

# A Gibbs point process for road extraction from remotely sensed images

Stoica Radu

*International Journal of Computer Vision*

## Cite this paper

Downloaded from [Academia.edu](#) 

[Get the citation in MLA, APA, or Chicago styles](#)

## Related papers

[Download a PDF Pack](#) of the best related papers 



[Marked point process in image analysis](#)

Josiane Zerubia

[The Gibbs fields approach and related dynamics in image processing](#)

Elena Zhizhina

[Rapid automated detection of roots in minirhizotron images](#)

Christina Wells, Stanley Birchfield



## A Gibbs Point Process for Road Extraction from Remotely Sensed Images

RADU STOICA\*, XAVIER DESCOMBES AND JOSIANE ZERUBIA

*Ariana, Joint Research Group, CNRS/INRIA/UNSA, INRIA, 2004, route des Lucioles, BP93,  
06902 Sophia Antipolis Cedex, France*

Radu.Stoica@inria.fr

Xavier.Descombes@inria.fr

Josiane.Zerubia@inria.fr

*Received June 25, 2001; Revised July 10, 2003; Accepted July 11, 2003*

**Abstract.** In this paper we propose a new method for the extraction of roads from remotely sensed images. Under the assumption that roads form a thin network in the image, we approximate such a network by connected line segments.

To perform this task, we construct a point process 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. These constraints are enforced by the interaction model (called the Candy model). The specific properties of the road network in the image are described by the data term. This term is based on statistical hypothesis tests.

The proposed probabilistic model can be written within a Gibbs point process framework. The estimate for the network is found by minimizing an energy function. In order to avoid local minima, we use a simulated annealing algorithm, based on a Monte Carlo dynamics (RJCMC) for finite point processes. Results are shown on SPOT, ERS and aerial images.

**Keywords:** Gibbs point processes, Candy model, hypothesis tests, reversible jump MCMC dynamics, aerial and satellite images, road extraction

### 1. Introduction

A classical approach to the extraction of roads and linear structures in satellite images is based on a two step supervised procedure. The first step consists of detecting pixels that are situated on roads, using adapted operators: edge detectors (Graffigne and Herlin, 1989), morphological operators (Serendero, 1989), or, as in Fischler et al. (1981), the combination of two operators (one very specific which gives incomplete results, and another one which is more sensitive and gives complete results but also a lot of false alarms). A cost function is usually built using the results.

The second step consists of using starting points or regions of interest to find the best candidate pixels for belonging to a road. The best road candidates are then found by minimizing the cost function using dynamic programming (Fischler et al., 1981; Graffigne and Herlin, 1989; Merlet and Zerubia, 1996; Serendero, 1989).

The  $F^*$  algorithm developed in Fischler et al. (1981) remains a point of reference today because of its robustness, rapidity and accuracy. In Merlet and Zerubia (1996), the authors construct an energy (or cost) function starting from the  $F^*$  algorithm. This new function groups together all the needed information (contrast, grey-level, curvature). The introduction of the local curvature enables reliable road tracking.

\*Currently working as a post-doc at University Jaume I, Campus Riu Sec, E-12071 Castellon, Spain.

Geman and Jedynek (1996) 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 the given direction. Candidates to be part of a road are selected using hypothesis tests. The hypothesis tests compare the radiometry of the pixels along the initial direction and along the direction orthogonal to it. The best candidates are chosen by minimizing an entropy function.

Active contour 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 final contour fits the road or the building following a differential equation whose solution corresponds to a local minimum of the energy (Fua and Leclerc, 1990). In Neuenschwander et al. (1997), the authors introduce control points to survey the evolution of the contour along the road.

Bogess (1994) trains a neural network to detect roads in LANDSAT images. The author obtains a lot of disconnected segments and a lot of false alarms.

All the above approaches are supervised pixel-based techniques: the user has to give control points for the roads. These methods are generally quite sensitive to noise and local minima. All the cited authors make the following assumptions for road and linear network extraction:

- the radiometry within the road is locally homogeneous,
- there is a significant contrast between the road and the background,
- the width of the road is varying slowly,
- its curvature is regular (i.e. a road may be considered as locally rectilinear). The main unsolved difficulties in performing this task are the following:
- the noise present in the image,
- interruptions in the road network caused by trees, shadows, cars, ...
- false alarms generated by the presence of buildings and sometimes similar radiometry of urban regions and agricultural fields.

More elaborate models based on road network geometry have been able to improve detection. The following papers include such information in their models.

Barzohar and Cooper (1996) have built the first completely automatic method for the extraction of roads from aerial images. The authors model 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 obtain the final result, by choosing the best small images containing roads. This method only detects thick roads. The final result depends on the choice of the small images containing roads. Some wrong small images may be connected or the algorithm may stop before detecting the entire road.

In Tupin et al. (1998), the authors use a two-step procedure to detect roads. First, the main segments in the image are extracted. 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 represent roads or not. This model gives the probability that two segments are connected. An energy function is derived and the roads are extracted by minimizing this energy function using a simulated annealing algorithm. 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 was the first approach to be object-oriented. However, it would be interesting to be able to create new segments or to move their location and orientation during the optimization process.

Such changes can be handled using *point processes* (Baddeley and van Lieshout, 1993; van Lieshout, 1994; Rue and Syversveen, 1998; Rue and Hurn, 1999; Stoyan et al., 1987). In the method proposed here, we suppose that the road network is built from small line segments. The segment network is modeled by an interaction point process (called the *Candy Model*). The basic idea is that we randomly throw segments into the image, and randomly move them around, until they fit the road network. The moves are driven by a Reversible Jump Monte Carlo Markov Chain dynamics (RJMCMC) (Green, 1995).

In the first two sections of this paper, the basic ideas of point processes are briefly recalled. Then the Candy model, its probability density and its stability properties are described. Next, an RJMCMC dynamics is built to simulate the Candy model. Results are presented on different kinds of remotely sensed images (with and without ground truth). Finally, some conclusions are drawn and possible directions for future research indicated.

## 2. Point Processes

This section recalls the basic ideas of point processes. For further details we recommend (van Lieshout, 2000).

Let  $(K, \mathcal{B}, \nu)$  be a measure space, with  $K$  a compact subset of  $\mathbb{R}^2$ ,  $\mathcal{B}$  the associated Borel  $\sigma$ -algebra of subsets of  $K$  and  $\nu$  the Lebesgue measure,  $0 < \nu(K) < \infty$ . For  $n \in \mathbb{N}$ , let  $\Omega_n$  be the set of all unordered configurations  $\omega = \{\omega_1, \dots, \omega_n\}$  that consist of  $n$  not necessarily distinct points  $\omega_i \in K$ . We consider the configuration space  $\Omega = \bigcup_{n=0}^{\infty} \Omega_n$  equipped with the  $\sigma$ -algebra  $\mathcal{F}$  generated by the mappings  $\{\omega_1, \dots, \omega_n\} \rightarrow \sum_{i=1}^n \mathbf{1}_{\{\omega_i \in B\}}$  that count the number of points in Borel sets  $B \in \mathcal{B}$ . Hence, a point process on  $K$  is a measurable map from a probability space into  $(\Omega, \mathcal{F})$ .

The ideal reference point process, exhibiting no interaction between the points, is the unit rate Poisson point process. For any  $F \in \mathcal{F}$ , its probability measure:

$$\begin{aligned} \mu(F) &= \exp(-\nu(K)) \\ &\times \left[ \mathbf{1}_{\{\emptyset \in F\}} + \sum_{n=1}^{\infty} \frac{1}{n!} \int_K \dots \int_K \mathbf{1}_{\{\{\omega_1, \dots, \omega_n\} \in F\}} \right. \\ &\left. \times \nu(d\omega_1) \dots \nu(d\omega_n) \right]. \end{aligned} \quad (1)$$

distributes the points in  $K$  according to a Poisson process with intensity  $\nu$ .

*Example 1.* If the set  $F \in \mathcal{F}$  consists of all finite configurations with exactly  $n$  points in  $K$ , then by (1) we get

$$\begin{aligned} \mu(F) &= P(\omega \in F) = \exp(-\nu(K)) \frac{\nu(K)^n}{n!} \\ &= P(N(\omega) = n) \end{aligned}$$

where  $N(\omega)$  is a Poisson random variable with parameter  $\nu(K)$ .

Interactions between points, which are crucial for the problem at hand, may be introduced by specifying a probability density  $f$  with respect to the reference measure  $\mu$ :

$$\begin{aligned} f(\omega) &= \alpha \beta^{n(\omega)} \prod_{\omega_i \in \omega} \phi^{(1)}(\omega_i) \dots \prod_{\{\omega_{i_1}, \dots, \omega_{i_k}\} \in \omega} \\ &\times \phi^{(k)}(\omega_{i_1}, \dots, \omega_{i_k}) \end{aligned} \quad (2)$$

where  $\alpha$  is the normalizing constant,  $\phi^{(j)}$ ,  $j = 1, \dots, k$  are the interaction functions up to a given order  $k$  for  $k$ -tuplets of points,  $n(\omega)$  the number of points and  $\beta$  a positive constant. The  $\phi^{(j)}$  functions require special conditions for the density function to be integrable (van Lieshout, 2000; Stoyan et al., 1987).

We may be interested in attaching different characteristics to the points. In this case we consider a point process on  $K \times M$  as the random sequence  $\Psi = \{(\omega_n, m_n)\}$  where the  $\omega_n \in K$  represent the locations and  $m_n \in M$  is the label corresponding to each  $\omega_n$ .  $M$  is the space of labels and  $\mathcal{M}$  the associated Borel  $\sigma$ -algebra. When the label space  $M$  is equipped with a probability measure  $\nu_M$ , we say that we have a *marked point process* if the distribution of the locations only is a point process on  $K$ .

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

### 2.1. Gibbs Point Processes

The probability density of a Gibbs point process on  $K$  is given by:

$$f(\omega) = c \beta^{n(\omega)} \exp(-U(\omega)) \quad (3)$$

$c$  being the normalizing constant,  $\beta$  the intensity of the point process,  $n(\omega)$  the number of points in the configuration  $\omega$ .  $U(\omega)$  is the energy of the system.

The relation with an interaction point process follows naturally by writing the energy as the sum of interaction potentials:

$$\begin{aligned} U(\omega) &= \sum_{\omega_i \in \omega} \theta^{(1)}(\omega_i) \\ &+ \dots + \sum_{\{\omega_{i_1}, \dots, \omega_{i_k}\} \in \omega} \theta^{(k)}(\omega_{i_1}, \dots, \omega_{i_k}) \end{aligned} \quad (4)$$

$\theta^{(j)} : K^j \rightarrow \mathbb{R}$ ,  $j = 1, \dots, k$  being the interaction potentials.

### 2.2. Stability of Point Processes

Ruelle gives in Ruelle (1969) an important definition for the stability of point processes. From a statistical physics point of view, this condition implies that the energy needed to add a new point to the configuration is bounded. This condition is useful in establishing proofs of convergence in the simulation of Markov chains that

have the distribution of a point process as equilibrium distribution (Geyer, 1998).

*Definition.* A point process on  $K$  with an unnormalized density  $f$  with respect to the measure  $\mu$  is locally stable if there exists  $\Lambda \in \mathbb{R}$  such that:

$$f(\omega \cup \{\zeta\}) \leq \Lambda f(\omega), \quad \forall \omega = \{\omega_1 \cdots \omega_n\} \in \Omega, \quad \zeta \in K \quad (5)$$

### 3. A Stochastic Model for Line Network Extraction

We want to extract road networks using an object based approach. The road network will be approximated by a configuration of connected segments.

A segment is given by  $s_i = (\omega_i, m_i)$ , with  $\omega_i = (x_i, y_i) \in K \subset \mathbb{R}^2$  the coordinates of its centre. The characteristics of a segment  $m_i = (w_i, l_i, \theta_i) \in M$  are its width, its length and its orientation, respectively.  $M$  is the continuous space  $[w_{\min}, w_{\max}] \times [l_{\min}, l_{\max}] \times [0, \pi]$ . In fact the segments represent ribbons in order to take into account road width.

Let  $D$  be the observed image. The image is digitized on a finite pixel lattice  $T$ . The gray level of a pixel is given by  $d_t, t \in T$ . Let  $I(s)$  be the continuous image of a segment given by its geometric figure. The “silhouette” of a segment  $W(s) = I(s) \cap T$  is the intersection between the image of a segment and the finite grid, i.e. the set of pixels covered by  $I(s)$  (Baddeley and van Lieshout, 1993).

Hence, the line network  $\mathbf{s} = \{s_i, i = 1, \dots, n \in \mathbb{N}\}$  we want to extract is considered as the realisation of a point process on  $K \times M$ .

The probability density model of our point process has two components. The first component is the interaction model (Candy model), which is determined by the interactions between segments: connection, attraction, rejection, orientation, and the dimension of the line network. The second term is the data model, which gives the location in the image of the different segments within the network.

Within the framework of a Gibbs point process, the probability density of our model is:

$$f(\mathbf{s}) \propto \beta^n \exp(-U(\mathbf{s})) = \beta^n \exp(-(U_D(\mathbf{s}) + U_I(\mathbf{s}))) \quad (6)$$

where  $U_D(\mathbf{s})$  is the data energy, and  $U_I(\mathbf{s})$  is the interaction energy.

Under this assumption, the estimate of the network is obtained by minimizing the energy function  $U(\mathbf{s})$ :

$$\hat{\mathbf{s}} = \arg \min_{\mathbf{s}} \{U_D(\mathbf{s}) + U_I(\mathbf{s}) - n \log \beta\} \quad (7)$$

The term  $n \log \beta$  is the energy term corresponding to a Poisson point process on  $K \times M$  with intensity  $\beta \nu(K) \pi(l_{\max} - l_{\min})(w_{\max} - w_{\min})$ . Here it may be interpreted as a penalty on the total number of segments in the image.

The global minimum of the energy function  $U(\mathbf{s})$  is found by a simulated annealing technique. This algorithm iteratively simulates the law:

$$f(\mathbf{s}, H) \approx [f(\mathbf{s})]^{\frac{1}{H}} \quad (8)$$

while slowly decreasing the temperature  $H$ . When  $H \rightarrow 0$ , the result of the simulations converges in probability to the global minimum.

The main difficulty of this problem is that the configuration sub-spaces have different dimensions. One solution is to use an RJMCMC dynamics (Green, 1995) within the simulated annealing algorithm.

#### 3.1. Candy Model: A Region Interaction Model

For a segment  $s_i = (p_i, m_i)$ , with  $m_i = (w_i, l_i, \theta_i)$ , we define a prior on its characteristics as:

$$\begin{aligned} w_i &\sim \mathcal{U}([w_{\min}, w_{\max}]) \\ l_i &\sim \mathcal{U}([l_{\min}, l_{\max}]) \\ \theta_i &\sim \mathcal{U}([0, \pi]) \end{aligned} \quad (9)$$

$\mathcal{U}$  being the uniform law.

The unnormalized density of the Candy model is the density of an interaction point process as defined in (2):

$$f_I(\mathbf{s}) \propto \beta^n \prod_{s_i \in S} g(s_i) \prod_{s_i \diamond s_j, i < j} h(s_i, s_j) \quad (10)$$

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

The line network is composed of connected segments. A segment has two extremities to which another segment can be connected. Their coordinates are  $t_s = T(t_x, t_y)$  and  $q_s = Q(q_x, q_y)$  (see Fig. 1). A segment  $s$  is connected to the network  $S$  if at least one of its extremities is closer than  $r_c$  to another segment in

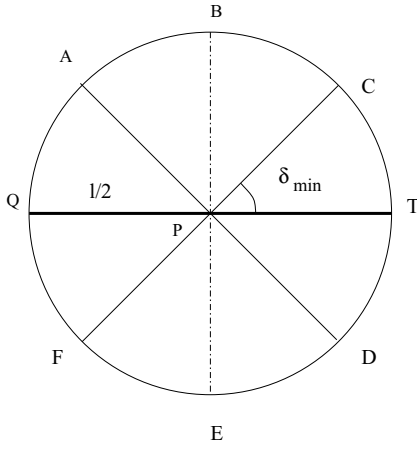


Figure 1. Rejection region for a segment  $\overline{QT}$ .

the network. Hence, every segment in the configuration has two connection regions at its extremities,  $\epsilon_t$  and  $\epsilon_q$ , such that, for all the segments in the configuration, we have  $v(\epsilon_t) = v(\epsilon_q) = v(\epsilon) = \pi r_c^2 > 0$ .

A segment can be in three different states: not connected ( $s = s^\emptyset$ ), with only one extremity connected ( $s = s^{t||q}$ ), or with both extremities connected ( $s = s^{tq}$ ). There is no maximal number of connections for a segment.

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

$$g(s_i) = g_1(s_i) \times g_2(s_i) \quad (11)$$

with:

$$g_1(s_i) = \exp\left(-\frac{l_{\max} - l_i}{l_{\max}}\right) \quad (12)$$

which penalizes short segments.

The other term is:

$$g_2(s_i) = \begin{cases} g_{21}, & \text{if } s_i = s_i^\emptyset \\ g_{22}, & \text{if } s_i = s_i^{t||q} \\ g_{23} = 1, & \text{if } s_i = s_i^{tq} \end{cases} \quad (13)$$

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

The interaction function  $h$  penalizes segments that overlap and neighboring segments that are not well aligned. We consider two types of interaction between the segments: rejection and attraction.

To introduce the rejection interaction, we have to define a rejection region  $R_r$  around a segment  $s = (p, m)$  as in the Fig. 1,  $R_r(s) = C_r[p, r_r = l/2]$  where  $C_r[p, r]$  is the disk of center  $p$  and radius  $r$ .

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

$$\begin{cases} (x_i, y_i) \in R_r(s_j) & \text{or} & (x_j, y_j) \in R_r(s_i) \\ \Xi(s_i, s_j) \end{cases} \quad (14)$$

where  $\Xi(s_i, s_j)$  is the condition given below. First, let us consider  $\delta$  given by:

$$\delta = \begin{cases} \frac{\|\theta_j - \theta_i\| - \pi/2}{\pi}, & \text{if } \|\theta_j - \theta_i\| \leq \pi \\ \frac{\|\theta_j - \theta_i\| - 3\pi/2}{\pi}, & \text{if } \|\theta_j - \theta_i\| \leq 2\pi \end{cases} \quad (15)$$

$\theta_i, \theta_j$  being the orientations of the two segments. The condition  $\Xi(s_i, s_j)$  is verified if  $\delta > \delta_{\max}$ , with  $\delta_{\max}$  a predefined constant, or if the two segments do not cross each other.

In Fig. 2, we give an example of a rejection configuration: the segments  $\overline{TQ}$  and  $\overline{T_2Q_2}$  exhibit a rejection interaction. The segments  $\overline{TQ}$  and  $\overline{T_1Q_1}$  satisfy  $\delta \leq \delta_{\max}$  which means that they are considered as an acceptable crossing in the network and as such do not have a rejection interaction.

If two segments present a rejection interaction, the function  $h$  is:

$$h(s_i, s_j) = h_r \leq 1 \quad (16)$$

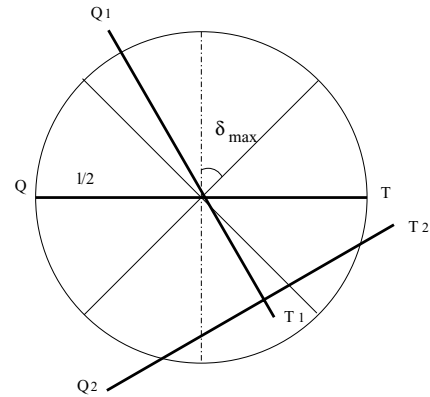


Figure 2. Rejection interaction between segments.

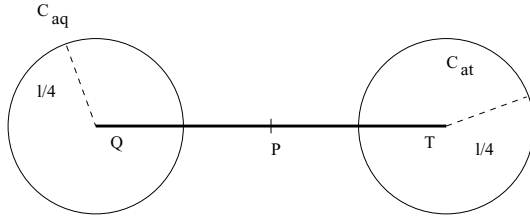


Figure 3. Attraction region for a segment.

Conversely, two segments may attract each other if they are well aligned. This property is modeled by defining an attraction region  $R_a$ , as shown in Fig. 3. This region is represented by disks centered at the extremities of the segment:  $R_a(s) = C_{at}[t_s, r_a = l/4] \cup C_{aq}[q_s, r_a = l/4]$ .

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

$$\begin{cases} (x_i, y_i) \notin R_r(s_j) \\ (x_j, y_j) \notin R_r(s_i) \\ A(s_i, s_j) \end{cases} \quad (17)$$

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

$$\begin{aligned} t_i &\in R_a(s_j) \quad \text{and} \quad q_i \notin R_a(s_j) \\ q_i &\in R_a(s_j) \quad \text{and} \quad t_i \notin R_a(s_j) \\ t_j &\in R_a(s_i) \quad \text{and} \quad q_j \notin R_a(s_i) \\ q_j &\in R_a(s_i) \quad \text{and} \quad t_j \notin R_a(s_i) \end{aligned} \tag{18}$$

We show examples of some attraction configurations in Fig. 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 define the curvature as:

$$\tau = \begin{cases} \tau_1 = \frac{\|\theta_j - \theta_i\|}{\pi}, & \theta_j \neq \theta_i \\ \tau_2 = \frac{\theta_c}{\pi}, & \theta_j = \theta_i \end{cases} \quad (19)$$

$\theta_j, \theta_i$  being the segment orientations. If both segments have the same orientation, we define  $\theta_c$  as the angle between one of the segments and the line which passes through the centers of the two segments. An example is given in Fig. 5.

The function  $h$  in this case penalizes high curvature between attracting segments:

$$h(s_i, s_j) = h_a \leq 1 \quad \text{if } \tau > \tau_{\max} \quad (20)$$

where  $\tau_{\max}$  is the pre-defined maximal acceptable curvature.

The interaction energy is derived by taking the logarithm of Eq. (10). First, we have:

$$U_I(s_i) = -\log g(s_i) - \sum_{s_i \diamond s_j, i \neq j} \log h(s_i, s_j) \quad (21)$$

Then we obtain for the interaction energy:

$$\begin{aligned} U_I(\mathbf{s}) &= \sum_{s_i \in \mathbf{s}} U_I(s_i) \\ &= - \sum_{s_i \in \mathbf{s}} \left[ \log g(s_i) + \sum_{s_i \diamond s_j, i < j} \log h(s_i, s_j) \right] \end{aligned} \quad (22)$$

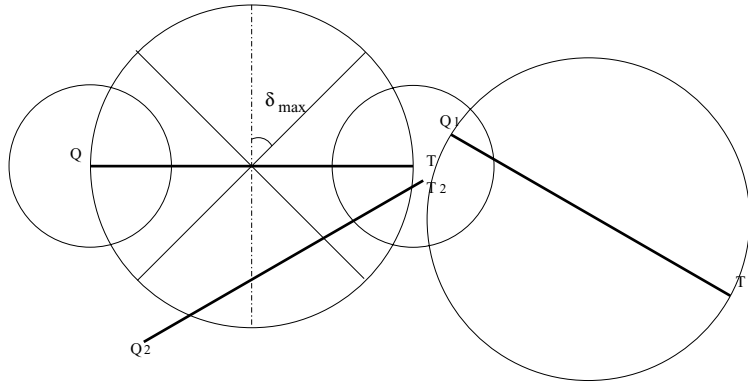


Figure 4. Attraction interaction between segments.

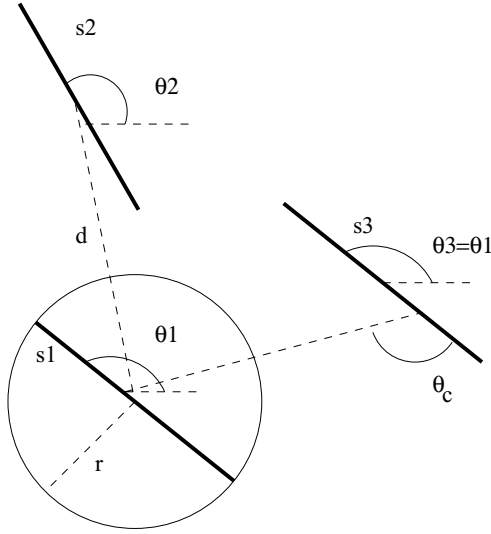


Figure 5. Segments and their orientations.

It is not very difficult to show that if we add a segment to the configuration, there is a maximum number of free or one extremity connected segments that can be connected by the new segment. In this case, we can prove that when the model parameters satisfy  $\beta > 0$  and  $g_{21}, g_{22}, h_r, h_a \in (0, 1)$ , there exists a positive finite  $\Delta$  which gives us the local stability condition (cf. Eq. (5)) for the Candy model (Stoica, 2001).

#### 4. Data Energy

The data term checks whether a segment belongs to the network or not. This task is performed using hypothesis tests. The following situations are analyzed:

- the segment is in the middle of an homogeneous region ( $H_1$ ),
- the segment is situated at the frontier of two homogeneous regions (i.e. an edge) ( $H_2$ ),
- the segment belongs to the network ( $H_3$ ).

To construct the hypothesis test ( $H_3$ ), we consider a segment to be part of the network if the statistics of the pixels covered by the segment silhouette are different from the statistics of the pixels situated to the left and to the right of the segment (Refregier et al., 1997).

Let us define three regions as in Fig. 6(a):

- $D_3^s$  is the region in the image corresponding to the pixels of the segment we want to analyze (the segment “silhouette”),  $n_3^s$  being the corresponding number of pixels;
- $D_3^{sl}$  is the region in the image corresponding to the pixels to the left of the segment we want to analyze,  $n_3^{sl}$  being the corresponding number of pixels;
- $D_3^{sr}$  is the region in the image corresponding to the pixels to the right of the segment we want to analyze,  $n_3^{sr}$  being the corresponding number of pixels.

Under the assumption that the pixels values are Gaussian i.i.d variables in each region, the likelihood function of the hypothesis ( $H_3$ ) can be written as follows:

$$L(H_3) = \prod_{i \in D_3^s} \frac{1}{\sigma_3^s \sqrt{2\pi}} \exp \left[ -\frac{1}{2} \left( \frac{d_i - \mu_3^s}{\sigma_3^s} \right)^2 \right] \times \prod_{i \in D_3^{sr}} \frac{1}{\sigma_3^{sr} \sqrt{2\pi}} \exp \left[ -\frac{1}{2} \left( \frac{d_i - \mu_3^{sr}}{\sigma_3^{sr}} \right)^2 \right] \times \prod_{i \in D_3^{sl}} \frac{1}{\sigma_3^{sl} \sqrt{2\pi}} \exp \left[ -\frac{1}{2} \left( \frac{d_i - \mu_3^{sl}}{\sigma_3^{sl}} \right)^2 \right] \quad (23)$$

$\mu_3^s, \sigma_3^s, \mu_3^{sr}, \sigma_3^{sr}, \mu_3^{sl}, \sigma_3^{sl}$  being the parameter estimates of the mean and the standard deviation of each region.

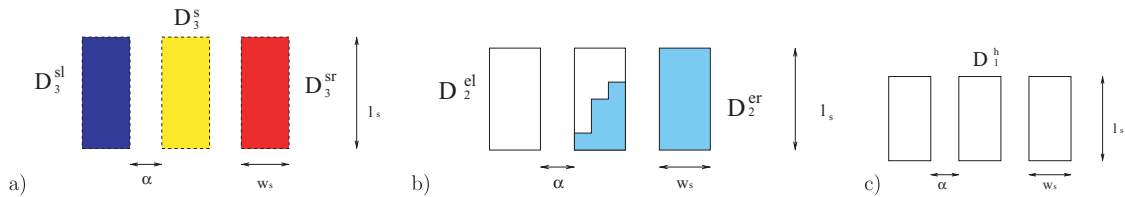


Figure 6. Region mask: (a) three homogeneous regions, (b) two homogeneous regions, (c) one homogeneous region.



We can write the log likelihood as:

$$\begin{aligned} \log L(H_3) = & -n_3^s \log(\sigma_3^s \sqrt{2\pi}) \\ & - \sum_{i \in D_3^s} \frac{1}{2} \left( \frac{d_i - \mu_3^s}{\sigma_3^s} \right)^2 - n_3^{sr} \log(\sigma_3^{sr} \sqrt{2\pi}) \\ & - \sum_{i \in D_3^{sr}} \frac{1}{2} \left( \frac{d_i - \mu_3^{sr}}{\sigma_3^{sr}} \right)^2 - n_3^{sl} \log(\sigma_3^{sl} \sqrt{2\pi}) \\ & - \sum_{i \in D_3^{sl}} \frac{1}{2} \left( \frac{d_i - \mu_3^{sl}}{\sigma_3^{sl}} \right)^2 \end{aligned} \quad (24)$$

Substituting the empirical estimates of the means and variances into the log-likelihood function as in Refregier et al. (1997), we obtain:

$$\begin{aligned} \log L(H_3) = & -\frac{n_3^s + n_3^{sr} + n_3^{sl}}{2} - n_3^s \log(\sigma_3^s \sqrt{2\pi}) \\ & - n_3^{sr} \log(\sigma_3^{sr} \sqrt{2\pi}) - n_3^{sl} \log(\sigma_3^{sl} \sqrt{2\pi}) \end{aligned} \quad (25)$$

For testing the second hypothesis ( $H_2$ ), i.e. that the segment is situated on an edge, let us define two regions around a segment as in the Fig. 6(b):

- $D_2^{er}$  is the region in the image corresponding to the pixels located to the right of the longest symmetry axis of the segment,  $n_2^{er}$  being the corresponding number of pixels;
- $D_2^{el}$  is the region in the image corresponding to the pixels located to the left of the longest symmetry axis of the segment,  $n_2^{el}$  being the corresponding number of pixels.

As in the previous case, under the assumption that the pixel values are Gaussian and i.i.d in each region, substituting the estimated parameters into the corresponding log-likelihood function, we get:

$$\begin{aligned} \log L(H_2) = & -\frac{n_2^{er} + n_2^{el}}{2} - n_2^{er} \log(\sigma_2^{er} \sqrt{2\pi}) \\ & - n_2^{el} \log(\sigma_2^{el} \sqrt{2\pi}) \end{aligned} \quad (26)$$

$\sigma_2^{er}, \sigma_2^{el}$  being the estimates of the standard deviation of each region.

Finally, to verify the hypothesis ( $H_1$ ), i.e. that the segment is in the middle of a homogeneous region, let us define a homogeneous region around the segment we analyze, as in Fig. 6(c):

- $D_1^h$  is the image patch corresponding to the pixels of the homogeneous region;  $n_1^h$  is its number of pixels.

Hence, as in the first two cases, the log likelihood of the hypothesis  $H_1$  can be written as follows:

$$\log L(H_1) = -\frac{n_1^h}{2} - n_1^h \log(\sigma_1^h \sqrt{2\pi}) \quad (27)$$

where  $\sigma_1^h$  is the estimate of the standard deviation of the region.

Let  $\varrho(s)$  be the test function for a segment:

$$\begin{aligned} \varrho(s) = & \min\{\log L(H_3) - \log L(H_1), \log L(H_3) \\ & - \log L(H_2)\} \end{aligned} \quad (28)$$

We use this model to extract line networks from satellite and aerial images. In optical images, the roads usually look like ridges. In the radar images we have tested, the roads generally look like valleys. We may augment the test function for a segment by adding the term:

$$U_{\text{ridge}}(s) = \min\{(\mu_{D_3^s} - \mu_{D_3^{sl}}), (\mu_{D_3^s} - \mu_{D_3^{sr}})\} \quad (29)$$

which increases the probability of detecting “white” networks, or the term:

$$U_{\text{valley}}(s) = \min\{(\mu_{D_3^{sl}} - \mu_{D_3^s}), (\mu_{D_3^{sr}} - \mu_{D_3^s})\} \quad (30)$$

which favours the extraction of “black” networks.

Hence, the data energy for a segment becomes:

$$U_D(s) = \varrho(s) + U_{\text{ridge/valley}}(s) \quad (31)$$

depending on the type of network we want to extract.

The data energy for the entire configuration of segments is then given by:

$$U_D(\mathbf{s}) = - \sum_{s_i \in \mathbf{s}} U_D(s_i) \quad (32)$$

This data term is interpreted as an inhomogeneous external field but not as a likelihood in a Bayesian sense. The likelihood defined as the joint distribution of the intensity data given the segment configuration is not accessible in this context.

Finally, the total energy to minimize including the Candy model is:

$$\begin{aligned}
 U(\mathbf{s}) &= U_D(\mathbf{s}) + U_I(\mathbf{s}) \\
 &= - \sum_{s_i \in \mathbf{s}} \left[ U_D(s_i) + \log g(s_i) + \sum_{s_i \diamond s_j, i < j} \log h(s_i, s_j) \right]
 \end{aligned} \tag{33}$$

## 5. Simulation of Finite Point Processes

One of the most common ways to simulate a point process is to use spatial birth-and-death processes (Preston, 1977). This technique simulates a Markov chain allowing only two types of transitions: the addition of a point to the configuration (birth) and the deletion of a point from the configuration (death). The mixing properties of the chain can be increased by means of a Metropolis-Hastings (MH) dynamics (Geyer, and Møller, 1994; Geyer, 1998; Møller, 1998). This dynamics enables us to add moves that modify the characteristics of the points inside a configuration. The MH dynamics can be considered as a particular case of the generalization of Monte Carlo Markov Chain (MCMC) methods to the case of a continuous state space (Green, 1995). This class of algorithms is known under the name “Reversible Jump Monte Carlo Markov Chains” (RJMCMC), and it has already been applied to several image analysis problems in (Green, 1996; Rue and Syversveen, 1998; Rue and Hurn, 1999).

### 5.1. Reversible Jump Markov Chain Monte Carlo

Suppose the Markov chain is in state  $\mathbf{s}$ . We randomly choose a move of type  $i$ , which transforms  $\mathbf{s}$  to  $\mathbf{s}'$  with probability  $\gamma_i(\mathbf{s} \rightarrow d\mathbf{s}')$ . The new state  $\mathbf{s}'$  is then accepted with probability:

$$\alpha_i(\mathbf{s} \rightarrow \mathbf{s}') = \min \left\{ 1, \frac{f(\mathbf{s}')\gamma_i(\mathbf{s}' \rightarrow d\mathbf{s})}{f(\mathbf{s})\gamma_i(\mathbf{s} \rightarrow d\mathbf{s}')} \right\} \tag{34}$$

Green presented in (Green, 1995) the RJMCMC method for jumping between states in spaces of different dimensions. It is necessary to match the dimension of state  $\mathbf{s}$  to  $\mathbf{s}'$ . This is done by sampling a vector  $u$  of continuous random variables independently of  $\mathbf{s}$ .  $\mathbf{s}'$  is obtained by using an invertible deterministic function  $\varphi(\mathbf{s}, u)$ . The acceptance probability is modified by the

Jacobian of the transformation:

$$\begin{aligned}
 \alpha_i(\mathbf{s} \rightarrow \mathbf{s}') &= \min \left\{ 1, \frac{f(\mathbf{s}')}{f(\mathbf{s})\gamma(u^{(i)})} \left| \frac{\partial \varphi}{\partial(\mathbf{s}, u)} \right| \right. \\
 &\quad \times \left. \frac{p(\text{choosing to propose the reverse jump})}{p(\text{choosing to propose the forwards jump})} \right\}
 \end{aligned} \tag{35}$$

where  $\gamma(u^{(i)})$  is the density of the random vector  $u$ .

If the dimension of the configuration space is fixed, we have the classical Metropolis-Hastings dynamics.

### 5.2. Moves for Simulating the Candy Model

Sampling from the distribution of the proposed model using an RJMCMC dynamics needs a careful design. The convergence properties of the chain when the different specific moves are incorporated in the transition kernel need to be guaranteed. To simulate a reversible chain, every move must have a reverse. For instance: to the birth of a free segment corresponds the death of a free segment. Another important point is the selection of moves. We have decided to make a stochastic mixing of the transition kernels, independent of the current configuration. These precautions, taken together, ensure that the simulated chain will have the distribution of the Candy model as equilibrium distribution.

A line network  $\mathbf{s}$  has  $n_t$  segments. Among them,  $n_d$  are not connected and  $n_c$  are connected. The connected segments are divided into  $n_{c1}$  (for one extremity connected) and  $n_{c2}$  (for doubly-connected) segments:

$$n_t(\mathbf{s}) = n_d(\mathbf{s}) + n_c(\mathbf{s}) = n_d(\mathbf{s}) + n_{c1}(\mathbf{s}) + n_{c2}(\mathbf{s}) \tag{36}$$

Theoretically, to sample the Candy model we only need uniform birth and death moves. The birth move adds a segment to the configuration, with the segment parameters uniformly chosen in the parameter space. The death move can uniformly eliminate a segment from the configuration (Geyer, 1998). This kind of simulation performs well when the interactions between the points are simple, as in the Strauss model for example. The Candy model embeds different interactions, hence it exhibits a much more complex interaction structure. Using only a uniform birth and death transition kernel in order to obtain a connected network of well aligned segments may lead to a very long running time for the algorithm. Therefore we need to build specific moves that help the model by increasing the mixing properties of the simulation dynamics.

To build connected networks, the segments are added to or deleted from the current configuration depending on their state (connected or not). To further increase the mixing properties of the chain, we use moves which *modify* the characteristics of a segment.

Below we give the list of moves we use in the simulations:

- birth and death of a free segment,
- birth and death of a singly-connected segment,
- birth and death of a doubly-connected segment,
- modify the orientation of a singly-connected segment,
- modify the length of a singly-connected segment.

In the following, some examples are given in order to understand better how we compute the acceptance ratio for some of the above mentioned moves. For the moves which are not presented, the computation of the acceptance ratio is very similar to those described below.

*Example 2.* Birth and death of a free segment.

This move consists of adding a free segment  $\zeta$  to the configuration  $\mathbf{s}$ . The segment  $\zeta$  will have a random position. The new state is:  $\mathbf{s}' = \mathbf{s} \cup \{\zeta\}$ . The death move (i.e. deleting a free segment) is the opposite of a birth.

The independent random vector  $u$ :

$$u = [(x_u, y_u), l_u, \theta_u, w_u] \quad (37)$$

is uniformly sampled on the segment parameter space. The invertible transformation is the identity:

$$\varphi(\mathbf{s}, u) = \mathbf{s}' = \{s'_1 = s_1, \dots, s'_n = s_n, \zeta = u\} \quad (38)$$

so the corresponding Jacobian equals 1.

We neglect the measure of the connection areas of current segments with respect to the total measure  $v(K)$ . Then, the density of the vector  $u$  is:

$$\gamma(u) = \frac{1}{v(K)} \times \frac{1}{l_{\max} - l_{\min}} \times \frac{1}{w_{\max} - w_{\min}} \times \frac{1}{\pi} \quad (39)$$

Let  $P_n$  be the probability of choosing to propose the birth of a free segment and  $P_m$  the probability of choosing to propose the deletion of a free segment.  $n_d(\mathbf{s}')$  is the number of free segments in the configuration  $\mathbf{s}'$ .

Therefore, the probability of doing the reverse jump is  $P_m \times \frac{1}{n_d(\mathbf{s})}$ .

The acceptance probability is:

$$\alpha = \min\{1, R\} \quad (40)$$

with  $R$  the acceptance ratio given by:

$$R = \frac{P_m}{P_n} \times \frac{\pi v(K)(l_{\max} - l_{\min})(w_{\max} - w_{\min})}{n_d(\mathbf{s}')} \times \frac{f(\mathbf{s} \cup \{\zeta\})}{f(\mathbf{s})} \quad (41)$$

The acceptance ratio for the death of a free segment is computed by inverting the ratio of the birth move (41).

*Example 3.* Birth and death of a segment with a single connection.

This move adds a singly-connected segment to the initial configuration.

Let  $P_{nc} \times \frac{1}{2n_{c2}(\mathbf{s}) + n_{c1}(\mathbf{s})}$  and  $P_{mc} \times \frac{1}{n_{c1}(\mathbf{s}')}$  be the proposal probabilities for adding and removing a segment with a single connection.  $n_{c1}(\mathbf{s}')$  is the number of singly-connected segments in the configuration  $\mathbf{s}'$ .

The independent random vector is:

$$u = [(x_u, y_u), l_u, \theta_u, w_u] \quad (42)$$

with  $l_u, \theta_u, w_u$  uniformly chosen on the corresponding space of parameters. The coordinates  $(x_u, y_u)$  are uniformly chosen in a neighborhood  $\epsilon$  of an extremity of the segment (connection area). The new segment in the configuration becomes:

$$\begin{aligned} \zeta &= [(x_\zeta = x_u + x_{sg}, y_\zeta = y_u + y_{sg}), l_\zeta = l_u, \\ &\theta_\zeta = \theta_u, w_\zeta = w_u] \end{aligned} \quad (43)$$

with  $(x_{sg}, y_{sg})$  an extremity in the configuration where the segment can be connected.

The invertible transformation:

$$\begin{aligned} \varphi(\mathbf{s}, u) = \mathbf{s}' &= \{s'_1 = s_1, \dots, s'_n = s_n, s'_{n+1} \\ &= \zeta(u)\} \end{aligned} \quad (44)$$

is not the identity function, but its Jacobian still equals 1.

The density of the vector  $u$  is:

$$\gamma(u) = \frac{1}{\pi v(\epsilon)(l_{\max} - l_{\min})(w_{\max} - w_{\min})} \quad (45)$$

where  $v(\epsilon) = \pi r_c^2$  is the connection area around the extremity of a segment.

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

$$R = \frac{f(s')}{f(s)} \times \frac{P_{mc}}{P_{nc}} \times \frac{(2n_{c2}(s) + n_{c1}(s)) \times \pi v(\epsilon)(l_{\max} - l_{\min})(w_{\max} - w_{\min})}{n_{c1}(s')} \quad (46)$$

and by inverting the ratio in (46), we get the acceptance for the death of a singly-connected segment.

## 6. Simulation and Results

We first present some simulations of the Candy model. Then, results on satellite (optical and radar) and aerial images are shown for different applications: road and hydrographic network extraction.

### 6.1. Simulation of the Candy Model

The Candy model was simulated using the described dynamics. The interaction parameters were fixed to  $\log g_{21} = -6.9$ ,  $\log g_{22} = -0.69$ ,  $\log h_r = -4.6$ ,  $\log h_a = -2.3$ ,  $\tau_{\max} = \delta_{\max} = 0.1$  while the density

parameters was to  $\log \beta = -1.38$  and  $-0.69$  for the samples obtained in the Figs. 7(a) and (b) respectively. As expected, we control the number of segments with the parameter  $\log \beta$ .

Geyer proved in Geyer (1998) that for a locally stable point process, a Markov chain simulated using uniform birth and death moves is Harris-recurrent and ergodic. The Candy model is locally stable and reversible by construction.

Therefore for an empirical evaluation of our dynamics, some graphical tests have been derived (Robert and Casella, 1999). We have looked at the evolution of the cumulative sum of some of the sufficient statistics of the model. Figure 8 shows the evolution of the total number of segments (a), the number of free segments (b), the number of segments with only one connexion (c), and the number of segments with two connexions (d).

We notice that the shape of the curves corresponds to a stable system. Moreover, the statistics correspond to the penalties we have 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 stable.

Of course, this heuristic is not sufficient to prove the convergence of the algorithm, but we use it as a qualitative indicator.

In all the simulations, we have initialized the proposed algorithm with no segments in the initial configuration. The RJMCMC dynamics was run for  $10^5$  iterations. One iteration means one proposed move, which may be accepted or not. The sufficient statistics were

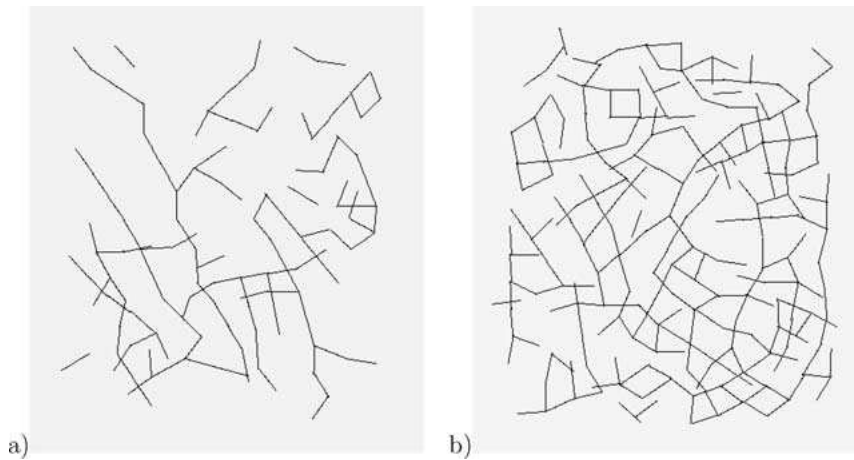
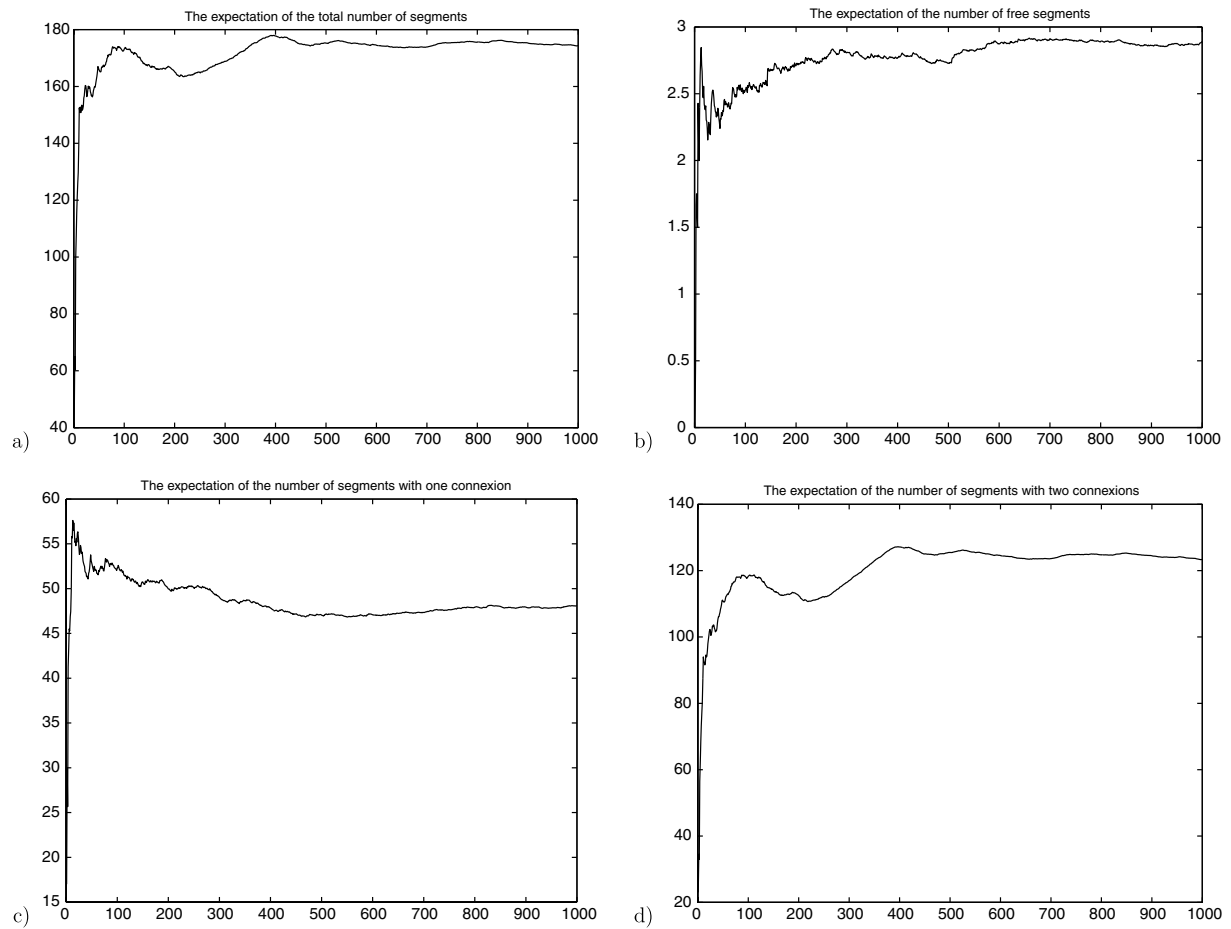


Figure 7. Simulation of the Candy model with different densities: (a)  $\beta = 0.25$  (b)  $\beta = 0.5$ .



*Figure 8.* Statistics of segments. The X axis represents the number of iterations ( $\times 10^2$ ). The Y axis the evolution of the cumulative means of the sufficient statistics of the Candy model: (a) the total number of segments, (b) the number of free segments, (c) the number of segments connected at only one extremity and (d) the number of segments with both extremities connected.



*Figure 9.* Extracting roads from a SPOT satellite image: (a) original image (b) the final result.

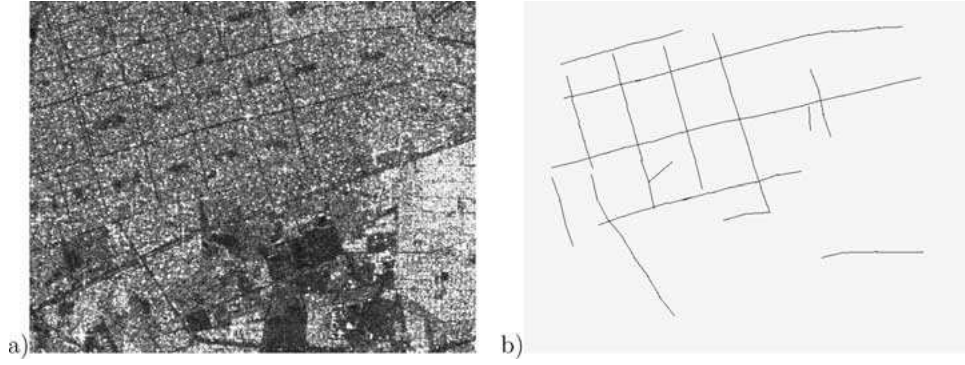


Figure 10. Extracting roads from an ERS radar image: (a) original image (b) the final result.

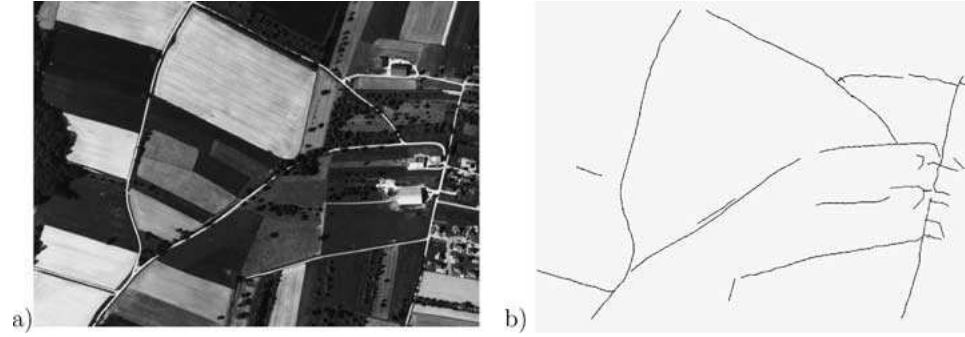


Figure 11. Extracting roads from an aerial image: (a) original image (b) the final result.

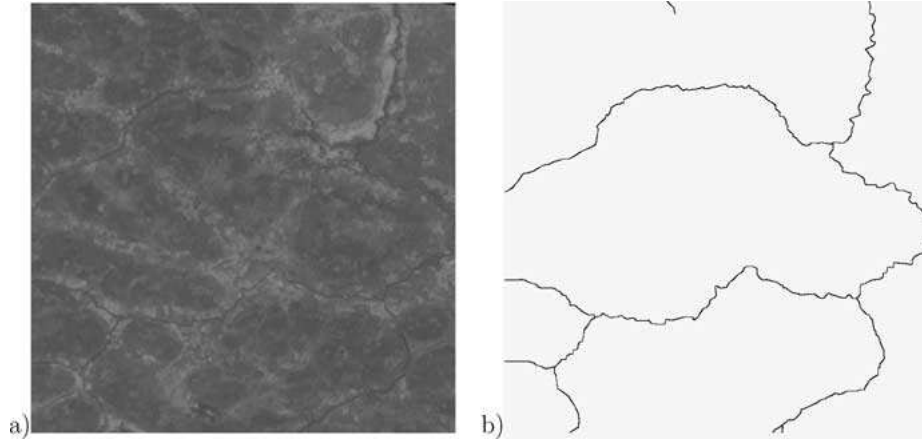


Figure 12. Hydrographic network in a SPOT image: (a) original image; (b) ground truth.

sampled every 100 iterations in order to reduce the correlation between samples.

## 6.2. Test on Real Data

The proposed model was first tested on pieces of SPOT, ERS and aerial images containing roads. We chose

the configuration with no segments as initial state. The RJMCMC simulated annealing algorithm required  $3 \times 10^6$  iterations. The temperature was decreased every  $10^3$  iterations. The initial temperature was  $H_0 = 25$ . The Candy model parameters were fixed at  $\log \beta = -5.0$ ,  $\log g_{21} = -75.0$ ,  $\log g_{22} = -5.0$ ,  $\log h_r = -60.0$ ,  $\log h_a = -45.0$ ; the data energy range for a segment

was [10.0, 50.0]; while the values of the length of a segment lay in [11.0, 21.0]. The width of the segments was set to one. These choices were adopted after several simulations, trying to find the best compromise between exhaustivity of the network and presence of false alarms. In the SPOT and aerial images, roads look like ridges, whereas in the ERS image they look like valleys. The terms given in (29) and (30) respectively were used to reinforce the data term.

There is a formal proof of the existence of a temperature-reduction scheme for simulated annealing based on spatial birth-and-death processes. However, in practice the optimization algorithm is run using a fixed low temperature (van Lieshout, 1994). In order to take advantage of the principle of simulated annealing, for our method based on an RJMCMC dynamics, we chose a logarithmic cooling scheme, which is one of the slowest:

$$H_{n_{\text{iter}}} = \frac{H_0}{\log(1 + n_{\text{iter}})} \quad (47)$$

with  $n_{\text{iter}}$  the number of iterations. The results of road extraction are presented in Figs. 9–11. They were obtained using the same parameters for all the images.

To capture the basic structure of the network, we need around 15–20 minutes on a Sun Sparc station (Ultra 10 at 300 MHz) depending on the density of the network in the image. To complete the network, it may take 2–3 hours for an image. Therefore, a deterministic post-processing after the simulated annealing may reduce the computation time.

In order to evaluate our method, the French Geological Survey (BRGM) has kindly provided a SPOT image representing an hydrographic network, together with its corresponding ground truth (see Fig. 12). The parameters used for this application are the same as in the previous case,  $\log h_r = -40.0$  and  $\log h_a = -25.0$  excepted, in order to penalize less orientation and crossing between segments. In Fig. 13, different results are shown depending on the value of the parameter  $g_{22}$ .

The method discussed in this paper correctly detects the main network structure in the three types of images that have been tested: satellite (optical and radar) and aerial data, without any pre- or post-processing. Furthermore, we can easily adapt our model to extract hydrographical networks. We can avoid false alarms due to isolated segments and we can pass an obstacle whose length is not greater than the maximum length of a segment. Roads are prolonged in appropriate

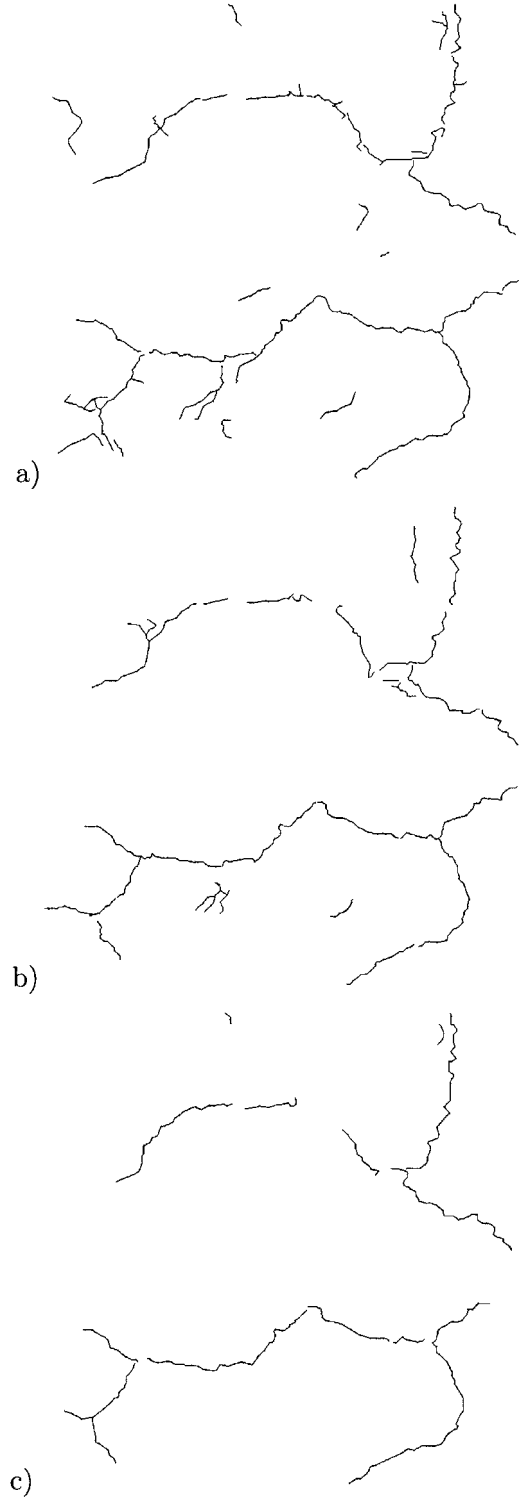


Figure 13. Results of hydrographical network extraction depending on the model parameters: (a)  $\log g_{22} = -2.5$ ; (b)  $\log g_{22} = -5.0$ ; (c)  $\log g_{22} = -7.5$ .

directions. To the best of our knowledge, these problems were not solved in an unsupervised way by previous methods proposed for the extraction of road networks.

## 7. Conclusion and Future Work

In this paper, we have presented a method for thin network extraction from images using the Gibbs point process framework. Our work uses ideas promoted by Baddeley and van Lieshout (1993), van Lieshout (1994), Rue and Syversveen (1998) and Rue and Hurn (1999), which adopt point processes in order to solve image analysis problems using an object-oriented rather than a pixel-oriented approach.

Two main directions may be outlined for future work. From a theoretical point of view, the properties of the Candy model need to be more deeply investigated. Parameter estimation for the Candy model and new simulation dynamics are currently under study. Convergence assignment is also currently being considered within the exact simulation framework. From an application point of view, there is ongoing work studying different statistical tests for the data term and a few new models with different types of interaction between segments. The concrete application goal is map calibration.

Within the same framework of interaction point processes, a model for building extraction is currently being tested (Garcin et al., 2001; Ortner et al., 2002). The most ambitious of our future projects is to construct models that deal with interactions between different kinds of objects, hence taking a first step towards scene modeling.

## Acknowledgments

The SPOT images were provided by the French Space Agency (CNES), the ERS images by the European Space Agency (ESA), the aerial images by the French Mapping Institute (IGN) and the hydrographic network image and ground truth by the French Geological Survey (BRGM). The authors are grateful to Dr. M.N.M. van Lieshout (CWI—Amsterdam) for helpful discussions, to Dr. Ian Jermyn (INRIA—Sophia Antipolis) for his constructive remarks, and to the anonymous referees for useful comments. The first author thanks the French Ministry of Foreign Affairs for partial financial support.

## References

- Baddeley, A.J. and van Lieshout, M.N.M. 1993. Stochastic geometry models in high-level vision. In *Statistics and Images*, Vol. 1, K.V. Mardia and G.K. Kanji (Eds.), *Advances in Applied Statistics*, a supplement to *Journal of Applied Statistics* 20:231–256. Abingdon: Carfax.
- Barzohar, M. and Cooper, D.B. 1996. Automatic finding of main roads in aerial images by using geometric-stochastic models and estimation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18:707–721.
- Bogess, J.E. 1994. Using artificial neural networks to identify roads in satellite images. In *Proceedings of the World Congress on Neural Networks*, San Diego, pp. 410–415.
- Fua, P. and Leclerc, Y.G. 1990. Model driven edge detection. *Machine Vision and Applications*, 3:45–56.
- Fischler, M.A., Tenenbaum, J.M. and Wolf, H.C. 1981. Detection of roads and linear structures in low-resolution aerial imagery using a multisource knowledge integration technique. *Computer Graphics and Image Processing*, 15:201–223.
- Garcin, L., Descombes, X., Zerubia, J. and Le Men, H. 2001. Building detection by Markov object processes and a MCMC algorithm. *INRIA Research Report*, 4206.
- Geman, D. and Jedynak, B. 1996. An active testing model for tracking roads in satellite images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18:1–14.
- Geyer, C. J. and Möller, J. 1994. Simulation and likelihood inference for spatial point process. *Scandinavian Journal of Statistics*, B, 21:359–373.
- Geyer, C.J. 1998. Likelihood inference for spatial point processes. In *Stochastic Geometry, Likelihood and Computation*, O.E. Barndorff-Nielsen, W.S. Kendall and M.N.M. van Lieshout (Eds.), Chapman and Hall: London.
- Graffigne, C. and Herlin, I. 1989. Modélisation de réseaux pour l'Imagerie Satellite SPOT. AFCET, 7eme Congrès de Reconnaissance des Formes et Intelligence Artificielle (Paris), pp. 833–842.
- Green, P. 1995. Reversible jump MCMC computation and bayesian model determination. *Biometrika*, 82:711–732.
- Green, P. 1996. MCMC in image analysis In *Markov Chain Monte Carlo in Practice*, W.R. Gilks, S. Richardson, and D.J. Spiegelhalter (Eds.) pp. 381–399, Chapman and Hall: London.
- van Lieshout, M.N.M. 1994. Stochastic annealing for nearest-neighbour point processes with application to object recognition. *Advances in Applied Probability*, 26:281–300.
- van Lieshout, M.N.M. 2000. *Markov Point Processes and their Applications*. Imperial College Press/World Scientific Publishing: London/Singapore.
- Merlet, N. and Zerubia, J. 1996. New prospects in line detection by dynamic programming. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18:426–431.
- Meyn, S.P. and Tweedie, R.L. 1993. *Markov Chains and Stochastic Stability* Springer-Verlag: London.
- Møller, J. 1998. Markov Chain Monte Carlo and spatial point processes. In *Stochastic Geometry, Likelihood and Computation*, O.E. Barndorff-Nielsen, W.S. Kendall and M.N.M. van Lieshout (Eds.) Chapman and Hall: London.
- Neuenschwander, W.M., Fua, P., Iverson, L., Szekely, G. and Kubler, O. 1997. Ziplock Snakes. *International Journal of Computer Vision*, 25(3):191–201.



- Ortner, M., Descombes, X. and Zerubia, J. 2002. Building extraction from digital elevation models. *INRIA Research Report*, 4517.
- Preston, C.J. 1977. Spatial birth-and-death processes. *Bulletin of the International Statistical Institute*, 46:371–391.
- Refregier, P., Germain, O. and Gaidon, T. 1997. Optimal snake segmentation of target and background with independent Gamma density probabilities application to speckled and preprocessed images. *Optics Communications*, 137:382–388.
- Robert, C. and Casella, G. 1999. *Monte Carlo Statistical Methods* Springer-Verlag.
- Rue, H. and Syversveen, A.R. 1998. Bayesian object recognition with Baddley’s delta loss. *Advances in Applied Probability (SGSA)*, 30(1):55–75.
- Rue, H. and Hurn, M. 1999. Bayesian object identification. *Biometrika*, (3), 649–660.
- Ruelle, D. 1969. *Statistical Mechanics: Rigorous Results*, W.A. Benjamin: Reading, Massachusetts.
- Serendero, M.A. 1989. *Extraction d’Informations Symboliques en Imagerie SPOT: Réseaux de Communication et Agglomérations*. PhD Thesis (in French), Nice-Sophia Antipolis University.
- Stoica, R. 2001. Processus ponctuels pour l’extraction des réseaux linéiques dans les images satellitaires et aériennes. PhD Thesis (in French), Nice Sophia-Antipolis University.
- Stoyan, D., Kendall, W.S., and Mecke, J. 1987. *Stochastic Geometry and its Applications*. John Willey and Sons.
- Tupin, F., Maître, H., Mangin, J.-F., Nicolas, J.-M. and Pechersky, E. 1998. Detection of linear features in SAR images: Application to road network extraction. *IEEE Trans. Geoscience and Remote Sensing*, 36(2):434–453.