

# Recovering Line-networks in Images by Junction-Point Processes

Dengfeng Chai  
Zhejiang University  
chaidf@zju.edu.cn

Wolfgang Förstner  
University of Bonn  
wf@ipb.uni-bonn.de

Florent Lafarge  
INRIA Sophia Antipolis  
florent.lafarge@inria.fr

## Abstract

*The automatic extraction of line-networks from images is a well-known computer vision issue. Appearance and shape considerations have been deeply explored in the literature to improve accuracy in presence of occlusions, shadows, and a wide variety of irrelevant objects. However most existing works have ignored the structural aspect of the problem. We present an original method which provides structurally-coherent solutions. Contrary to the pixel-based and object-based methods, our result is a graph in which each node represents either a connection or an ending in the line-network. Based on stochastic geometry, we develop a new family of point processes consisting in sampling junction-points in the input image by using a Monte Carlo mechanism. The quality of a configuration is measured by a probability density which takes into account both image consistency and shape priors. Our experiments on a variety of problems illustrate the potential of our approach in terms of accuracy, flexibility and efficiency.*

## 1. Introduction

Line-network extraction is a computer vision topic widely explored during the two last decades. The pioneer works have been led on the well-known road extraction problem from remote sensed images [2, 11, 12]. Line-network extraction is also of interest in other problems like blood vessel detection from medical images [9, 10], or structure extraction from natural textures [8]. This research topic requires high-level considerations based on shape and structure. Few methods are able to recover the line-networks from images in an automatic and robust way because of occlusions, shadows, and a wide variety of irrelevant objects present in the scenes.

### 1.1. Related works

Three types of approach can be distinguished in the literature: pixel-based, line-segment-based and graph-based.

The pixel-based approaches constitute the most common type of models. The network extraction is seen as a bi-

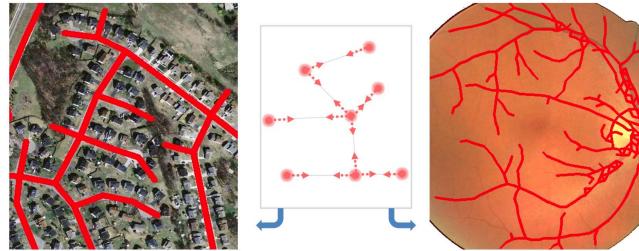


Figure 1. Our Junction-point process (middle) explores the graph configurations in images in order to extract line-networks, here a road network (left) from an aerial image and blood vessels (right) from a retinal image.

nary segmentation problem in which each pixel belongs either to the line-network or to the background. These models can benefit from powerful segmentation and tracking tools based on shape and appearance information. However, they are not adapted to recover the network structure as they ignore the notion of objects as shown in Fig. 2. McKeown and Denlinger proposed road-surface texture correlation and road-edge to recover the road center, its width and local properties from aerial images [12]. Barzohar and Cooper proposed geometric-probabilistic models for road appearance, and applied dynamic programming to track roads [2]. Poullis et al. merged perceptual grouping and segmentation techniques into a unified framework to segment road pixels from multi-sensor data[17]. Marin et al. [10] and Mnih and Hinton [14] proposed neural network based approaches. The former developed a 7-D vector composed of gray-level and invariant features for segmenting blood vessels, whereas the latter used a massive amount of training data to detect roads in aerial images. Active contour methods are also popular, especially to impose shape priors. Mayer et al. modeled road boundaries as active contours and exploited scale-space behavior to extract road boundaries [11]. Rochery et al. developed higher-order active contours, which generalize the linear functionals in the energies to the arbitrary polynomial functionals. This method is dedicated to road detection from remotely sensed images [18]. Furthermore, Peng et al. developed a phase field higher-order active contours for road network

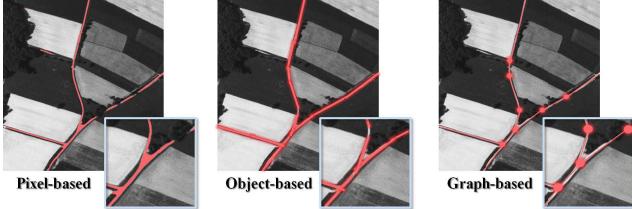


Figure 2. Different types of network extraction models. The graph-based models constitute a natural way to describe the line-network, contrary to pixel-based and object-based methods which do not take into account its structural aspect.

extraction [16]. Several methods exploiting geodesic distances have also been proposed, as [15] where tubular structures are extracted from bi-dimensional images by computing geodesic curves over a four-dimensional space that includes local orientation and scale.

The object-based models represent the line-networks as a configuration of geometric objects, typically line-segments. These models are usually robust in the presence of occlusions as they allow certain types of structures to be favored. However, finding the optimal object configuration is a difficult task and the object connection is hard to obtain in practice as shown in Fig. 2. Lacoste et al. [7] proposed a model to describe the interaction between line-segments based on overlapping, alignment and connection considerations. This model is complex and relies on a large number of parameters whose tuning is a delicate task. Lafarge et al. developed a more general model by reducing the object interactions [8], but at the expense of result accuracy. This model is extended in [20] to efficiently tackle large-scale data using a parallelization scheme.

The graph-based representation is the most natural way to address the problem as the network structure is guaranteed by construction. Each node of the graph corresponds either to a junction between at least two line-segments, or to an ending point as illustrated in Fig. 2. Graphs have rarely been used in the literature mainly because these mathematical tools are often difficult to manipulate, especially to extract hierarchical networks. Turetken et al. [19] construct a general graph from seed points in which an optimal path is searched for. Hu et al. utilized road footprints as features which are tracked to detect intersections and extract the network from aerial images [6]. The length of the graph edges are fixed, which makes the representation not flexible. Note also that some works have been proposed to artificially generate road networks using procedural models, *e.g.* [4].

## 1.2. Motivations and contributions

Most of the existing methods focused on appearance and shape considerations, and have ignored the structural aspect of the problem. Although graph-based representations are relatively unpopular, they remain the most natural way

of modeling line-networks. In particular, they guarantee a structurally-coherent solution, and are adapted to multi-scale networks using graph attributes as edge width.

The solutions proposed by Turetken et al. [19] and Hu et al. [6] underline the potential of the graph-based methods, but they suffer from some strong limitations. The former generates only graphs without cycles, *i.e.* tree structures, whereas the latter provides a non-flexible representation with a constant edge length for road extraction only. Finding an efficient and flexible graph-based method is an open challenge. Our solution brings several contributions:

- **Structural guarantees.** Every potential solution has a coherent network structure as our configuration space is the set of the planar graphs embedded in the image support. This constitutes a significant advantage compared to the pixel-based and object-based methods.
- **New point processes.** We propose a new family of point processes, called *junction-point processes*. Contrary to the conventional marked point processes used in the literature, *e.g.* [7, 8, 20], junction-point processes do not require complex geometric priors. The sampling procedure is thus more stable and the model parameters are highly reduced.
- **Flexibility.** Our algorithm can be applied to a variety of network extraction problems, as illustrated in Fig. 1. A learning procedure is proposed for selecting the optimal model parameters with respect to a given problem. In addition, the algorithm is automatic and does not depend on any initialization.

## 2. Junction-point processes

**Point process.** Point processes describe an unordered set of points in a compact set  $F \subset R^k$ , where  $R^k$  is a  $d$ -dimensional space (here,  $k = 2$ ). For  $n = 1, 2, \dots$ , let  $\Omega_n$  be the set of configurations  $\omega = \{\omega_1, \dots, \omega_n\}$  that consists of  $n$  unordered points  $\omega_i \in F$ . A point process on  $F$  is a mapping  $\Psi$  from a probability space to the set of configurations  $\Omega = \bigcup_{n=1}^{\infty} \Omega_n$ , such that, for all bounded Borel sets  $S \subset F$ , the number of points  $N_{\Psi}(S)$  falling in  $S$  is a finite random variable.

The reference point process is the homogeneous Poisson process for which the number of points follows a discrete Poisson distribution with parameter  $\lambda_F$ . The position of the points is uniformly and independently distributed in  $F$ . Point processes can provide more complex realizations of points by using a probability density  $h(\cdot)$  defined in  $\Omega$  and a reference measure  $\mu(\cdot)$  under the condition that the normalization constant of  $h(\cdot)$  is finite:

$$\int_{\omega \in \Omega} h(\omega) d\mu(\omega) < \infty \quad (1)$$

The measure  $\mu(\cdot)$  having the density  $h(\cdot)$  is usually defined via the intensity measure  $\nu(\cdot)$  of an homogeneous Poisson process. Specifying a density  $h(\cdot)$  allows the insertion of data consistency, and also the creation of spatial interactions between the points. Note that the probability density  $h(\cdot)$  can be expressed by a Gibbs energy  $U(\cdot)$  such that

$$h(\cdot) \propto \exp -U(\cdot) \quad (2)$$

**Sampling a point process.** Searching for the configuration  $\omega^*$  which maximizes the probability density  $h$ , such that

$$\omega^* = \arg \max_{\omega \in \Omega} h(\omega) \quad (3)$$

is not a conventional optimization problem as the probability density  $h$  is multi-modal and defines in a configuration space having a variable dimension. A Monte Carlo sampler is usually required to find an approximation of  $\omega^*$ , and more precisely the Reversible Jump Markov Chain Monte Carlo (RJMCMC) sampler [5]. This algorithm consists of simulating a discrete Markov Chain  $(X_t)_{t \in \mathbb{N}}$  on the configuration space  $\Omega$ , converging towards an invariant measure specified by  $h$ . At each iteration, the current configuration  $\omega$  of the chain is locally perturbed to a configuration  $\omega'$  according to a density function  $Q(\omega \rightarrow \cdot)$ , also called a kernel. The perturbations are local, which means that  $\omega$  and  $\omega'$  are very close, and differ by no more than one or two points. The configuration  $\omega'$  is then accepted as the new state of the chain with a probability depending on the probability density variation between  $\omega$  and  $\omega'$ , and a relaxation parameter  $T_t$ . The kernel  $Q$  can be formulated as a mixture of sub-kernels  $Q_m$  chosen with a probability  $p_m$  such that

$$Q(\omega \rightarrow \cdot) = \sum_m p_m Q_m(\omega \rightarrow \cdot) \quad (4)$$

Each sub-kernel is usually dedicated to specific types of moves, as the creation/removal of a point (Birth and Death kernel) or the modification of parameters of a point (*e.g.* translation kernel). The kernel mixture must allow any configuration in  $\Omega$  to be reached from any other configuration in a finite number of perturbations (irreducibility condition of the Markov chain), and each sub-kernel has to be reversible, *i.e.* able to propose the inverse perturbation. The RJMCMC sampler is controlled by the relaxation parameter  $T_t$ , called the temperature, depending on time  $t$  and approaching zero as  $t$  tends to infinity. Although a logarithmic decrease of  $T_t$  is necessary to ensure the convergence to the global minimum from any initial configuration, one uses a faster geometric decrease which gives an approximate solution close to the optimum [1].

**From parametric objects to planar graphs.** Point processes are attractive tools in vision as they allow the manipulation of parametric objects. Indeed, some additional attributes  $w_i$  can be added to each point  $\omega_i$  in order to create

---

### Algorithm 1 RJMCMC sampler [5]

---

- 1- Initialize  $X_0 = \omega^{(0)}$  and  $T_0$  at  $t = 0$ ;
- 2- At iteration  $t$ , with  $X_t = \omega$ ,

- Choose a sub-kernel  $Q_m$  according to probability  $p_m$
- Perturb  $\omega$  to  $\omega'$  according to  $Q_m(\omega \rightarrow \cdot)$
- Compute the Green ratio

$$R = \frac{Q_m(\omega' \rightarrow \omega)}{Q_m(\omega \rightarrow \omega')} \left( \frac{h(\omega')}{h(\omega)} \right)^{\frac{1}{T_t}} \quad (5)$$

- Choose  $X_{t+1} = \omega'$  with probability  $\min(1, R)$ , and  $X_{t+1} = \omega$  otherwise
- 

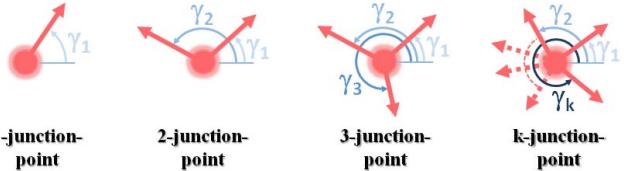


Figure 3. Junction-point. A junction-point is specified by the spatial coordinates in  $F$ , a number  $k$  of directions, and the angle  $\gamma_i$  giving the direction  $i$  where is located an adjacent point.

parametric objects  $x_i = (\omega_i, w_i)$ . By adding a length and an orientation to each point for instance, one can generate configurations of line-segments with geometric interactions, as in [7]. As explained in Section 1, such an object-based technique, called marked point processes in the literature, is not adapted to address line-network extraction problems.

We propose a new family of point processes able to manipulate planar graphs, called *junction-point processes*. The main idea relies on the fact that each point of a realization informs the directions where its adjacent points are located, as illustrated on Fig. 3. A junction-point process on  $F$  is a point process on  $F$  for which every point  $\omega_i \in F$  is completed by a set of directions  $(\gamma_i^{(1)}, \dots, \gamma_i^{(k)})$ , and optionally a set of additional parameters  $(w_i^{(1)}, \dots, w_i^{(k)})$ , so that a point configuration is associated to a unique planar graph in  $F$ . We define a  $k$ -junction-point  $x_i$  by

$$x_i = (\omega_i, \gamma_i^{(1)}, \dots, \gamma_i^{(k)}, w_i^{(1)}, \dots, w_i^{(k)}) \quad (6)$$

where  $k$  represents the number of directions. We denote by  $G_x$ , the planar graph associated to the junction-point configuration  $x = \{x_1, \dots, x_n\}$ . The additional parameters  $(w_i^{(1)}, \dots, w_i^{(k)})$  of the junction-point  $x_i$  correspond to user-defined attributes on adjacent edges of the node  $i$  in the graph  $G_x$ . Contrary to the conventional marked point processes used in the literature, junction-point processes do not require complex geometric priors as a graph structure is

directly guaranteed by construction. Fig. 4 illustrates the advantages of junction-point processes for addressing line-network extraction problems.

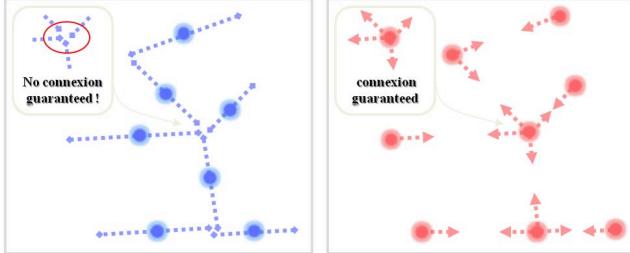


Figure 4. Point process. A marked point process of line-segments (left) cannot describe a line-network with accuracy as the line-segments are not ideally connected. To the contrary, a junction-point process (right) brings structural guarantees as a unique planar graph is associated to each junction-point configuration.

### 3. Problem formulation

Let us consider a junction-point process on the 2D domain supporting the input image  $I$ . In order to be able to extract hierarchical line-networks, a line width is considered for each edge of a graph  $G_x$ . The parameter  $w_i^{(k)}$  (see Eq. 6) represents the line width associated to the  $k^{\text{th}}$  direction of the junction-point  $x_i$  in this model. In a hierarchical network, one can assume that the widths of the lines can take only a few possible values. In practice, the domain of this parameter is discrete.

The probability density  $h$  extended to the junction-point configurations can be expressed as a product of a density  $h_d$  measuring the consistency of a junction-point configuration with the data, and a density  $h_p$  acting as a shape prior on the line-network

$$h(x) \propto h_d(x)h_p(x) \quad (7)$$

#### 3.1. Data consistency

As shown in Fig. 5, pixels on the lines usually have similar colors, and pixels on the line boundaries usually have large gradients. We use these two conventional assumptions to design the data consistency density  $h_d$ . In order to extract the dominant characteristics of the lines, two normalized histograms  $H_r$  and  $H_n$  are learned from samples to represent the color distributions of the line regions and non-line regions respectively. Using histograms allow us to be relatively robust to the presence of clutter objects as small vehicles, trees and shadows for road extraction.

By assuming the conditional independence of pixels in the image, one can express the global data density through a product of local likelihoods on each pixel  $p$  of the image

$$h_d(x) \propto \prod_{p \in I} P(I_p|G_x) \quad (8)$$

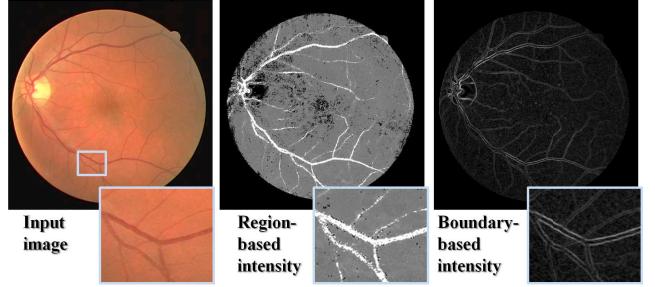


Figure 5. Data consistency. Both region-based intensity (middle) and boundary-based intensity (right) are used to detect the class of interest from the input image (left, here a retinal image). These two criteria bring complementary information (see the blood vessels with different widths on the close-ups).

where  $I_p$  is the color intensity of the image at  $p$ . The local likelihood at pixel  $p$  is expressed by taking into account both color similarity on the lines and discontinuity on the line borders.

$$P(I_p|G_x) = \begin{cases} d_{\text{inside}}(I_p) \cdot \nabla I(p) & \text{if } p \in G_x \\ d_{\text{outside}}(I_p) \cdot \nabla I(p) & \text{otherwise} \end{cases} \quad (9)$$

where  $d_{\text{inside}}$  (respectively  $d_{\text{outside}}$ ) represents the color distributions of the line regions (resp. non-line regions) induced from the normalized histogram  $H_r$  (resp.  $H_n$ ).  $\nabla I(p)$  is the distribution of the image gradient magnitude at  $p$  equal to 1 when  $p$  does not belong to the boundary  $\partial G_x$  of the line regions.

#### 3.2. Shape priors

The density  $h_p$  is introduced to favor certain shapes of graphs. Three different criteria are taken into account to characterize the shape of a graph  $G_x$  from a junction-point configuration  $x$ : the graph connectivity, the edge orientation and the line width. The density  $h_p$  can thus be formulated through a product of three densities by

$$h_p(x) \propto h_{\text{connectivity}}(x) \cdot h_{\text{orientation}}(x) \cdot h_{\text{width}}(x) \quad (10)$$

**Graph connectivity.** The complexity of a graph can be analyzed through the graph connectivity. The number of edges is determined by the number of junction-points and their number of directions for each of them. The number of faces is determined by the numbers of edges and vertices according to Euler's formula. In other words, one can encourage graphs to have a certain complexity knowing the occurrence probability of a  $k$ -junction-point for every  $k \geq 1$ . The graph connectivity density can thus be expressed as

$$h_{\text{connectivity}}(x) = \frac{(\sum_{k \geq 1} n_k)!}{\prod_{k \geq 1} n_k!} \prod_{k \geq 1} p_k^{n_k} \quad (11)$$

where  $n_k$  is the number of  $k$ -junction-point in  $x$ , and  $p_k$  is the occurrence probability of a  $k$ -junction-point in  $x$  whose value is learned from samples. In particular, we have  $\sum_{k \in \mathbb{N}^*} p_k = 1$ . In the sequel, we denote by  $\mathcal{P}$  the set of the occurrence probabilities  $p_k$  with  $k \geq 1$ .

**Edge orientation.** The relative orientation of the edges in a graph constitutes an important criterion to characterize graph shapes. As shown on Fig. 1, the angles between the lines are significantly different between a road network and blood vessels. Given a  $k$ -junction-point  $x_i$ , we denote by  $\beta_i^{(l)}$  the relative normalized angle between the  $l^{th}$  and the  $l+1^{th}$  directions, such that  $\beta_i^{(l)} = \frac{(\gamma_i^{(l+1)} - \gamma_i^{(l)})}{2\pi}$ . A Dirichlet law is considered to model the distribution of the relative normalized angles  $\beta_i^{(l)}$ :

$$\text{Dir}(\beta_i^{(1)}, \dots, \beta_i^{(k)}; \alpha_1, \dots, \alpha_k) = \begin{cases} 1 & \text{if } k = 1 \\ \frac{\prod_{l=1}^k \Gamma(\alpha_l)}{\Gamma(\sum_{l=1}^k \alpha_l)} \prod_{l=1}^k \beta_i^{(l)^{\alpha_l-1}} & \text{otherwise} \end{cases} \quad (12)$$

where  $\{\alpha_1, \dots, \alpha_k\}$  are the parameters of the Dirichlet distribution of the  $k$ -junction-points. The edge orientation density can thus be formulated as a product of Dirichlet distributions on every junction-point of  $x$ :

$$h_{\text{orientation}}(x) = \prod_{i=1..n} \text{Dir}(\beta_i^{(1)}, \dots, \beta_i^{(k)}; \alpha_1, \dots, \alpha_k) \quad (13)$$

In the sequel, we denote by  $\alpha$ , the collection of the parameter sets  $\{\alpha_1, \dots, \alpha_k\}$  of the Dirichlet distribution of the  $k$ -junction-points with  $k \geq 1$ .

**Line width.** The line width density consists of favoring the line widths whose occurrences are the highest. A normalized histogram  $H_w$  is estimated from samples to represent the width distribution. Then, we have:

$$h_{\text{width}}(x) = \prod_{j=1..m} d_{\text{width}}^{(j)}(x) \quad (14)$$

where  $d_{\text{width}}^{(j)}$  measures the difference between the ratio of the width of value  $j$  predicted in  $H_w$  with the one in  $x$ .

### 3.3. Model parameters

The proposed model depends on a set of parameters  $\{\lambda_F, \mathcal{P}, \alpha, H_w, H_r, H_n\}$ . These parameters can be learned from annotated image samples by a Maximum Likelihood Estimation. The Poisson parameter  $\lambda_F$  which represents the expected number of junction-points is calculated as the average number of junction-points in the annotated image samples. The Maximum Likelihood Estimate of each occurrence probability  $p_k \in \mathcal{P}$  is given by the rate of  $k$ -junction-points in the annotated image samples. Since

there is no closed-form solution for the Maximum Likelihood Estimate of Dirichlet distribution, we apply a fixed-point iteration to estimate the parameter set  $\alpha$  [13]. The histograms  $H_r$  and  $H_n$  are estimated by counting the number of pixels with different colors in line regions and non-line regions respectively.  $H_w$  is estimated by counting the number of lines with different widths from the annotated image samples.

### 3.4. Optimization

The RJMCMC sampler, described in Alg. 1, is used to find an junction-point configuration close to the configuration maximizing the density  $h$ . Two kinds of proposition kernels are considered in the sampler: the birth and death kernel  $\mathcal{Q}_{BD}$ , and the translation kernel  $\mathcal{Q}_T$ . The use of these two kernels is sufficient to guarantee the irreducibility condition of the Markov chain.

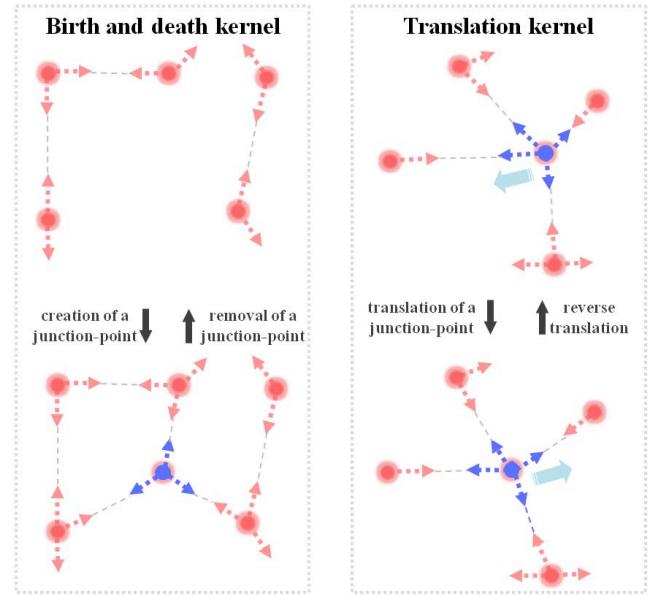


Figure 6. Proposition kernels. The birth and death kernel (left) proposes to create or remove a junction-point randomly in the current configuration. The translation kernel (right) is devoted to the displacement of a junction-point.

**Birth and death kernel.** The uniform birth and death kernel allows a junction-point to be added or removed randomly in a configuration  $x$ , as illustrated in Fig. 6. Such transformations, which correspond to jumps into the subspaces of higher (birth) and lower (death) dimensions, modify the complexity of the graph  $G_x$ . As detailed in [3], the proposition kernel ratio of a birth can be expressed by

$$\frac{\mathcal{Q}_{BD}(x' \rightarrow x)}{\mathcal{Q}_{BD}(x \rightarrow x')} = \frac{p_d}{p_b} \frac{\lambda_F}{n(x')} \quad (15)$$

where  $\lambda_F$  is the Poisson parameter representing the expected number of junction-points in the domain  $F$ ,  $n(x')$  is

the number of junction-points in the proposed configuration  $x'$ , and  $p_d$  (resp.  $p_b$ ) is the probability of choosing a death (resp. a birth). In practice, we have  $p_d = p_b = 0.5$ . In case of a death, the proposition kernel ratio corresponds to the inverse birth's ratio.

Note that we impose some restrictions on the creation/removal of the junction-points in order to guarantee the coherence of the graph. In particular a new junction-point cannot be proposed if it generates crossing edges. Note also that when a junction-point is added or removed, the directions of the adjacent junction-points are updated.

**Translation kernel.** The translation kernel allows a junction-point to be moved without modifying the graph complexity (Fig. 6). As the translation of a junction-point is proposed randomly, the proposition kernel ratio is simply given by

$$\frac{\mathcal{Q}_T(x' \rightarrow x)}{\mathcal{Q}_T(x \rightarrow x')} = 1 \quad (16)$$

In practice, a junction-point is moved in a small domain centered around its initial position. This proposition kernel is particularly interesting at the end of the sampling procedure to locally adjust the shape of the graph, as shown on Fig. 7.

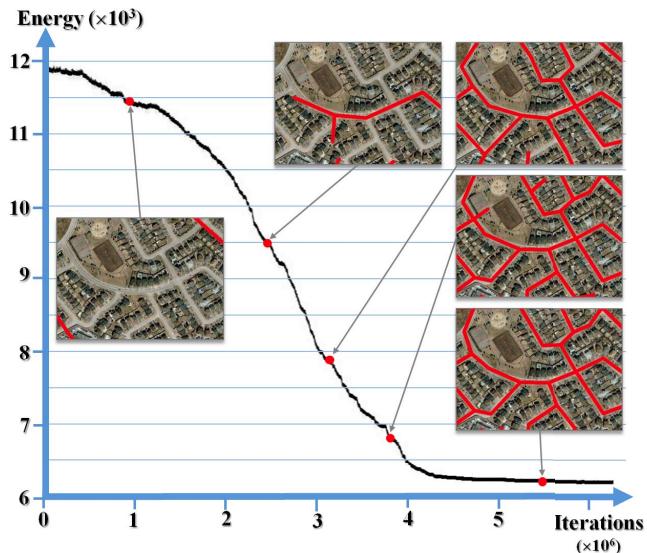


Figure 7. Evolution of the junction-point configuration during the sampling. At high temperature, junction-points of low quality are frequently accepted, leading to non relevant graphs (left close-ups). When the temperature decreases, the process becomes progressively selective (top right close-ups) as the Gibbs energy of our model ( $U(x) \propto -\ln[h(x)]$ ) is decreasing. At low temperature, the current junction-point configuration evolves through some local adjustments: the process is stabilizing close to the global minimum (bottom right close-up).

## 4. Experimental results

### 4.1. Flexibility

The algorithm has been tested on a variety of line-network extraction problems ranging from road extraction from satellite and aerial images to blood vessel extraction from retinal images through more atypical problems as facade structure extraction (Fig. 9). Although each of these problems has its own specificities, our model formulation is general enough to provide coherent results in each case. In particular, the data consistency term is able to distinguish a variety of lines with different colors and widths in images. In addition, the model parameters are robustly estimated. In order to validate our shape prior, our model has been simulated without input images. As shown on Fig. 8, the shape prior gives simulated networks more realistic and structured than the Quality Candy model proposed in [7]. The shape prior also allows free-form line-networks to be recovered as shown on Fig. 9-right.

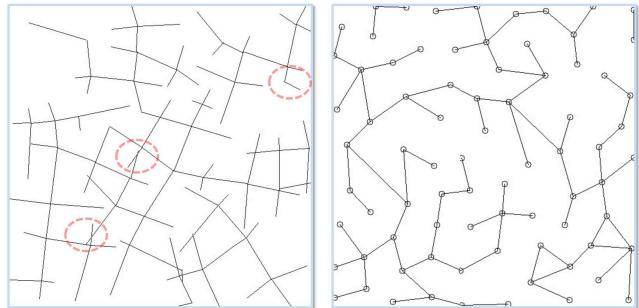


Figure 8. Comparison between our junction-point process and a marked point process of line-segments [7]. Networks are simulated without using input images. Our junction-point process (right) proposes a more natural and realistic representation of line-networks than the line-segment-based model (left). In particular, the line-segments are not correctly connected (see red marks) and the network representation is not structured.

### 4.2. Accuracy

The accuracy of our algorithm has been evaluated on the road extraction problem from satellite images, and compared to existing methods [16, 21, 22]. The input image shown on Fig. 10 is particularly challenging as it describes a dense urban area containing many irrelevant objects and whose roads are difficult to distinguish. Our algorithm provides a more structured result than these methods. Note that they do not produce the vectorization of roads contrary to our algorithm. This constitutes a notable advantage for industrial applications. As illustrated on Fig. 9, the results obtained on hierarchical networks in which lines have different widths are convincing as few minor lines are omitted.

Tab. 1 presents some quantitative comparisons with two line-segment point processes [8, 20]. Results produced by

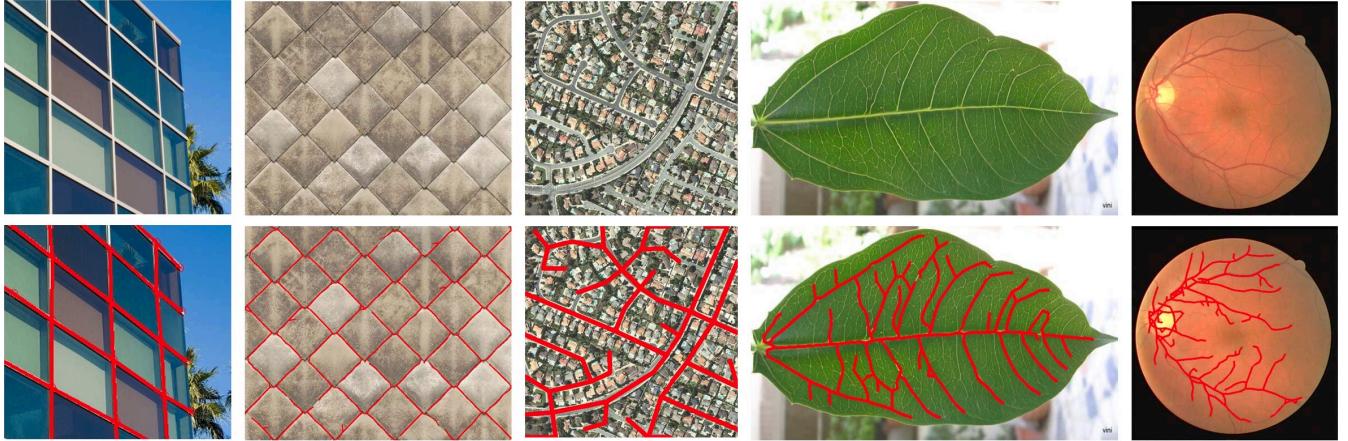


Figure 9. Shape prior flexibility. Our algorithm is able to extract both regular (first and second columns, facade and tiles) and free-form (three left columns, roads in a residential area, leaf and blood vessels in a retinal image) line-networks. Note in particular that the line-networks with different widths are recovered with few omissions, eg blood vessels or leaf.

junction-points are competitive in terms of correctness and completeness. Additional comparisons can be found at <http://www.sop.inria.fr/members/Florent.Lafarge/benchmark/evaluation.html>.

### 4.3. Performance

The computation time of our algorithm is relatively fast compared to existing methods. For example, 270 seconds are necessary to obtain the road extraction result presented on Fig. 9-middle. More than half an hour is required from such an image by the methods proposed by Lacoste et al. [7]. In particular, the conventional point process methods waste time in trying to impose object structure through complex geometric priors. In [7], the sampling is long and tedious as the object connection is never accomplished. We avoid this problem as each possible configuration is structured by construction.

Table 1. Quantitative comparison with different point processes from the *Tiles* image presented on Fig. 9. As the methods use different types of representation (complexity), a pixel-based evaluation is used (Correctness = TP/(TP+FP), Completeness = TP/(TP+FN)).

	Junction-points	Verdie[20]	Lafarge[8]
Correctness	0.462	0.673	0.346
Completeness	0.649	0.783	0.518
Complexity	graph	line	line
Time	227s	103s	293s

### 4.4. limitations

Our model is based on the assumption that a network is represented by piecewise straight lines. The curved parts of a network can thus be more delicate to extract, at least com-

putation times are increased as more junction-points are required to correctly recover these parts. Moreover, sampling junction-points require more fineness than sampling line-segments. Indeed, missing a junction-point penalizes all its adjacent lines whereas missing a line-segment is less disadvantageous in terms of network coverage.

## 5. Conclusion

We propose an original stochastic model for extracting line-networks from images. The algorithm has several interesting characteristics. Based on a graph representation, every potential result has a coherent network structure and is vectorized contrary to the pixel-based and object-based methods. The algorithm is also flexible and can be applied to a variety of network extraction problems without tuning parameter models by trial and error. Finally the junction-point processes have significant advantages compared to the conventional point processes as (i) they do not require the introduction of complex geometric priors, and (ii) their sampling is more stable.

In future works, it would be interesting to improve the sampling procedure of the junction-point processes to reduce the computation time. One option could be to develop a parallelization scheme as in [20] so that several perturbations can be simultaneously proposed. Another interesting challenge is to adapt our approach to the networks of 3D-lines whose configuration spaces are significantly larger.

## Acknowledgments

This work is supported by the National Natural Science Foundation of China (No.41071263) and the Scientific Research Fund of Zhejiang Provincial Education Department (No.Y200804874). We thank the reviewers for their val-

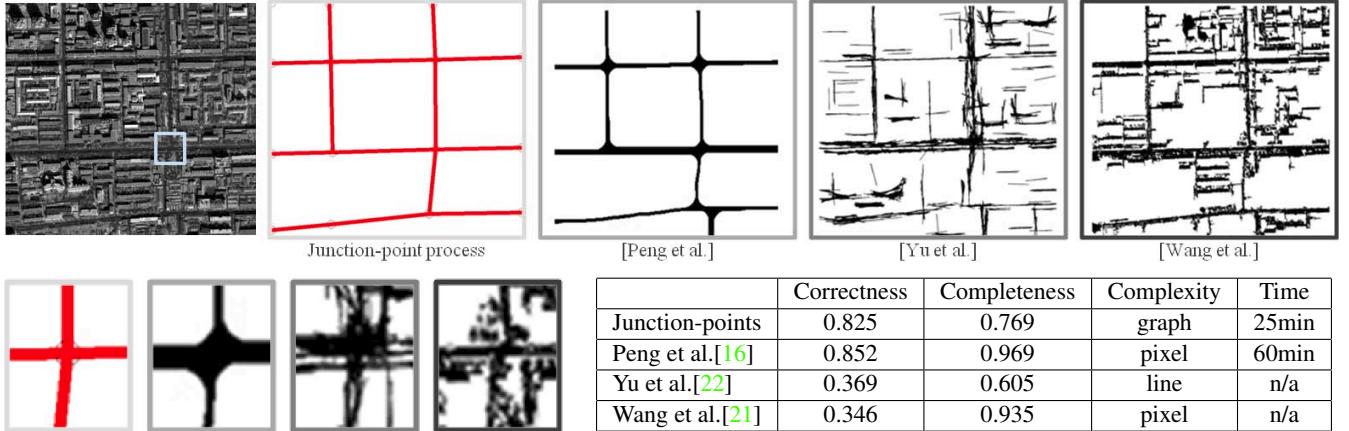


Figure 10. Comparison with three specialized algorithms dedicated to road-network extraction [16, 21, 22]. Our algorithm provides a more structured result than these methods. Only 14 junction-points are detected leading to a very low representation complexity without altering the accuracy. In particular, the road connections are ideally recovered contrary to other algorithms (see close-ups). Although our junction-point process is slightly less accurate than the higher-order active contour model proposed by [16], running times are significantly faster (see table).

able comments, Ting Peng and the DRIVE database for providing images and Ground Truth.

## References

- [1] A. J. Baddeley and M. V. Lieshout. Stochastic geometry models in high-level vision. *Journal of Applied Statistics*, 20(5-6), 1993. 3
- [2] M. Barzohar and D. Cooper. Automatic finding of main roads in aerial images by using geometric-stochastic models and estimation. *T-PAMI*, 18(7), 1996. 1
- [3] F. Chatelain, X. Descombes, F. Lafarge, C. Lantuejoul, C. Mallet, R. Minlos, M. Schmitt, M. Sigelle, R. Stoica, and E. Zhizhina. *Stochastic Geometry for Image Analysis*. Wiley-ISTE, 2011. 5
- [4] E. Galin, A. Peytavie, N. Marechal, and E. Guerin. Procedural generation of roads. In *Eurographics*, 2010. 2
- [5] P. J. Green. Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 82(4), 1995. 3
- [6] J. Hu, A. Razdan, J. Femiani, M. Cui, and P. Wonka. Road network extraction and intersection detection from aerial images by tracking road footprints. *T-GRS*, 45(12), 2007. 2
- [7] C. Lacoste, X. Descombes, and J. Zerubia. Point process for unsupervised line network extraction in remote sensing. *T-PAMI*, 27(10), 2005. 2, 3, 6, 7
- [8] F. Lafarge, G. Gimelfarb, and X. Descombes. Geometric feature extraction by a multi-marked point process. *T-PAMI*, 32(9), 2010. 1, 2, 6, 7
- [9] D. Lesage, E. D. Angelini, I. Bloch, and G. Funka-Lea. A review of 3d vessel lumen segmentation techniques: Models, features and extraction schemes. *Medical Image Analysis*, 13(6), 2009. 1
- [10] D. Marin, A. Aquino, M. Gegundez-Arias, and J. Bravo. A new supervised method for blood vessel segmentation in retinal images by using gray-level and moment invariants-based features. *IEEE Trans. on Medical Imaging*, 30(1), 2011. 1
- [11] H. Mayer, I. Laptev, and A. Baumgartner. Multi-scale and snakes for automatic road extraction. In *ECCV*, 1998. 1
- [12] J. McKeown, D.M. and J. Denlinger. Cooperative methods for road tracking in aerial imagery. In *CVPR*, 1988. 1
- [13] T. P. Minka. Estimating a Dirichlet distribution. 2003. 5
- [14] V. Mnih and G. Hinton. Learning to detect roads in high-resolution aerial images. In *ECCV*, 2010. 1
- [15] M. Pechaud, R. Keriven, and G. Peyre. Extraction of tubular structures over an orientation domain. In *CVPR*, 2009. 2
- [16] T. Peng, I. Jermyn, V. Prinet, and J. Zerubia. Extended phase field higher-order active contour models for networks. *IJCV*, 88, 2010. 2, 6, 8
- [17] C. Poullis and S. You. Delineation and geometric modeling of road networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65(2), 2010. 1
- [18] M. Rochery, I. Jermyn, and J. Zerubia. Higher order active contours. *IJCV*, 69, 2006. 1
- [19] E. Turetken, F. Benmansour, and P. Fua. Automated reconstruction of tree structures using path classifiers and mixed integer programming. In *CVPR*, 2012. 2
- [20] Y. Verdie and F. Lafarge. Efficient monte carlo sampler for detecting parametric objects in large scenes. In *ECCV*, 2012. 2, 6, 7
- [21] R. Wang and Y. Zhang. Extraction of urban road network using quickbird pan-sharpened multispectral and panchromatic imagery by performing edge-aided post-classification. In *ISPRS*, 2003. 6, 8
- [22] Z. Yu, V. Prinet, C. Pan, and P. Chen. A novel two-steps strategy for automatic gis-image registration. In *ICIP*, 2004. 6, 8