

Chapitre 2

La détection des contours dans les images

Chapitre rédigé par Henri MAÎTRE

La détection de contours dans les images a débuté de façon extrêmement empirique par des opérateurs locaux qui, soit estimaient un gradient, soit convoluait l'image par des masques caractéristiques des contours [Haralick et Shapiro, 1985]. Dans les années 80, des approches plus systématiques ont été mises en place par Marr [Marr et Hildreth, 1980], puis Canny [Canny, 1986], pour obtenir des contours plus significatifs. Ces travaux ont abouti à une bonne compréhension de ce qu'il faut faire pour détecter les contours, mais la définition même des contours demeure très vague, ce qui rend ces techniques encore peu efficaces sur un problème concret. De plus, on a pu montrer que le problème de détection de contours est généralement mal posé (au sens de la résolution des systèmes) [Torre et Poggio, 1986]. Les seuls modèles de contours utilisables sont ceux de contours idéalisés, comme ceux présentés sur la figure 2.1 ; ils sont bien loin de la réalité. C'est pourquoi, même si de très gros progrès ont été accomplis dans ce domaine, les techniques empiriques d'estimation du gradient proposées dans les années 70-80 restent souvent encore employées en concurrence de techniques plus modernes. Une excellente référence à ce problème est l'ouvrage collectif [Cocquerez et Philipp, 1996]. Plusieurs articles existent sur la comparaison des performances de détecteurs de contours, en particulier [Palmer et al., 1996, Heath et al., 1997].

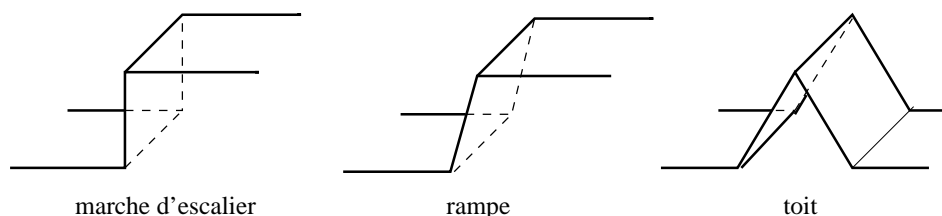


FIG. 2.1 – Quelques modèles de contours. Le plus utilisé est celui en marche d'escalier.

Nous présenterons tout d'abord une approche formelle (mais malheureusement stérile !) de la détection des contours, puis les méthodes empiriques encore utilisées seront vues dans la partie 2. Une approche plus analytique, proposée par Canny, sera ensuite présentée, avec ses dérivées qui sont aujourd'hui les plus employées en raison de leur efficacité. Enfin les techniques fondées sur le principe des **contours actifs** de Kass, Witkins et Terzopoulos seront présentées, ainsi que les approches par **ensembles de niveaux** (*level sets*). Nous terminerons par une brève présentation des méthodes de poursuite et de fermeture des contours.

2.1 Un modèle continu de contour.

Dans une image supposée continue $f(x, y)$, un contour apparaît comme une ligne où sont localisées les très fortes variations de f . Soit \vec{G} le gradient de f :

$$\vec{G} = \vec{\nabla} f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

On associe à f une image du module du gradient de f :

$$G = |\vec{\nabla} f| = \left[\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2 \right]^{1/2} \quad (2.1)$$

ainsi qu'une image des directions du gradient à partir de :

$$\vec{g} = \frac{\vec{\nabla} f}{|\vec{\nabla} f|}$$

et par application de $[-\pi, \pi[$ sur $\{0, 255\}$. Un contour est alors défini comme **le lieu des maximums du gradient dans la direction \vec{g} du gradient**. Un point de contour vérifie donc ;

$$\frac{\partial G}{\partial g} = 0 \quad \text{et} \quad \frac{\partial^2 G}{\partial g^2} \leq 0 \quad (2.2)$$

avec :

$$\frac{\partial}{\partial g} = \vec{g} \cdot \vec{\nabla} \quad (2.3)$$

L'équation obtenue est complexe et non linéaire :

$$\frac{\partial f}{\partial x} \cdot \frac{\partial}{\partial x} \sqrt{\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2} + \frac{\partial f}{\partial y} \cdot \frac{\partial}{\partial y} \sqrt{\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2} = 0$$

Elle n'a donc en général pas de solution explicite et ne peut être résolue que par tâtonnement. On procède alors en deux étapes : on calcule tout d'abord le gradient, puis on recherche les extrémums dans la direction du gradient.

Afin de garantir la double dérivation même en présence de discontinuités de type *marche d'escalier*, on prétraite l'image f par convolution avec une fonction au moins deux fois dérivable. On a beaucoup utilisé pour cela la gaussienne, mais on peut également prendre des polynômes de faible degré.

On peut également se placer dans les axes locaux définis par la tangente \vec{t} et la normale \vec{g} à la surface $f(x, y)$ (cf. figure 2.2). Ce repère est tourné d'un angle Φ par rapport au repère $\{x, y\}$:

$$\Phi = \text{Arctg} \left[\frac{\partial f}{\partial x} / \frac{\partial f}{\partial y} \right] \quad (2.4)$$

On a alors :

$$\frac{\partial f}{\partial g} = \frac{\partial f}{\partial x} \cos \Phi + \frac{\partial f}{\partial y} \sin \Phi$$

et les équations 2.2 donnent :

$$\frac{\partial^2 f}{\partial g^2} = \frac{\partial^2 f}{\partial x^2} \cos^2 \Phi + 2 \cdot \frac{\partial^2 f}{\partial x \partial y} \cos \Phi \sin \Phi + \frac{\partial^2 f}{\partial y^2} \sin^2 \Phi$$

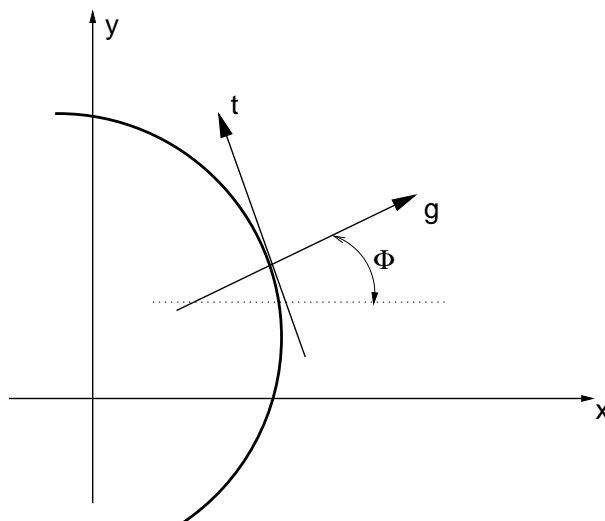


FIG. 2.2 – Repère local défini par le vecteur normal et le vecteur tangent à l'image.

Cette équation n'est pas plus simple à résoudre que la précédente. Cependant elle se relie assez aisément à l'équation du laplacien de f :

$$\Delta f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

qui s'exprime également (par invariance du laplacien par changement de repère euclidien) dans le repère local $\{t, g\}$:

$$\Delta f = \frac{\partial^2 f}{\partial g^2} + \frac{\partial^2 f}{\partial t^2}$$

On voit que si le terme $\frac{\partial^2 f}{\partial t^2}$ est négligeable (c'est-à-dire si le contour a une courbure très faible), alors :

$$\Delta f = 0 \quad \approx \quad \frac{\partial^2 f}{\partial g^2} = 0 \quad (2.5)$$

Cette propriété sera utilisée dans les approches dites par passage par zéro du laplacien (cf détecteur de Marr).

2.2 Les approches classiques

Cette section présente un ensemble de méthodes qui ont eu historiquement une grande importance en traitement des images. Bien que certaines soient encore régulièrement employées, un lecteur intéressé par les techniques plus actuelles pourra se rendre directement à la section 2.3.

2.2.1 Les détecteurs de gradient par filtrage

Ces détecteurs reposent tous sur une recherche d'un extremum de la dérivée première (ou d'un passage par zéro d'une dérivée seconde), celle-ci étant calculée de diverses manières, mais généralement par un filtrage passe-haut précédé d'un léger filtrage passe-bas pour s'affranchir des bruits. Ces approches par **filtrage linéaire** sur des critères très simples de sélection en fréquence ont reçu une base théorique dans [Modestino et Fries, 1977, Shanmugam et Green, 1979]. La somme des connaissances sur ce sujet se trouve dans [Torre et Poggio, 1986].

Ainsi, Modestino et Fries ont proposé de détecter les contours par un filtrage de Wiener, le but de ce filtrage étant d'effectuer sur l'image non bruitée une opération de laplacien qui sera suivie par la suite d'une détection de zéros allée à un seuillage (nous verrons ces étapes ultérieurement).

Le filtre de base de Modestino et Fries est un filtre récursif de la forme :

$$H_0(z_1, z_2) = \frac{\sum_i^n \sum_j^n b_{i,j} z_1^{-i} z_2^{-j}}{1 + \sum_i^n \sum_j^n a_{i,j} z_1^{-i} z_2^{-j}}$$

où les coefficients $a_{i,j}$ et $b_{i,j}$ sont déterminés à partir de 3 paramètres spécifiques de l'image à traiter : λ , ρ et ϵ . λ est le paramètre poissonnien de densité de contours (il exprime la distance moyenne entre 2 contours lors d'un balayage de l'image), ρ la constante de la loi exponentielle de l'auto-corrélation de l'image (cf. chapitre ??) et ϵ la puissance du bruit. Comme un tel filtre ne filtre que du coin supérieur gauche au coin inférieur droit, on lui associe les 3 autres filtres obtenus par rotation de $\pi/2$, pour obtenir un filtre final. Dans la pratique pour assurer des temps de calcul raisonnables, on choisit des filtres d'ordre 1 ($n = 1$).

2.2.2 Les détecteurs de gradient par masques

A côté de ces approches très inspirées du traitement du signal, des filtres de dérivation plus empiriques ont été proposés à partir d'estimateurs locaux de l'image f ou de ses dérivées $\frac{\partial f}{\partial x}$. Ces estimées sont obtenues à l'aide de masques (cf. figure 2.3) appliqués sur des fenêtres de 2×2 pixels ou 3×3 pixels (exceptionnellement, en cas d'images très bruitées, sur des fenêtres plus grandes). On note sans surprise que la somme des coefficients de ces filtres est nulle (fonction de transfert nulle à la fréquence 0), et que les coefficients sont anti-symétriques. Les filtres les plus utilisés sont, dans l'ordre décroissant :

Sobel > Roberts > Gradient > Prewitt

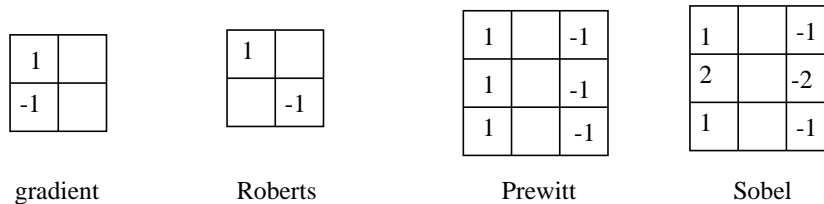


FIG. 2.3 – Quatre filtres de détection de contours par estimation du gradient. Les filtres représentés estiment une seule dérivée. Par rotation de $\pi/2$, on calcule la seconde dérivée.

Les filtres 3×3 sont un peu moins précis (c'est-à-dire que les contours qu'ils détectent sont moins bien localisés et souvent épais), mais les images ainsi obtenues sont généralement plus fiables et permettent des post-traitements plus poussés, ils sont également centrés sur un pixel et non entre des pixels (filtres à phase nulle).

La réponse de l'un quelconque de ces filtres s'obtient de la façon suivante : le filtre est centré en chaque pixel successivement (au coin supérieur gauche pour les filtres 2×2 , au point central pour les filtres 3×3). Le produit du masque par les valeurs des pixels correspondant étant fait, la valeur absolue de la somme est retenue. Puis le masque est tourné de 90° autour de son centre et la même mesure est répétée. Les deux mesures sont alors ajoutées et leur somme constitue la mesure du gradient en ce point selon la formule de l'équation 2.1, ou selon la formule approchée (en norme \mathcal{L}_1), un peu plus rapide à calculer :

$$\hat{G} = \left| \frac{\partial f}{\partial x} \right| + \left| \frac{\partial f}{\partial y} \right|.$$

On peut également mesurer l'orientation Φ du contour au point donné en faisant le rapport des réponses des deux filtres comme dans l'équation 2.4.

De tels filtres sont particulièrement simples de mise en œuvre, et rapides de calcul, permettent en particulier d'assurer des calculs en temps réel pour des applications industrielles.

2.2.3 Pré- et post traitements

La qualité de la détection est très liée à la qualité des contours dans l'image (cela sera vérifié pour toutes les méthodes, mais particulièrement pour ces méthodes peu élaborées). Ainsi, elles sont souvent bonnes pour les applications de robotique, en environnement artificiel, pour des images aériennes à forte résolution (en milieu urbain par exemple), elles se dégradent très vite pour les scènes d'extérieur, les images médicales, et deviennent totalement inapplicables en imagerie bruitée (thermographie, imagerie ultrasonore, radar SAR, etc.). On a alors intérêt à revenir à des démarches moins intuitives comme celles exposées dans [Modestino et Fries, 1977, Shanmugam et Green, 1979].

Ces détecteurs doivent toujours être suivis d'une étape de post-filtrage, et souvent précédés d'une étape de pré-filtrage.

Parmi les filtres de **pré-traitement**, on recherche ceux qui diminuent le bruit tout en préservant les discontinuités. Ils ont fait l'objet du chapitre ?? : filtre médian par exemple, mais aussi filtre toboggan [Fairfield, 1990] (cf. section 1.3.1), ou filtre de diffusion anisotrope de Perona et Malik [Perona et Malik, 1987] (cf. section 1.2.4). Ces filtres ont cependant le défaut de créer des contours artificiels qui peuvent être par la suite difficiles à éliminer.

Les **post-traitements** commencent le plus souvent par l'élimination des points de contour trop faibles par un seuillage (seuillage adaptatif si l'image est très hétérogène). On élimine ensuite les points qui ne sont pas des extrémums locaux dans la direction du gradient. Cela s'obtient à partir de la direction Φ mesurée précédemment et une comparaison simple des points rencontrés (ou interpolés si l'on souhaite une précision meilleure que le pixel). On peut opérer également des seuillages *par hystérésis*, afin de ne conserver que les composantes les plus importantes des contours. Pour cela, on procède à deux seuillages, le second étant très tolérant (c'est-à-dire laissant un très grand nombre de contours candidats) ; on ne garde des contours sélectionnés par ce second seuil que les points connexes d'un point de contour préservé par le premier (cela se fait aisément par un double étiquetage des images seuillées) cf. figure 2.4.

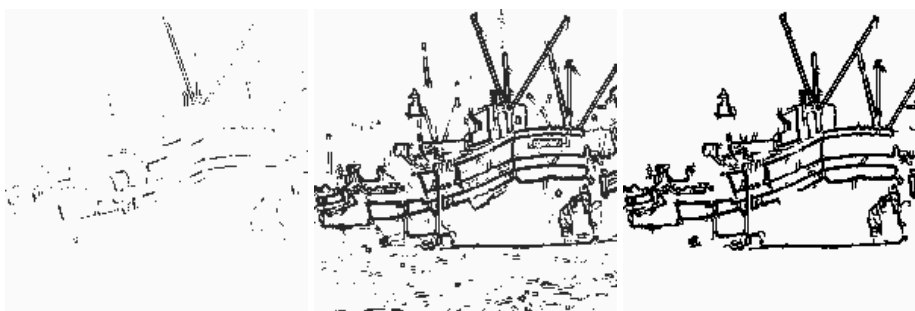


FIG. 2.4 – Filtrage de contours par hystérésis : à gauche, application d'un seuil sévère (70) au gradient de l'image bateau, seuls les contours les plus fiables sont détectés. Au centre, application d'un seuil tolérant (25), beaucoup de contours sont détectés, mais également beaucoup de bruit. A droite, résultat du filtre par hystérésis : seuls les contours du second filtrage connexes à ceux du premier sont conservés. Le détecteur de contours est un détecteur de Sobel. Aucun filtrage n'est appliqué pour réduire l'épaisseur des contours.

Enfin on procède à des étapes de **poursuite** et de **fermeture** des contours. La première opération a pour objectif de supprimer les petites disparitions de contours qui peuvent se produire par suite du bruit ou d'une perte de contraste, la seconde se propose de détecter des objets topologiquement clos, c'est-à-dire des zones fermées. Ces étapes seront vues plus en détail plus loin.

2.2.4 Les détecteurs de passage par zéro du laplacien

Ces méthodes ont été proposées en 1976 [Marr et Hildreth, 1980]. Elles ont eu une grande importance historique, étant considérées comme le prototype du détecteur de contour inspiré des systèmes biologiques (*primal sketch* de Marr). Elles utilisent le fait que le passage par zéro du laplacien permet de bien mettre en évidence les extrémums de la dérivée. Nous avons eu l'occasion de voir (cf. équation 2.5) qu'en l'absence de forte courbure, le passage par zéro du laplacien correspond en effet bien au maximum du gradient dans la direction du gradient. Ces méthodes tirent en outre profit du fait que les zéros de la dérivée seconde constituent un réseau de lignes fermées (évitant donc, en principe, les étapes de poursuite et de fermeture). Il en est de même du réseau des lignes de crête du gradient, mais le premier est plus aisément détecté à partir d'un simple étiquetage des zones positives et négatives. Mais l'estimation de la dérivée seconde étant très sensible aux bruits, il convient de filtrer très fortement l'image avant d'en mesurer le laplacien. Cela conduit au filtrage suivant :

$$\psi * \left(\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \right) = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) [\psi * f].$$

où ψ est un filtre passe-bas ; ce qui se récrit symboliquement :

$$image\ des\ contours = passage\ par\ zéro \left(f * \left[\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} \right] \right).$$

cela exprime qu'une image de contours est obtenue par filtrage de l'image par la dérivée seconde d'un filtre passe-bas, puis détection des zéros de la fonction ainsi obtenue.

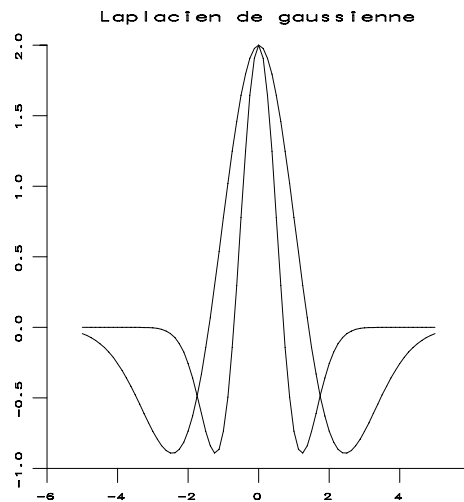


FIG. 2.5 – Filtre LOG pour deux valeurs différentes de σ .

Les filtres les plus utilisés pour ces filtrages passe-bas sont les filtres gaussiens [Marr et Hildreth, 1980]. Marr a montré en effet qu'avec de tels filtres on pouvait approcher de très près les effets donnés par le système visuel humain. Le filtre obtenu par convolution avec le laplacien d'une gaussienne est connu sous le nom de LOG (cf figure 2.5). Un autre filtre très utilisé et très proche du LOG est le DOG (*Difference of Gaussians*) (figure 2.6, qui procède par différence de deux gaussiennes dont les écarts-type sont dans un rapport 1,6 (pour simuler au mieux le système visuel). Les gaussiennes sont bien sûr affectées des facteurs adéquats pour que la somme des parties positives du filtre soit égale à la somme des parties négatives (valeur nulle à la fréquence 0 de la fonction

de transfert). Sa mise en œuvre peut tirer profit des diverses qualités de la gaussienne : séparabilité, noyau auto-reproducteur, limite de suites de polynômes, approximation par des splines, etc., ce qui permet d'en accélérer la réalisation.

Dans des approches très sommaires, le DOG est parfois remplacé par le DOB (*Difference of Box*), filtre très rapide même pour des grandes tailles (on tire profit alors du fait que les filtres de moyenne (*Box*) ont une complexité de calcul indépendante de la taille de la fenêtre d'intégration au prix d'un coup mémoire important). Les résultats de ces filtres est cependant généralement de piètre qualité.

Les filtres LOG et DOG, n'ont plus aujourd'hui qu'un intérêt historique. On leur préfère les filtres analytiques que nous verrons plus loin.

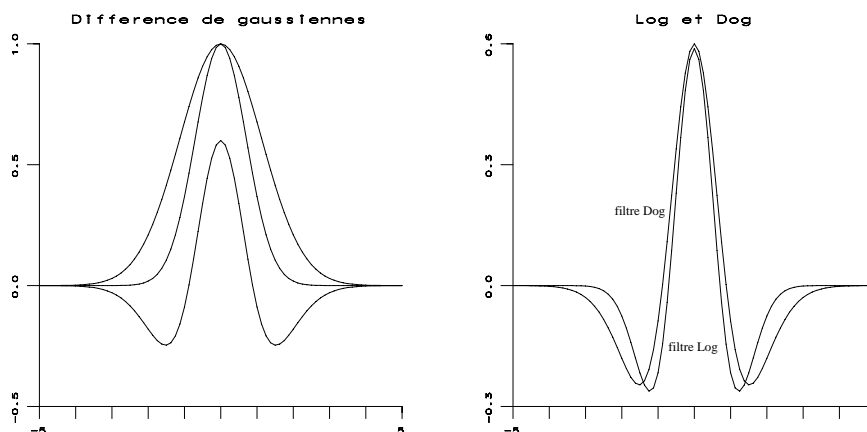


FIG. 2.6 – A gauche : filtre DOG et les deux gaussiennes dont il est la différence (leurs écarts-types sont dans un rapport 1,6). A droite filtre LOG et filtre DOG comparés.

2.2.5 Les détecteurs par masquage adapté

Dans ces techniques, on recherche en chaque point de l'image la présence d'une configuration conforme à un gabarit appartenant à un *dictionnaire de contours*. Pour cela, on définit une distance entre un contour type et une fenêtre de l'image. C'est par exemple l'inverse du produit scalaire des deux fenêtres : si l'image est dénotée $f(x, y)$, et le $k^{\text{ème}}$ gabarit $g_k(x, y)$, on calcule par exemple :

$$r_k(x, y) = \frac{\left(\sum_{x, y \in \text{fenêtre}} f^2(x, y) \cdot \sum_{x, y \in \text{fenêtre}} g_k^2(x, y) \right)^{1/2}}{\sum_{x, y \in \text{fenêtre}} f(x, y) g_k(x, y)}$$

pour toutes les valeurs de k . On ne conserve que la meilleure des valeurs et seulement si elle est suffisamment faible. L'approche par masques adaptés (*template matching*), permet de connaître aisément la direction du contour, elle permet également de soumettre, en parallèle, ces calculs à des architectures adaptées. Mais la qualité des contours, ainsi que les post-traitements qu'ils requièrent sont très semblables à ceux que l'on a vus pour les filtres de maximum du gradient.

Les filtres adaptés les plus utilisés sont le filtre de Kirsch (cf. tableau 2.1), le filtre de Nevatia et Babu (cf. tableau 2.2) et les boussoles directionnelles de Prewitt (cf. tableau 2.3).

Au-delà de ces approches souvent heuristiques, Hueckel a proposé une démarche beaucoup plus rationnelle, s'appuyant sur la décomposition du signal d'image sur une base de fonctions orthogonales en coordonnées polaires (polynômes d'Hermite utilisés également pour l'oscillateur harmonique), tronquée à ses 8 premiers termes

5	5	5
-3	0	-3
-3	-3	-3

-3	5	5
-3	0	5
-3	-3	-3

TAB. 2.1 – Masques de Kirsch : il y a 8 filtres issus de ces 2 fenêtres par rotation de $\pi/4$.

-100	-100	0	100	100
-100	-100	0	100	100
-100	-100	0	100	100
-100	-100	0	100	100
-100	-100	0	100	100

-100	32	100	100	100
-100	-78	92	100	100
-100	-100	0	100	100
-100	-100	-92	78	100
-100	-100	-100	32	100

TAB. 2.2 – Masques de Babu et Nevatia : il y a 12 filtres issus de ces 2 fenêtres par rotation, la première de $\pi/2$, la seconde de $\pi/4$.

[Hueckel, 1971]. Malheureusement les résultats n'ont pas été au niveau des investissements et cette voie est maintenant abandonnée.

2.3 Les approches analytiques

Nous allons voir maintenant une approche qui a permis une bien meilleure compréhension des conditions d'une bonne détection de contours et qui a ainsi conduit à des détecteurs de très bonne qualité. On les voit émerger dans les années 85, à partir des travaux : [Torre et Poggio, 1986, Shen et Castan, 1986, Canny, 1986].

2.3.1 Les critères de Canny

Canny, dans une approche originale, [Canny, 1986] a proposé un filtre déterminé analytiquement à partir de 3 critères :

1. garantir une bonne détection, c'est-à-dire une réponse forte même à de faibles contours,
2. garantir une bonne localisation,
3. assurer que pour un contour il n'y aura qu'une seule détection (éviter les effets de *rebonds* dus, par exemple, à la troncature des filtres).

Ces 3 critères s'expriment par l'optimisation conjointe de 3 fonctionnelles qui permettent de définir le **filtre linéaire optimal** pour la détection d'une **marche d'escalier** sous l'hypothèse d'un bruit additif indépendant du signal ¹.

Si l'on considère que le filtre a pour réponse impulsionnelle $\psi(x)$, ces fonctionnelles s'écrivent :

$$\text{bonne détection : } \Sigma = \frac{\int_0^{\infty} \psi(x) dx}{\sqrt{\int_{-\infty}^{\infty} \psi^2(x) dx}}$$

$$\text{bonne localisation : } \Lambda = \frac{|\psi'(0)|}{\sqrt{\int_{-\infty}^{\infty} \psi'^2(x) dx}}$$

¹Par rapport aux approches proposées précédemment, on note que le filtre reste linéaire, mais que le critère d'optimisation est plus complet.

1	1	1
1	-2	1
-1	-1	-1

1	1	1
-1	-2	1
-1	-1	1

TAB. 2.3 – Masques de boussole directionnelle : il y a 8 filtres issus de ces 2 fenêtres par rotation.

$$\text{réponse unique : } \frac{|\psi'(0)|}{\sqrt{\int_{-\infty}^{\infty} \psi'^2(x) dx}} = k \frac{\int_{-\infty}^0 \psi(x) dx}{\sqrt{\int_{-\infty}^{\infty} \psi^2(x) dx}}$$

La résolution du système est assez complexe (on maximise le produit $\Sigma\Lambda$ sous la contrainte du $3^{ème}$ terme constant). Par ailleurs, Canny souhaite obtenir un filtre monodimensionnel (dans la direction orthogonale au contour), à réponse impulsionnelle finie (RIF) ; celui-ci a alors une expression complexe, composée de 4 termes, chacun combinant des lignes exponentielles et trigonométriques :

$$\psi(x) = a_1 e^{x/\sigma} \sin \omega x + a_2 e^{x/\sigma} \cos \omega x + a_3 e^{-x/\sigma} \sin \omega x + a_4 e^{-x/\sigma} \cos \omega x$$

où les coefficients a_i et ω sont déterminés à partir de la taille du filtre. Le paramètre σ est un paramètre de grande importance que nous retrouverons dans tous les autres filtres dérivés de l'approche de Canny. C'est un paramètre d'échelle qui indique en-deça de quelle distance deux contours parallèles seront confondus en un seul. Canny montre que la dérivée d'une gaussienne est une bonne approximation de son filtre.

$$\psi(x) \approx -x \exp\left(-\frac{x^2}{2\sigma^2}\right)$$

C'est donc la réduction à 1 dimension du filtre proposé par Marr à 2 dimensions (le maximum de la dérivée de la gaussienne est obtenu pour le passage par zéro du laplacien de gaussienne). Ce filtre donne des résultats de bonne qualité. Le critère de Canny (critère $\Sigma\Lambda$) vaut 1,12 pour le filtre RIF optimal, 0,97 pour le filtre gaussien. Remarquons cependant que le filtre gaussien n'est pas RIF.

2.3.2 Les filtres de Deriche et Shen et Castan

Au filtre de Canny, on préfère souvent le détecteur de Deriche, qui répond exactement aux mêmes critères de qualité que celui de Canny, mais qui possède une réponse impulsionnelle infinie (filtre RII). Il a pu donc être synthétisé de façon récursive particulièrement efficace [Deriche, 1987]. Le filtre de Deriche a une expression générale de la forme :

$$\psi(x) = -cx \exp(-\alpha|x|),$$

avec :

$$c = \frac{[1 - \exp(-\alpha)]^2}{\exp(-\alpha)}.$$

avec les critères même de Canny il est supérieur au filtre de Canny ($\Sigma\Lambda = 2$). Le paramètre α de Deriche représente alors l'inverse de l'écart type σ de la gaussienne du filtrage de Canny ($\alpha = \frac{\sqrt{\pi}}{\sigma}$). Le filtre de Deriche s'écrit de façon récursive en fonction des valeurs $f(x)$ de l'image, par l'intermédiaire de 2 filtres l'un décrivant l'image de gauche à droite et l'autre de droite à gauche :

$$z^+(x) = c\epsilon + 2\epsilon z^+(x-1) - \epsilon^2 z^+(x-2)$$

$$z^-(x) = -c\epsilon + 2\epsilon z^-(x+1) - \epsilon^2 z^-(x+2)$$

$$z(x) = z^+(x) + z^-(x)$$

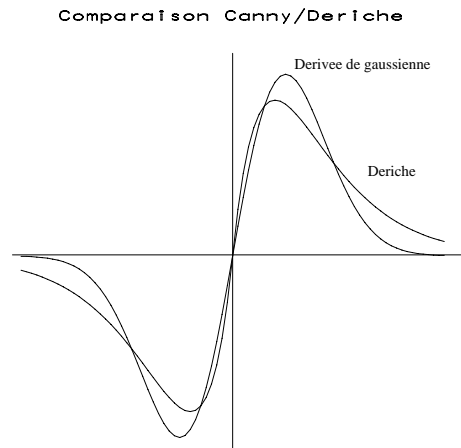


FIG. 2.7 – Le filtre de Deriche : $x \cdot \exp(-|x|)$ et la dérivée de la gaussienne : $\frac{x}{\sqrt{\pi}} \cdot \exp(-x^2/\pi)$.

avec :

$$\epsilon = \exp(-\alpha)$$

Le filtre de Shen et Castan [Shen et Castan, 1986] possède également de très bonnes qualités de détection, il s'apparente à la même famille des filtres exponentiels (mais comme il n'est pas dérivable à l'origine, il ne se prête pas à une comparaison avec le critère de Canny) :

$$f(x) = c \cdot \text{sgn}(x) \exp(-\alpha|x|)$$

Le filtre de Shen et Castan, en raison de sa forme à l'origine, donne une localisation très précise des contours, mais il est sensible aux bruits.

Dans [Demigny et Kamlé, 1997], Demigny a étendu les critères d'optimalité des filtres de Canny à des images discrètes. Il les a ensuite étendus à des profils en rampe et non plus en marche d'escalier.

2.3.3 L'extension à 2D

Tous ces filtres sont mono-dimensionnels, le filtrage 2D est donc obtenu par l'action de deux filtres croisés, l'un en x , l'autre en y .

Pour définir l'un de ces deux filtres, lorsque l'on a choisi le gabarit du filtre dans la direction perpendiculaire au contour, afin de mieux intégrer le signal, on utilise, dans la direction du contour, une **fonction de prolongement** qui permet de filtrer les bruits. C'est souvent une gaussienne de même écart-type que celle qui est dérivée dans le détecteur. Par exemple la dérivée en x pourra être estimée par convolution de l'image par :

$$\psi(x, y) = -x \exp(-(\alpha x^2)) \exp(-(\alpha y^2))$$

Ces filtres ne sont donc généralement pas isotropes. C'est-à-dire qu'ils ne répondent pas de façon identique à des contours identiques mais d'orientations différentes. Les seules fonctions séparables et isotropes sont les fonctions gaussiennes. Mais même si l'on utilise des filtrages gaussiens, la mise en œuvre discrète du filtre sur une maille carrée rend les filtres anisotropes. Des travaux importants ont été consacrés à la réduction de cet effet [Kundur et al., 1999] utilisant une combinaison de deux détecteurs à $\pi/4$ l'un de l'autre.

En conclusion, dans une approche moderne de la détection des contours, un filtre de détection se compose de deux estimateurs de dérivées, l'un selon Ox , l'autre selon Oy . L'un de ces détecteurs (choisissons celui selon Ox) se compose du produit de 2 fonctions :

- selon Oy c'est une fonction passe-bas, symétrique, (la fonction de prolongement) dont l'étendue est fonction de l'importance du bruit et de la distance que l'on souhaite respecter entre deux contours ;
- selon Ox c'est une fonction passe-haut (anti-symétrique), souvent égale à la dérivée de la précédente.

Les couples « fonctions de prolongement - dérivées » sont typiquement issus de la gaussienne (Canny), de l'exponentielle décroissante (Shen et Castan) ou du produit de l'exponentielle par x (Deriche) (cf. figure 2.8), mais on peut également les concevoir très librement pour une application spécifique.

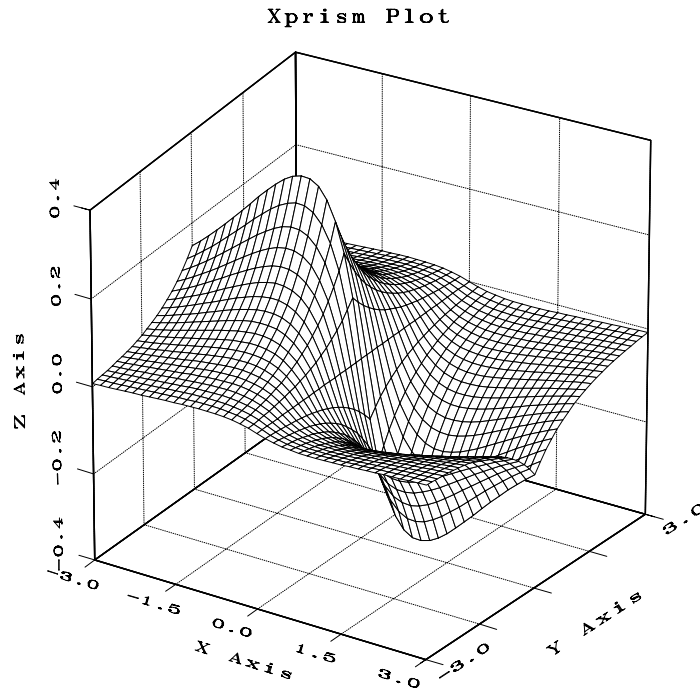


FIG. 2.8 – Profil général d'un filtre d'estimation de la dérivée dans la direction Ox . Selon Oy le filtre est un intégrateur (ici $\exp(-|y|)$), selon Ox , c'est un dérivateur (ici $x.\exp(-|x|)$).

2.3.4 Variantes autour des filtres analytiques

D'autres filtres aux performances comparables, mais résultats d'optimisations différentes, ont été proposés dans :

- [Spacek, 1986] : On modifie le critère de Canny, et on cherche pour filtre une spline cubique de forme $x^3 - 2x^2 + x$.
- [Petrou et Kittler, 1991] : On considère alors que le contour est également le résultat d'un flou, ce n'est donc plus une marche, mais une rampe. Les critères de sélection sont ceux de Spacek et le filtre est une combinaison non-linéaire de lignes trigonométriques et d'exponentielles décroissantes.
- [Sarkar et Boyer, 1991] : On modifie dans ce cas les critères de Canny, on adopte une approche variationnelle et on choisit la représentation récursive d'un filtre RII comme Deriche.

Une excellente comparaison et analyse des filtres de détection des contours est donnée dans [Cocquerez et Philipp, 1996].

Utilisant les résultats précédents ainsi que des méthodes éprouvées d'optimisation semi-globale développées dans les contours actifs (cf. section 2.4), des solutions variationnelles ont également été proposées qui sont fort efficaces et permettent d'ajouter des informations connues a priori sur l'allure des contours lors de leur détection. Voir par exemple [Fua et Leclerc, 1990].

2.4 Les contours actifs = les *snakes*

2.4.1 Une approche physique des contours

Une approche très différente des méthodes antérieures de détection de contours a été proposée en 1987 par Kass, Witkins et Terzopoulos [Kass et al., 1988] (voir aussi [Blake et Isard, 1998]), appelée contours actifs ou *snakes*. Il s'agit d'une méthode semi-interactive dans laquelle l'opérateur place dans l'image, au voisinage de la forme à détecter, une ligne initiale de contour. Cette ligne sera amenée à se déformer sous l'action de plusieurs forces :

- une énergie propre, assimilée à l'énergie mécanique de tension et de torsion d'une ligne matérielle,
- une énergie potentielle imposée par l'image qui vise à plaquer la courbe sur les contours,
- une énergie externe, introduite par l'utilisateur pour traduire les contraintes spécifiques du problème qu'il se pose.

Sous ces énergies, le contour *actif* va évoluer pour rechercher la position d'énergie minimale, qui sera ainsi un compromis entre les diverses contraintes du problème.

L'écriture formelle du problème passe par la définition paramétrique du contour, en fonction d'une variable s généralement l'abscisse curviligne :

$$v(s) = [x(s), y(s)]^t \quad s \in [0, 1]$$

$$E_{totale} = \int_0^1 [E_{interne}(v(s)) + E_{image}(v(s)) + E_{externe}(v(s))] ds \quad (2.6)$$

avec :

$$E_{interne} = \alpha(s) \left(\frac{dv}{ds} \right)^2 + \beta(s) \left(\frac{d^2v}{ds^2} \right)^2$$

où :

- la première dérivée prend en compte les variations de longueur de la courbe, (c'est donc un terme de tension (résistance à la rupture), qui est contrôlé par l'élasticité que l'on attribue au contour),
- tandis que la seconde exprime les variations de la courbure (c'est un terme de flexion contrôlé par la raideur du contour).

Ces deux termes agiront donc pour produire une courbe régulière (cf. figure 2.9).

Le second terme d'énergie : E_{image} caractérise les lignes que l'on souhaite suivre. Dans le cas de détection de contours, ce sont des lignes de fort gradient, il vaut donc généralement :

$$E_{image} = -\nabla f$$

mais il peut être adapté pour suivre les maximums des niveaux de gris (dans ce cas $E_{image} = f$), ou toute autre fonction définie à partir de ceux-ci.

Enfin le dernier terme d'énergie : $E_{externe}$ est choisi par l'utilisateur. Il peut avoir des formes très variées afin, par exemple, de contraindre le contour à ressembler à un gabarit donné, à s'approcher d'un contour déjà détecté sur une autre image (suivi de séquences, ou images à 3D)².

²On notera cependant que, dans le cas de volumes numériques tridimensionnels $f(x, y, z)$, il est possible d'étendre les contours actifs à des surfaces actives pour lesquelles l'énergie s'exprime comme pour des plaques minces.

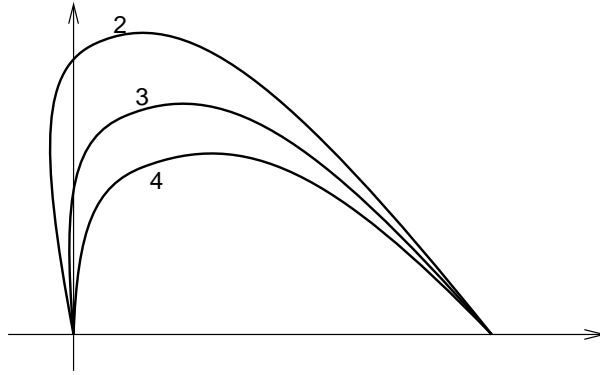


FIG. 2.9 – Déformation d'un contour actif à extrémités fixes pour des valeurs du rapport α/β (élasticité/raideur) égales à 2, 3 et 4, en l'absence d'une énergie liée à l'image.

En l'absence de ce dernier terme, et dans les nombreux cas où l'on recherche les lignes de fort gradient, l'équation 2.6 s'écrit alors :

$$E_{totale} = \int_0^1 \left[-\nabla f + \alpha(s) \left(\frac{dv}{ds} \right)^2 + \beta(s) \left(\frac{d^2v}{ds^2} \right)^2 \right] ds \quad (2.7)$$

2.4.2 Mise en équation des contours actifs

L'équation intégrale 2.7 est résolue généralement de façon variationnelle. On suppose que le contour évolue vers un minimum d'énergie, soit vers un zéro de $\frac{\partial E_{totale}}{\partial t}$. En désignant par v' et v'' les dérivées de v le long de la courbe, on obtient une équation différentielle vectorielle :

$$\gamma \frac{\partial v}{\partial t} + (\alpha v')' - (\beta v'')'' = \frac{\partial |\nabla f|^2}{\partial v}$$

où α , β et γ sont potentiellement variables le long de s . Il y a plusieurs façons de concevoir la discrétisation de la courbe :

1. selon les différences finies : les éléments de la courbe sont réduits en des points auxquels sont attachés les éléments mécaniques (masse, raideur, etc.) de la courbe considérée concentrée en ces points ;
2. selon les éléments finis on remplace chaque portion de courbe par le segment élémentaire et les éléments mécaniques sont calculés sur ces segments.

Nous ne décrivons ici que la première approche la plus fréquemment employée. Après discrétisation de la courbe en un nombre n de points et en posant :

$$V^t = [v_0^t, v_1^t, v_2^t, \dots, v_{n-1}^t]^t$$

et :

$$h_v = -|\nabla f(v)|^2$$

on obtient une équation matricielle de la forme :

$$(A + \gamma I)V^t = \gamma V^{t-1} - h_v(V^{t-1})$$

où A est une matrice pentadiagonale³ (figure 2.11) de taille $n \times n$ fonction de α et β , I est la matrice unité de taille $n \times n$ et γ exprime une *inertie* de la courbe aux déplacements. Sa résolution donne :

$$V^t = (A + \gamma I)^{-1}(\gamma V^{t-1} - h_v(V^{t-1}))$$

Si les paramètres α , β et γ sont invariants le long de la courbe, il suffit alors de calculer une seule fois $(A + \gamma I)^{-1}$ pour résoudre le système pour tous les temps t . Sinon, on inverse la matrice à chaque instant. Les choix des paramètres α , β et γ qui ne sont pas dictés par le problème demeurent souvent délicats pour garantir une convergence convenable, ainsi que le choix de l'estimateur de ∇f . L'utilisation d'un estimateur par filtre de Deriche est parfois préconisée.

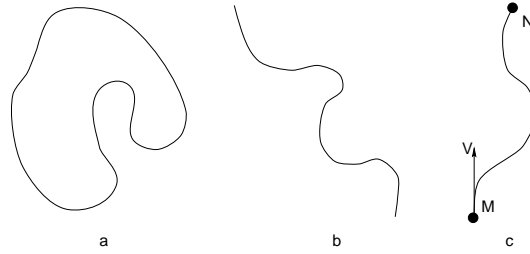


FIG. 2.10 – Contour actif fermé en a, ouvert à extrémités libres en b et ouvert à extrémités fixes en c (on impose en M une extrémité fixe et une orientation fixe donnée par le vecteur V).

$$\begin{bmatrix} 2\alpha+6\beta & -\alpha-4\beta & \beta & 0 & 0 & \dots & \dots \\ -\alpha-4\beta & 2\alpha+6\beta & -\alpha-4\beta & \beta & 0 & \dots & \dots \\ \beta & -\alpha-4\beta & 2\alpha+6\beta & -\alpha-4\beta & \beta & & \\ 0 & \beta & -\alpha-4\beta & 2\alpha+6\beta & -\alpha-4\beta & & \\ 0 & 0 & \beta & -\alpha-4\beta & 2\alpha+6\beta & & \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix}$$

FIG. 2.11 – Un contour actif fermé est représenté par une matrice circulante pentadiagonale.

Malgré les difficultés de réglage de la convergence, les contours actifs apportent une solution heureuse à la détection de contour, intermédiaire entre les solutions purement locales (opérateurs de type Sobel) et les segmentations globales. Trois types de contours actifs différents sont utilisés (cf. figure 2.10) :

1. les contours actifs fermés (où $v_0^t = v_{n-1}^t$),
2. les contours actifs à extrémités libres,
3. les contours actifs à extrémités fixes (où les positions v_0 et v_{n-1} sont fixes dans le temps, ainsi que, potentiellement les premières dérivées en ces points).

Selon le type de contours actifs, les matrices A présentent la propriété d'être circulante (contours actifs fermés), Töplitz (contours actifs à extrémités libres) ou quelconques (figure 2.11).

³Dans le cas d'une discrétisation par éléments finis, la matrice est heptadiagonale

Laisser libre d'évoluer seul, en l'absence de forces d'attraction dues au gradient dans l'image, le contour actif a tendance à se réduire à un point s'il est fermé, à une droite s'il est à extrémités libres. Pour compenser cette tendance, on est parfois amené à introduire des forces internes de *gonflage*. On a alors affaire à des *ballons* [Cohen, 1991]. Comme les équations des forces de gonflage ne dérivent plus d'un potentiel, les écritures s'en trouvent un peu complexifiées.

Des solutions élégantes ont également été proposées pour rechercher la meilleure position du snake en en déplaçant ses nœuds sur toute la grille discrète de l'image par une technique de programmation dynamique [Amini et al., 1990] (cf. figure 2.12).

Par ailleurs, le couplage de techniques de contours actifs avec des méthodes s'appuyant sur des propriétés statistiques distinguant la forme de son fond, et optimisées par un maximum de vraisemblance au sens bayésien, a montré sa très grande efficacité, permettant de faire le lien entre techniques de contours et de régions [Chesnaud et al., 1999].

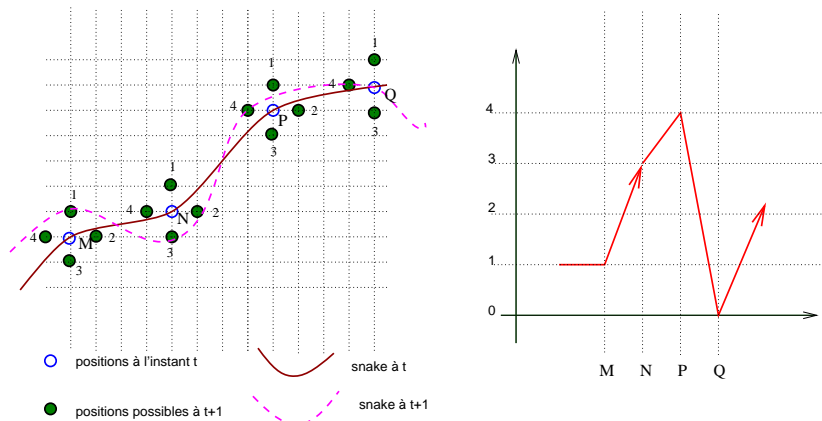


FIG. 2.12 – Dans la technique d'optimisation des contours actifs par programmation dynamique, chaque nœud M, N, P, Q peut se déplacer dans l'un des nœuds 1, 2, 3, ou 4, ou rester en place (position 0). L'optimisation se fait dans le graphe de droite par propagation de gauche à droite.

2.4.3 Les ensembles de niveaux (*level sets*)

Ce sont également des représentation variationnelles des contours qui conduisent donc à des solutions qui évoluent au cours du temps dans l'image, régies par un critère global. Ces modèles ont cependant le mérite remarquable de pouvoir changer de topologie si les contours l'imposent : par exemple un contour simple peut évoluer en deux contours séparés, ou, à l'inverse, deux contours séparés se réunir en un seul contour. Cela est rendu possible par l'utilisation de fonctions d'une dimension supérieure à celle des contours recherchés (si l'on cherche des lignes de contours, on introduira pour inconnue une surface de \mathbf{R}^3 , si on cherche des surfaces de \mathbf{R}^3 , on introduit des volumes de \mathbf{R}^4). Le contour est alors défini comme l'ensemble de niveau zéro (*level sets*) de cette fonction : $v(s)$ est une ligne de niveau $z = 0$ de la surface $z = f(x, y)$, souvent prise comme la distance au contour [Osher et Sethian, 1988, Sethian, 1996].

Sur le contour :

$$\frac{dz}{ds} = 0 = \frac{\partial z}{\partial x} \frac{dx(s)}{ds} + \frac{\partial z}{\partial y} \frac{dy(s)}{ds}$$

et la courbe de niveau évolue en fonction des itérations t selon l'équation fondamentale :

$$\frac{\partial z}{\partial t} = \|\text{grad}(z)\| \vec{N} \wedge \frac{d\vec{v}}{dt}$$

où \vec{N} représente la normale au contour v .

Des équations semblables existent à 3D [Zhao et al., 2000].

2.4.4 Les modèles géodésiques actifs

Les approches par contours actifs peuvent être reliées à la famille des problèmes d'optimisation géodésique dans des espaces dont la géométrie est fixée par la dynamique de l'image (cf. chapitre 3.5.1). Ces techniques, appelées modèles géodésiques actifs, ont été proposées dans [Caselles et al., 1997] et le lien avec les contours actifs y est discuté.

Le problème du snake, s'il n'y a pas d'élasticité, peut s'écrire comme la minimisation de l'énergie :

$$J_1(v) = \int_a^b |v'(s)| ds + \lambda \int_a^b \psi(|\nabla f(v(s))|)^2 ds$$

où ψ représente le détecteur de contour. Caselles a introduit une fonctionnelle du type :

$$J_2(v) = 2\sqrt{\lambda\alpha} \int_a^b |v'(s)| \psi(|\nabla f(v(s))|) ds$$

Minimiser J_2 revient à trouver le chemin de longueur minimale dans l'espace riemannien dont la métrique est induite par l'image. On a pu montrer un certain nombre d'équivalences entre les propriétés des minimums de J_1 et de J_2 dans [Aubert et Blanc-Féraud, 1999].

Les contours géodésiques utilisent des solutions proches de celles des ensembles de niveaux pour le codage et la représentation de la ligne de contours.

2.5 La poursuite et la fermeture des contours

Si l'on excepte les contours actifs et les passages par zéro des laplaciens, la plupart des détecteurs de contours fournissent des contours ouverts, c'est-à-dire qu'ils ne séparent pas les composantes de l'image dans des objets topologiquement distincts. Cela rend souvent plus difficile l'étape de reconnaissance des formes qui doit suivre. Pour cela, on a proposé plusieurs méthodes permettant d'obtenir des contours clos. C'est ce que l'on appelle la **fermeture de contours**.

De nombreuses solutions ont été proposées pour cette étape fondamentale, reposant sur des principes très différents (les contours actifs à extrémités fixes que nous venons de voir en sont un exemple). La qualité des résultats obtenus est généralement directement liée au coût informatique consenti, de très bonnes solutions existant, mais à des coûts prohibitifs pour la plupart des applications.

2.5.1 Méthodes de recherche dans des graphes

Ce sont les méthodes qui reposent sur les bases les plus solides et susceptibles de donner les meilleurs résultats, mais leurs coûts sont souvent très élevés, car elles se proposent d'explorer l'espace de toutes les solutions et de retenir la meilleure en fonction d'un critère que l'opérateur s'est donné. L'écueil de la recherche combinatoire exhaustive peut être évité par des recherches polynomiales assez complexes, surtout fondées sur la programmation dynamique.

La technique de base, présentée par Martelli [Martelli, 1972] et Montanari [Montanari, 1971], est celle d'une optimisation combinatoire minimisant la fonction de coût (ou maximisant une fonction de mérite). Les algorithmes de programmation dynamique (Viterbi, A^*), sont particulièrement bien adaptés à ce problème, mais, malgré leur

complexité polynomiale sont généralement limités par des profondeurs de recherche de quelques pixels si le graphe est la totalité de l'image.

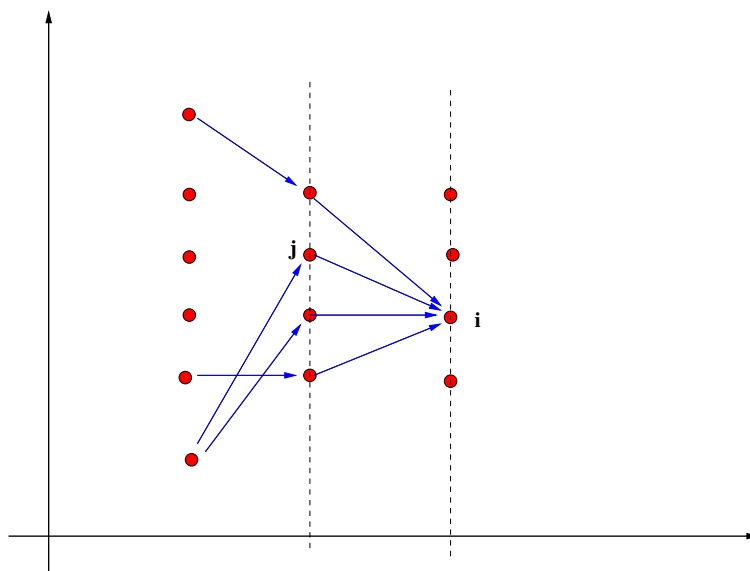


FIG. 2.13 – Résolution d'un problème de recherche de chemin optimal dans un graphe par programmation dynamique : la propagation se fait de la gauche vers la droite. On trouve le meilleur chemin arrivant à l'état i en optimisant la fonction de coût sur tous les états j qui ont été calculés à l'étape antérieure.

Dans une approche de type Viterbi, on considère l'évolution pas à pas d'un contour. À chaque pas le coût du contour est évalué pour toutes les provenances possibles du contour et n'est retenu pour tout point du parcours que le seul chemin qui minimise le coût pour arriver à ce point (cf. figure 2.13). On garde alors en chaque point ce coût minimum ainsi que l'adresse du *père* qui a permis d'y aboutir. Un nouveau pas est alors fait et toutes les destinations sont explorées que l'on peut atteindre en un pas à partir des points déjà atteints. La fonction de coût se calcule de la façon suivante :

$$\Gamma(i) = \gamma(i) + \min_{j \in \text{antécédents}} [\Gamma(j) + \delta(i, j)] = \gamma(i) + \epsilon(i)$$

où $\gamma(i)$ est inversement proportionnel à la qualité du point i comme point de contour et $\delta(i, j)$ exprime l'incompatibilité de i et j le long d'un contour. Le terme γ est un terme d'innovation le long du chemin tandis que ϵ est un terme de mémoire. La pondération de ces deux termes est un problème délicat en programmation dynamique.

Un contour est fermé lorsque le point i est confondu avec un point d'un autre contour. Le coût de la fermeture est alors l'intégrale le long de la chaîne des coûts de liaison δ et des coûts individuels γ . Pour éviter de trop pénaliser les chaînes longues, il est possible de calculer le *coût moyen* le long du parcours en pondérant le terme de mémoire ϵ par le chemin parcouru.

En limitant les domaines de recherche à des champs convexes, ou plus généralement en introduisant des limitations sur les types de chemin que l'on s'autorise à trouver, on peut étendre notablement la portée de ces fermetures (cf. figure 2.14). Des approches multi-résolutions permettent de travailler de façon hiérarchique, d'une fermeture grossière à une fermeture précise, en limitant progressivement le domaine de recherche à une bande étroite autour du contour détecté à l'étape précédente. Dans d'autres cas, on se limite à des horizons faibles pour entreprendre une recherche par programmation dynamique qui est validée, puis relancée à partir du résultat précédent.

Dans une recherche de type A^* , on dispose d'une estimée de la fonction de coût ($\hat{\Gamma} > \Gamma_{opt}$) et la fonction de coût doit être croissante le long d'une fermeture. On fait une recherche en largeur d'abord en comparant à chaque

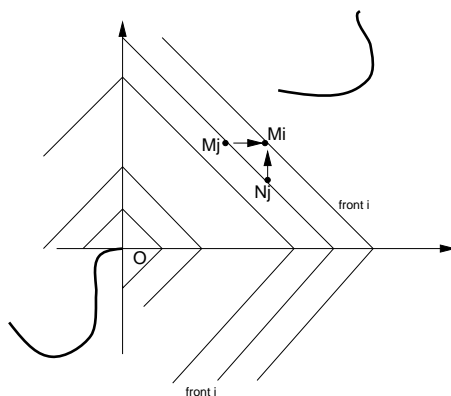


FIG. 2.14 – Exemple d’une fermeture de contours par recherche d’un chemin optimal dans un graphe. Le contour à fermer s’arrête en O . Les horizons de recherche à chaque étape de la programmation dynamique sont des carrés (fronts d’ordre i), centrés en O . Chaque point M_i peut avoir 2 antécédents (M_j et N_j). Une fermeture s’obtient lorsqu’un des points appartient à un autre contour. Pour éviter que le contour ne se ferme trop rapidement sur lui-même (près de O), il faut invalider les points du contour lui-même. Une telle propagation contraint les contours à une courbure toujours de même signe.

nœud la valeur locale du coût à cette estimée et on invalide le nœud si le coût dépasse l’estimée. On a tout intérêt à trouver donc une estimée très proche du coût optimal réel.

Les fonctions de coût sont les points déterminants de la qualité de la méthode. Cette fonction de coût doit être locale pour préserver le caractère markovien nécessaire à la programmation dynamique (le coût en un point ne dépend que de γ et du coût au point précédent), mais elle doit également incorporer toutes les informations que l’on connaît sur un *bon contour* : contraste entre les deux plages séparées par le contour bien sûr, mais aussi, longueur et courbure, voire écart à un contour modèle dans certains cas. C’est dans cette fonction de coût que l’on rajoute l’information qui a manqué au détecteur local placé en amont pour détecter le contour.

2.5.2 Les automates

Ces solutions, au contraire des précédentes, ne proposent que des solutions très sous-optimales, fondées sur de seuls critères locaux (il n’y a pas d’optimum global comme par programmation dynamique). Elles sont particulièrement rapides de mise en œuvre [Giraudon, 1987], mais peuvent parfois faire diverger la combinatoire si l’on gère mal la prolifération des chemins par ramification. Les résultats d’une détection par masquage adapté sont les candidats idéaux à ces techniques de fermeture puisqu’ils fournissent des directions de recherche.

Les diverses méthodes se distinguent par le choix des successeurs d’un point (parmi des configurations prédéfinies dans [Robinson, 1977, Cocquerez et Philipp, 1996, Nevatia et Babu, 1980], par prédiction de type Kalman) et le mode de balayage de l’image (par exemple 1 balayage dans [Nevatia et Babu, 1980], 2 balayages, l’un horizontal l’autre vertical dans [Cocquerez et Philipp, 1996], traitement de listes chaînées dans [Giraudon, 1987]). Il est également important de définir des fonctions de mérite pour valider les chaînes ainsi trouvées, car les automates, comme la programmation dynamique, trouvent toujours un contour, mais ne garantissent pas toujours qu’il correspond véritablement au contenu de l’image.

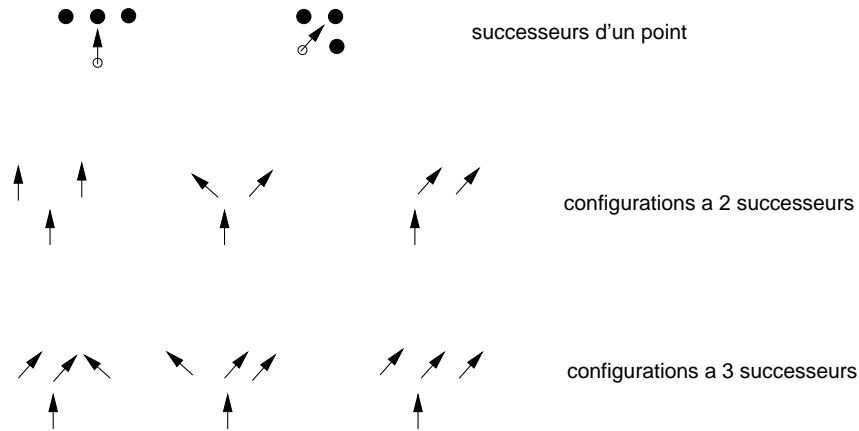


FIG. 2.15 – Configurations retenues par l’automate de Nevatia et Babu pour orienter un contour vers 3 successeurs au plus.

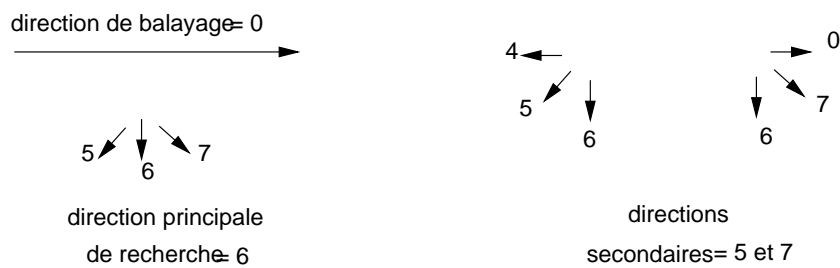


FIG. 2.16 – L’automate de Cocquerez lors d’un balayage horizontal utilise les directions 5, 6 et 7 de Freeman (cf. chapitre 5.4) comme directions principales. Lorsque l’une d’elle a été retenue, les directions adjacentes (par exemple le 4 et le 6 pour la direction principale 5) sont utilisées pour élargir la recherche. Cocquerez propose de faire 2 balayages successifs, l’un horizontal, l’autre vertical.

