

Restauration automatique d'images satellitaires par une méthode MCMC

André JALOBEANU¹, Laure BLANC-FÉRAUD¹, Josiane ZERUBIA¹

¹Ariana - projet commun CNRS/INRIA/UNSA
Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles, B.P. 93, 06902 Sophia Antipolis Cedex, France
Ce travail a été effectué en collaboration avec le GdR ISIS (CNRS)

Prenom.Nom@inria.fr

Résumé – Le problème que nous abordons ici est la déconvolution d'images satellitaires floues et bruitées. Les dégradations sont supposées connues. Nous utilisons un modèle de régularisation introduisant une fonction φ , qui interdit l'amplification du bruit tout en préservant les discontinuités de l'image cherchée. Ce modèle variationnel admet deux hyperparamètres, qui sont estimés de manière automatique. Nous utilisons pour cela l'estimateur du maximum de vraisemblance appliqué à l'image observée. Nous avons développé un algorithme d'estimation MCMC qui fait appel à une méthode d'échantillonnage inspirée des travaux de [6], pour lequel nous utilisons une transformée en cosinus. Nous présentons ici un nouvel algorithme de déconvolution, rapide, dérivé de cette méthode d'échantillonnage, permettant simultanément de restaurer l'image dégradée et d'estimer les hyperparamètres.

Abstract – The problem presented herein is the reconstruction of blurred and noisy satellite images. Image degradations are supposed to be known. We use a regularizing model based on a φ -function, avoiding noise amplification while preserving the edges of the solution. This variational model has two hyperparameters, which are automatically estimated. This is achieved by the Maximum Likelihood estimator, applied on the observed image. We have developed a MCMC estimation algorithm, which uses a sampling method inspired from [6]. We use a cosine transform instead of the Fourier transform. We propose a new and fast reconstruction algorithm, derived from the sampling method, which ables to simultaneously deblur the degraded image and to estimate the hyperparameters.

1 Introduction

Les dégradations subies par les images satellitaires sont modélisées par une convolution par un opérateur H , et par un bruit N additif, blanc et gaussien, de variance σ^2 :

$$Y = HX + N$$

où X est l'image originale, et Y l'image observée. Dans notre cas $\sigma \simeq 1.35$, et le noyau de convolution est symétrique par rapport aux lignes et aux colonnes. En présence de bruit, le problème de déconvolution est mal posé au sens d'Hadamard, c'est pourquoi il faut le régulariser.

2 Régularisation d'un problème mal posé

Nous utilisons pour la restauration une méthode variationnelle, qui consiste à minimiser un critère U traduisant l'attache aux données et le modèle de régularisation :

$$U(X, \lambda, \delta) = \|Y - HX\|^2 / 2\sigma^2 + \lambda^2 \sum_{i,j} [\varphi(D_x X_{ij} / \delta) + \varphi(D_y X_{ij} / \delta)]$$

X est la solution, $D_{x,y} X_{ij}$ représentent la différence entre deux pixels voisins au pixel (i, j) dans les directions x ou y , et les réels λ et δ sont les *hyperparamètres*.

La fonction φ est *non quadratique*, choisie de manière à lisser le bruit tout en préservant les contours. Elle est

définie [3] par $\varphi(t) = 2\sqrt{1+t^2} - 2$ de manière à conserver la convexité du critère.

3 Forme semi-quadratique

Pour traiter le problème de minimisation en X et l'estimation des paramètres λ et δ , nous utilisons un développement semi-quadratique [6] de la fonction φ , basé sur la transformée de Legendre [2] :

$$\varphi(u) = \arg \inf_{b \in \mathbb{R}} (b - u)^2 + \psi(b) \\ \text{avec } b_{\text{inf}} = u - \varphi'(u) / 2$$

Cela permet de remplacer la minimisation de $U(X)$ en X par la minimisation de $U^*(X, B^x, B^y)$ en X , B^x et B^y où B^x et B^y sont des variables auxiliaires associées aux gradients de l'image selon les colonnes et les lignes.

L'intérêt de cette forme semi-quadratique est d'obtenir une forme quadratique lorsque B^x et B^y sont fixées avec une matrice indépendante des données dans le terme carré :

$$U^*(X, B^x, B^y) = X^t \left[\frac{1}{2\sigma^2} H^t H + \frac{\lambda^2}{\delta^2} (D_x^t D_x + D_y^t D_y) \right] X \\ + 2 \left[\frac{1}{2\sigma^2} H^t Y + \frac{\lambda^2}{\delta} (D_x^t B^x + D_y^t B^y) \right]^t X + \text{cte}$$

Lorsque les conditions aux bords sont périodiques, les opérateurs de dérivation $D_{x,y}$ et de convolution H sont des matrices circulantes par blocs et le terme quadratique est diagonalisé par une FFT.

On préfère imposer des conditions de symétrie aux bords de l'image, ce qui évite l'introduction de discontinuités arbitraires dans le signal résultant et correspond de plus aux conditions de Neumann qui apparaissent naturellement à la dérivation du critère. La FFT sur l'image symétrisée est équivalente à une DCT (transformée en cosinus) sur l'image d'origine. Les opérateurs sont étendus à des images de taille quadruple, symétrisées par rapport aux lignes et aux colonnes, et diagonalisés par la FFT. Le temps de calcul reste le même car le volume de données à traiter est constant.

4 L'algorithme de restauration

Pour λ et δ connus, l'algorithme procède par minimisations alternées [2, 3].

$$W = \frac{1}{2\sigma^2} |\text{FFT}[h^4]|^2 + \frac{\lambda^2}{\delta^2} (|\text{FFT}[d_x^4]|^2 + |\text{FFT}[d_y^4]|^2)$$

est la forme diagonale de la forme quadratique U^* . En outre, d_x^4 , d_y^4 et h^4 sont les générateurs des matrices de dérivation et de convolution sur des images de taille quadruple et circulantes par blocs.

L'algorithme est initialisé avec l'image observée Y .

Ensuite il faut répéter les étapes suivantes (10-15 itérations sont nécessaires pour la convergence) :

- Minimisation par rapport à B^x, B^y à X fixé :
 $B^k = D_k X / \delta - \varphi'(D_k X / \delta) / 2$ avec $k = \{x, y\}$ d'après le théorème semi-quadratique.
- Minimisation par rapport à X à B^x, B^y fixés :

$$\hat{X} = \text{DCT}^{-1} \left\{ W^{-1} \left(\frac{\lambda^2}{\delta} \text{DCT} [D_x^t B^x + D_y^t B^y] + \frac{1}{2\sigma^2} \text{FFT}[h^4] \text{DCT}[Y] \right) \right\}$$

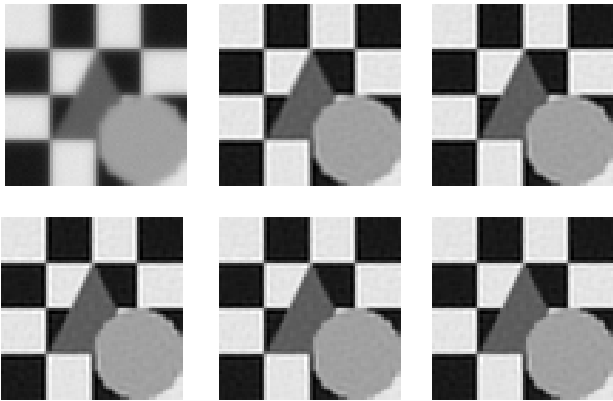


FIG. 1 – Evolution de la solution en fonction du nombre d'itérations (de 0 à 5) de l'algorithme de restauration

5 Estimation des hyperparamètres

Les résultats de la restauration par cet algorithme dépendent beaucoup de la valeur de λ et δ . Nous avons développé une méthode stochastique pour les estimer. Celle-ci est basée sur le *Maximum de Vraisemblance*.

Les hyperparamètres que nous utilisons sont ceux qui maximisent la vraisemblance de λ et δ , calculée avec l'image observée Y [11] : $(\hat{\lambda}, \hat{\delta}) = \arg \max_{\lambda, \delta} P(Y | \lambda, \delta)$.

En se plaçant dans un cadre bayésien [4], on montre que cette probabilité est proportionnelle à Z_Y / Z , où Z_Y et Z sont les constantes de normalisation correspondant aux distributions a priori $P(X | \lambda, \delta)$ et a posteriori $P(Y | X, \lambda, \delta)$. Ces constantes ne sont pas calculables directement. En revanche, il est possible d'estimer les dérivées de la log-vraisemblance par une méthode de Monte Carlo (MCMC) [9]. En considérant que les images sont des champs de Markov [1], on génère deux chaînes de Markov d'images obéissant aux deux distributions. On utilise pour cela un échantillonneur de Geman et Yang [6] modifié (avec une DCT comme pour l'algorithme de restauration ci-dessus). Cette méthode est fondée sur la forme semi-quadratique (voir paragraphe 3) de φ . L'échantillonnage est effectué successivement sur X à B^x, B^y fixés (la probabilité est gaussienne et les pixels sont tirés simultanément), et sur B^x, B^y à X fixé, les pixels de B^x, B^y étant indépendants.

Pour maximiser la $-\log$ -vraisemblance, on utilise un algorithme de descente (gradient à pas constant). Les dérivées de la vraisemblance par rapport aux hyperparamètres, qui interviennent dans cet algorithme, sont calculées au moyen d'espérances $E_X[\cdot]$ et $E_Y[\cdot]$ par rapport aux lois a priori et a posteriori. Ces espérances sont estimées par des moyennes empiriques sur les échantillons générés selon la méthode expliquée précédemment.

6 Estimation et restauration simultanées

L'estimation et la restauration sont *simultanées*, car l'algorithme d'échantillonnage de la distribution a posteriori est initialisé avec l'image restaurée avec les paramètres courants. On répète en effet les étapes suivantes :

- calcul de \hat{X} par restauration de Y avec (λ_n, δ_n) ,
- estimation des dérivées ∇_λ et ∇_δ de $\log P(Y | \lambda, \delta)$ par échantillonnage dans le plan DCT,
- calcul de $(\lambda_{n+1}, \delta_{n+1}) = (\lambda_n - \alpha \nabla_\lambda, \delta_n - \alpha \nabla_\delta)$, avec α constante positive.

Au bout de 5 à 10 itérations, l'image \hat{X} converge vers le résultat de la déconvolution avec les hyperparamètres qui maximisent la vraisemblance $P(Y | \lambda, \delta)$.

Des études expérimentales ont montré qu'une variation de moins de 10% de la valeur des hyperparamètres estimés n'a pas d'effet sur la qualité de l'image déconvoluée.

7 Résultats sur une image satellitaire extraite de Nîmes

Les images de la ville de Nîmes (figure 2) et le modèle de dégradation sont fournis par le CNES. Ce sont des simulations du futur satellite SPOT 5. L'algorithme présenté ici a été appliqué avec succès à des images de la DGA (ces données étant confidentielles, elles ne peuvent être montrées ici). Pour l'image entière (512×512) le temps de calcul est de 13 s pour l'estimation et 12.6 s pour la restauration, l'estimation étant effectuée sur une zone de 64×64 pixels (Sun Ultra 1, 167 Mhz).

Coût des algorithmes

Restauration : $100 \text{ op. pixel}^{-1} + 140 \text{ op. pixel}^{-1} \text{ iter}^{-1}$
($\simeq 10$ itérations, image 512×512)

Estimation : $6500 \text{ op. pixel}^{-1} \text{ iter}^{-1}$ ($\simeq 5$ itérations)

Estimation + restauration : Nouvel algorithme rapide d'estimation, total $1440 \text{ op. pixel}^{-1}$

8 Conclusion

Nous avons développé un nouvel algorithme automatique pour la déconvolution des images satellitaires. Les paramètres du modèle de régularisation non linéaire sont estimés par le maximum de vraisemblance. Cet algorithme converge rapidement et respecte les conditions aux bords symétriques, évitant l'apparition d'artefacts.

Il est possible d'améliorer la qualité de la restauration en introduisant des dérivées d'ordre supérieur à 1 dans le critère de régularisation. Cela introduirait alors des paramètres supplémentaires qu'il faudrait estimer.

Références

- [1] **R. Azencott** : *Image analysis and Markov fields*, in Int. Conf. on Ind. and Appl. Math. SIAM Philadelphia, 1988.
- [2] **P. Charbonnier** : *Reconstruction d'image : régularisation avec prise en compte des discontinuités*, Thèse de Doctorat, Univ. Nice-Sophia Antipolis, Sept. 1994.
- [3] **P. Charbonnier, L. Blanc-Féraud, G. Aubert, M. Barlaud** : *Deterministic edge-preserving regularization in computed imaging*, IEEE Trans. on IP, Vol 6, No 2, pp 298-311, Feb. 1997.
- [4] **S. Geman, D. Geman** : *Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images*, IEEE Trans. on PAMI., Vol 6, No 6, pp 721-741, Nov. 1984.
- [5] **D. Geman, G. Reynolds** : *Constrained restoration and recovery of discontinuities*, IEEE Trans. on PAMI, Vol 16, No 3, pp 367-383, Mar. 1992.
- [6] **D. Geman, C. Yang** : *Nonlinear image recovery with half-quadratic regularization and FFTs*, IEEE Trans. on IP, Vol 4, No 7, pp 932-946, Jul. 1995.
- [7] **C. J. Geyer** : *Markov Chain Monte Carlo Maximum Likelihood*, School of Statistics, Univ. of Minnesota, Minneapolis, MN 55455, 1993.
- [8] **A. Jalobeanu, L. Blanc-Féraud, J. Zerubia** : *Estimation d'hyperparamètres pour la restauration d'images satellitaires par une méthode « MCMCML »*, Rapport de Recherche INRIA No 3469, Août 1998.
- [9] **C. Robert** : *Méthodes de Monte Carlo par chaînes de Markov*, Economica, Paris, 1996.
- [10] **A. N. Tikhonov** : *Regularization of incorrectly posed problems*, Sov. Math. Dokl., Vol 4, pp 1624-1627, 1963.
- [11] **L. Younes** : *Parametric inference for imperfectly observed Gibbsian fields*, Prob. Th. Fields, No 82, pp 625-645, Springer-Verlag, 1989.



a)



b)



c)

FIG. 2 – a) Image originale ©CNES, b) image observée ($SNR=16.6 \text{ dB}$), c) image restaurée (hyperparamètres estimés, $SNR=22.1 \text{ dB}$)

