A Markov Point Process for Road Extraction in Remote Sensed Images

Radu S. Stoica, Xavier Descombes, Josiane Zerubia

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

A Markov Point Process for Road Extraction in Remote Sensed Images

Radu Stoica — Xavier Descombes — Josiane Zerubia

N° 3923

April 2000

THÈME 3

ISSN 0249-6399 ISRN INRIA/RR--3923--FR+ENG

INRIA
SOPHIA ANTIPOLIS

A Markov Point Process for Road Extraction in Remote Sensed Images

Radu Stoica , Xavier Descombes , Josiane Zerubia

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

Rapport de recherche n° 3923 — April 2000 — 38 pages

Abstract: In this paper we propose a new method to extract roads in remote sensed images. Our approach is based on stochastic geometry theory and reversible jump Monte Carlo Markov Chains dynamic. We consider that roads consist of a thin network in the image. We make the hypothesis that such a network can be approximated by a network composed of connected line segments.

We build a marked point process, which is able to simulate and detect thin networks. The segments have to be connected, in order to form a line-network. Aligned segments are favored whereas superposition is penalized. Those constraints are taken in account by the prior model (Candy model), which is an area-interaction point process.The location of the network and the specifities of a road network in the image are given by the likelihood term. This term is based on statistical hypothesis tests.

The proposed probabilistic model yelds a MAP estimator of the road network. In order to avoid local minima, a simulated annealing algorithm, using a reversible jump MCMC dynamic is designed. Results are shown on SPOT, ERS and aerial images.

Key-words: road extraction,stochastic geometry,marked point process, Candy model, hypothesis tests, reversible jump MCMC dynamics

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles, B.P. 93, 06902 Sophia Antipolis Cedex (France)
Téléphone : 04 92 38 77 77 - International : +33 4 92 38 77 77 —Fax : 04 92 38 77 65 - International : +33 4 92 38 77 65

Extraction des Routes dans les Images Aeriennes et Satellitalles utilisant un Processus Ponctuel de Markov

Résumé : Nous proposons une nouvelle méthode pour extraire les routes dans les images satellitales et aériennes. Notre approche est basée sur la géométrie stochastique et les dynamiques MCMC à saut réversible. Nous considérons que le réseau routier est un réseau fin, et que ce réseau peut être approximé par des segments connectés.

Nous construisons un processus ponctuel marqué qui peut simuler et détecter des réseaux fins. La densité de probabilité de ce processus comporte deux termes : le terme d'attache aux données et le terme a priori. Pour former un réseau, les segments doivent être connectés. Nous souhaitons que les segments soient bien alignés et qu'ils ne se superposent pas. Toutes ces contraintes sont prises en compte par le modèle a priori (Candy modèle). L'emplacement du réseau est donné par le terme d'attache aux données. Ce terme est construit à partir des tests d'hypothèses.

Notre modèle probabiliste permet de construire le MAP de l'estimateur du réseau linéique. Pour éviter les minima locaux, nous utilisons un algorithme de type recuit simulé, construit sur une dynamique MCMC à sauts réversibles. Nous montrons des résultats sur des images SPOT, ERS et aériennes.

Mots-clés : extraction de routes, géometrie stochastique, processus ponctuel marqué, Candy modèle, tests d'hypothèses, dynamique MCMC à sauts reversibles

Introduction

A classical approach to extract roads and linear structures in satellite images consists of a two step supervised procedure : the first step consists of detecting pixels which are situated on the roads, using adapted operators : edge detectors [15], morphological operators [16], or as in [14] the combination of two operators (one very specific which gives incomplete results, and another which is more sensitive but gives complete results but also a lot of false alarms). A cost function is usually built using those measures.

The second step consists of using starting points or interest regions in order to find the best pixel candidates belonging to the road . The best road candidates are then found using the cost function in a dynamic programming technique[14, 16, 15, 18].

The F^* algorithm developed in [14] remains a reference till today because of its robustness, rapidity and accuracy. In [18] the authors are constructing an energy (cost) function starting from the F^* algorithm. This new function is grouping together all the needed information (contrast, grey-level, curvature). Especially the introduction of the curvature enables a reliable road tracking.

Geman et al. [20] proposed an original approach, based on information theory. Given a starting point on a road and an orientation, the algorithm starts to track the road in a given direction. The candidates to a road are selected using hypothesis tests. The hypothesis tests compare the radiometry of the pixels along the initial direction and the radiometry of the pixels along the direction orthogonal to the initial orientation. The best candidates are chosen by minimizing an entropy function.

Active contours techniques have also been used in road detection. The roads are considered as ribbons with parallel borders. The curvature and the gradient are used to define an energy function. The initial contour is fitting the road or the building following a differential equation whose solution corresponds to a minimum of the energy [21]. In [22], the authors introduce control points to survey the evolution of the contour along the road.

Bogess in [23] is training a neural network to detect roads in LANDSAT images. The author is obtaining a lot of disconnected segments, and a lot of false alarms.

All these approaches are supervised pixel-based techniques, the user has to give control points of the roads. These methods are quite sensitive to noise and local minima. All the cited authors are making the following assumptions in road and linear network extraction :

· the radiometry inside the road is locally homogeneous,
· there is a significant contrast between the road and the background,
· its width is varying slowly,

· its curvature is regular (*i.e.* a road may be considered locally rectilinear),

The main unsolved difficulties in performing this task are :

· the noise in the image,
· interruptions in the road network caused by trees, shadows, cars, ...
· false alarms generated by the presence of buildings and the similarity of the radiometry of the urban region or some crop fields.

A more elaborate model of the road network geometry seems to be able to improve the detection. The following papers include such an information in the model.

Barzohar et al. [19] are building the first completely automated method in extracting roads on aerial images. The authors are modelling the geometry of the road and the contrast between the road and the background, using AR Gaussian models. Under a Gibbs hypothesis, they compute the MAP estimate of the road in small images, using dynamic programming. Dynamic programming is used once again to give the final result, by choosing the best small images containing roads. This method only detects thick roads. The final results is depending on the choice of small images containing roads. It is possible to connect wrong small images or to stop the algorithm before detecting the entire road.

In [17], the authors use a two step procedure for detecting roads. First, the main segments in the image are detected . Second, a Markov random field is built on a graph composed of the detected segments plus some admissible segments connecting the previous ones. A binary random variable assesses the cost, depending on whether the segments defining the graph represents roads or not.This model gives the probability that two segments are connected. An energy function is derived, and the roads are extracted by minimising this energy function using a simulated annealing algorithm. However, the drawback of this approach is that the road network has to be included in the graph. Therefore, the problem of defining the graph is crucial. This is the first approach which is object oriented. It seems interesting to be able to test new segments or to move their localisation and orientation during the optimisation process.

Such a property can be handled using *marked point processes* [8]. In the proposed method we suppose that the road network is built from small line segments. We model the segments by a marked point process (*Candy Model*). The basic idea is that, we randomly throw segments in the image. The segments move until they fit the road network. The movements are driven by a reversible jump Monte Carlo dynamic.

In the first two sections, we will recall some fondemental notions on point process and Monte Carlo Markov Chain **MCMC** dynamics. Then we present the Candy Model, its probability density and its stability properties. Next, we explain how we have built the

MCMC dynamics we use to simulate the Candy Model. Results are presented on different kinds of remote sensed images. Finally, we draw some conclusions and we point out a few possible directions for future research.

1 Point processes

Let $(T, \mathcal{B}, \nu)$ be a measure space, with $\mathcal{B}$ a Borel algebra and ν a Lebesgue measure, $0 < \nu(T) < \infty$. Let $(\Omega, \mathcal{F}, \mu)$ be the measure space where Ω is the set of point configurations such that :

$$\begin{array}{l} \Omega = \cup_{n=0}^{\infty} \phi_n, \\ \phi_n = \{x_1 \dots x_n\} \subset T \end{array} \tag{1}$$

$\mathcal{F}$ is the associated $\sigma-$algebra and, for $F \in \mathcal{F}$, we have :

$$\mu(F) = \exp(-\nu(T)) \left[1(\emptyset \in F) + \sum \frac{1}{n!} \int \int ...1(\{x_1, ..., x_n\} \in F) \nu(dx_1) ... \nu(dx_n) \right]. \tag{2}$$

which is a Poisson process with intensity measure ν on T. $1(\cdot)$ is the indicator function.

The set Ω contains all $\phi \subset T$ which satisfy :
(i) ϕ is locally finite (every bounded set contains a finite number of points)
(ii) ϕ is simple ($x_i \neq x_j$ if $i \neq j$).
$\mathcal{F}$ is the smallest algebra making the function $\phi : \mathcal{B} \rightarrow \mathcal{F}$ measurable.

A point process Φ on T is a measurable correspondence from a probability space into the measure space $(\Omega, \mathcal{F})$. From an intuitive point of view, a point process is a random choice of ϕ in Ω.

Hereafter, we consider point processes with interacting points. The general definition of the probability density of a pairwise interaction point process is :

$$f(x) = \alpha \beta^n \prod_{\{x_i, x_j\} \subset \phi, x_i \neq x_j} g(x_i, x_j) \tag{3}$$

where α is the normalization constant, g is the interaction function, n the number of points and β a positive constant. The g function requires special conditions for the probability function to be integrable.

1.1 Marked point process

A marked point process on T is a random sequence $\Psi = \{[x_n; m(x_n)]\}$ where the x_n are non-marked points on T and $m(x_n)$ is the label corresponding to each point x_n. Let M be

the space of labels and $\mathcal{M}$ the associated Borel $\sigma-$algebra.

In our case, x is the center of a segment, and $m(x)$ is its label. The label is a vector containing the orientation, the width and the length of the considered segment.

We may consider a marked point process as a non-marked point process defined on $T \times M$. For the Borel sets $B \subset \mathcal{B}$ and $L \subset \mathcal{M}$, the number of points which lies in B with labels in L is $\Psi(B \times L)$.

When the label vector defines the parameters of an object, such process is sometime called object process.

1.2 Markov point process

Let $\diamond$ be a symmetric relation on T. Two points $\zeta, \eta \in T$ are neighbors if $\zeta \diamond \eta$. The neighborhood of a set $A \subset T$ is given by :

$$V(A) = \{\zeta \in T : \zeta \diamond \eta, \eta \in A, \zeta \notin A\} \tag{4}$$

A set $x \in \Phi_A = \{x_1 \dots x_n\}$ is a clique, if all points of x are neighbors.

The function $f : \Omega \to [0, \infty)$ is a Markov function with respect to the relation $\diamond$, if for all $x \in \Omega$:

(a) $f(x) > 0 \Rightarrow f(y) > 0, \quad \forall y \subset x$ (hereditary);
(b) if $f(x) > 0$ and $\zeta \in T$:

$$\lambda(x, \zeta) = \frac{f(x \cup \zeta)}{f(x)} \tag{5}$$

is depending only on ζ and $V(\zeta) \cap x$. $\lambda(x, \zeta)$ is called the Papangelou conditional intensity.

A Markov point process is a finite point process whose density is a Markov function.

A function $g : \Omega \to [0, \infty)$ is called an interaction function if $g(x) \neq 1$ implies that x is a clique.

The function $f : \Omega \to [0, \infty)$ is a Markov function if and only if there is an interaction function g such as :

$$f(x) = \prod_{y \subset x} g(y) \tag{6}$$

for all $x \in \Omega$.

1.3 Gibbs point processes

Let Φ be a Gibbs point process with a random number of points in B. Its density of probability is given by :

$$f(x) = c\beta^{n(x)} \exp(-U(x)) \tag{7}$$

c being the normalization constant, β the intensity of the point process, and $n(x)$ the number of points in B.

The energy can be written as the sum of potentials of interaction :

$$U(x) = U(x_1, \ldots, x_n) = \sum \sum_{1 \leq i < j \leq n} \theta(x_i, x_j) \tag{8}$$

$\theta : T^n \rightarrow \mathbb{R}$ being the pairwise interaction potentials.

Using this notation, the density of a Gibbs point process becomes :

$$f(\phi) = c\beta^{n(x)} \exp(- \sum_{1 < i < j < n(x))} \sum \theta(x_i, x_j)) \tag{9}$$

1.4 Stability of point processes

Geyer in [2] gives an important condition for the stability of point processes. This condition is useful in establishing proofs of convergence in the simulation of Markov chains that have the distribution of a point process as the equilibrium distribution.

CONDITION : *A point process with an unnormalised density with respect to the measure μ is stable if there exists $M \in \mathbb{R}$ such as :*

$$f(x \cup \zeta) \leq M f(x), \forall x = \{x_1 ... x_n\} \subset T, \zeta \in T \tag{10}$$

2 Simulation of Point Process

Usually to simulate point processes two techniques are used : the birth and death algorithms, and Reversible-Jump-Monte-Carlo-Markov-Chains (**RJMCMC**) techniques. Imberty et al. [24] showed the superiority of RJMCMC techniques against birth and death algorithms. Therefore we will first recall in this section the Metropolis-Hastings dynamics, and then present the **RJMCMC** dynamics.

2.1 Monte Carlo Markov Chains

The Monte Carlo Markov Chain (**MCMC**) techniques allow to simulate Markov chains with an equilibrium distribution which is an unnormalized density of probability. In this paragraph we briefly present two techniques of sampling from an equilibrium distribution. First of all, we recall the standard Metropolis-Hastings method. Then, we present the Metropolis-Hastings-Green method, which enables to sample the distribution on a continuum state space (for more details see : [2, 3, 4]).Here we will adopt the Geyer's way of presentation [2].

2.2 Metropolis-Hastings

The Metropolis-Hastings (**MH**) update consists in simulating a Markov chain which converges to an unnormalized distribution h with respect to the measure μ. For this purpose, we need an auxiliary (proposal) distribution $q(x \rightarrow \cdot)$.

The proposal distribution has to be measurable with respect to the measure μ, for all x :

$$\int q(x \rightarrow y)\mu(dy) = 1, \tag{11}$$

$q(x \rightarrow y)$ can be evaluated for all x and y (the chain is irreducible) and for all x it is possible to simulate $q(x \rightarrow \cdot)$ with respect to μ.

The Metropolis-Hastings update mechanism for one step can be described as follows :

ALGORITHM :

1. *Simulate* $y \sim q(x \rightarrow \cdot)$
2. *Compute the acceptance ratio :*

$$R = \frac{h(y)q(y \rightarrow x)}{h(x)q(x \rightarrow y)} \tag{12}$$

3. *Accept the new state* y *with the probability* $\min(1, R)$.

If $q(x \rightarrow y) = q(y \rightarrow x)$, then $R = \frac{h(y)}{h(x)}$. This is the Metropolis update, which has the advantage to eliminate the computation of the proposal density in the acceptance ratio.

2.3 Metropolis-Hastings-Green

If we have different state sub-spaces, we may use what Tierney in [5] called a hybrid sampler. The sub-spaces are traversed using different transition kernels. The transition kernels (or

types of move) are chosen randomly. If, for every transition kernel the detailed balance is attained, then the movement across the sub-space is equivalent to a linear combination of transition kernels. This guarantees the detailed balance for the whole Markov chain.

Green gives in [3] a more general formulation of this dynamics. We may simulate Markov chains on continuous state sub-spaces. The state sub-spaces may have different dimensions. Moreover, the probability of choosing the different movements may depend on the current configuration.

This dynamics is called Metropolis-Hastings-Green (**MHG**). But it is also known as the Reversible-Jump-Monte-Carlo-Markov-Chains (**RJMCMC**) dynamics.

In **MHG** algorithms, the unnormalized density h is replaced by an unnormalised measure π on the state space S. The proposal density is replaced by the kernel $Q(x, A)$. We need a symmetric measure ν on $S \times S$ which plays the same role that μ in the **MH** algorithm. This measure must dominate $\pi(dx)Q(x, dy)$ such that there exists a Radon-Nikodym derivative :

$$f(x, y) = \frac{\pi(dx)Q(x, dy)}{\nu(dx, dy)} \tag{13}$$

which replaces $h(x)q(x \rightarrow y)$ in the **MH** sampler. The acceptance ratio becomes :

$$R = \frac{f(y, x)}{f(x, y)} \tag{14}$$

The algorithm is then the following :
ALGORITHM :

1. *Simulate* $y \sim Q(x \rightarrow \cdot)$
2. *Evaluate Green's ratio :*

$$R = \frac{f(y, x)}{f(x, y)} \tag{15}$$

3. *Move to the state* y *with the probability* $\min(1, R)$.

The practical issue of this algorithm is that we may solve problems defined on several state sub-spaces S_i. This operation will be done by constructing movements which permit to move within a sub-space and movements which permit to jump from a sub-space to another one. We have to be able to browse the sub-spaces entirely (from a statistical point of view), in order to find the optimum state of the system. That means that simulated Markov chain has to be irreducible and aperiodic.

Green [3] gave for the first time a general formulation of the method using the state dependent mixing. Let $Q_i(x, A), \quad i \in I$ be a countable set of transition kernels which are

not necessarily stochastic.The transition kernels have to satisfy certain conditions :

a) $Q(x_i, S)$ is known for all i ;
b) $\sum_{i \in I} Q_i(x, S) \leq 1, \forall x \in S$
c) The density :

$$f_i(x, y) = \frac{\pi(dx) Q_i(x, dy)}{\mu_i(dx, dy)} \tag{16}$$

is known for all i and we may evaluate $f_i(x, y)$ for all x and y. A different and symmetric measure μ_i must be used for each i.

d) For all x and i, it is possible to simulate realizations from the normalized proposal density :

$$P_i(x, \cdot) = \frac{Q_i(x, \cdot)}{Q_i(x, S)} \tag{17}$$

ALGORITHM :

1. Choose a transition kernel Q_i, with the probability $p_i(x) = Q_i(x, S)$. With the probability $1 - \sum_{i \in I} p_i(x)$ don't do the remaining steps and stay at x.
2. Simulate $y \sim P_i(x, \cdot)$;
3. Evaluate Green's ratio :

$$R = \frac{f_i(y, x)}{f_i(x, y)} \tag{18}$$

4. Accept y with the probability $\min(1, R)$.
To simulate and optimize the proposed model we define such an algorithm.

3 A stochastic model for line network extraction

We wish to extract road networks using an object based approach. We make the hypothesis that a road network is formed by several connected segments. The principle of our technique is as follows : we are throwing segments in the image randomly. The segments move in the image until they fit the road network.

The segments have a center, an orientation, a width and a length. We make the hypothesis that a line network is the realization of a marked point process. The probabilistic model of this point process has two components. The first component is the prior model (Candy model), which is determined by the interactions between segments : connexion, attraction, rejection, orientation, and the dimension of the line network. The second term, is the data model which gives the location of the different segments within the network.

Hence, the problem of line network detection in an image can be stated as in [13]. Let D be the observed image. The image is digitised on a finite pixel lattice T. The gray level of the pixel is given by d_t, $t \in T$.

The line network, S we want to extract is composed of a finite number of segments s_i :

$$\begin{array}{l} S = \{s_i, i = 1, \ldots, n\} \\ n \in N \end{array} \tag{19}$$

A segment is given by $s_i = (p_i, m_i)$, with $p_i = (x_i, y_i)$ the coordinates of its centre. The segment is also characterised by its width, its length, its orientation. Those characteristics are given by $m_i = (w_i, l_i, \theta_i)$. All these quantities are discrete random variables, except the orientation which is continuous and uniformly distributed over the interval $[0, 2\pi]$.

Considering only the position of the center of segments, we have a simple point process. Adding the width, the length and the orientation of the segments, we obtain a marked point process.

To detect the number of segments, their location and their parameters, we use a Bayesian approach. The *Maximum a Posteriori* (MAP) estimator of the line network is given by :

$$P(S/D) = \frac{P(D/S)P(S)}{P(D)} \tag{20}$$

having only a single observation of the image, we may consider $P(D)$ as constant. The estimator is then :

$$\hat{S} = \arg\max_S \{P(D/S)P(S)\} \tag{21}$$

Considering the defined point process under the framework of Gibbs point process, we may write the probability density as follows:

$$f(S/D) \propto \beta^n \exp(-U(S)) = \beta^n \exp -(U_D(S) + U_P(S)) \tag{22}$$

$U_D(S)$ is the data energy, and $U_P(S)$ is the prior energy.

Under this assumption, the network estimator is obtained by minimizing the energy function U :

$$\hat{S} = \arg\min_S \{U_D(S) + U_P(S) - n \log \beta\} \tag{23}$$

To find the global optimum of this energy function, we will use a simulated annealing technique. This algorithm iteratively simulates the law :

$$f_T(S/D) \approx [f(S/D)]^{\frac{1}{T}} \tag{24}$$

while decreasing slowly the temperature T. When $T \to 0$, the result of the simulations converges in law to the global optimum.

The main difficulty of this problem is that the configuration sub-spaces have different dimensions. The solution is to use a **RJMCMC** dynamics within the simulated annealing algorithm.

3.1 Candy Model : a region interaction model

The parameters of a random segment are independent random variables :

$$\begin{array}{l} (x_n, y_n) \in T \\ w_n \sim \mathcal{U}(\{w_{min}, \ldots, w_{max}\}) \\ l_n \sim \mathcal{U}(\{l_{min}, \ldots, l_{max}\}) \\ \theta_n \sim \mathcal{U}([0, 2\pi]) \end{array} \tag{25}$$

with $\mathcal{U}$ the uniform law, (x_n, y_n) the coordinates of the centre of the segment, w_n, l_n, θ_n are respectively the width, the length and the orientation of the segment. The parameter vector $m_n = [w_n \quad l_n \quad \theta_n]$ is the label of the non-marqued point process derived from (x_n, y_n).

Let S be the line network as we stated before :

$$\begin{array}{l} S = \{s_i, \quad i = 1, \ldots, n\} \\ n \in N \end{array} \tag{26}$$

with s_i the segments, and n the number of the segments.

The unnormalized density of the Candy model is the density of a pairwise marked point process (3):

$$f(S) \propto \beta^n \prod_{s_i \in S} g(i) \prod_{s_i \diamond s_j, i<j} h(s_i, s_j) \tag{27}$$

$s_i \diamond s_j$ means that the segment s_i and s_j are interacting segments.

The line network is composed of connected segments. Therefore, a segment has two extremities to be connected. Their coordinates are $t_s = T(t_x, t_y)$ and $q_s = Q(q_x, q_y)$ (see Figure 1). A segment can have no connection ($s = s^{\emptyset}$), one connected extremity ($s = s^{t||q}$) or both connected extremities ($s = s^{tq}$). There is no maximal number of connections for a segment.

The g function penalizes the connectivity of the line network and its dimension. We express it as follows :

$$g(s_i) = g_1(s_i) \times g_2(s_i) \tag{28}$$

with :

$$g_1(s_i) = exp\left(\frac{h_{s(i)} - h_{max}}{h_{max}}\right) \tag{29}$$

which penalizes the short segments.

The other term is :

$$g_2(s_i) = \begin{cases} g_{21}, & s_i = s_i^{\emptyset} \\ g_{22}, & s_i = s_i^{t||q} \\ g_{23} = 1, & s_i = s_i^{tq} \end{cases} \tag{30}$$

it takes into account the fact that a segment is free, connected at one extremity or at both extremities. We will strongly penalize a free segment. The segments with only one connexion are less penalised. The completely connected segments are not penalized by the g function. In general, we have : $g_{21} \ll g_{22} < 1$.

The h function is penalizing segments which overlaps as well as the segments which are not well aligned. To define interactions between two segments we consider two cases. There is either a rejection interaction between segments or an attraction interaction.

To introduce the rejection interaction, we have to define a rejection region R_r around a segment as in the Figure 1, $R_r(s) = \{C_r[O(x_n, y_n), r_r = l/2\}$.

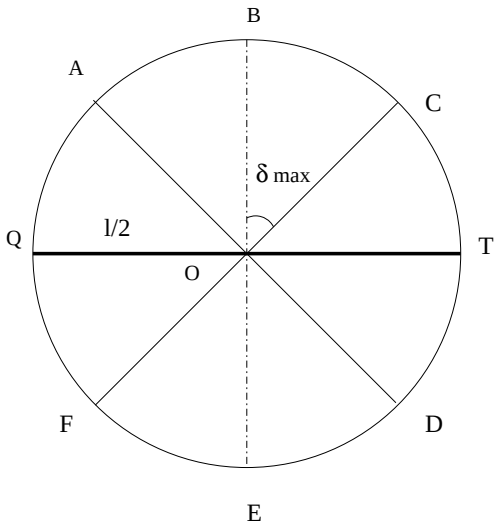


Figure 1: Rejection region for a segment $\overline{QT}$

Two segments s_i and s_j present a *rejection interaction* if :

$$\begin{cases} (x_{s_i}, y_{s_i}) \in R_r(s_j) \quad \text{or} \quad (x_{s_j}, y_{s_j}) \in R_r(s_i) \\ \Xi(s_i, s_j) \end{cases} \tag{31}$$

where $\Xi(s_i, s_j)$ is the condition which enables the crossing between segments. We define δ as follows :

$$\delta = \begin{cases} \frac{\|\theta_{s_j} - \theta_{s_i}\| - \pi/2}{\pi}, & if \|\theta_{s_j} - \theta_{s_i}\| \leq \pi \\ \frac{\|\theta_{s_j} - \theta_{s_i}\| - 3\pi/2}{\pi}, & if \|\theta_{s_j} - \theta_{s_i}\| \leq 2\pi \end{cases} \tag{32}$$

$\theta_{s_i}, \theta_{s_j}$ being the orientations of the segments. The condition $\Xi(s_i, s_j)$ is verified if :

$$\delta > \delta_{max} \tag{33}$$

δ_{max} is the maximal curvature for crossing segments.

We show an example of rejection configuration in the Figure (2) : the segments $\overline{TQ}$ and

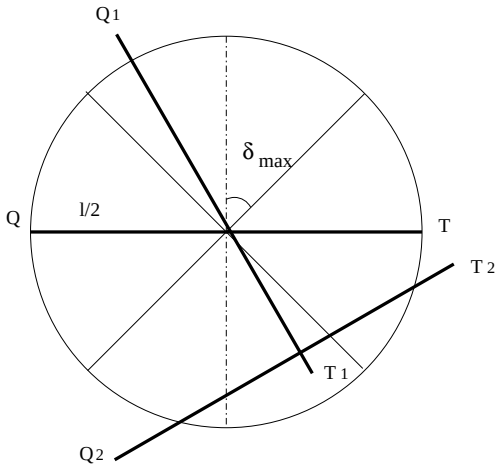


Figure 2: Rejection interaction between segments

$\overline{T_2Q_2}$ exihibit a rejection interaction. The segments $\overline{TQ}$ and $\overline{T_1Q_1}$ satisfy $\delta \leq \delta_{max}$ which means that they are considered as a crossing in the network and as such do not have a rejection interaction. It is the same for the segments $\overline{T_1Q_1}$ and $\overline{T_2Q_2}$.

So, if two segments present a rejection interaction, the function h is :

$$h(s_i, s_j) = h_r \leq 1 \tag{34}$$

In an opposite way, two segments may attract each other. Therefore, a segment has also an attraction region R_a, as in the Figure 3. This region is represented by circles centered at the extremities of the segment : $R_a(s) = \{C_{at}[t(s), r_a = l/4] \cup C_{aq}[q(s), r_a = l/4]\}$.

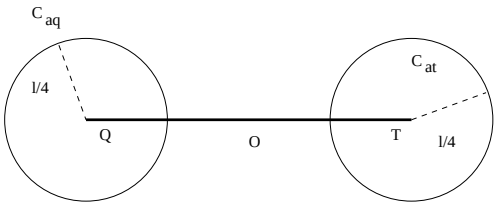


Figure 3: Attraction region for a segment

Two segments s_i and s_j present a *attraction interaction* if all the following conditions are verified :

$$\begin{cases} (x_{s_i}, y_{s_i}) \notin R_r(s_j) \\ (x_{s_j}, y_{s_j}) \notin R_r(s_i) \\ A(s_i, s_j) \end{cases} \tag{35}$$

where $A(s_i, s_j)$ is satisfied if one of the following conditions is true :

$$\begin{array}{llll} t_{s_i} \in R_a(s_j) & \text{and} & q_{s_i} \notin R_a(s_j) \\ q_{s_i} \in R_a(s_j) & \text{and} & t_{s_i} \notin R_a(s_j) \\ t_{s_j} \in R_a(s_i) & \text{and} & q_{s_j} \notin R_a(s_i) \\ q_{s_j} \in R_a(s_i) & \text{and} & t_{s_j} \notin R_a(s_i) \end{array} \tag{36}$$

We show some examples of attraction configuration in the Figure 4 : the segments $\overline{TQ}$ and $\overline{T_1Q_1}$ are attracting segments. The segments $\overline{T_1Q_1}$ and $\overline{T_2Q_2}$ present an attraction interaction while the segments $\overline{TQ}$ and $\overline{T_2Q_2}$ present a rejection interaction.

We want the segments to be aligned. We have to check whether the segments have the same orientation or not. Therefore, we are defining the curvature as :

$$\tau = \begin{cases} \tau_1 = \frac{\|\theta_{s_j} - \theta_{s_i}\|}{\pi}, & \theta_{s_j} \neq \theta_{s_i} \\ \tau_2 = \frac{\theta_c}{\pi}, & \theta_{s_j} = \theta_{s_i} \end{cases} \tag{37}$$

$\theta_{s_j}, \theta_{s_i}$ being the segment orientations. If both segments have the same orientation, we define θ_c as the angle between one of the segments and the line which passes by the centers of the two segments. This definition, with the neighbourhood system, guarantees the reversibility of the Markov chain that simulates the model.

The function h in this case is penalizing the curvature of attracting segments :

$$h(s_i, s_j) = h_a \leq 1 \quad \text{if} \quad \tau > \tau_{max} \tag{38}$$

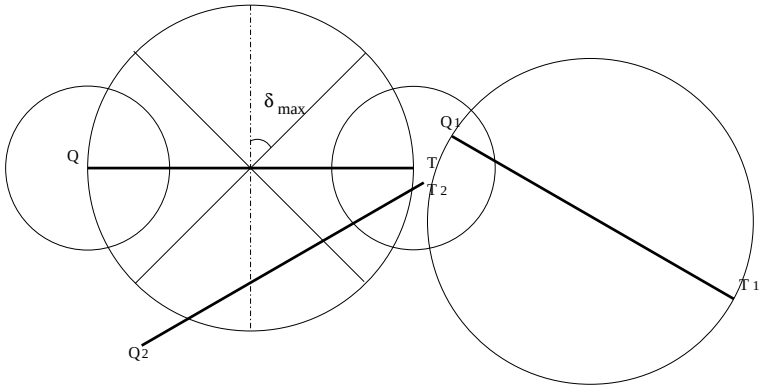


Figure 4: Attraction interacting segments

where τ_{max} is the maximal curvature.

The prior energy is derived by taking the logarithm of Eq. (27). First, we have :

$$U_P(s_i) = -\log g(s_i) - \sum_{s_i \diamond s_j, i \neq j} \log h(s_i, s_j) \tag{39}$$

then, we obtain for the prior energy :

$$U_P(S) = \sum_{s_i \in S} U_P(s_i) = -\sum_{s_i \in S} \log g(s_i) + \sum_{s_i \diamond s_j, i<j} \log h(s_i, s_j) \tag{40}$$

The neighborhood system we have defined is symmetric. This condition assures that the Markov chain defined to simulate the model is aperiodic and irreducible.

3.2 Stability of the Candy Model

PROPOSITION : *The point process described by the density of probability of the Candy model Eq. (27) satisfies the first condition of stability in the sense of Ruelle Eq. (10).*

PROOF : Let $f(S)$ be the density of the Candy model Eq. (27). With respect to the interactions defined in the previous section, we have :

$$h(\zeta, s_j) \leq h_{max} = \max\{h_a, h_r\} < 1, \quad \forall \zeta, s_j \in S \tag{41}$$

This implies :

$$\prod_{\zeta \diamond s_j} h(\zeta, s_j) \leq h_{max}^{n_\zeta} < 1 \tag{42}$$

where n_ζ is the number of segments in the configuration S which interact with the segment ζ.

We also have :

$$g(\zeta) \leq g_{max} = \max\{g_1, g_2\} < 1, \quad \forall \zeta \in S \tag{43}$$

Let $\lambda(\zeta/S)$ be the Papangelou's conditional intensity :

$$\lambda(\zeta/S) = \frac{f(S \cup \zeta)}{f(S)} \tag{44}$$

then, we obtain the condition of stability :

$$\frac{f(S \cup \zeta)}{f(S)} = \beta g(\zeta) \prod_{\zeta \diamond s_j} h(\zeta, s_j) < \beta \tag{45}$$

4 Data Model : conditional energy

We consider that a segment is a small region of a thin network if the statistics of the pixels covered by the segment mask are different from the statistics of the pixels which are situated "left" and "right" to the segment.

Let us define three regions as in Figure 5 :

- D_3^s is the region in the image corresponding to the pixels of the segment we want to analyse (the segment mask); $n_{D_3^s}$ is the corresponding number of pixels ;

- D_3^{sl} is the region in the image corresponding to the pixels of the segment which is at the left of the segment we want to analyse ; $n_{D_3^{sl}}$ is the corresponding number of pixels ;

- D_3^{sr} is the region in the image corresponding to the pixels of the segment which is at the left of the segment we want to analyse ; $n_{D_3^{sl}}$ is the corresponding number of pixels.

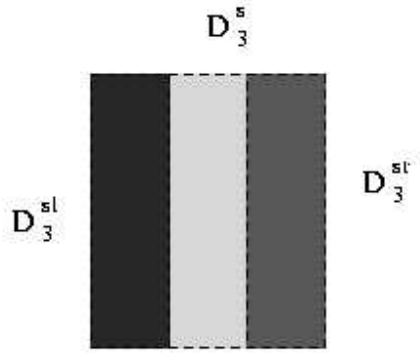


Figure 5: Three region mask

The likelihood function of the hypothesis H_3 : *we have three different regions*, can be written, using a Gaussian hypothesis, as :

$$\begin{aligned} L(H_3) &= \prod_{i=1}^{n_{D_3^s}} \frac{1}{\sqrt{2\pi\sigma^2_{D_3^s}}} \exp\left(-\frac{(x_i^{D_3^s} - \mu_{D_3^s})^2}{2\sigma^2_{D_3^s}}\right) \times \\ &\times \prod_{i=1}^{n_{D_3^{sr}}} \frac{1}{\sqrt{2\pi\sigma^2_{D_3^{sr}}}} \exp\left(-\frac{(x_i^{D_3^{sr}} - \mu_{D_3^{sr}})^2}{2\sigma^2_{D_3^{sr}}}\right) \times \\ &\times \prod_{i=1}^{n_{D_3^{sl}}} \frac{1}{\sqrt{2\pi\sigma^2_{D_3^{sl}}}} \exp\left(-\frac{(x_i^{D_3^{sl}} - \mu_{D_3^{sl}})^2}{2\sigma^2_{D_3^{sl}}}\right) \end{aligned} \tag{46}$$

$\mu_{D_3^s}, \sigma^2_{D_3^s}, \mu_{D_3^{sr}}, \sigma^2_{D_3^{sr}}, \mu_{D_3^{sl}}, \sigma^2_{D_3^{sl}}$, being the estimates of the mean and variance of each segment region.

We can write the log function of Eq. (46) as :

$$\begin{aligned}\log L(H_3) &= -\frac{n_{D_3^s}}{2}\log(2\pi\sigma^2_{D_3^s}) - \sum_{i=1}^{n_{D_3^s}} \frac{(x_i^{D_3^s} - \mu_{D_3^s})^2}{2\sigma^2_{D_3^s}} \\ &- \frac{n_{D_3^{sl}}}{2}\log(2\pi\sigma^2_{D_3^{sl}}) - \sum_{i=1}^{n_{D_3^{sl}}} \frac{(x_i^{D_3^{sl}} - \mu_{D_3^{sl}})^2}{2\sigma^2_{D_3^{sl}}} \\ &- \frac{n_{D_3^{sr}}}{2}\log(2\pi\sigma^2_{D_3^{sr}}) - \sum_{i=1}^{n_{D_3^{sr}}} \frac{(x_i^{D_3^{sr}} - \mu_{D_3^{sr}})^2}{2\sigma^2_{D_3^{sr}}} \end{aligned} \quad (47)$$

then, we have :

$$\begin{aligned}\log L(H_3) &= -\frac{n_{D_3^s} + n_{D_3^{sl}} + n_{D_3^{sr}}}{2} - \frac{n_{D_3^s}}{2}\log(2\pi\sigma^2_{D_3^s}) \\ &- \frac{n_{D_3^{sl}}}{2}\log(2\pi\sigma^2_{D_3^{sl}}) - \frac{n_{D_3^{sr}}}{2}\log(2\pi\sigma^2_{D_3^{sr}}) \end{aligned} \quad (48)$$

Let us define two regions around a segment as in the Figure 6 :

- D_2^{er} is the region in the image corresponding to the pixels at the right of longest symmetry axis of the segment ; $n_{D_2^{er}}$ is the corresponding number of pixels ;

- D_2^{el} is the region in the image corresponding to the pixels at the left of the longest symmetry axis of the segment ; $n_{D_2^{el}}$ is the corresponding number of pixels.

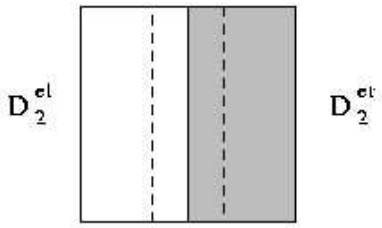


Figure 6: Two region mask

Again with the Gaussian assumption, the likelihood function of the hypothesis H_2 : *existence of an edge*, is :

$$L(H_2) = \prod_{i=1}^{n_{D_2^{er}}} \frac{1}{\sqrt{2\pi\sigma^2_{D_2^{er}}}} \exp\left(-\frac{(x_i^{D_2^{er}} - \mu_{D_2^{er}})^2}{2\sigma^2_{D_2^{er}}}\right) \times \times \prod_{i=1}^{n_{D_2^{el}}} \frac{1}{\sqrt{2\pi\sigma^2_{D_2^{el}}}} \exp\left(-\frac{(x_i^{D_2^{el}} - \mu_{D_2^{el}})^2}{2\sigma^2_{D_2^{el}}}\right) \quad (49)$$

$\mu_{D_2^{er}}, \sigma^2_{D_2^{er}}, \mu_{D_2^{el}}, \sigma^2_{D_2^{el}}$ being the estimates of the mean and the variance of each region.

The log likelihood function is given by :

$$\log L(H_2) = -\frac{n_{D_2^{el}} + n_{D_2^{er}}}{2} - \frac{n_{D_2^{el}}}{2}\log(2\pi\sigma^2_{D_2^{el}}) - \frac{n_{D_2^{er}}}{2}\log(2\pi\sigma^2_{D_2^{er}}) \quad (50)$$

Let us define an homogeneous region around the segment we analyse, see Figure 7 :

- D_1^h is the image patch corresponding to the pixels of the homogeneous region around the segment ; $n_{D_1^h}$ is its number of pixels.

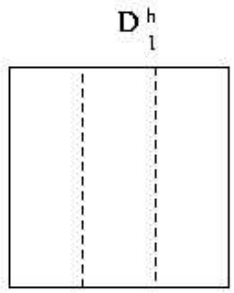


Figure 7: One region mask

The log likelihood function for the test hypothesis H_1 : *the segment is in the middle of an homogeneous region*, can be written as follows :

$$\log L(H_1) = -\frac{n_{D_1^h}}{2} - \frac{n_{D_1^h}}{2}\log(2\pi\sigma^2_{D_1^h}) \quad (51)$$

We define the data energy for a segment as :

$$U_D(s) = \min\{\log L(H_3) - \log L(H_1), \log L(H_3) - \log L(H_1)\} \quad (52)$$

in order to guarantee the stability of the proposed model, we have to bound the energy of the segment.

The data energy for the entire configuration of segments is :

$$U_D(S) = -\sum_{s_i \in S} U(D/s_i) \tag{53}$$

The total energy to minimize is given by the equations (40) and (53) :

$$U(S/D) = U_D(S) + U_P(S) = -\sum_{s_i \in S} U_D(s_i) + \log g(s_i) + \sum_{s_i \diamond s_j, i<j} \log h(s_i, s_j) \tag{54}$$

We want to use this model to extract road networks on satellite images. Basically on optical images, the roads are looking like ridges. On the radar images we have tested, the roads are looking like valleys. We may enforce the data energy for a segment with the term :

$$U_{ridge}(s) = \min\{(\mu_{D_3^s} - \mu_{D_3^{sl}})(\mu_{D_3^s} - \mu_{D_3^{sr}})\} \tag{55}$$

or with the term :

$$U_{valey}(s) = \min\{(\mu_{D_3^{sl}} - \mu_{D_3^s})(\mu_{D_3^{sr}} - \mu_{D_3^s})\} \tag{56}$$

In Figure 8 we show the performance of the data energy on a patch of SPOT satellite image.

5 Construction of a MHG scheme

We want to sample the distribution of the proposed model. To do so, we use the **MHG** algorithm. Therefore, we need to construct the movements between the different sub-spaces *i.e.* the transition kernels. Every movement must have an opposite movement, for example : to the birth of a free segment corresponds the death of a free segment. This guarantees the reversibility of the chain.

Another important problem is the selection of movements. This problem is the equivalent to the kernel mixing. Green [3] proved the convergence of the algorithms using the state dependent mixing. For the safe of simplicity, we use the stochastic mixing of kernels. This mechanism of mixing is independent of the state of configuration, being equivalent to a linear combination of kernels. This type of mixing guarantees the convergence of the algorithm, but it is not optimal. To accelerate the convergence, we increase the probability to choose the movements that are "helping" the model.

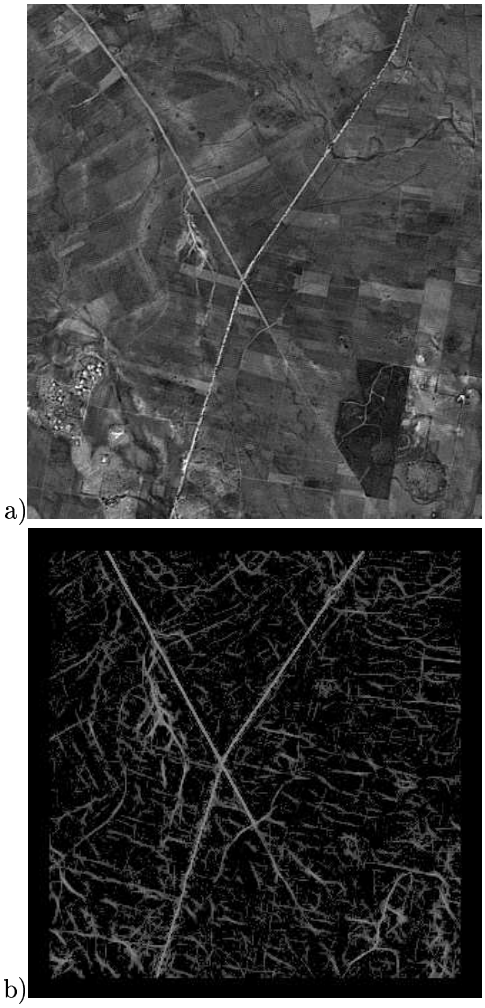


Figure 8: Statistical test result : a) original patch of a SPOT satellite image b) the energy of the proposed statistical test

Let us define for a segment s, the number of connections at each extremity c_t and c_q respectively. A *free* segment $s^{\emptyset}$ has $c_t = c_q = 0$. A segment $s^{t||q}$ is a segment with a *single* connection if one of the following condition is verified :

$$\begin{cases} c_t > 1 \quad \text{and} \quad c_q = 0 \\ c_q > 1 \quad \text{and} \quad c_t = 0 \end{cases} \tag{57}$$

we say that a segment has a *simple* connection if one of the conditions below is accomplished :

$$\begin{cases} c_t = 1 \quad \text{and} \quad c_q = 0 \\ c_q = 1 \quad \text{and} \quad c_t = 0 \end{cases} \tag{58}$$

A segment s^{tq} with both extremity connected has $c_t > 1$ and $c_q > 1$. We say that it has a *double* connection. A segment with a *double simple* connection has $c_t = 1$ and $c_q = 1$.

A line network S has n_t segments. Among them, n_d are not connected and n_c are connected. The connected segments are divided in $n_c^{(1)}$ single connected and $n_c^{(2)}$ double connected segments :

$$n_t = n_d + n_c = n_d + n_c^{(1)} + n_c^{(2)} \tag{59}$$

We have two kinds of movements. We may add or remove a segment to the initial configuration.Those are the *birth and death* movements which enable the jump between the different sub-spaces of configuration. We can also modify the characteristics of a segment. Those are the *modify* movements which enable the movement within a sub-space of configuration.

Below we give the list of movements we use in our dynamics :

- birth and death of a free segment
- birth and death of single connected segment
- birth and death of a simple connected segment
- birth and death of a double connected segment
- birth and death of double simple connected segment
- modify the orientation of a single or simple connected segment
- modify the length of a single or simple connected segment
- modify the position of a single connected segment
- modify the position of a simple connected segment
- modify a simple connected segment to a free segment
- modify a free segment to a simple connected segment
- modify a single connected segment to a free segment
- modify a free segment to a single connected segment

In the following paragraphs, we explain how we compute the acceptance ratio for some of the enumerate movements. For the movements which are not presented, the computation

of the acceptance ratio is very similar to those we introduce in the next paragraph.

5.1 Birth and death of a free segment

This movement consists of adding a free segment ζ to the configuration S. The segment ζ will have a random position. The new state is : $S' = S \cup \zeta$. The death movement is the opposite of birth, *i.e.* deleting a free segment

Let be P_n the probability to choose the birth of a free segment and P_m the probability to choose to delete one.

For the initial state S, we have :

$$n_t = n_c + n_d \tag{60}$$

If we add a free segment, we get for the final state S' :

$$n'_t = n_t + 1 = n_c + n'_d = n_c + n_d + 1 \tag{61}$$

The acceptance ratio is :

$$R = \frac{f(S')Q(S' \to S)}{f(S)Q(S \to S')} \tag{62}$$

with :

$$Q(S' \to S) = P_m \times \frac{1}{n_d + 1} \tag{63}$$

and :

$$Q(S \to S') = P_n \times \frac{1}{\nu(T)} \times \frac{1}{l_{max} - l_{min}} \times \frac{1}{w_{max} - w_{min}} \times \frac{1}{2\pi} \tag{64}$$

We are obtaining :

$$R = \frac{P_m}{P_n} \times \frac{2\pi\nu(T)(l_{max} - l_{min})(w_{max} - w_{min})}{n_d + 1} \times \frac{f(S \cup \zeta)}{f(S)} \tag{65}$$

The acceptance ratio for the death of a free segment is computed by inverting the ratio for the birth movement :

$$R = \frac{P_n}{P_m} \times \frac{n_d}{2\pi\nu(T)(l_{max} - l_{min})(w_{max} - w_{min})} \times \frac{f(S \setminus \zeta)}{f(S)} \tag{66}$$

5.2 Birth and death of a segment with a single connection

This movement adds a single connected segment to the initial configuration.Let P_{nc} et P_{mc} be the probabilities of choosing to add or to remove a single connexion segment.

We have :

$$Q(S' \rightarrow S) = P_{mc} \times \frac{1}{n_c^{(1)\prime}} \tag{67}$$

and :

$$Q(S \rightarrow S') = P_{nc} \times \frac{1}{2 \times n_t} \times \frac{1}{2\pi(l_{max} - l_{min})(w_{max} - w_{min})} \tag{68}$$

$n_c^{(1)\prime}$ is the number of single connected segments in the configuration S' and n_t is the total number of segments in the initial configuration S.

The acceptance ratio for the birth of a single connected segment is :

$$R = \frac{f(S')}{f(S)} \times \frac{P_{mc}}{P_{nc}} \times \frac{2 \times n_t \times 2\pi(l_{max} - l_{min})(w_{max} - w_{min})}{n_c^{(1)\prime}} \tag{69}$$

and by reversing Eq. (69), we get the acceptance for the death of a single connected segment :

$$R = \frac{f(S')}{f(S)} \times \frac{P_{nc}}{P_{mc}} \times \frac{n_c^{(1)}}{2 \times n_t' \times 2\pi(l_{max} - l_{min})(w_{max} - w_{min})} \tag{70}$$

5.3 Birth and death of a segment with a simple connection

Let P_{ns} and P_{ms} be the probabilities to choose a birth or a death respectively, of a segment which has a simple connection.

This movement adds a segment with a random orientation to any other segment which has a free extremity.

The proposal probabilities are :

$$Q(S' \rightarrow S) = P_{ms} \times \frac{1}{n_c^{\prime(s)}} \tag{71}$$

and :

$$Q(S \rightarrow S') = P_{ns} \times \frac{1}{n_c^{(1)} + n_d} \times \frac{1}{2\pi(l_{max} - l_{min})(w_{max} - w_{min})} \tag{72}$$

where $n_c^{\prime(s)}$ is the number of segments with a simple connection in the new configuration S'.

The acceptance ratio becomes :

$$R = \frac{f(S')}{f(S)} \times \frac{P_{ms}}{P_{ns}} \times \frac{(n_c^{(1)} + n_d)2\pi(l_{max} - l_{min})(w_{max} - w_{min})}{n_c^{\prime(s)}} \tag{73}$$

Inverting Eq. (73), we obtain the acceptance ratio for the opposite movement :

$$R = \frac{f(S')}{f(S)} \times \frac{P_{ns}}{P_{ms}} \times \frac{n_c^{(s)}}{(n_c^{\prime(1)} + n'_d)2\pi(l_{max} - l_{min})(w_{max} - w_{min})} \tag{74}$$

5.4 Birth and death of a segment with a double simple connection

Let P_{ns}^d and P_{ms}^d be the probabilities to choose a birth, respectively a death, of a segment which has a double simple connection.

The proposal probabilities are :

$$Q(S' \rightarrow S) = P_{ms}^d \times \frac{1}{n_c^{\prime(sd)}} \tag{75}$$

and :

$$Q(S \rightarrow S') = P_{ns}^d \times \frac{1}{n_c^{(1)} \times (n_c^{(1)} - 1)} \times \frac{1}{2\pi(l_{max} - l_{min})(w_{max} - w_{min})} \tag{76}$$

where $n_c^{\prime(sd)}$ is the number of segments with a double simple connection in the new configuration S'.

The acceptance ratio becomes :

$$R = \frac{f(S')}{f(S)} \times \frac{P_{ms}^d}{P_{ns}^d} \times \frac{n_c^{(1)} \times (n_c^{(1)} - 1) \times 2\pi(l_{max} - l_{min})(w_{max} - w_{min})}{n_c^{\prime(sd)}} \tag{77}$$

Inverting (77) we obtain the acceptance ratio for the opposite movement :

$$R = \frac{f(S')}{f(S)} \times \frac{P_{ns}^d}{P_{ms}^d} \times \frac{n_c^{(sd)}}{n_c^{\prime(1)} \times (n_c^{\prime(1)} - 1) \times 2\pi(l_{max} - l_{min})(w_{max} - w_{min})} \tag{78}$$

5.5 Modify the orientation of a single connected segment

This movement is choosing randomly a single connected segment in the configuration. A new random orientation is given to the segment. This movement allows to overcome the curvature constraints of the model.

The acceptance ratio is :

$$R = \frac{f(S')}{f(S)} \tag{79}$$

The changes in the new configuration are only related to the parameters of the segments. The parameters are uniformly distributed. Therefore : $Q(S \to S') = Q(S' \to S)$.

5.6 Modify the orientation of a single connected segment

The difference between this movement and the previous one is that this time we choose randomly a new length for the segment.

The acceptance ratio is :

$$R = \frac{f(S')}{f(S)} \tag{80}$$

5.7 Modify a free segment into a single connected segment

Let be P_{fc} the probability to choose to connect a free segment and P_{cf} the probability to choose to free a single connected one.

The proposal probabilities are :

$$Q(S' \to S) = P_{cf} \times \frac{1}{n'_c \times \nu(T)} \tag{81}$$

and :

$$Q(S \to S') = P_{fc} \times \frac{1}{n_d \times 2 \times (n_t - 1)} \tag{82}$$

The acceptance ratio is then :

$$R = \frac{f(S')}{f(S)} \times \frac{P_{fc}}{P_{cf}} \times \frac{n_d \times 2 \times (n_t - 1)}{n'_c \times \nu(T)} \tag{83}$$

To obtain the acceptance ratio if we want to set free a single connected segment, we just need to reverse the ratio (83).

6 Results

First of all, we present some simulations of the Candy model. Then, results on satellite (optical and radar) and aerial images are shown.

6.1 Simulation of the Prior Model

In the Figure 9, we are presenting two samples of the Candy model with different densities. As expected, we control the number of segments with the parameter β.

The proof of convergence of an **RJMCMC** dynamics is an open problem [7]. Geyer proved in [2] that for a stable point process we have the Harris-recurrence and the ergodicity of the Markov chain that uses simple birth and death movements to sample the equilibrium distribution. The Candy model is stable, but we still have to prove the good properties of the chain for all the movements we use.

Therefore, we have derived some graphical tests. We looked at the evolution of some statistics simulation. The Figure 10 is showing the evolution of the total number of segments (a), the number of segments with only one connexion (c), the number of segments with two connexions (d) and the number of free segments (b).

We notice that the shape of the curves corresponds to a stable system. Moreover, the statistics corresponds to the penalties we imposed : very few free segments, the number of segments with two connections is significantly greater than the number of segments with only one connection, and the total number of segments is getting stable.

Of course, this heuristic it is not sufficient to prove the convergence of the algorithm. But we may use it like a qualitative measure.

We initialized the proposed algorithm with no segments in the initial configuration.

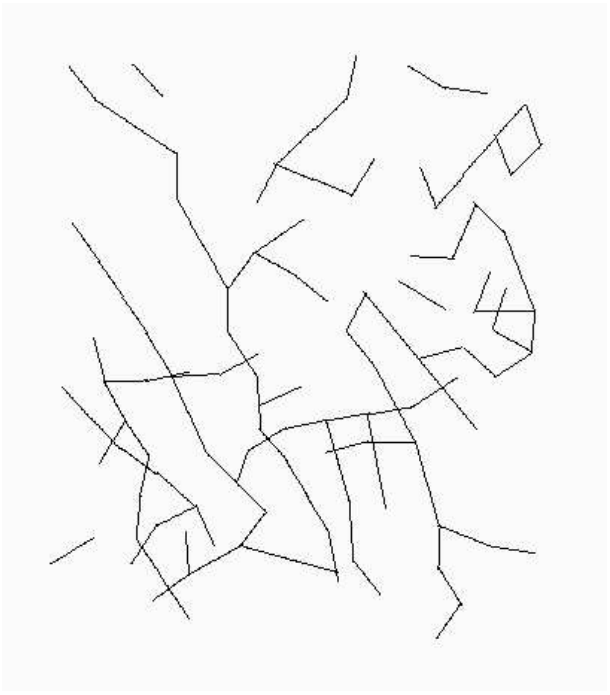
6.2 Real data

We have tested our algorithm on patches of SPOT, ERS and aerial images. We choose no segments as for the initial configuration. The width of the segments is one. We let the **RJMCMC** dynamics run for a fixed number of iterations. We opted for a logarithmic scheme to decrease the temperature of the simulated annealing :

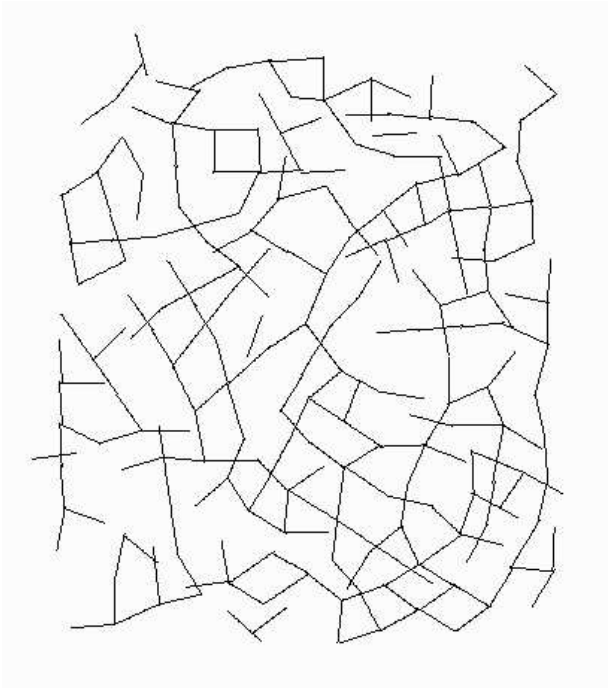
$$T_{n_{iter}} = \frac{T_0}{\log(1 + n_{iter})} \tag{84}$$

with n_{iter} the number of iterations. There is no formal proof for a scheme of decreasing the temperature. We have chosen one of the slower ones.

To catch the basic structure of the network, we need around 15-20 minutes on Sun Sparc station Ultra 10 at 300 MHz. To complete the network, it may take 2-3 hours for an image. Therefore, a good initialization or a deterministic processing after the simulated annealing



a)



b)

Figure 9: Realization of the prior model with different densities : a) $\beta = 0.25$ b) $\beta = 0.5$

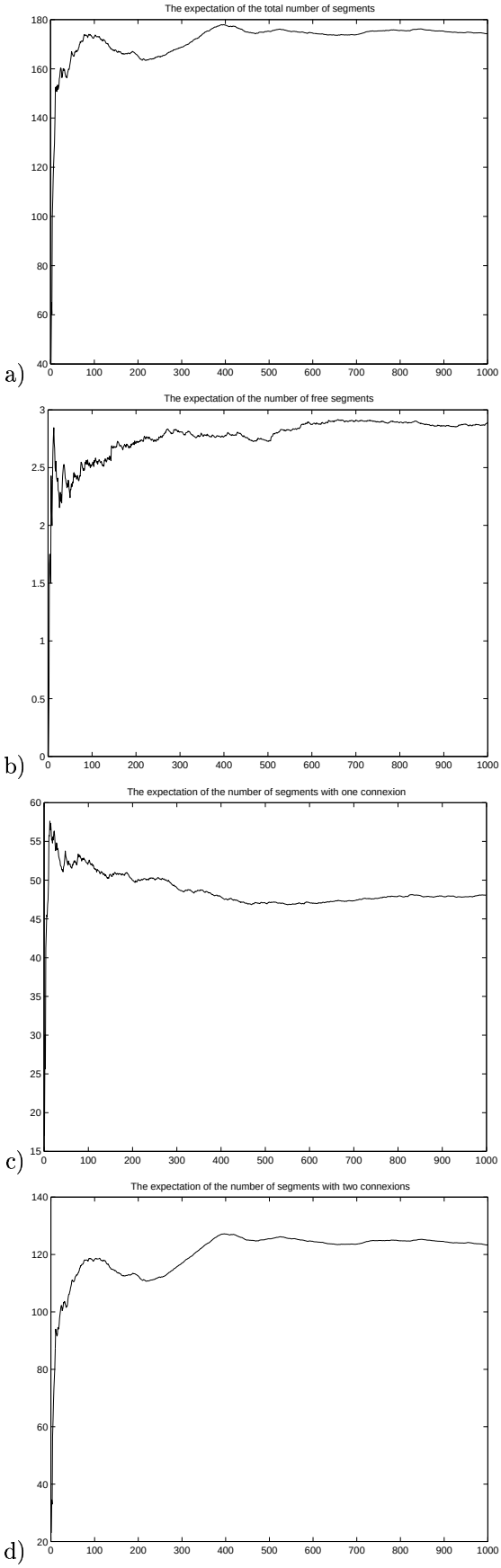
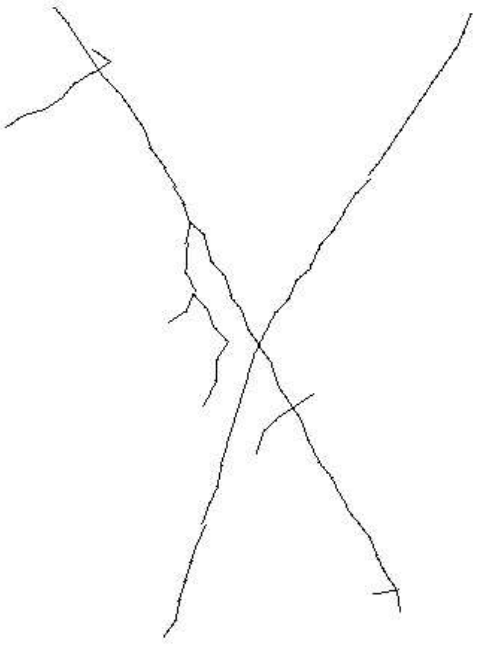


Figure 10: Statistics of the segments

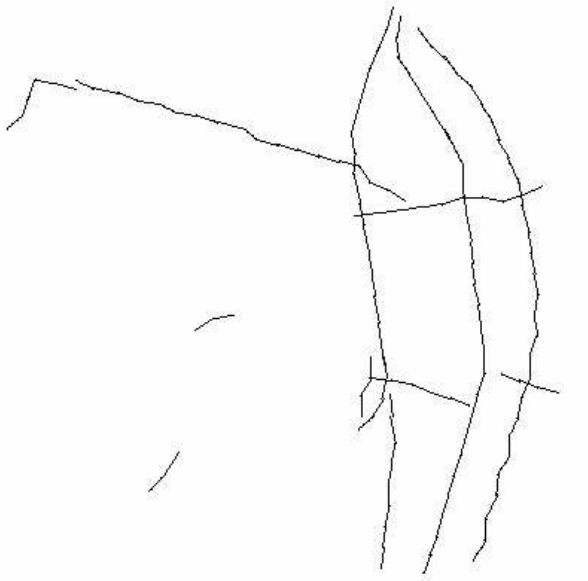
a)



b)

Figure 11: Extracting roads on a SPOT satellite image : a) original image b) the final result

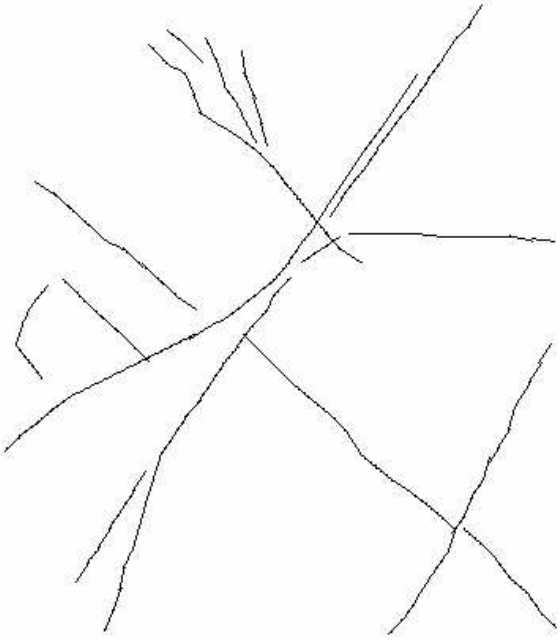
a)



b)

Figure 12: Extracting roads on a SPOT satellite image : a) original image b) the final result

a)



b)

Figure 13: Extracting roads on a SPOT satellite image : a) original image b) the final result

a)

b)

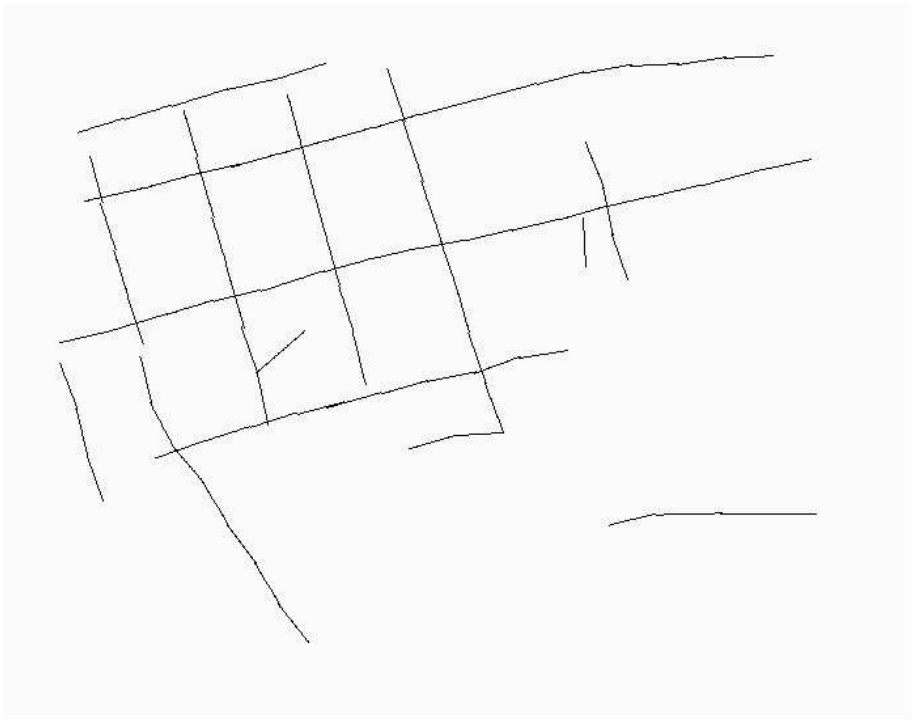


Figure 14: Extracting roads on a ERS radar image : a) original image b) the final result

a)

b)

Figure 15: Extracting roads on an aerial image : a) original image b) the final result

may reduce computational time.

The proposed method is correctly detecting the main roads in the three types of images. We can avoid isolated points and we pass obstacles which are not greater than the maximum dimension of a segment. The prolongation of a road is made in the good direction. To the best of our knowledege these are problems which were not solved in a unsupervised way.

We may have difficulties with the urban areas in satellite images. The easiest solution could be to do a pre-extraction of the urban areas [25], and then to forbid the segments to be placed in this region. A more elegant solution would be to create objects which detect urban regions to interact with objects which detect road networks. Within the framework of the stochastic geometry, this idea is not unrealistic.

7 Conclusion and future work

In this, paper we present a method for thin network extraction in images. The thin networks are supposed to be the realisation of a point process. We simulate this new point process using a **RJMCMC** dynamics.

The next step in our work will consist in studying the estimation of the parameters. We also want, to compute the false alarm rate with respect to road detection.

The terms Eq. (55) and Eq. (56) of the data energy are used under a very strong assumption. An interesting problem to solve is the segmentation of a thin network taking in account the different widths of the different types of roads.

We did not mention the work done by the geographers in this domain. Basically, this scientific community works with large data-bases containing maps and the description of maps. This kind of information could increase the extraction of primitives, such as roads in the images, if this information can be used as an initialization point. Therefore, we think that our method could be used for road registration and map updating.

The interest of such a method is manyfold : it is a completely unsupervised method, it is robust with respect to the noise, and with respect to the sensors of different resolutions. In our point of view, it is more natural to consider the image as a collection of objects, than as a collection of pixels, especially at a high or a very high resolution. The point process framework gives the possibility to create models which manage interactions between objects.

Acknowledgements

The SPOT images have been provided by the French Space Agency (CNES), the ERS image by the European Space Agency (ESA) and the aerial image by the French Mapping Agency (IGN).

The first author thanks the French Ministery of Foreign Affairs for partial financial support.

References

[1] C. J. Geyer, J. Møller, "Simulation and likelihood inference for spatial point process", *Scandinavian Journal of Statistics, Series B*, 21, 359-373, 1994.

[2] C. J. Geyer, "Likelihood Inference for Spatial Point Processes", in O. E. Barndoff-Nielsen, W. S. Kendall and M. N. M van Lieshout, editors, *Stochastic Geometry, Likelihood and Computation*, Chapmann and Hall, London, 1998.

[3] P. Green, "Reversible Jump MCMC Computation and Bayesian Model Determination", *Biometrika* 82 : 711-732, 1995.

[4] P. Green,"MCMC in image analysis". In W. R. Gilks, S. Richardson and D. J. Spiegelhalter, editors, *Markov Chain Monte Carlo in Practice*, pp. 381-399, Chapman and Hall, London, 1996.

[5] L. Tierney, "Markov Chains for Exploring Posterior Distributions", *Tech. Report. No. 560*, University of Minnesota, March, 1994.

[6] S. P. Meyn, R. L. Tweedie, "*Markov Chains and Stochastic Stability*", Springer-Verlag, London, 1993.

[7] C. Robert, G. Casella, "*Monte Carlo Statistical Methods*", Springer-Verlag, 1999.

[8] D. Stoyan, W. S. Kendall, J. Mecke, "*Stochastic Geometry and its Applications*", John Willey and Sons, 1987.

[9] "*Stochastic Models*", Editors : D. P. Heyman, M. J. Sobel, North-Holland, 1990.

[10] B. D. Ripley, "Modelling Spatial Patterns", *Journal of the Royal Statistical Institute, Series B* 39, pp. 172-212, 1997.

[11] B. D. Ripley, F. P. Kelly, "Markov Point Processes", *J. London Math. Soc.*, 15, 188-192, 1977

[12] A. Baddeley, J. Møller, "Nearest-Neighbour Markov Point Process and Random Sets", *International Statistical Review*, pp. 89-121, 57, 2, 1989.

[13] M. N. M. van Lieshout, "Stochastic annealing for nearest-neighbour point process with application to object recognition", CWI Report BS-R9306, 1993.

[14] M. A. Fischler, J. M. Tenenbaum, H. C. Wolf, "Detection of Roads and Linear Structures in Low-Resolution Aerial Imagery Using a Multisource Knowledge Integration Technique ", *Computer Graphics and Image Procesing* vol. 15, pp. 201-223, 1981

[15] C. Graffigne, I. Herlin, "Modélisation de réseaux pour l'Imagerie Satellite SPOT, AFCET, 7eme Congrès de Reconnaissance des Formes et Intelligence Artificielle (Paris), pp. 833-842, 1989.

[16] M.A. Serendero "*Extraction d'Informations Symboliques en Imagerie SPOT : Réseaux de Communication et Agglomerations*", Thèse de Doctorat, Université de Nice-Sophia Antipolis 1989

[17] F. Tupin, H. Maître, J.-F. Mangin, J.-M. Nicolas, et E. Pechersky, "Detection of linear features in SAR images: application to road network extraction" *IEEE Trans. Geosci. and Remote Sensing* vol. 36, no. 2, March 1998.

[18] N. Merlet, J. Zerubia, "New Prospects in Line Detection by Dynamic Programming", *IEEE Trans. on PAMI* vol. 18, pp. 426-431, April 1996.

[19] M. Barzohar, D. B. Cooper, "Automatic Finding of Main Roads in Aerial Images by Using Geometric-Stochastic Models and Estimation", *IEEE Trans. on PAMI* vol. 18, pp. 707-721, July 1996

[20] D. Geman, B. Jedynak, "An Active Testing Model for Tracking Roads in Satellite Images", *IEEE Trans. on PAMI* vol. 18, pp. 1-14, 1996.

[21] P. Fua, Y. G. Leclerc, "Model Driven Edge Detection", *Machine Vision and Applications* 3, pp. 45-56, 1990.

[22] W. M. Neuenschwander, P. Fua, L. Iverson, G. Szekely, O. Kubler, "Ziplock Snakes", *International Journal of Computer Vision* 25(3), pp. 191-201, 1997.

[23] J. E. Bogess "Using Artificial Neural Networks to Indentify Roads in Satellite Images", Proceedings of the World Congress on Neural Networks - San Diego, pp. 410-415, 1994.

[24] M. Imberty, X. Descombes "Simulation de processus objets : Etude de faisabilité pour une application à la segmentation" *INRIA Research Report* No.3881, February 2000.

[25] A. Lorette, X. Descombes, J. Zerubia. "Texture Analysis through a Markovian Modelling and Fuzzy Classification: Application to Urban Area Extraction from Satellite Images" *International Journal of Computer Vision* Vol. 36, no. 3, pp. 219-234, 2000.

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - B.P. 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Lorraine : Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - B.P. 101 - 54602 Villers lès Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38330 Montbonnot St Martin (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - B.P. 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, B.P. 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399