

Segmentação de Imagens Multiespectrais pelo Algoritmo ICM: Integração ao Ambiente SPRING

GUARACI J. ERTHAL¹
ALEJANDRO C. FRERY²

Instituto Nacional de Pesquisas Espaciais – INPE
Divisão de Processamento de Imagens – DPI
Avenida dos Astronautas, 1758
12227-010 São José dos Campos, SP
Brasil

¹gaia@dpi.inpe.br

²frery@dpi.inpe.br

Abstract. We present a derivation and an implementation of the ICM (Iterated Conditional Modes) algorithm for the K -ary segmentation of images. The ICM algorithm is based on the modelling of the *a priori* distribution under the Markov Random Fields framework. The algorithm is integrated within a user-friendly Geographic Informations System: the SPRING. The obtained results with optical and synthetic aperture radar images display a significant improvement with respect to the pixel-by-pixel maximum likelihood classification, at a reasonable computational cost, with the incorporation of the spatial context.

Key Words: iterative algorithms, Markov random fields, modelling, pseudolikelihood, segmentation.

1 Introdução

A segmentação de uma imagem é um procedimento pelo qual a partir de uma imagem observada \mathbf{y} obtém-se uma nova imagem $\hat{\mathbf{x}}$, onde a cada posição é atribuído um único rótulo, entre K rótulos possíveis. Para fixar idéias, pode-se pensar nesses K rótulos como diferentes cores. O intuito deste procedimento é encontrar uma partição da imagem original \mathbf{y} em K regiões homogêneas, a partir de hipóteses a respeito da distribuição da imagem verdadeira \mathbf{x} e do processo que a transforma na imagem observada \mathbf{y} . Nesse sentido, a imagem segmentada $\hat{\mathbf{x}}$ é um estimador da imagem verdadeira \mathbf{x} .

2 Técnicas de Segmentação

O método estatístico mais empregado para fazer esta partição é o método de máxima verossimilhança, que consiste das seguintes etapas: a) Estabelecer o número de populações diferentes que aparecem na imagem \mathbf{y} ; b) Estabelecer a distribuição de cada uma dessas populações; c) Estimar os parâmetros relevantes, para cada classe; e d) Atribuir cada *pixel* à classe cujo valor de verossimilhança é máximo, usando a informação coletada.

Na prática, a etapa 1 é feita pela observação da imagem, ou pelo conhecimento adquirido de outras

fontes de informação. Na etapa 2 é desejável levar em conta características da parte física do processo de formação da imagem. A etapa 3 é realizada escolhendo-se pequenas sub-regiões representativas de cada classe, e calculando-se certas estatísticas delas, tais como média amostral, variância amostral, etc. A etapa 4 consiste em procurar-se, entre as K classes, aquela cuja densidade atinge o valor máximo, em cada valor de *pixel*.

O método pode não funcionar adequadamente em algumas situações, por ser derivado de algumas hipóteses não necessariamente realistas, tais como: a) equiprobabilidade de classes e b) independência da classe (observação) de uma posição com respeito à verdadeira classe (observação) das posições vizinhas, dadas as respectivas verdades terrestres.

Supondo válida a hipótese de normalidade para as observações, o método de máximo a posteriori pontual para atribuir a observação $\mathbf{y}_s = (y_1(s), \dots, y_k(s))$ a uma dentre K classes é dado pelo critério: atribuir a observação \mathbf{y}_s à classe $\ell \in \{1, \dots, K\}$ tal que

$$f_\ell(\mathbf{y}_s) > f_j(\mathbf{y}_s) \forall \ell \neq j \in \{1, \dots, K\},$$

onde $f_j(\mathbf{y}_s)$ é o logaritmo de $f'_j(\mathbf{y}_s)$, a densidade de uma variável aleatória com distribuição normal k -variada com vetor de médias $\boldsymbol{\mu}_j$, matriz de covariân-

cias Σ_j e probabilidade *a priori* π_j :

$$\ln \pi_j - \frac{1}{2} \left[(\mathbf{y}_s - \boldsymbol{\mu}_j)^T (\Sigma_j)^{-1} (\mathbf{y}_s - \boldsymbol{\mu}_j) + \ln |\Sigma_j| \right],$$

onde $|\Sigma_j|$ denota o determinante da matriz de covariâncias Σ_j . Caso $\pi_j = \pi_\ell$ para todo $j \neq \ell$, tem-se o método de máxima verossimilhança.

O procedimento de segmentação por máxima verossimilhança, tal como formulado acima, não permite incorporar nenhum tipo de informação *a priori*. Já procedimentos derivados da modelagem Markoviana das classes, permitem incorporar a dependência espacial das observações dentro de cada classe.

2.1 Estimador ICM

O algoritmo ICM é determinístico, isto é, a unicidade de sua solução é garantida por definição.

Aqui, entende-se o algoritmo ICM como uma solução de compromisso entre qualidade de segmentação e tempo de implementação e execução. Este método consiste em procurar, iterativamente, até alcançar uma configuração de equilíbrio, o valor \hat{x}_s que maximiza $\text{IPr}(x_s \mid \mathbf{y}, \mathbf{x}_{S \setminus \{s\}})$ e em substituir x_s por \hat{x}_s , para toda coordenada s . Tem a propriedade de parar em um máximo (possivelmente local) da distribuição *a posteriori* $\text{IPr}(\mathbf{X} \mid \mathbf{Y} = \mathbf{y})$. Em [Frery-Mascarenhas, 1992] foi apresentada uma implementação deste algoritmo para imagens de radar de abertura sintética.

Denotaremos por S o suporte de uma imagem. Por definição $S \subset \mathbf{Z}^2$ é finito e da forma $S = S_1 \times S_2$, com $S_1 = \{1, \dots, m\}$ e $S_2 = \{1, \dots, n\}$. A função indicadora do conjunto A será denotada $\mathbb{I}_A(\mathbf{x})$, isto é

$$\mathbb{I}_A(\mathbf{x}) = \begin{cases} 1 & \text{se } \mathbf{x} \in A \\ 0 & \text{caso contrário.} \end{cases}$$

O vetor aleatório $\mathbf{X} = [X_s]_{s \in S}$ é chamado *modelo de Potts-Strauss* [Georgii, 1988] se tem por contradomínio, para cada sítio $s \in S$, os valores $\{1, \dots, K\}$, $K \geq 2$, e se a sua distribuição conjunta é

$$\text{IPr}(\mathbf{X} = \mathbf{x}) = \frac{1}{Z_\beta} \exp \left(\beta \sum_{\|s-u\|=1} \mathbb{I}_{\{x_s, x_u\}}(x_u) \right) \quad (1)$$

com $\beta \in \mathbb{R}$, $Z_\beta = \sum_{\mathbf{x} \in \Xi} \exp \{ \beta \sum_{\|s-u\|=1} \mathbb{I}_{x_s, x_u}(\mathbf{x}) \}$, e o conjunto das $K^{\#S}$ configurações possíveis $\Xi = \{1, \dots, K\}^S$.

Sendo este o modelo adotado como distribuição *a priori* para o algoritmo ICM, o método consiste em se substituir, na iteração t , $\hat{x}_s(t)$ pela classe ℓ que satisfaça $\max_{\ell \in \{1, \dots, K\}} \{f_\ell(\mathbf{y}_s) \cdot \beta v_s(\ell)\}$, onde $v_s(\ell) = \#\{u \in \partial_s : x_u = \ell\}$ e $\partial_s = \{u \in S : \|s - u\| = 1\}$. O

conjunto ∂_s costuma chamar-se *vizinhança* do ponto s , e os seus elementos *vizinhos* do ponto s .

Resta, então, se conhecer o valor do parâmetro β (também chamado *parâmetro de atratividade*) para se ter a especificação completa do algoritmo. Escolhe-se estimá-lo, iteração a iteração baseado na última segmentação completa, pelo método de pseudoverossimilhança [Jensen-Møller, 1989], dado pelo zero da equação:

$$\sum_{s \in S} \left[v_s(x_s) - \frac{\sum_{\ell \in \{1, \dots, K\}} v_s(\ell) \exp(\hat{\beta} v_s(\ell))}{\sum_{\ell \in \{1, \dots, K\}} \exp(\hat{\beta} v_s(\ell))} \right] = 0. \quad (2)$$

Este estimador possui boas propriedades de consistência e eficiência assintóticas, porém, nessa forma geral, é computacionalmente muito *caro*.

Uma simplificação de ordem computacional consiste em se usar somente aquelas coordenadas de S que tenham quatro vizinhos. Para tanto, definem-se $E = \{(1, 1), (1, m), (1, n), (m, n)\}$ o conjunto das esquinas, $W = \{s = (i, j) \in S : 1 < i < m, 1 < j < n\}$ o *corpo*, e $B = S \setminus (W \cup E)$, as bordas.

Pode redefinir-se, então, $\hat{\beta}(\mathbf{x})$ como a solução da equação (2), só que somando sobre W em vez de fazê-lo sobre S . Desta maneira, o problema de se achar $\hat{\beta}$ pode dividir-se em três casos simples. Utilizar-se-á a seguinte notação:

$$\Upsilon(t) = \frac{\exp(t)}{1 + \exp(t)}.$$

Quando $K = 2$, tem-se que $\hat{\beta}$ é dado pelo zero da equação:

$$2(C_4^1 - C_0^1 + C_0^0 + C_4^1 + C_3^1) - C_1^1 + C_1^0 - 2(C_0^0 + 2C_4^1 + C_0^0)\Upsilon(4\hat{\beta}) - (C_1^0 + 2C_3^1 + C_3^0)\Upsilon(2\hat{\beta}) = 0, \quad (3)$$

onde $C_k^\ell = \#\{s \in W : x_s - 1 = \ell, \sum_{u \in \partial_s} x_u - 4 = k\}$.

Para os dois casos seguintes ($K = 3$ e $K \geq 4$) utilizar-se-á a notação $x_{\partial_s} = \{x_t, x_u, x_v, x_w\}$. Quando $K = 3$, tem-se que $\hat{\beta}$ é dado pelo zero da equação:

$$\sum_{s \in W} v_s(x_s) - 4V^4 \Upsilon(4\hat{\beta}) - V^3 \frac{3 \exp(3\hat{\beta}) + \exp(\hat{\beta})}{\exp(3\hat{\beta}) + \exp(\hat{\beta})} - 2V^2 = 0, \quad (4)$$

onde

$$\begin{aligned} V^4 &= \#\{x_{\partial_s} \subset \mathbf{x} : x_t = x_u = x_v = x_w\}, \\ V^3 &= \#\{x_{\partial_s} \subset \mathbf{x} : x_t = x_u = x_v \neq x_w\}, \text{ e} \\ V^2 &= \#\{x_{\partial_s} \subset \mathbf{x} : x_t = x_u \neq x_v = x_w\}. \end{aligned}$$

Quando $K \geq 4$, tem-se que $\hat{\beta}$ é dado pelo zero da equação:

$$\begin{aligned} \sum_{s \in W} v_s(x_s) - 4U^4 \frac{\exp(4\hat{\beta})}{\exp(4\hat{\beta}) + K - 1} - \\ - U^3 \frac{3\exp(3\hat{\beta}) + \exp(\hat{\beta})}{\exp(3\hat{\beta}) + \exp(\hat{\beta}) + K - 2} - \\ - U^2 \frac{4\exp(2\hat{\beta})}{2\exp(2\hat{\beta}) + K - 1} - \\ - U^1 \frac{2(\exp(2\hat{\beta}) + \exp(\hat{\beta}))}{\exp(2\hat{\beta}) + 2\exp(\hat{\beta}) + K - 3} - \\ - U^0 \frac{4\exp(\hat{\beta})}{4\exp(\hat{\beta}) + K - 4} = 0. \end{aligned} \quad (5)$$

onde

$$\begin{aligned} U^4 &= \#\{x_{\partial_s} \subset \mathbf{x}: x_t = x_u = x_v = x_w\}, \\ U^3 &= \#\{x_{\partial_s} \subset \mathbf{x}: x_t = x_u = x_v \neq x_w\}, \\ U^2 &= \#\{x_{\partial_s} \subset \mathbf{x}: x_t = x_u \neq x_v = x_w\}, \\ U^1 &= \#\{x_{\partial_s} \subset \mathbf{x}: x_t = x_u \neq x_v \neq x_w \neq x_i\}, \\ U^0 &= \#\{x_{\partial_s} \subset \mathbf{x}: x_i \neq x_j \forall i \neq j\}. \end{aligned}$$

É importante notar que as constantes envolvidas nas equações (3), (4) e (5) podem determinar-se com uma única leitura da última segmentação completa. Assim, o custo computacional de se determinar $\hat{\beta}$ é relativamente baixo.

O conjunto suporte $S = \{1, \dots, m\} \times \{1, \dots, n\}$ foi particionado nos quatro subconjuntos descritos abaixo (o subíndice "p" usado é por *partição*):

$$\begin{aligned} S_1 &= \{(i, j) \in S: i = 1(2)m, j = 1(2)n\}, \\ S_2 &= \{(i, j) \in S: i = 2(2)m, j = 2(2)n\}, \\ S_3 &= \{(i, j) \in S: i = 2(2)m, j = 1(2)n\}, \\ S_4 &= \{(i, j) \in S: i = 1(2)m, j = 2(2)n\}, \end{aligned}$$

Utilizamos esta partição para evitar os efeitos indesejáveis que varreduras sistemáticas podem acarretar na obtenção das segmentações [Dubes-Jain, 1989].

Finalmente, o critério de parada é dado quando, de uma iteração para a seguinte, muda apenas uma quantidade de *pixels* igual ou inferior a uma porcentagem pré-estabelecida do total de *pixels* da imagem.

3 Integração ao Ambiente SPRING

O Sistema de Processamento de Informações Georeferenciadas (SPRING) é um sistema que incorpora ferramentas de processamento de imagens, sistema de informações geográficas e banco de dados [Câmara

et al., 1992]. Está sendo desenvolvido no INPE para estações de trabalho, sistema operacional UNIX, na linguagem de programação C++.

A abordagem utilizada para o desenvolvimento do sistema foi a de programação orientada a objetos. O sistema constitui-se basicamente de um conjunto de classes (Class em C++) que modelam objetos como, por exemplo, imagens, polígonos, linhas e texto. Além de objetos, processos também são modelados utilizando o conceito de classe. Por exemplo, o processo de geração de uma composição colorida a partir de três imagens digitais é modelado por uma classe.

No SPRING, o módulo de classificação de imagens é composto, basicamente, de uma hierarquia de classes. Na raiz da árvore encontra-se a classe *GenericClassifier*. Dela derivam-se duas classes: *SupervisedClassifier* e *Isodata*. A primeira destas duas classes se especializa nas classes *MaxverPixel* e *EuclidianDistancePixel*, que modelam os classificadores por máxima verossimilhança e por distância euclidiana, respectivamente.

A integração do classificador ICM foi feita pela criação da classe *MaxverPixelICM*, como especialização da classe *MaxverPixel*. Desta forma, a implementação dos ancestrais (dados e métodos) é reutilizada diretamente para definir esta nova classe. Como exemplo de reutilização, pode-se citar o método *ClassifySampleTest()*, pertencente à classe *SupervisedClassifier*, que classifica uma amostra de teste. Já os novos métodos, tais como o método *GetBeta()* que calcula o valor do parâmetro β em cada iteração e o método *ClassifyPixels()* que classifica um conjunto de dados, foram integrados diretamente a partir de uma implementação já existente. O esforço de modificação na interface de comunicação com o usuário foi mínimo pois a premissa para a integração do ICM ao SPRING foi a de não requerer, do usuário, maiores conhecimentos para o seu uso do que os requeridos pelo MAXVER, já implementado.

4 Resultados

A seguir mostra-se um exemplo de aplicação desta técnica a uma imagem de radar de abertura sintética (SAR) aerotransportado obtida pelo sensor SAR-580 (Bandas L e X), sobre a região de Freiburg, Alemanha, com uma visada e detecção linear. Dado que o ruído neste tipo de imagens não possui distribuição Gaussiana [Frery-Mascarenhas, 1992], e que a implementação do algoritmo ICM prevê essa distribuição, aplicou-se uma transformação logarítmica, que faz com que observações provindas da distribuição Rayleigh (um modelo razoável para as bandas originais) sejam mais parecidas com observações provindas de

uma Gaussiana. Aplica-se o algoritmo MAXVER a estas duas novas imagens combinadas, estimando-se os parâmetros relevantes para segmentar em três regiões distintas; o resultado deste algoritmo mostra-se na Figura 1. Empregando a segmentação MAXVER como solução inicial, o resultado do algoritmo ICM é mostrado na Figura 2. O critério de parada para obtê-la foi: continuar iterando enquanto mudarem 0.5% ou mais do total de *pixels* de uma iteração para a seguinte.

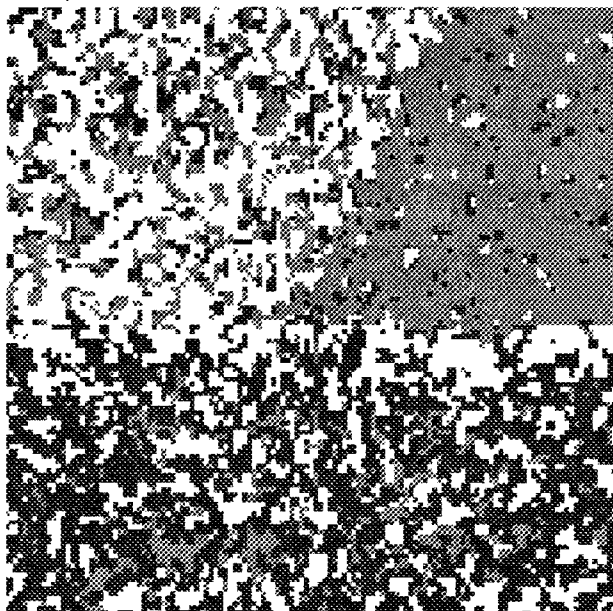


Figura 1: Segmentação por máxima verossimilhança das bandas L e X.

O algoritmo ICM, usando a segmentação MAXVER como configuração inicial, precisou de quatro iterações para gerar uma solução. Esta solução, quando comparada com a obtida pelo algoritmo de máxima verossimilhança, mostra-se bastante mais suave facilitando, assim, a sua aplicação à interpretação visual da área sob estudo.

5 Agradecimentos

Este trabalho foi parcialmente desenvolvido com recursos do Projeto Temático de Equipe No. 91/3532-2 da Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP).

6 Referências

- Besag, J. Towards bayesian image analysis. *Journal of Applied Statistics*, 16:395-407, 1989.
- Câmara, G.; Souza, R.C.M.; Freitas, U.M.; Casanova, M.A. SPRING: processamento de imagens e dados georeferenciados. In: Simpósio

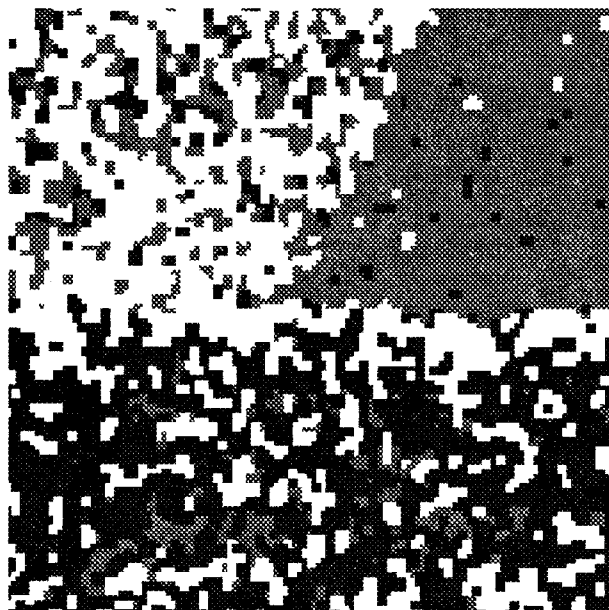


Figura 2: Segmentação pelo algoritmo ICM das bandas L e X.

- Brasileiro de Computação Gráfica e Processamento de Imagens, 5., Águas de Lindóia, nov. 1992. São José dos Campos, INPE/SBC, 1992, p. 233-242.
- Dubes, R.C.; Jain, A.K. Random fields in image analysis. *Journal of Applied Statistics*, 16(2):131-164, 1989.
- Frery, A.C.; Mascarenhas, N.D.A. Segmentação binária de imagens SAR pelo algoritmo ICM. In: Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens, 5., Águas de Lindóia, nov. 1992.
- Geman, S.; Geman, D. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6:721-741, 1984.
- Georgii, H. *Gibbs Measures and Phase Transitions*. Holland, Walter De Gruyter, 1988. (De Gruyter Studies in Mathematics, 9).
- Jensen, J.; Møller, J. Pseudolikelihood for Exponential Family Models of Spatial Processes. Research Report; Department of Theoretical Statistics, Århus University. 1989.