HAL
open science

Estimation of Markov Random Field Prior Parameters Using Markov Chain Monte Carlo Maximum Likelihood

Xavier Descombes, Robin Morris, Josiane Zerubia, Marc Berthod

To cite this version:

Xavier Descombes, Robin Morris, Josiane Zerubia, Marc Berthod. Estimation of Markov Random Field Prior Parameters Using Markov Chain Monte Carlo Maximum Likelihood. RR-3015, INRIA. 1996. inria-00073679

HAL Id: inria-00073679

https://inria.hal.science/inria-00073679v1

Submitted on 24 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Estimation of Markov Random Field prior parameters using Markov chain Monte Carlo Maximum Likelihood

Xavier Descombes, Robin Morris, Josiane Zerubia, Marc Berthod

N° 3015

October 1996

THÈME 3

Rapport de recherche

ISSN 0249-6399

Estimation of Markov Random Field prior parameters using Markov chain Monte Carlo Maximum Likelihood

Xavier Descombes, Robin Morris, Josiane Zerubia, Marc Berthod

Thème 3 — Interaction homme-machine,
images, données, connaissances
Projet PASTIS

Rapport de recherche n° 3015 — October 1996 — 28 pages

Abstract: Recent developments in statistics now allow maximum likelihood estimators for the parameters of Markov Random Fields to be constructed. We detail the theory required, and present an algorithm which is easily implemented and practical in terms of computation time. We demonstrate this algorithm on three MRF models – the standard Potts model, an inhomogeneous variation of the Potts model, and a long-range interaction model, better adapted to modeling real-world images. We estimate the parameters from a synthetic and a real image, and then resynthesise the models to demonstrate which features of the image have been captured by the model. Segmentations are computed based on the estimated parameters and conclusions drawn.

Key-words: Estimation, Maximum Likelihood, Potts model, Chien-model, image segmentation, image restoration

(Résumé : tsvp)

* R.D. Morris was supported by a grant from the Commission of the European Communities under the HCM program.

** email: *name*@sophia.inria.fr

Unité de recherche INRIA Sophia-Antipolis
2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex (France)
Téléphone : (33) 93 65 77 77 – Télécopie : (33) 93 65 77 65

Estimation au sens du Maximum de Vraisemblance des paramètres d'un modèle markovien par méthode de Monte Carlo

Résumé : De récents développements en statistiques rendent maintenant possible l'estimation au sens du maximum de vraisemblance des paramètres associés aux champs de Markov. Dans ce rapport, nous détaillons la théorie requise et présentons un algorithme facilement implémentable et abordable du point de vue du temps CPU. Nous appliquons cet algorithme à trois modèles markoviens : le modèle de Potts, une variante inhomogène de ce modèle et au chien-modèle (modèle comprenant des interactions mettant en jeu plus de deux pixels et plus adapté pour modéliser des images réelles). Nous estimons les paramètres à partir d'images de synthèse et d'images réelles, puis re-synthétisons les modèles pour étudier les caractéristiques de l'image captées par le modèle. Nous faisons aussi des segmentations en utilisant les paramèters que nous avons estimés, et nous donnons des conclusions.

Mots-clé : Estimation, Maximum de Vraisemblance, modèle de Potts, chien-modèle, segmentation d'images, restauration d'images

Contents

1 Introduction **4**

2 Maximum Likelihood estimators **5**
2.1 The log-likelihood 5
2.2 Importance sampling 6

3 An MCMCML algorithm **8**
3.1 Estimate a robustness criterion 8
3.2 Estimation algorithm 10

4 Validation on Markovian priors **11**
4.1 The Potts model 11
4.2 An inhomogeneous variation of the Potts model 11
4.3 The chien-model 14

5 Pros and cons of the priors **18**
5.1 Image constraints modeled by priors 18
5.2 Bayesian inference using Markovian priors 18
5.3 Sampling considerations 24

6 Conclusion **26**

1 Introduction

Early vision algorithms extract some information from observed data without any specific knowledge about the scene. However, these data (remote sensing data, medical images,...) are usually disturbed by noise. To improve the algorithms, regularization techniques are used, incorporating constraints on the solution. These constraints represent a general knowledge about what a natural scene should be. A popular way to define these constraints is to consider a probabilistic model (the prior) of the expected result. Using the Bayesian approach, we search for a realization which optimizes the probability of the solution, given the data. A key point to obtain unsupervised algorithms in this paradigm is to be able to estimate the different parameters involved in the prior. Accurate estimators of these parameters are necessary to control the impact of the prior on the properties desired for the solution.

Because of their ability to model global properties using local constraints, Markov Random Fields (MRFs) are very popular priors. Several optimization algorithms converging either toward a global minimum of the energy [1] or a local one [2], [3] are now well defined. But accurate estimation of the parameters is still an open issue. Indeed, the partition function (normalization constant) leads to intractable computation. Parameter estimation methods are then either devoted to very specific models [4], [5] or based on approximations such as Maximum Pseudo Likelihood (MPL) [6], [7]. Unfortunately, these approximations lead to inaccurate estimators for the prior parameters. Maximum Likelihood Estimators (MLEs) have more interesting properties. Gidas proves in [8] that MLEs are consistent. Asymptotic normality can be reached in some special case, such as the 2-dimensional Ising model. For high dependency models, the MPL gives poor results as reported in [9]. The aim of MRFs in image processing is to obtained regularized solutions. High dependencies are required to get homogeneous realizations. Thus, MLEs should improve image segmentation and image restoration algorithms based on Markovian priors.

Markov Chain Monte Carlo algorithms (MCMC) [10] are very popular in image processing to derive optimization methods when using a Markovian prior [2], [1]. In fact, MCMC algorithms can be developed for other purposes than Bayesian inference. Indeed, they can be used to derive MLE. The partition function of Gibbs Fields can be estimated using an MCMC procedure.

A Maximum Likelihood estimation using an MCMC algorithm is proposed in [11]. Geyer proves the convergence in probability of the MCMC toward the MLE. This method can be applied to a wide range of models such a Point Processes [12] or Markov Random Fields. In this paper, we propose an estimation algorithm for Markovian prior parameters based on an MCMCML procedure. We validate this algorithm on three different priors.

In section 2, we compute the Maximum Likelihood estimators of a given Gibbs Field whose energy is linear with respect to parameters. Importance sampling is also introduced. This allows us to compute statistics of the model associated with given parameters using samples obtained with other parameter values. These results lead to an MCMCML estimation method described in section 3. Results are detailed in section 4. We consider three different Markovian priors: the Potts model, an inhomogeneous variation of the Potts model and the Chien-model. This Maximum Likelihood estimation allows us to derive some comments about the priors. Section 5 is devoted to a comparison between the priors considered. Finally, we conclude in section 6.

2 Maximum Likelihood estimators

2.1 The log-likelihood

Let P_Θ be a random field defined on S and parameterized with vector $\Theta = (\theta_i)$. We consider P_Θ to be a Gibbs Field, whose energy is linear with respect to the parameters θ_i. We then have :

$$P_\Theta(Y) = \frac{1}{Z(\Theta)} \exp - \langle \Theta, Y \rangle = \frac{1}{Z(\Theta)} \exp \left[- \sum_i \theta_i N_i(Y) \right], \tag{1}$$

where $N_i(Y)$ are functions of the configuration Y. In this paper, we consider a continuous framework for the state space Λ. Results are still valid in the discrete case by changing integrals into sums. The partition function $Z(\Theta)$ is then written:

$$Z(\Theta) = \int_{\Lambda^S} \exp \left[- \sum_i \theta_i N_i(X) \right] dX, \tag{2}$$

where S is the site set and Λ is the state space.

We consider that we have data Y. We want to fit the model to the data. The log-likelihood is then defined by :

$$\log P(Y|\Theta) = \log \left[\frac{1}{Z(\Theta)} \exp - \sum_i \theta_i N_i(Y) \right], \tag{3}$$

$$\log P(Y|\Theta) = - \sum_i \theta_i N_i(Y) - \log Z(\Theta). \tag{4}$$

The maximum likelihood estimators are obtained by maximizing the log-likelihood. We then have:

$$\forall i, \frac{\partial \log P(Y|\Theta)}{\partial \theta_i} \left(\hat{\Theta}\right) = 0, \tag{5}$$

and then:

$$\forall i, -N_i(Y) + \frac{\int_{\Lambda^s} N_i(X) \exp \left[- \sum_i \hat{\theta}_i N_i(X) \right] dX}{\int_{\Lambda^s} \exp \left[- \sum_i \hat{\theta}_i N_i(X) \right] dX} = 0, \tag{6}$$

where $\hat{\theta}_i$ is the maximum likelihood estimator of θ_i.

Denoting by $< a(x) >_\Theta$, the expectation of $a(x)$ with respect to P_Θ, we finally get:

$$\forall i, \langle N_i(X) \rangle_{\hat{\Theta}} = N_i(Y). \tag{7}$$

To evaluate the log-likelihood function, we have to compute the partition function. The partial derivatives of the log-likelihood requires the computation of the different $\langle N_i(X) \rangle_\Theta$. Unfortunately, the computation of these quantities is intractable. We can estimate the $\langle N_i(X) \rangle_\Theta$ by sampling the distribution. Nevertheless, to sample the distribution for each value of Θ is inconceivable from CPU time considerations.

2.2 Importance sampling

We introduce importance sampling to avoid having to sample the model for each value of Θ. Indeed, importance sampling allows us to estimate statistical moments corresponding to P_Θ using samples obtained from P_Ψ.

Consider first the partition function. We have:

$$Z(\Theta) = \int_\Omega \exp \left[- \sum_i \theta_i N_i(X) \right] dX, \tag{8}$$

then:

$$\begin{aligned} Z(\Theta) &= \int_\Omega \exp\left[-\sum_i (\theta_i - \psi_i) N_i(X)\right] \exp\left[-\sum_i \psi_i N_i(X)\right] dX \\ Z(\Theta) &= \int_\Omega \exp\left[-\sum_i (\theta_i - \psi_i) N_i(X)\right] Z(\Psi) dP_\Psi(X). \end{aligned} \tag{9}$$

For each couple (Θ, Ψ), the ratio of the partition functions is given by:

$$\frac{Z(\Theta)}{Z(\Psi)} = E_\Psi\left(\exp\left[-\sum_i (\theta_i - \psi_i) N_i(X)\right]\right), \tag{10}$$

where E_Ψ refers to the expectation with respect to the law P_Ψ.

The partition function corresponding to P_Θ can thus be estimated from the sampling of P_Ψ. We just have to sample the law with parameter Ψ to get an estimator of the ratio $\frac{Z(\Theta)}{Z(\Psi)}$ for all Θ by computing from the samples the expectation given by formula (10).

Consider now the log-likelihood. The maximum likelihood estimator is given by the vector Θ which maximizes formula (3). This is equivalent to minimizing the following expression:

$$-\log P_\Theta(Y) = \sum_i \theta_i N_i(Y) + \log \frac{Z(\Theta)}{Z(\Psi)}. \tag{11}$$

The partial derivative of the partition function can be written:

$$\begin{aligned} \frac{\partial Z(\Theta)}{\partial \theta_i} &= \int_\Omega -N_i(X) \exp\left[-\sum_j (\theta_j - \psi_j) N_j(X)\right] Z(\Psi) dP_\Psi(X) \\ \frac{\partial Z(\Theta)}{\partial \theta_i} &= -Z(\Psi) E_\Psi\left(N_i(X) \exp\left[-\sum_j (\theta_j - \psi_j) N_j(X)\right]\right). \end{aligned} \tag{12}$$

Then, we have:

$$\frac{\partial - \log P_\Theta(Y)}{\partial \theta_i} = N_i(Y) - \frac{E_\Psi\left(N_i(X) \exp\left[-\sum_j (\theta_j - \psi_j) N_j(X)\right]\right)}{E_\Psi\left(\exp\left[-\sum_j (\theta_j - \psi_j) N_j(X)\right]\right)}. \tag{13}$$

From a sampling of P_Ψ we thus can theoretically estimate the log-likelihood of P_Θ and its partial derivatives for all Θ. The same kind of computation allows us to compute the Hessian, and we have:

$$\frac{\partial^2 - \log P_\Theta(Y)}{\partial\theta_i \partial\theta_j} = N_i(Y)N_j(Y) - \frac{E_\Psi\left(N_i(X)N_j(X)\exp - \left[\sum_k (\theta_k - \psi_k)N_k(X)\right]\right)}{E_\Psi\left(\exp - \left[\sum_k (\theta_k - \psi_k)N_k(X)\right]\right)}. \tag{14}$$

3 An MCMCML algorithm

3.1 Estimate a robustness criterion

Consider an image Y from which we wish to compute the maximum likelihood estimator of Θ using P_Θ as a model. From the image we can extract the value of the $N_i(Y)$. Then, we can sample the law P_Ψ for a given Ψ. From the samples, the different expectations involved in formulas (10) and (13) can be estimated. Then, for all Θ, we can estimate the log-likelihood and its derivatives. An optimization algorithm (gradient descent, conjugate gradient for example) leads then to the maximum likelihood estimator of Θ, when the log-likelihood is a convex function.

Nevertheless, if the two parameters Θ and Ψ are too far from each other, the estimation of the expectations will be inaccurate. Indeed, the robustness of the estimation of the expectations requires the overlap between the two distributions P_Θ and P_Ψ to be large enough. The proposed method is practically valid only in a neighborhood of parameter Ψ. During the optimization, when the current value of Θ is too far from Ψ, we have to re-sample the model using a new value for Ψ (we take the current value of Θ for Ψ). Such a sampling requires CPU time. So, we need a criteria to define the neighborhood of Ψ on which the estimation is robust to avoid un-necessary sampling. A first idea is to use a metric between the distributions P_Ψ and P_Θ given by:

$$d\left(P_\Theta, P_\Psi\right) = \frac{1}{2}\int \left|P_\Theta(X) - P_\Psi(X)\right| dX. \tag{15}$$

By definition, we have:

$$P_\Theta(X) - P_\Psi(X) = \frac{\exp - \langle X, \Theta\rangle}{Z(\Theta)} - \frac{\exp - \langle X, \Psi\rangle}{Z(\Psi)}, \tag{16}$$

$$P_\Theta(X) - P_\Psi(X) = \left(\frac{Z(\Psi)}{Z(\Theta)} \exp\left[-\langle X, \Theta\rangle + \langle X, \Psi\rangle\right] - 1\right) \frac{\exp - \langle X, \Psi\rangle}{Z(\Psi)}. \quad (17)$$

Thus, we define a distance between the distribution by:

$$\begin{aligned} \int |P_\Theta(X) - P_\Psi(X)|\, dX &= \int \left| \left(\frac{Z(\Psi)}{Z(\Theta)} \exp\left[-\langle X, \Theta\rangle + \langle X, \Psi\rangle\right] - 1\right) \right| dP_\Psi(X), \\ &= E_\Psi \left(\left| \frac{Z(\Psi)}{Z(\Theta)} \exp\left[-\langle X, \Theta\rangle + \langle X, \Psi\rangle\right] - 1 \right| \right). \quad (18) \end{aligned}$$

By using formula (10), we then have:

$$d\left(P_\Theta, P_\Psi\right) = \frac{1}{2} E_\Psi \left(\left| \frac{Z(\Psi)}{Z(\Theta)} \exp\left[-\langle X, \Theta\rangle + \langle X, \Psi\rangle\right] - 1 \right| \right). \quad (19)$$

We can compute this distance to test the robustness of estimates and decide whether we should sample the model once more or not. However, this distance is also estimated and can be biased. Therefore, we define a heuristic criterion, considering the current samples used for estimating the expectations. For each sample X_i, we define a weight by:

$$\omega_i = U_\Theta(X_i) - U_\Psi(X_i), \quad (20)$$

$$\omega_i = \sum_j (\Theta_j - \Psi_j) N_j(X_i). \quad (21)$$

Computing the expectations, we use the following trick to avoid overflow and numerical instabilities:

$$\sum_{samples\ i} \exp\left[-\omega_i\right] = \exp\left[-\omega_{max}\right] \sum_{samples\ i} \exp\left[-(\omega_i - \omega_{max})\right], \quad (22)$$

where $\omega_{max} = \max_{samples_i} \omega_i$.

Consider the different samples. If ω_{max} is too far from most of the ω_i the quantity $\exp\left[-(\omega_i - \omega_{max})\right]$ will be close to 0 and results in a poor estimation. So, we can consider the estimation robust only if $\omega_{max} - \omega_{min}$ is lower than a given threshold.

3.2 Estimation algorithm

We can now derive an algorithm based on the conjugate gradient principle. Consider the current parameter estimate $\hat{\Theta}$ and a sampling of $P_{\hat{\Theta}}$. We can estimate the gradient and the Hessian of the log-likelihood function at $\hat{\Theta}$. We then compute the conjugate directions [13]. Along each conjugate direction we define an interval using the distance defined by equation (19), where the log-likelihood estimation is robust. We then maximize the log-likelihood along these intervals.

The algorithm can be written as follows:

1. Compute the $N_i(Y)$
2. Initialize $\hat{\Theta}_0$, $n = 0$
3. Sample the distribution $P_{\hat{\Theta}_n}$
4. Estimate the gradient and the Hessian of the log-likelihood at $\hat{\Theta}_n$, using equations (13) and (14)
5. Compute the conjugate directions Δ_i
6. For each conjugate direction define a search interval using either the distance defined in equation (19): $I_i = \left[\hat{\Theta}_n - \alpha_i \Delta_i, \hat{\Theta}_n + \beta_i \Delta_i\right]$ where $\alpha_i = sup_{\alpha \in \mathbb{R}^+} \left\{\alpha : d\left(P_{\hat{\Theta}_n}, P_{\hat{\Theta}_n - \alpha\Delta_i}\right) < T\right\}$, $\beta_i = sup_{\beta \in \mathbb{R}^+} \left\{\beta : d\left(P_{\hat{\Theta}_n}, P_{\hat{\Theta}_n + \beta\Delta_i}\right) < T\right\}$ where T is a threshold, or the proposed heuristic criterion.
7. Compute $\hat{\Theta}_{n+1}$ by maximizing the log-likelihood along each search interval using golden section search [13]
8. if $||\hat{\Theta}_{n+1} - \hat{\Theta}_n|| > T_2$ put $n = n+1$ and go back to 3, where T_2 is another threshold.

4 Validation on Markovian priors

In this section, we consider different Markov models used as priors in image processing. We validate the estimation method on these models and demonstrate the generality of its applicability.

4.1 The Potts model

The Potts model is commonly used as a prior in image segmentation. It depends on a single parameter β and is defined by:

$$P_\beta(X) = \frac{1}{Z(\beta)} \exp\left[-\beta \sum_{c=\{s,s'\}\in\mathcal{C}} \delta_{x_s \neq x_{s'}}\right], \tag{23}$$

where $\mathcal{C}$ is the set of cliques. In this case, a clique consists in two neighboring pixels. We consider the case where the lattice S is a subset of $\mathbb{Z}^2$. For simulations and estimations we have considered the 4 nearest-neighbors. The Potts model can be embedded in the general form of equation (1):

$$P_\beta(X) = \frac{1}{Z(\beta)} \exp\left[-\beta N_0(X)\right] = \frac{1}{Z(\beta)} \exp\left[-\beta \#_X\right], \tag{24}$$

where $N_0(X) = \#_X$ is the number of inhomogeneous cliques in the configuration X. The model depends on one single parameter, so the proposed algorithm is simplified as we do not have to compute the conjugate directions. Table 1 shows estimates obtained using the MCMCML method for different values of β.

4.2 An inhomogeneous variation of the Potts model

We can extend the procedure to the case of a non-stationary Potts model. Consider a Potts mode for which the parameter β depends on the localization of the clique. We suppose for simplicity that this dependency is linear and that β is written:

$$\beta_{c=\{x_{i,j},x_{p,q}\}} = a\left(\frac{i+p}{2}\right) + b\left(\frac{j+q}{2}\right) + c. \tag{25}$$

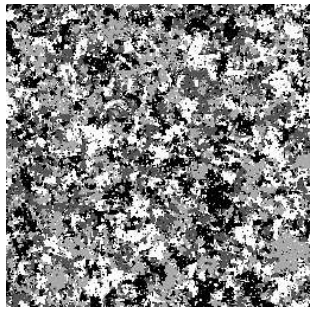
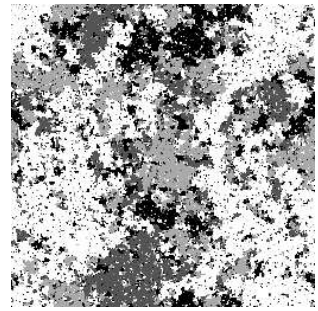
Parameters	$\beta = 0.53$ ($N_0 = 48070$)	$\beta = 0.5493$ ($N_0 = 35699$)
Estimates	$\hat{\beta} = 0.529$ ($\langle N_0 \rangle = 48072$)	$\hat{\beta} = 0.5488$ ($\langle N_0 \rangle = 35699$)
Sample		

Table 1: Estimation of the Potts model parameter

To estimate β we have to estimate a, b and c. The associated distribution is written:

$$P_{a,b,c}(X) = \frac{1}{Z(a,b,c)} \exp\left[- \sum_{c=\{s,s'\}\in\mathcal{C}} \left(a\left(\frac{i+p}{2}\right) + b\left(\frac{j+q}{2}\right) + c \right) \delta_{x_s \neq x_{s'}} \right]. \tag{26}$$

This model can be written in the form of equation (1):

$$P_{a,b,c}(X) = \frac{1}{Z(a,b,c)} \exp\left[-cN_0(X) - bN_1(X) - aN_2(X) \right], \tag{27}$$

where:

$$\begin{aligned} N_0(X) &= \#_X, \text{the number of inhomogeneous cliques} \\ N_1(X) &= \sum_{\text{inhomogeneous cliques}} \frac{j+q}{2} = N_0(X) \left\langle \frac{j+q}{2} \right\rangle_{inh.cl.} \\ N_2(X) &= \sum_{\text{inhomogeneous cliques}} \frac{i+p}{2} = N_0(X) \left\langle \frac{i+p}{2} \right\rangle_{inh.cl.} \end{aligned}$$

Table 2 shows samples from this model and the parameters estimated from these samples.

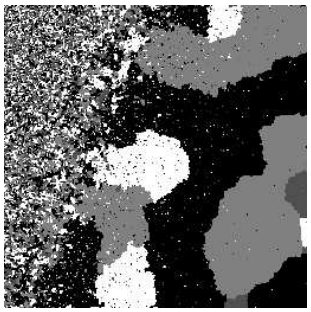
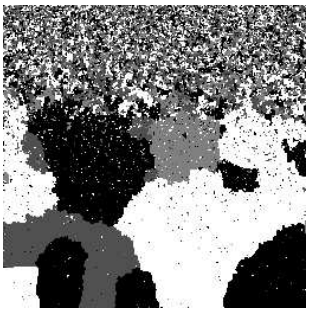
Parameters	c	0.3	N_0	26603	c	0.5	N_0	29171
	b	0.005	N_1	1703201	b	0.005	N_1	1816390
	a	0.002	N_2	2777495	a	0.0	N_2	3704620
Estimates	$\hat{c}$	0.3002	$\langle N_0 \rangle$	27960	$\hat{c}$	0.535	$\langle N_0 \rangle$	30495
	$\hat{b}$	0.0052	$\langle N_1 \rangle$	1771015	$\hat{b}$	0.0052	$\langle N_1 \rangle$	1859293
	$\hat{a}$	0.0025	$\langle N_2 \rangle$	2757615	$\hat{a}$	0.0003	$\langle N_2 \rangle$	3760871
Sample								

Table 2: Estimation of an inhomogeneous variation of the Potts model.

4.3 The chien-model

To improve segmentations some more complex models have been proposed in the last few years. These models consider cliques of more than two pixels to define more accurately local configurations and their contribution to the model. Such a model based on 3×3 cliques was proposed in [14]. Another model on an hexagonal lattice can be found in [15]. These models consider only the clique configurations. In [16], a binary model (the chien-model) taking into account links between neighboring cliques is proposed. This model has been generalized to the m-ary case in [17]. This model, although regularizing, preserves fine structures and linear shapes in images. In this model, the set of cliques is composed of 3×3 squares. The chien-model is defined from the discrimination between noise, lines and edges. Three parameters (n, l and e) are associated to these patterns.

Before constructing the model the different configurations induced by a 3×3 square are classified using the symmetries (symmetry black-white, rotations, etc.) This classification and the number of elements in each class are

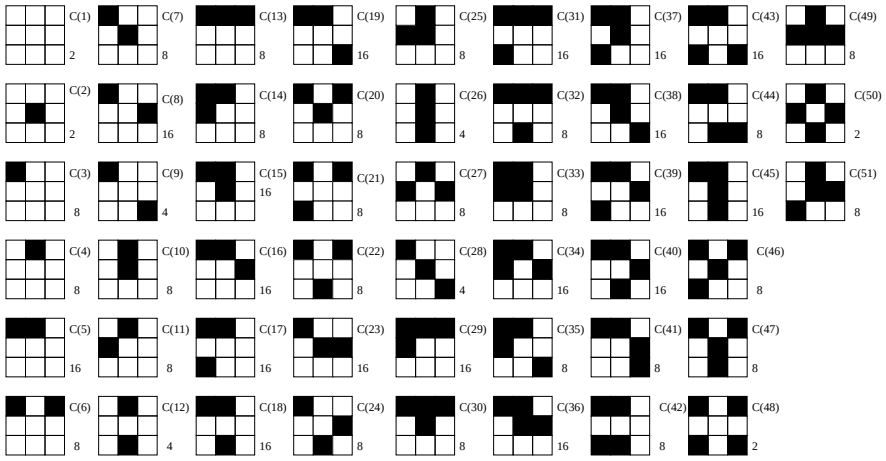


Figure 1: **The different classes induced by a binary 3×3 model and their number of elements**

described in Figure 1. A parameter is associated to each class and refers to the value of the potential function for the considered configuration. So, under the

hypothesis of isotropy of the model which induces the symmetries, we have for such a topology (cliques of 3×3) fifty one degrees of freedom. The construction of the model consists in imposing constraints by relations between its parameters. Two energy functions which differ only by a constant are equivalent, so we suppose that the minimum of the energy is equal to 0. The global realization of 0 energy are called the ground states of the model and represent the realization of maximal probability. We suppose that uniform realizations are ground states, so we have the first equation for the parameters given by $C(1) = 0$. We then define the different constraints with respect to those two uniform realizations. The first class of constraints concerns the energy of edges which is noted e per unit of length. Due to symmetries and rotations we just have to define three orientations of edges corresponding to the eight ones induced by the size of cliques. These constraints and the derived equations are represented on figure 2. These constraints are defined for features of width at

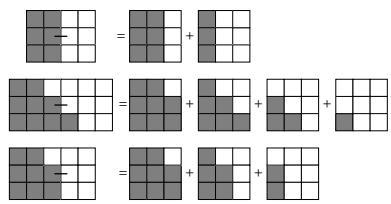


Figure 2: **Equations associated with edges constraints**

least equal to 3. For other features, we have to define other constraints. When the width of the considered object is one pixel, it is referred as a line and has energy per unit l. For larger features (double and triple lines), we consider that the energy per unit is given by $2 \times e$, which correspond to left and right edge, and get more equations. All these constraints induce eleven equations which depend on fourteen parameters leading to the following solution (see [16] for full details):

$$C(3) = C(5) = \frac{e}{4} \quad C(26) = l - e \quad C(14) = \frac{\sqrt{2}e}{4}$$
$$C(16) = C(23) = \frac{\sqrt{5}}{6}l - \frac{e}{4} \quad C(11) = C(28) = \frac{\sqrt{2}l}{3} - \frac{e}{6}$$
$$C(35) = \frac{\sqrt{2}e}{2} \quad C(29) = \frac{\sqrt{5}e}{6} \quad C(13) = C(9) = C(19) = \frac{e}{2}$$

Noise is defined by assigning to every other configuration a positive value n.

To extend the binary chien-model in an m-ary model, we define the energy of a given configuration as the sum of several energies given by the binary model. Consider a configuration and a given label σ_0. We put every pixels of the configuration which are in state σ_0 to 0 and others to 1. We then have a binary configuration. The energy of the m-ary model is the sum of the energies obtained by these deduced binary configurations for the m labels (see figure 3). The potential associated with each configuration is then a linear combination

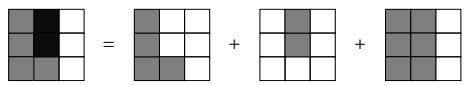


Figure 3: **M-ary extension of the chien-model**

of the three parameters e, l and n:

$$\forall i = 0, ..., 51 \; C(i) = \epsilon(i)e + \lambda(i)l + \eta(i)n. \tag{28}$$

The resulting distribution is written:

$$P_{e,l,n}(X) = \frac{1}{Z(e,l,n)} \exp\left[-eN_0(X) - lN_1(X) - nN_2(X)\right], \tag{29}$$

where:

$$\begin{aligned} N_0(X) &= \sum_{i=1,...,51} \epsilon(i)\#_i(X), \\ N_1(X) &= \sum_{i=1,...,51} \lambda(i)\#_i(X), \\ N_2(X) &= \sum_{i=1,...,51} \eta(i)\#_i(X). \end{aligned}$$

$\#_i(X)$ being the number of configurations of type i in the realization X.

Results are summarized in table 3. The chosen parameter values show the properties of the model. Indeed, we can control the total length of edges as well as the total length of lines. Moreover, parameter n allows us to control the amount of noise. The table also shows that we can accurately estimate the parameters of this model.

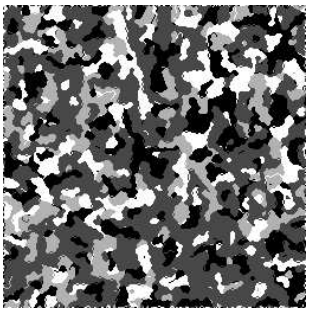
Parameters	e 0.2 N_0 32950 l 0.4 N_1 11724 n 0.6 N_2 49708	e 0.4 N_0 21089 l 0.8 N_1 1710 n 1.0 N_2 2880
Estimates	$\hat{e}$ 0.2008 $\langle N_0 \rangle$ 32857 $\hat{l}$ 0.4038 $\langle N_1 \rangle$ 11669 $\hat{n}$ 0.5997 $\langle N_2 \rangle$ 49606	$\hat{e}$ 0.3905 $\langle N_0 \rangle$ 21358 $\hat{l}$ 0.7843 $\langle N_1 \rangle$ 1585 $\hat{n}$ 0.9993 $\langle N_2 \rangle$ 2947
Sample		

Table 3: Estimation of chien-model parameters

5 Pros and cons of the priors

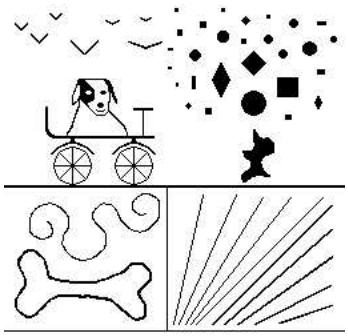
5.1 Image constraints modeled by priors

In this section, we provide tests to evaluate the properties the different models can incorporate into, for example, a segmentation result. First, we consider a binary synthetic image (see figure 4.a). A segmented SPOT image is the second proposed test (see figure 4.b). We estimate the corresponding parameters for each model and then synthesize the model using estimated parameters. In this way, we can observe the properties of the initial image which are captured by the different models. We first consider the synthetic image shown in figure 4.a. Results obtained for the Potts and chien models using a MCMCML estimation are summarized in table 4. As we consider a general model for segmented images, the realization of the models are visually far from the original image. Nevertheless, we can point out that in the case of the Potts model, the only image characteristic represented in the simulation is the number of inhomogeneous cliques. For a given number of inhomogeneous cliques, there are many more configurations composed of an uniform background with noise than configurations composed of several homogeneous shapes. Therefore, using estimated parameters, the Potts model does not seem to be a regularizing prior. As it considers cliques of 3×3 pixels, the chien-model allows us to define edge and line lengths. The realization of the chien-model obtained with estimated parameters contains different shapes. The global edge and line lengths are the same as the original image. Therefore, it seems more appropriate to be used as a prior for image modeling.

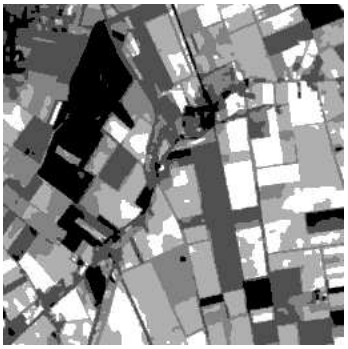
5.2 Bayesian inference using Markovian priors

We consider in this subsection a classical application to validate the previous assertions concerning the priors in this study. We first consider a noisy version of the binary synthetic image shown on figure 4.a. The original image is corrupted by a channel noise of ratio 0.15 (15% of the original pixels are reversed) (see figure 5). We perform a restoration in a Bayesian framework.

Denote the noisy image by $X = (x_s)_{s \in S}$ and the restored image by $Y = (y_s)_{s \in S}$ where S is the lattice and $s = (i, j)$ is a pixel. The data are X and we search for Y which minimizes some cost function under the probability

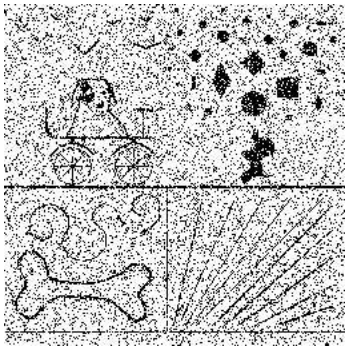


a: Binary synthetic image

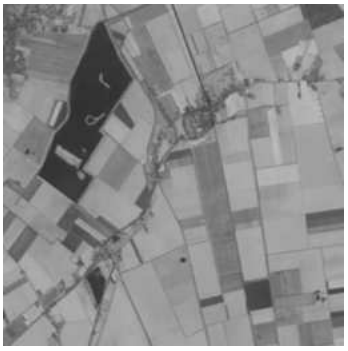


b: Segmented SPOT image, 5 classes

Figure 4: **Test images:** 256×256



a: channel noise corrupted ($\tau = 0.15$)



b: SPOT image

Figure 5: **Test images:** 256×256

Model	N_0	Es. par.	$< N_0 >$	N_1	Es. par.	$< N_1 >$
Potts	8706	0.4981	8734			
Chien	3259	0.82	3261	1462	1.5	1464

Model	N_2	Es. par.	$< N_2 >$			
Potts						
Chien	1101	1.6	1094			

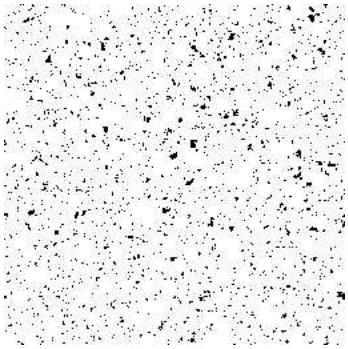
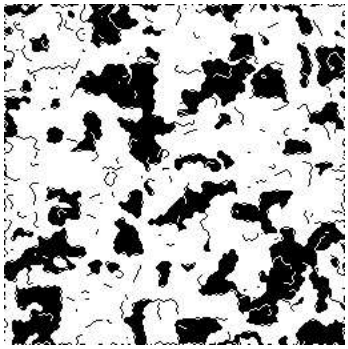
Model	Sample with estimated parameters	Comments
Potts		Inhomogeneous cliques are represented. The simulation is far from the original image. It represents noise in a homogeneous background.
Chien		Edge and line lengths are captured. Surface mean of black areas is greater than in the original image due to the double and triple lines in the original image (bone, birds, horizontal line...)

Table 4: Comparison of Potts model and chien-model as prior for the binary synthetic image

Model	N_0	Es. par.	$< N_0 >$	N_1	Es. par.	$< N_1 >$
Potts	18915	0.6043	18129			
Chien	17856	0.4261	18585	1605	1.0203	1651

Model	N_2	Es. par.	$< N_2 >$			
Potts						
Chien	11175	0.7353	11750			

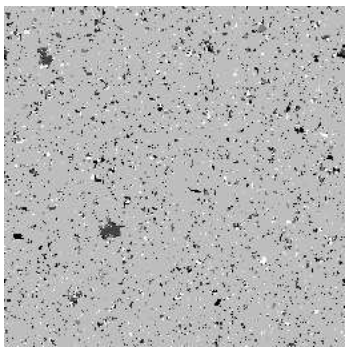
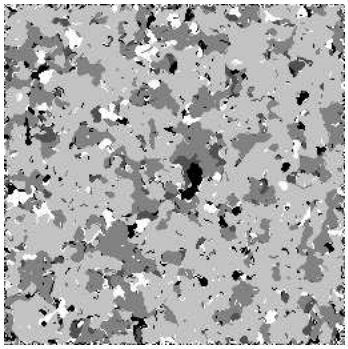
Model	Sample with estimated parameters	Comments
Potts		The Potts model results in a noisy uniform image. The amount of noise is defined by the number of inhomogeneous cliques in the SPOT image.
Chien		The resulting image is composed of homogeneous areas. The edge length is given by the SPOT image. The lines in the SPOT image are represented by little segments in the synthetic realization.

Table 5: Comparison of Potts model and chien-model as prior for a segmented SPOT image

$P(Y|X)$. Using Bayes law, we have:

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)} \propto P(X|Y)P(Y). \tag{30}$$

$P(Y)$ is defined by the prior whereas $P(X|Y)$ represents the data attachment term. As we have considered an uncorrelated channel noise of ratio 0.15, we have:

$$\forall s \in S, \begin{cases} p(x_s = 0|y_s = 0) = 0.85 \\ p(x_s = 1|y_s = 0) = 0.15 \\ p(x_s = 0|y_s = 1) = 0.15 \\ p(x_s = 1|y_s = 1) = 0.85 \end{cases} \tag{31}$$

The global probability $P(X|Y)$ is written as follows:

$$P(X|Y) = \prod_{s \in S} p(x_s|y_s) = \exp \sum_{s \in S} \ln p(x_s|y_s). \tag{32}$$

We consider a prior given by equation (1). We then have:

$$P(Y|X) \propto \frac{1}{Z(\Theta)} \exp \left[-\sum_i \theta_i N_i(Y) + \sum_{s \in S} \ln p(x_s|y_s) \right]. \tag{33}$$

As $P(Y)$ is a Markov Random Field, $N_i(Y)$ is written as a sum of local potentials: $\sum_{c \in \mathcal{C}} V_c(y_s, s \in c)$. We use the Maximiser of the Posterior Marginal (MPM) estimate, found using a Gibbs Sampler [18].

The results obtained are shown in table 6. The restored image obtained with the Potts model is still noisy. It is not surprising when we consider the simulation shown in table 4. However, we can find in the literature better restorations using a Potts model as a prior. In that case the parameter β is increased in order to over-regularize the solution. By using this trick, the noise is erased but details are lost (see figure 6). In some cases, the parameter β is estimated using a Maximum Pseudo-likelihood criteria [19]. Such an estimator tend to over-estimate the parameter [11]. The chien model, however, successfully regularises the segmentation. The noise is removed and for the most part the structures in the image are retained. The model has truly captured the salient characteristics of the original image.

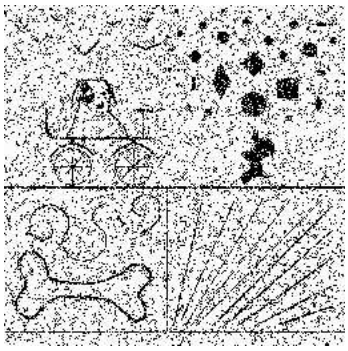
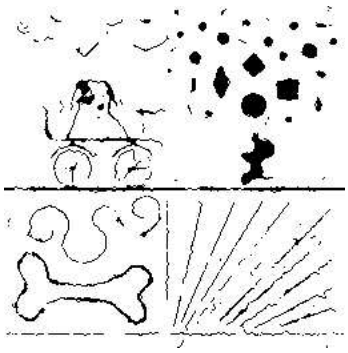
Model	Restored image	Comments
Potts		The result is still noisy. We can not have the right number of homogeneous cliques and obtain a regularized solution. To regularize the result we have to increase the parameter β.
Chien		The chien-model is more adapted to restoration as it controls edge and line lengths. The prior really models some of the image characteristics.

Table 6: Segmentation of a noisy binary image using MCMCML estimators for the prior parameters

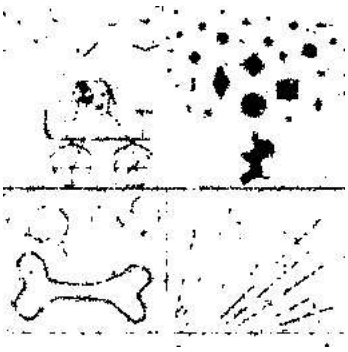


Figure 6: Restoration of figure 5 with a Potts model over-regularizing ($\beta = 1.0$)

We now consider the original SPOT image (see figure 5.b). We compare the behavior of the models being studied when segmenting this image. We suppose that each of the five classes can be represented by a normal law. The means and variances of these classes are estimated using the empirical estimators on the subsets defined on figure 4.b. The conditional probability $P(Y|X)$ is then written as follows:

$$P(Y|X) \propto \frac{1}{Z(\Theta)} \exp\left[-\sum_i \theta_i N_i(Y) + \sum_{s\in S} \sum_{\text{class c}} \left(\frac{(x_s-\mu_c)^2}{2\sigma_c} - \ln(2\pi\sigma_s^2)\right)\delta_{y_s=c}\right] \tag{34}$$

This distribution is sampled and an MPM estimation performed. The resulting segmentations using the Potts model and the chien model are given in table 7. Both segmentations are close to that shown in figure 4. This is because the data (figure 5.b) is very clear and noise free – it is in effect easy to segment; little regularisation is required. The errors in the segmentations occur at the edges of the regions. This partially explains the apparent success in the literature of segmentations performed with the Potts model as prior – in many cases the precise form of the regularisaton is unimportant. As we have seen with the binary image corrupted by channel noise, this is not always true and accurate prior modeling is important in these cases.

5.3 Sampling considerations

In the proposed MCMCML algorithm, more than 99% of the required CPU time consists in sampling the model. Computing the log-likelihood and its derivatives is very fast when we have the samples, indeed, once the samples are in place performing the maximum likelihood estimation takes less than a minute on a Sun-20. This sampling is obtained using a Metropolis-Hasting algorithm. We first have to iterate the algorithm until we reach convergence and then achieve enough iterations to get accurate estimates of the different statistical moments involved in the MCMCML estimation. Among Metropolis-Hasting algorithms, the Gibbs Sampler is the most used in image processing. However, when considering the Potts model the Swendsen-Wang algorithm [20] is more efficient. The Swendsen-Wang algorithm considers clusters instead of pixels. The convergence rate is then faster than a single site updating algorithm. Moreover, as it moves freely within the distribution, we need fewer

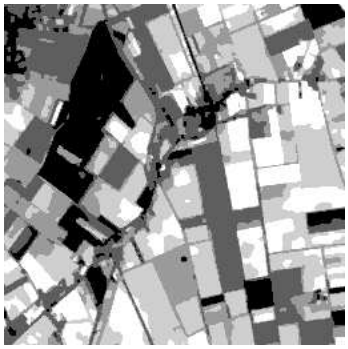
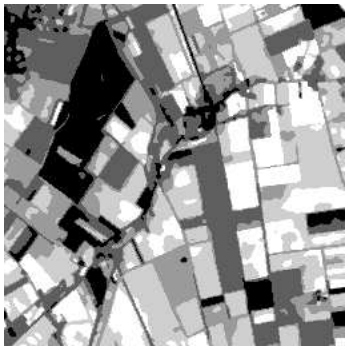
Model	Restored image	Comments
Potts		The segmentation using the Potts model consists mainly of zones. The classification errors are along the region boundaries. The data is sufficiently good that little regularisation is needed.
Chien		The chien-model segmentation is visually very similar. The model is well matched to the data, but the data are sufficiently good that the exact form of the regularisation is unimportant.

Table 7: Segmentation of a SPOT using MCMCML estimators for the prior parameters

samples to obtain accurate estimates of the statistical moments than when using a Gibbs Sampler. Unfortunately, this algorithm can not be applied to the Chien-model. Finding an auxiliary variable to define clusters in this case is still an open issue. In a CPU time point of view, users may then prefer the Potts model.

Notice that, for given parameters, we only ever have to sample the model once. To compute the estimation, we need for each sample the values of the N_i. We can store these values in a data base. The parameter space can be discretized. The discretization step depends on the robustness of the importance sampling. Once we have sample the model for each value of the discrete parameter space the proposed algorithm requires a few seconds on a SUN 20. We can initialize the parameters with the values corresponding to the closest $< N_i >$ in the data base and derive the estimators without further sampling of the model.

6 Conclusion

In this paper, we have used recent development in statistics to propose an algorithm performing maximum likelihood estimation of Markovian prior parameters. Using importance sampling, the proposed algorithm avoids too much sampling which would require huge CPU time. Moreover, a data base can be computed which suppresses sampling. We are currently working on such a data base.

Using the maximum likelihood criterion leads to accurate estimators of the prior parameters. Therefore, we can compare the different priors and the regularizing properties they handle. In this paper, we have considered three Markovian priors: the Potts model and a nonstationary variation of this model, and the Chien-model. The property handled by the two first models consists essentially of homogeneous cliques. The chien model seems more appropriate to image processing as it controls image features (edge length, line length, noise) independently. On the other hand, this model requires more CPU time as it considers higher order interactions. The quality of the data can also influence on a practical level the choice of a priori model.

References

[1] S. Geman, D. Geman. Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images. *IEEE trans. on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.

[2] J. Besag. Spatial interaction and statistical analysis of lattice systems. *Journal of the Royal Statistical Society Series B*, 36:721–741, 1974.

[3] J. Zerubia, R. Chellappa. Mean field approximation using Compound Gauss Markov Random fields for edge detection and image estimation. *IEEE trans. on Neural Networks*, 8(4):703–709, July 1993.

[4] J.K. Goutsias. Mutually compatible Gibbs random fields. *IEEE trans. on Information Theory*, 35:1233–1249, 1989.

[5] J.K. Goutsias. Unilateral approximation of Gibbs random field images. *Computer Vision, Graphics and Image Processing: Graphical Models and Image Processing*, 53:240–257, 1991.

[6] B. Chalmond. Image restoration using an estimated Markov model. *Signal Processing*, 15:115–129, 1988.

[7] A.J. Gray, J.W. Kay, D.M. Titterington. On the estimation of noisy binary Markov Random Fields. *Pattern Recognition*, 25:749–768, 1992.

[8] B. Gidas. *Markov Random Fields (Theory and Application)*, chapter 17, Parameter Estimation for Gibbs Distributions from Fully Observed Data, pages 471–498. Ed. R. Chellappa, A.K. Jain Academic Press Inc., 1993.

[9] C.J. Geyer. Markov Chain Monte Carlo Maximum Likelihood. *School of Statistics University of Minnesota Minneapolis, MN 55455*, 1993.

[10] W.K. Hastings. Monte Carlo sampling methods using Markov Chains and their applications. *Biometrika*, 57:97–109, 1970.

[11] C. J. Geyer, E.E. Thompson. Constrained Monte Carlo Maximum Likelihood for dependent data. *J. R. Statist. Soc. B*, 54(3):657–699, 1992.

[12] Y. Ogata, M. Tanemura. Estimation for interaction potentials of spatial point patterns through the maximum likelihood procedure. *Annals of the Institute of Statistical Mathematics, B*, 33:315–338, 1981.

[13] W. Press, S. Teukolski, W. Vetterling, B. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 2nd ed., 1992.

[14] G. Wolberg, T. Pavlidis. Restoration of binary images using stochastic relaxation with annealing. *Pattern Recognition Letters*, 3:375–388, 1985.

[15] H. Tjelmeland, J. Besag. Markov Random Fields with higher order interactions. submitted to JASA. Preprint.

[16] X. Descombes, J.F. Mangin, E. Pechersky, M. Sigelle. Fine structures preserving model for image processing. In *Proc. 9th SCIA 95 Uppsala, Sweden*, pages 349–356, 1995.

[17] X. Descombes. Application of stochastic techniques in image processing for automatic tissue classification in MRI and blood vessel restoration in MRA. Technical Report KUL/ESAT/MI2/9603, Laboratory for Medical Imaging Research (ESAT-Radiology), K.U.Leuven, Belgium, 1996.

[18] X. Descombes, R. Morris, J Zerubia. Quelques améliorations la segmentation d'images bayesienne. Première partie : modélisation. *Traitement du signal (submitted)*, 1996.

[19] J. Besag. On the statistical analysis of dirty pictures. *J. Roy. Statis. Soc., Series B*, 48:259–302, 1986.

[20] R.H. Swendsen, J.S. Wang. Nonuniversal critical dynamics in Monte Carlo simulations. *Physical Review Letters*, 58(2):86–88, January 1987.

Unité de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unité de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unité de recherche INRIA Rhône-Alpes, 655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN
Unité de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unité de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
ISSN 0249-6399