approximation polygonale. Nous obtenons ainsi un graphe non orienté G=(X,U) où X est l'ensemble des sommets et U l'ensemble des arêtes.

Formellement, nous définissons un tiret comme une chaîne, au sens de la théorie des graphes [2], reliant deux sommets de degré différent de 2 et de longueur totale inférieure à un seuil donné s.

En appliquant ce prédicat à chaque chaîne reliant deux sommets de degré différent de 2 du graphe initial, nous obtenons le sous graphe des tirets et le sous graphe des traits pleins (image 4b et 4c). On dit que G'=(X',U') est un sous graphe de G=(X,U) si et seulement si $X' \subset X$ et $U' \subset U$.

Afin de n'oublier aucun tiret, le seuil s est assez grand, et nous sommes de fait confrontés à une sur-détection. Les fausses détections, dues à des traits pleins de petite longueur, sont en nombre très limité (moins de 1%), et ne sont pas gênantes pour la raison suivante: elles ne respecteront pas les règles cartographiques propres aux tracé des tirets, or la transformation des tirets en traits pleins est, comme nous le verrons, très stricte quant à l'application de ces règles.



Images 4a) graphe initial, 4b) graphe des tirets, 4c) graphe des traits pleins. (500x500 pixels, grossissement de 1).

Par la suite, nous décrivons les formalismes et les algorithmes utilisés pour la reconstruction du réseau linéaire, à partir de ces deux graphes.

6.2 Transformation des lignes tiretées en traits pleins

6.2.1 Formalisme

La transformation des lignes tiretées en traits pleins est modélisée comme un problème d'arbre recouvrant de poids minimum (ARPM) d'un graphe non orienté [2]. L'algorithme traite ce problème en tenant compte de la contrainte suivante, qui permet de conserver la planarité du graphe:

Contrainte de reconstruction.
Une arête n'est rajoutée dans le graphe que si elle ne coupe aucune arête de celui-ci.

Puisque nous utilisons l'ARPM, nous définissons une fonction de poids que nous utilisons pour pondérer les arêtes à rajouter aux graphes. Celle-ci a été élaborée en fonction des connaissances spécifiques précisées ci-dessous.

6.2.2 Connaissances spécifiques sur les lignes tiretées

De façon générale, nous distinguons deux règles cartographiques concernant les lignes tiretées:

- L'espace entre deux tirets ne varie pas beaucoup.
- Il n'y a pas de grande courbure entre deux tirets successifs.

Par défaut, le réseau est rectiligne, donc la courbure est une information importante. Aussi, pour éviter toute ambiguïté de lecture, le cartographe représentera très souvent de façon explicite, les courbures présentes dans le réseau réel. Ainsi, pour connecter deux tirets, deux paramètres sont importants: la distance séparant les deux tirets, et leur tangence. Seules sont spécifiques, pour chaque réseau, les intervalles de valeur utilisés pour ces deux paramètres

6.2.3 Fonction de poids

Soient B et C deux sommets de degré 1 du graphe, A (resp. D) l'unique voisin de B (resp. C) et α (resp. β) l'angle entre le vecteur BC et CD (resp. BC et AB).

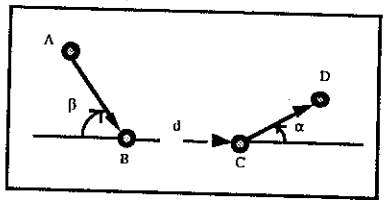


Figure 2: paramètres de la fonction de poids.

La valeur de la fonction est petite si les deux critères suivants sont vérifiés: 1) la distance entre B et C est petite. 2) α et β sont proches de 0.

Nous écrivons w(B,C) la fonction de poids de BC. Soit d_{BC} la distance euclidienne entre B et C. Nous avons choisi:

$$w(B,C) = d_{BC} \times \left(1 + \mu\left((1-\cos\alpha) + (1-\cos\beta)\right)\right)$$
$$\text{avec} \quad \mu > 0 \quad \text{et } |\alpha| + |\beta| < \delta.$$

où δ est une constante permettant de rejeter les connexions dont les angles α et β n'appartiennent pas à l'ensemble $\{(\alpha,\beta) \ / \ |\alpha| + |\beta| < \delta\}$, et μ une constante utilisée pour pondérer le critère de tangence par rapport au critère de distance.

6.2.4 Algorithme

Nous appelons une *connexion possible*, une arête pondérée entre deux sommets de degré 1 du graphe, et qui ne coupe aucune arête du graphe.

Dans un premier temps, nous calculons la liste de toutes les connexions possibles. Ensuite, nous utilisons l'algorithme de l'ARPM en considérant les contraintes précitées. Ceci est réalisé de façon itérative, en sélectionnant la meilleure connexion possible, en l'ajoutant au graphe et en supprimant de la liste, toutes celles qui, désormais, ne respectent plus ces contraintes.

- Sélectionner la meilleure connexion possible consiste à choisir celle qui a le poids le plus faible.
- Supprimer les connexions possibles qui ne respectent pas la topologie, se fait en éliminant celles qui coupent la dernière arête ajoutée au graphe.

A l'issue de cette étape, les lignes tiretées ont été transformées en traits pleins. Ces derniers sont rajoutés au graphe des traits pleins initiaux (cf. Image 5c). Nous allons maintenant reconnecter les traits pleins entre eux.

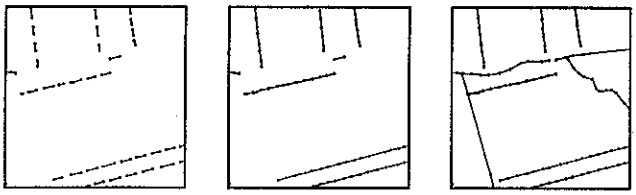


Image 5: a) images des tirets, b) des tirets reconnectés et c) des traits pleins. (500x500 pixels, grossissement de 1).

6.3 Connexions des traits pleins entre eux

Nous présentons tout d'abord le formalisme utilisé, puis les connaissances spécifiques au tracé des traits pleins. Enfin, nous donnons les algorithmes utilisés qui permettent de reconstruire le réseau linéaire des traits pleins.

6.3.1 Formalisme

Ici encore, nous utilisons une variante de l'ARPM dont les noeuds de l'arbre sont les composantes connexes du graphe des traits. Nous lui associons une fonction de poids adaptée à chaque planche et élaborée grâce aux règles cartographiques spécifiques aux traits pleins. La connexion des traits pleins se termine par une phase de fermeture des cycles.

Suite aux exceptions précisées au paragraphe 2.2, nous ne pouvons pas utiliser directement la propriété de connexité globale des deux réseaux. Aussi, nous lui substituons une propriété, dite de connexité locale, définie de la façon suivante: " Si la distance géométrique entre deux points du réseau est petite, la longueur du plus court chemin entre ces deux points ne doit pas être grande ". Cette propriété est utilisée lors de la fermeture des cycles, car concrètement elle signifie que si un cycle est presque totalement représenté, il doit être fermé.

Nous appelons *une extrémité* un sommet de degré un du graphe et *une arête source* une arête ayant au moins une extrémité. Une connexion possible est calculée à partir d'une arête source; elle vérifie la contrainte de reconstruction (§ 6.2.1) et lie un sommet à un arc.

Le problème lié à la topologie, c'est à dire celui de la conservation de la planarité du graphe, est traité au fur et à mesure de la reconstruction grâce à un algorithme de découpage polygonal [18], dont la tâche est d'en préserver la cohérence. Cet algorithme sera précisé par la suite.

6.3.2 Connaissances spécifiques sur les traits pleins

Les interruptions des traits pleins sont dues à la présence, au même endroit, d'autres objets de la carte. Ces objets peuvent appartenir à la même planche ou bien à d'autres planches. Par exemple, une chaîne de caractères peut partiellement recouvrir une rivière. Les interruptions ont une taille déterminée par les objets qui créent les occlusions.

Contrairement aux tirets, où la courbure est une information très importante pour la reconnection, ici elle n'a pas beaucoup d'importance pour la raison suivante:

- Deux traits pleins qui doivent être connectés, peuvent ne pas être alignés. Il est même fréquent qu'ils soient presque perpendiculaires.

Ainsi, une connexion entre deux traits pleins constitue souvent le prolongement d'un de ces deux traits.

6.3.3 Fonction de poids

Nous voulons une fonction qui privilégie les connexions par prolongement de traits pleins. Les traits pleins étant souvent rectilignes, il est en effet plus naturel d'utiliser le prolongement plutôt que la distance la plus courte (cf. figure 3).

Soit a, un point situé sur une arête candidate à la connexion, d_{Ba} la distance euclidienne entre B et a. Soit α l'angle que fait le vecteur Ba avec la droite AB.

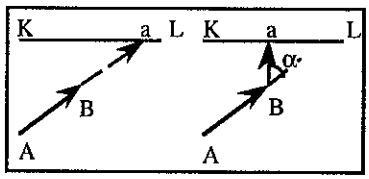


Figure 3: préférence pour le prolongement a) plutôt que pour la distance la plus courte b).

Nous avons W_a la fonction de poids associée au point a:

$$\boxed{W_a = d_{Ba}^2 \cdot (1 + \tau . \tan|\alpha|) \text{ avec } \alpha \in \left[-\frac{\pi}{2};\frac{\pi}{2}\right] \text{ et } \tau > 0}$$

La constante τ permet de pondérer le critère de tangence par rapport au critère de distance.

Soit KL l'arête candidate à la connexion. L'algorithme calcule le point $a = a(\lambda_o)$ de KL tel que:

$$\boxed{W_{a(\lambda_o)} = \min_{\lambda=0..1} \{W_{a(\lambda)} / a(\lambda) = \lambda.K + (1-\lambda).L\}}$$

avec un pas de variation de λ égal à 1/KL.

6.3.4 Algorithmes

La reconnexion des traits pleins entre eux se déroule en deux étapes: 1) reconstruction de l'ARPM du graphe; 2) fermeture des cycles.

Au fur et à mesure de la reconstruction, l'algorithme considérera qu'une connexion à rajouter est valide si elle respecte la contrainte topologique de reconstruction définie précédemment (§ 6.2.1). De plus, les connexions qui fermeront les cycles ne devront pas vérifier pas la propriété de connexité locale.

Dans un premier temps, nous présentons une méthode qui préserve la topologie, en utilisant un découpage polygonal [18]. Ensuite, nous donnons une définition formelle de la connexité locale. Nous terminons par la présentation de l'algorithme d'ARPM spécifique aux traits pleins, et par celui de la fermeture des cycles.

Algorithme de visibilité pour une préservation incrémentale de la topologie.

L'algorithme utilisé pour sélectionner les connexions possibles partant d'une arête, appelée *arête source* , procède en deux étapes (cf. figure 4): 1) présélection rapide des arêtes géométriquement voisines de l'arête source; 2) calcul précis des connexions envisagées tout en respectant la topologie. Ceci est réalisé à l'aide d'un algorithme de visibilité (découpage polygonal): nous ne conservons que les morceaux d'arêtes visibles de B en regardant dans la direction du vecteur AB. Le domaine de visibilité de B est caractérisé par une demi-ellipse, orientée selon la direction de AB et centrée en B. Les longueurs des deux rayons

principaux de l'ellipse sont les longueurs maximales de la connexion autorisée dans la direction de l'arête source et dans une direction perpendiculaire à celle-ci.

Pour chaque arête source AB, nous calculons le découpage polygonal: pour chaque arête [JK] présente dans la demi-ellipse, nous considérons la surface qu'elle dissimule (cf. figure 4 (IJKL)), et nous découpons les autres arêtes avec cette surface. Tout ce qui est présent dans cette zone (IJKL) est invisible pour l'extrémité B de l'arête source. A la fin, nous ne conservons que les morceaux d'arêtes visibles.

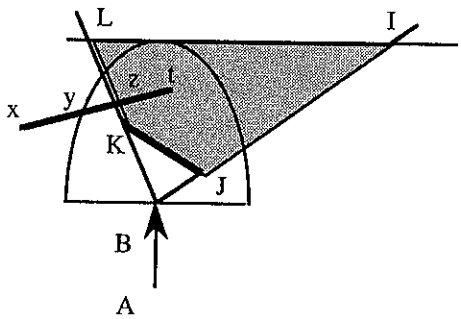


Figure 4: découpage elliptique et polygonal.

Le vecteur AB représente une arête source dont l'extrémité est le point B. [xt] est d'abord découpé par l'ellipse. Seulement [yt] est conservé, [xy] étant à l'extérieur de l'ellipse. Ensuite, [yt] est découpé par le polygone (IJKL), (représentant la zone cachée par [JK], invisible pour B) et nous ne conservons qu'[yz]. En effet, il est impossible de connecter B à [zt], car il faudrait passer au dessus de [JK] (violation de topologie). Or ce graphe du réseau linéaire est planaire.

Algorithme de l'arbre recouvrant de poids minimum.

Le graphe non orienté est modélisé par une liste de composantes connexes numérotées. Une composante connexe est constituée d'une liste d'arêtes; chaque arête, obtenue par vectorisation, est caractérisée par deux sommets.

Nous calculons toutes les connexions possibles partant des arêtes sources du graphe, puis nous les filtrons par l'algorithme de l'ARPM en considérant la contrainte de reconstruction. Le calcul de l'ARPM s'effectue de façon itérative et en quatre étapes: 1) choisir la meilleure connexion possible entre deux composantes connexes et l'ajouter au graphe; 2) fusionner les deux composantes connexes; 3) détruire, parmi la liste des connexions possibles, celles qui deviennent invalides; 4) calculer les connexions possibles en prenant en compte la dernière arête ajoutée.

- La meilleure connexion possible est celle qui lie deux composantes connexes différentes et qui a le poids le plus faible.
- Après la fusion de deux composantes connexes en une seule, ce résultat est propagé dans la liste des connexions possibles.
- Une connexion possible devient invalide si elle coupe la dernière arête ajoutée, ou bien si aucun de ses deux sommets n'est une extrémité. En effet, l'ajout d'une connexion au graphe détruit une ou deux extrémités.
- Ajouter à l'ensemble des connexions possibles, celles qui partent des arêtes sources restantes et qui arrivent à cette dernière arête ajoutée.

A la fin de cette étape, ne reste dans la liste des connexions possibles que celles qui créent des cycles.

Propriété de connexité locale.

Nous rappelons la propriété de connexité locale: "Si la distance géométrique entre deux points est faible, la longueur du plus court chemin entre ces deux points à travers le réseau ne doit pas être très grande". La figure 5 donne trois exemples pour illustrer cette propriété.

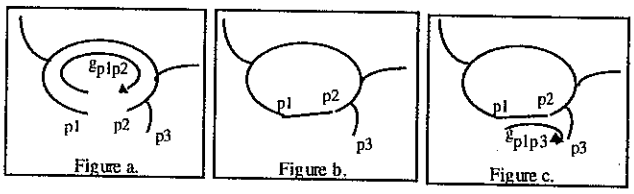


Figure 5: La connexité locale et son application.

Dans la figure a), les couples (p1 p2) et (p1 p3) ne vérifient pas la connexité locale. Ces deux connexions sont donc permises. Nous avons choisi de connecter p1 et p2 car c'est la meilleure des deux connexions au sens de la fonction de poids (figure b). Maintenant, (p1 p3) respecte la propriété de connexité locale, et donc cette connexion est rejetée (figure c).

En pratique, nous ajoutons au graphe, les arêtes qui ne vérifient pas cette propriété. De cette façon, nous évitons de créer de petits cycles.

Formellement, à l'aide d'un algorithme de plus court chemin [2], nous avons:

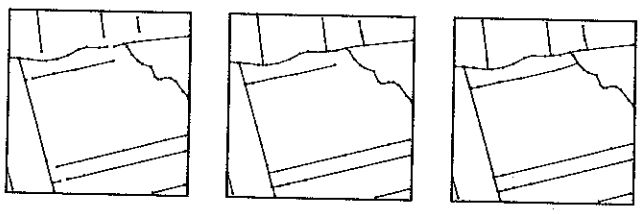
Soit d_{AB} la distance géométrique entre A et B, g_{AB} la longueur du plus court chemin entre A et B dans le graphe. Nous avons la relation suivante:

$$\boxed{\text{Connecter}(A,B) = (g_{AB} \div d_{AB} > K) \text{ et } (g_{AB} + d_{AB} > L).}$$

K est la valeur minimale que peut prendre le rapport entre g_{AB} et d_{AB}. L est la longueur minimale autorisée d'un cycle créé par la connexion AB. Si la condition sur K ou sur L n'est pas vérifiée, la connexion est rejetée.

Remarque: *cette propriété est utilisée pour la fermeture des cycles, mais elle est, en fait, implicite pour la reconstruction par l'ARPM si nous considérons que la distance entre deux points n'appartenant pas aux mêmes composantes connexes dans le graphe est infinie.*

Après la première étape de reconstruction (cf. Image 6b), les connexions possibles restantes sont celles qui ferment les cycles. Nous sélectionnons de façon itérative, la meilleure connexion possible et l'ajoutons au graphe (cf. Image 6c). Celle-ci vérifie trois critères: elle a le poids le plus faible, elle respecte la topologie et elle ne vérifie pas la propriété de connexité locale. Ces deux derniers critères sont calculés à partir du graphe obtenu après chaque itération.



Images 6a: image des traits pleins, 6b après l'ARPM, 6c après fermeture des cycles. (500x500 pixels, grossissement de 1).

6.4 Conclusion

A l'issue de cette étape, nous avons transformé les lignes tiretées en traits pleins, puis reconnecté les traits pleins entre eux. Afin de préserver la topologie, ces opérations sont réalisées en tenant compte d'une contrainte de reconstruction. Pour la reconnexion des traits pleins, nous avons utilisé un découpage polygonal, précédé pour des raisons de performance, d'un découpage elliptique où l'ellipse définit un domaine de visibilité pour une arête source donnée.

7 Extensions pour le réseau hydrographique

Le réseau hydrographique a la particularité, par rapport au réseau routier, de contenir des objets surfaciques. Ces objets sont fragmentés. Il faut les reconstruire puis les reconnecter aux traits pleins.

7.1 Reconstruction de surfaces

Ce traitement concerne la planche des bleus. Ici, la fragmentation des objets surfaciques est due à la présence, au même endroit, de routes de la planche des noirs (cf Image 1). Par opposition aux routes, qui ont une épaisseur constante, les objets surfaciques du réseau hydrographique ont une épaisseur variable et ne sont pas forcément linéaires. Ainsi, nous avons développé une méthode appropriée de reconnexion de surfaces, différente de celle utilisée pour la reconnexion des routes.

Nous travaillons à partir d'une image binaire I qui ne contient que les objets surfaciques (codés 1). Le principe (cf. figure 6) est de calculer le squelette de l'image J, complémentaire de I, de l'ébarbuler et d'en calculer l'approximation polygonale. L'ébarbulage se fait selon des critères d'angle, d'aire et de distance, amplement décrits dans [3]. Nous obtenons alors un graphe, dont chaque arête passe par une zone où l'objet est fragmenté.

Ensuite, nous associons à chaque arête, deux rectangles r1 et r2 ayant pour côtés adjacents, l'arête en question, et pour hauteur une valeur fixée.

Soit F, une fonction calculant le nombre de pixels de l'image binaire I de valeur 1 (objets) dans la zone définie par le rectangle r. On a:

$$F(r) = \text{Card}\ \{p \,/\, I(p) = 1 \text{ et } p \in (\text{Dom}(I) \cap r)\}.$$

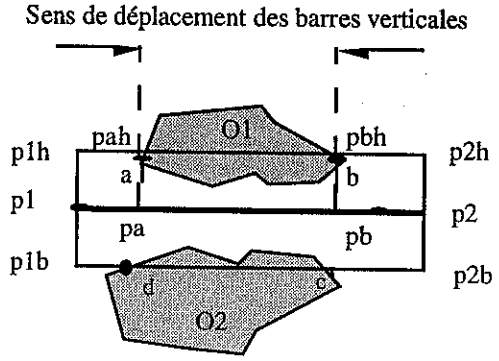
avec I(p), valeur au point p de I, et Dom(I) le domaine de définition de l'image.

Pour les deux rectangles r1 et r2, nous définissons G telle que:

$$G(r1,r2) = \min(F(r1),F(r2))$$

Puis, en faisant tourner légèrement et simultanément r1 et r2 autour du centre de l'arête, nous calculons le couple (a,b) qui réalise le maximum de G.

A la fin, nous remplissons un polygone dont les quatre sommets sont calculés comme le montre la figure suivante:



r1=(p1,p1h,p2h,p2) r2=(p1,p2,p2b,p1b)

Figure 6: Synthèse de la connexion.

L'exemple ci-dessus montre comment sont calculés les points a et b, à partir du rectangle r1 porté par l'arête [p1p2] du squelette de l'image J. Ici, on veut reconnecter l'objet fragmenté composé de O1 et O2. Pour calculer le point a, nous déplaçons un point pa sur le segment [p1p2] en partant de p1, et considérons le segment [pa,pah] perpendiculaire à [p1p2]. Ceci est répété jusqu'à ce que [pa,pah] coupe l'objet O1. Nous appelons a, le point de plus proche de pa sur le segment [pa,pah]. Le calcul du point b est analogue, avec pb qui varie de p2 vers pa. Le calcul des points c et d, pour l'objet O2 s'effectuent de la même manière à partir du rectangle r2. A la fin, on remplit le polygone (abcd).

Les images qui suivent montrent un exemple de reconstruction. L'image de gauche représente l'image initiale et celle de droite l'image reconstruite après transformation de la texture lors de la phase de simplification (§ 4.1.1).

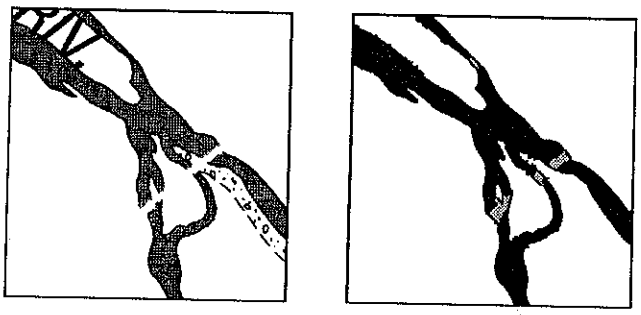


Image 7: reconstruction de surface. (500x500 pixels, grossissement de 1).

7.2 Connexions des traits pleins aux surfaces

Cette étape est la dernière du processus de reconstruction du réseau et est en cours de réalisation. Elle ne concerne que le réseau hydrographique.

8 Résultats

Pour le réseau hydrographique, nous avons choisi comme paramètres de la fonction de poids des tirets (§ 6.2.3): $\delta=\pi/4$; $\mu=1.0$, et comme seuil de rejet d'une connexion 100. Pour la connexion des traits pleins, les seuls paramètres à régler sont les longueurs des demi-axes principaux de l'ellipse de visibilité. Pour le prolongement, nous avons choisi 100.0 pixels, et pour la perpendiculaire 40.0 pixels. Ici, l'image de gauche représente l'image initiale et celle de droite l'image du réseau reconstruit.

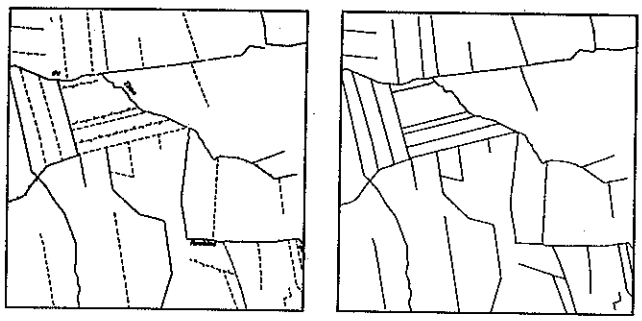


Image 8: reconstruction de réseau hydrographique. (2000x2000 pixels, grossissement de 0.3)

Pour le réseau routier, nous avons choisi $\delta=\pi/3$; $\mu=1.0$ et comme seuil de rejet d'une connexion 200. Les longueurs des demi-axes principaux de l'ellipse ont pour valeur: prolongement: 120 pixels et perpendiculaire: 40 pixels.

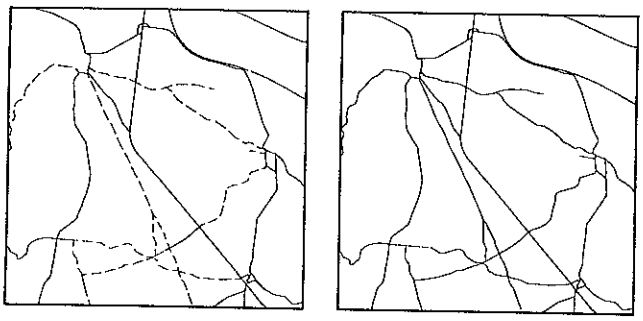


Image 9: reconstruction de réseau routier. (Image 5000x5000, grossissement de 1).

Il y a deux sources possibles de mauvaise reconstruction que le système doit être en mesure d'alerter: celles qui sont dues à une mauvaise simplification de la carte, et celles dues à des situations ambiguës. Une mauvaise simplification de la carte, est obtenue par exemple, en oubliant un disque ou un caractère trop bruité (cas rare) dans l'image initiale. Une situation ambiguë est celle pour laquelle il n'existe pas de solution triviale, voire pas de meilleure solution. Un tel cas, pour lequel il existe plusieurs reconstructions possibles a priori, est présenté dans l'image suivante:

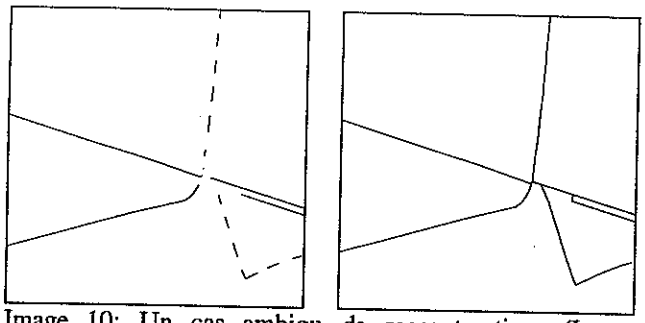


Image 10: Un cas ambigu de reconstruction. (Image 500x500 pixels, grossissement de 1).

Ce cas ne s'est présenté qu'une seule fois pour la planche de Noves, et concerne uniquement les carrefours du réseau hydrographique, les carrefours du réseau routier étant toujours représentés. Pour corriger ou alerter de telles reconstruction, il faudra réaliser un outil spécialisé sur les carrefours du réseau hydrographique.

9 Conclusion et perspectives

Nous avons proposé un algorithme général de reconstruction de réseau. Celui-ci a été appliqué aux réseaux hydrographiques et routiers. Cet article insiste surtout sur la transformation des lignes tiretées en traits pleins, et sur la reconnexion des traits pleins entre eux. Ces deux opérations sont réalisées à l'aide de l'algorithme de l'ARPM et de la propriété de connexité locale. La cohérence topologique est maintenue au fur et à mesure du déroulement de la reconstruction avec un algorithme de découpage polygonal. Les fonctions de poids utilisées ont été élaborées en utilisant des connaissances communes aux deux réseaux, et spécifiques pour chaque type de tracé: le tracé des lignes tiretées et celui des traits pleins.

Une étape préliminaire de simplification de la planche des bleus et de la planche des rouges, a été effectuée pour faciliter la reconstruction de chaque réseau. Pour le réseau hydrographique, un algorithme de reconstruction de surface a été implanté. Un autre algorithme de connexion de contours en mode vecteur est en cours de réalisation. Il servira, entre autre, à valider la cohérence topologique d'une connexion synthétisée, en utilisant une orientation des contours du type "matière à droite". Un travail sur la connexion des traits pleins aux objets surfaciques est en cours de réalisation.

La méthode doit être validée sur les autres réseaux présents sur la carte: limites administratives, lignes ferroviaires, lignes électriques. Par contre, le réseau orographique (courbes de niveaux), devra certainement être reconstruit avec une algorithmique fondamentalement différente. En effet, ce réseau présente deux particularités: 1) il peut être arbitrairement dense, ce qui pourrait engendrer beaucoup de fausses connexions; 2) il possède des règles topologiques particulières et très strictes (c'est un graphe de cycles). Le seul moyen de résoudre les difficultés liées à sa densité est d'appliquer systématiquement les connaissances strictes liées à sa topologie.

Enfin, les connexions ajoutées aux réseaux sont hypothétiques et d'origine différente. Il faut mettre en place un système de validation. Ce dernier s'inspirera des travaux de [15][16]. Pour les traits pleins et les objets surfaciques, les connexions sont calculées en supposant que les interruptions sont dues à d'autres objets. Ici, un processus de validation inter-planches doit être développé. Pour les lignes tiretées, les connexions sont principalement justifiées par les règles cartographiques utilisées pour les représenter. Un processus de validation intra-planche doit aussi être mis en oeuvre.

Bibliographie

[1] Marc Pierrot Deseilligny, Herve Le Men, Georges Stamon. *Map Understanding for GIS Data Capture. Algorithms for Road Network Graph Reconstruction.* Proc. ICDAR 93, pp 676-679. Tsukuba Science City, Japan, October 1993.

[2] Michel Gondran, Michel Minoux. *Graphes et algorithmes.* Collection de la Direction des Etudes et Recherches d'Electricite de France. pp 112-115.

[3] Marc Pierrot Deseilligny. *Lecture Automatique de Cartes.* Thèse de l'université René Descartes. Paris V. Centre Universitaire des Saints-Pères. UFR de Mathématiques et Informatique.

[4] Marc Pierrot Deseilligny, Hervé le Men, Georges Stamon. *Lecture automatique des écritures sur cartes scannées.* in Proc CNED' 94. 3ème Colloque National sur l'Ecrit et le Document, Rouen 6-7-8 Juillet 1994, France, pp 195-202.

[5] T. Kasvand. *Linear Texture in Line Drawing.* Proc. 8th' ICPR Vol. 2, pp 398-401. Paris, 27-31 Octobre 1986.

[6] C.P. Lai, R. Kasturi. *Detection of dashed lines in engineering drawings and maps.* In Proc. ICDAR 91. pp 507-515. Saint Malo, France

[7] L.O' Gorman, R. Kasturi. *Document Image Analysis.* IEEE Computer Society Press. Los Alamitos, California. Washington-Brussels-Tokyo, 1995. pp 101-105.

[8] R.O. Duda, P.E. Hart. *Use of the Hough Transform to Detect Lines and Curves in Pictures.* Comm. ACM, 15, 1, pp 11-15, January 1972.

[9] H. Yamada, K. Yamamoto, K. Hosokawa. *Directional Mathematical Morphology and Reformalized Hough Transformation for the Analysis of Topographic Maps.* IEEE Trans. PAMI Vol. 15, Number 4, pp 380-387, 1993.

[10] G. Agam, H. Luo, I. Dinstein. *Morphological Approah for Dashed Lines Detection.* In Proc. IWGR'95, pp 23-32, August 10-11, 1995, The Penn State Scanticon.

[11] T. Nagao, T. Agui, M. Nakajima. *An Automatic Road Vector Extraction Method from Maps.* In Proc. 9th ICPR'88. pp 585-587.

[12] D. Antoine. *A Technical Document Understanding System Based on a priori* . In Proc. 6th SCIA. pp 843-846. 21 Dec. 1989.

[13] O. Hori, A. Okasaki. *High Quality Vectorization Based On a Generic Object Model.* In Pre-Proc. IAPR Workshop on SSPR. pp 137-153, 28 August 1990.

[14] Sergey V. Ablameyko, Boris S. Beregov, Aleksandr N. Kryuchkov. *Computer-Aided Catrographical System for Map Digitizing.* In Proc. ICDAR 93. pp 115-118. Tsukuba Science City, Japan, October 1993.

[15] J.E. Den Hartog, T.K. ten Kate, J.J. Gerbrands. *Knowledge-based Segmentation for Automatic Map Interpretation.* In Proc. IWGR'95, pp 71-80, August 10-11, 1995, The Penn State Scanticon.

[16] G.K. Myers, P.G. Mulgaonkar, C.H. Chen, J.L. DeCurtins, E. Chen. *Verification-Based Approach for Automated Text and Feature Extraction From Raster-Scanned Maps.* In Proc. IWGR'95, pp 99-99, August 10-11, 1995, The Penn State Scanticon.

[17] J.M. Ogier, J. Labiche, R. Mullot, Y. Lecourtier. *Attributes Extraction for French Map Interpretation.* In Proc. ICDAR' 93. pp 672-675. Tsukuba Science City, Japan, October 1993.

[18] Yvon Gardan. *Eléments de CAO. Matériels et logiciels de base.* pp 179-197. Volume 1 Edition Hermès. Paris-Londres-Lausanne 1988.

[19] M. Coster, J.L.. Chermant. *Précis d'analyse d'images.* pp 82-86; Presses du CNRS, 1989.

MMT 2001

IMPROVING AERIAL IMAGE MATCHING TECHNIQUES IN URBAN AREAS USING A NEW TRUE MULTI-STEREO APPROACH GUIDED FROM OBJECT SPACE

Nicolas PAPARODITIS [*,**], **Grégoire Maillet** [**]

[*] ESGT/CNAM - 1, boulevard Pythagore - 72000 Le Mans
nicolas.paparoditis@ign.fr
[**] MATIS/IGN - 2-4, avenue Pasteur - 94165 Saint-Mandé
gregoire.maillet@ign.fr

KEY WORDS : aerial images, aerial digital frame camera, multiple views, DSM, surface reconstruction, object space, multi-image matching, multi-dimensional correlation, self reliability indicator.

ABSTRACT

This paper describes a novel approach for automated surface reconstruction of urban areas from digital aerial multiple-stereo views. The general reconstruction approach is based on a concept of multi-image matching guided from object space which provides the DSM and the corresponding orthoimage at the same time. The general idea is to match all the images at the same time with a multi-image correlation function instead of merging all elementary DSM computed on all stereopairs. This multi-image correlation function has very interesting properties, we can use very small windows in the image matching process, the correlation scores give a real reliability indicator of the matches and the false matches can be filtered with a deterministic threshold.

1. INTRODUCTION

The reconstruction of Digital Surface Models from aerial stereopairs of urban landscapes has been a subject of constant attention from both photogrammetric and computer vision communities [Baillard & al 96][Cord & al 98]. DSM in urban areas are useful as such for a great number of applications: inter-visibility calculations for the optimisation of the location of telecommunication antennas, risk mapping (floods, etc.), change detection for data base updating, mission planning, virtual reality, etc. Moreover, they are essential for the elaboration of other cartographic products such as orthoimages, 3D building models [Paparoditis & al 98][Fuchs & al 99][Jibrini & al 00], etc.

For the latter application, i.e. building reconstruction, images of very high resolution (<20 cm) are required in order to bring out conspicuously the whole set of 3D structures which are necessary for the elaboration of a fine description of a building. At these levels of resolutions and in a context of dense urban tissue, there are a considerable number of hidden parts to the extent that a dense and thorough description of the scene by a stereo analysis becomes impossible. Besides in urban areas, the classical stereo processing admits some flaws because of all the matching ambiguities encountered due to: homogeneous surfaces, discontinuities, occlusions, mobile vehicles, non lambertian surfaces, and the poor image quality due to the image scanning process.

The acquisition of multiple-stereo views (a particular spot of the landscape is seen in more than two images) solves many problems. Actually, provided a sufficient overlapping exists, there will always be one or more couples among the whole set of couples in which a given point of the landscape is visible and where some of the problems described previously, i.e. mobile vehicles and non lambertian surfaces, will not appear. Ways of exploiting this data has been largely treated in recent literature [Okotumi & al 91][Gabet & al 94][Canu & al 95][Leloglu & al 98]. In most of the techniques, the general idea is to reconstruct the DSM by an *a posteriori* merging of all the elementary DSM calculated on all the possible stereopairs by a voting technique (e.g. based on the median value). This merging allows a densification and reliabilisation of the results to a large extent when the elementary DSM are complementary and when the results are valid in most of all the DSM.

In the contrary, the stereo matching process requires the use of non-negligible window sizes if one is looking for sufficiently reliable measures. One knows well that the morphological quality of the DSM, i.e. the ability to render discontinuities, slopes, slope breaks, surface microstructures, all of utmost importance for numerous applications, is directly dependent on the window size. The larger the windows and the more the "high frequencies" of the surface are smoothed out, the more the depth discontinuities are delocalised, and the more the matching of steep slopes is difficult (due to image deformations).

One way to reduce the window size is to increase the signal to noise ratio of the images so as to reduce matching ambiguities. This can be obtained naturally by having an imaging system of higher quality, or artificially by increasing the number of observations. The novelty of the work presented in this paper lies in the use of high quality images (SNR=300) acquired with IGN's 12 bits panchromatic digital frame camera (see figure 1) [Thom & al 99] and of highly overlapping images to improve the surface reconstruction problem with simple algorithms but really adapted to the nature and to the complementarity of the data.

Figure 1: Digital frame camera of IGN

2. OUR APPROACH: A MULTI-IMAGE MATCHING GUIDED FROM OBJECT SPACE

A new matching process concept adapted to these data has therefore been developed. This process allows the matching of the images and the construction of the raster DSM, and the orthoimage all at the same time without having to go through the processing (epipolar resampling, image matching, 3D reconstruction, and 3D resampling) of all stereopairs and the merging of all elementary DSMs. This process is based on the concept of direct multi-image matching guided from object space (see Figure 2).

The process is based on the following algorithm. For every single node (x,y) of the DSM raster grid, a correlation profile is constructed gathering all the correlation scores calculated for each of the possible z through a plausible interval $[Z_{min},Z_{max}]$ depending on an a priori knowledge of the scene (given by a map, a gross DTM, etc.). For each given z, one calculates in the whole set of images, the hypothetical corresponding (i_k,j_k) image co-ordinates in each image space k (where $0 \leq k \leq n$ and n is the number of images where this (x,y,z) point is seen). The likelihood of these hypotheses is given by a direct measurement of the similarity of the set of windows centred on the (i_k,j_k). The "estimated" Z value retained for a given (x,y) is the one for which the optimal value of the correlation profile is reached. Besides, for this (x,y) and for this estimated Z we can calculate directly the corresponding grey level in the orthoimage from the set of associated (i_k,j_k) grey-levels by taking for instance the average or the median of the values.

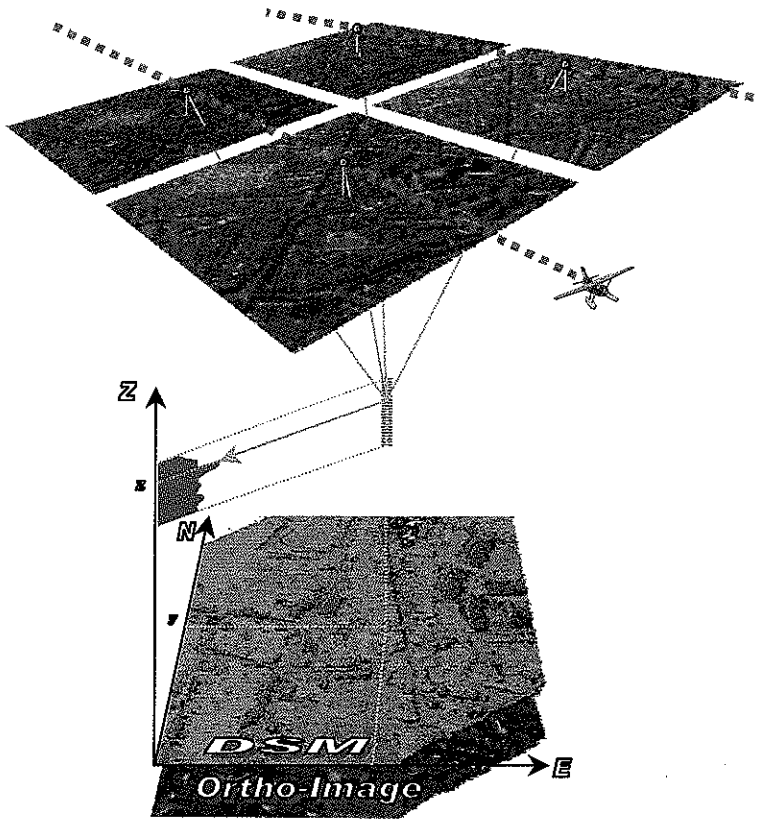


Figure 2: Principle of the multi-image matching process guided from object space

Working from object space offers a good number of advantages. One can work in a transparent way on images of different resolutions, and the parameters are expressed in a metric form. Also we avoid the blind resampling of all the 3D samples corresponding to all image matches in order to generate a regular grid DSM in object space. Indeed, these samples when the matching process is guided from image space follow, although their distribution is regular in image space, an irregular spatial distribution in object space. Finally, its major advantage is that it allows, on the one hand to treat N images in a natural and transparent way, and on the other hand, to construct at the same time the DSM and the corresponding orthoimage (See Figure 2).

3. A MULTI-IMAGE SIMILARITY FUNCTION

How can one measure the similarity of a set of image windows? The direct extension of the two window cross-correlation function is not possible because this provides the scalar product and therefore an information on the angle formed by both texture vectors whose components are the spatially ordered set of grey-levels inside the window. Therefore we have defined a new adapted Multi-Image Correlation (MIC) function:

$$0 \leq \mathrm{MIC}(i_1, j_1, i_2, j_2 \ldots, i_n, j_n) = \frac{Var(\sum_{k=1}^{n} \vec{V_k}(i_k, j_k)))}{\sum_{k=1}^{n} Var(\vec{V_k}(i_k, j_k))} \leq n \qquad (1)$$

where $\vec{V_k}(i_k, j_k)$ is the texture vector associated to the window centred on the pixel (i_k, j_k) in image k and $Var(\vec{V_k}(i_k, j_k))$ the variance of the texture vector $\vec{V_k}(i_k, j_k)$, i.e. the grey levels inside the window centred on the pixel (i_k, j_k) in image k and n is the number of images.

4. RADIOMETRIC WEIGHTING

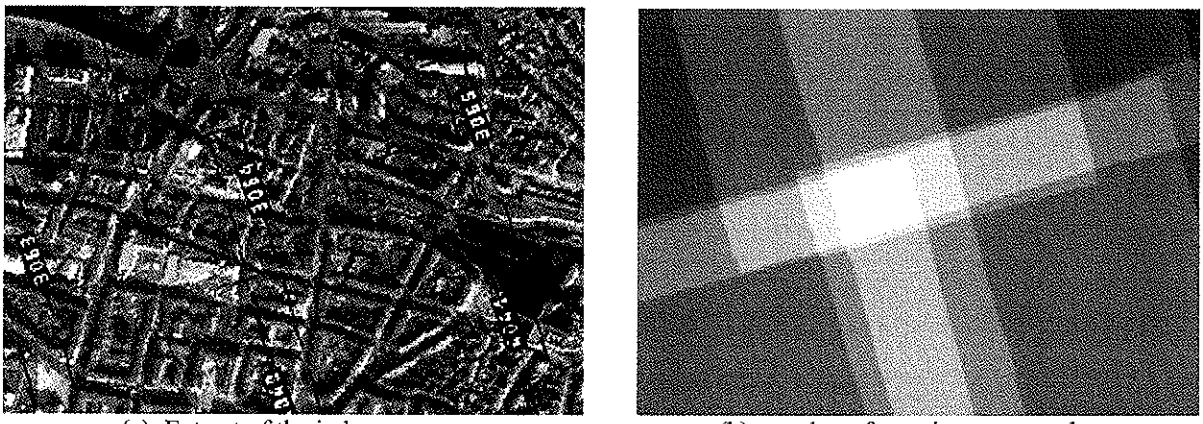
In view of the great radiometric stability of the images of the digital aerial frame camera (which is not the case of scanned images) and under the assumption that most of the objects which describe the landscape have lambertian characteristics, we impose an additional constraint on the absolute radiometry of homologous neighbourhoods, in the shape of a weighting function. This weighting is necessary. Indeed, the MIC similarity function is "centred average". This function measures the similarity of the textures of all neighbourhoods but not their radiometric similarity; this can give rise to mismatches. Our new similarity function called MICRA (Multi Image Correlation with Radiometric Attenuation) can be expressed in the following way:

$$\mathrm{MICRA}(i_1, j_1, \ldots, i_n, j_n) = \mathrm{MIC}(i_1, j_1 \ldots, i_n, j_n) . \exp - \left(\frac{\mathrm{Var}_k(I_k(i_k, j_k))}{k} \right). \qquad (2)$$

where Var is the variance, I(i,j) the grey level of pixel (i,j), and k a normalising coefficient. The application of this weighting function permits to obtain a similarity criterion which characterises at the same time the texture and the radiometric resemblance of the neighbourhoods. Therefore the MICRA similarity function is more robust and discriminating.

5. SPLITTING THE SURVEY IN ELEMENTARY AREAS WITH DIFFERENT VIEWING CONFIGURATIONS

The process described above considers a set of overlapping images and aims at producing 3D measurements on the landscape seen within the images. The reconstruction of 3D measurements in view of generating a DSM on a complete survey thus requires a splitting of the cartographic space in elementary areas (see Figure 3.b) which are covered by the same given set of images. These areas are determined using the absolute orientations of every image (provided by the aerial triangulation process or by GPS and INS measurements or both) and an a priori knowledge on the landscape elevations, e.g. bounding elevations or a gross DTM or DSM.

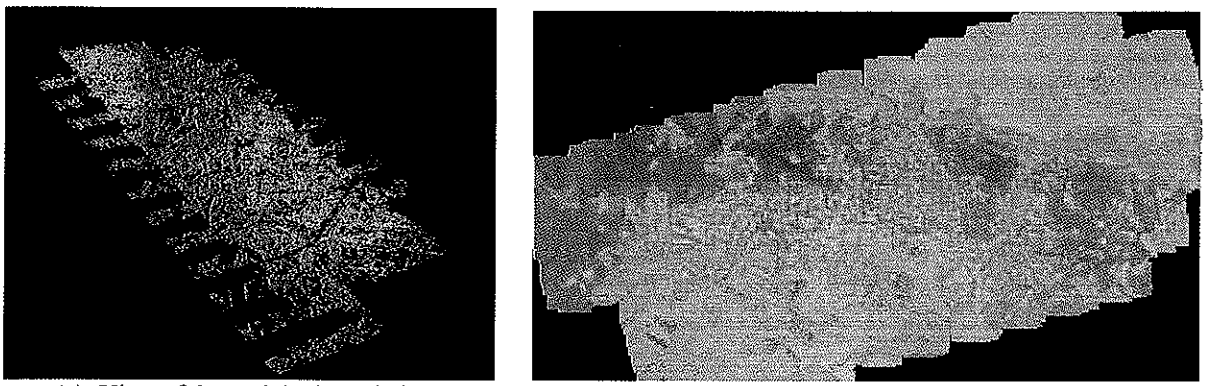


(a): Extract of the index map (b): number of seen image ground map

Figure 3: 20cm digital survey on the town of Amiens with a 60 % across track and along track overlap.

6. INITIALISING THE MATCHING PROCESS

One of the interests of automated aerial-triangulation vs. GPS and INS measurements is that the features used for the triangulation process can also be used for the matching and the DSM reconstruction process. Indeed, tie points extracted manually or automatically from the images can be localised in 3D (see figure 3.a) and a gross DSM can be built from these 3D measurements using for instance a Delaunay triangulation. This gross surface can be used to limit the matching search space and thus matching ambiguities. In the case of our digital images, the tie points extracted have a very high reliability and spatial density [Boldo 01] and the resulting DSM is not so gross (see figure 3.b) and helps for a very good initialisation of the multi-image matching process.



(a): View of the aerial-triangulation tie points located in 3D (b): DSM built from the 3D tie points with the help of a Delaunay triangulation

Figure 4: Building a gross DSM from the automatic tie points extracted from the Amiens survey

7. RAW RESULTS OF THE MATCHING PROCESS AND COMMENTS

The increase in the number of observations, allows, on the one hand to reduce, sizeably, matching ambiguities met within the classical stereo-processing, and thus to increase the reliability of the process. On the other hand, it allows, also thanks to the high image quality provided by our digital frame camera, to use sizes for the correlation windows of 3x3. The results on a dense urban area, show clearly on Figure 5 a dense DSM, a good rendering of the landscape microstructures, and a good localisation of the discontinuities (especially the building edges). However the very small size of the windows leads to some false matches in very homogeneous areas. Moreover, false matches also occur in occluded areas and for non lambertian areas due to the fact that the contents of the image windows corresponding to the correct surface element are not alike which in consequence disadvantages the similarity measure criterion.

Figure 5: 4000x4000 raw DSM obtained with our process with 3x3 windows

Figure 6 : Correlation scores corresponding to the DSM of figure 5

8. AUTOMATIC FILTERING OF FALSE MATCHES

On the contrary of correlation scores obtained with classical stereo-matching, our multi-correlation score provides a good self-indicator of the measurement reliability. Figure 5 and Figure 6 show that most of all the DSM erroneous points have much weaker correlation scores. A threshold value can be determined automatically for our multi-image correlation score through the comparative study of the distributions of correlation scores for good and false matches [Paparoditis & al 00]. The filtering using this threshold is extremely efficient (see figure 7). Even though, this filtering is efficient and allows the elimination of most of false matches, there are still some false remaining samples (isolated points or very small aggregate areas) especially in homogeneous areas due to the very small size of the correlation window. Using bigger windows e.g. 5x5 or 7x7 would improve the results in these areas but would alter the rendering of, for instance, relief microstructures and buildings limits. These features being essential for many applications, we prefer using 3x3 windows and applying grey-level morphological filters to clean all ill remaining areas.

Figure 7 : DSM of Figure 5 filtered with an automatic threshold on the correlation scores of Figure 6.

The previously described radiometric weighting function penalises those areas which are not seen (hidden areas) or rather which are "well seen in the same way" only in a subset of images (non lambertian surfaces, mobile vehicles, etc.). Our matching process tends voluntarily to determine solely the landscape points seen in a rather similar way on all the images. As a result, the filtering processes will remove all other points and leave missing areas inside the DSM. In consequence, our general strategy is to complete in a progressive and hierarchical way the DSM starting from the most reliable measures in order to avoid the propagation of errors.

9. CONCLUSION

We have described an original and novel approach for the automatic surface reconstruction of urban areas from multiple-stereo aerial images (across track as well as along track). This system with all its simplicity in use, supplies extremely promising results with particularly very interesting properties for orthoimage production, relief micro-structure detection, and especially for building reconstruction. On the one hand, the generated 3D measurements are dense, reliable, accurate, and correctly reproduce all the relief morphology (discontinuities, steep slopes, slope breaks, microstructures, etc.) thanks to the very small window sizes we use.

On the other hand, a reliability indicator for each DSM sample is obtained, which can be used either for a direct back-tracking on our processes and a real hierarchical strategy, or to point out "uncertain" areas so that other processes based on the analysis of the DSM may take these into account. The morphological quality of the DSMs produced with our image processing approach and strategy on high quality digital images should be better than that obtained with LASER ranging techniques, and the accuracy should be very close.

REFERENCES

[Baillard &al 96] Baillard C., Dissard O., Jamet O., Maître H. Extraction and characterization of above-ground areas in a peri-urban context, *Proccedings of the IAPR/TC-7, Workshop on Methods for Extracting and Mapping Buildings, Roads and Other Man-made structures from Images*, pp.159-174, Graz, Austria, 2-3 September, 1996.

[Boldo 01] D. Boldo, Trends in a aerial triangulation automation using only the images, 3rd International Symposium on Mobile Mapping Technology, Cairo, Egypt, 6-10 January, 2001.

[Canu & al 95] D. Canu, N. Ayache, J.A. Sirat, « Accurate and robust stereovision with a large number of aerial images », *SPIE*, Vol. 2579, pp. 152-160, Paris, September, 1995.

[Cord & al 98] M. Cord, N. Paparoditis, M. Jordan. « Dense, reliable and depth discontinuity preserving DEM computation from. H.R. V. images», proceedings of ISPRS Commission II symposium, Data Integration : Systems ans techniques, vol 32 n°2, pp. Cambrige, Grande-Bretagne, July, 1998.

[Fuchs & al 99] F. Fuchs, H. Le Men, « Building reconstruction on aerial images through multi-primitive graph matching», Proceedings of the 2nd IAPR workshop on graph-based representations, 1999.

[Gabet & al 94] L. Gabet, G. Giraudon, L. Renouard, « Construction automatique de Modèles Numériques de Terrain à haute résolution », Bulletin SFPT n°135, 1994.

[Jibrini & al 00] H. Jibrini, , N. Paparoditis, M.P. Deseilligny, H. Maître, Building reconstruction from very high resolution digital stereopairs using cadastral ground plans, *International Archives of Photogrammetry and Remote Sensing, ISPRS, Geo-Information for all*, Amsterdam, Pays-Bas, July, 2000.

[Leloglu & al 98] U.M. Leloglu, M. Roux, H. Maître. « Dense urban DEM with three or more high resolution aerial images », ISPRS Symposium on GIS, Between Vision and applications, Stuttgart, Septembre 1998.

[Okotumi & al 91] M. Okotumi and T. Kanade, « Multiple Baseline Stereo », IEEE transactions on Pattern Analysis and Machine Intelligence, vol 15 n°4, Avril 1993.

[Paparoditis & al 98] N. Paparoditis, M. Cord, M.Jordan, J.P. Cocquerez. Building detection and Reconstruction from Mid to High Resolution Images. *Computer Vision and Image Understanding*, 72(2), pp. 122-142, November 1998.

[Paparoditis & al 00] N. Paparoditis, C. Thom, H. Jibrini, Surface reconstruction in urban areas from multiple views of digital aerial frame cameras, *International Archives of Photogrammetry and Remote Sensing, ISPRS, Geo-Information for all*, Amsterdam, Pays-Bas, July, 2000.

[Thom & al 99] C. Thom, J.P Souchon, « IGNs digital aerial frame camera: work in progress », *In proceedings of the Photogrammetric Week*, Stuttgart, September 1999.