DIRO
IFT 6150

TRAVAIL PRATIQUE N° 5

SEGMENTATIONS MV ET MARKOVIENNE

Max Mignotte

DIRO, Département d'Informatique et de Recherche Opérationnelle.
http : //www.iro.umontreal.ca/∼mignotte/ift6150
e-mail : mignotte@iro.umontreal.ca

INTRODUCTION

Dans ce TP, on se propose d'explorer plusieurs méthodes de segmentation spatiale. Une segmentation spatiale est une opération ayant pour but de diviser une image en régions ayant certains paramètres (ici un niveau de gris) en commun. Pour illustrer ce processus, trois images segmentées en deux classes sont présentées à la Figure 1. Pour ces trois exemples, les classes correspondent respectivement aux niveaux de gris sombres (classe 0) et aux niveaux de gris clairs (classe 1). Le résultat de chaque segmentation est contenu dans une image qu'on appelle un *champ d'étiquettes* (*label field* en anglais) et qu'on représente communément par la lettre x. Les pixels de l'image x contiennent une étiquette identifiant à quelle classe (0 ou 1) appartiennent les pixels. Par conséquent, partant d'une image d'entrée y connue, une segmentation spatiale a pour objectif d'estimer un champ d'étiquettes x initialement inconnu.

A. SEGMENTATION SUPERVISÉE PAR SEUIL

Partant d'un seuil fourni par l'usager, segmenter en deux classes l'image d'entrée (y) à l'aide d'un seuil appliqué indépendamment sur chaque pixel. Sauvegarder le résultat dans une image binaire (x) dont chaque pixel contient l'étiquette calculée.

B. SEGMENTATION PAR SEUIL AU SENS DU MAXIMUM DE VRAISEMBLANCE

Dans cette section, on se propose de segmenter une image en deux classes à l'aide d'un seuil *optimal* apprit à l'aide d'un critère statistique. Le critère ici retenu est le *maximum de vraisemblance* (MV) (*maximum likelihood* en anglais). Pour fonctionner, le MV a besoin de connaître le profil statistique de chacune des classes. En effet, à l'aide du profile statistique de chaque classe, le MV peut quantifier la probabilité d'appartenance de chaque pixel à chacune des classes de l'image. Exemple : un pixel ayant un niveaux de gris n_g aura une probabilité $p(n_g|0)$ conditionnée à la classe 0 et une probabilité $p(n_g|1)$ conditionnée à la classe 1[1]. En supposant que la distribution de chaque classe suit une loi gaussienne[2] , il s'ensuit que

$$\begin{aligned} & p(n_g|0) &=& \mathcal{N}(n_g;\mu_0,\sigma_0^2), \\ \text{et} \qquad & p(n_g|1) &=& \mathcal{N}(n_g;\mu_1,\sigma_1^2) \end{aligned}$$

où (μ_i,σ_i) sont la moyenne et l'écart-type de la classe i. Pour segmenter une image y, le MV affecte à chaque

[1] p(a|b) est une probabilité conditionnelle qui se lit ainsi : la probabilité d'observer a étant donné b.

[2] $\mathcal{N}(g;m_0,\sigma_0^2)=\frac{1}{\sqrt{2\Pi\sigma_0^2}}\exp\Big(-\frac{(g-m_0)^2}{2\sigma_0^2}\Big)$.

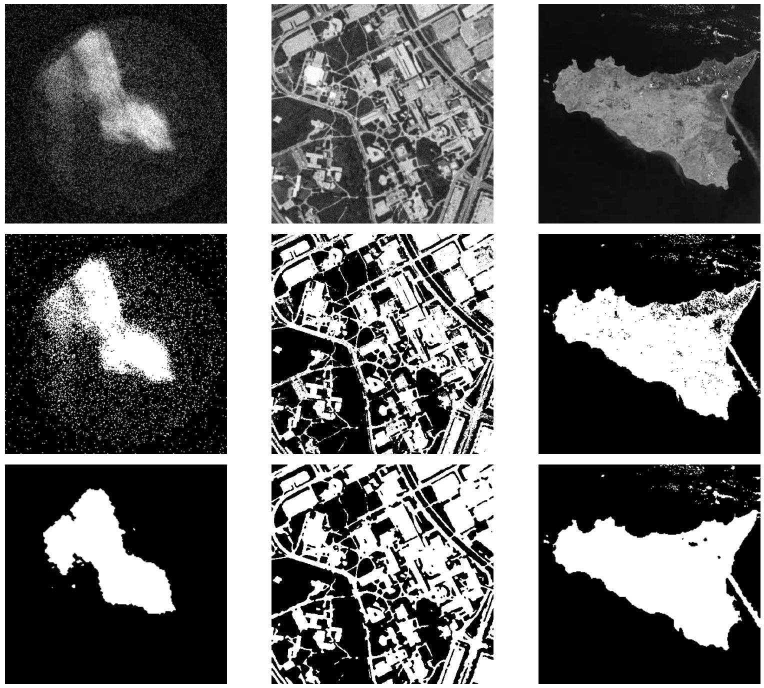

FIG. 1 – *Images* coeur.pgm, satelliteVille.pgm *et* satelliteIles.pgm *avec les champs d'étiquettes binaires obtenus à l'aide des algorithmes* K-MOYENNES *et* ICM.

pixel la classe dont la probabilité est la plus grande, c'est-à-dire

$$\begin{aligned} x(i,j) &= \arg\max_k P(n_g|k) \\ &= \arg\max_k \mathcal{N}(n_g; m_k, \sigma_k^2) \end{aligned}$$

Il arrive parfois que les paramètres statistiques des deux classes (ici (μ_0, σ_0) et (μ_1, σ_1)) soient inconnus au moment de la segmentation. Advenant une telle situation, il faut les estimer à l'aide d'un algorithme d'estimation de paramètres tel l'algorithme des K-Moyennes. Cet algorithme fonctionne de façon à minimiser l'indice de dispersion quadratique suivant :

$$J = \sum_{(i,j)} \mid y(i,j) - \mu_{x(i,j)} \mid^2,$$

où x est un champ d'étiquettes et y l'image d'entrée à segmenter. On peut montrer facilement que, pour un champ d'étiquettes donné x, l'indice de dispersion J est minimal lorsque le centre de chaque classe μ_i est donné par $\mu_k = 1/N_k \sum_{x(i,j)=k} y(i,j)$ où $k \in [0,1]$ et N_k est le nombre d'échantillons appartenant à la classe k. L'algorithme des K-Moyennes est résumé dans le Tableau 1[4].

Dans ce numéro, vous devez donc faire une segmentation au sens du Maximum de Vraisemblance (MV) à l'aide de l'algorithme des K-Moyennes.

C. SEGMENTATION MARKOVIENNE SUPERVISÉE AVEC l'ICM

Dans ce numéro, on se propose de modéliser mathématiquement des informations dites *a priori* sur la solution. L'hypothèse *a priori* qu'il vous faudra implémenter est le fameux modèle d'Ising qu'on peut résumer

[4] On dit que cette segmentation est *aveugle* car chaque pixel est classé indépendamment de ses voisins.

□ **Algorithme K-Moy.**

| y : Image à segmenter.

- Initialiser le centre de chaque classe à l'aide de 2 échantillons choisis au hasard dans y

$$\begin{array}{rcl} \mu_k^{[0]} & = & y(i,j), \quad k \in [0,1] \\ \mu_0^{[0]} & \neq & \mu_1^{[0]}. \end{array}$$

- Calcul de $x^{[p+1]}$ et $\mu_k^{[p+1]}$ à partir de $x^{[p]}$ et $\mu_k^{[p]}$, p=0

1. Tout d'abord, il faut estimer $x^{[p]}$. Pour ce faire, à chaque pixel (i,j), associer l'étiquette k tel que

$$\| y(i,j) - \mu_k^{[p]} \| < \| y(i,j) - \mu_l^{[p]} \|, \quad \forall\, k \neq l.$$

   En fait, on associe $x(i,j)$ à la classe k dont le centre μ_k est le plus proche de la valeur $y(i,j)$. En cas d'égalité, on associe $x(i,j)$ arbitrairement au groupement 0 ou 1.

2. Maintenant que $x^{[p]}$ est connu, déterminer le nouveau centre de chaque classe à l'aide d'une simple moyenne,

$$\mu_k^{[p+1]} = \frac{1}{N_k} \sum_{x(i,j)=k} y(i,j), \quad \forall\, k.$$

   avec N_k le nombre de pixels appartenant à la classe k.

3. $p \leftarrow p+1$
4. Si $\mu_k^{[p+1]} = \mu_k^{[p]}, \quad \forall k$ fin de l'algorithme sinon on retourne en 1.
5. Estimer la variance des deux classes.

□ **ALGORITHME ICM**

| $x[0]$: Configuration initiale.
| y : Image à segmenter.

- Initialisation : on choisit pour $x^{[0]}$ une segmentation au sens du MV (cf. question B)
- Calcul de $x^{[p+1]}$ à partir de $x^{[p]}$

1. On balaie l'ensemble des pixels s de l'image (par exemple de gauche à droite et de bas en haut de l'image)
   - En chaque pixel s de coordonnée (i,j) on calcule U_0 et U_1

$$U_0 = -\ln\Big(\mathcal{N}(y(i,j); m_0, \sigma_0^2)\Big) + \alpha \times \text{ Nb de voisins avec une classe} \neq \text{de } 0$$
$$U_1 = -\ln\Big(\mathcal{N}(y(i,j); m_1, \sigma_1^2)\Big) + \alpha \times \text{ Nb de voisins avec une classe} \neq \text{de } 1$$

   - On sélectionne la classe ayant une énergie minimale, c'est-à-dire :
     Si $U_0 > U_1$ alors $x^{[p]}(i,j) = 1$, autrement $x^{[p]}(i,j) = 0$.

2. $p \leftarrow p+1$
3. retour en 1 si $x^{[p]} \neq x^{[p-1]}$,sinon fin de l'algorithme.

ainsi : *un pixel a de grandes chances d'avoir la même classe que ses voisins* (i.e., on souhaite que x contienne des formes homogènes). Nous verrons plus loin comment le modèle d'Ising s'exprime mathématiquement mais avant, voyons quel critère le champ d'étiquette x doit satisfaire pour être *optimal.*

Dans cette section, nous recherchons la solution x qui maximise la fonction,

$$\hat{x} \quad = \quad \arg\max_x P(x|y) \tag{1}$$

où $P(x|y)$ est une loi de probabilité conditionnelle *a posteriori* qu'on lit : «la probabilité d'observer le champ d'étiquettes x étant donné l'image d'entrée y». Un champ d'étiquette qui maximise la fonction $P(x|y)$ porte le nom de *maximum a posteriori*(MAP). En vertu du théorème de Bayes, on peut redéfinir (1) par

$$\hat{x} = \arg\max_x \frac{P(y|x)P(x)}{P(y)} \tag{2}$$

et, sans perte de généralité,

$$\hat{x} = \arg\max_x P(y|x)P(x) \tag{3}$$

puisque $P(y)$ ne dépend pas de x. En supposant que x et y sont des champs markoviens et que y contient du bruit non corrélé, on peut dire que

$$\hat{x} = \arg\max_x \prod_{i,j} P(y(i,j)|x(i,j))P_{ij}(x(i,j)) \tag{4}$$

où y est l'image à segmenter, $P(y(i,j)|x(i,j))$ est la probabilité dite *de vraisemblance* du pixel (i,j) et $P_{ij}(x(i,j))$ est la probabilité *a priori* associée au pixel (i,j). Pour simplifier les calculs, il est fréquent de redéfinir (4) par

$$\hat{x} = \arg\min_x \sum_{i,j} -\ln P(y(i,j)|x(i,j)) + \alpha U_{ij}(x) \tag{5}$$

$$= \arg\min_x E(x,y) \tag{6}$$

où $E(x,y)$ est l'énergie globale du système, U_{ij} est l'énergie *a priori* du pixel (i,j) et α est une constante. Remarquer que pour $\alpha = 0$, on retrouve la segmentation au sens du MV telle que décrite à la question B. Pour un pixel de coordonné (i,j),

$$P(y(i,j)|x(i,j)) = \begin{cases} \mathcal{N}(y(i,j);\mu_0,\sigma_0^2) & \text{si } x(i,j)=0, \\ \mathcal{N}(y(i,j);\mu_1,\sigma_1^2) & \text{si } x(i,j)=1, \end{cases}$$

et $\mathcal{U}_{ij}(x)$ = NOMBRE DE VOISINS DONT LA CLASSE EST DIFFÉRENTE DE $x(i,j)$ À L'INTÉRIEUR D'UNE FENÊTRE 3×3 (cf. Figure 2).

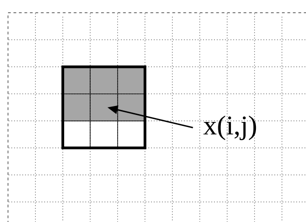

FIG. 2 – Dans cet exemple, le nombre de voisins dont la classe est différente de $x(i,j)$ est 3. pour un système de voisinage 3×3.

Dans cette section, on se propose de trouver un minimum *local* de cette fonction d'énergie (i.e., une solution sous optimale) grâce à l'algorithme Iterated Condional Modes (ICM) proposé par Besag[5]. Cet algorithme est itératif et consiste à générer une succession d'images segmentées ; i.e., $x^{[0]}, x^{[1]}, \ldots, x^{[final]}$, conduisant à la solution. Le principe de cet algorithme est donné au Tableau 2.

(1) Faire une segmentation Markovienne de l'image d'entrée y. Faites en sorte que votre programme s'arrête automatiquement lorsqu'il n'y a plus aucun changement de classe entre deux itérations (i.e., lorsque

[5]J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society*, B-48 :259-302, 1986.

$y^{[p+1]} = y^{[p]}$). Afficher pour chaque itération le numéro de l'itération, le nombre de pixels modifiés et l'énergie globale $E(x, y)$ obtenue. Normalement, l'énergie globale doit TOUJOURS diminuer d'une itération à l'autre. Observez ce qui arrive lorsqu'on varie α.

D. SEGMENTATION MARKOVIENNE SUPERVISÉE AVEC RECUIT SIMULÉ (1)

Implémenter maintenant une segmentation Markovienne au sens du Maximum a posteriori (MAP) mais avec le recuit simulé (cf. Notes de cours, chapitre 11, slide 19). On prendra comme température initiale $T_0 = 50$, Température finale $T_{\min} = 0.1$ et comme fonction de décroissance : $T_{p+1} = 0.99T_p$. Afficher pour chaque série de 10 itérations, l'énergie globale $E(x, y)$ obtenue ainsi que la température courante.

REMISE & RAPPORT

A. DATE DE REMISE ET CONDITIONS

Vous devez rendre électroniquement le(s) programme(s) fait en C (fichier .c et exécutable ainsi que la ligne de compilation ou le makefile) avant la date de remise spécifiée dans le fichier *barème* situé sur la page web du cours dans la rubrique *Introduction & Program*. Utilisez le programme *remise* (*man remise* pour plus de détails) pour remettre votre code dans le répertoire TP<Numéro du Tp>. N'oubliez pas d'inscrire vos noms, courrier électronique en commentaire en haut du fichier .c remis. Les noms des programmes à remettre devront avoir le format suivant *TpIFT6150– <Numéro du Tp>-<Numéro de la question>.c*. Les programmes devront se compiler et s'exécuter sur Linux.

B. RAPPORT

Il n'y a AUCUN rapport a remettre.