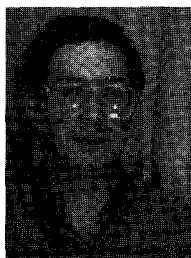


## Fermeture des contours

### par un opérateur local

Contour closing with local operator



**M. MILGRAM**

ENSEA, Équipe Traitements d'Images, impasse des Chênes-Pourpres, 95000 CERGY

M. Milgram a passé l'agrégation de mathématiques en 1971 et a enseigné pendant 3 ans au lycée de Compiègne. Il a ensuite occupé des postes d'assistant et de maître-assistant dans le département de Génie Informatique de l'Université de Compiègne. En 1982, il a passé une thèse d'État sur les Réseaux d'Automates. En 1983, il a quitté l'Université de Compiègne pour l'ENSEA de Cergy où il occupe depuis un poste de professeur. Il a été responsable du laboratoire de recherche de cette école jusqu'en avril 1986.



**J.-P. COCQUEREZ**

ENSEA, Équipe Traitements d'Images, impasse des Chênes-Pourpres, 95000 CERGY

J. P. Cocquerez a obtenu le diplôme d'ingénieur de l'ENREA en 1971. Il est Docteur ès Sciences depuis 1984. Il a été responsable du département informatique de l'ENSEA d'octobre 1983 à octobre 1986. Il a étudié les problèmes de localisation de structures artificielles dans les images aériennes. Il est actuellement en délégation à l'INRIA dans l'équipe de vision par ordinateur où il travaille sur les problèmes de mise en correspondance d'images stéréo.

### RÉSUMÉ

Cet article présente une nouvelle méthode de fermeture de contours. Elle utilise les distances de « chamfer ». Une étude théorique débouche sur la définition d'un opérateur simple et performant. Des essais ont été faits sur des images aériennes. Les lacunes de 5 à 10 pixels sont comblées après deux à trois applications successives de l'opérateur. Ce dernier peut être câblé; un schéma de principe est proposé.

### MOTS CLÉS

Traitement d'images, détection de contours, segmentation, fermeture.

### SUMMARY

*This paper presents a new method to close extracted contours in digitalized pictures. This method use "chamfer array". A theoretical study results in the definition of an efficient and simple operator. Breaks of 3 to 10 pixels are closed after two to three applications of the operator. We propose an hardware implementation of this one.*

### KEY WORDS

*Image processing, edge detection, segmentation, closure of contours.*

## TABLE DES MATIÈRES

### 1. Introduction

### 2. L'opérateur de fermeture

- 2.1. Motivations
- 2.2. Image des distances
- 2.3. Détermination des points-selle
- 2.4. Utilisation des points-selle
- 2.5. Suppression de certaines détections de faux points-selle dans le cas discret

### 3. Proposition d'opérateur câblé de fermeture de contour

### 4. Résultats expérimentaux

### 5. Conclusion

## 1. Introduction

L'extraction de contours est, depuis le début du traitement numérique des images, un des outils les plus puissants pour segmenter et analyser une image. De nombreux opérateurs ont été proposés pour réaliser cette tâche, utilisant presque toujours la composante « haute fréquence » du signal « intensité lumineuse » sur les points de contraste. Nous réserverons dans la suite le terme de *point contraste* pour un point où l'intensité varie brutalement (référence à l'image) et *point de contour* pour les images des points de la scène pouvant donner ces variations. Pour nous, un point de contraste est, soit un point de contour, soit un artefact ou bruit. Inversement, dans une image donnée, des points de contour peuvent ne pas être classés comme points de contraste par les opérateurs utilisés (Sobel, Nagao + différentiel, etc.). Il est possible de les récupérer par des méthodes moins locales utilisant certaines caractéristiques topologiques, géométriques, radiométriques ou sémantiques.

Les caractères *topologiques* les plus fréquents sont la connexité d'un ensemble de points de contours et le fait qu'une région doit être fermée (ensemble connexe du plan-discret pour la topologie des 8-voisins tel qu'il reste connexe lorsqu'on lui retire un point quelconque et devient non connexe dès qu'on lui retire deux points quelconques).

*Géométriquement*, un contour doit être simple, c'est-à-dire de courbure constante (droite, cercle) ou du moins être approché par une courbe du second degré. Le caractère géométrique le plus fréquemment utilisé est le caractère rectiligne.

Sur le plan *radiométrique*, un contour doit séparer deux zones respectivement homogènes mais dont la

réunion ne l'est plus. Si un point de contour est *manquant* (c'est-à-dire non détecté comme point de contraste), la variation locale d'intensité ne suffit pas à rendre compte de cette propriété.

Les propriétés *sémantiques* d'un contour sont sûrement les plus robustes mais aussi les moins générales, invoquant un modèle de ce qui est vu et utilisant de nombreuses informations contextuelles au sens le plus général du terme. Le recours aux contraintes sémantiques est néanmoins indispensable pour lever certaines ambiguïtés comme dans de nombreux domaines de la reconnaissance des formes.

Nous allons examiner le problème de la fermeture des contours, c'est-à-dire de la récupération des points de contour manquant d'un double point de vue :

– *Opérationnel* : temps et type de calcul, parallélisme, flot de donnée uniforme, etc.

– *Qualitatif* : caractère erroné des points ajoutés dans le cadre d'une interprétation des régions obtenues.

## 2. L'opérateur de fermeture

### 2.1. MOTIVATIONS

L'idée de base de notre méthode est la suivante :

- (a) remplacer le tableau des intensités par un tableau des distances au point de contraste le plus proche;
- (b) trouver les points où la fonction passe simultanément par un maximum dans une direction donnée et par un minimum dans une autre direction;
- (c) éliminer les points détectés qui possèdent une distance associée trop importante;
- (d) ajouter ces points-selle aux points de contraste et refaire (a), (b), (c) et (d) tant que c'est possible;
- (e) éliminer les points de contours que l'on a ajoutés et qui ne ferment pas de région « interprétable ».

Nous allons tout d'abord justifier notre approche par quelques exemples puis par des considérations théoriques.

La première transformation que subit l'image est donc l'extraction des points de contraste. Cette opération est une classification, elle nécessite donc le choix de seuils et amène deux difficultés antinomiques :

- ( $\alpha$ ) *fausse détection et sur-détection* : points de contraste qui ne correspondent pas à un point de contour, contour trop épais ou dédoublé en raison du caractère discret, du flou du contour, etc.;
- ( $\beta$ ) *sous-détection* par contraste insuffisant (seuils inadéquats localement).

Remarquons que ces deux types d'erreur peuvent coexister dans différentes zones d'une même image ou même se combiner pour donner des contours chaotiques non connexes (fig. 1).

Nous avons pris l'option de ne conserver qu'une information binaire de l'opération d'extraction des points de contraste.

# Fermeture des contours

## par un opérateur local

Contour closing with local operator

M. MILGRAM  
J.-P. COCQUEREZ

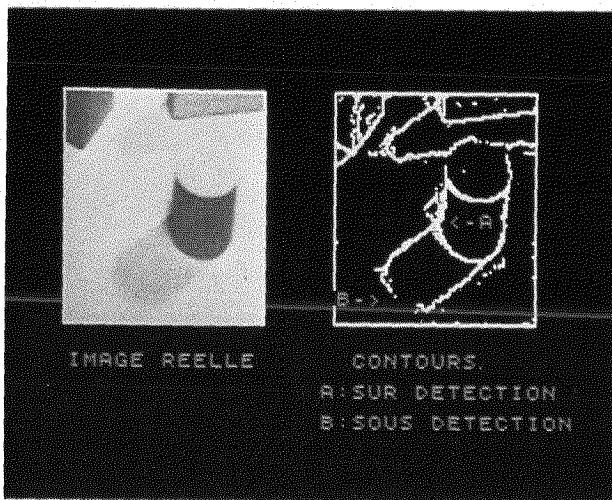


Fig. 1. — Sous-détection et sur-détection de contours.

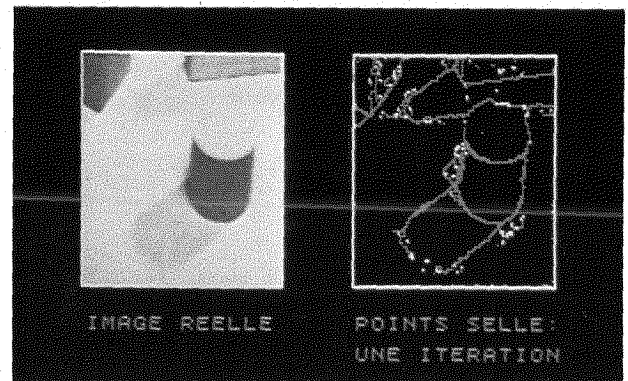


Fig. 17. — Exemple de résultats obtenus sur une image après une seule itération.

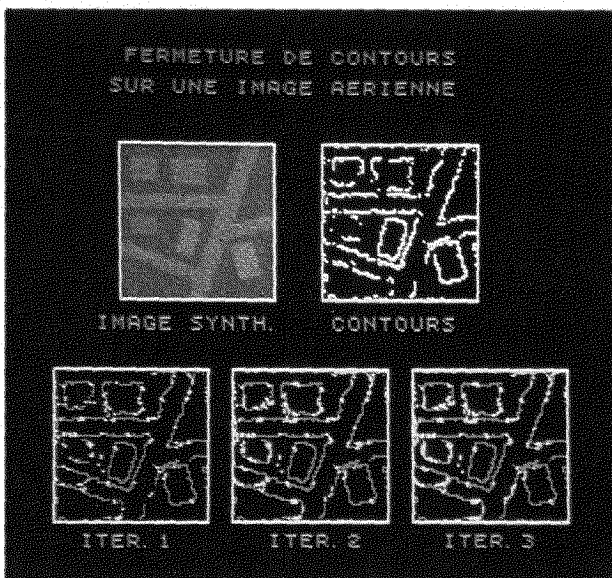


Fig. 18. — Fermeture de contours sur une image synthétique; effets obtenus après chaque itération.

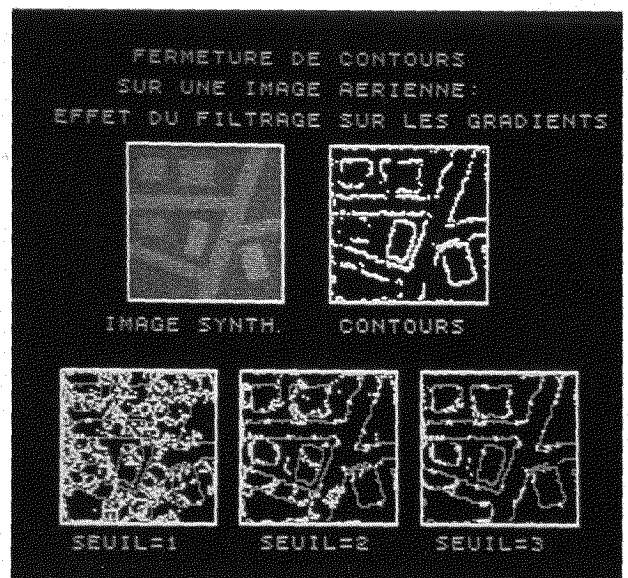


Fig. 19. — Fermeture de contours; effets du filtrage sur les gradients.

# APPLICATIONS

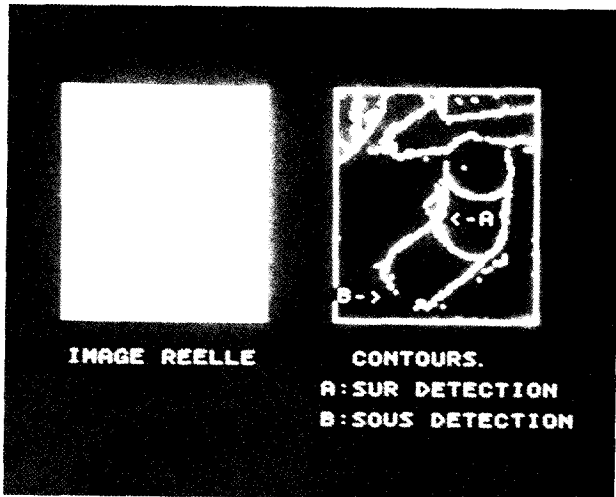


Fig. 1. — Sous-détection et sur-détection de contours.

Les informations intensité et orientation du gradient ont à notre avis un caractère trop local et parfois trop entaché d'erreurs pour être utiles à la fermeture du contour, du moins dans un premier temps. Leur prise en compte alourdit aussi considérablement les calculs.

La première opération à réaliser sur l'image binaire qu'est le graphique des points de contraste va être décrite maintenant.

## 2.2. IMAGE DES DISTANCES

Nous allons remplacer le graphique des points de contraste par une image où la valeur de chaque pixel sera la distance séparant ce pixel du point de contraste le plus proche. Les points de contraste recevront donc la valeur zéro dans cette nouvelle image.

La distance est une pseudo-distance euclidienne (en fait, on calcule le double de la distance pour travailler avec des nombres entiers et utiliser l'approximation de 2 par 3/2) introduite par Barrow *et al.* [Bar 1] puis Borgfors [Bor 1], [Bor 2] et Danielson [Dan 1].

Nous donnons ici sans démonstration le principe de ce calcul. Celui-ci utilise deux masques 3x2 passés chacun une seule fois sur l'image des contours et impliquant pour chaque pixel quatre additions et la recherche du minimum de cinq termes (quatre comparaisons).

L'idée de base est que le chemin le plus court d'un point de contraste A à un pixel donné B (fig. 2a) est un segment.

Le masque suivant est utilisé lors d'un premier balayage de l'image de la gauche vers la droite et du haut vers le bas :

	$j-1$	$j$	$j+1$
$i-1$	3	2	3
$i$	2	0	

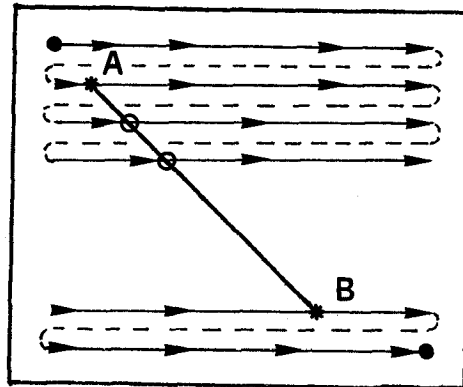


Fig. 2a. — Balayage classique.

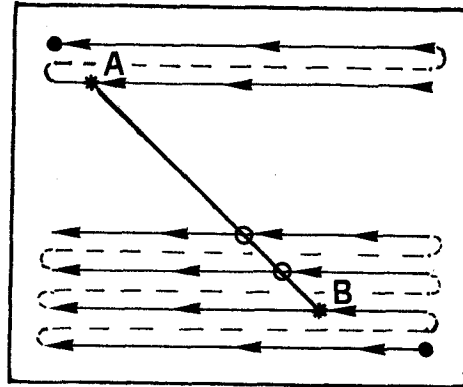


Fig. 2b. — Balayage inverse.

A chaque déplacement du masque, la distance optimale pour le pixel  $(i, j)$  du masque est actualisée par la formule :

$$d(i, j) = \text{MIN} \{ d(i, j), d(i, j-1)+2, d(i-1, j-1)+3, d(i-1, j)+2, d(i-1, j+1)+3 \}$$

Au préalable le graphe des distances a été initialisé de la manière suivante :

$$d(i, j) = \begin{cases} 0 & \text{si } d(i, j) \text{ est un point de contraste} \\ +\infty & \text{dans les autres cas} \end{cases}$$

On imagine donc, dans le cas de la figure 2a, que le premier balayage permet de tenir compte du point A et de tout point de contraste situé au-dessus de la ligne balayée.

Mais, au fur et à mesure que l'on se rapproche du point B, l'influence de ce dernier devient prépondérante. Il n'est pas possible d'en tenir compte tant qu'il demeure exclus du masque. Il est donc nécessaire d'effectuer un deuxième balayage de l'image de la droite vers la gauche et du bas vers le haut en utilisant un deuxième masque (fig. 2b).

		0	2	$i$
	3	2	3	$i+1$
	$j-1$	$j$	$j+1$	

avec :

$$d(i, j) = \text{MIN} \{ d(i, j), d(i, j+1)+2, \\ d(i+1, j-1)+3, d(i+1, j)+2, \\ d(i+1, j+1)+3 \}$$

On peut remarquer que l'ordre de déroulement des deux balayages avec leur masque associé est indifférent.

Un exemple de calcul sur une petite image est présenté figures 3 et 4 :

A	0	2	4	6	8	10	
	2	3	5	7	9	11	
	4	5	6	8	10	0	B
	6	7	8	9	3	2	
	8	9	10	6	5	4	

La zone entourée d'une ligne pointillée a subi l'influence du point de contraste B, le reste celle du point A.

Fig. 3. — Tableau des distances après un balayage direct seulement.

A	0	2	4	6	5	4
	2	3	5	5	3	2
	4	5	6	4	2	0
	6	7	7	5	3	2
	8	9	8	6	5	4

... Influence de A      -- Influence de B

Fig. 4. — Tableau des distances après un balayage direct puis inverse.

Cette image des distances est une bonne approximation de deux fois la distance euclidienne et se calcule très rapidement sur une image. Nous allons maintenant l'utiliser pour déterminer les points-selle qui vont nous servir à fermer les contours.

### 2. 3. DÉTERMINATION DES POINTS-SELLE

Soit une fonction réelle  $f(x, y)$  de deux variables réelles définie sur un sous-ensemble ouvert  $\theta \subseteq \mathbb{R}^2$  et dérivable par rapport à ces deux variables sur  $\theta$ . On appelle *point-selle* pour  $f$  un point  $(a, b) \in \theta$  tel que :

- (1)  $(\partial f / \partial x)(a, b) = (\partial f / \partial y)(a, b) = 0$ ;
- (2)  $(a, b)$  n'est ni un maximum local, ni un minimum local pour  $f$ .

Considérons maintenant dans  $\theta$  un sous-ensemble  $K$  fermé borné d'intérieur vide (c'est ce qui jouera le rôle de point de contour).

Soit la fonction  $f_K : \theta \rightarrow \mathbb{R}$  définie par

$$f_K(x, y) = \text{MIN} \{ d((x, y), (u, v)) \mid (u, v) \in K \}$$

$$f_K(x, y) = d((x, y), K)$$

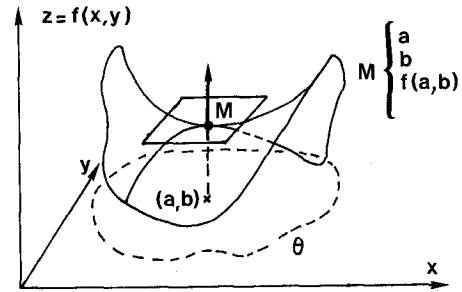
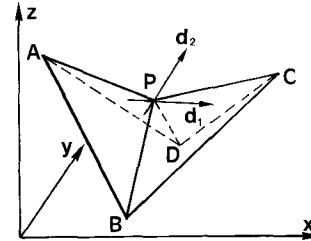


Fig. 5. — Point-selle, première présentation.

On sait que  $f_K$  est continue et nulle sur  $K$ . Cependant  $f_K$  n'est pas partout dérivable et nous devons modifier la définition classique du point-selle.

Nous appellerons *point-selle* un point  $P \in \theta$  tel qu'il existe deux directions  $d_1$  et  $d_2$  dans  $\mathbb{R}^2$  telles que  $f(P)$  soit un maximum relatif dans la direction  $d_1$  et un minimum relatif dans la direction  $d_2$ .



P est un point-selle mais la fonction dont le graphe est un polyèdre n'est pas dérivable en P.

Fig. 6. — Point-selle : deuxième présentation.

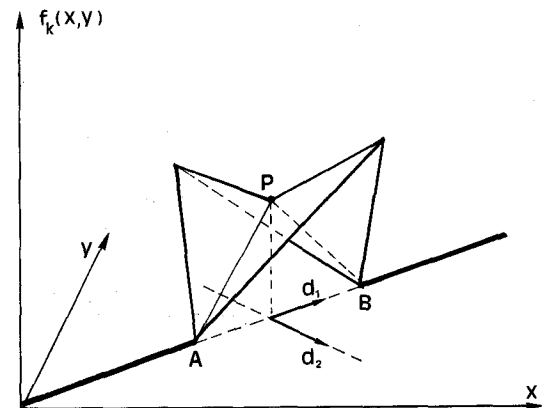
On aura donc :

$$\exists d_1, d_2 \in \mathbb{R}^2, \exists R > 0, \forall \lambda$$

$$|\lambda| \leq R \Rightarrow f(P + \lambda d_1) < f(P) \text{ et } f(P + \lambda d_2) > f(P)$$

Il est possible de transformer les inégalités strictes en inégalités larges comme nous le verrons plus loin.

Considérons la situation décrite par la figure 7, celle de segments ayant des extrémités proches, et la fonction  $f_K(x, y)$  associée :



Dans la direction  $d_2$  on s'approche de  $K$ .  
Dans la direction  $d_1$  on s'éloigne de  $K$ .

Fig. 7. — Exemple de point-selle.

# APPLICATIONS

Nous avons alors les propriétés suivantes qui caractérisent un point-selle (fig. 8) :

Soit  $K$  une réunion d'un nombre fini de segments du plan et  $A, B$  deux extrémités de segments distincts de  $K$ . Soit  $P$  le milieu de  $(A, B)$  et  $r = d(A, P) = d(B, P)$ . On note  $D(M; t)$  le disque fermé de centre  $M$  et de rayon  $t$ .

Supposons que :

- (1)  $D(P; r) \cap K = \{A, B\}$ .
- (2)  $\exists \varepsilon > 0, D(P; r + \varepsilon) \cap K = (A, A') \cup (B, B')$ .

Alors,  $P$  est un point-selle de  $f_K$ , c'est-à-dire :  $\exists \mathbf{d}_1, \mathbf{d}_2 \in \mathbb{R}^2$  tels que :

- (a)  $f_K(P)$  soit un minimum local dans la direction  $\mathbf{d}_2$ .
- (b)  $f_K(P)$  soit un maximum local dans la direction  $\mathbf{d}_1$ .

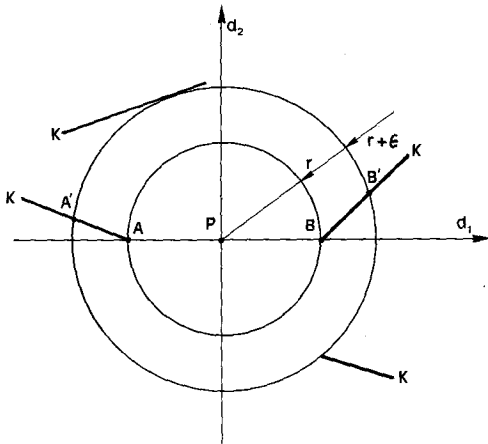


Fig. 8. — Caractérisation d'un point-selle.

Les propriétés (a) et (b) ci-dessus se démontrent de la manière suivante :

On a  $f_K(P) = r = d(P, A) = d(P, B)$ .

Posons  $\mathbf{d}_1 = \overrightarrow{AB}$ ,  $\forall M \in (A, B) - \{P\}$ . On a  $f_K(M) < r = f_K(P)$  et donc  $P$  est bien un maximum local strict dans la direction  $\mathbf{d}_1$ .

Choisissons  $\mathbf{d}_2$  orthogonal à  $\overrightarrow{AB}$  et considérons la frontière de  $\lambda$  :

$$g(\lambda) = f_K(P + \lambda \mathbf{d}_2)$$

On a :

$$g(0) = f_K(P) = r$$

Prenant  $\lambda < \varepsilon$ , on a :

$$D(P + \lambda \mathbf{d}_2; r), \quad D(P; r + \varepsilon)$$

Donc :

$$D(P + \lambda \mathbf{d}_2; r) \cap K \subset (A, A') \cup (B, B')$$

La position des segments  $(A, A')$  et  $(B, B')$  par rapport aux tangentes à  $D(P + \lambda \mathbf{d}_2)$  issues de  $A$  et  $B$  amène :

$$d((A, A'), P + \lambda \mathbf{d}_2) > d(A, P) = r$$

et

$$d((B, B'), P + \lambda \mathbf{d}_2) > d(B, P) = r$$

Ce qui prouve que

$$g(\lambda) \geq r$$

et donc  $P$  est bien un point-selle.

Remarque : Si les segments  $(A, A')$  et  $(B, B')$  sont strictement contenus dans l'extérieur de la bande déterminée par :

$$\mathcal{B} = \{M \mid d(M, \text{Médiatrice}(A, B)) \leq r\}$$

les inégalités ci-dessus sont strictes.

Nous allons maintenant considérer le cas discret, correspondant à une image numérisée (l'ensemble  $\theta$ ) et à un ensemble de points de contours (les points de contraste).

Dans ce cadre, nous devons préciser ce qu'est un point-selle en utilisant les 8-voisins d'un pixel  $P$ .

Les directions  $\mathbf{d}_1$  et  $\mathbf{d}_2$  vont nous donner deux paires  $(I, J)$  et  $(Q, R)$  de pixels voisins situés de part et d'autre du point selle  $P$ . Du fait des effets de la discrétisation nous considérons comme alignées les configurations de la figure 9. Dans le 8-voisinage de  $P$ , il y a alors en tout 12 configurations d'alignements suivant les différentes positions de  $J$ .

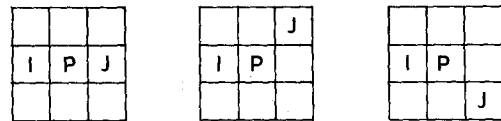


Fig. 9. — Représentation des trois configurations de base pour lesquelles les points  $I$  et  $J$  sont considérés comme alignés avec  $P$ .

Le point  $P$  est un point-selle au sens discret, s'il existe quatre points  $I, J, Q, R$  tels que :

- (1)  $I$  et  $J$  sont alignés avec  $P$ .
- (2)  $Q$  et  $R$  sont alignés avec  $P$ .
- (3) Le segment  $(I, J)$  coupe le segment  $(Q, R)$ .
- (4)  $f_K(I) \leq f_K(P)$  et  $f_K(J) \leq f_K(P)$ .
- (5)  $f_K(Q) \geq f_K(P)$  et  $f_K(R) \geq f_K(P)$ .
- (6) Au moins trois des inégalités représentées en (4) et (5) sont strictes.

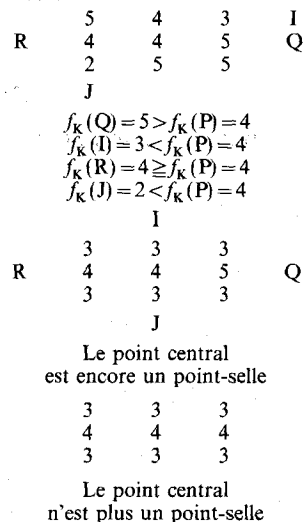


Fig. 10. — Exemples d'application de la définition du point-selle dans le cas discret.

2. 4. UTILISATION DES POINTS-SELLE

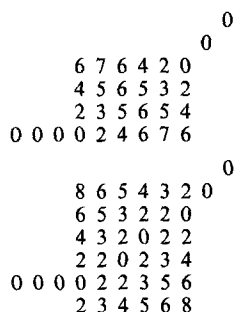
Considérons la figure 11 qui représente deux segments qui, intuitivement pourraient être liés pour former une ligne polygonale convexe. Soit  $K_0$  l'ensemble des points des deux segments et  $f_0(M) = f_{K_0}(M)$  la fonction définie sur l'image par  $f_0(M) = d(M, K_0)$ .

Observons l'effet de l'algorithme suivant sur l'image de la figure 11.

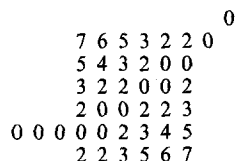
**Algorithme A :**

- (a)  $K := K_0$ .
- (b) Calculer  $\mathcal{P}(f_K)$ .
- (c) Si :  $\mathcal{P}(f_K) = \emptyset$  alors aller en (d).  
Sinon :  $K := K \cup \mathcal{P}(f_K)$ .  
Aller en (b).
- (d) Stop.

où  $\mathcal{P}(f_K)$  désigne l'ensemble des points-selle de  $f$ .



Après une itération



Après deux itérations  $\mathcal{P}(f_K) = \emptyset$ .

Fig. 11. — Exemple de fusion de deux segments.

Nous constatons qu'en deux itérations, les deux segments ont été reliés. La distance initiale approchée était de  $13/2 = 6,5$  entre les deux segments de  $K$ . Puisque chaque itération divise par 2 cette distance, un majorant du nombre d'itérations sera un entier  $i$  tel que :

$$d/2^i < 3/2, \quad i \geq \log_2 d - 0,58$$

avec  $d = 6,5$  nous obtenons  $i = 3$ .

En fait, une itération pouvant amener plus d'un point-selle (c'est le cas ici), le nombre réel d'itérations sera plus faible.

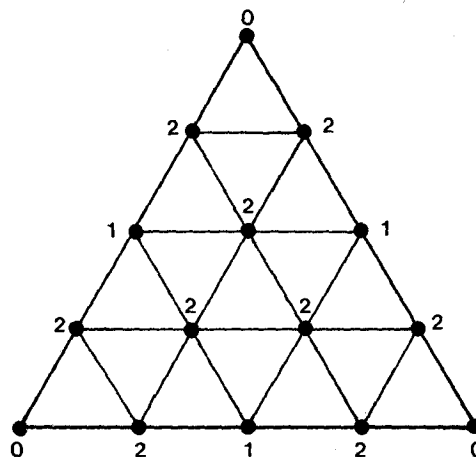
Considérons maintenant un ensemble  $K = A, B, C$ . Si le triangle  $(ABC)$  n'a pas d'angle de mesure supérieure ou égale à  $\pi/2$  les trois milieux de segments sont des points-selle de  $f_K$ .

Par contre, si l'angle  $A$ , par exemple, vaut plus que  $\pi/2$ , le milieu de  $(BC)$  étant plus proche de  $A$  que de  $B$  ou  $C$ , ce n'est plus un point-selle. En conclusion,

pour des points isolés, c'est le caractère aigu ou obtus des angles qui détermine la fermeture.

Considérons un triangle équilatéral  $(ABC)$  (mais ceci reste vrai pour d'autres triangles proches).

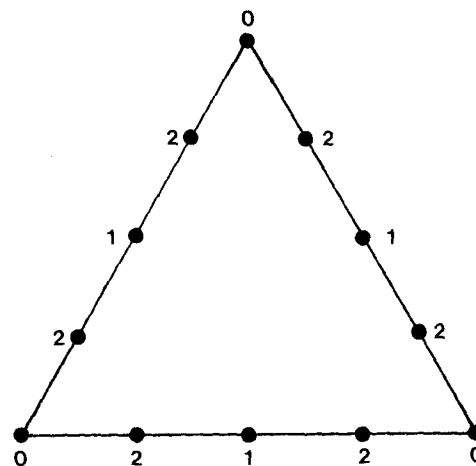
Si on applique plusieurs fois l'opérateur « point-selle » sur  $A, B, C$  on remplit progressivement l'intérieur de ce triangle puisque les milieux des segments constitués de milieux vont intervenir comme point-selle (fig. 12).



Les numéros indiquent l'itération où le point apparaît.

Fig. 12. — Application de l'algorithme A sur un triangle.

On peut alors modifier l'algorithme A en ne prenant en compte que les points-selle tels que parmi ses voisins les plus proches, il existe un point initial ou provenant de deux itérations précédentes. Ceci donne l'algorithme A' :



Aucun point-selle ne vient « remplir » l'intérieur du triangle.

Fig. 13. — Application de l'algorithme A' sur un triangle.

**Algorithme A' :**

- (a)  $K := K_0; i := 0$ ; pour  $k \in K_0$  numéro  $(k) = 0$ .
- (b) Calculer  $\mathcal{P}'(f_K) = \mathcal{P}(f_K) \cap f_K^{-1}(\{(OS)\})$  (point-selle à une distance inférieure ou égale à  $S$ ).

# APPLICATIONS

(c) Si :  $K' := \mathcal{P}'(f_K) = \emptyset$  alors aller en (d).

Sinon :

```

pour  $l \in \mathcal{P}'(f_K)$  faire :
   $V := D(l; f_K(l)) \cap K$ 
  (ensemble des voisins de  $l$  les plus proches dans  $K$ )
   $M(V) := \text{MIN}_{x \in V}(\text{numéro}(x))$ 
  (recherche du point le plus ancien dans  $V$ )
  Si  $M(V) = 0$  ou  $M(V) \leq i - 2$ 
  alors  $K' := K' \cup \{l\}$ 
      numéro( $l$ ) :=  $i$ 
  fin faire
  Si  $K' = \emptyset$  alors aller en (d)
  Sinon  $K := K \cup K'$ 
       $i := i + 1$ 
  aller en (b)
  
```

(d) FIN

## 2.5. SUPPRESSION DE CERTAINES DÉTECTIONS DE FAUX POINTS-SELLE DANS LE CAS DISCRET

L'une des principales utilisations des points-selle est relative à la fermeture de contour quand il y a des lacunes de quelques pixels. Du fait de la discrétisation, il y a des fausses détections, nous en présentons quelques cas figure 14.

Ceci peut être fort gênant, en particulier dans le cas de deux contours parallèles, il y a création d'un isthme après quelques itérations. Ce problème peut être résolu en effectuant deux types de filtrages à chaque itérations.

- lorsque ces fausses détections interviennent pour des points éloignés des contours, il est possible d'effectuer un filtrage en fixant une distance minimale d'intervention de l'opérateur;

- en supposant que les contours séparent des régions de niveaux de gris homogène, les fausses détections de points-selle ne correspondent pas à des points de contraste. Les variations de gradient en ces points sont donc faibles, ce phénomène peut donc être utilisé pour ne pas les valider.

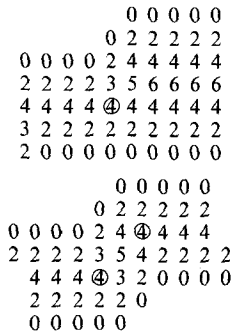


Fig. 14. – Détection de faux points-selle.

## 3. Proposition d'opérateur câblé de fermeture de contour

Cet opérateur va traiter le 8-voisinage d'un point P du tableau des distances pour détecter un point-selle. Appelons  $d_i$  les valeurs des distances pour chaque point de ce voisinage avec  $i=0, 1, 2, \dots, 7, 8$  et en respectant les conventions de la figure 15.

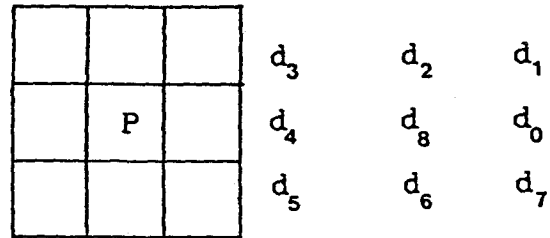


Fig. 15. – Conventions utilisées pour l'opérateur câblé.

Nous posons

$$\bar{d}_i = d_i - d_8, \quad i \in [0, 7]$$

et

$$p(i) = \begin{cases} 0 & \text{si } \bar{d}_i = 0 \\ 1 & \text{si } \bar{d}_i < 0 \\ 2 & \text{si } \bar{d}_i > 0 \end{cases}$$

Nous serons en présence d'un point-selle si par exemple  $p(0)=1, p(4)=1, p(1)=2, p(6)=0$ . En fait, pour ce voisinage il y a  $3^8$  combinaisons possibles de valeurs de  $p(i)$ . Certaines d'entre elles représentant un point-selle.

A chaque configuration, il est possible d'affecter une valeur

$$a = \sum_{i=0}^7 p(i) 3^i.$$

Pour chaque valeur de  $a$ , suivant les valeurs des coefficients  $p(i)$  il est possible d'indiquer si l'on est en présence d'un point-selle ou non. En conséquence, il est possible de créer un tableau  $t$  de  $3^8$  éléments avec :

$$t[a] = \begin{cases} 1 & \text{si la valeur de } a \text{ est celle d'un point-selle.} \\ 0 & \text{autrement.} \end{cases}$$

En pratique, pour chaque point de l'image, la valeur «  $a$  » est calculée. Ensuite, une lecture de  $t[a]$  permet de statuer sur la présence d'un point-selle. L'opérateur câblé comprendra un premier module composé de huit soustracteurs pour calculer les différents  $p(i)$ . Le deuxième module est une mémoire morte contenant les  $3^8$  éléments du tableau  $t$ . La valeur «  $a$  » calculée à partir des différents  $p(i)$  représente l'adresse d'un élément dans la mémoire.



Il est possible de diminuer considérablement le temps de calcul de la valeur « a » en l'exprimant en base 4 :  
On pose

$$p(i) = 2x_i + y_i$$

avec

$$\begin{aligned} x_i &= 0, & y_i &= 0 & \text{si } \bar{d}_i &= 0 \\ x_i &= 0, & y_i &= 1 & \text{si } \bar{d}_i < 0 \\ x_i &= 1, & y_i &= 0 & \text{si } \bar{d}_i > 0 \end{aligned}$$

On obtient alors :

$$a = \sum_{i=0}^7 x_i 2^{2i+1} + y_i 2^{i+1}$$

Les  $x_i$  et  $y_i$  sont utilisés pour adresser chaque élément de la mémoire. Si les soustracteurs employés possèdent les indicateurs N et Z de résultats négatifs et nuls, on établit les relations suivantes :

$$x_i = \overline{N + Z}; \quad y_i = N;$$

Un comparateur permet d'effectuer simplement un filtrage sur les distances. Nous proposons alors, pour l'opérateur câblé, le schéma de principe de la figure 16.

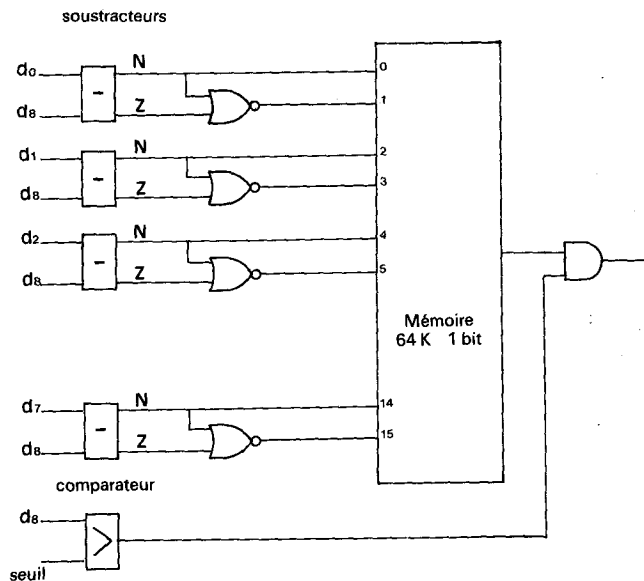


Fig. 16. — Opérateur câblé de détection de points-selle.

#### 4. Résultats expérimentaux

Nous avons fait des essais sur une image réelle de  $90 \times 104$  sur 8 bits. Les contours ont été obtenus en appliquant l'opérateur NAGDIF [COC 1] qui effectue un lissage de l'image puis une détection de contour avec deux opérateurs différentiels. Les résultats pour une seule itération sont fournis sur la figure 17. Le

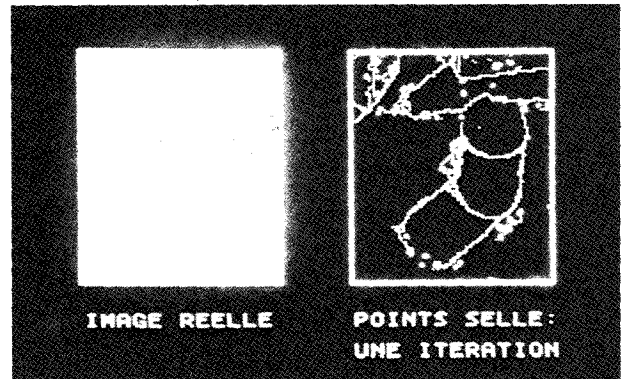


Fig. 17. — Exemple de résultats obtenus sur une image après une seule itération.

temps de calcul est de l'ordre de 16 secondes sur un calculateur HP 1000 A 700.

Ensuite, nous avons appliqué notre méthode sur une image synthétique de photographie aérienne de grande résolution de zone urbaine. Nous montrons sur la figure 18 les effets obtenus après chaque itération.

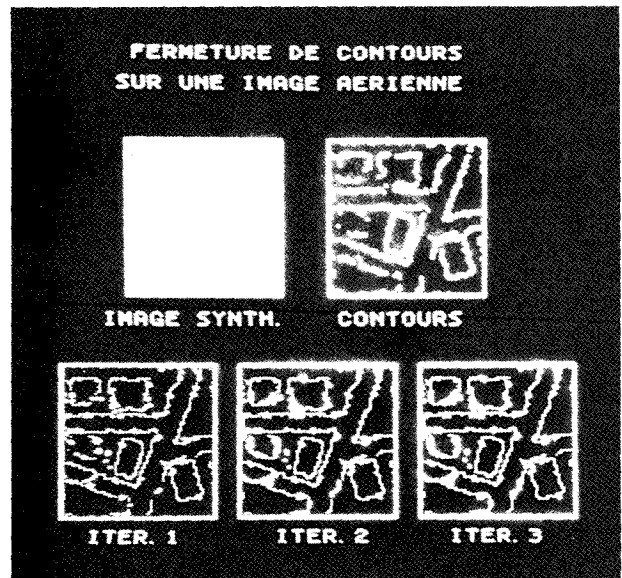


Fig. 18. — Fermeture de contours sur une image synthétique; effets obtenus après chaque itération.

Pour éliminer les faux points-selle nous avons employé la démarche exposée dans le paragraphe 2. 5. Par exemple, pour l'image aérienne le seuil exprimé en niveau de gris de l'opérateur NAGDIF est de 5. Pour supprimer certains faux points-selle, nous avons appliqué, à chaque itération, un opérateur différentiel à quatre masques [Coc 1] avec un seuil égal à 3. Nous présentons sur la figure 19 les résultats obtenus, après trois itérations, pour différentes valeurs de seuil dans le filtrage sur les gradients. Le filtrage sur les gradients intervient peu sur le temps de calcul car il n'est appliqué que sur les points-selle qui représentent un

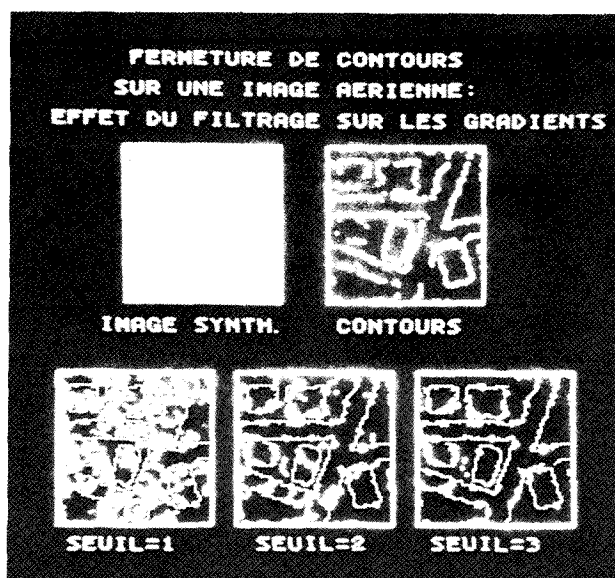


Fig. 19. — Fermeture de contours; effets du filtrage sur les gradients.

faible pourcentage des points de l'image. En ce qui concerne le filtrage sur la distance, nous avons fixé un seuil de 10 pour ne pas traiter les lacunes de dimension supérieure à 10 pas de quantification spatiale.

## 5. Conclusion

Nous avons proposé une méthode locale de fermeture de contour basée sur l'utilisation des distances de

chamfer. Cette méthode demande peu de calcul, puisque la décision de présence d'un point-selle est prise au cours d'un deuxième balayage de l'image, après consultation d'une table de transcodage. L'opérateur câblé se réalise sans problème et peut s'intercaler dans une chaîne de traitement. Les résultats obtenus sont bons et permettent la fermeture effective d'une majorité de lacunes de 5 à 10 pixels après deux ou trois itérations. Cette méthode peut donc être utilisée avec profit pour faire des prétraitements de l'image avant les opérations de segmentation par les régions ou de suivi de contours.

*Manuscrit reçu le 22 mai 1986*

## BIBLIOGRAPHIE

- [Bar 1] H. G. BARROW, J. M. TENNENBAUM et R. C. BOLLES, Parametric correspondance and chamfer matching: Tow new techniques for images matching, *Proc., 5th IJCAI*, 1977.
- [Bor 1] G. BORGEFORS, *Matching edges in maps and Aerial images using the chamfer matching algorithm*, FAO report C 30 315 1, 1983.
- [Bor 2] G. BORGEFORS, Distance transformation in arbitrary dimensions, *CVGIP*, 27, 1984, p. 321-345.
- [Coc 1] J. P. COCQUEREZ et J. DEVARIS, Détection de contours dans les images aériennes : présentation de nouveaux opérateurs, *Traitement du signal*, 2, n° 1, 1985, p. 45-65.
- [Dan 1] P. E. DANIELSON, Euclidian distance mapping, *CVGIP*, 14, 1980, p. 227-228.