Cours IMA 203

Polycopié

E. Angelini, I. Bloch, S. Ladjal, M. Sigelle, F. Tupin

Chapitre 2

Méthodes markoviennes en traitement d'images

Chapitre rédigé par Florence TUPIN et Marc SIGELLE

L'information véhiculée par une image va bien au-delà de la seule donnée des niveaux de gris en chaque site (pixel), et la description se fait en termes de zones, contours, structures définis par les contrastes, textures, etc. qui peuvent être présents dans l'image. Le niveau de gris en un site n'est donc souvent pas significatif en lui-même, mais dans ses relations et interactions avec les pixels voisins.

Cette propriété des images, à savoir les interactions locales entre niveaux de gris voisins pour définir les différentes régions de l'image, va nous permettre d'utiliser un formalisme markovien dans de nombreux traitements, qu'il s'agisse de restauration, de segmentation ou plus tard d'analyse complète des images. Le principe est de définir des énergies locales entre groupes de sites reflétant les interactions entre niveaux de gris. L'énergie globale est alors reliée à la probabilité d'apparition de l'image dans le cadre des champs de Gibbs.

Dans ce chapitre, nous introduisons tout d'abord de façon intuitive la notion d'énergie locale avant de définir plus formellement un champ de Markov et d'énoncer le théorème d'équivalence entre champs de Markov et champs de Gibbs. Les algorithmes d'échantillonnage d'un champ de Markov (échantillonneur de Gibbs et algorithme de Métropolis) sont ensuite présentés, ainsi que les modèles markoviens les plus courants. L'utilisation des champs markoviens en traitement d'images dans un cadre bayésien montre la nécessité de pouvoir accéder aux configurations les plus probables d'un champ markovien et nous amène à la présentation du recuit simulé. Dans les parties suivantes, nous abordons les différents estimateurs (MAP, MPM, TPM), le problème de l'estimation des paramètres du champ, les processus de bords, avant de mentionner l'application de la modélisation markovienne à des graphes de primitives de plus haut niveau que les pixels.

2.1 Définition et simulation d'un champ de Markov

L'image est formée d'un ensemble fini S de sites s_i correspondant aux pixels. S est donc essentiellement un réseau discret fini, partie de Z^d, si on note d la dimension de l'espace (2 le plus classiquement, 3 pour les volumes, etc.). À chaque site est associé un descripteur, représentant l'état du site et qui peut être son niveau de gris, une étiquette, ou une information plus complexe, et prenant ses valeurs dans E.

La notion d'interactions locales nécessite de structurer les relations spatiales entre les différents sites du réseau.

Pour ce faire, on munit S d'un système de voisinage $\mathcal{V}$ défini de la façon suivante :

$$\mathcal{V}_s = \{t\} \text{ tels que } \begin{cases} s \notin \mathcal{V}_s \\ t \in \mathcal{V}_s \Rightarrow s \in \mathcal{V}_t \end{cases}$$

À partir d'un système de voisinage, un système de cliques peut être déduit : une clique est soit un singleton de S, soit un ensemble de sites tous voisins les uns des autres. En fonction du système de voisinage utilisé, le système de cliques sera différent et fera intervenir plus ou moins de sites. On notera $\mathcal{C}$ l'ensemble des cliques relatif à $\mathcal{V}$, et $\mathcal{C}_k$ l'ensemble des cliques de cardinal k.

Les interactions locales entre niveaux de gris (ou descripteurs) de sites voisins peuvent alors s'exprimer comme un potentiel de clique. Soit c une clique, on lui associe le potentiel U_c dont la valeur dépend des niveaux de gris (ou descripteurs) des pixels constituant la clique. En poursuivant ce raisonnement, on peut définir l'énergie globale de l'image comme la somme des potentiels de toutes les cliques :

$$U = \sum_{c \in \mathcal{C}} U_c$$

et l'énergie locale en un site comme la somme des potentiels de toutes les cliques auxquelles il appartient :

$$U_s = \sum_{c \in \mathcal{C} / s \in c} U_c$$

2.1.1 Modélisation probabiliste de l'image

La définition des champs de Markov qui sera donnée dans la section suivante nécessite une modélisation probabiliste de l'image. Ainsi, l'image dont nous disposons va être considérée comme une réalisation d'un champ aléatoire. Soit s un site de l'image, on peut en effet lui associer une variable aléatoire (v.a) X_s prenant ses valeurs dans E. Le niveau de gris x_s en s n'est ainsi qu'une réalisation[1] de la v.a X_s. On définit alors le champ aléatoire $X = (X_s, X_t, ..)$ prenant ses valeurs dans $\Omega = E^{|S|}$. On trouvera aussi le terme de processus aléatoire pour X ; en toute rigueur, « processus » devrait être réservé au cas d'un ensemble d'indexation continu, et champ au cas discret.

Dans ce cadre probabiliste, l'image considérée est simplement une réalisation x du champ. La probabilité globale de x, $P(X = x)$, permet d'accéder en quelque sorte à la vraisemblance de l'image, et les probabilités conditionnelles locales d'une valeur en un site permettent de mesurer le lien statistique entre un niveau de gris et le reste de l'image. L'hypothèse markovienne permet d'évaluer ces quantités.

2.1.2 Champs de Markov - Champs de Gibbs

Définition d'un champ de Markov

Considérons x_s la valeur du descripteur prise au site s et $x^s = (x_t)_{t \neq s}$ la configuration de l'image excepté le site s. La définition d'un champ de Markov est alors la suivante :

X est un champ de Markov ssi la probabilité conditionnelle locale en un site n'est fonction que de la configuration du voisinage du site considéré

ce qui s'exprime de façon formelle par :

$$P(X_s = x_s \ / \ x^s) = P(X_s = x_s \ / \ x_t, t \in \mathcal{V}_s)$$

Ainsi, le niveau de gris en un site ne dépend que des niveaux de gris des pixels voisins de ce site.

[1] On notera généralement en lettres majuscules les variables aléatoires et en minuscules leurs réalisations.

Equivalence entre champs de Markov et champs de Gibbs

La modélisation markovienne prend toute sa puissance grâce au théorème que nous allons voir maintenant. En effet, celui-ci permettra d'accéder aux expressions des probabilités conditionnelles locales. Il nous faut au préalable définir un certain nombre de notions relatives aux mesures et champs de Gibbs.

Définition d'une mesure de Gibbs : La mesure de Gibbs de fonction d'énergie (ou d'hamiltonien) $U : \Omega \rightarrow R$ est la probabilité P définie sur Ω par :

$$P(X = x) = \frac{1}{Z} \exp\left(-U(x)\right)$$

avec

$$U(x) = \sum_{c \in \mathcal{C}} U_c(x)$$

où $\mathcal{C}$ est le système de cliques associé au système de voisinage $\mathcal{V}$ de U[2].

$Z = \sum_{x \in \Omega} \exp(-U(x))$ est une constante de normalisation appelée fonction de partition de Gibbs.

Nous pouvons maintenant définir le champ de Gibbs de potentiel associé au système de voisinage $\mathcal{V}$: c'est le champ aléatoire X dont la probabilité est une mesure de Gibbs associée au système de voisinage $\mathcal{V}$, ce qui implique :

$$P(X = x) = \frac{1}{Z} \exp\left(-U(x)\right) = \frac{1}{Z} \exp\left(-\sum_{c \in \mathcal{C}} U_c(x)\right)$$

L'énergie globale d'un champ de Gibbs possède donc la propriété de se décomposer sous forme d'une somme d'énergies locales, qui comme on le verra par la suite permettront d'accéder aux probabilités conditionnelles locales.

Le théorème de Hammersley-Clifford [4] établit alors le résultat fondamental suivant sous les hypothèses :
- S fini ou dénombrable,
- le système de voisinage $\mathcal{V}$ est borné,
- l'espace des états E est discret

X est un champ de Markov relativement à $\mathcal{V}$ et $P(X = x) > 0 \ \forall x \in \Omega$ si et seulement si X est un champ de Gibbs de potentiel associé à $\mathcal{V}$.

Le théorème de Hammersley-Clifford, et la forme bien spécifique de probabilité de X qui en résulte, va permettre de lier les probabilités globales et locales comme nous allons le voir maintenant.

Définissons l'énergie locale U_s par :

$$U_s(x_s \ / \ x_t, t \in \mathcal{V}_s) = \sum_{c \in \mathcal{C} \ / \ s \in c} U_c(x_s, x_t, t \in \mathcal{V}_s) = \sum_{c \in \mathcal{C} \ / \ s \in c} U_c(x_s, V_s)$$

en notant $V_s = (x_t, t \in \mathcal{V}_s)$. Cette énergie locale ne fait donc intervenir que les voisins de s.

$$P(X_s = x_s \mid X^s = x^s) \quad = \quad \frac{\exp(-U_s(x_s \ / \ V_s)}{\sum_{x_s \in E} \exp(-U_s(x_s \ / \ V_s))} \tag{2.1}$$

L'expression obtenue, qui ne fait intervenir que les potentiels des cliques contenant le site s (ce qui nous permet de retrouver au passage l'hypothèse markovienne), est très importante. En effet, autant il n'est pas possible partant d'une configuration x d'accéder à sa probabilité à cause de la constante de normalisation, autant il est possible de calculer en chaque site la probabilité conditionnelle locale. Cette expression sera à la base de tous les algorithmes de simulation de champs markoviens que nous verrons dans la section suivante.

[2]Il est toujours possible de trouver un système de voisinage $\mathcal{V}$ permettant de décomposer U ; le cas extrême correspondant à des sites tous voisins les uns des autres.

2.1.3 Echantillonnage de MRF

Le problème qui se pose alors est, étant défini un champ de Markov, comment pouvons-nous réaliser le tirage d'une configuration (une image ici) en suivant la loi de probabilité de Gibbs caractéristique de ce champ ? Deux algorithmes ont été proposés pour synthétiser des réalisations d'un champ de Markov qui sont :

– l'échantillonneur de Gibbs,
– l'algorithme de Métropolis

que nous allons décrire maintenant.

L'échantillonneur de Gibbs

Cet algorithme, proposé par Geman et Geman [19], repose sur la construction itérative d'une suite d'images. A la convergence, i.e après un nombre d'itérations suffisant, les images construites sont des réalisations tirées selon la loi de Gibbs globale.

La méthode de construction de l'image à l'itération n, partant de l'image à l'itération $n-1$ se fait par mises à jour successives des sites de l'image. À l'étape n :

- choix d'un site s ;
- au site s, selon la configuration des voisins V_s pour l'image $x^{(n-1)}$, calcul de la probabilité conditionnelle locale :

$$P(X_s = x_s \mid V_s) = \frac{\exp(-U_s(x_s \mid V_s))}{\sum_{\xi \in E} (\exp(-U_s(\xi \mid V_s))}$$

- mise à jour du site s par tirage aléatoire selon la loi $P(X_s = x_s \mid V_s)$.

On considère que l'algorithme a convergé après un grand nombre d'itérations ou lorsque le nombre de changements est faible. Le choix du site s considéré à l'étape n peut se faire de n'importe quelle façon à condition de balayer tous les sites un très grand nombre de fois (théoriquement un nombre infini de fois). Les méthodes usuelles consistent à tirer un site selon une loi uniforme, ou effectuer un balayage classique, ligne par ligne, de l'image.

L'algorithme de Metropolis

L'échantillonneur de Gibbs est un algorithme très utilisé en traitement d'images pour la synthèse de champs de Markov. Néanmoins, un algorithme antérieur et issu de la physique statistique avait été mis au point dans les années 50 par Metropolis [36].

Cet algorithme repose sur un principe similaire à l'échantillonneur de Gibbs, et il s'agit également d'un algorithme de relaxation probabiliste. Le principe est là encore de construire une suite d'images qui seront des tirages selon la loi du champ de Markov après un nombre suffisamment grand d'itérations. Mais la mise à jour en un site s'effectue de façon différente. Ainsi à l'étape n :

- choix d'un site s
- tirage aléatoire d'un descripteur λ dans E selon une loi uniforme ;
- calcul de la variation d'énergie pour le passage du label du site s de $x_s^{(n-1)}$ à λ :

$$\Delta U = U_s(\lambda \mid V_s^{(n-1)}) - U_s(x_s^{(n-1)} \mid V_s^{(n-1)})$$

- deux cas sont alors possibles :
  1. $\Delta U < 0$, le changement est accepté : $x_s^{(n)} = \lambda$;
  2. $\Delta U \geq 0$, le changement est accepté ou refusé par tirage selon les probabilités $p = \exp(-\Delta U)$ et $1-p$.

2.1.4 Le recuit simulé

Nous avons vu dans les paragraphes précédents comment échantillonner selon la loi de probabilité de Gibbs associée au champ de Markov. À chaque application des précédents algorithmes, une nouvelle réalisation est obtenue. Il peut être utile également de pouvoir calculer la ou les configurations les plus probables qui correspondent aux états d'énergie minimale. C'est l'algorithme du recuit simulé qui permet de trouver ces réalisations.

Avant de présenter cet algorithme, nous avons besoin de quelques résultats sur les distributions de Gibbs avec paramètre de température que nous présentons maintenant.

Distribution de Gibbs avec température

Une distribution de Gibbs avec paramètre de température est une probabilité qui s'écrit :

$$P_T(X = x) = \frac{1}{Z(T)} \exp(-\frac{U(x)}{T})$$

avec $Z(T) = \sum_x \exp(-\frac{U(x)}{T})$ et $T > 0$. Le terme de température provient de l'analogie avec la physique statistique.

Il est intéressant d'étudier le comportement de cette distribution pour des valeurs extrêmes du paramètre de température.

• $T \to \infty$:

On a $\exp(-\frac{U(x)}{T}) \to 1$ et comme $\sum_x P_T(X = x) = 1$, on obtient

$$P_T(X = x) \to \frac{1}{\text{Card } \Omega}$$

Donc P_T converge vers la probabilité uniforme sur Ω, i.e pour une température infinie tous les états sont équiprobables.

• $T \to 0$:

Notons U^* l'énergie minimale et Ω^* l'ensemble des configurations atteignant l'énergie minimale $\Omega^* = \{x_1, ..., x_k\}$ ($x_1, ...x_k$ sont les minima globaux de l'énergie). On peut écrire :

$$\begin{aligned} P_T(X = x) &= \frac{\exp(-\frac{U(x)}{T})}{\sum_y \exp(-\frac{U(y)}{T})} = \frac{\exp(-\frac{U(x)-U^*}{T})}{\sum_y \exp(-\frac{U(y)-U^*}{T})} \\ &= \frac{\exp(-\frac{U(x)-U^*}{T})}{\sum_{y \notin \Omega^*} \exp(-\frac{U(y)-U^*}{T}) + \sum_{y \in \Omega^*} 1} \end{aligned}$$

◇ Si $x \notin \Omega^*$, on a $U(x) - U^* > 0$ et $\exp(\frac{(U(x)-U^*)}{T}) \to 0$ pour $T \to 0$. Donc $P_T(x) \to 0$ si x n'est pas un minimum global de l'énergie.

◇ Si $x \in \Omega^*$, on a : $P_T(x_1) = P_T(x_2) = ... = P_T(x_k) = \frac{1}{k}$ (il y a une somme finie de termes qui tendent vers 0 au dénominateur).

Ce qui signifie que lorsque la température est nulle P_T est uniformément distribuée sur les minima globaux de l'énergie, i.e sur les configurations les plus probables. C'est ce résultat qui est à la base de l'algorithme de recuit simulé.

Algorithme du recuit simulé

Cet algorithme est dédié à la recherche d'une configuration d'énergie minimale d'un champ de Gibbs (on ne cherche plus ici à échantillonnner contrairement à précédemment). L'idée d'intégrer un paramètre de température et de simuler un recuit a été initialement proposée par Kirkpatrick [31] et reprise par Geman et Geman [19] qui ont proposé l'algorithme suivant.

Comme les algorithmes de simulation, c'est un algorithme itératif qui construit la solution au fur et à mesure. Le déroulement de l'algorithme est le suivant (en notant n le numéro de l'itération) :

- choix d'une température initiale $T^{(0)}$ suffisamment élevée
- choix d'une configuration initiale quelconque $x^{(0)}$
- à l'étape n
  - simulation d'une configuration $x^{(n)}$ pour la loi de Gibbs d'énergie $\frac{U(x)}{T^{(n)}}$ à partir de la configuration $x^{(n-1)}$; la simulation peut se faire par l'échantillonneur de Gibbs ou l'algorithme de Métropolis ; on réalise en général un balayage complet de l'image à la température $T^{(n)}$;
  - diminution lente de la température : $T^{(n)} > \frac{c}{\log(1+n)}$
- arrêt lorsque le taux de changement est faible.

La décroissance logarithmique de la température est un rythme très lent ; en pratique des décroissances géométriques sont utilisées, souvent sans dégradation notable des résultats obtenus. La constante c intervenant dans la décroissance dépend de la variation énergétique globale maximale sur l'espace des configurations.

Notons que contrairement aux algorithmes de l'échantillonneur de Gibbs et de Métropolis qui échantillonnent selon la loi de Gibbs et qui sont en mesure de donner toutes les configurations possibles, les images obtenues par recuit simulé sont uniques et doivent en théorie correspondre aux minima globaux de l'énergie.

Il existe une preuve de convergence de cet algorithme, qui repose à nouveau sur la construction d'une chaîne de Markov, mais qui est hétéroggène cette fois-ci à cause de la variation du paramètre de température [19]. Intuitivement, le recuit simulé permet d'atteindre un optimum global, car il accepte des remontées en énergie. Avec la décroissance de la température, ces sauts énergétiques sont progressivement supprimés au fur et à mesure qu'on se rapproche de l'optimum global. La descente en température doit donc se faire suffisamment lentement pour que l'algorithme ne reste pas piégé dans un minimum local de l'énergie.

Algorithme des modes conditionnels itérés (ICM)

Malheureusement, l'algorithme du recuit simulé est très lourd en temps de calcul puisqu'il demande la génération d'un grand nombre de configurations au fur et à mesure que la température décroît. Des algorithmes sous-optimaux sont donc souvent utilisés en pratique. Besag [5] a ainsi proposé un autre algorithme, beaucoup plus rapide, mais pour lequel nous n'avons pas de preuve de convergence vers un minimum global. Il s'agit de l'ICM, *Iterated Conditional Mode*, que nous allons présenter ici.

Cet algorithme est un algorithme itératif modifiant à chaque étape les valeurs x_s de l'ensemble des sites de l'image. Mais à la différence de ces algorithmes qui étaient stochastiques par essence, la modification d'une valeur se fait ici de façon déterministe.

On construit donc, partant d'une configuration initiale $x(0)$, une suite d'images $x(n)$, convergeant vers une approximation du MAP $\hat{x}$ recherché. Soit un tour la visite de tous les sites de l'image, on parlera dans la suite d'itérations à chaque mise à jour d'un site et d'étape à chaque mise à jour de toute l'image (i.e accomplissement d'un tour).

Le déroulement de l'étape n s'effectue de la façon suivante : on parcourt tous les sites et en chaque site, on effectue les deux opérations suivantes :

1. calcul des probabilités conditionnelles locales, pour toutes les valeurs possibles de λ dans E du site :

$$P(X_s = \lambda \ / \ \hat{x_r}(k), r \in \mathcal{V}_s)$$

(en pratique, calcul plus simplement des énergies conditionnelles locales)

2. mise à jour de la valeur par le λ maximisant la probabilité conditionnelle locale :

$$\hat{x_s}(k+1) = \text{Argmax}_\lambda P(X_s = \lambda \ / \ \hat{x_r}(k), r \in \mathcal{V}_s)$$

(ou de façon équivalente, minimisant l'énergie conditionnelle locale).

Le processus s'arrête lorsque le nombre de changements d'une étape à l'autre devient suffisamment faible.

On peut montrer que l'énergie globale de la configuration $\hat{x}$ diminue à chaque itération. Cet algorithme, contrairement au recuit simulé, est très rapide (une dizaine de balayages permettent d'arriver à convergence) et peu coûteux en temps de calcul puisqu'il ne nécessite que le calcul des énergies conditionnelles locales. En contrepartie, ses performances dépendent très fortement de l'initialisation (par rapport à la forme du paysage énergétique) puisqu'il converge vers un minimum local. L'ICM s'apparente à une descente en gradient (on fait baisser l'énergie à chaque itération) ou à un recuit simulé gelé à température nulle, et peut donc rester bloqué dans le minimum énergétique local le plus proche de l'initialisation. Le recuit simulé, au contraire, grâce au paramètre de température et aux remontées en énergie qu'il autorise permet d'accéder au minimum global.

Notons qu'il a également été proposé d'utiliser la programmation dynamique pour estimer le MAP [12]. Mais il est alors nécessaire d'être dans une configuration simple de segmentation (peu d'étiquettes, dimensions raisonnables) et seule une approximation peut être obtenue.

2.1.5 Quelques MRF fondamentaux

Nous présentons ici quelques uns des champs de Markov les plus utilisés. Comme indiqué précédemment, ces champs sont définis par leur voisinage et leurs fonctions de potentiel. Ils sont illustrés par le tirage de réalisations selon l'échantillonneur de Gibbs.

◇ Modèle d'Ising :

Ce modèle est le plus ancien (1925 [29]) et a été développé lors de l'étude du ferro-magnétisme en physique statistique. L'espace des descripteurs est celui des états des spins, i.e $E = \{-1, 1\}$ (espace binaire), et le voisinage est constitué par les 4 ou 8 plus proches voisins dans un espace bidimensionnel. Les potentiels sont des potentiels en tout ou rien :

$$\begin{aligned} U_{c=(s,t)}(x_s, x_t) &= -\beta \text{ si } x_s = x_t \\ &= +\beta \text{ si } x_s \neq x_t \end{aligned}$$

Ce qui s'écrit également $U_{c=(s,t)}(x_s, x_t) = -\beta x_s x_t$.

Les potentiels des cliques d'ordre 1 (clique constituée par un seul spin) sont de la forme $-Bx_s$. L'énergie totale s'écrit :

$$U(x) = -\sum_{c=(s,t)\in\mathcal{C}} \beta x_s x_t - \sum_{s\in S} B x_s$$

β est la constante de couplage entre sites voisins et B représente un champ magnétique externe. Lorsque β est positif, les configurations les plus probables (i.e d'énergies plus faibles) sont celles pour lesquelles les spins sont de même signe (ferro-magnétisme), alors que dans le cas de β négatif, au contraire, on favorisera l'alternance de spins de signes opposés (anti-ferromagnétisme). La valeur (signe et valeur absolue) de β conditionne donc la régularité du modèle d'Ising. Quant au champ magnétique externe relatif au potentiel d'ordre 1, il favorise a priori par son signe un spin ou un autre.

◇ Modèle de Potts :

Il s'agit d'une extension du modèle d'Ising [60] pour un espace m-aire, i.e. $E = \{0, m-1\}$. Il peut s'agir de plusieurs niveaux de gris, mais plus souvent pour ce modèle, d'étiquettes (labels) pouvant représenter une classification de l'image (par exemple les classes *eau, forêt, champ, ville*). Le voisinage considéré est 4- ou 8-connexe et les potentiels sont comme précédemment en tout ou rien mais définis seulement pour les cliques d'ordre 2 :

$$\begin{aligned} U_{c=(s,t)}(x_s, x_t) &= -\beta \text{ si } x_s = x_t \\ &= +\beta \text{ si } x_s \neq x_t \end{aligned}$$

Lorsque β est positif, les configurations les plus probables correspondent à des sites voisins de même niveau de gris ou descripteur, ce qui donne des réalisations constituées par des larges zones homogènes. La taille de ces regions est gouvernée par la valeur de β.

Il est possible de définir des modèles utilisant des pondérations β différentes en fonction des directions des cliques, et de privilégier ainsi certaines directions.

Ce modèle permet également de prendre en compte différentes relations entre les régions (i.e. entre différentes valeurs des descripteurs). On peut par exemple définir des pondérations $\beta(e_s, e_t)$ pour $e_s, e_t \in E$. Dans notre exemple de classification en 4 étiquettes *eau, forêt, champ, ville*, une configuration de sites avec les étiquettes *champ / forêt* peut être supposée plus probable qu'une configuration *ville / forêt*, d'où des valeurs $\beta(champ, forêt)$ et $\beta(ville, forêt)$ différentes [50].

◇ Modèle markovien gaussien :

Ce modèle est réservé aux images en niveaux de gris $E = \{0, ..., 255\}$ et ne convient pas bien aux images d'étiquettes. Le voisinage est 4 ou 8-connexe et l'énergie est de la forme :

$$U(x) = \beta \sum_{c=(s,t)} (x_s - x_t)^2 + \alpha \sum_{s \in S} (x_s - \mu_s)^2$$

Le premier terme correspondant aux cliques d'ordre 2 est un terme de régularisation, qui favorise les faibles différences de niveaux de gris entre sites voisins pour $\beta > 0$. Le second terme peut correspondre à un terme d'attache aux données dans le cas où on possède une image de données extérieures. Le rapport $\frac{\alpha}{\beta}$ pondère les influences respectives de l'attache aux données et de la régularisation, et les valeurs absolues des paramètres caractérisent le caractère plus ou moins piqué ou équiréparti au contraire de la distribution.

2.1.6 Applications : restauration et segmentation

Cadre bayésien

Pour ces deux applications, on peut modéliser le problème dans un cadre bayésien de la façon suivante. Nous disposons d'une certaine donnée (image) que nous noterons y et que nous pouvons considérer comme une réalisation d'un champ aléatoire Y. Nous cherchons une réalisation x de l'image restaurée ou segmentée, que nous pouvons modéliser comme un champ de Markov X. X est le champ des étiquettes (labels) dans le cas de la segmentation, le champ des intensités dans le cas de la restauration. Les espaces de configurations ne sont donc pas nécessairement les mêmes pour X et Y. Ces deux champs sont liés par le processus d'acquisition de l'image, qui conduit du champ idéal X, le processus image originel que nous cherchons, au champ bruité Y que nous observons. La restauration ou la segmentation ont pour objectif d'inverser le processus et donc de remonter à une réalisation de X à partir de l'observation des données bruitées y. On parle dans ce contexte de champ de Markov caché pour X, ou de données incomplètes puisque y n'est pas une réalisation de X.

On peut par exemple utiliser le critère du maximum a posteriori et rechercher la configuration $\hat{x}$ maximisant la probabilité de X conditionnellement à la donnée y i.e $P(X = x \,/\, Y = y)$. Or la règle de Bayes permet d'écrire :

$$P(X = x \,/\, Y = y) = \frac{P(Y = y \,/\, X = x)P(X = x)}{P(Y = y)}$$

Expression dans laquelle il s'agit alors d'analyser chacun des termes $P(Y = y \,/\, X = x)$ et $P(X = x)$, sachant que $P(Y)$ est une constante (indépendante de la réalisation x). Le premier terme $P(Y = y|X = x)$ décrit justement le processus d'observation et d'acquisition des données. L'hypothèse la plus courante (dont la validité reste à justifier) consiste à supposer l'indépendance conditionnelle des pixels (bruit non corrélé par exemple) :

$$P(Y = y \,/\, X = x) = \prod_s P(Y_s = y_s \,/\, X_s = x_s)$$

Cette écriture n'est plus valable lorsqu'il y a une convolution par la fonction de transfert du système d'acquisition, mais on peut montrer que le champ a posteriori reste markovien.

Par ailleurs, on fait sur le champ X recherché une hypothèse markovienne selon un voisinage $\mathcal{V}$ et un modèle donné dépendant de l'application. On peut alors écrire :

$$P(X = x) = \frac{\exp(-U(x))}{Z}$$

Si on revient maintenant à la distribution a posteriori, celle-ci s'exprime par :

$$\begin{aligned} P(X = x \,/\, Y = y) &\propto P(Y \,/\, X)P(X) \propto e^{\ln P(Y \,/\, X) - U(x)} \\ &\propto e^{-\mathcal{U}(x \,/\, y)} \end{aligned}$$

avec :

$$\mathcal{U}(x \,/\, y) = \sum_{s \in S} - \ln p(y_s \,/\, x_s) + \sum_{c \in \mathcal{C}} U_c(x) \tag{2.2}$$

Par conséquent, sous les hypothèses précédentes, on constate que la distribution a posteriori est une distribution de Gibbs et que donc le champ X conditionnellement à y est également un champ de Markov (théorème de Hammersley-Clifford). Ainsi, il est possible de simuler des réalisations de ce champ à l'aide de l'échantillonneur de Gibbs ou de l'algorithme de Metropolis. Mais la configuration x qui nous intéresse est celle maximisant la probabilité a posteriori, donc la réalisation la plus probable du champ de Gibbs, ou encore celle qui minimise l'énergie $\mathcal{U}(x \,/\, y)$. L'algorithme du recuit simulé décrit plus haut permet d'atteindre ce (ou ces) état(s) d'énergie minimale.

Cas de la restauration

Reprenons la démarche précédente et exprimons plus en détails l'énergie $\mathcal{U}(x \,/\, y)$ dans un cas particulier de restauration.

Dans le cas où le processus d'acquisition entraîne une dégradation de l'image sous forme d'un bruit blanc gaussien de variance σ^2, on a la probabilité conditionnelle suivante :

$$p(y_s \,/\, x_s) = \frac{1}{\sqrt{2\pi}\sigma} \exp - \frac{(x_s - y_s)^2}{2\sigma^2}$$

La probabilité a priori $P(X = x)$ permet d'introduire les contraintes que nous souhaitons imposer à la solution (i.e. que nous supposons pour le processus originel). En faisant l'hypothèse que X est markovien nous nous

restreignons à des contraintes locales, le plus souvent de régularité entre sites voisins. On choisit fréquemment un modèle avec des potentiels d'ordre 2 :

$$P(X = x) = \frac{1}{Z} \exp(-\beta \sum_{(s,t)\in\mathcal{C}_2} \phi(x_s - x_t))$$

On a alors l'énergie suivante correspondant à la distribution de Gibbs du champ a posteriori :

$$\mathcal{U}(x \ / \ y) = \sum_{s\in S} \frac{(x_s - y_s)^2}{2\sigma^2} + \beta \sum_{(s,t)\in\mathcal{C}_2} \phi(x_s - x_t) \tag{2.3}$$

Le champ X conditionnellement à y est donc un champ de Gibbs pour le même système de voisinage que X. La constante β pondère l'influence entre le terme d'attache aux données (cliques d'ordre 1) qui impose des niveaux de gris x_s de l'image restaurée proches de ceux y_s de la donnée bruitée, et le terme de régularisation (cliques d'ordre 2) qui impose une solution constituée de zones homogènes. Le modèle pour X peut être soit markovien gaussien, soit plus adapté à la restauration des contours avec une fonction ϕ appropriée. En effet, le modèle gaussien qui correspond à un fonction ϕ quadratique favorise des niveaux de gris proches pour des pixels voisins dans tous les cas. Or si on considère une image naturelle cet aspect est néfaste à proximité des contours car il favorisera la présence d'un dégradé. Aussi, de nombreuses fonctions ϕ ont été proposées pour modéliser les potentiels des cliques d'ordre 2 : $U_{c=(s,t)} = \phi(x_s - x_t)$. L'idée est de supprimer la pénalisation lorsque la variation de niveaux de gris est supérieure à une certaine valeur considérée comme représentant un contour. La partie 2.4 détaille ces aspects.

Cas de la segmentation

Dans ce contexte, le champ markovien X est défini sur un autre espace de configurations que Y car seulement quelques étiquettes sont considérées : $E = \{1, ..., m-1\}$ (correspondant aux différentes classes cherchées). Dans ce cas le processus de passage de X (champ des labels) à Y ne décrit pas tant le processus d'acquisition que l'apparence des classes dans l'image. Le terme $P(Y = y \ / \ X = x)$ traduit donc la probabilité de réalisation d'une configuration donnée connaissant son étiquetage (i.e. connaissant la classe de chaque pixel). En supposant l'indépendance des sites les uns par rapport aux autres, et en supposant que le niveau de gris y_s en un site s ne dépend que de l'étiquette x_s en ce site, on a :

$$P(Y = y \ / \ X = x) = \prod_s P(y_s \ / \ x_s)$$

Les valeurs des probabilités conditionnelles sont données par l'histogramme conditionnel des niveaux de gris pour une classe donnée. Par exemple, si on suppose que chaque classe i a une distribution gaussienne de moyenne μ_i et d'écart-type σ_i, on a :

$$P(y_s \ / \ x_s = i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp(-\frac{(y_s - \mu_i)^2}{2\sigma_i^2})$$

Si comme précédemment on fait une hypothèse markovienne sur X et qu'on se limite aux cliques d'ordre 2, on a :

$$P(X = x) = \frac{1}{Z} \exp(-\beta \sum_{(s,t)\in\mathcal{C}_2} \phi(x_s - x_t))$$

D'où l'énergie a posteriori :

$$\mathcal{U}(x \ / \ y) = \sum_s \frac{(y_s - \mu_{x_s})^2}{2\sigma_{x_s}^2} + \log\sqrt{2\pi}\sigma_{x_s} + \beta \sum_{(s,t)\in\mathcal{C}_2} \phi(x_s, x_t) \tag{2.4}$$

Le champ des étiquettes conditionnellement à y est markovien et d'énergie de Gibbs $\mathcal{U}(x \ / \ y)$. Là encore, comme pour la restauration, le terme d'ordre 1 exprime le respect des données (le niveau de gris doit correspondre à la classe), et le terme d'ordre 2 la contrainte de régularisation introduite. On choisit souvent un modèle de Potts pour X, ce qui donne une image segmentée avec de larges zones homogènes.

La figure 2.1 montre un exemple de segmentation d'une image satellitaire obtenue par le radar à ouverture synthétique ERS-1. L'utilisation du modèle de Potts pour le terme d'attache aux données donne des régions compactes.

Dans les deux applications précédentes il est nécessaire de pouvoir déterminer le ou les états d'énergie minimale qui correspondent au maximum de la probabilité d'un champ markovien. L'algorithme du recuit simulé présenté permet de trouver ces configurations. Nous reviendrons sur ce point en présentant d'autres estimateurs de la solution dans la section suivante.

2.2 Estimateurs dans un cadre markovien

2.2.1 Introduction

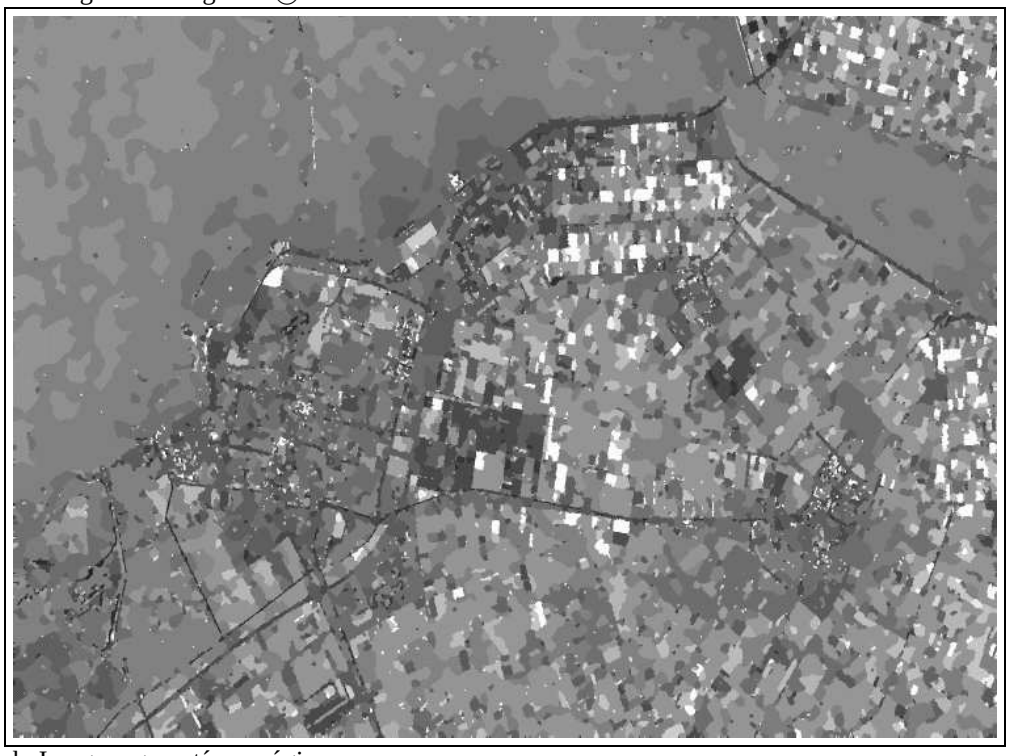
Nous avons vu précédemment comment il était possible d'utiliser le formalisme markovien à des fins de restauration et de segmentation. On se situe alors dans le cadre de données incomplètes (on parle aussi de champs de Markov cachés) car la réalisation dont on dispose est une réalisation bruitée (ou plus généralement vue à travers le système d'acquisition) du champ de Markov originel. En notant Y le champ dont on observe une réalisation, et X le champ initial, l'objectif est alors d'obtenir la meilleure réalisation $\hat{x}$ de X connaissant l'observation y, autrement dit, reconstruire x de manière optimale vis-à-vis d'un certain critère. Dans le précédent paragraphe, nous nous étions intéressés à la réalisation maximisant la probabilité a posteriori $P(X = x \ / \ Y = y)$, et nous avions vu un algorithme permettant d'obtenir cette réalisation : le recuit simulé. En réalité d'autres choix sont possibles, auxquels correspondent d'autres méthodes de résolution, et que nous allons aborder dans ce chapitre.

2.2.2 Modélisation bayésienne et fonction de coût

Si nous reprenons rapidement le raisonnement effectué précédemment, on peut écrire, en appliquant la règle de Bayes :

$$P(X = x \ / \ Y = y) = \frac{P(Y = y \ / \ X = x)P(X = x)}{P(Y = y)}$$

a. Image radar originale ©ERS-1



b. Image segmentée en régions

FIG. 2.1 – Exemple de segmentation markovienne sur une image ERS-1 du Flevoland.

On montre alors que sous certaines hypothèses (indépendance des sites dans la probabilité conditionnelle $P(Y \ / \ X)$ et hypothèse markovienne pour le champ X), la distribution a posteriori est une distribution de Gibbs et donc que le champ X conditionnellement à la donnée y est markovien. Cette propriété n'est pas nécessaire pour les notions suivantes, mais elle sera primordiale pour les algorithmes de résolution.

Le problème est alors de déterminer une estimation $\hat{x}$ optimisant un certain critère, où $\hat{x}$ est une fonction déterministe ϕ de la donnée y :

$$\hat{x} = \phi(y) \text{ avec } \phi : \Omega \rightarrow \Omega$$

L'estimation bayésienne procède alors comme suit. On se donne une fonction de coût, L définie de $\Omega \times \Omega$ dans R^+, qui représente le coût de remplacer x par $\phi(y)$, et qui possède les propriétés suivantes :

$$\forall x, x' \in \Omega \times \Omega : \quad \bullet L(x, x') \geq 0 \quad \bullet L(x, x') = 0 \Leftrightarrow x = x'$$

L'estimateur optimal, i.e la fonction ϕ optimale est alors la fonction minimisant l'espérance (notée $E[]$) du coût, c'est à dire :

$$E[L(X, \phi(y)) \ / \ Y = y] = \sum_{x \in \Omega} L(x, \phi(y)) P(x \ / \ y)$$

La fonction ϕ^{opt} minimise donc l'erreur moyenne conditionnellement à y, et l'estimateur optimal est $\hat{x} = \phi^{\text{opt}}(y)$.

Suivant les fonctions de coût envisagées, on obtient différents estimateurs et différentes méthodes de résolution associées.

2.2.3 Estimateur MAP

Considérons la fonction de coût suivante :

$$\begin{aligned} L(x, x') &= 1 \text{ si } x \neq x' \\ L(x, x') &= 0 \text{ sinon} \end{aligned}$$

Cette fonction consiste donc à pénaliser toute différence entre deux configurations, et ce, quel que soit le nombre de sites en lesquels elles diffèrent. Nous pouvons alors écrire :

$$\begin{aligned} E[L(X, \phi(y)) \ / \ y] &= \sum_{x \in \Omega} L(x, \phi(y)) P(x \ / \ y) \\ &= 1 - P(X = \phi(y) \ / \ y) \end{aligned}$$

Par conséquent, la fonction ϕ^{opt} minimisant l'espérance pour cette fonction de coût est celle qui maximimise la probabilité a posteriori :

$$\hat{x} = \phi^{\text{opt}}(y) = \text{Argmax}_{\phi}[P(X = \phi(y) \ / \ y)]$$

Il nous faut donc trouver la réalisation $\hat{x}$, fonction de y, maximisant la probabilité a posteriori $P(X \ / \ y)$. On parle de l'estimateur MAP (maximum a posteriori) ou de maximum de vraisemblance a posteriori.

On retrouve donc, avec cette fonction de coût en tout ou rien, la démarche intuitive que nous avions présentée dans la partie précédente 2.1.6 à savoir chercher la configuration maximisant la probabilité conditionnellement à la donnée disponible.

Les solutions algorithmiques associées à cet estimateur sont le recuit simulé et l'ICM que nous avons présentés dans le paragraphe 2.1.4, utilisés avec la distribution a posteriori avec paramètre de température.

2.2.4 Estimateur MPM

Considérons maintenant la fonction de coût définie par :

$$L(x, x') = \sum_{s \in S} L(x_s, x'_s) = \sum_{s \in S} \mathbf{1}_{x_s \neq x'_s}$$

La fonction de coût pénalise cette fois-ci une configuration proportionnellement au nombre de différences entre deux configurations. Elle paraît donc plus naturelle que la fonction de coût en tout ou rien précédente.

Dans le cas d'une fonction de coût définie comme somme de coûts en chaque site, on peut montrer le résultat suivant :

$$\begin{aligned} E[L(X, \phi(y)) \ / \ Y = y] &= \sum_{x \in \Omega} L(x, \phi(y)) P(x \ / \ y) \\ &= \sum_{s \in S} \sum_{x_s} L(x_s, \phi(y)_s) \sum_{x^s} P(x^s, x_s \ / \ y) \end{aligned}$$

Or $\sum_{x^s} P(x^s, x_s \ / \ y) = P(X_s = x_s \ / \ y)$, donc on peut faire apparaître les probabilités conditionnelles et espérances en chaque site s :

$$\begin{aligned} E[L(X, \phi(y)) \ / \ Y = y] &= \sum_{s \in S} \sum_{x_s} L(x_s, \phi(y)_s) P(X_s = x_s \ / \ y) \\ &= \sum_{s \in S} E[L(X_s, \phi(y)_s) \ / \ y] \end{aligned}$$

On passe donc de la probabilité conditionnelle globale d'une configuration à la probabilité conditionnelle en un site. Il s'agit d'une somme de termes positifs, et par conséquent la fonction ϕ optimale minimise en chaque site l'espérance conditionnelle du coût local $E[L(X_s, \phi(y)_s) \ / \ y]$. Ce résultat est valable pour toutes les fonctions de coût définies par une somme de coûts en chaque site.

Dans le cas de la fonction définie ci-dessus, on a alors comme précédemment :

$$E[L(X_s, \phi(y)_s) \ / \ y] = 1 - P(X_s = \phi(y)_s \ / \ y)$$

Ainsi, la valeur optimale de $\phi(y)$ ou de $\hat{x}$ en chaque site est telle que :

$$\hat{x}_s = \phi^{\text{opt}}(y)_s = \text{Argmax}_{\phi}[P(X_s = \phi(y)_s \ / \ y)]$$

i.e. on maximise en chaque site la marginale a posteriori $P(X_s \ / \ y)$.

On obtient donc des estimateurs du maximum a posteriori locaux, à calculer en chaque pixel contrairement à la recherche précédente qui était globale. L'estimateur est appelé maximum de vraisemblance a posteriori local ou maximum posterior marginal (pour maximum a posteriori de la marginale) abrégé en MPM.

D'un point de vue algorithmique, la taille de l'espace des configurations Ω ne permet pas un calcul direct des quantités $P(x_s \ / \ y)$. Aussi réalise-t-on en pratique des approximations de type Monte-Carlo. En effet, supposons que l'on soit capable de tirer des réalisations de X selon sa loi conditionnelle à y, et notons les $x(1), ..., x(N)$. Il est alors possible de calculer une approximation de l'estimateur MPM. Le tirage des réalisations ne pose quant à lui pas de problème particulier car nous avons vu dans le paragrpahe précédent 2.1.3 comment tirer des réalisations d'un champ de Gibbs avec l'échantillonneur de Gibbs et l'algorithme de Métropolis. Or sous les hypothèses rappelées en début de la section 2.2.1, la probabilité a posteriori $P(X \ / \ y)$ est une distribution de Gibbs.

Supposons donc que nous disposions de N échantillons de X tirés selon la loi a posteriori et que nous cherchions à estimer la distribution conditionnelle en chaque site $P(X_s = \lambda \ / \ y) \ \ \forall \lambda \in E$. Nous allons estimer cette quantité par la fréquence empirique de λ au site s dans les échantillons $x(k)$ de X, i.e :

$$\widehat{P}(X_s = \lambda \ / \ y) = \frac{1}{N} \sum_{k=1}^{N} \mathbf{1}_{x(k)_s = \lambda}$$

L'estimation au sens du MPM est alors donnée en chaque site en choisissant la valeur de x_s dans Ω maximisant $P(X_s \ / \ y)$.

2.2.5 Estimateur TPM

Considérons maintenant la fonction de coût définie par :

$$L(x, x') = \|x - x'\|^2 = \sum_{s \in S} (x_s - x'_s)^2$$

Il s'agit de l'erreur quadratique et elle pénalise cette fois-ci directement la somme des différences entre les deux configurations. Elle peut donc être plus adaptée dans certains cas que les précédentes, puisqu'elle tient compte non seulement du nombre de différences comme le MPM, mais aussi de leurs valeurs.

Dans ce cas, on a en utilisant le résultat établi précédemment :

$$E[L(X, \phi(y)) \ / \ Y = y] \quad = \quad \sum_{s \in S} E[(X_s - \phi(y)_s)^2 \ / \ y]$$

On cherche donc ϕ, telle que $E[(X_s - \phi(y)_s)^2 \ / \ y]$ soit minimum. Nous allons écrire cette espérance sous une nouvelle forme en utilisant la moyenne conditionnelle au site s, $\overline{x_s} = E[X_s \ / \ y] = \sum_{x_s \in E} x_s P(x_s \ / \ y)$:

$$\begin{aligned} E[(X_s - \phi(y)_s)^2 \ / \ y] \quad &= \quad \sum_{x_s \in E} (x_s - \overline{x_s})^2 P(x_s \ / \ y) + \sum_{x_s \in E} (\overline{x_s} - \phi(y)_s)^2 P(x_s \ / \ y) \\ &= \quad K + (\overline{x_s} - \phi(y)_s)^2 \end{aligned}$$

où K est une constante ne dépendant pas de ϕ donc n'intervenant pas dans la minimisation. Par conséquent, le minimum de l'erreur est atteint pour la fonction ϕ telle que :

$$\phi^{\text{opt}}(y)_s = E[X_s \ / \ y]$$

Cet estimateur consiste à prendre en chaque site la moyenne conditionnelle locale donnée par la loi a posteriori, d'où le nom de TPM (*Thresholded Posteriori Mean*).

D'un point de vue algorithmique, la démarche est similaire à celle effectuée dans le paragraphe précédent pour le MPM. On approxime l'espérance conditionnelle en chaque site par la moyenne empirique en ce site des N échantillons tirés selon la loi a posteriori :

$$\widehat{E}(X_s \ / \ y) = \frac{1}{N} \sum_{k=1}^{N} x(k)_s$$

L'estimation au sens du TPM est alors donnée en chaque site par sa moyenne empirique. Remarquons que cet estimateur est mal adapté à une problématique de segmentation car la moyenne des étiquettes n'a alors aucun sens.

2.2.6 Comparaison des estimateurs MAP, MPM, et TPM

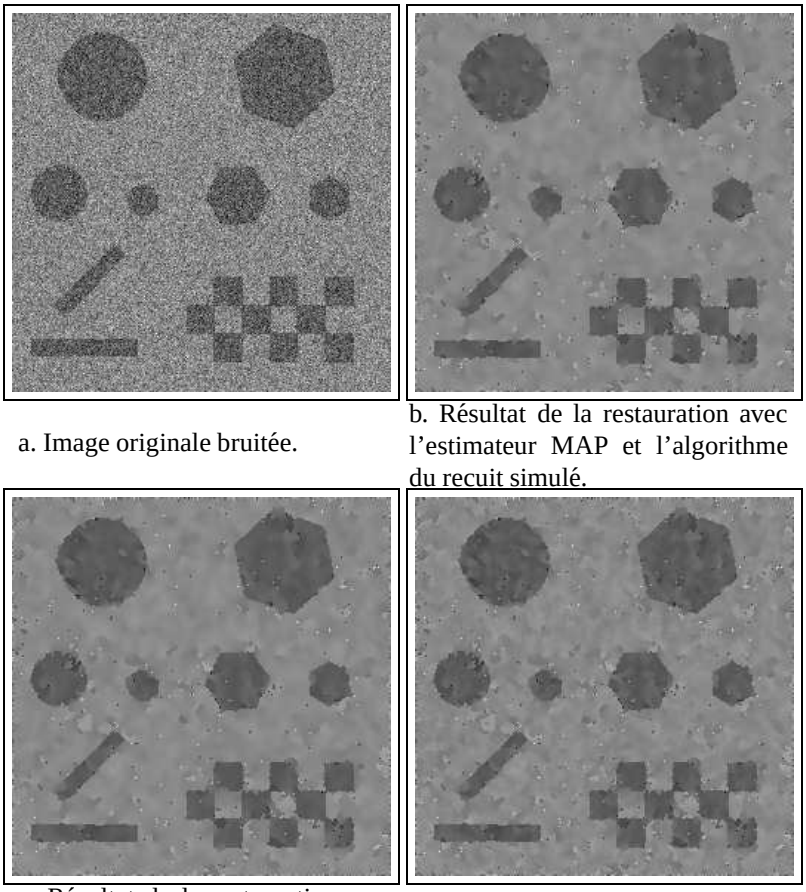
Nous comparons dans cette section les trois estimateurs dans le cadre de la restauration. Dans le cas du MAP, les résultats sont obtenus par ICM et par recuit simulé. L'image à restaurer (figure 2.2.a) est une image bruitée par un bruit blanc gaussien. L'énergie a posteriori utilisée s'écrit :

$$\mathcal{U}(x \,/\, y) = \sum_s \frac{(y_s - x_s)^2}{\sigma^2} + \sum_{c \in \mathcal{C}} +\beta\phi(x_r - x_s) \text{ avec } \beta > 0$$

$$\text{et } \phi(u) = \frac{u^2}{1+u^2}$$

Comme indiqué dans le paragraphe 2.1.6, cette fonction permet de seuiller les pénalités imposées par le terme de régularisation en présence de contours dans l'image.

Les paramètres utilisés sont fixés aux valeurs suivantes : $\sigma = 28$ (écart-type du bruit), $\delta = 10$ (saut en amplitude à partir duquel on considère qu'il y a un contour), $\beta = 0,5$ pondération de l'influence relative des deux termes). L'initialisation est donnée par l'image à restaurer. Pour tous les estimateurs, on réalise 600 itérations. Dans le cas du recuit simulé, la température initiale est de 6. Les résultats sont montrés sur la figure 2.2.



a. Image originale bruitée.

b. Résultat de la restauration avec l'estimateur MAP et l'algorithme du recuit simulé.

c. Résultat de la restauration avec l'estimateur MAP et l'algorithme ICM.

d. Résultat de la restauration avec l'estimateur TPM.

FIG. 2.2 – Comparaison des algorithmes ICM, Recuit Simulé et TPM en restauration.

On constate visuellement que le meilleur résultat est obtenu par le MAP du recuit simulé. Comme l'image originale est une assez bonne initialisation, il n'y a pas de grandes différences entre les algorithmes de recuit simulé et d'ICM pour l'estimateur du MAP. Ce n'est pas vrai dès que l'initialisation s'écarte du résultat à obtenir et les différences avec le recuit simulé peuvent être très importantes. Par ailleurs, on constate que l'estimateur TPM, qui est par définition plus local, donne un résultat plus bruité et moins régularisé. Cette analyse visuelle est confirmée par l'étude statistique qui peut être effectuée sur des zones homogènes de l'image. Le tableau ci-dessous donne les statistiques d'une zone sombre et d'une zone claire de l'image originale et pour les différents résultats de restauration. Les écarts-types les plus faibles sont obtenus pour l'estimateur MAP.

Statistiques sur l'image originale		
zone 1	135.7	27.8
zone 2	90.9	27.1
Estimateur MAP ICM		
zone 1	136.7	11.5
zone 2	91.6	6.9
Estimateur TPM		
zone 1	136.2	11.3
zone 2	91.6	6.9
Estimateur MAP recuit simulé		
zone 1	136.2	10.3
zone 2	91.9	6.1

TAB. 2.1 – Statistiques sur des zones homogènes pour les différents résultats de restauration

On notera également que des points isolés de faible ou fort niveau de gris subsistent dans l'image restaurée. Cela est lié à l'utilisation de la ϕ fonction qui ne régularise plus au-delà d'un certain seuil contrôlé par la valeur de δ.

En ce qui concerne les temps de calcul, les méthodes se répartissent comme suit : l'algorithme le plus rapide pour converger est sans conteste l'ICM, les algorithmes de recuit simulé et de TPM (ou MPM) étant à peu près équivalents. En effet, plus le nombre d'itérations est grand, meilleure est l'estimation de la moyenne a posteriori.

Les conclusions qui sont données ici ne sont pas nécessairement valables pour une application en segmentation. L'estimateur du TPM peut en effet dans certains cas donner de meilleurs résultats que le MAP. Par ailleurs, l'ICM peut s'avérer très utile lorsqu'on connaît une configuration proche de la configuration optimale.

2.3 Estimation des paramètres

2.3.1 Introduction

Le problème de l'estimation de paramètres (encore appelés hyperparamètres dans la littérature), revient très fréquemment en traitement d'image par champs de Markov. Donnons-en plusieurs exemples :

1. On se donne une réalisation d'un champ de Markov associé au modèle d'Ising, mais on ne connaît pas ses paramètres. Quels sont-ils ?
2. On veut généraliser ceci à une image de texture donnée dont on connaît le modèle sous-jacent (par exemple un modèle gaussien en 4-connexité) mais pas les paramètres, qui sont du type : moyenne locale, variance locale, poids de la régularisation locale. Quels sont-ils ? Leur connaissance pourrait bien en effet servir à la classification d'images composées de zones texturées en se basant sur l'estimation locale de tels paramètres. On classifierait alors selon les valeurs de ces attributs locaux.

3. On veut segmenter une image, et pour cela apprendre les paramètres de chaque classe, ainsi que le coefficient optimal du modèle de régularisation adapté à cette tâche. On sait en effet que le résultat d'une segmentation par estimateur MAP par exemple dépend fondamentalement du poids respectif de la régularisation par rapport à celui de l'attache aux données. Il faut donc là aussi estimer ce poids d'une façon optimale dans un sens à définir.

L'ensemble forme un problème réputé difficile. Nous ne démentirons pas ici la difficuté de ces problèmes, accentuée par le fait que de très nombreuses variantes ont été mises au point dans la littérature pour le résoudre. Certaines tentatives de comparaison de ces approches ont déja été effectuées [41]. Nous exposerons ici les variantes les plus fréquemment utilisées.
Elles se décomposent en deux classes fondamentales :

- le cas des données dites complètes, correspondant aux deux premiers problèmes cités plus haut : un échantillon d'une distribution de Gibbs est connu. Il s'agit de remonter aux paramètres de cette distribution.
- le cas des données dites incomplètes. Là non seulement le résultat de traitement est inconnu, mais les paramètres sont également à estimer.

2.3.2 Données complètes

Notons dans ce qui va suivre x une configuration observée relativement à une distribution de Gibbs donnée P_θ dont l'énergie associée puisse s'écrire sous la forme d'une fonction linéaire d'un paramètre θ, par exemple $U(x) = \theta\ \Phi(x)$, où Φ est un potentiel donné. Un principe naturel en vue de la recherche de θ est d'écrire la vraisemblance de la donnée x :

$$L(\theta) = P_\theta(x) = \frac{\exp\ -\theta\ \Phi(x)}{Z_\theta}$$

et de chercher par exemple la valeur de l'hyperparamètre $\hat{\theta}$ maximisant cette vraisemblance $L(\theta)$. Le problème essentiel est que l'on ne sait en général pas calculer exactement la fonction de partition Z_θ. Même pour des modèles aussi simples et fondamentaux que ceux d'Ising et de Potts, le résultat (analytique) est obtenu après des calculs excessivement compliqués [39, 33]. Dans les autres cas on sera donc amené :

- soit à effectuer des approximations de la fonction de partition globale au moyen des fonctions de partition conditionnelles locales (codages paragraphe 2.3.2, pseudo-vraisemblance paragraphe 2.3.2)
- soit à employer des algorithmes itératifs (gradient stochastique paragraphe 2.3.2) à partir de la vraisemblance exacte, mais dont il s'agit alors de prouver la convergence ainsi que le type d'optimum trouvé (local, global).

Méthode des codages

Le principe de la méthode des codages [4] est le suivant. Une fois défini un système de voisinage pour un champ de Markov, nous sommes capables de définir un certain nombre de sous réseaux, chacun formé de sites/pixels indépendants les uns des autres : chacun de ces sous réseaux est appelé un codage. Par exemple avec un voisinage en 4 connexité il existe deux codages comme le montre la figure ci dessous (à gauche), et 4 codages différents dans le cas de la 8-connexité (à droite) :

					3	4	3	4	3
2	1	2	1	2	2	1	2	1	2
1	2	1	2	1	3	4	3	4	3
2	1	2	1	2	2	1	2	1	2
					3	4	3	4	3

Nous allons poser le problème d'estimation dans le cadre de chaque codage. Pour un codage donné les différents sites/pixels le constituant sont indépendants les uns des autres puisqu'ils ne sont pas des voisins pour le champ de Markov de départ. La probabilité globale d'un codage se trouvera donc être le produit des probabilités individuelles de chacun des sites du codage. Or du fait de la structure de Markov du champ de départ cette probabilité individuelle se trouve être la probabilité conditionnelle locale du site/pixel dans le champ de Markov. Pour un codage Cod_n donné nous pouvons donc écrire :

$$P_\theta(\{X_s = x_s\}_{s\in \text{Cod}_n} \,/\, \{X_r = x_r\}_{r\notin \text{Cod}_n}) = \prod_{s\in \text{Cod}_n} P_\theta(X_s = x_s \,/\, V_s) \tag{2.5}$$

Etant donné la structure des probabilités locales telles qu'elles ont été décrites dans les chapitres précédents la fonction de vraisemblance devient calculable, puique les fonctions de partition conditionnelles locales le sont. Dans le cas où la dépendance des énergies locales est linéaire vis-à vis des paramètres, la log-vraisemblance associée,

$$\log P_\theta(\{X_s = x_s\}_{s\in \text{Cod}_n} \,/\, \{X_r = x_r\}_{r\notin \text{Cod}_n}) = -\theta \sum_{c\in\mathcal{C}} U_c(x) - \sum_{s\in \text{Cod}_n} \log(Z_s)$$

est une fonction concave du (des) paramètre(s), car somme de fonctions concaves [51]. Elle se prête donc bien à la recherche d'un optimum par une méthode classique de type gradient. On peut également montrer qu'il s'agit dans ce cas d'un simple problème de moindres carrés [13].

Pseudo-vraisemblance

Il apparaît en fait expérimentalement que la méthode des codages n'est pas fiable. La méthode du maximum de vraisemblance vrai paraît quant à elle incalculable. Des algorithmes ont cependant été étudiés pour tenter de résoudre ce problème [61], voir paragraphe 2.3.2. En fait nous allons utiliser une méthode intermédiaire qui aura de bonnes propriétés : la méthode du pseudo-maximum de vraisemblance [23]. Du maximum de vraisemblance vrai, nous allons prendre l'idée de travailler sur l'ensemble de l'image et non séparément sur des réseaux indépendants. De la méthode de codage, nous conservons l'idée de manipuler une fonction de vraisemblance produit des probabilités locales de chacun des sites/pixels. Cette fonction sera appelée pseudo-maximum de vraisemblance, et elle s'écrira :

$$PL_\theta(X = x) = \prod_{s\in S} P(X_s = x_s \,/\, V_s) \tag{2.6}$$

ou, encore, en considérant le logarithme de cette fonction, et l'expression de la probabilité locale de chaque site/pixel :

$$\log PL_\theta(X = x) = -\theta \sum_{c\in\mathcal{C}} U_c(x) - \sum_{s\in S} \log(Z_s) \tag{2.7}$$

Maintenant l'expression $-\log(Z_s)$ devient calculable, puisque reliée à la fonction de normalisation de la probabilité conditionnelle locale telle qu'elle a été décrite dans le paragraphe 2.1.2. Donc un raisonnement identique à celui du paragraphe précédent conduit au fait que la log-pseudo-vraisemblance est une fonction concave des paramètres lorsque l'énergie en dépend de façon linéaire. Les algorithmes usuels de type gradient ou gradient conjugué s'appliquent donc aussi ici naturellement à la recherche de l'optimum (qui est unique comme précédemment).
Il s'agit alors de qualifier la valeur des paramètres obtenus par cette méthode par rapport à la valeur vraie. Des résultats théoriques importants ont été obtenus à ce sujet [23, 24] : la méthode de la pseudo-vraisemblance est consistante et convergente.

Algorithme du gradient stochastique

Partons de la vraisemblance exacte du paramètre. Elle s'écrit bien sûr :

$$L(\theta) = P_\theta(x) = \frac{\exp\, -\theta\, \Phi(x)}{Z_\theta}$$

La valeur de l'hyperparamètre satisfaisant au principe du maximum de vraisemblance $\hat{\theta} = \arg\max_\theta P_\theta(x)$ doit donc vérifier l'équation :

$$\left(\frac{\partial \log P_\theta(x)}{\partial \theta}\right)_{\hat{\theta}} = -\,\Phi(x) - \left(\frac{\partial \log Z_\theta}{\partial \theta}\right)_{\hat{\theta}} = 0 \tag{2.8}$$

Comme $\mathbf{E}_{\hat{\theta}}[\Phi] = -\left(\frac{\partial \log Z_\theta}{\partial \theta}\right)_{\hat{\theta}}$ [51], cette valeur optimale est donc unique et doit satisfaire l'équation suivante :

$$\mathbf{E}_{\hat{\theta}}[\Phi] = \Phi(x) \tag{2.9}$$

Il s'agit là de ce que l'on appelle une équation stochastique. Comme remarqué en introduction de la Section 2.3.2 cette équation ne peut être résolue exactement, pour la raison que $\mathbf{E}_\theta[\Phi]$, qui dérive de la fonction de partition Z_θ, ne peut en général être calculé de façon analytique exacte. On est donc amené à employer des algorithmes basés sur un schéma itératif de type Newton-Raphson, mais adapté à ce cadre stochastique. Un schéma rigoureux conduirait à chaque étape (n) à :

$$\theta_{n+1} = \theta_n - \frac{\mathbf{E}_{\theta_n}[\Phi] - \Phi(x)}{\left(\frac{\partial(\mathbf{E}_\theta[\Phi])}{\partial \theta}\right)_{\theta_n}}$$

c'est-à-dire au schéma itératif :

$$\theta_{n+1} = \theta_n + \frac{\mathbf{E}_{\theta_n}[\Phi] - \Phi(x)}{\mathrm{var}_{\theta_n}(\Phi)}$$

L'idée est alors de remplacer les grandeurs statistiques mises en jeu par leurs valeurs empiriques approchées. Ainsi pour l'espérance du potentiel de régularisation Φ, on prendra sa moyenne empirique au cours d'une seule itération (c'est à dire la valeur effective obtenue !) d'un échantillonneur de Gibbs ou de Metropolis mené avec la valeur courante du paramètre. Quant à la variance de ce potentiel, on l'estime encore plus crûment par une grandeur positive fixée V ! On peut montrer que le prix à payer pour cette approximation est l'introduction d'un terme correctif supplémentaire en $\frac{1}{n+1}$ dans le schéma itératif à l'itération (n). C'est le principe de l'algorithme de gradient stochastique [61] :

$$\left\{ \begin{array}{ll} \theta_0 & \text{arbitraire} \\ x^{(0)} & \text{tiré au hasard} \end{array} \right. , \quad \theta_{n+1} = \theta_n + \frac{\Phi(x^{(n)}) - \Phi(x)}{(n+1)\, V} \quad \text{pour } n \geq 1 \tag{2.10}$$

Il est très important de noter ici que $x^{(n)}$ $(n \geq 1)$, échantillon de la distribution P_{θ_n} obtenu par une dynamique de Gibbs (ou de Metropolis) associée à la valeur courante du paramètre θ_n, est généré à partir de l'échantillon $x^{(n-1)}$

obtenu à l'itération précédente (c'est à dire lors de la valeur précédente du paramètre). On peut alors montrer que cet algorithme stochastique converge presque sûrement, en termes de probabilité, vers la valeur optimale $\hat{\theta}$ lorsque le coefficient V est choisi suffisamment grand.

2.3.3 Données incomplètes

Dans cette section, nous abordons le problème de l'estimation des paramètres dans le cas des données incomplètes dites encore manquantes.

Dans ce cas nous connaissons une observation y, échantillon de la v.a. Y, mais elle est appelée incomplète (ou dégradée), car reliée à une scène originale x, non-dégradée, dont le champ aléatoire correspondant sera noté X. La relation entre y et x s'effectue via une loi de probabilité conditionnelle représentant l'attache aux données (cf. 2.1.6) dont nous explicitons la dépendance p.r. à un paramètre λ positif :

$$P_\lambda(Y = y \ / \ X = x) = \frac{\exp \ - U_\lambda(y \ / \ x)}{Z_\lambda}$$

Ainsi, dans le cas d'un bruit blanc gaussien additif en restauration ou en déconvolution et l'approximation discrète finie, on a :

$$P_\lambda(Y = y \ / \ X = x) = \frac{\exp \ - \lambda \ || \ y - Rx \ ||^2}{Z_\lambda}$$

où R est la matrice associée à la réponse impulsionnelle de la fonction de flou (l'identité en restauration) et $Z_\lambda = \left(\sqrt{\frac{\pi}{2\lambda}}\right)^{|S|}$. On notera que la fonction de partition $Z_\lambda = \sum_{y \in \Omega} \exp \ - U_\lambda(y \ / \ x)$ est ici indépendante de la variable cachée x.

On suppose aussi que l'on dispose d'une connaissance a priori sur la scène à retrouver x, que ce soit en segmentation ou restauration, via la distribution de Gibbs suivante :

$$P_\theta(x) \frac{\exp \ - \theta \ \Phi(x)}{Z_\theta}$$

Nous aurons besoin dans la suite de ce chapitre d'utiliser la distribution de Gibbs a posteriori :

$$P_{\theta,\lambda}(X = x \ / \ Y = y) = \frac{\exp \ - U_\lambda(y \ / \ x) - \theta \ \Phi(x)}{Z_{\theta,\lambda}}$$

de fonction énergie $\mathcal{U}(x \ / \ y) = U_\lambda(y \ / \ x) + \theta \ \Phi(x)$, et de fonction de partition associée $Z_{\theta,\lambda}$.

Nous commencerons par généraliser, dans un cadre de Maximum de Vraisemblance (MV), la méthode de gradient stochastique vue au paragraphe précédent lorsque la loi d'observation (attache aux données) est complètement connue, c'est-à-dire que nous nous focaliserons sur l'estimation du meilleur paramètre de régularisation θ. Puis nous comparerons cette méthode à des variantes importantes similaires répertoriées dans la littérature.

Nous aborderons ensuite l'estimation des paramètres d'attache aux données, en particulier dans le cadre de la segmentation d'images. Cela nous permettra de décrire ensuite la seconde grande classe de méthodes adaptée à l'estimation des paramètres : l'EM (Expectation-Maximisation).

Gradient stochastique généralisé [62]

On va prouver dans cette partie que l'estimation du paramètre de régularisation connaissant la forme du potentiel a priori ne peut être dissociée dans la plupart des cas de la forme de l'attache aux données, supposée connue ici par simplicité, c'est à dire que λ est connu. On suppose accéder par exemple d'une façon ou d'une autre à la variance d'un bruit gaussien en restauration d'image bruitée (et à la réponse impulsionnelle du flou si l'on est en

déconvolution).
Dans le cas supposé où aucune information *a priori* sur le paramètre de régularisation n'est disponible, c'est-à-dire lorsque θ suit la distribution uniforme sur R, la vraisemblance de ce paramètre est la grandeur adéquate à étudier connaissant l'observation incomplète y et le paramètre λ. Elle peut se calculer de façon élégante [40] en remarquant que la loi jointe de l'observation et des variables cachées s'écrit :

$$\begin{aligned} P_{\theta,\lambda}(X=x,Y=y) &= P_\lambda(X=x \,/\, Y=y) P_\theta(X=x) \\ &= \frac{\exp -U_\lambda(y \,/\, x)}{Z_\lambda} \cdot \frac{\exp -\theta\, \Phi(x)}{Z_\theta} = \frac{\exp -U_\lambda(y \,/\, x) - \theta\, \Phi(x)}{Z_\lambda \,.\, Z_\theta} \end{aligned}$$

Si au moins une valeur de x n'annule pas $P(Y=y \,/\, X=x)$, on en déduit que :

$$L(\theta) = P_{\theta,\lambda}(Y=y) = \frac{P_{\theta,\lambda}(Y=y, X=x)}{P_{\theta,\lambda}(X=x \,/\, Y=y)} = \frac{Z_{\theta,\lambda}}{Z_\lambda \,.\, Z_\theta} \tag{2.11}$$

Toute valeur optimale du paramètre de régularisation $\hat{\theta}$ satisfait donc l'équation suivante :

$$\left(\frac{\partial \log L(\theta)}{\partial \theta}\right)_{\hat{\theta}} = \mathbf{E}_{\hat{\theta}}[\Phi] - \mathbf{E}_{\hat{\theta},\lambda}[\Phi] = 0 \Rightarrow \mathbf{E}_{\hat{\theta}}[\Phi] = \mathbf{E}_{\hat{\theta},\lambda}[\Phi] \tag{2.12}$$

où l'on rappelle que $\mathbf{E}_\theta[.]$ est l'espérance d'une v.a. sous la distribution de Gibbs a priori d'énergie $\theta\ \Phi(x)$, tandis que $\mathbf{E}_{\theta,\lambda}[.]$ signifie l'espérance statistique sous la distribution de Gibbs a posteriori. Notons également que les deux grandeurs statistiques $\mathbf{E}_{\theta,\lambda}[\Phi]$ et $\mathbf{E}_\theta[\Phi]$ sont des fonctions monotones décroissantes de θ, ce qui implique que plusieurs valeurs optimales de l'hyperparamètre peuvent exister (voir fig.2.3.3).

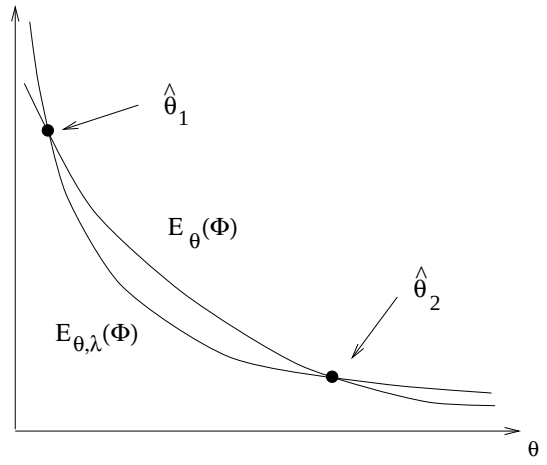


FIG. 2.3 – Valeur(s) optimale(s) $\hat{\theta}$ du paramètre de régularisation en données incomplètes.

Implémentation : Si l'on applique un schéma de Newton-Raphson à l'équation stochastique (2.12) on obtient directement :

$$\theta_{n+1} = \theta_n - \frac{\mathbf{E}_{\theta_n}[\Phi] - \mathbf{E}_{\theta_n,\lambda}[\Phi]}{\left(\dfrac{\partial(\mathbf{E}_{\theta_n}[\Phi] - \mathbf{E}_{\theta_n,\lambda}[\Phi])}{\partial \theta_n}\right)} = \theta_n + \frac{\mathbf{E}_{\theta_n}[\Phi] - \mathbf{E}_{\theta_n,\lambda}[\Phi]}{\mathrm{var}_{\theta_n}(\Phi) - \mathrm{var}_{\theta_n,\lambda}(\Phi)} \tag{2.13}$$

Il faut d'abord noter que dans ce schéma le dénominateur du dernier terme peut être de signe quelconque contrairement au cas des données complètes (2.10), ce qui est relié à l'existence de plusieurs solutions possibles en données

incomplètes. Maintenant, en s'inspirant du raisonnement effectué pour le gradient stochastique pour les données complètes, on s'aperçoit qu'il est nécessaire d'approximer des quantités statistiques (moyenne, variance) reliées aux distributions a posteriori et a priori. On a donc besoin d'échantillonner deux champs de Markov en général : celui lié à la régularisation pure (champ a priori) et le champ de Markov postérieur comprenant l'attachement aux données. Le schéma empirique qui en résulte naturellement est donc le suivant [62] :

$$\begin{cases} \text{un échantillon } x^{(n)} \text{ généré par } P_{\theta_n} \text{ (distribution a priori)} \\ \text{un échantillon } \tilde{x}^{(n)} \text{ généré par } P_{\theta_n,\lambda} \text{ (distribution a posteriori)} \end{cases}$$

avec la procédure d'évolution de l'hyperparamètre :

$$\theta_{n+1} = \theta_n + \frac{1}{n} \cdot \frac{\Phi(x^{(n)}) - \Phi(\tilde{x}^{(n)})}{(< \text{var}_{\theta_n}(\Phi) > - < \text{var}_{\theta_n,\lambda}(\Phi) >)}$$

Il faut noter en particulier la présence des estimateurs empiriques des variances $< \text{var}_{\theta_n}(\Phi) >$ et $< \text{var}_{\theta_n,\lambda}(\Phi) >$ (qui nécessitent donc en fait au moins deux échantillons pour chacune des distributions et à chaque étape du schéma d'évolution de l'hyperparamètre). Ainsi en segmentation, on doit échantillonner aussi bien la distribution a posteriori, ce qui sera par ailleurs utile en vue de la segmentation à convergence des paramètres par l'un des estimateurs MPM ou TPM, que la distribution a priori qui correspond au modèle de régularisation pur : modèles d'Ising, de Potts, généralisation, et ceci pour la valeur courante du (des) hyperparamètre(s).
La convergence de cette méthode est à montrer dans le cadre général des algorithmes stochastiques [35], qui dépasse largement l'objet de ce chapitre.

Comparaison avec d'autres variantes d'estimation

Dans ce paragraphe on présente d'autres méthodes usuelles d'estimation des hyperparamètres en données incomplètes, qui sont en fait reliées à d'autres estimateurs que le maximum de vraisemblance. Toute la subtilité réside ici dans le choix de la fonction de type vraisemblance à maximiser. On va voir en effet que de très légères différences conduisent à des formes fort différentes d'équations stochastiques, et donc des résultats d'estimation a priori fort différents.
On suppose dans les deux premiers cas qu'un résultat " optimal " de restauration (ou segmentation) x^* est connu à l'étape d'estimation où l'on se situe.

Une première approche consiste à utiliser la loi jointe de l'observation et du résultat connaissant la valeur courante des hyperparamètres [32, 16]. On peut alors écrire que le paramètre optimal doit satisfaire :

$$\hat{\theta} = \arg\max_{\theta} P_{\theta,\lambda}(X = x^* \ , \ Y = y)$$

En utilisant comme précédemment le fait que l'attache aux données (resp. la régularisation) est indépendante de l'hyperparamètre θ (resp. de λ), on obtient :

$$P_{\theta,\lambda}(X = x^* \ , \ Y = y) = P_{\lambda}(Y = y \ / \ X = x^*) P_{\theta}(X = x^*)$$

Et comme le premier terme (attache aux données) ne dépend pas de θ,

$$\hat{\theta} = \arg\max_{\theta} P_{\theta}(X = x^*) = \frac{\exp \ - \theta \ \Phi(x^*)}{[\ Z_{\theta} = \sum_{x \in \Omega} \exp \ - \theta \ \Phi(x) \ ]}$$

On retombe alors sur l'estimateur au sens du maximum de vraisemblance pour le champ de Markov a priori d'énergie $\theta \ \Phi(x)$ et pour la donnée ici complète x^*, c'est à dire : $\mathbf{E}_{\hat{\theta}}[\Phi] = \Phi(x^*)$. On peut donc utiliser une technique de gradient stochastique classique pour estimer ce paramètre (cf. paragraphe 2.3.2). On peut ensuite itérer

en effectuant le traitement désiré (restauration, segmentation) avec les nouvelles valeurs des hyperparamètres obtenues. On obtient donc une nouvelle configuration optimale à partir de laquelle on peut re-estimer les paramètres, et ainsi de suite [30, 64]. La méthode est théoriquement convergente vers les valeurs optimales des paramètres et de la configuration.

Une seconde approche consiste à utiliser la probabilité du résultat conditionnellement à l'observation et aux hyperparamètres. Dans ce cas, le paramètre optimal doit vérifier :

$$\hat{\theta} = \arg\max_{\theta} P_{\theta,\lambda}(X = x^* \ / \ Y = y)$$

En appliquant la seconde formule de Bayes et en utilisant le même argument d'indépendance des différentes lois de probabilités vis-à-vis des différents hyperparamètres mis en jeu, on obtient :

$$\hat{\theta} \quad = \quad \arg\max_{\theta} \frac{\exp \, - \lambda \, U(y \ / \ x^*) - \ \theta \ \Phi(x^*)}{[\ Z_{\theta,\lambda} = \sum_{x \in \Omega} \exp \, - \lambda \, U(y \ / \ x) - \ \theta \ \Phi(x) \ ]}$$

On tombe cette fois sur l'estimateur au sens du maximum de vraisemblance pour la distribution de Gibbs a posteriori d'énergie $\theta \ \Phi(x) + \lambda \ U(y \ / \ x)$ et pour la donnée complète x^*, c'est-à-dire : $\mathrm{E}_{\hat{\theta},\lambda}[\Phi] = \Phi(x^*)$. Les deux méthodes aboutissent donc à l'estimation d'un paramètre unique, et peuvent donc être implémentées avec une technique de gradient stochastique classique [61, 30]. Nous citerons également l'alternative des méthodes MCMCML (Monte Carlo Markov Chains Maximum Likelihood) [16].

Il est très intéressant de comparer les deux résultats précédents avec le véritable gradient stochastique généralisé vu précédemment. On rappelle que celui-ci permet de maximiser la probabilité du paramètre de régularisation conditionnellement à l'observation uniquement [62] :

$$\hat{\theta} = \arg\max_{\theta} P_{\theta,\lambda}(Y = y)$$

En toute rigueur, il est l'estimateur exact au sens du MV du paramètre de régularisation. Les deux autres cités précédemment ont essentiellement l'avantage d'être plus rapides, car ne nécessitant qu'un échantillonnage à chaque étape, et permettant de faire alterner le traitement d'image désiré avec l'estimation des paramètres.
En pratique, les trois estimateurs cités aboutissent à des valeurs du paramètre de régularisation semblables, ce qui provient du fait que les modèles adoptés pour l'attache aux données et pour la régularisation sont souvent choisis en cohérence avec l'observation fournie.

Estimation des hyperparamètres d'attache aux données : le cas de la segmentation

Supposons que l'on veuille estimer également les paramètres intervenant dans l'attache aux données, que nous regrouperons sous une variable λ. Ainsi, dans le cas important de la segmentation où le niveau de gris de chacune des m régions d'une image est supposée suivre une loi gaussienne de moyenne μ_i et de variance σ_i^2, λ est l'ensemble $\{\sigma_i, \ \mu_i\}_{i=1,m}$. Nous étudierons ici l'estimation de ces paramètres dans le cas gaussien par raison de commodité. La probabilité des observations conditionellement aux labels s'écrit donc :

$$P_\lambda(Y = y \ / \ X = x) = \prod_{s \in S} \frac{1}{\sqrt{2\pi}\sigma_{x_s}} \ \exp - \frac{1}{2 \ \sigma_{x_s}^2} \ (y_s - \mu_{x_s})^2$$

On peut écrire la distribution a posteriori sous la forme :

$$P_{\theta,\lambda}(X = x \ / \ Y = y) = \frac{\exp \, - \mathcal{W}_{\theta,\lambda}(x \ / \ y)}{Z_{\theta,\lambda}}$$

avec comme définition de la fonction énergie a posteriori effective :

$$\mathcal{W}_{\theta,\lambda}(x\ /\ y) = \sum_{s\in S}\left(\frac{1}{2\ \sigma_{x_s}^2}\ (y_s-\mu_{x_s})^2 +\ \log\sigma_{x_s}\right) + \theta\ \Phi(x) \tag{2.14}$$

et comme fonction de partition a posteriori associée $Z_{\theta,\lambda}$. La vraisemblance des paramètres d'attache aux données s'écrit donc selon le même principe que précédemment

$$L(\lambda) = P_{\theta,\lambda}(Y=y) = \frac{Z_{\theta,\lambda}}{(\sqrt{2\pi})^{|S|}Z_\theta}$$

Intéressons nous à maximiser cette quantité vis-à-vis d'un des paramètres de λ_i. Il faut remarquer ici que les fonctions énergie mises en jeu dépendent de façon non-linéaire de l'ensemble de paramètres $\lambda = \{\lambda_i\}$. D'où l'on déduit pour chacun des paramètres λ_i :

$$\frac{\partial \log L(\lambda)}{\partial\lambda_i} = \frac{\partial \log Z_{\theta,\lambda}}{\partial\lambda_i} = -\ \mathbf{E}_{\theta,\lambda}[\frac{\partial\mathcal{W}_{\theta,\lambda}}{\partial\lambda_i}] = 0 \tag{2.15}$$

Précisons qu'il s'agit bien de l'espérance a posteriori. Pour analyser la dépendance précise de l'énergie a posteriori vis-a-vis des paramètres $\{\sigma_i,\ \mu_i\}_{i=1,m}$, il est commode de re-écrire la formule (2.14) en employant les fonctions caractéristiques d'appartenance à chaque classe :

$$\mathcal{W}_{\theta,\lambda}(x\ /\ y) \quad = \quad \sum_{i=1}^{m}\left[\sum_{s\in S}\left(\frac{1}{2\ \sigma_i^2}\ (y_s-\mu_i)^2 + \log\sigma_i\right)\ .\ \mathbf{1}_{x_s=i}\right] + \theta\ \Phi(x)$$

Il en résulte des formules importantes pour la suite :

$$\left\{\begin{array}{lcl} \dfrac{\partial\mathcal{W}_{\theta,\lambda}(x\ /\ y)}{\partial\mu_i} & = & \dfrac{1}{\sigma_i^2}\ \displaystyle\sum_{s\in S}\ (\mu_i - y_s)\ \mathbf{1}_{x_s=i} \\ \dfrac{\partial\mathcal{W}_{\theta,\lambda}(x\ /\ y)}{\partial\sigma_i} & = & \dfrac{1}{\sigma_i}\ \displaystyle\sum_{s\in S}\left(\frac{-\ (y_s-\mu_i)^2}{\sigma_i^2}\ +1\right)\ \mathbf{1}_{x_s=i} \end{array}\right. \tag{2.16}$$

• Examinons d'abord le cas d'un paramètre de moyenne μ_i donné. On peut montrer que (si $\sigma_i < +\infty$) :

$$\forall i\in[1..m]\ \ ,\ \ \mu_i = \frac{\displaystyle\sum_{s\in S} y_s\ P_{\theta,\lambda}(X_s=i)}{\displaystyle\sum_{s\in S} P_{\theta,\lambda}(X_s=i)} \tag{2.17}$$

L'interprétation physique en est claire : on obtient le barycentre des observations en chaque site de l'image pondérées par la probabilité a posteriori d'avoir une classe déterminée.
• De la même façon, on obtient pour les variances de chaque classe :

$$\forall i\in[1..m]\ \ ,\ \ \sigma_i^2 = \frac{\displaystyle\sum_{s\in S}(y_s-\mu_i)^2\ P_{\theta,\lambda}(X_s=i)}{\displaystyle\sum_{s\in S} P_{\theta,\lambda}(X_s=i)} \tag{2.18}$$

les μ_i pouvant être calculés par la formule (2.17). Le résultat est similaire à une variance empirique, mais avec pondération en chaque site par la probabilité a posteriori que ce site ait le label étudié.

En définitive on aboutit donc à des équations (2.17) et (2.18) auto-cohérentes, de forme bien plus complexe que celles des gradients stochastiques simple et généralisé. On aimerait pouvoir les remplacer par des formes itératives simples : c'est ce qui va justifier l'emploi des méthodes de type EM, à adapter dans le cadre gibbsien défini ici.

Expectation-maximization (EM)

Récapitulons les résultats précédents obtenus concernant l'estimation au Maximum de Vraisemblance des hyperparamètres dans le cas des données incomplètes :

• pour l'hyperparamètre de régularisation nous sommes arrivés à l'équation stochastique :

$$\mathbf{E}_{\hat{\theta}}[\Phi] = \mathbf{E}_{\hat{\theta},\lambda}[\Phi]$$

• pour les paramètres d'attache aux données en segmentation avec m classes nous sommes arrivés au système d'équations :

$$\forall i \in [1..m] \ , \ \begin{cases} \mu_i = \dfrac{\sum\limits_{s\in S} y_s \ P_{\theta,\lambda}(X_s = i)}{\sum\limits_{s\in S} P_{\theta,\lambda}(X_s = i)} \\ \sigma_i^2 = \dfrac{\sum\limits_{s\in S} (y_s - \mu_i)^2 \ P_{\theta,\lambda}(X_s = i)}{\sum\limits_{s\in S} P_{\theta,\lambda}(X_s = i)} \end{cases}$$

Nous en avons conclu qu'il serait désirable de résoudre ces équations de manière itérative. Par exemple si les probabilités a posteriori $P_{\theta,\lambda}(.)$ étaient connues (ou apprises) à une étape donnée on pourrait les injecter dans la première équation à condition de pouvoir estimer pour toute valeur de θ l'espérance a priori du potentiel de régularisation ainsi que dans le deuxième système. Ces deux arguments vont nous fournir les principes de l'EM, avec un certain nombre de résultats théoriques très puissants à la clé.

La méthode EM est par essence itérative. Des résultats théoriques très généraux montrent que la vraisemblance des paramètres estimés croît à chaque itération de l'algorithme associé [3, 42]. On pourra également consulter en particulier [9] pour une présentation claire de cette méthode. De plus cette méthode permet d'une certaine façon l'optimisation séparée des hyperparamètres d'attache aux données et de celui de régularisation. En effet elle considère essentiellement la probabilité jointe de l'observation et des données cachées (un étiquetage de l'image par exemple), c'est-à-dire le produit de la loi d'observation et de la probabilité a priori.
Supposons que les paramètres θ_n et λ_n soient connus à l' étape courante n. On suppose aussi que l'on a accès d'une façon ou d'une autre aux statistiques liées à la loi a posteriori des données cachées courantes (ppartie *Expectation* de l'EM, voir plus loin). On définit alors la notation :

$$Q(\theta, \lambda, \theta_n, \lambda_n) = \mathbf{E}_{\theta_n,\lambda_n}[\log P_{\theta,\lambda}(X = x \ , \ Y = y)]$$

La partie optimisation (*Maximisation*) de l'EM consiste à rechercher :

$$(\theta_{n+1}, \lambda_{n+1}) \quad = \quad \arg\max_{\theta,\lambda} Q(\theta, \lambda, \theta_n, \lambda_n)$$

Du fait de la séparabilité de la loi jointe, la fonction objectif $Q(\theta, \lambda, \theta_n, \lambda_n)$ s'écrit :

$$Q(\theta, \lambda, \theta_n, \lambda_n) = \mathbf{E}_{\theta_n,\lambda_n}[\log P_\lambda(Y = y \ / \ X = x) + \log P_\theta(X = x)]$$

Cela correspond donc à l'optimisation séparée :

$$\begin{cases} \theta_{n+1} & = & \arg\max\limits_{\theta} \mathbf{E}_{\theta_n,\lambda_n}[\log P_\theta(X = x)] \\ \lambda_{n+1} & = & \arg\max\limits_{\lambda} \mathbf{E}_{\theta_n,\lambda_n}[\log P_\lambda(Y = y \ / \ X = x)] \end{cases}$$

On rappelle que la vraisemblance des paramètres $L(\theta_n, \lambda_n)$ croît en fonction de l'itération n par cette méthode ce qui aboutit à terme à un optimum local pour les valeurs des paramètres.
Examinons plus précisément ce qui se passe pour chacune des catégories de paramètres :

• en ce qui concerne le paramètre de régularisation, cela correspondra à :

$$\mathbf{E}_{\theta_n,\lambda_n}[\frac{\partial \log P_\theta(X=x)}{\partial \theta}] = \mathbf{E}_{\theta_n,\lambda_n}[-\Phi - \frac{\partial \log Z_\theta}{\partial \theta}] = 0$$

c'est-à-dire en vertu de résultats supposés maintenant acquis :

$$\mathbf{E}_{\theta_{n+1}}[\Phi] = \mathbf{E}_{\theta_n,\lambda_n}[\Phi] \tag{2.19}$$

Supposons que l'on sache d'une manière ou d'une autre calculer ou estimer l'espérance a posteriori du potentiel de régularisation pour la valeur courante des paramètres. Ainsi, en pratique, on approxime cette quantité par sa moyenne empirique au cours d'un échantillonneur de Gibbs (ou de Metropolis) pris pour la valeur courante des paramètres [3] . On est alors ramené à un problème d'estimation du paramètre de régularisation pour la donnée complète $\mathbf{E}_{\theta_n,\lambda_n}[\Phi]$! On peut donc appliquer la technique du pseudo-maximum de vraisemblance [8] ou bien encore celle du gradient stochastique [63] pour re-estimer la nouvelle valeur du paramètre de régularisation.

• pour les paramètres d'attache aux données, on a vu plus haut que

$$\frac{\partial \log P_\lambda(Y=y \ / \ X=x)}{\partial \lambda_i} = -\frac{\partial \mathcal{W}_{\theta,\lambda}(x)}{\partial \lambda_i}$$

Le principe EM s'écrit donc

- pour les paramètres de moyenne :

$$\begin{aligned} \mathbf{E}_{\theta_n,\lambda_n}[\frac{\partial \mathcal{W}_{\theta,\lambda}}{\partial \mu_i}] &= \frac{1}{\sigma_i^2} \sum_{s\in S} (\mu_i - y_s)\, P_{\theta_n,\lambda_n}(X_s = i) = 0 \\ \Rightarrow \forall i \in [1..m] \ , \ \mu_i(n+1) &= \frac{\sum_{s\in S} y_s \, P_{\theta_n,\lambda_n}(X_s = i)}{\sum_{s\in S} P_{\theta_n,\lambda_n}(X_s = i)} \end{aligned} \tag{2.20}$$

- pour les paramètres de variances :

$$\begin{aligned} \mathbf{E}_{\theta_n,\lambda_n}[\frac{\partial \mathcal{W}_{\theta,\lambda}}{\partial \sigma_i}] &= \sum_{s\in S} \left(\frac{-(y_s - \mu_i)^2}{\sigma_i^2} + 1 \right) P_{\theta_n,\lambda_n}(X_s = i) = 0 \\ \Rightarrow \forall i \in [1..m] \ , \ \sigma_i(n+1)^2 &= \frac{\sum_{s\in S} (y_s - \mu_i(n+1))^2 \, P_{\theta_n,\lambda_n}(X_s = i)}{\sum_{s\in S} P_{\theta_n,\lambda_n}(X_s = i)} \end{aligned} \tag{2.21}$$

Deux remarques insistantes :

1. Ce sont bien les distributions a posteriori qui sont mises en jeu dans l'estimation itérative (eqs. 2.20 et 2.21). Elles sont donc à re-estimer à chaque itération n. Une manière bien naturelle est de remplacer ces probabilités à l'étape courante du processus EM par la fréquence empirique d'apparition des labels lors d'une série d'échantillonnages de la distribution de Gibbs a posteriori courante menés à l'aide d'un échantillonneur de type Gibbs ou Metropolis [8]. Notons pour cela N le nombre d'itérations effectué avec l'échantillonneur ainsi sélectionné :

$$\underset{(n)}{\bullet} \quad \underset{N}{- - - - - - - - - - - -} \quad \underset{(n+1)}{\bullet}$$

[3] Ceci correspond à l'étape Estimation de l'algorithme EM.

Notons également $N_s(i)$ le nombre de fois que le label i a été tiré en un site s au cours de l'ensemble de ces N échantillonnages. On peut donc écrire : $P_{\theta_n,\lambda_n}(X_s = i) \approx \frac{N_s(i)}{N}$, et il en résulte les estimations empiriques des paramètres de moyennes et de variance de chaque classe :

$$\forall i \in [1..m] \ , \ \left\{ \begin{array}{rcl} \mu_i(n+1) & = & \dfrac{\sum\limits_{s\in S} y_s \ N_s(i)}{\sum\limits_{s\in S} N_s(i)} \\ \\ \sigma_i(n+1)^2 & = & \dfrac{\sum\limits_{s\in S} (y_s - \mu_i(n+1))^2 \ N_s(i)}{\sum\limits_{s\in S} N_s(i)} \end{array} \right.$$

De la même façon, notons $x^{(n)}(k)$ la série d'images échantillons ainsi obtenue pour $k = 1..N$. On obtient pour l'espérance a posteriori du potentiel de régularisation

$$\mathbf{E}_{\theta_n,\lambda_n}[\Phi] \approx \frac{1}{N} \left[\sum_{k=1}^{N} \Phi(x^{(n)}(k)) \right]$$

C'est donc ensuite que l'on procéde à l'étape d'estimation EM comme indiqué plus haut, à partir de ces valeurs empiriques a posteriori.

2. On retrouve le caractère itératif que nous voulions anticiper à propos des équations plus haut. Là aussi, à convergence, on doit satisfaire à la forme exacte des équations (2.17) et (2.18) obtenues par le Maximum de Vraisemblance.

Variantes Une variante importante, appelée ICE ou Iterative Conditional Estimation est la suivante [41]. Considérons les N échantillons $x^{(n)}(k)$ obtenus précédemment à une étape courante n de l'EM. On pourrait songer à estimer moyenne et variance des classes pour chacun d'eux considéré comme segmentation courante, puis prendre la moyenne empirique (arithmétique) des valeurs ainsi obtenues

$$\forall i \in [1..m] \ , \ \left\{ \begin{array}{rcl} \mu_i(n+1) & = & \dfrac{1}{N} \sum\limits_{k=1}^{N} \left(\dfrac{\sum\limits_{s\in S} y_s \ \mathbf{1}_{x_s^{(n)}(k)=i}}{\sum\limits_{s\in S} \mathbf{1}_{x_s^{(n)}(k)=i}} \right) \\ \\ \sigma_i(n+1)^2 & = & \dfrac{1}{N} \sum\limits_{k=1}^{N} \left(\dfrac{\sum\limits_{s\in S} (y_s - \mu_i(n+1))^2 \ \mathbf{1}_{x_s^{(n)}(k)=i}}{\sum\limits_{s\in S} \mathbf{1}_{x_s^{(n)}(k)=i}} \right) \end{array} \right.$$

Cette estimation est différente de l'EM. On peut montrer qu'elle correspond en fait à l'estimateur de la moyenne a posteriori des paramètres.

Conclusion pour l'estimation en données incomplètes

On voit donc au terme de cette partie que l'estimation en données incomplètes se prête à un nombre très riche de variantes. On pourrait ainsi parfaitement juger préférable d'estimer certains types de paramètres comme moyenne et variance par l'EM, tandis que l'on adopterait la méthode de Lakhsmanan-Derin pour le paramètre de régularisation, ou réciproquement ! Ces variantes méritent encore d'être examinées et comparées entre elles de façon exhaustive, dans la lignée de l'approche suivie dans [41].

Il faut aussi préciser que les méthodes utilisant un échantillonnage de la distribution a posteriori (comme l'EM) permettent en même temps de fournir une série de configurations échantillons se prêtant favorablement à la segmentation ou à la restauration dans le cadre des estimateurs de type TPM ou MPM, lorsque la convergence des paramètres vers leur valeur optimale est supposée atteinte.

2.4 Processus de bords

L'un des grands domaines d'application des modèles markoviens est la restauration comme nous l'avons mentionné au paragraphe 2.1.6. Connaissant une réalisation y, on cherche une solution x minimisant le critère :

$$\mathcal{U}(x \,/\, y) = U_1(x, y) + \beta U_2(x)$$

Le terme de régularisation $U_2(x)$ permet alors d'agir directement sur les dérivées de la solution cherchée. Dans le cas d'un potentiel quadratique, il s'agit d'une régularisation de Tikhonov qui entraîne un fort lissage de la solution au détriment des discontinuités souvent naturellement présentes dans l'image.

De nombreux travaux ont porté sur l'introduction de discontinuités et sur les avantages potentiels que cela impliquait dans la recherche de la solution. Nous nous limiterons ici à des cliques d'ordre deux, c'est-à-dire à l'introduction de contraintes sur la dérivée première de la solution cherchée. Nous mettons en évidence l'équivalence entre processus de bords et certaines fonctions avant de présenter quelques algorithmes déterministes d'optimisation.

2.4.1 Processus de bords explicites et implicites

Dès l'article fondateur de Geman et Geman [19], l'introduction de processus de bords permettant de désactiver l'effet de lissage entre deux sites est envisagé. Celui-ci, que nous noterons B dans la suite est défini sur une grille duale de celle de l'image et est souvent décomposé en deux sous champs, l'un représentant les interactions verticales B^v et l'autre les interactions horizontales B^h. Il s'agit souvent (mais pas nécessairement) de processus de bords booléens, prenant la valeur 1 en présence de discontinuités et 0 en son absence. Le terme de régularisation s'écrit alors [6] [18] :

$$U_2(x, b) = \sum_{(s,t)\in\mathcal{C}^v} [(x_s - x_t)^2(1 - b^v_{st}) + \gamma b^v_{st}] + \sum_{(s,t)\in\mathcal{C}^h} [(x_s - x_t)^2(1 - b^h_{st}) + \gamma b^h_{st}] \tag{2.22}$$

D'une façon intuitive, en l'absence de discontinuités, on a $b_{st} = 0$ et on retrouve un potentiel quadratique visant à lisser la solution. En revanche, en présence de discontinuité ($b_{st} = 1$), la pénalité est arbitraire et vaut γ[4] de façon à limiter le nombre de discontinuités introduites par le modèle. Ce modèle (modèle de la membrane mince -*weak membrane*-) est très simple et n'introduit pas d'interactions entre les éléments du processus bords.

En réalité, il n'est pas nécessaire d'introduire explicitement un processus de bords et un choix de fonction judicieux peut procurer le même résultat. En effet, notons $\phi^v(x_s, x_t, b^v_{st})$ le terme $(x_s - x_t)^2(1 - b^v_{st}) + \gamma b^v_{st}$ (et de même pour ϕ^h), alors on cherche la solution (x, b) telle que $U_2(x, b)$ soit minimale. Or :

$$\begin{aligned}
\min_{(x,b)} U_2(x, b) &= \min_{(x,b)} \sum_{(s,t)\in\mathcal{C}^v} \phi^v(x_s, x_t, b^v_{st}) + \sum_{(s,t)\in\mathcal{C}^h} \phi^h(x_s, x_t, b^h_{st}) \\
&= \min_{x} \sum_{(s,t)\in\mathcal{C}^v} \min_{b^v_{st}} \phi^v(x_s, x_t, b^v_{st}) + \sum_{(s,t)\in\mathcal{C}^h} \min_{b^h_{st}} \phi^h(x_s, x_t, b^h_{st}) \\
&= \min_{x} \sum_{(s,t)\in\mathcal{C}^v} \psi(x_s - x_t) + \sum_{(s,t)\in^h} \psi(x_s - x_t)
\end{aligned}$$

[4] Ce paramètre peut varier pour des modèles anisotropes ou non stationnaires (on a alors γ^v_{st} et γ^h_{st}).

en notant :

$$\psi(u) = \min(u^2, \gamma)$$

Par conséquent, la quadratique est remplacée dans ce modèle par une quadratique tronquée (figure 2.4). L'utilisation d'un processus de bords explicite est dans ce cas équivalente à l'utilisation de la fonction ψ (on parle de processus de bords implicite). La valeur de γ détermine à partir de quelle valeur du gradient on introduira une discontinuité.

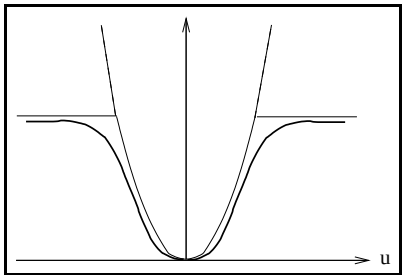


FIG. 2.4 – Fonctions quadratique, quadratique tronquée et la fonction de régularisation proposée par Geman et McClure préservant les discontinuités.

Un très grand nombre de fonctions de régularisation préservant les discontinuités ont été proposées et étudiées (on pourra se référer à [10] pour une étude comparative). La fonction suivante, qui présente l'intérêt d'être dérivable partout, est très utilisée en restauration [20] :

$$\phi(u) = \frac{u^2}{1 + u^2}$$

Notons qu'il est intéressant d'utiliser des fonctionnelles convexes pour la recherche de la solution, mais la prise en compte des discontinuités repose sur le comportement de $\frac{\phi'(u)}{2u}$.

Le choix d'un processus de bords implicite ou explicite va donner lieu à différents algorithmes de minimisation. En effet, le recuit simulé permettant d'accéder au minimum global de l'énergie est un algorithme long et coûteux. Il peut dans de nombreux cas être remplacé par un algorithme déterministe pour accélérer la recherche de la solution.

2.4.2 Algorithmes de minimisation

Nous donnons ici le principe des quelques algorithmes déterministes :

– GNC [6] : Blake et Zisserman ont proposé un algorithme déterministe appelé le GNC (Graduated Non Convexity) pour l'optimisation du critère (sous sa forme implicite). Le principe de l'algorithme consiste à approximer le critère par une fonction convexe qui permet de définir une bonne solution initiale (calculée par une descente de gradient par exemple, puisque la solution est dans ce cas unique). Puis le critère est modifié graduellement perdant sa propriété de convexité pour se rapprocher du critère initial. À chaque étape une solution est trouvée de façon déterministe en utilisant pour initialisation la solution donnée par l'étape précédente. Si des preuves de convergence peuvent être obtenues dans certains cas particuliers, cette démarche n'assure cependant pas de trouver le minimum global pour toutes les fonctions d'énergie.

– MFA [18] : une autre approche est le recuit par champ moyen MFA (Mean Field Annealing) qui utilise cette fois un processus bords explicite. L'algorithme met en œuvre une descente de température au cours de laquelle à chaque étape une solution (image restaurée et processus de bords) est estimée au sens du champ moyen. On a des expressions explicites pour le processus de bords, tandis que x est estimée de façon itérative.

– Artur et Legend [11] : Charbonnier a également proposé deux algorithmes déterministes appelés Artur et Legend pour l'optimisation. Il utilise un processus de bords explicite et exploite le fait qu'à b fixé, le critère

est quadratique en x (donc convexe et minimisable rapidement par une descente de type gradient), tandis qu'à x fixée, il existe une expression analytique du processus de bords.

Malgré de nombreuses différences (notamment sur les propriétés de convergence et sur leur généralité), il s'agit toujours d'algorithmes itératifs minimisant des suites d'énergies (GNC ou MFA) ou faisant varier les variables auxiliaires (le processus de bords pour Artur et Legend) au cours des itérations. Une comparaison est effectuée dans [11] et [65].

2.5 Quelques applications des champs markoviens

Cette partie présente quelques applications des champs markoviens en traitement d'images pour illustrer les potentialités de ce domaine. On s'intéressera dans un premier temps à des applications de bas niveau (analyse de textures et segmentation) avant de présenter des applications manipulant des graphes construits à partir de primitives plus complexes (régions, segments).

2.5.1 Applications sur le graphe des pixels

Cette première partie présente trois applications de bas-niveau, i.e qui travaillent sur le graphe des pixels :

- La première est une analyse de textures s'appuyant sur un modèle markovien ; elle permet de discriminer différentes textures de l'image à l'aide des paramètres du champ extraits ; un exemple d'applications pour la détection des zones urbaines en imagerie satellitaire est donné ;
- La seconde est un simple exemple de segmentation appliquée en imagerie radar mettant en évidence la souplesse de ce modèle pour prendre en compte des statistiques très variées dans les images et détaillant quelques aspects pratiques de segmentation ; on se placera dans un cadre supervisé pour le paramètre de régularisation, c'est à dire que celui-ci sera fixé de façon empirique ;
- La troisième application présentée est un schéma de fusion markovien, relativement général qui permet de combiner plusieurs sources d'informations ; un exemple est donné dans le cas de l'analyse d'images satellitaires NOAA dans plusieurs bandes spectrales.

Analyse de textures

Nous nous intéressons dans cette partie à l'utilisation des modèles markoviens pour discriminer différents types de textures dans les images. L'idée est de se placer dans le cas de données complètes, i.e de faire l'hypothèse que l'image dont nous disposons est la réalisation d'un champ markovien, et d'extraire les paramètres de ce champ. La variabilité des paramètres en fonction du type de textures devant alors permettre de réaliser une classification de l'image.

Nous avons déjà vu au paragraphe 2.3.2 un certain nombre de techniques (méthode des codages, maximum de pseudo-vraisemblance, gradient stochastique) pour calculer les paramètres d'un champ markovien. Toutes ces méthodes, qui ne présupposent pas de modèle pour le champ, sont assez lourdes à mettre en œuvre. Nous allons supposer ici que nous nous situons dans le cadre d'un champ markovien gaussien pour lequel nous pouvons obtenir une expression exacte des paramètres. Dans le cas d'un champ markovien gaussien, nous pouvons écrire l'énergie $U(x)$ sous la forme :

$$U = \frac{1}{T}\left(\lambda \sum_{s\in S}(x_s-\mu)^2 + \sum_{c=(s,t)\in\mathcal{C}}(x_s-x_t)^2\right)$$

Avec cette écriture, les paramètres caractérisant le champ et que nous cherchons à estimer sont donc la température T (correspondant à une sorte de "variance généralisée"), la moyenne μ et la pondération du terme d'attache aux données de l'énergie, λ.

La méthode des queues de comètes La méthode que nous allons voir a été proposée dans [14]. Considérons la probabilité conditionnelle en un site :

$$P(X_s = x_s \ / \ x_t, t \in \mathcal{V}_s) = \frac{1}{Z(V_s)} \exp\{-\frac{1}{T}\left(\lambda(x_s - \mu)^2 + \sum_{t \in \mathcal{V}_s}(x_s - x_t)^2\right)\}$$

En notant n le nombre de sites voisins de s, et m_s la moyenne locale des niveaux de gris des voisins du site s, $m_s = \dfrac{\sum_{t \in \mathcal{V}_s} x_t}{n}$, on peut montrer qu'on a :

$$\begin{aligned} P(X_s = x_s \ / \ x_t, t \in \mathcal{V}_s) &= \frac{1}{Z(m_s)} \exp\{-\frac{n+\lambda}{T}\left(x_s - \frac{1}{n+\lambda}(nm_s + \lambda\mu)\right)^2\} \\ &= P(X_s = x_s \ / \ m_s) \end{aligned}$$

On a donc remplacé le conditionnement local par l'ensemble des voisins de s, par une seule variable conditionnante m_s, ce qui améliorera la robustesse des estimateurs. En effet, le nombre de sites concernés par un conditionnement par m_s sera bien plus grand que celui des sites concernés par une configuration $V_s = (x_t, t \in \mathcal{V}_s)$ du voisinage.

Nous constatons que $P(X_s \ / \ m_s)$ est une loi gaussienne, de moments (moyenne et variance) suivants :

$$\begin{aligned} \mathbf{E}(X_s \ / \ m_s) &= \frac{nm_s + \lambda\mu}{n+\lambda} \\ \text{var}(X_s \ / \ m_s) &= \frac{T}{2(n+\lambda)} \end{aligned}$$

L'espérance et la variance de X_s conditionnellement à m_s étant fonction des paramètres λ, T et μ que nous recherchons, il nous suffit d'estimer ces moments de façon empirique. La variance ne dépendant pas de la valeur conditionnante m_s, on notera le moment théorique par σ^2. Pour faire ces estimations, on construit l'image dite des "queues de comètes" en raison de son aspect, qui est simplement l'image des fréquences normalisées des niveaux de gris x_s conditionnellement à la moyenne m_s du voisinage. Si on met les valeurs de m_s en colonne et celles de x_s en ligne, chaque ligne de l'image des queues de comètes représente $P(X_s \ / \ m_s)$, i.e une gaussienne de moyenne $\dfrac{nm_s + \lambda\mu}{n+\lambda}$ et de variance $\dfrac{T}{2(n+\lambda)}$.

On considère alors pour la variance, l'estimateur suivant :

$$\widehat{\sigma^2} = \sum_{m_s} P(m_s)\widehat{\text{var}}(X_s \ / \ m_s)$$

en notant $\widehat{\text{var}}(X_s \ / \ m_s)$ l'estimation empirique de la variance faite selon une ligne des queues de comètes. Cet estimateur permet d'accorder à chaque probabilité conditionnelle locale une importance proportionnelle au nombre d'échantillons qui la constituent et entraîne une plus grande robustesse de l'estimation.

En ce qui concerne l'espérance, nous avons la relation suivante :

$$\begin{aligned} \mathbf{E}(X_s \ / \ m_s) &= \frac{n}{n+\lambda}m_s + \frac{\mu}{n+\lambda} \\ &= \alpha m_s + \beta \end{aligned}$$

Par conséquent, les espérances des probabilités conditionnelles se situent sur une droite. L'estimation empririque des moyennes $\widehat{\mathbf{E}}(X_s \ / \ m_s)$ permet de faire une estimation aux moindres carrés des paramètres α et β de la droite.

Une fois estimés empiriquement $\widehat{\sigma^2}$, $\widehat{\alpha}$, $\widehat{\beta}$, on peut déduire λ, μ, et T par les relations :

$$\begin{aligned}\widehat{\sigma}^2 &= \frac{\widehat{T}}{2(n_s+\widehat{\lambda})}\\ \widehat{\alpha} &= \frac{n_s}{n_s+\lambda}\\ \widehat{\beta} &= \frac{\widehat{\lambda}\widehat{\mu}}{n_s+\widehat{\lambda}}\end{aligned}$$

En explicitant l'estimation aux moindres carrés des paramètres de la droite, on peut exprimer directement λ, μ, et T en fonction des moments d'ordre 1 et 2, conditionnés ou non. On retrouve dans ce cas pour μ, la moyenne empirique des x_s.

Application à la détection des zones urbaines

Cet exemple d'application est tiré de la thèse de X. Descombes [14].

On peut utiliser les résultats précédents pour analyser les textures présentes sur une image satellitaire SPOT. L'image étant par essence non stationnaire (sinon l'analyse aurait peu d'intérêt !), le calcul des paramètres se fait localement, sur une fenêtre glissante centrée en chaque pixel. La fiabilité des estimateurs s'accroît avec la taille de la fenêtre, en même temps, et de façon antagoniste, que le risque de considérer des mélanges de textures différentes à l'intérieur de la fenêtre d'étude. Une solution pour remédier à ce problème peut être de ne considérer que les échantillons les plus représentés. En cas de mélange, les échantillons appartiendront à la texture la plus présente dans la fenêtre.

Le paramètre de température, est un bon indicateur du milieu urbain qui se présente sur une image SPOT sous une forme assez texturée, type "poivre et sel" avec alternance de niveaux de gris faibles et forts. En effet, ce paramètre qui mesure en quelque sorte le chahut de la zone, a des valeurs plus élevées dans les régions urbaines. Il permet d'obtenir une bonne discrimination du milieu urbain.

2.5.2 Segmentation

Nous allons aborder dans cette partie une application très classique des champs markoviens qui est la segmentation. Nous commençons par rappeler le principe de la segmentation markovienne en prenant l'exemple des images radar [53], puis présentons une méthode de fusion dans un cadre markovien [14].

Segmentation d'une image radar *Cet exemple d'application est tiré de la thèse de F. Tupin [53].*

La modélisation est similaire à celle qui a été présentée dans le paragraphe 2.1.6. Ecrivons à nouveau les deux termes intervenant dans la probabilité a posteriori. Pour la probabilité du champ des observations conditionnellement au champ des étiquettes, en supposant l'indépendance des pixels, on a :

$$P(Y \,/\, X = x) = \prod_s P(Y_s = y_s \,/\, X_s = x_s)$$

Les images radar sont des images très bruitées par le phénomène de speckle. En revanche, le processus d'acquisition est bien modélisé statistiquement et on a l'expression suivante pour une image radar en amplitude :

$$p(Y_s = y_s \,/\, X_s = i) = \frac{2L^L}{\mu_i^L \Gamma(L)} y_s^{(2L-1)} \exp(-\frac{L y_s^2}{\mu_i})$$

avec L un paramètre du système connu[5] appelé nombre de vues, Γ la fonction Gamma, et μ_i les moyennes en intensité (carré de l'amplitude) des différentes classes i considérées.

Le champ des étiquettes est supposé markovien avec un modèle de Potts qui vise à obtenir des zones homogènes compactes sur l'image segmentée :

$$P(X = x) = \frac{1}{Z} \exp(-\beta \sum_{(s,t)\in\mathcal{C}_2} \phi(x_s - x_t))$$

avec $\beta > 0$, $\phi(0) = 1$ et $\phi(x) = 1 \quad \forall x \neq 0$.

Le champ a posteriori résultant est donc markovien et son énergie s'écrit :

$$\mathcal{U}(x \,/\, y) = L \sum_s (\frac{y_s^2}{\mu_{x_s}} + \ln \mu_{x_s}) + \sum_{c=(s,t)} \phi(x_s - x_t)$$

En choisissant l'estimateur MAP, la solution est obtenue par recuit simulé avec une décroissance géométrique en température, et une température initiale fixée arbitrairement. Le choix des classes se fait de la façon suivante. On se fixe le nombre de classes (15 dans les illustrations ci-dessous) et on applique un algorithme de k-moyennes dont le résultat[6] sert à calculer les valeurs des moyennes en intensité μ_i des différentes classes. Notons que ces classes n'ont pas de contenu sémantique et que la segmentation correspond ici à un "découpage" de l'image.

Le choix de β qui pondère l'influence entre attache aux données et régularisation se fait de façon ad hoc après quelques essais. L'augmentation de β entraîne une augmentation de la taille des zones obtenues par la segmentation. Il serait bien sûr possible d'estimer ce paramètre à l'aide d'une des méthodes décrites au chapitre 3. Notons que ce modèle n'est pas adapté à la préservation des cibles ponctuelles et des lignes qui sont des configurations de forte énergie pour le champ des étiquettes (il faut donc une forte attache aux données pour que ces configurations subsistent dans le résultat final). Il est bien sûr possible de prendre en compte un champ externe pour mieux respecter les lignes par exemple [54] ou d'utiliser un modèle plus approprié [15].

Le tableau ci-dessous 2.2 résume les paramètres utilisés et la figure 2.1 montre le résultat de la segmentation.

TAB. 2.2 – Valeurs des paramètres de la segmentation

Température initiale	5
Facteur de décroissance géométrique	0,95
Paramètre de régularisation β	0,4
Nombre de classes	15
Nombre d'itérations pour les k-moyennes	20

Schéma de fusion dans un cadre markovien • Principe du schéma de fusion

Nous nous plaçons maintenant dans le cas où plusieurs sources d'information sont disponibles pour réaliser la classification de l'image. Ces différentes sources peuvent provenir de l'extraction de différents paramètres à partir d'une même image ou directement de plusieurs capteurs.

Notons y le vecteur d'attributs correspondant aux différentes sources d'informations $y = (y^1, ..., y^K)$ avec K le nombre de données (ou canaux) et M le nombre de classes $E = \{\lambda_1, ..., \lambda_M\}$. La probabilité a posteriori s'écrit :

[5] Pour les produits PRI du satellite ERS1 $L = 3$.

[6] Le résultat des k-moyennes est très bruité à cause du bruit multiplicatif présent sur les images radars et l'absence de modes dans l'histogramme. Il ne peut donc pas dans ce cas être utilisé directement comme résulat de segmentation.

$$p(X \ / \ Y = y) \propto p(Y \ / \ X)p(X)$$

Si nous faisons l'hypothèse que les sources sont **indépendantes** entre elles, et les pixels de chaque source entre eux, on a :

$$\begin{aligned} p(Y \ / \ X = x) &= \prod_{s \in S} P(Y_s \ / \ X_s = x_s) \\ &= \prod_{s \in S} P(\{Y_s^1, Y_s^2, ..., Y_s^K\} \ / \ X_s = x_s) \\ &= \prod_{s \in S} P(Y_s^1 \ / \ X_s = x_s)...P(Y_s^K \ / \ X_s = x_s) \\ &= \prod_{s \in S} \prod_{k=1}^{K} P(Y_s^k \ / \ X_s = x_s) \end{aligned}$$

Le potentiel d'attache aux données résultant s'exprime alors sous forme d'une somme des potentiels individuels de chaque source :

$$U_s(y_s = (y_s^k)_k \ / \ x_s = \lambda) = \sum_k U_s(y_s^k \ / \ \lambda)$$

Les différentes sources ne renseignant pas de la même façon sur toutes les classes, il est possible d'introduire des coefficients de pondération exprimant la confiance (la fiabilité) qu'on veut accorder à chacune des sources par rapport à une classe. On notera $\gamma_{(k,\lambda)}$ la "confiance" accordée à la source k pour la classe λ. Il faut par ailleurs ne favoriser aucune classe, donc les coefficients doivent vérifier $\sum_k \gamma(k, \lambda) = 1 \ \ \forall \lambda$. L'expression de l'attache aux données s'écrit alors :

$$U_s(y_s = (y_s^k)_k \ / \ x_s = \lambda) = \sum_k \gamma_{(k,\lambda)} U_s(y_s^k \ / \ \lambda)$$

En pratique, il n'est pas toujours nécessaire de faire des modélisations statistiques compliquées pour calculer les potentiels d'attache aux données. Souvent des potentiels très simples, linéaires par morceaux, permettent d'obtenir de bons résultats. On attache un potentiel faible (typiquement 0) à la plage de niveaux de gris qui correspond à la classe considérée et un potentiel élevé (typiquement 1) ailleurs. Cette définition peut se faire de façon supervisée en analysant l'histogramme ou de façon automatique par une recherche des modes de l'histogramme de l'image (analyse multi-échelle [1], recuit simulé sur l'histogramme [7], etc.).

La définition des coefficients de fiabilité des sources est souvent plus problématique. On se limite en général à des remarques de bon sens en affectant $\gamma_{k,\lambda} = 0$ lorsque la source k n'est pas significative pour la classe λ, 0.5 si l'information délivrée est approximative, et 1 si la source est pertinente (avant normalisation).

• Application à l'analyse des images SPOT

Cet exemple d'application est tiré de la thèse de X. Descombes [14]. On trouvera d'autres exemples dans [54] [1].

Nous décrivons ici l'application de ce schéma à l'analyse de plusieurs canaux délivrés par le satellite NOAA. Il s'agit de 5 canaux de basse résolution (1.1km) correspondant aux domaines visible, proche infra-rouge, moyen infra-rouge et 2 canaux thermiques. Les classes qu'on cherche à discriminer sont les suivantes : mer, nuages, continent sans relief, continent avec relief et icebergs.

Une première étape consiste à définir les "sources" que nous allons utiliser. La méthodes des queues de comète que nous avons présentée plus haut permet en effet de déduire de chaque image, 3 images de paramètres (moyenne locale, température et paramètre d'attache aux données λ). Au total 20 images sont donc disponibles, dont les plus

significatives pour notre objectif, sont sélectionnées : l'image 1 (visible), l'image de température et l'image de moyenne associées, l'image 3 (moyen infra-rouge) et l'image de température associée (soit 5 en tout).

Pour chacune de ces images, une analyse supervisée de l'histogramme est effectuée, permettant de définir les potentiels (linéaires par morceaux) et la pertinence du canal pour chaque classe. Le terme d'attache aux données de l'énergie a posteriori est alors défini comme mentionné précédemment. Quant au terme de régularisation, les classes recherchées étant relativement compactes, on utilise un modèle de Potts.

2.6 Applications sur des graphes de primitives

Comme nous l'avons mentionné dans la section 2.1, le formalisme markovien est défini sur tout graphe et son champ d'applications est donc bien plus vaste que la simple grille des pixels. De nombreux problèmes se prêtent à la manipulation de primitives plus complexes, soit pour des raisons de rapidité comme dans le premier exemple que nous décrivons ci-dessous, soit parce qu'il s'agit d'un problème plus proche de l'interprétation d'image, difficile à traiter au niveau du pixel et nécessitant l'introduction d'informations de haut niveau comme dans le second exemple décrit.

2.6.1 Graphes de régions : application à la segmentation d'une image d'IRM cérébrale

Cet exemple d'application est tiré de la thèse de T. Géraud [21].

Pour accélérer la segmentation des images, plusieurs applications partent d'une sur-segmentation de l'image qui est ensuite améliorée en fusionnant les régions. Cette fusion, qui procure la segmentation finale, peut se faire dans un cadre markovien. Le graphe est construit à partir des régions de la sur-segmentation, chaque région correspondant à un sommet du graphe et la relation de voisinage étant définie par la relation d'adjacence entre les régions. Le terme d'attache aux données dépend alors des attributs de la région (niveau de gris moyen des pixels la constituant, moments d'ordre supérieur, etc.) et le terme de régularisation dépend de l'application, un potentiel de Potts pouvant être utilisé lorsqu'on essaye de trouver des zones relativement compactes.

L'objectif de l'exemple décrit ici est de réaliser une segmentation d'images IRM cérébrales. Les classes considérées sont la matière grise, blanche, le liquide céphalo-rachidien, les ventricules et une classe ASI représentant les autres structures internes (noyaux caudés, thalamus, putamen) qui sont difficiles à segmenter et auxquelles on s'intéresse particulièrement dans cette application. Le nombre de pixels de ces images volumiques ($256\times256\times128$) limite l'utilisation de méthodes markoviennes en raison du temps de calcul. Par contre, l'utilisation d'un graphe de régions construit à partir d'une sur-segmentation, en réduisant drastiquement le nombre de sites permet de réaliser un recuit simulé à un coût raisonnable.

• **Sur-segmentation :** L'étape de sur-segmentation 3D est réalisée par un algorithme calculant la ligne de partage des eaux sur l'image du gradient après fermeture morphologique pour réduire le nombre de bassins. L'image résultat est constituée de zones de niveaux de gris homogènes, auxquelles on associe les attributs suivants : volume (noté vol), niveau de gris moyen (qui sera l'observation y du champ des données), coordonnées du centre du bassin. Le résultat de cette méthode appliquée à l'image 2.5.a est montré sur la figure 2.5.b.

• **Relaxation markovienne :** Un graphe est construit comme indiqué précédemment à partir de la sursegmentation (fig. 2.5.c). On associe aux arcs du graphe la surface d'adjascence entre les deux régions (notée surf). Les potentiels du champ markovien sont alors définis comme suit :

– Terme d'attache aux données :

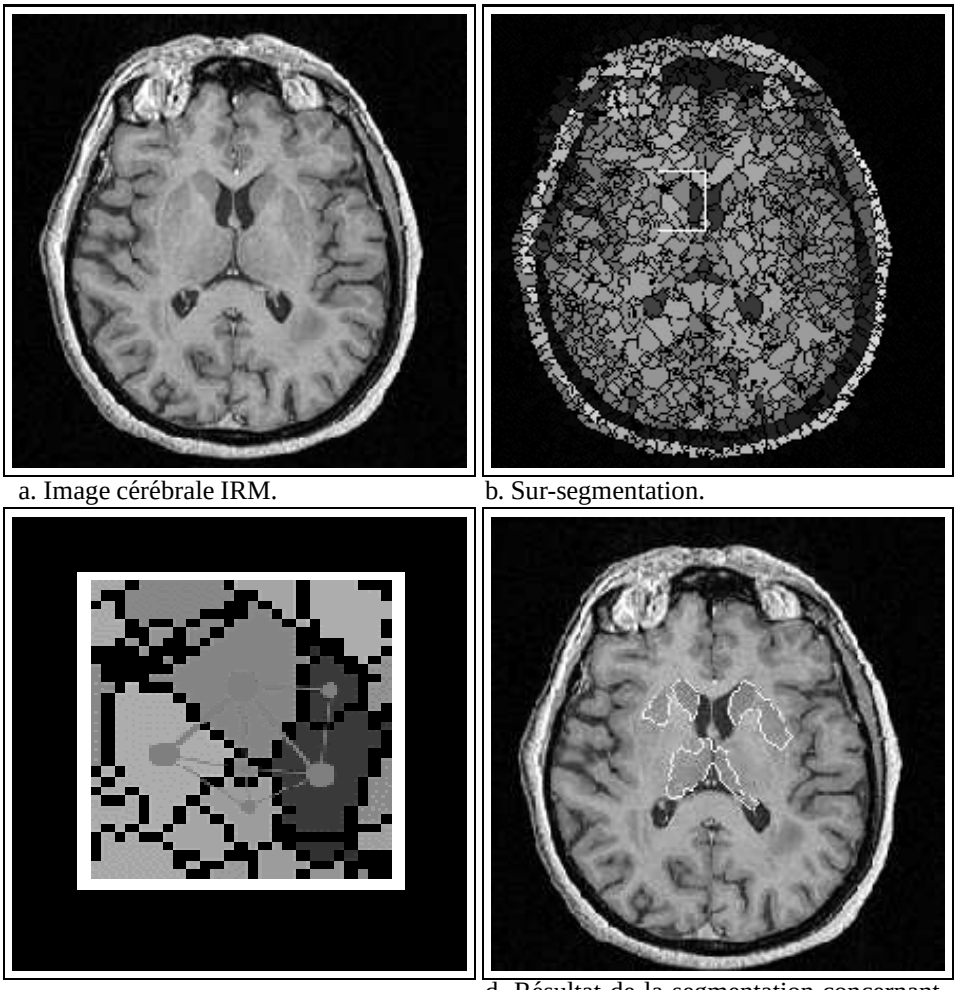
$$P(Y_s \,/\, x_s) = \exp\left(-\sum_s \frac{\mathrm{vol}_s}{2\sigma^2_{x_s}}(y_s - \mu_{x_s})^2\right)$$

(déduit d'une étude statistique des classes [21])

– Terme de régularisation :

$$U_{c=(s,t)}(x) = \text{surf}_{s,t} Q_{(x_s,x_t)}$$

Seuls les potentiels des cliques d'ordre 2 sont choisis non nuls. La matrice Q est une matrice d'adjascence permettant de pondérer les voisinages entre classes suivant qu'ils sont favorisés ou non [50].



a. Image cérébrale IRM. b. Sur-segmentation.

c. Extrait du graphe d. Résultat de la segmentation concernant les ASI

FIG. 2.5 – Etapes de la segmentation sur le graphe des régions

Le critère utilisé est un critère MAP et la solution est obtenue par un recuit simulé. Le temps de calcul est de moins d'une minute pour un graphe de 32 000 sites. Les résultats pour les ASI sont montrés sur la figure 2.5.c.

2.6.2 Graphes de segments : application à la détection du réseau routier

Cet exemple d'application est tiré de la thèse de F. Tupin [53].

Nous nous intéressons dans cette application à la détection du réseau routier et hydrographique dans le cas des images radar. Le bruit de speckle présent sur ces images ne permet pas d'obtenir de très bons résultats de bas

niveau et les détecteurs de lignes même adaptés à ce type d'imagerie ont des taux de fausses alarmes élevés si on veut obtenir des taux de détection suffisants.

L'idée est donc de faire suivre l'étape de bas niveau de détection des lignes par une étape de plus haut niveau, dans laquelle on injectera des informations a priori sur la forme des routes. Le cadre markovien, par l'intermédiaire du terme a priori se prête bien à l'introduction de connaissances sur les objets recherchés, à condition que celles-ci puissent s'exprimer de façon locale à l'échelle du graphe. Dans le cas du réseau cette hypothèse (qui assure que le champ soit markovien) est vérifiée, puisse que la pratique montre qu'il nous suffit d'informations locales au niveau des segments pour prendre une décision (présence ou absence de réseau routier).

La démarche adoptée pour la détection du réseau est donc la suivante. L'étape de bas niveau permet de détecter des segments candidats. Parmi ceux-ci, certains appartiennent aux objets à détecter, quand d'autres sont de fausses détections. On fait alors l'hypothèse que les segments détectés et toutes les connexions possibles entre ces segments contiennent le réseau routier. Les connexions "possibles" ne sont pas toutes les connexions, mais les connexions raisonnables : entre des segments suffisamment proches, et à peu près "alignés" par exemple. L'ensemble de ces segments (ceux détectés et les connexions) consituent les sommets du graphe (voir figure 2.6). La relation de voisinage entre deux segments est définie par le partage d'une extrémité par ces 2 segments. Les figures ci-dessous illustrent les étapes de la construction du graphe sur un extrait d'image radar (fig. 2.9).

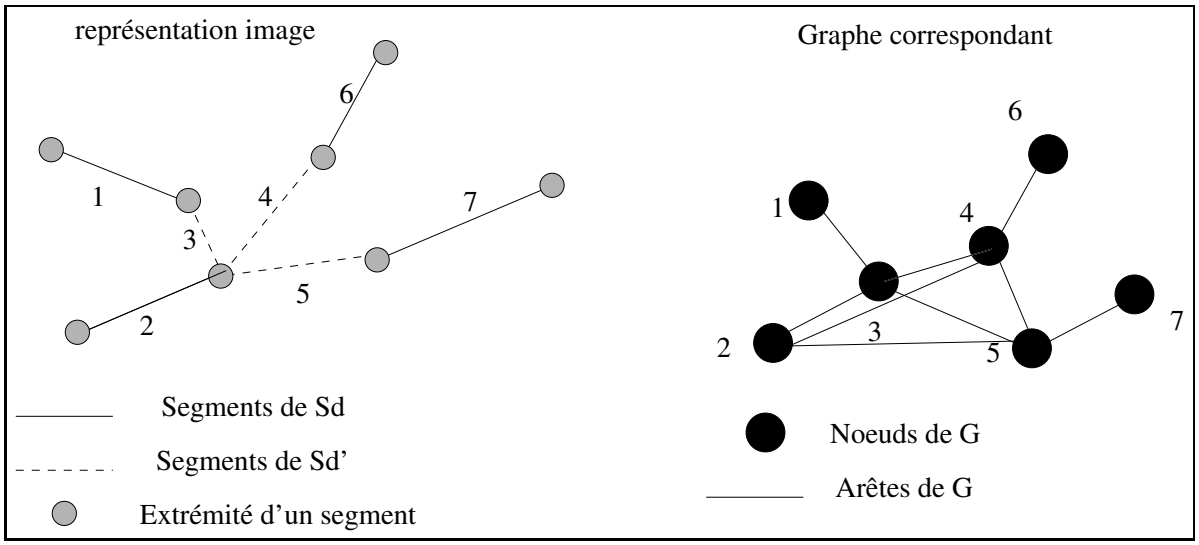
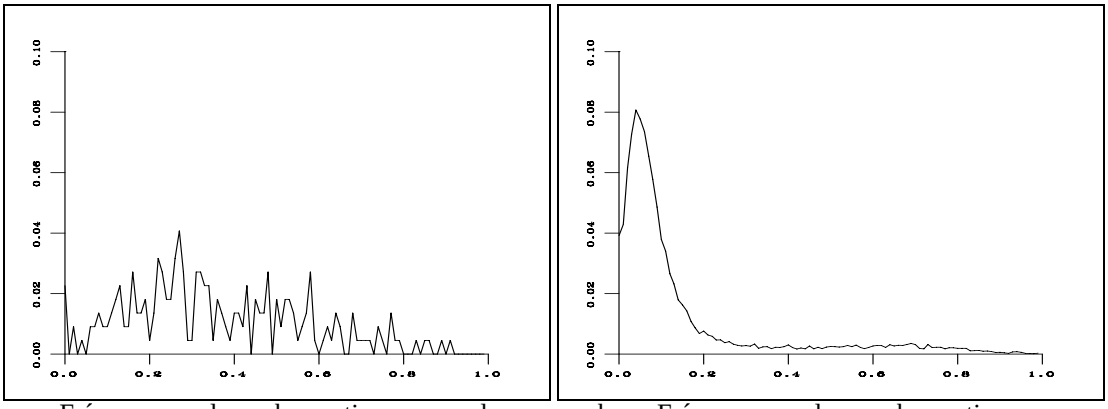


FIG. 2.6 – Construction du graphe de segments

Une fois ce graphe construit, on définit un champ d'observation associé Y et un champ d'étiquettes X (les étiquettes étant simplement 0 pour "non-route" et 1 pour "route") dont on cherche la configuration optimale au sens du critère MAP.

Les termes de l'énergie sont définis de la façon suivante :

– Terme d'attache aux données : l'observation en un site est définie comme la réponse moyenne du détecteur de lignes le long de ce segment ; les potentiels $U_s(y_s \mid x_s)$ pour les labels 0 et 1 sont alors obtenus par une étape d'apprentissage le long de quelques segments appartenant à une vraie route et de quelques segments de "fausse alarme" ; la figure 2.7 montre les fréquences des observations y (approximant les $P(y_s \mid x_s)$) et les potentiels linéaires par morceaux qui en sont déduits (fig. 2.8) ;
– Énergie a priori : elle permet justement d'intégrer toutes les connaissances a priori qu'on veut prendre en compte pour la détection du réseau, par exemple du type :
  - (i) les routes sont longues et, dans l'absolu, elles ne s'arrêtent pas ;
  - (ii) elles ont une courbure relativement faible ;



a. Fréquences des observations sur des segments-routes détectés manuellement (approximant $P(Y_s \ / \ X_s = 1)$).

b. Fréquence des observations sur des segments-non routes (approximant $P(Y_s \ / \ X_s = 0)$).

FIG. 2.7 – Fréquences conditionnelles des observations sur une zone test.

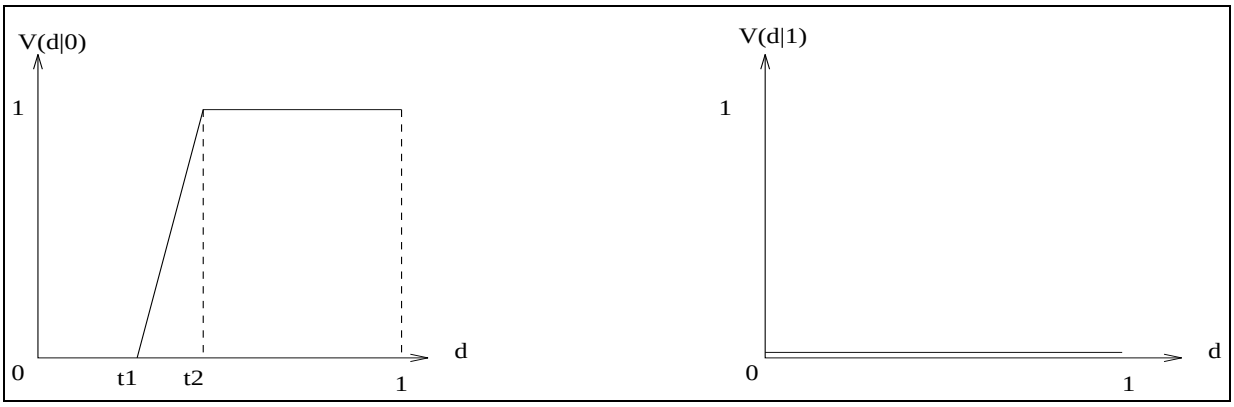


FIG. 2.8 – Potentiels linéaires par morceaux utilisés

- (iii) un segment de route est plus souvent connecté en une extrémité à un unique segment de route qu'à plusieurs.

Ces a priori peuvent s'exprimer en définissant les potentiels des cliques d'ordre maximal (rappelons qu'une clique est un ensemble de sites tous voisins les uns des autres, donc dans notre cas, un ensemble de segments qui se rejoignent en un même point) ; quatre paramètres suffisent pour les contraintes mentionnées précédemment : un paramètre contrôlant les extrémités (qui vise à défavoriser la configuration d'une clique où un seul segment a le label "route") ; deux paramètres contrôlant la longueur des routes (qui visent à favoriser des configurations où il y deux segments routes qui se joignent en une extrémité et qui sont "alignés") ; un paramètre contrôlant les carrefours (qui vise à défavoriser les configurations où une multitude de segments sont "route").

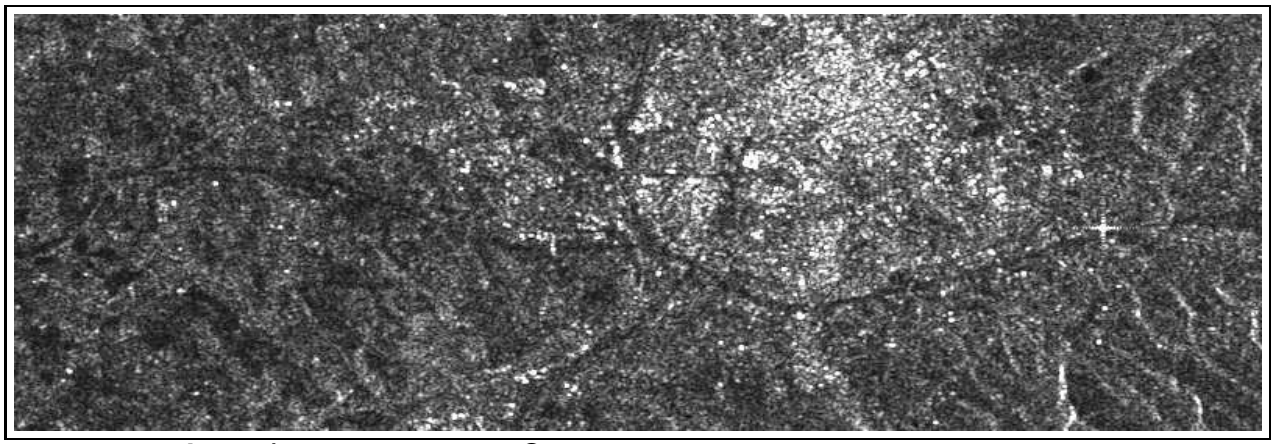
L'apprentissage de ces paramètres se fait difficilement par les méthodes du chapitre 3. En revanche l'étude de configurations extrêmes (chaîne de segments tous à "route", etc.), plus connue sous le nom de "Boîtes qualitatives d"Azencott" [2], permet de fixer des intervalles de valeurs pour ces paramètres.

Un exemple de résultat obtenu par recuit simulé est montré sur la figure 2.10.

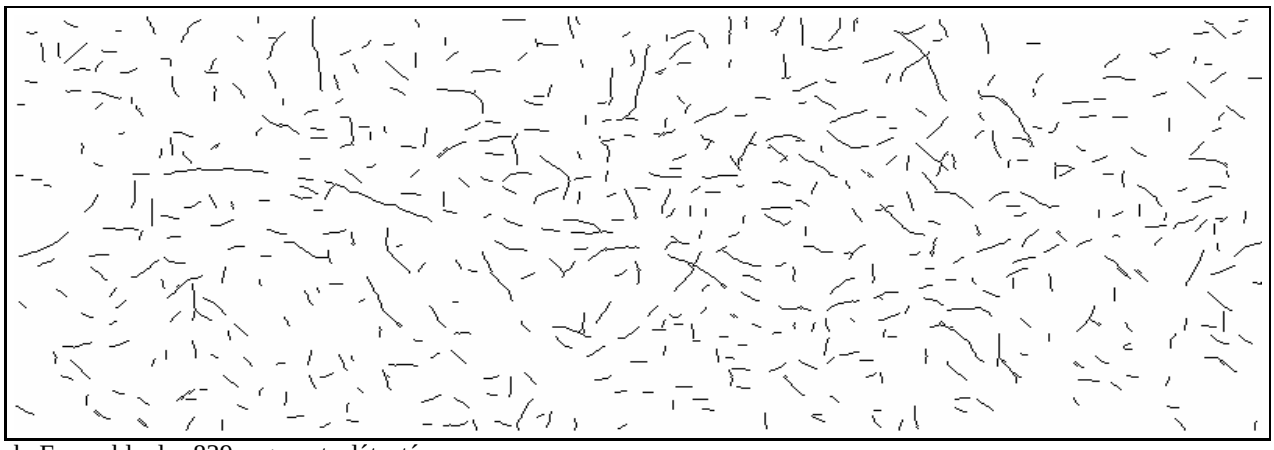
2.6.3 Conclusion

Ce chapitre ne se veut pas une présentation exhaustive des applications possibles des champs markoviens, mais présente quelques unes de leurs potentialités dans des domaines et à des niveaux de traitement d'image variés. Leur grande souplesse permet en effet d'introduire toutes sortes d'informations, que ce soit pour le terme d'attache aux données, que ce soit pour les a priori possibles, ou même encore sur la forme du graphe à utiliser.

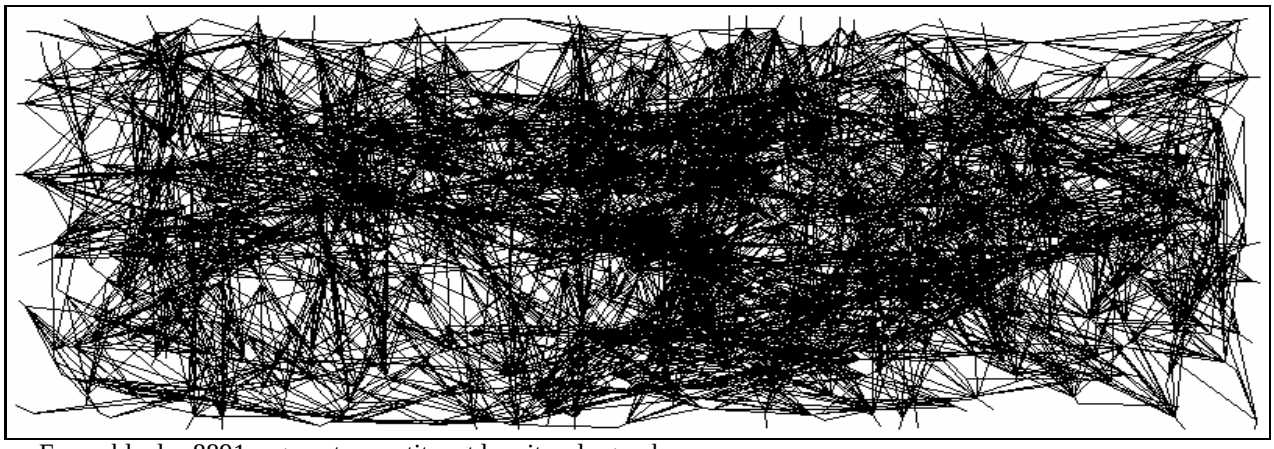
A côté de la multitude d'applications qui peuvent trouver une solution dans un cadre markovien, s'ouvrent des champs de recherche plus théoriques sur l'estimation des paramètres et l'accélération des techniques de recherche de solution.



a. Image originale centrée sur Aix en Provence ©ESA .



b. Ensemble des 839 segments détectés.



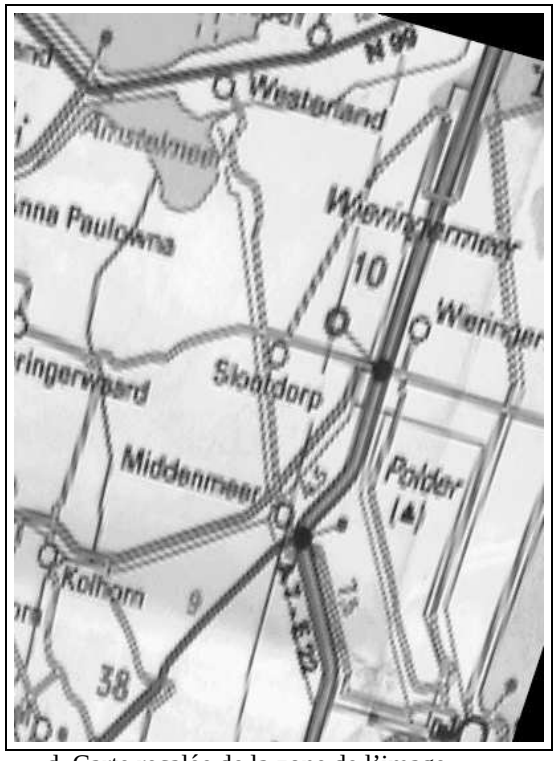
c. Ensemble des 8891 segments constituant les sites du graphe.

FIG. 2.9 – Les étapes de la construction du graphe

a. Image originale (Wieringermeer) ©ESA.

b. Image des segments détectés.

c. Résultat de la détection markovienne.



d. Carte recalée de la zone de l'image.

FIG. 2.10 – Détection des routes sur le Wieringermeer

Bibliographie

[1] L. Aurdal. *Analyse d'images IRM 3D multi-échos pour la détection et la quantification de pathologies cérébrales*. PhD thesis, Ecole Nationale Supérieure des Télécommunications, 1997.

[2] R. Azencott. Markov field approach : parameter estimation by qualitative boxes. *Cours : Les Houches*, 1992.

[3] L. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occuring in the statistical ananlysis of probabilistic functions of markov chains. *The Annals of Mathematical Statistics*, 41 :164–171, 1970.

[4] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *J. Royal Statist. Soc. (series B)*, 36 :192–326, 1974.

[5] J. Besag. On the statistical analysis of dirty pictures. *J. R. Statist. Soc. B*, 48(3) :259–302, 1986.

[6] A. Blake and A. Zisserman. *Visual Reconstruction*. MIT Press, 1987.

[7] I. Bloch, L. Aurdal, D. Bijno, and J. Muller. Estimation of class membership functions for grey-level based image fusion. *ICIP'97 (Santa Barbara)*, 1997.

[8] B. Chalmond. An iterative Gibbsian technique for reconstruction of m-ary images. *Pattern Recognition*, 22(6) :747–761, 1989.

[9] B. Chalmond. *Eléments de modélisation pour l'analyse d'images*, volume 33. Springer - Mathématiques & Applications, November 1999.

[10] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud. Deterministic edge-preserving regularization in computed imaging. *IEEE Transactions on Image Processing*, 5(12), December 1996.

[11] Pierre Charbonnier. *Reconstruction d'image : régularisation avec prise en compte des discontinuités*. PhD thesis, Université de Nice Sophia Antipolis, 1994.

[12] H. Derin and H. Elliott. Modeling and segmentation of noisy and textured images using gibbs random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(1), January 1987.

[13] H. Derin, H. Elliott, and J. Kuang. A new approch to parameter estimation for gibbs random field. In *Int. Conf. on ASSP*, March 1985.

[14] X. Descombes. *Champs Markoviens en analyse d'images*. PhD thesis, Ecole Nationale Supérieure des Télécommunications (ENST 93 E 026), 1993.

[15] X. Descombes, J. F. Mangin, E. Pechersky, and M. Sigelle. Fine structures preserving Markov model for image processing. *The 9th Scandinavian Conference on Image Analysis (Uppsala, Sweden)*, 2 :349–356, June 1995.

[16] X. Descombes, R. Morris, J. Zerubia, and M. Berthod. Estimation of markov random field prior parameters using markov chain monte carlo likelihood. *IEEE Transactions on Image Processing*, 8(7) :954–963, 1999.

[17] E. Dougherty (Ed.). *Image Processing and Mathematical Morphology*. Marcel Dekker, 1992.

[18] D. Geiger and F. Girosi. Parallel and deterministic algorithms from MRF's : Surface reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(5) :401–412, 1991.

[19] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distribution, and the Bayesian Restauration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6) :721–741, November 1984.

[20] S.. Geman and D. E. McClure. Bayesian image analysis : an application to single photon emission tomography. *Proc. Statist. Comput. sect. (Amer. Statist. Assoc. Washington DC)*, pages 12–18, 1985.

[21] T. Géraud, J. F. Mangin, I. Bloch, and H.Maître. Segmenting internal structures in 3D MR images of the brain by Markovian relaxation on a watershed based adjacency graph. *IEEE ICIP (Austin)*, III :548–552, 1995.

[22] J. Goutsias and H. J. A. M. Heijmans. Nonlinear Multiresolution Signal Decomposition Schemes : Part I : Morphological Pyramids. *IEEE Transactions on Image Processing*, 9(11) :1862–1876, 2000.

[23] C. Graffigne. *Experiments in Texture Analysis and Segmentation*. PhD thesis, Division of Applied Mathematics - Brown University, 1987.

[24] X. Guyon. *Champs aléatoires sur un réseau - modélisations, statistique et applications*. Collection Techniques Stochastiques, Masson, 1992.

[25] H. Hadwiger. *Vorlesungen über Inhalt, Oberfläche und Isoperimetrie*. Springer-Verlag, Berlin, 1957.

[26] H. J. A. M. Heijmans and J. Goutsias. Nonlinear Multiresolution Signal Decomposition Schemes : Part II : Morphological Wavelets. *IEEE Transactions on Image Processing*, 9(11) :1897–1913, 2000.

[27] H. J. A. M. Heijmans and C. Ronse. The Algebraic Basis of Mathematical Morphology – Part I : Dilations and Erosions. *Computer Vision, Graphics and Image Processing*, 50 :245–295, 1990.

[28] H.J.A.M. Heijmans. Theoretical Aspects of Gray-Level Morphology. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13 :568–582, 1991.

[29] E. Ising. Beitrag zur theorie des ferromagnetisms. *Zeitschrift fur Physik*, 31 :253–258, 1925.

[30] M. Khoumri. Estimation d'hyperparamètres pour la déconvolution d'images satellitaires - rapport de stage dea. Technical report, INRIA Sophia, September 1997.

[31] S. Kirkpatrick, C. D. Gellatt, and M. P. Vecchi. Optimization by simulated annealing. *IBM Thomas J. Watson research Center, Yorktown Heights, NY*, 1982.

[32] S. Lakshmanan and H. Derin. Simultaneous parameter estimation and segmentation of gibbs random fields using simulated annealing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(8) :799–813, 1989.

[33] L. Landau and E. Lifschitz. *Cours de Physique Tome 5 - Physique Statistique*. Editions Mir, 1961.

[34] A. Meijster and M. Wilkinson. A comparison of algorithms for connected set openings and closings. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4), 2002.

[35] M. Métivier and P. Priouret. Théorèmes de convergence presque sûre pour une classe d'algorithmes stochastiques à pas décroissant. *Probability Theory and Related Fields*, 74 :403–428, 1987.

[36] N. Metropolis, A. W. Rosenbluth, N. M. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *J. Chemical Physics*, 21 :1087–1091, 1953.

[37] L. Najman and M. Schmitt. Geodesic saliency of watershed contours and hierarchical segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(12) :1163–1173, 1996.

[38] H.T. Nguyen, M. Worring, and R. van den Boomgaard. Watersnakes : energy-driven watershed segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(3) :330–342, 2003.

[39] L. Onsager. *Physical Review*, 65 :117, 1944.

[40] N. Peyrard. *Approximations de type champ moyen des modèles de champ de Markov pour la segmentation de données spatiales*. PhD thesis, Université J. Fourier, Grenoble, 2001.

[41] W. Pieczynski. Hidden markov fields and iterative conditional estimation. *Traitement du Signal*, 11(2) :141–153, 1994.

[42] R.A. Redner and H.F. Walker. Mixture densities, maximum likelihood and the em algorithm. *SIAM Review*, 26 :195–239, 1984.

[43] C. Ronse and H. J. A. M. Heijmans. The Algebraic Basis of Mathematical Morphology – Part II : Openings and Closings. *Computer Vision, Graphics and Image Processing*, 54 :74–97, 1991.

[44] P. Salembier. Morphological multiscale segmentation for image coding. *Signal Processing*, 38 :359–386, 1994.

[45] P. Salembier, A. Oliveras, and L. Garrido. Anti-extensive connected operators for image and sequence processing. *IEEE Transasctions on Image Processing*, 7 :555–570, 1998.

[46] L. A. Santalo. *Integral Geometry and Geometric Probability*. Addison Wesley, 1976.

[47] M. Schmitt and J. Mattioli. *Morphologie mathématique*. Masson, Paris, 1994.

[48] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, New-York, 1982.

[49] J. Serra (Ed.). *Image Analysis and Mathematical Morphology, Part II : Theoretical Advances*. Academic Press, London, 1988.

[50] M. Sigelle. *Champs de Markov en traitement d'images et modèles de la physiques statistique : applications en relaxation d'images de classification*. PhD thesis, Ecole Nationale Supérieure des Télécommunications, 1993.

[51] M. Sigelle. Simultaneous image restoration and hyperparameter estimation for incomplete data by a cumulant analysis. Technical report, INRIA Sophia Antipolis, September 1997.

[52] P. Soille. *Morphological Image Analysis*. Springer-Verlag, Berlin, 1999.

[53] F. Tupin. *Reconnaissance des formes et analyse de scénes en imagerie radar à ouverture synthétique*. PhD thesis, Ecole Nationale Supérieure des Télécommunications, September 1997.

[54] F. Tupin, E. Trouvé, X. Descombes, J.-M. Nicolas, and H. Maître. Improving IFSAR phase unwrapping by early detection of non-interferometric features. *European Symposium on Satellite Remote Sensing III (Taormina, Italy)*, September 1996.

[55] Corinne Vachier and Fernand Meyer. The viscous watershed transform. *Journal of Mathematical Imaging and Vision*, 22(2 - 3) :251–267, 2005.

[56] L. Vincent. Graphs and Mathematical Morphology. *Signal Processing*, 16 :365–388, 1989.

[57] L. Vincent. Morphological Algorithms. In E. Dougherty, editor, *Mathematical Morphology in Image Processing*, pages 255–288. Marcel Dekker, 1992.

[58] L. Vincent and P. Soille. Watersheds in Digital Spaces : an Efficient Algorithm based on Immersion Simulations. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13(6) :583–598, 1991.

[59] M. Wilkinson and J. Roerdink. Fast morphological attribute operations using tarjan 's union-find algorithm. In *Mathematical Morphology and its Applications to Image and Signal Processing, Kluwer*, pages 311–320, 2000.

[60] F. Y. Wu. The potts model. *Reviews of Modern Physics*, 54(1) :235–267, January 1982.

[61] L. Younes. Estimation and annealing for gibbsian fields. *A. Inst Henri Poincaré*, 24(2) :269–294, 1988.

[62] L. Younes. Parametric inference for imperfectly observed gibbsian fields. *Probability Theory and Related Fields*, 82 :625–645, 1989.

[63] L. Younes. Parameter estimation for imperfectly observed gibbs fields and some comments on chalmond's em gibbsian algorithm. In *stochastic models, statistical methods and algorithms in image analysis*. P. Barone and A. Frigessi, Lecture Notes in Statistics, Springer, 1991.

[64] J. Zerubia and L. Blanc-Féraud. Hyperparameter estimation of a variational model using a stochastic gradient method. In *SPIE Bayesian Inference for Inverse Problems Proceedings*, volume 3459, July 1998.

[65] J. Zerubia and R. Chellapa. Mean field annealing using compound Gauss-Markov random fiel ds for edge detection and image estimation. *IEEE Transactions on neural networks*, 4(4) :703–709, 1993.

Restauration, MATIM (SI 343)

Saïd Ladjal

Introduction

Dans ce court texte nous rappelons le principe des méthodes variationnelles dans le cadre de la restauration des images. Ensuite, nous étudions dans le détail une méthode variationnelle de débruitage utilisant la variation totale comme terme de régularisation.

1 Didactique de la régularisation

Les images numériques sont acquises par des appareils physiques. L'appareil photographique mesure l'énergie lumineuse qui atteint le capteur à partir de la scène photographiée. D'abord, une image est stockée sous la forme d'un nombre fini de valeurs. Cela implique que la représentation que l'on a de l'image est discrète, en particulier nous ne pouvons pas espérer obtenir une valeur d'énergie lumineuse qui varie trop finement en fonction de la position. À l'opposé, on conçoit que la luminosité réelle de la scène peut varier très rapidement en fonction de la position (penser à un appareil qui photographie un paysage arboré, on imagine bien que l'on ne pourra pas distinguer dans l'image les détails trop fins tels que les pores des feuilles des arbres. À moins d'augmenter de beaucoup la résolution de l'appareil...). Ainsi l'acquisition des images introduit une dégradation inévitable. Le but de la restauration est d'essayer d'atténuer l'effet de la dégradation.

1.1 Modèle d'acquisition

Une image parfaite, f_0 est observée par un appareil photographique qui acquiert l'image discrète g. La relation qui lie g à f_0 est :

$$g = \Pi(f_0 * h + b).$$

Où h représente tous les flous subis par l'image (flou optique et flou d'intégration de l'énergie de chaque pixel du capteur) et b est le bruit qui s'ajoute à l'image. Enfin, Π est l'opération d'échantillonnage sur la grille du capteur.

f_0 est une fonction de $\mathbb{R}^2$ dans $\mathbb{R}$ et g est une fonction de $\mathbb{Z}^2$ dans $\mathbb{R}$. Le but de restauration étant de retrouver f à partir de g, on pourrait penser que la tâche est impossible car l'espace des fonctions définie sur $\mathbb{R}^2$ est beaucoup plus gros que celui des fonctions définies sur $\mathbb{Z}^2$. Nous éclaircissons ce problème au paragraphe suivant.

1.1.1 Cas de la bande limitée

Ici, nous voulons simplifier le modèle d'acquisition afin de ne plus faire apparaître d'images définies sur des domaines différents. Pour cela on va supposer que l'image parfaite, f_0 est à bande limitée (**i.e.** sa transformée de Fourier est à support borné). On sait, par le théorème de Shannon, que cela implique qu'il y a équivalence entre la donnée de f_0 ou la donnée de ses échantillons sur une grille régulières dont les points sont espacés de $1/L$ où L est la largeur de la bande spectrale qui supporte toute la transformée de Fourier de f_0. Si nous appelons f_d la fonction de $\mathbb{Z}^2$ dans $\mathbb{R}$ qui représente l'échantillonnage de f_0, il est clair que f_d dépend linéairement de f. On ne rentre pas dans le détail de cette dépendance (voir les cours de base de traitement du signal).

S'il y a une équivalence entre la donnée de f_0 et la donnée de ses échantillons, alors g (l'image effectivement acquise) dépend linéairement de f_d. Par ailleurs si l'on translate f_d d'une distance entière, alors il est évident que cela correspond à une translation de f_0 de la même distance entière et, comme g dépend de f_0 par une convolution, g est encore translatée de la même distance. Il y a alors une relation linéaire et invariante par translation qui donne g à partir de f_d. g est donc le résultat de la convolution de f_d par un certain filtre que nous noterons h_d (attention, h_d n'est pas forcément le résultat de l'échantillonnage du noyau h). Tout ceci se résume dans le modèle suivant :

$$g = h_d * f_d + b.$$

Dans la pratique on ne manipule que des images de taille finie. Si l'on adapte le dernier modèle au cas de la dimension finie et que l'on représente les images par des vecteurs de $\mathbb{R}^N$ (où N est le nombre de pixels de l'image) alors on a

$$G = A.F_0 + B$$

où G, F et B sont des vecteurs de $\mathbb{R}^N$ et A une matrice de taille $N \times N$. De plus, A représente une convolution discrète. *a priori* la donnée de A demande de connaître N^2 coefficients (ce qui fait 10^{12} coefficients si on manipule des images de un million de pixels), mais comme A est la matrice d'une convolution elle a pour vecteurs propre les ondes pures (vecteurs de Fourier). A est donc diagonalisable dans la base de Fourier et sa description ne demande que N coefficients qui sont ses valeurs propres dans la base de Fourier. Ces coefficients sont la transformée de Fourier du noyau h_d.

1.2 Inversion directe

La solution la plus simple au problème de restauration est d'inverser la matrice A et d'appliquer le résultat à G cela donne :

$$F = A^{-1}G = F_0 + A^{-1}B$$

où F est l'image estimée. La matrice A^{-1} est encore diagonalisable dans la base de Fourier et ses valeurs propres sont les inverses des valeurs propres de A. Or, les noyaux de convolution ont souvent une décomposition dans la base de Fourier (les valeurs propres de A) qui décroît vite en fonction de la fréquence. Par contre le bruit B (que l'on peut supposer blanc c'est-à-dire décorrélé spatialement) distribue son énergie de manière uniforme en fonction de la fréquence. Le terme $A^{-1}.B$ a toutes les chances d'être aberrant du fait du

rehaussement de l'énergie du bruit dans les hautes fréquences (voir les transparents et TP pour les expériences).

Pour trouver une estimation F qui soit plus acceptable il faut ajouter une hypothèse de régularité sur l'image F afin de refuser les termes aberrants tels que $A^{-1}.B$. Cela justifie l'introduction de la régularisation et son implémentation pratique par le biais des méthodes variationnelles.

1.3 Régularisation

L'inversion correspond à la minimisation de $\|g - h * f\|^2$, c'est-à-dire que l'on cherche une image, f, qui une fois qu'elle a subi les dégradations de l'appareil photographique donne l'image observée g. Pour introduire une connaissance sur les images naturelles, l'approche variationnelle consiste à demander à ce que l'image restaurée soit régulière. La régularité que l'on demande à l'image peut être déduite de modèles probabilistes (voir cours sur Markov) ou tout autre régularité (voir ce qui suit sur la variation totale) qui se constate dans les images. La manière d'introduire cette contrainte de régularité est de minimiser une fonctionnelle qui mélange la fidélité aux observations (attache aux données, $\|g - h * f\|^2$) et une énergie de régularité, par exemple l'intégrale du gradient au carré. Soit le problème

$$f \text{ minimise } E(f) = \|Af - g\|^2 + \lambda \iint \|\nabla f\|^2. \ (A \text{ représente la convolution par } h)$$

1.4 Méthodes de de minimisation

Nous donnons ici une méthode de minimisation *ad hoc* adaptée au cas où l'énergie à minimiser est quadratique et le modèle d'acquisition ne fait intervenir que la convolution. Cette méthode a l'avantage d'être très rapide car elle passe par la transformation de Fourier. Par ailleurs elle exprime le résultat dans le domaine de Fourier ce qui permet de faire le lien avec le filtrage de Wiener qui se déduit directement d'hypothèses statistiques sur les signaux traités.

Soit donc à minimiser la fonctionnelle :

$$E(f) = \|Af - g\|^2 + \lambda \iint \|\nabla f\|^2$$

où A est une matrice représentant une convolution, f l'image à trouver et g l'image observée. f et g sont des vecteurs de $\mathbb{R}^N$ où N est le nombre de pixels de l'image. A est une matrice carrée de taille $N \times N$. On peut réécrire cette énergie sous la forme :

$$E(f) = \|Af - g\|^2 + \lambda(\|D_x f\|^2 + \|D_y f\|^2)$$

où D_x est la matrice qui représente la dérivation en x. Elle transforme une image $I(x, y)$ en l'image $I(x + 1, y) - I(x, y)$. Aux bords près, on constate que D_x est la matrice de la convolution par le noyau $h_x = [1\ -1]$. De même D_y représente le dérivateur en y et elle effectue la convolution par le noyaux $h_y = [1\ -1]^T$.

Or, l'égalité de Parseval nous dit que la norme au carré[1] d'un vecteur de $\mathbb{R}^N$ est égale à la norme au carré[2] de sa transformée de Fourier. On a donc

$$E(f) = \sum_w \left[|\hat{h}(w)\hat{f}(w) - \hat{g}(w)|^2 + \lambda |\hat{f}(w)|^2 \left(|\widehat{h_x}(w)|^2 + |\widehat{h_y}(w)|^2 \right) \right]$$

où le $\hat{k}$ signifie "transformée de Fourier de k" et w parcourt l'ensemble des fréquences possibles (qui est un ensemble bi-dimensionnel pour les images). On constate que, pour chaque w, la valeur $\hat{f}(w)$ n'apparaît que dans un seul terme de la somme définissant E. Pour minimiser E nous arrivons donc à la conclusion qu'il faut minimiser chacun des termes de la dernière somme par rapport à une variable $\hat{f}(w)$ qui n'apparaît dans une aucun autre terme. Si z est une variable complexe et que l'on veut minimiser l'expression

$$|z\alpha - \beta|^2 + |z|^2\gamma$$

où α, β sont des complexes et γ un réel positif (trouver leurs expressions pour faire coïncider cette expression avec un terme de la somme définissant E). Alors z doit valoir

$$z = \frac{\overline{\alpha}\beta}{|\alpha|^2 + \gamma}$$

On obtient donc comme minimiseur de E une fonction f dont la transformée de Fourier vérifie

$$\hat{f}(w) = \frac{\overline{\hat{h}(w)}}{|\hat{h}(w)|^2 + \lambda \left(|\widehat{h_x}(w)|^2 + |\widehat{h_y}(w)|^2 \right)} \hat{g}(w).$$

Ainsi pour minimise E il suffit de calculer la transformée de Fourier de g et de la multiplier point par point par le coefficient décrit ci-dessus. Enfin, on calcule la transformée de Fourier inverse pour trouver le minimiseur.

1.5 Débruitage

On peut appliquer les même raisonnements que ci-dessus pour trouver une solution au problème du débruitage. Dans un problème de débruitage la dégradation est réduite à un ajout de bruit. Ceci revient à prendre pour noyau h le Dirac, dont la transformée de Fourier est constante égale à 1. En reprenant l'équation ci-dessus dans ce cas-là on a

$$\hat{f}(w) = \frac{1}{1 + \lambda \left(|\widehat{h_x}(w)|^2 + |\widehat{h_y}(w)|^2 \right)} \hat{g}(w)$$

Dans la suite nous introduisons un autre type d'énergie de régularité. Comme vous le verrez en TP, cette nouvelle régularité donne de meilleurs résultats de débruitage, mais la méthode de minimisation est bien plus compliquée.

[1] D'où l'intérêt d'avoir choisi des normes quadratiques.

[2] à une constante près, mais comme nous voulons minimiser E, cela ne change rien au problème de minimiser E ou une constante fois E.

2 Débruitage par variation totale

Au début des années 1990 les auteurs Rudin, Osher et Fatemi ont proposé une méthode de débruitage basée sur la variation totale. Ayant observé une image g suivant le modèle d'observation

$$g = f_0 + b$$

où b est le bruit (supposé blanc de puissance totale σ^2) et f_0 l'image parfaite. Les auteurs proposent comme version débruitée de g l'image f qui vérifie

$$\begin{cases} f \text{ minimise } \iint \|\nabla f\| \\ \text{sous la contrainte } \|g - f\|^2 = \sigma^2 \end{cases}$$

Ceci signifie que l'on suppose l'espace des images mieux représenté par la régularité que l'on appelle "variation totale". On note TV la variation total

$$TV(f) = \iint |\nabla f|.$$

Pour résoudre ce problème sous contrainte il suffit de choisir un paramètre de régularisation λ et de minimiser sans contrainte la fonctionnelle

$$E(f) = \|g - f\|^2 + \lambda TV(f)$$

Nous allons introduire dans la section suivantes des notions qui généralisent la notion de gradient et qui seront des outils pour une méthode de minimisation de fonctionnelles faisant intervenir la variation totale.

2.1 Sous-différentielles, transformée de Fenchel et fonctionnelles d'ordre 1

Dans cette section on se donne une fonctionnelle convexe que l'on note J définie de $\mathbb{R}^N$ à valeurs dans $\mathbb{R} \cup \{+\infty\}$. Le produit scalaire usuel est noté $< x|y >$ où x et y sont des vecteurs de $\mathbb{R}^N$.

Dire que J est convexe signifie que

$$\forall x, y \in \mathbb{R}^N, \theta \in [0,1], J(\theta x + (1-\theta)y) \leqslant \theta J(x) + (1-\theta)J(y)$$

avec la convention que la somme de $+\infty$ avec n'importe quel réel (ou $+\infty$) vaut $+\infty$. L'ensemble de définition de J est l'intérieur de l'ensemble des points où elle prend des valeurs finies, il est convexe. En dimension finie, ce qui est notre cas, J est continue sur son ensemble de définition.

Pour illustrer les définitions on prendra la fonctionnelle exemple : $J_0(x) = |x|$ définie sur $\mathbb{R}$ ($N = 1$).

Définition 1. Sous différentielle

Aux points x intérieurs à l'ensemble de définition de J la sous-différentielle est un ensemble de vecteurs de $\mathbb{R}^N$ que l'on not $\partial J(x)$ qui vérifient :

$$v \in \partial J(x) \Leftrightarrow \forall y \in \mathbb{R}^N J(y) \geqslant J(x) + < v|y - x >$$

Cet ensemble est non vide et il est réduit à un seul élément lorsque J est différentiable en x, de plus il est convexe et fermé. On peut le regarder comme l'ensemble des vecteurs qui définissent (par produit scalaire) un hyperplan qui passe sous le graphe de J au point x. On comprends alors que lorsqu'il est réduit à un point, il n'y a qu'un seul hyperplan qui passe sous le graphe ce qui caractérise les points où une fonction convexe est différentiable.

Exemple :

$$\partial J_0(x) = \begin{cases} \{1\} & \text{si} \quad x > 0 \\ [-1,1] & \text{si} \quad x = 0 \\ \{-1\} & \text{si} \quad x < 0 \end{cases}$$

Proposition 1. critère de minimisation

Si x_0 est un point de l'ensemble de définition de J alors : J atteint sont minimum en x_0 si et seulement si $0 \in \partial J(x_0)$.

démonstration :
J atteint son minimum en x_0 si et seulement si
$\forall y \in \mathbb{R}^N, J(y) \geqslant J(x_0)$ si et seulement si
$\forall y \in \mathbb{R}^N, J(y) \geqslant J(x_0) + < 0 | y - x >$ si et seulement si
$0 \in \partial J(x_0)$.□

Définition 2. Transformée de Fenchel

On appelle transformée de Fenchel de J que l'on note J^* la fonctionnelle définie sur $\mathbb{R}^N$ par

$$J^*(v) = \sup_{x \in \mathbb{R}^N} < v | x > - J(x).$$

Il faut la voir comme une fonctionnelle définie sur l'ensemble des hyperplans (dualité introduite par le produit scalaire). Pour un vecteur v définissant un hyperplan, $J^*(v)$ vaut le maximum de la différence entre le graphe de J et l'hyperplan défini par v.

Si le sup est atteint en un point x alors $v \in \partial J(x)$. (le démontrer)

Proposition 2.

J^* est une fonctionnelle convexe et

$$J^{**}(x) = J(x)$$

(aux points x de l'ensemble de définition de J). Autrement dit appliquer deux fois la transformation de Fenchel à une fonctionnelle revient à ne pas la modifier.

Démonstration :
Convexité : Soit u, v deux vecteurs et θ un réel de l'intervalle $[0, 1]$.

$$\forall x, < \theta u + (1-\theta) v | x > - J(x) = \theta(< u | x > - J(x)) + (1-\theta)(< v | x > - J(x)) \leqslant \theta J^*(u) + (1-\theta) J^*(v).$$

La dernière inégalité vient de la définition de J^* comme un sup. Donc

$$J^*(\theta u + (1-\theta)v) \leqslant \theta J^*(u) + (1-\theta) J^*(v)$$

Égalité entre J et J^{**} :

$$J^{**}(x) = \sup_{v\in\mathbb{R}^N} < x|v > -J^*(v).$$

Or, pour tout vecteur v

$$J^*(v) \geqslant < v|x > -J(x) \Rightarrow < x|v > -J^*(v) \leqslant < x|v > -(< v|x > -J(x)) = J(x)$$

D'où

$$J^{**}(x) \leqslant J(x)$$

Soit $v_0 \in \partial J(x)$ alors $J^*(v_0) =< v_0|x > -J(x)$ (par définition de la sous-différentielle (faire un dessin...))

On a

$$J^{**}(x) \geqslant < x|v_0 > -J^*(v_0) = J(x)$$

Ce qui termine la preuve de $J^{**} = J\square$.

Définition 3. Fonctionnelle d'ordre 1

J est dite d'ordre 1 si

$$\forall\lambda \in \mathbb{R}, x \in \mathbb{R}^N, J(\lambda x) = |\lambda|J(x)$$

Proposition 3.

Si J est d'ordre 1 alors il existe un fermé convexe $K \subset \mathbb{R}^N$ tel que

$$J^*(v) = \begin{cases} 0 & si \quad v \in K \\ +\infty & si \quad v \notin K \end{cases}$$

De plus $K = \partial J(0)$.

Démonstration :

D'abord comme $J(0) = J(0x) = 0J(x) = 0$ on a toujours $J^*(v) \geqslant 0$ (prendre $x = 0$ dans le sup définissant J^*). Si $J^*(v) > 0$ alors il existe x tel que

$$< v|x > -J(x) = A > 0$$

Mais alors la suite nx vérifie

$$< v|nx > -J(nx) = n(< v|x > -J(x)) = nA$$

et nA tend vers $+\infty$ avec n. Donc $J^*(v) = +\infty$.

On a donc montré que J^* ne peut prendre que les deux valeurs 0 ou $+\infty$. Si on appelle $L = \partial J(0)$, on a

$v \in L$ si et seulement si

$\forall y, J(y) \geqslant < v|y > -J(0)$ si et seulement si

$\forall y, 0 \geqslant < v|y > -J(y)$ si et seulement si

$J^*(v) \leqslant 0$ (on a déjà que $J^*(v) \geqslant 0$). Il y a donc équivalence entre $J^*(v) = 0$ (i.e. $v \in K$) et $v \in \partial J(0)$ qui est un ensemble convexe fermé[3].$\square$

Exemple :

Montrer directement que

$$J_0^*(v) = \begin{cases} 0 & \text{si} \quad v \in [-1, 1] \\ +\infty & \text{si} \quad |v| > 1 \end{cases}$$

[3]montrer en exercice que tout sous-différentielle est un ensemble fermé

Proposition 4. *Pour tous vecteurs x et v on a*

$$v \in \partial J(x) \Leftrightarrow x \in \partial J^*(v)$$

On a
$v \in \partial J(x)$ si et seulement si (1)
$\forall y J(y) \geqslant < v|y - x > + J(x)$ ssi
$\forall y, < v|y > - J(y) \leqslant < v|x > - J(x)$ ssi
$J^*(v) = < v|x > - J(x)$ ssi
$J^*(v) + J(x) = < v|x >$ ssi (2)
$J^{**}(x) + J^*(v) = < v|x >$ (on a remplacé J par J^{**} qui lui est égal) ssi
$x \in \partial J^*(v)$ (on retourne le raisonnement de la ligne (2) vers la ligne (1) en changeant J par J^*) $\square$.

2.2 Retour au problème du débruitage

Soit donc la fonctionnelle

$$E(f) = \frac{1}{2}\|f - g\|^2 + \lambda TV(f)$$

La sous-différentielle de E est la somme[4] du gradient de $\frac{1}{2}\|f - g\|^2$ et de la sous-différentielle de TV[5] (démontrer que la sous-différentielle d'une somme de fonctionnelle est la somme des sous-différentielles). Le gradient de $\frac{1}{2}\|f - g\|^2$ est $f - g$ (le démontrer). E est définie sur tout l'espace des images. D'après le critère de minimisation on a

$$\begin{gathered} f \text{ minimise } E \Leftrightarrow \\ 0 \in \partial E(f) = f - g + \lambda \partial TV(f) \Leftrightarrow \\ \frac{g - f}{\lambda} \in \partial TV(f) \Leftrightarrow \\ f \in \partial TV^*(\frac{g - f}{\lambda}) \text{ (proposition 4)} \end{gathered} \tag{1}$$

Si maintenant on cherche w qui minimise

$$F(w) = TV^*(\frac{w}{\lambda}) + \frac{1}{2}\|w - g\|^2$$

alors on a

$$\begin{gathered} w \text{ minimise } F \Leftrightarrow \\ 0 \in \partial TV^*(\frac{w}{\lambda}) + w - g \Leftrightarrow \\ g - w \in \partial TV^*(\frac{w}{\lambda}) \end{gathered} \tag{2}$$

[4]la somme de deux ensembles est définie comme la somme des points qui s'écrivent comme somme d'un élément de chaque ensemble

[5]le facteur $\frac{1}{2}$ a été ajouté pour la clarté des calculs, il revient à prendre un λ différent pour se rament au cas sans facteur

En identifiant les équations (1) et (2) on a

$$f \text{ minimise } E \Leftrightarrow$$

$$w = g - f \text{minimise } F$$

Or TV est une fonctionnelle d'ordre 1 et alors TV^* ne prend que les valeur 0 et $+\infty$. Donc si w minimise F il faut au moins que $\frac{w}{\lambda} \in K = \partial TV(0) \Leftrightarrow w \in \lambda K$ (proposition 3). Réciproquement parmi tous les $w \in \lambda K$ il faut choisir le minimiseur de $\|g - w\|^2$ pour minimiser F. Il faut donc choisir w comme le projeté orthogonal de g sur l'ensemble λK. Et enfin prendre $f = g - w$ pour obtenir un minimiseur de E.

On a donc ramené la minimisation de notre fonctionnelle à la projection sur un convexe. Dans la section suivante on donne une définition explicite de K ainsi que l'algorithme de projection sur K qui a été développé par Antonin Chambolle.

2.3 Algorithme de minimisation

On commence par définir dans le cas discret ce que sont le gradient et la divergence. Nous considérons une image de taille $M \times N$ définie sur l'ensemble $[\![0, M-1]\!] \times [\![0, N-1]\!]$.

Définition 4. gradient discret

Le gradient d'une image discrète est une fonction définie sur le même ensemble $[\![0, M-1]\!] \times [\![0, N-1]\!]$ *dont la valeur en chaque point est un vecteur à deux dimensions (le nuémro de la dimension est noté en exposant) :*

$$(\nabla u)_{i,j} = ((\nabla u)^1_{i,j}, (\nabla u)^2_{i,j})$$

avec

$$(\nabla u)^1_{i,j} = \begin{cases} u_{i+1,j} - u_{i,j} & si \quad i < M-1 \\ 0 & si \quad i = M-1 \end{cases}$$

$$(\nabla u)^2_{i,j} = \begin{cases} u_{i,j+1} - u_{i,j} & si \quad j < N-1 \\ 0 & si \quad j = N-1 \end{cases}$$

Définition 5. divergence discrète

Si p *est un champ de vecteurs défini sur* $[\![0, M-1]\!] \times [\![0, N-1]\!]$*, c'est-à-dire qu'il fait correspondre pour chaque point* (i, j) *un vecteur* $(p^1_{i,j}, p^2_{i,j})$*. La divergence de* p *notée div* p *est une fonction à valeurs réelles définie sur* $[\![0, M-1]\!] \times [\![0, N-1]\!]$ *par*

$$(div\ p)_{i,j} = \begin{cases} p^1_{i,j} - p^1_{i-1,j} & si \quad 0 < i < M-1 \\ p^1_{i,j} & si \quad i = 0 \\ -p^1 i, j & si \quad i = M-1 \end{cases} + \begin{cases} p^2_{i,j} - p^2_{i,j-1} & si \quad 0 < j < N-1 \\ p^2_{i,j} & si \quad j = 0 \\ -p^2 i, j & si \quad j = N-1 \end{cases}$$

Hors des bords la divergence d'un champ de vecteurs est la somme de la dérivée en x de la composante x avec la dérivée en y de la composante y. Le gradient transforme une image en un champ de vecteur et la divergence transforme un champ de vecteur en une image. La divergence du gradient d'une image est le laplacien de l'image.

L'ensemble $K = \partial J(0)$ que nous avons vu plus haut est caractérisé ainsi

$$K = \{\text{div } p \ : \ \forall i, j \|p_{i,j}\| \leqslant 1\}$$

Autrement dit K est l'ensemble des images qui peuvent s'écrire comme la divergence d'un champ de vecteurs dont la norme de chaque vecteur ne dépasse pas 1. Nous ne faisons pas la démonstration ici.

Il nous faut projeter g sur λK. Si on sait projeter g sur K alors pour projeter g sur λK il suffit de projeter g/λ sur K puis multiplier le résultat par λ.

Nous décrivons l'algorithme de projection sur K. τ est une constante qui doit être prise $< \frac{1}{8}$. L'algorithme est itératif.

Projection Pour projeter une image g sur K, faire :

Initialiser un champ de vecteurs p^0 à $p^0 = 0$.

Calculer le champ de vecteur p^{n+1} à partir de p^n par la formule

$$p_{i,j}^{n+1} = \frac{p^n + \tau \left(\nabla \left(\text{div } (p^n) - g\right)\right)_{i,j}}{1 + \tau \| \left(\nabla \left(\text{div } (p^n) - g\right)\right)_{i,j} \|}$$

Dans la pratique 20 ou 30 itérations suffisent.

À la fin, le projeté de g sur K est div p^∞ (la divergence du champ de vecteurs limite)

2.4 Exemples

Voir transparents et TP.

Chapitre 3

Modèles Déformables pour la Segmentation des Images

Chapitre rédigé par Elsa Angelini

Ce chapitre présente l'utilisation de modèles déformables pour la segmentation des images. Une introduction générale de ces modèles est fournie.

3.1 Introduction

Les modèles déformables sont des objets (contours, surfaces,..) placés dans l'espace des données d'image et qui se déforment jusqu'à atteindre une forme et une position optimale. les modèles déformables ont été utilisés sur un large domaine d'applications comme la reconnaissance de formes, la réalité virtuelle, le suivi d'objet ou la segmentation. Dans le cadre de la segmentation, cette position optimale correspond aux contours d'un ou des objets de la scène.

Les modèles déformables ont été introduits par Kaas en deux dimensions et étendus à trois dimensions par Terzopoulos.

Un modèle déformable est représenté par une courbe

$$\vec{C}(s) = [x_1(s)\ ,\ x_2(s)\ ..., x_n(s)] \tag{3.1}$$

où $s \in [0, 1]$ représente l'abscisse curviligne le long de la courbe.

On peut rappeler ici quelques éléments de géométrie en 2D. On a $d\vec{C} = \frac{d\vec{C(s)}}{ds} ds$. La paramétrisation Euclidienne du contour, en longueur d'arc l est donnée par $dl = \left\| d\vec{C} \right\| = \left\| \frac{d\vec{C}}{ds} \right\| ds$. Les dérivées du contour par rapport à la longueur d'arc donnent les vecteurs unité de la normale et de la tangente. Ainsi, $\vec{T} = \frac{d\vec{C}(l)}{dl}$ définit le vecteur unité tangent et $\frac{d\vec{T}}{dl} = \kappa \vec{N}$ définit le vecteur unité normal et la mesure de courbure κ du contour.

L'utilisation des modèles déformables pour la segmentation d'une image considère le contour $\vec{C}$ comme un système mécanique dont on cherche son état d'équilibre, correspondant à :

1. un minimum d'énergie

$$E_{totale}(\vec{C}) = E_{interne}(\vec{C}) + E_{externe}(\vec{C}) \tag{3.2}$$

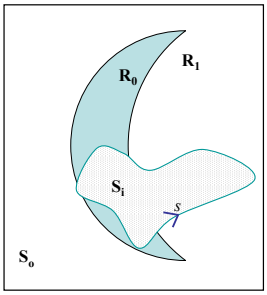


FIG. 3.1 – Définition des régions sur un modèle déformable et les contours de l'objet à segmenter

2. une position immobile, d'équilibre de forces :

$$\mu\frac{\partial^2\vec{C}}{\partial t^2}+\gamma\frac{\partial\vec{C}}{\partial t}=\vec{F}_{interne}+\vec{F}_{interne} \tag{3.3}$$

La segmentation d'une image est alors effectuée par la donnée d'une position du contour dans l'image et l'identification du contour $\vec{C}_0$ qui correspond à une valeur d'équilibre du système. La première partie de ce chapitre présente différents termes d'énergie et de force, tels que $\vec{C}_0$ corresponde aux contours de l'objet à segmenter. La deuxième partie de ce chapitre est consacrée au problème de la minimisation de cette énergie. Ceci fait appel à des méthodes mathématique de résolution d'équations aux dérivées partielles, de méthodes variationnelles et d'analyse fonctionnelle, qui ont pris une très grande importance dans les méthodes actuelles de traitement d'image.

3.2 Représentations du contour $\vec{C}$

Le contour déformable $\vec{C}$ peut être représenté de manière paramétrique, en explicitant ses coordonnées spatiales, ou de façon implicite, comme une ligne de niveau d'une fonction scalaire.

3.2.1 Représentations paramétriques

En représentation paramétrique, le contour est déformé en contrôlant et mettant à jour les positions des ses coordonnées spatiales. On peut également utiliser une paramétrisation de la courbe, sur de fonctions Spline par exemple, et reformuler le contour et ses déformations, sur ces fonctions.

3.2.2 Représentations implicites (Level set)

En représentation implicite, le contour à déformer est représenté comme une ligne de niveau ("'level set"' en anglais) d'une fonction définie sur le domaine de l'image et à valeur scalaire. Cette approche, permet de reformuler toutes les équations d'énergie et de minimisation, avec des intégrales et des dérivées définies sur une fonction scalaire, fournissant des équations et des règles de calcul beaucoup plus simples. En terme de calcul, ceci signifie que le contour $\vec{C}$ est représenté par une fonction scalaire $\Phi(\vec{C})$:

$$\begin{array}{c} \Phi(\vec{C}) > 0, \text{ à l'intérieur de } \vec{C} \\ \Phi(\vec{C}) < 0, \text{ à l'extérieur de } \vec{C} \\ \Phi(\vec{C}) = 0, \text{sur } \vec{C} \end{array} \tag{3.4}$$

Cette approche permet de proposer des fonctionnelle pour le problème de la segmentation d'image beaucoup plus librement que dans le cadre paramétrique. Cependant, il introduit des contraintes calculatoires, nécessitant de formuler la fonctionnelle sur tout le domaine de l'image Ω et non plus sur le contour uniquement. Ceci peut amener à des résolutions beaucoup plus lentes et couteuses en calcul. Il est également important de savoir que la fonction Φ est en générale définie comme la fonction distance signée au contour définissant son niveau 0. Ce cadre permet de travailler avec une fonction scalaire de gradient constant : $\left\|\vec{\nabla}(\Phi)\right\| = 1$. Cependant, la solution des équations variationnelles sur Φ n'est en général pas une fonction distance, amenant de nombreuses instabilités numériques et la nécessité de ré-initialiser la fonction Φ à une fonction distance par rapport au niveau 0 régulièrement pendant le processus itératif de segmentation, ce qui est également couteux en calcul.

3.3 Forces de régularisation du contour $E_{interne}$

Afin de déformer le contour pour effectuer une segmentation, il est important de contraindre les déformations que peut subir le contour, en le régularisant. L'idée sous-jacente de la régularisation de contour est de dire qu'un objet a une forme lisse avec seulement quelques points éventuellement anguleux. Par analogie avec la mécanique, ceci peut se traduire par la modélisation du contour comme un objet qui possède une masse (donc une inertie) et une certaine élasticité et rigidité. On peut alors définir une énergie potentielle sur cette objet, dépendant de sa forme. Kass en introduisant les modèles déformable a proposé le terme d'énergie interne suivant :

$$E_{interne}(\vec{C}) = \alpha \int_0^1 \left\|\vec{C}\,'(s)\right\|^2 ds + \beta \int_0^1 \left\|\vec{C}\,''(s)\right\|^2 ds \tag{3.5}$$

pour un contour bi-dimensionnel $\vec{C} = [x_1(s), x_2(s)]$ défini avec une seule abcisse curviligne s. Le premier terme correspond au carré de la longueur du contour et le deuxième terme correspond au carré de la mesure de courbure du contour. Le paramètre α modélise l'élasticité du contour, contrôlant la possibilité pour 2 points de s'éloigner l'un de l'autre, et le paramètre β modélise la rigidité du contour, contrôlant la possibilité du contour de se courber.

Pour une surface déformable, paramétrée par 2 abcisses curvilignes (s, r), on introduit de façon analogique, les termes de dérivée partielle en s, r et croisée en $s \times r$

En formulation implicite, ces termes contraignent le déplacement de la fonction Φ suivant sa normale de la façon suivante :

$$\frac{\partial \Phi(\vec{C}, t)}{\partial t}\vec{N} = (\alpha + \beta\kappa)\left\|\vec{\nabla}\Phi\right\|\vec{N} \tag{3.6}$$

avec κ représentant la courbure de Φ et $\vec{N}$ la normale au contour. Ces termes ne sont définis de façon exacte qu'au niveau 0 de la fonction, et doivent être étendus a tout le domaine de l'image de manière judicieuse (par extension du plus proche voisin au niveau 0 par exemple). La courbure s'exprime en fonction de Φ de la manière suivante :

$$\kappa = div\left(\frac{\vec{\nabla}\Phi)}{\left\|\vec{\nabla}\Phi\right\|}\right) \tag{3.7}$$

3.4 Forces issues des images $E_{externe}$

Du point de vue mécanique, $E_{externe}$ peut être considérée comme une composante de l'énergie potentielle du système, qui dépend de sa position. L'énergie potentielle définit un champ de forces conservatives : $\vec{F} = -\vec{\nabla} E_{externe}$.

Afin de pouvoir segmenter un objet, il faut que ce terme d'énergie $E_{externe}$ utilise des informations issues de l'image I.

3.4.1 Approche par gradient

Une première approche très intuitive propose d'utiliser le gradient de l'image et de définir une énergie minimale aux positions de forts gradients. Une exemple très populaire est :

$$\begin{array}{c} E_{externe}(v) = -\int_0^1 \left\| \nabla I(\vec{C}) \right\|^2 ds \\ E_{externe}(v) = \int_0^1 \frac{1}{\left\| \nabla I(\vec{C}) + \epsilon \right\|^2} ds \end{array} \tag{3.8}$$

Une limitation de cette approche est que le terme d'énergie ne prends pas en compte l'information de direction de gradient.

De plus il arrive souvent que la position initiale du contour ne soit pas assez proche de l'objet et ne soit donc pas attiré par les gradients de celui ci, ou que le contour s'arrête sur des gradients issus du bruit de l'image, détectés avant celui de l'objet visé. Pour résoudre ce problème, deux modifications ont été proposées.

1. Tout d'abord on peut ajouter une **force de ballon**, orientée suivant la normale au contour, qui " gonfle" ou dégonfle le contour.

$$\begin{array}{c} E_{externe}(v) = -\int_0^1 P(\vec{C}) ds = -\int_0^1 k \times dA_{int}\, \vec{C} ds \\ \vec{F}(v) = -\frac{\vec{\nabla} P}{\left\| \vec{\nabla} P \right\|} = k\vec{N} \end{array} \tag{3.9}$$

2. Deuxièmement, on peut créer un champ de vecteurs $\vec{G} = [G^1, G^2]$ sur toute l'image, dérivé des contours forts de l'image, agrandissant ainsi le domaine d'influence de ceux ci. Cette approche est appelée **flot de vecteur de gradient** (" gradient vector flow (GVF)") et se crée par la minimisation de l"énergie suivante :

$$E_{GVF}(I) = \int_\Omega \mu \left({G_x^1}^2 + {G_y^1}^2 + {G_x^2}^2 + {G_y^2}^2 \right) + \left\| \vec{\nabla}(I_{edge}) \right\|^2 \times \left\| \vec{G} - \vec{\nabla}(I_{edge}) \right\|^2 dxdy \tag{3.10}$$

avec I_{edge} qui représente une carte de contour calculée sur l'image I et G_x^i qui représente la dérivée de la composante i de $\vec{G}$ suivant x. Le premier terme correspond à une contrainte de régularité sur le champ de vecteur $\vec{G}$ tandis que le deuxième terme contraint ce champ de vecteurs à être similaire au gradient de la carte de contour $\vec{\nabla}(I_{edge})$ là où il y a des valeurs fortes de contour. L'intérêt d'utiliser une carte de contour est que le gradient de celle ci pointe vers les contours, orthogonalement aux contours et à de fortes valeurs uniquement là ou l'image à de forts contours.

3.4.2 Approche par mesures d'homogénéités

Une approche originale a été proposée par Mumford et Shah pour partitionner l'image I en zones homogènes créant une image I_0 et un nombre fini de discontinuités contenues dans le vecteur $\vec{\Gamma}$. Cette approche propose de segmenter une image en minimisant l'énergie suivante :

$$E_{MS}(\vec{\Gamma}, I_0) = \gamma \int_{\Omega|\Gamma} \left\| \vec{\nabla} I_0 \right\|^2 d\Omega + \gamma \oint_{\vec{\Gamma}} ds + \lambda \int_{\Omega|\Gamma} (I - I_0)^2 d\Omega \tag{3.11}$$

Le premier terme garantit que l'approximation de l'image par I_0 est régulière avec peu de variations. Le deuxième terme minimise le nombre de discontinuités (i.e. contours) trouvés dans l'image, et le troisième terme garantit que l'approximation I_0 est proche de l'image I. Il est important de noter que dans le cas où I_0 est constante par morceaux, le premier terme est nul.

Une approche très populaire a été proposée plus récemment par Chan et Vese pour une approximation constante ou lisse par morceaux.

$$E_{CV}(\vec{\Gamma}, c_1, c_2) = \alpha \oint_{\vec{\Gamma}} dA + \gamma \oint_{\vec{\Gamma}} ds + \lambda_1 \int_{inside\ \vec{\Gamma}} (I - c_1)^2 d\Omega + \lambda_2 \int_{outside\ \vec{\Gamma}} (I - c_2)^2 d\Omega \tag{3.12}$$

où le premier terme vise à minimiser l'aire inclue dans le contour $\vec{\Gamma}$ et où c_1 et c_2 représentent les valeurs moyenne de l'image à l'intérieur et à l'extérieur de la courbe $\vec{\Gamma}$. Cette approche est classiquement implémentée par une formulation implicite du contour dans une fonction Φ. La formulation implicite des quatre termes de l'énergie donne des intégrales sur tout le domaine Ω de l'image en utilisant une fonction porte (Heaviside) $H(\Phi)$ définie comme égale à 1 à l'intérieur du contour $(\Phi < 0)$ et 0 en dehors, ainsi que sa fonction dérivée δ qui correspond au dirac défini sur le contour $(\Phi = 0)$. Cette modélisation est illustrée Fig. 3.2

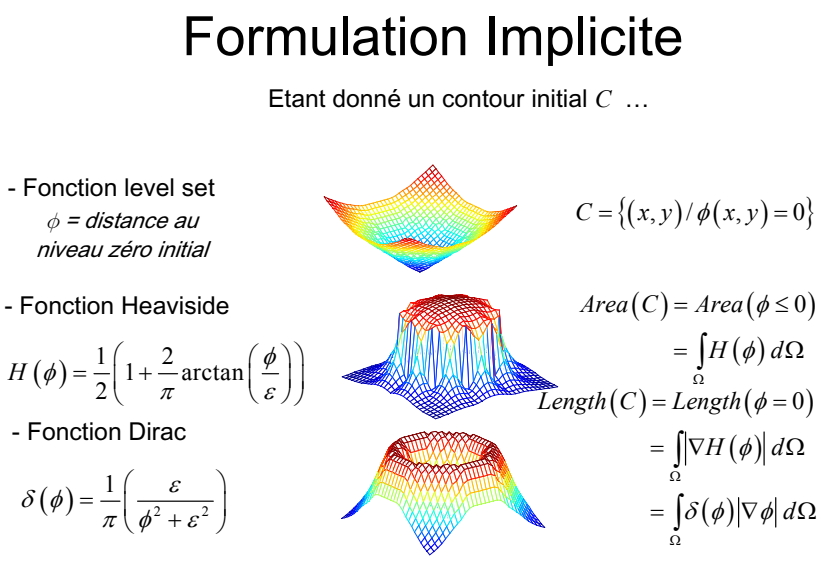


FIG. 3.2 – Formulation implicite d'un modèle déformable et les fonctions associées pour la résolution numérique

3.4.3 Approches statistiques

Pour formuler une approche statistique des informations d'image, on modélise le problème de segmentation de l'image comme le partitionnement de cette image en 2 régions R_0 et R_1, comme illustré Fig. 3.1. On utilise alors une fonction de probabilité $P(I)$ pour formuler l'énergie $E_{externe}$ comme :

$$E_{externe}(\vec{C}) = -\int_{S_i} log\left(P\left[I(\vec{C})|s \in R_0\right]\right)ds - \int_{S_o} log\left(P\left[I(\vec{C}|s \in R_1\right]\right)ds \tag{3.13}$$

Ici, S_i représente la zone de l'image intérieure au contour et S_o la zone extérieure. Cette énergie $E_externe$ est minimale quand $S_i = R_0$ et $S_o = R_1$.

On peut réécrire cette énergie avec des intégrales définies sur le sous-domaine S_i :

$$E_{externe}(v) = -\int_{S_i} log\left(P\left[I(\vec{C})|s \in R_0\right]\right)ds + \int_{S_i} log\left(P\left[I(\vec{C}|s \in R_1\right]\right)ds - \int_{S_i \cup S_o} log\left(P\left[I(\vec{C})|s \in R_0\right]\right)ds \tag{3.14}$$

On observe que le dernier terme, ne dépendant pas de la position du contour $V(s)$, n'influencera pas le processus de minimisation de $E_{externe}$. On peut donc l'éliminer et garder le forme d'énergie simplifiée :

$$E_{externe}(v) = -\int_{S_i} log\left(\frac{P\left[I(\vec{C})|s \in R_0\right]}{P\left[I(\vec{C}|s \in R_1\right]}\right)ds \tag{3.15}$$

Par exemple, si les deux régions ont des intensités qui suivent des lois de probabilité Gaussiennes $G(\mu_0, \sigma_0)$ et $G(\mu_1, \sigma_1)$, ont peu regarder le cas de figure où les variances sont égales : $\sigma_0 = \sigma_1 = \sigma$. Ceci peut être le cas pour une image constante par morceaux, dégradée par un bruit blanc uniforme. Dans ce cas, en supposant $\mu_0 > \mu_1$, le terme d'énergie devient :

$$E_{externe}(v) = \int_{S_i} 2\frac{\mu_0 - \mu_1}{\sigma^2}\left(I(\vec{C}) - \frac{\mu_0 + \mu_1}{2}\right)ds \tag{3.16}$$

On voit ici que le processus de minimisation mettra dans S_i tous les pixels de valeur supérieure à la moyenne des deux moyennes des distributions gaussiennes.

Dans le cas où les variances sont différentes, le terme‘’dans l'intégrale devient quadratique pour I. Il est intéressant de noter que cette approche permet théoriquement de résoudre le cas extrême où les moyennes des distributions sont identiques, mais les variances différentes. Quand les distributions des niveaux de gris ne sont pas connues, leurs paramètres sont estimés itérativement, durant la segmentation, en supposant $R_0 = S_i$ et $R_1 = S_o$.

3.4.4 Approches géodésiques

En relation avec l'approche classique des modèles déformables combinant contraintes de régularisation et information de gradients de l'image, une formulation géodésique a été proposée. L'idée est alors de trouver un chemin minimal (i.e. géodésique) pour une fonctionnelle $g(\vec{C})$ qui prend des valeurs minimales sur les forts gradients de l'image. L'énergie à minimiser s'écrit dans ce cas :

$$E(\vec{C}) = \int_0^1 g\left(\vec{\nabla} I\left(\vec{C}(s)\right)\right)\left\|\vec{C(s)}'\right\|ds \tag{3.17}$$

où g est une fonction positive continue et décroissante.

L'équation dynamique associée, pour une minimisation par descente de gradient, est donnée par :

$$\frac{\partial \vec{C}}{\partial t} = \left(\kappa g(I) - \left\langle \vec{\nabla} g(I), \vec{N} \right\rangle\right) \vec{N} \tag{3.18}$$

En formulation implicite, cela donne pour l'equation d'évolution du contour suivant la normale :

$$\frac{\partial \Phi}{\partial t} = \left(\kappa g(I) \left\| \vec{\nabla} \Phi \right\| + \left\langle \vec{\nabla} g(I), \vec{\nabla} \Phi \right\rangle\right) \tag{3.19}$$

Dans le cas très simple où l'on n'utilise pas de fonction g, on trouve l'énergie associée au mouvement par courbure, qui transformera n'importe quel contour en un cercle puis un point, par déformations dirigées suivant les normales au contour. Le mouvement par courbure est le mouvement qui minimise une forme donnée le plus rapidement possible au sens de la métrique Euclidienne L_E de ce contour. On peut alors voir l'approche géodésique comme la minimisation de la longueur du contour au sens d'une métrique L_g, définie par la fonction g.

3.5 Méthodes d'optimisation

3.5.1 Descente de gradient

Dans une approche par descente de gradient, on introduit un paramétrisation de $\vec{C}$ avec le temps t et on cherche le minimum du terme d'énergie $E(\vec{C}, t)$ par rapport au temps, suivant la direction de plus grande pente (i.e. le gradient). Ainsi, le terme général de l'énergie interne dans l'equation 3.5 est minimisé par :

$$\frac{\partial \vec{C}}{\partial t} = \frac{\partial}{\partial s}(\alpha \frac{\partial \vec{C}}{\partial s}) + \frac{\partial^2}{\partial s^2}(\beta \frac{\partial^2 \vec{C}}{\partial s^2}) \tag{3.20}$$

Par analogie, pour une surface minimisante, la descente de gradient est définie avec les dérivées partielles en (s, r)

On résout alors cette equation par différence finie ou éléments finis, en résolvant le système linéaire :

$$(Id + \Delta t A)\vec{C}^{\,t} = (\vec{C}^{\,t-1}) \tag{3.21}$$

Ce système fait apparaitre la matrice A qui caractérise les contraintes de forces internes sur le contour. Pour des schémas numériques explicites de 1er ordre pour les dérivées spatiales, cette matrice est penta-diagonale. Le calcul de cette matrice et son inversion à chaque itération peut être assez lourd en calculs.

En rajoutant le terme $F = -\nabla P$ avec $E_{externe} = \int_0^1 P(\vec{C})$ on a :

$$(Id + \Delta t A)\vec{C}^{\,t} = (\vec{C}^{\,t-1} + \Delta t F(\vec{C}^{\,t-1}) \tag{3.22}$$

Dans une approche implicite, il faut estimer les dérivées de la fonctionnelle en fonction de tous les paramètres de celle ci. Ces dérivées sont appelées les équation d'Euler-Lagrange de la fonctionnelle à minimiser.

De façon générale, la segmentation est donc effectuée par un processus d'optimisation itératif qui nécessite un critère d'arrêt. En général, on évalue la stabilité de la position du contour entre chaque itération et on arrête le processus itératif quand la déformation globale du contour passe en dessous d'un certain seuil. Le choix des paramètres de discrétisation spatiale et temporelle est très important et dépend du type de schéma numérique utilisé. Le choix des paramètres des fonctionnelles ou des énergie est également délicat et se fait généralement de façon empirique, en fonction de l'image à segmenter. Les méthodes de descente de gradient permettent de minimiser le problème donné, à partir d'une position initiale du contour qui peut prendre une très grande importance lorsque le problème n'est pas convexe, et possède donc des minima locaux. En d'autres termes, une méthode de descente de gradient fournira en général un minimum local au problème de la segmentation, dépendant de la position initiale.

3.5.2 Coupure de graphe (Graph cut)

Afin de pallier aux problèmes d'optimisation locale des énergies ou fonctionnelles proposées pour la segmentation par modèles déformables, des approches par coupure de graphes ont été proposées récemment. Ces approches sont décrite en détail dans une autre partie du cours.