



CLASSIFICATION HIERARCHIQUE ET CONNEXITE ADAPTEES A LA SEGMENTATION D'IMAGES

Lise BRENAC, Jacques LEMAIRE

Laboratoire I3S C.N.R.S. - U.R.A. 1376 Bât. 4 - 250, av. A. Einstein
Sophia Antipolis 06560 Valbonne

RESUME

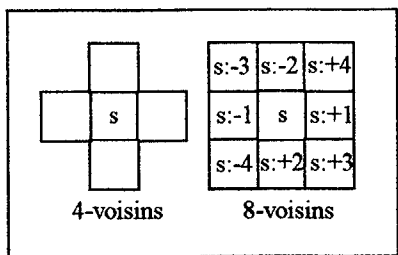
La segmentation d'une image consiste à rechercher une partition optimale de l'ensemble des sites d'une image, de manière à regrouper dans une même région des sites ayant des caractéristiques voisines. L'algorithme de Koepfler associe au principe classique de croissance de régions, celui de la minimisation d'une énergie. Dans cet article, nous en proposons des améliorations pour calculer l'estimation initiale, pour optimiser le critère utilisé, qui sera également modifié pour prendre en compte des quantités de dimension identique, ce qui n'est pas le cas de l'inertie et de la longueur.

ABSTRACT

Image segmentation consists in finding optimal partition of images, and grouping same-characteristics pixels in regions. Koepfler proposed an interesting algorithm, because the classical principle of regions merging is associated to an energy minimization. In this paper, different improvements of this algorithm are presented and concerned initial estimation, criterion optimization. It will also be modified to take in account same dimension quantities.

1. FORMULATION DU PROBLEME

Une image est définie sur une grille rectangulaire, composée de A colonnes et B lignes. L'intersection d'une ligne et d'une colonne constitue un site s, et la valeur observée en ce site, l'information décrivant ce site. On note S, l'ensemble fini des sites, de cardinal M : $S = \{0, \dots, A-1\} \times \{0, \dots, B-1\}$. On définit sur cet ensemble (en supposant que $S \subset \mathbb{Z}^2$), des systèmes de 4 ou 8-voisinages, avec les notations indiquées dans le schéma ci-contre pour désigner leurs éléments. Généralement, on considère comme information, le niveau de gris, que l'on note o_s .



L'image est donc définie par une famille $o = (o_s)_{s \in S}$ d'observations à valeurs dans \mathcal{O} . \mathcal{O} est égal par exemple à $\{0,1\}$ pour des images binaires, $\{0, \dots, 255\}$ pour des images à 256 niveaux de gris, \mathbb{R} pour des images réelles, \mathbb{C} pour des images complexes... Une sous-famille $(o_s)_{s \in T}$ sera notée o_T , lorsque T est contenu dans S.

La segmentation d'une image [Mong90] consiste à rechercher une partition de l'ensemble S des sites de l'image, en K classes connexes (appelées régions), sans connaître K a priori, de manière à regrouper dans une même région, des sites ayant des caractéristiques voisines.

Une classe $C_k \subset S$ est connexe si :

$$\forall s, s' \in C_k \exists s_0, \dots, s_n \in C_k \forall i (s_i, s_{i+1} \text{ 4-voisins}) \text{ et } (s_0 = s) \text{ et } (s_n = s')$$

On se propose de rechercher ces K régions, notées C_0, \dots, C_{K-1} , en considérant une mesure U d'homogénéité des régions que l'on décomposera en :

- une mesure d'homogénéité des observations, par la prise en compte de la notion d'inertie d'une région ;
- une mesure de la "rondure" des régions, par la prise en compte de la notion de bord.

Les régions seront définies par la donnée d'une nouvelle famille

$e = (e_s)_{s \in S}$ à valeurs dans \mathcal{E} , où \mathcal{E} est un ensemble fini d'étiquettes égal à $\{0, \dots, K-1\}$ et où :

$$\forall k \in \mathcal{E}, \forall s \in S \quad e_s = k \Leftrightarrow s \in C_k$$

Le problème s'écrit donc :

$$\min_{C_0, \dots, C_{K-1}} U(C)$$

où C est une partition de S en K régions C_0, \dots, C_{K-1} .

En matière de segmentation d'images, on peut classer les différentes méthodes proposées en trois grandes catégories : l'approche par régions, l'approche par contours et l'approche mixte qui combine les deux précédentes. Celle qui nous intéresse, la première, peut encore se diviser en deux sous-catégories : les algorithmes non hiérarchiques, type SA (Simulated Annealing), ICM (Iterate Conditional Mode) ([Gem84], [Besa86], [Dub89]), basés sur une modélisation markovienne, associée à des critères de classification bayésienne tels que le MAP (Maximum A Posteriori), le MPM (Maximum of Posterior Marginal)... Ces algorithmes résolvent assez bien des problèmes d'optimisation du type (1), même s'ils comportent des hypothèses parfois contraignantes liées au choix du modèle [BrLP93]. L'autre sous-catégorie concerne les algorithmes hiérarchiques, qui peuvent être ascendants [Koep88], descendants ([PerH91], [BouL91]) ou mixtes du type "split and



merge" [Pav182]. Les algorithmes de [PerH91] et [BouL91] sont une adaptation hiérarchique d'algorithmes non hiérarchiques, développés dans le but d'obtenir de meilleurs résultats d'estimation, avec une accélération de la convergence des algorithmes vers la solution. Ils présentent l'inconvénient d'aboutir à des segmentations, dépendant bien souvent de découpages initiaux trop arbitraires, produits par une fonction d'éclatement des sites. Les algorithmes "split and merge" présentent eux aussi l'inconvénient de réaliser une segmentation trop "rectiligne" des régions. Basés sur ce principe, des algorithmes adaptés ont été développés, certains n'effectuant que des divisions de régions (split), d'autres que des fusions (merge). Parmi ces derniers, nous avons retenu l'algorithme descendant proposé par G. Koepfler ([Koep88], [Lope93]), car il associe au principe classique de croissance de régions, celui de la minimisation d'une énergie. Nous le présentons donc dans le paragraphe suivant, car il constitue la base de notre proposition, visant à l'optimiser.

2. ALGORITHME DE KOEPLER

Cet algorithme procède par regroupements successifs de régions adjacentes, selon le critère U, donné par le modèle de Mumford et Shah ([MumS85], [MumS89]), dans sa version discrète :

$$U(C) = \sum_{k=0}^{K-1} I(C_k) + \beta \sum_{k=0}^{K-1} L(\partial C_k) \quad (2)$$

où :

- $I(C_k)$ est l'inertie de la région C_k , définie par :

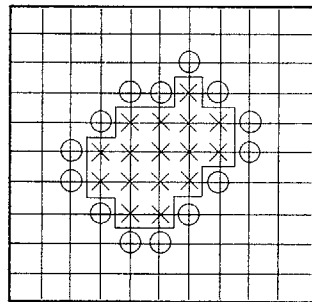
$$I(C_k) = \sum_{s \in \mathcal{C}_k} \|o_s - m_k\|^2 \quad \text{avec} \quad m_k = \frac{\sum_{s \in \mathcal{C}_k} o_s}{n_k} \quad \text{et} \quad n_k = \text{card}(C_k)$$

- β est un coefficient réel de pondération sur la longueur du bord de l'image. En effet, si β est petit, cela autorise un grand nombre de bords et la segmentation correspondante sera fine. Plus β augmente, plus la segmentation devient grossière. Il permet également de contrôler "la rondeur" des régions, en interdisant des bords de régions trop "accidentés".

- $L(\partial C_k)$ est la longueur du bord de la classe C_k . Ce bord est défini par :

$$\partial C_k = \{(s, s; i) ; s \in C_k, s; i \notin C_k, i \in \{1, -1, 2, -2\}\}$$

dans un système de 4-voisinage et sans prise en compte des bords extérieurs de l'image. Il peut être matérialisé par des segments de longueur 1 entre les éléments de l'ensemble ∂C_k , et sa longueur correspond à la somme des longueurs de ces segments. Le schéma ci-contre illustre ces notions de bord.



Ce critère, ainsi défini, peut apparaître comme une énergie gibbsienne qui s'écrirait :

$$U(C) = \sum_{s \in \mathcal{C}} \|o_s - m_{C_s}\|^2 + 2\beta \sum_{k=1,2} \sum_{s \in \mathcal{C}_k} (1 - \delta_{e_s, e_{s; k}})$$

où δ est le symbole de Kronecker défini par :

$$\delta_{e_s, e_{s; k}} = \begin{cases} 1 & \text{si } e_s = e_{s; k} \\ 0 & \text{sinon} \end{cases}$$

On peut maintenant définir la notion de bord entre deux régions, que l'on note $\partial C_k C_{k'}$:

$$\partial C_k C_{k'} = \{(s, s; i) ; s \in C_k, s; i \in C_{k'}, i \in \{1, -1, 2, -2\}\}$$

et l'égalité (2) devient :

$$U(C) = \sum_{k=0}^{K-1} I(C_k) + 2\beta \sum_{k \neq k'} L(\partial C_k C_{k'}) \quad (3)$$

Elle apparaît encore comme une énergie gibbsienne sur le graphe d'adjacence des régions car, dans cette formule, on peut évidemment se limiter aux couples (k, k') pour lesquels $\partial C_k C_{k'} \neq \emptyset$, ce qui correspond précisément à la définition du caractère adjacent de C_k et $C_{k'}$.

Soit C , la partition de S en K régions C_0, \dots, C_K , et C' la partition en K' régions ($K' = K-1$), obtenue en réunissant deux régions adjacentes de C , C_k et $C_{k'}$. La décision de réunir ces deux régions et de former C' dépend du signe et de la valeur de

$U(C') - U(C)$ qui s'écrit :

$$I(C_k \cup C_{k'}) - I(C_k) - I(C_{k'}) + \beta(L(\partial C_k \cup C_{k'}) - L(\partial C_k) - L(\partial C_{k'}))$$

ou encore [Ward63] :

$$U(C') - U(C) = \frac{n_k n_{k'}}{n_k + n_{k'}} \|m_k - m_{k'}\|^2 - 2\beta L(\partial C_k C_{k'}) \quad (4)$$

On note $\Delta_{kk'}$, cette différence, et on remarquera que $L(\partial C_k C_{k'})$ n'est autre que la longueur du bord commun aux deux régions C_k et $C_{k'}$. β étant fixé, on calcule, pour chaque région C_k , les $\Delta_{kk'}$ pour les régions $C_{k'}$ adjacentes à C_k . On regroupe C_k avec la région $C_{k'}$ qui réalise le minimum des $\Delta_{kk'}$, si ce minimum est négatif. Lorsque ce processus de regroupement s'arrête, on augmente la valeur de β afin de provoquer de nouveaux regroupements. L'algorithme s'arrête lorsqu'il ne reste plus qu'une seule région :

$\hat{C} \leftarrow$ segmentation initiale en K régions

$\beta \leftarrow \beta$ initial

tant que $K \geq 2$ **faire**

fusion \leftarrow faux

pour chaque région C_k de C **faire**

$$\Delta_{kk'} \leftarrow \min_{C_{k'} \text{ adjacentes à } C_k} \frac{n_k n_{k'}}{n_k + n_{k'}} \|m_k - m_{k'}\|^2 - 2\beta L(\partial C_k C_{k'})$$

si $\Delta_{kk'} < 0$

alors fusionner C_k et $C_{k'}$.

fusion \leftarrow vrai

finsi

finpour

si fusion = faux **alors** augmenter β

fantant que

On peut faire sur cet algorithme les commentaires suivants :

1. L'algorithme comprend une première phase qui consiste à construire la segmentation initiale selon trois options : en considérant des grilles 1 x 1, 2 x 2 ou 4 x 4. Dans le cas de grilles 1 x 1, la structure de liste de régions engendrée est très importante puisque chaque site constitue une région. Pour une image 256x256, on obtient 65536 régions, 130560 bords et 66045 sommets ! On peut également supposer que dans le cas de grilles 4 x 4, les 16 sites regroupés initialement ne constituent pas a priori une région homogène.
2. Les structures de données de Koepfler non présentées ici sont relativement complexes et peuvent être simplifiées si l'on se limite à la recherche d'une segmentation.
3. On peut améliorer le caractère optimal des regroupements effectués, en ne regroupant à chaque étape que les 2 régions adjacentes, telles que le gain sur le critère à optimiser soit maximum.



4. Le critère de Mumford et Shah présente l'inconvénient de mixer un terme ayant la dimension d'un carré (l'inertie intraclasse) avec un terme de comptage (la somme des longueurs des bords). Ceci implique un problème au niveau du choix de β , pour équilibrer les deux critères. Par exemple, à cause du caractère ascendant de la construction hiérarchique, un mauvais choix de β au départ risque de pénaliser les résultats des regroupements qui vont suivre. Ce changement du critère entre 2 segmentations successives supprime également la possibilité de visualiser les segmentations obtenues avec un même arbre hiérarchique, en vue de détecter une "bonne" segmentation par coupure du dendrogramme.

3. ALGORITHME PROPOSE

L'étude de l'algorithme précédent et de son implémentation nous conduit à en proposer une version modifiée, visant à améliorer son efficacité. Quatre points sont ici proposés, en réponse aux commentaires précédents :

1. Le premier est la recherche d'une segmentation initiale et cohérente, ce qui constitue un aspect important dans ce type de problème. Elle est obtenue par un algorithme de recherche de composantes connexes, suivi d'un algorithme de recherche de bords.
2. L'utilisation de la démarche hiérarchique classique des algorithmes d'analyse de données est le deuxième point. Elle consiste à regrouper, à chaque étape, les deux classes adjacentes de telle sorte que le gain sur le critère à optimiser soit maximum.
3. La simplification des structures de données utilisées pour effectuer ces regroupements constitue également une amélioration : on considère pour l'essentiel une liste de bords, décomposés en bords connexes et séparant les régions.
4. Le critère de Mumford et Shah est également modifié pour prendre en compte des quantités de dimension identique, ce qui n'est pas le cas de l'inertie et de la longueur.

3.1. Segmentation initiale

La segmentation initiale est obtenue par un algorithme classique de classification en composantes connexes, correspondant à un étiquetage en deux balayages de l'image, proposé dans [RosP66] et repris dans [ChaM90]. L'emploi de cet algorithme, basé sur la classification d'un site en fonction de ses prédécesseurs en 4 ou en 8-connexité, garantit la qualité de la pré-segmentation obtenue. L'image e obtenue est transformée en une structure de liste de régions et de bords par un algorithme de recherche de contours présenté ensuite.

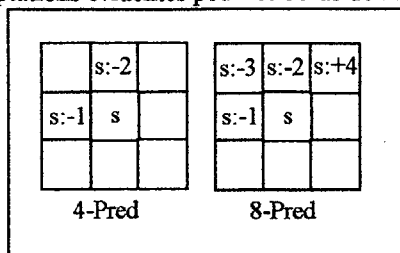
3.1.1. Algorithme de pré-classification

Pour tout site s de l'image, on considère l'ensemble de ses prédécesseurs $\text{Pred}(s)$ défini par :

$$\text{Pred}(s) = \{s:-2, s:-1\} \text{ en 4-connexité}$$

$$\text{Pred}(s) = \{s:-3, s:-2, s:-1, s:+4\} \text{ en 8-connexité}$$

avec des adaptations évidentes pour les bords de l'image.



On définit ensuite $P(s) = \{t \in \text{Pred}(s) \mid |o_s - o_t| \leq \sigma\}$.

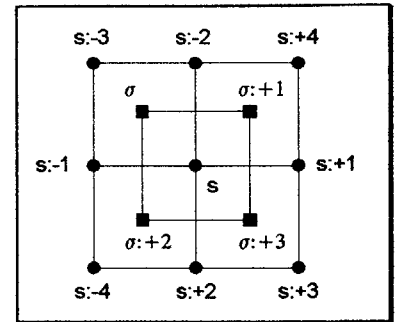
Dans le premier balayage de l'image initiale o , ligne par ligne, l'algorithme construit l'image e en étiquetant chaque site, avec une nouvelle valeur d'étiquette si ce site est tel que $P(s) = \emptyset$. Dans le cas contraire, on attribue à ce site, l'étiquette d'un de ses prédécesseurs et l'on examine les regroupements éventuels de régions qui pourraient subvenir via une connexion par ce site. Ces regroupements sont stockés dans une table des regroupements appelée T , définie sur l'ensemble des étiquettes et mise à jour après chaque regroupement :

$T(e)$ = étiquette des sites à regrouper avec les sites d'étiquette e . Ce premier balayage terminé, on élimine tous les chaînages de regroupements et l'on tasse les valeurs de la table, de manière à ce que toutes les étiquettes dans l'image soient de valeur croissante. On procède ensuite à un deuxième balayage de l'image e , en remplaçant simplement les étiquettes e_s par $T(e_s)$.

3.1.2. Algorithme de recherche de bords

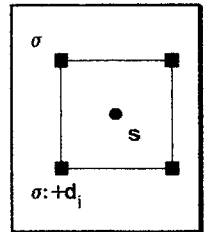
L'algorithme proposé ici est une adaptation de l'algorithme de suivi de contour présenté dans [ChaM91], en 4-connexité.

Il consiste à créer une liste intermédiaire de bords connexes pour l'ensemble des régions de l'image. Les informations stockées pour chaque bord connexe de cette liste sont les sommets extrémités, les numéros des régions qu'il sépare et sa longueur. Les sommets de ces bords sont situés dans une grille intermédiaire, avec les conventions ci-contre, où les sites sont représentés par des ronds et les sommets des bords par des carrés.



L'algorithme qui suit utilise les notations suivantes :

- **sommet(s)** renvoie le sommet situé entre les sites s et $s:-3$; il a donc les mêmes coordonnées que s ;
- **directions de déplacement vers les 4-voisins** : $d_0 = 1$, $d_1 = -2$, $d_2 = -1$, $d_3 = 2$;
- si σ est un sommet, **site(σ , d_i)** renvoie le



site situé au centre du carré $\{\sigma, \sigma + d_i, \sigma + d_{i+1 \bmod 4}, \sigma + d_{i+2 \bmod 4}\}$

construit en partant de σ dans la direction d_i puis en tournant dans le sens trigonométrique ;

- λ contient la longueur d'un bord connexe ;
- **trouvé** indique si une extrémité de bord connexe a été trouvée sur le bord de la classe, d'où partent d'autres bords connexes.

Algorithme :

pour chaque région C_k de l'image **faire**

$s \leftarrow$ site de C_k tel que $s_0:-1 \notin C_k$

$k' \leftarrow e_{s_0:-1}$; $\sigma_f \leftarrow$ sommet(s_0)+2

$\sigma' \leftarrow \sigma_f$; $i \leftarrow 2$; trouvé \leftarrow faux

itérer

extrémité \leftarrow faux

tant que site(σ' , d_i) $\notin C_k$ **faire**

si site(σ' , d_i) $\neq k'$

alors extrémité \leftarrow vrai



$k'' \leftarrow \text{site}(\sigma', d_i) ; i \leftarrow i + 1 \text{ mod } 4$

fin tantque

si extrémité = vrai

alors

si trouvé = vrai

alors ajouter le bord connexe $(\sigma, \sigma', k, k', \lambda)$

sinon trouvé \leftarrow vrai ; $\sigma_d \leftarrow \sigma'$

$\sigma \leftarrow \sigma' ; k' \leftarrow k'' ; \lambda \leftarrow 0$

sinon $\lambda \leftarrow \lambda + 1$

$\sigma' \leftarrow \sigma' + d_i ; i \leftarrow i - 1$

jusque $\sigma' = \sigma_f$

si trouvé = vrai

alors $\sigma' \leftarrow \sigma_d$; ajouter le bord connexe $(\sigma, \sigma', k, k', \lambda)$

sinon $\sigma \leftarrow \sigma'$; ajouter le bord connexe $(\sigma, \sigma', k, k', \lambda)$

finpour

3.2. Algorithme de segmentation

En s'inspirant des algorithmes hiérarchiques classiques en analyse de données, on propose de regrouper à chaque étape les deux régions adjacentes C_k et $C_{k'}$, qui réalise un accroissement

$\Delta_{kk'}$ minimum pour le critère à optimiser :

$$C^{(0)} \leftarrow \{C_0, \dots, C_{K-1}\} ; \mathcal{H}^{(0)} \leftarrow \{0, \dots, K-1\}$$

pour $k, k' \in \mathcal{H}^{(0)} ; k \neq k'$ faire calcul de $\Delta_{kk'}$

pour n de 1 à $K-1$ faire

$$kk' \leftarrow \arg \min_{\substack{I, I' \in \mathcal{H}^{(n-1)} \\ C_i \text{ et } C_{i'} \text{ adjacentes}}} \Delta_{II'}$$

$$\mathcal{H}^{(n)} \leftarrow \mathcal{H}^{(n-1)} - \{k, k'\} + \{kk'\} ;$$

$$C_{kk'} \leftarrow C_k \cup C_{k'} ;$$

pour $k'' \in \mathcal{H}^{(n)} ; k'' \neq kk'$ faire calcul de $\Delta_{kk''}$

finpour

3.3. Critères de regroupement

Nous terminerons cet article en précisant différentes formes possibles pour le critère $\Delta_{kk'}$, prenant en compte les inerties $I(C_k)$ et les longueurs des bords $L(\partial C_k)$ des régions. Comme la hiérarchie produite par notre algorithme est associé à un seul choix de β , il importe aussi de pouvoir dès le départ, fixer un choix "raisonnable" pour β . Une première option consiste à normaliser l'inertie intraclasse en considérant le rapport :

$$\frac{\sum_{k=0}^{K-1} I(C_k)}{\sum_{s \in \mathcal{S}} \|o_s - m\|^2} = \frac{\sum_{k=0}^{K-1} I(C_k)}{M\sigma^2}$$

entre l'inertie intraclasse et l'inertie totale, car celui-ci est toujours compris entre 0 et 1. On obtient ainsi un critère d'homogénéité des régions, invariant par homothétie dans l'espace \mathcal{O} des observations et robuste vis à vis du nombre de pixels dans chaque région, donc vis à vis d'un éventuel sous-échantillonnage des sites ou d'un changement de résolution par regroupement de sites voisins. Le problème de la normalisation des longueurs des bords est plus délicat. Plusieurs solutions sont proposées par exemple dans [CosC89] (chap. IX sur la description des formes). Beaucoup sont basées sur l'inégalité isopérimétrique [Osse78] :

$$L(\partial C)^2 \geq 4\pi m_C$$

que l'on peut songer à utiliser pour produire un terme normalisé. Par exemple, on obtient une quantité toujours supérieure ou égale à 1, avec égalité dans le cas de régions circulaires si l'on choisit :

$$\frac{1}{K} \sum_{k=0}^{K-1} \frac{L(\partial C_k)}{2\pi^{1/2} n_k^{1/2}}$$

Mais ce choix complique les calculs de mise à jour des $\Delta_{kk'}$.

Pour un choix de β , on pourra se reporter à [Besa86] qui, dans le cas de l'algorithme ICM, observe une relative stabilité des résultats avec $\beta = 1.55$, en dimension 1, pour un critère du type :

$$\frac{\sum_{k=0}^{K-1} I(C_k)}{2\sigma^2} - \beta \sum_{s \in \mathcal{S}} \sum_{k=1,2} \delta_{o_s, e_{s,k}}$$

On observera un problème également sous-jacent à la prise en compte de ces 2 types de critères : la croissance des $\Delta_{kk'}$ n'est pas garantie dans la construction hiérarchique, ce qui peut conduire à des inversions dans les représentations hiérarchiques. Par exemple avec le critère de Mumford et Shah, on peut établir la formule suivante :

$$\Delta_{kk', k''} = \frac{(n_k + n_{k''})\Delta_{kk''} + (n_{k'} + n_{k''})\Delta_{k'k''} - n_{k''}\Delta_{kk'}}{n_k + n_{k'} + n_{k''}} - 2\beta \frac{n_k \cdot L(\partial C_k \cdot C_{k''}) + n_{k'} \cdot L(\partial C_{k'} \cdot C_{k''}) + n_{k''} \cdot L(\partial C_k \cdot C_{k'})}{n_k + n_{k'} + n_{k''}}$$

lorsque la région $C_{k''}$ est adjacente à C_k et $C_{k'}$ et avec des adaptations évidentes si $C_{k''}$ n'est adjacente qu'à l'une des 2 classes C_k ou $C_{k'}$. A noter néanmoins l'expression très simple du terme correctif, sous la forme d'une moyenne pondérée des longueurs des 3 bords communs.

4. BIBLIOGRAPHIE

[Besa86] J. Besag. *On the statistical analysis of dirty pictures*. J. R. Statistic. Soc. B. Vol. 48, n° 3, p 259-302. 1986.

[BouL91] C. Bouman, B. Lui. *Multiple resolution segmentation of textured images*. IEEE Transactions on pattern analysis and machine intelligence. Vol. 13, n° 2, p 99-113. Février 1991.

[BrLP93] L. Brenac, J. Lemaire, I. Pons. *Méthodes de segmentation d'images selon des critères bayésiens*. Technical Report I3S n°92-24 1993.

[ChaM91] J.M. Chassery, A. Montanvert. *Géométrie discrète en analyse d'images*. Traité des Nouvelles Technologies, série Images. Hermès 1991.

[CosC89] M. Coster, J.L. Chermant. *Précis d'analyse d'images*. Presses du CNRS. 1989.

[DubJ89] R.C. Dubes, A.K. Jain. *Random field models in image analysis*. Journal of Applied Statistics. Vol. 16, n°2, p 131-164. 89.

[GemG84] S. Geman, D. Geman. *Stochastic relaxation, Gibbs distributions and the bayesian restoration of images*. IEEE Transactions on pattern analysis and machine intelligence. Vol. PAMI-6, n° 6, p 721-741. 1984.

[KoeP88] G. Koepfler. *Formalisation et analyse numérique de la segmentation d'images*. Thèse de Doctorat. Université de Paris IX Dauphine. 1988.

[Lope93] C. Lopez. *Théorie de la discrimination de textures. Axiomatization et algorithmiques numériques*. Thèse de Doctorat. Université de Paris IX Dauphine. 1993.

[Mong90] O. Monga. *Segmentation d'images : où en sommes nous ?* Rapp. de Recherche INRIA no 1216. Avril 1990.

[MumS85] D. Mumford, J. Shah. *Boundary detection by minimizing functionals*. IEEE Conference on Computer Vision and Pattern Recognition. San Francisco. 1985.

[MumS89] D. Mumford, J. Shah. *Optimal approximations by piecewise smooth functions and associated variational problems*. Communications on Pure and Applied Mathematics. vol XLII n°4 1989.

[Osse78] R. Osserman. *Isoperimetric inequalities*. Bull. Am. Math. Soc. Vol. 84, 1978, p 1182-1238. 1978.

[PerH91] P. Perez, F. Heitz. *Une approche multi-échelle à l'analyse d'images par champs markoviens*. Congrès RFIA. Lyon 25-29 Novembre 1991.

[RosP66] A. Rosenfeld, J.L. Pfaltz. *Sequential operations in digital picture processing*. Journal of ACM. Vol. 13, no 4, p 471-494. 1966.