Stochastic geometric models for image analysis

Marie-Colette van Lieshout

CWI / University of Twente

History

Infancy

- Besag (1986) and Geman and Geman (1984): seminal work on the restoration of pictures degraded by noise;
- 'low level' tasks, e.g. de-noise, sharpen, segment, or classify; overview in Mardia and Kanji (1993).

Growing up

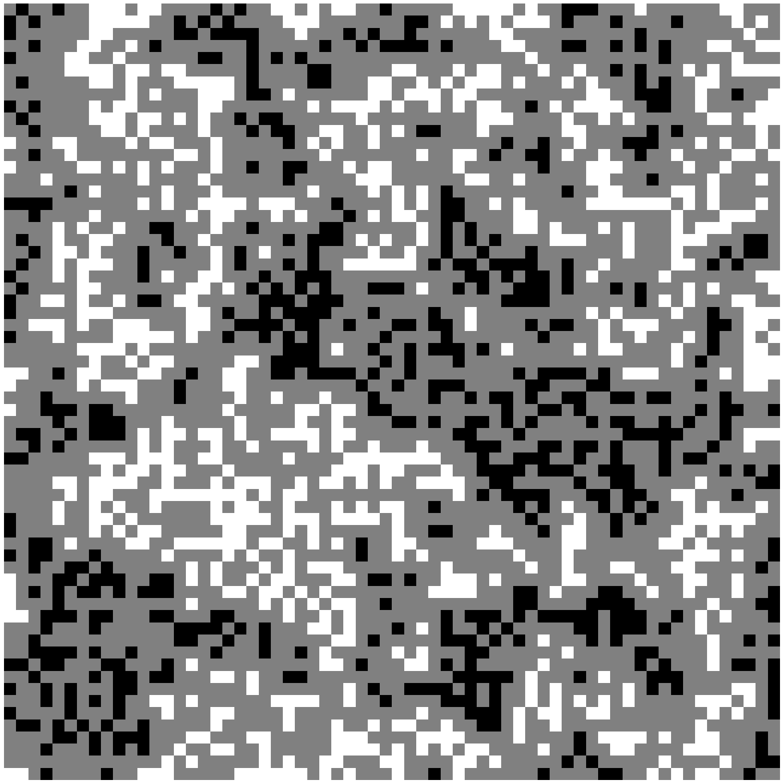
- 1990s: shift towards 'high level' tasks of describing image content;
- early work includes Molina and Ripley (1989), Ripley and Sutherland (1990), Baddeley and Van Lieshout (1992, 1993), Grenander and Miller (1994).

Maturity

- challenges arising from data explosion;
- intermediate approach building on tessellation models, e.g. Nicholls (1998), Møller and Skare (2001), Kluszczyński *et al.* (2007), Van Lieshout (2013), Kieu *et al.* (2013).

Random fields: Example - Lattice gas

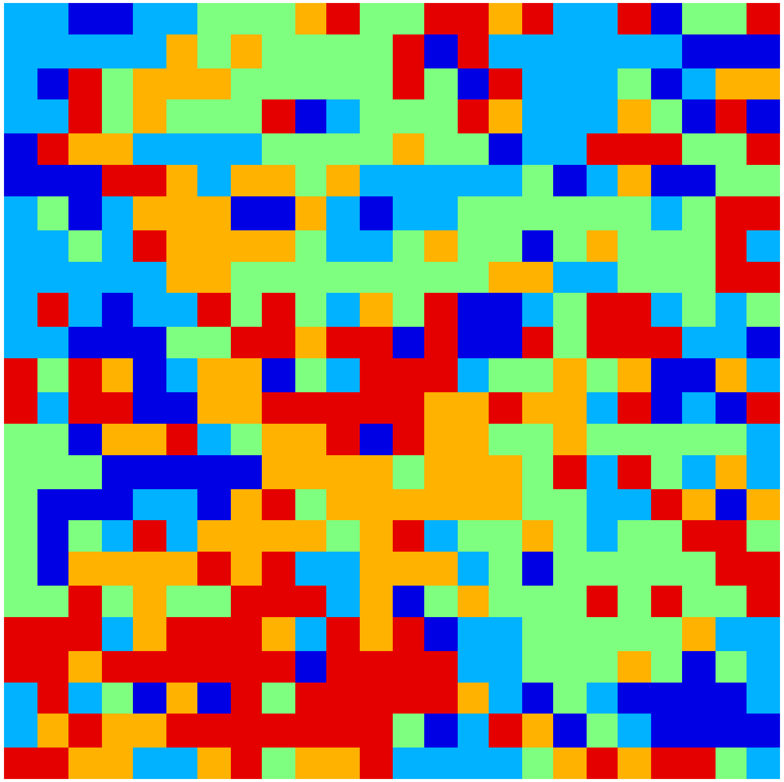
Non-overlapping black and white patches against a grey background.



Realisation of a lattice gas model. Black represents colour label '1', grey '0', and white '2'.

Random fields: Example - Potts

Patches of five different colours, no specific background.



Realisation of a Potts model with five labels.

Random fields: Definition

Pixel array $S = (s_1, \ldots, s_m)$.

Label set Λ (categorical or related to intensity values).

A **random field on S with values in L** is a random vector

$$X = (X_1, \ldots, X_m),$$

X_i being the label in Λ assigned to pixel s_i.

The distribution of X is given by the joint pdf

$$\mathbf{P}\{X_1 = x_1, \ldots, X_m = x_m\}; \quad x = (x_1, \ldots, x_m) \in \Lambda^S.$$

CWI

Markov random field

Let $\sim$ be a symmetric, reflexive relation on S.

The random field X is said to be **Markov with respect to** $\sim$ if for all $i = 1, \ldots, m$ the conditional distribution

$$\mathbf{P}\{X_i = x_i \mid X_j = x_j, j \neq i\} = \mathbf{P}\{X_i = x_i \mid X_j = x_j, s_i \sim s_j, j \neq i\}$$

depends only on x_i and the labels at those pixels s_j that share an edge with s_i, provided $\mathbf{P}\{X_j = x_j, j \neq i\} > 0$.

Besag, 1974.

Note: Markov property has considerable computational advantages.

Potts model (Ising, 1924; Potts, 1951)

Set $\Lambda = \{1, \ldots, L\}$, $\beta > 0$, and

$$\mathbf{P}\{X_1 = x_1, \ldots, X_m = x_m\} \propto \prod_{s_i \sim s_j, i<j} \exp\left[-\beta \mathbf{1}\{x_i \neq x_j\}\right].$$

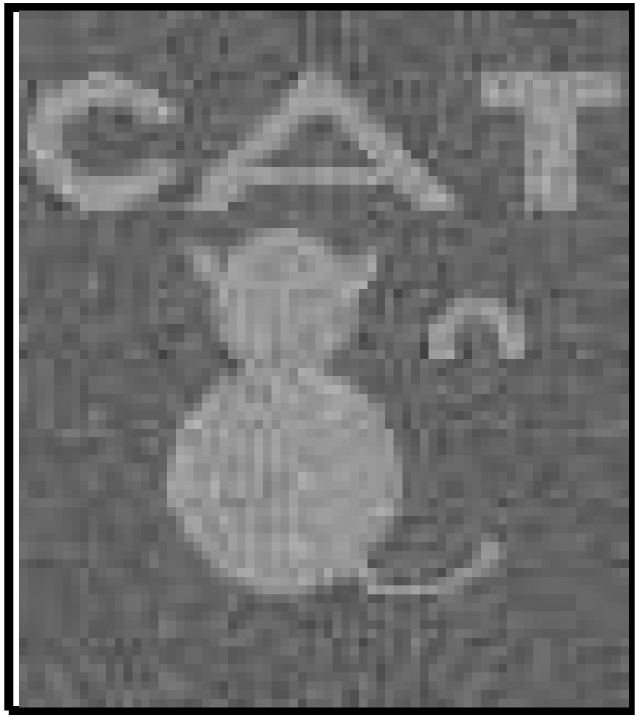
Then

$$\mathbf{P}\{X_i = x_i \mid X_j = x_j, j \neq i\} = \frac{\exp\left[-\beta \sum_{s_j \sim s_i} \mathbf{1}\{x_i \neq x_j\}\right]}{\sum_{l \in \Lambda} \exp\left[-\beta \sum_{s_j \sim s_i, j \neq i} \mathbf{1}\{l \neq x_j\}\right]}.$$

Note:

- $\beta < 0$: positive association;
- $\beta = 0$: independent labelling.

Image segmentation (analysing dirty pictures)



Model: Given the 'true' image $x = (x_1, \ldots, x_m)$, the observed pixel values $y_i \in \mathbb{R}$ are i.i.d. with pdf $g(y_i|x_i)$.

Goal: Reconstruct x from $y = (y_1, \ldots, y_m)$.

Nosebleed approach: MLE

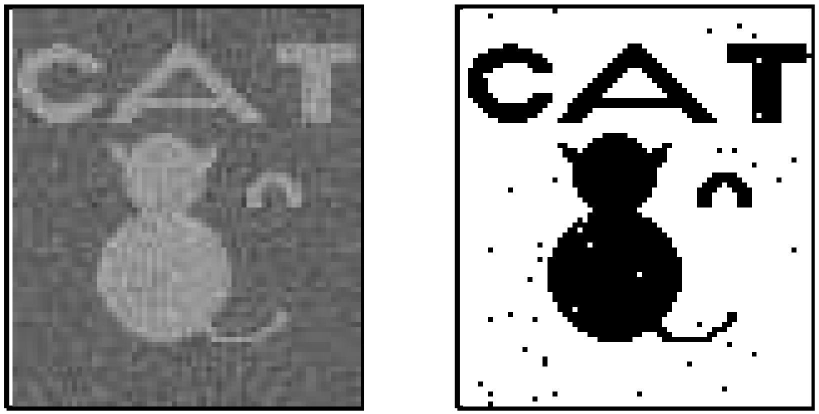
The likelihood reads

$$f(y|x) = \prod_{i=1}^{m} g(y_i|x_i)$$

and hence

$$\hat{x}_i = \text{argmax}\{g(y_i|x_i) : x_i \in \Lambda\}.$$

For Gaussian white noise, $\hat{x}_i$ is the label closest to y_i.



Conclusion: ignoring spatial coherence leads to rough solutions that are sensitive to noise.

Image segmentation - Hierarchical approach

Idea: Use a Markov prior π_X to regularise the solutions towards smoother, more spatially coherent ones.

Besag, 1986; Geman and Geman, 1984.

The posterior pdf

$$f(x|y) \propto f(y|x)\pi_X(x) = \pi_X(x) \prod_{i=1}^{m} g(y_i|x_i)$$

leads to the MAP classifier

$$\begin{aligned} \hat{x} &= \text{argmax}\{f(y|x)\pi_X(x) : x \in \Lambda^S\} \\ &= \text{argmax}\{\log f(y|x) + \log \pi_X(x) : x \in \Lambda^S\}. \end{aligned}$$

Interpretation:

- The term $\log f(y|x)$ ensures goodness of fit to data;
- $\log \pi_X$ for a Potts or lattice gas model favours spatial coherence.

Iterated Conditional Modes (Besag, 1986)

1. Start with a reasonable initial guess, e.g. the MLE.
2. Scan the pixel grid in an arbitrary prefixed order. At pixel i, update its value taking the values of its neighbours into account:

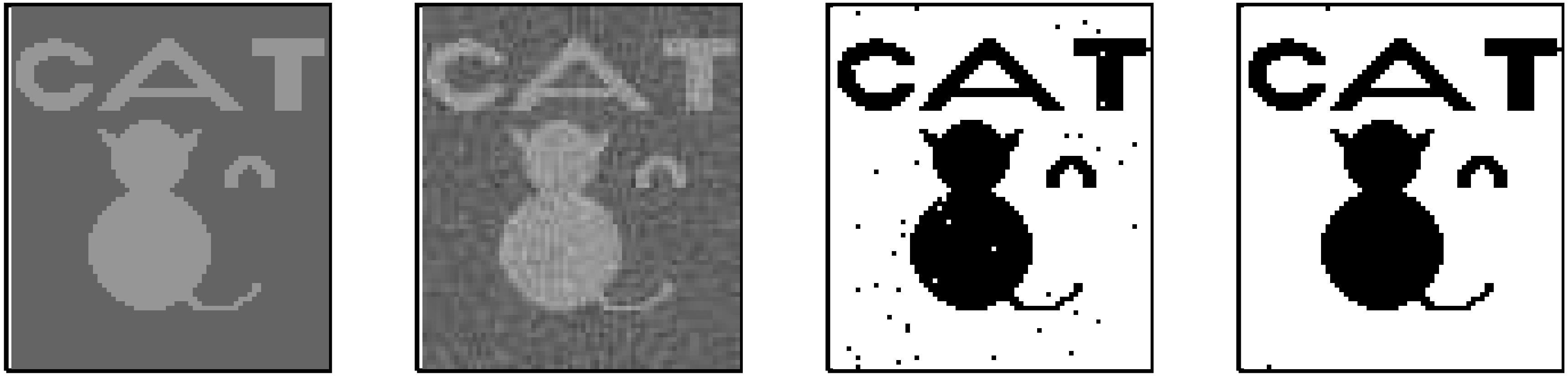
$$\tilde{x}_i = \text{argmax}\, \{g(y_i|x_i)\pi_X(x_i|x_j, j \neq i) : x_i \in \Lambda\}.$$

3. Repeat step 2 until convergence or for a predetermined number of scans.

Properties:

- ICM converges to a local maximum of the posterior;
- typically in very few cycles;
- for a Markov prior, the calculations are local, hence quick;
- since the prior cannot be expected to reflect the global data appearance, ICM tends to look better than the global optimum.

Results



From left to right: Truth, distortion by white noise ($\sigma = 10$), MLE and MAP classifiers (Potts with $\beta = 1.5$).

- For $\beta = 0$, MAP and MLE agree.
- For $\beta \to \infty$, MAP results in a single colour image, ICM carries out a recursive majority vote (data used to break ties).

Remarks and extensions

In case of unknown parameters, the joint distribution is of the form

> forward model(data | process, parameters) × prior(process | parameters),

optionally complemented by a hyper prior distribution on the model parameters.

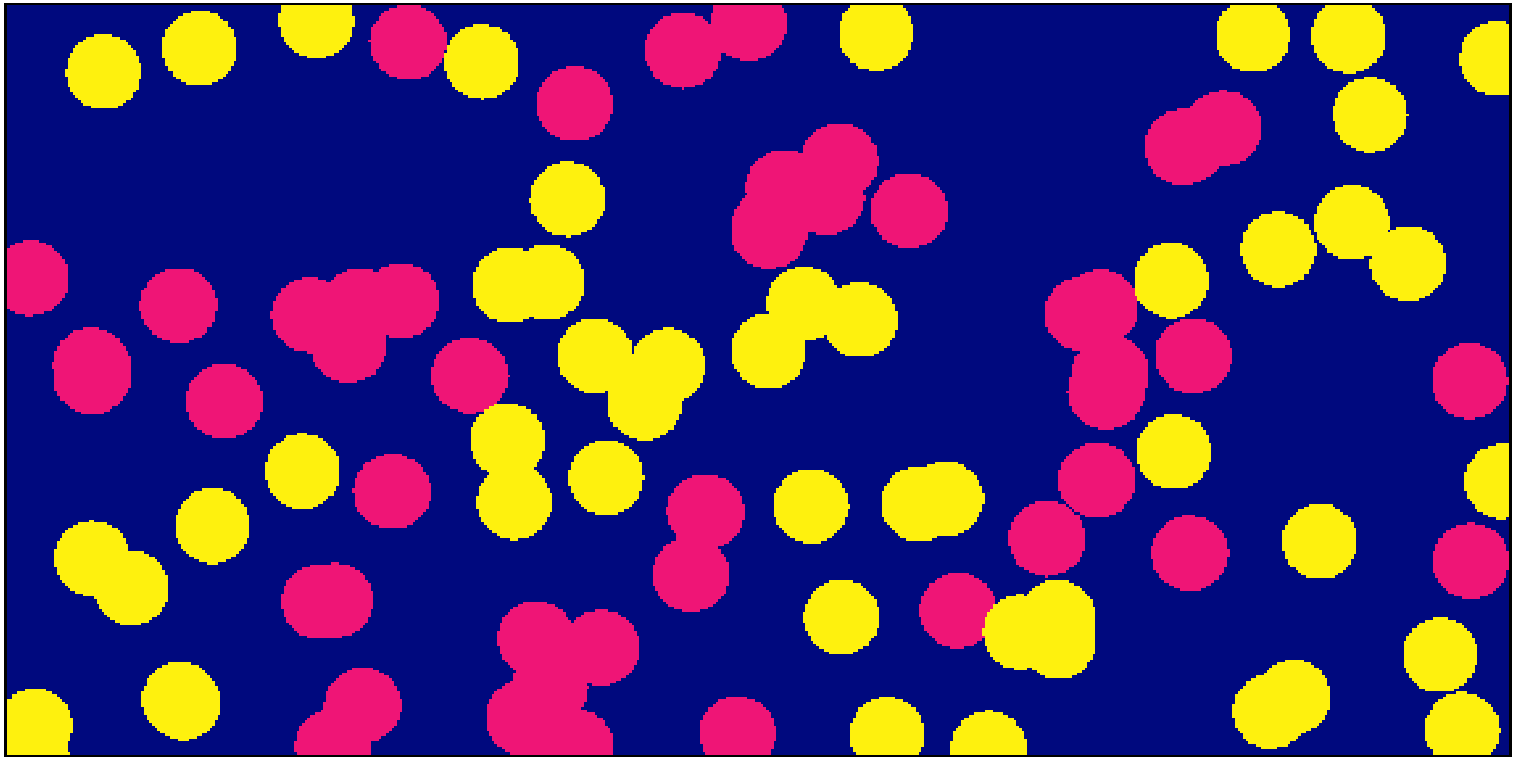
This framework is **extremely flexible**. Inference is based on the **posterior distribution** of the process and/or the parameters conditional on the observations, e.g.

- by Monte Carlo methods;
- a point estimate or optimal reconstruction of the process;
- histograms of any marginal of interest.

Banerjee, Carlin and Gelfand, 2015.

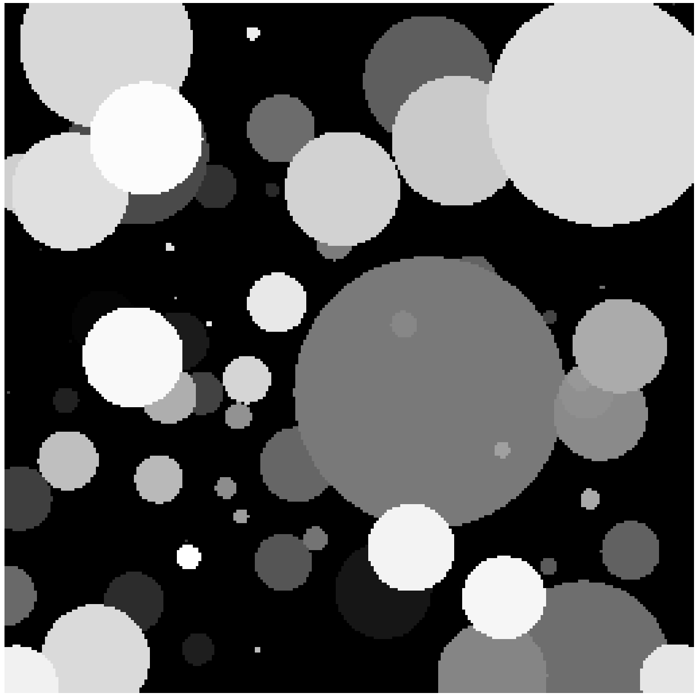
Object processes: Example - Penetrable spheres

Particles of different colour do not overlap.



Object processes: Example - Strauss

Light balls tend to avoid being centred in darker ones.



Realisation of a Strauss density

$$\exp\left[\sum_i \left(\log(\beta) + \log(\gamma) \sum_{j<i} \mathbf{1}\{||d_i - d_j|| \leq m_j\}\right)\right], \quad \beta > 0, \gamma \in [0,1].$$

Object processes: Definition

Bounded, open set $\emptyset \neq D \subseteq \mathbb{R}^2$ with border ∂D.

Polish space Q for object attributes equipped with probability distribution $\mathbb{Q}$.

Examples:

- simple geometric shapes (Baddeley and Van Lieshout (1992), Van Lieshout (1994, 1995));
- deformable templates (Amit *et al.* (1991), Hansen *et al.* (2002), Hurn (1998), Mardia *et al.* (1997), Pievatolo and Green (1998), Rue and Hurn (1999), Rue and Husby (1998));
- ensembles of simple shapes (Lacoste *et al.* (2005), Ortner *et al.* (2007)).

Point process distribution

Realisations $\mathbf{x} = \{x_1, \ldots, x_n\}$, where $n \in \mathbb{N}_0$ and $x_i = (d_i, m_i) \in \bar{D} \times Q$, $i = 1, \ldots, n$.

The distribution of X is given by its pdf f:

- the probability of n points is

$$\frac{e^{-\ell(D)}}{n!} \int_{\bar{D}\times Q} \cdots \int_{\bar{D}\times Q} f(\{x_1, \ldots, x_n\})\, d\ell \times \mathbb{Q}(x_1) \cdots d\ell \times \mathbb{Q}(x_n);$$

- conditionally on having n points, they follow pdf (w.r.t. $(\ell \times \mathbb{Q})^n$)

$$\frac{f(\{x_1, \ldots, x_n\})}{\int_{\bar{D}\times Q} \cdots \int_{\bar{D}\times Q} f(\{z_1, \ldots, z_n\})\, d\ell \times \mathbb{Q}(z_1) \cdots d\ell \times \mathbb{Q}(z_n)},$$

writing ℓ for Lebesgue measure.

Markov object process

Let $\sim$ be a symmetric, reflexive relation on $\bar{D} \times Q$.

The object process X is said to be **Markov with respect to** $\sim$ if

- f is hereditary: $f(\mathbf{x}) > 0$ implies $f(\mathbf{y}) > 0$ for all $\mathbf{y} \subseteq \mathbf{x}$;
- for all $u \in (\bar{D} \times Q) \setminus \mathbf{x}$, the conditional intensity

$$\lambda(u \mid \mathbf{x}) := \frac{f(\mathbf{x} \cup \{u\})}{f(\mathbf{x})},$$

depends only on u and $\{x_i \in \mathbf{x} \setminus \{u\} : u \sim x_i\}$, provided $f(\mathbf{x}) > 0$.

Ripley and Kelly, 1977.

Note: Markov property has considerable computational advantages.

Penetrable spheres (Widom and Rowlinson, 1970)

Set $Q = \{1, 2\}$, $\mathbb{Q}(1) = \mathbb{Q}(2) = 1/2$, $\beta > 0$, and consider

$$f(\mathbf{x}_1 \cup \mathbf{x}_2) \propto (2\beta)^{n(\mathbf{x}_1)+n(\mathbf{x}_2)} \, \mathbf{1}\{d(\mathbf{x}_1, \mathbf{x}_2) > R\}.$$

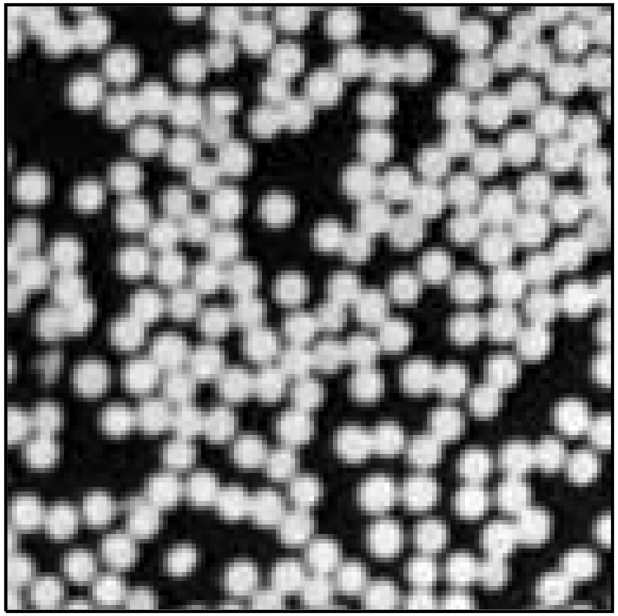
Then, provided $f(\mathbf{x}_1 \cup \mathbf{x}_2) > 0$, e.g.

$$\lambda((u, 1)|\mathbf{x}_1, \mathbf{x}_2) = 2\beta \, \mathbf{1}\{d((u, 1), \mathbf{x}_2) > R\}.$$

Notes:

- Assign points of Poisson(2β)-process i.i.d. to each type wp $1/2$, conditionally on respecting **hard core** distance R;
- The marginal distributions of X_i, $i = 1, 2$, are **area-interaction** processes (Baddeley and Van Lieshout (1995), Häggström *et al.* (1999)).

Object recognition



Forward model:

- object configuration x determines 'true' image $\theta_i^{(x)}$, $s_i \in S$;
- given x, the observed pixel values $y_i \in \mathbb{R}$ are i.i.d. with pdf $g(y_i|\theta_i^{(x)})$.

Goal: Reconstruct x from $y = (y_1, \ldots, y_m)$.

CWI

Object recognition - Hierarchical approach

Idea: Use a Markov overlapping object prior π_X to discourage multiple response, e.g. the **Strauss** pdf

$$\pi_X(x) \propto \beta^{n(x)} \gamma^{r(x)}$$

for $\beta > 0$, $\gamma \in [0, 1]$, $n(\cdot)$ cardinality, $r(\cdot)$ number of overlapping pairs.

Molina and Ripley, 1989; Baddeley and Van Lieshout, 1992.

The posterior pdf

$$f(x|y) \propto f(y|x)\pi_X(x) = \pi_X(x) \prod_{i=1}^{m} g(y_i|\theta_i^{(x)})$$

leads to the MAP classifier

$$\hat{x} = \text{argmax}\{\log f(y|x) + \log \pi_X(x)\}.$$

Steepest ascent algorithm

1. Start with a reasonable initial guess $x^{(0)}$, e.g. the MLE.
2. Given $x^{(k-1)}$, determine

$$a = \max_{u \in \bar{D} \times Q} \left\{ \log \frac{f(y|x^{(k-1)} \cup \{u\}) \pi_X(x^{(k-1)} \cup \{u\})}{f(y|x^{(k-1)}) \pi_X(x^{(k-1)})} \right\}$$

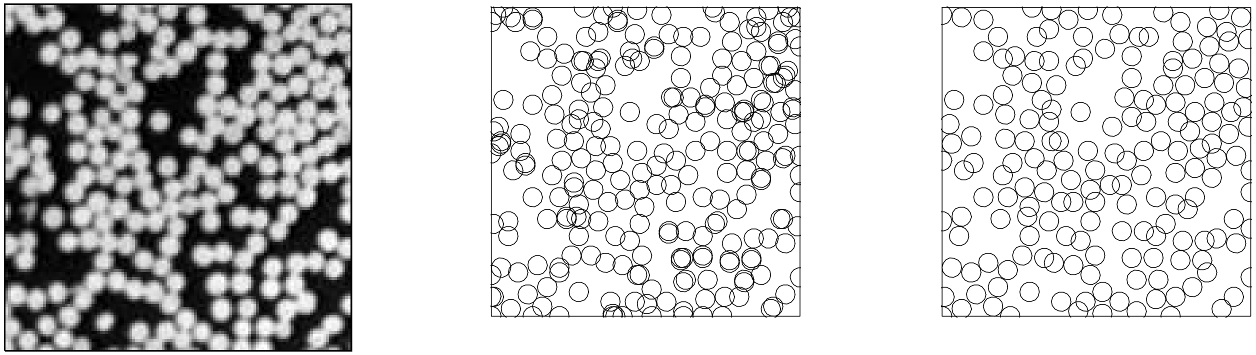
   and

$$b = \max_{x_i \in x^{(k-1)}} \left\{ \log \frac{f(y|x^{(k-1)} \setminus \{x_i\}) \pi_X(x^{(k-1)} \setminus \{x_i\})}{f(y|x^{(k-1)}) \pi_X(x^{(k-1)})} \right\}.$$

3. If $\max\{a, b\} \geq w$, implement the best change to get $x^{(k)}$.
4. Repeat step 2 until convergence.

Baddeley and Van Lieshout, 1992.

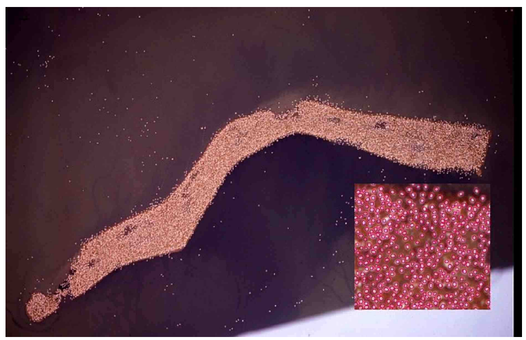
Results: counting pellets



Model: discs (radius 4), blurred by 3×3 linear filter (weights 4, 2, 1) and distorted by white noise ($\sigma^2 = 83.1$).

From left to right: Data, MLE and MAP classifiers ($w = 0$, Strauss prior with $\log \beta = \log \gamma = -1000$).

Flamingo's (INRIA, equipe ARIANA)



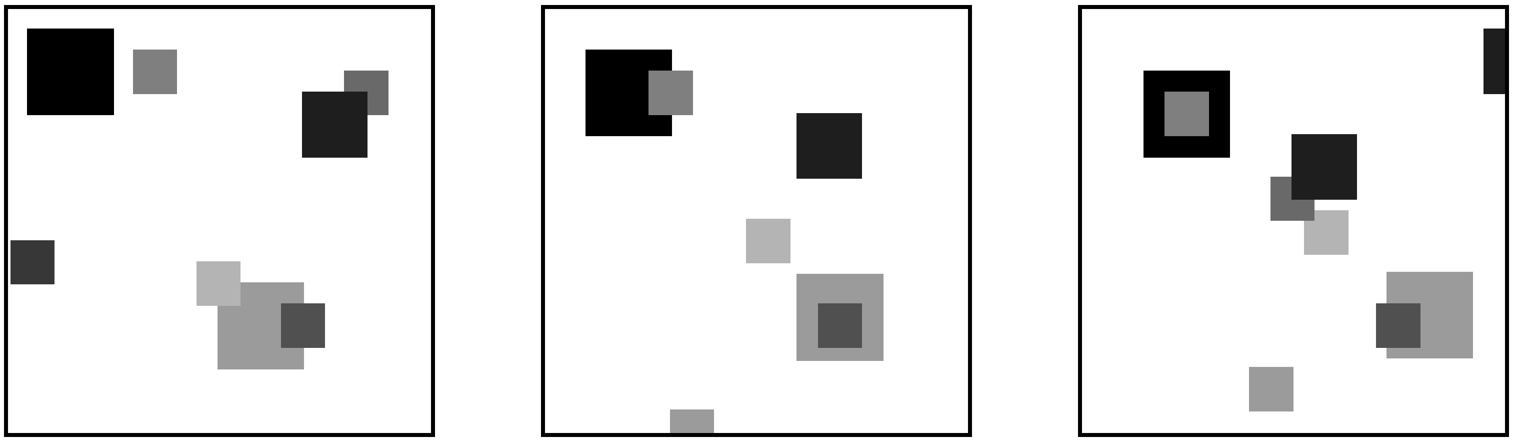
Remarks and extensions

- Markov object processes cannot model depth;
- nor non-symmetric neighbour relations.

In such cases, use **finite sequential spatial processes** (Van Lieshout 2006a, 2006b) with realisations

$$\vec{\mathbf{x}} = (x_1, \ldots, x_n), \quad x_i \in \bar{D} \times Q$$

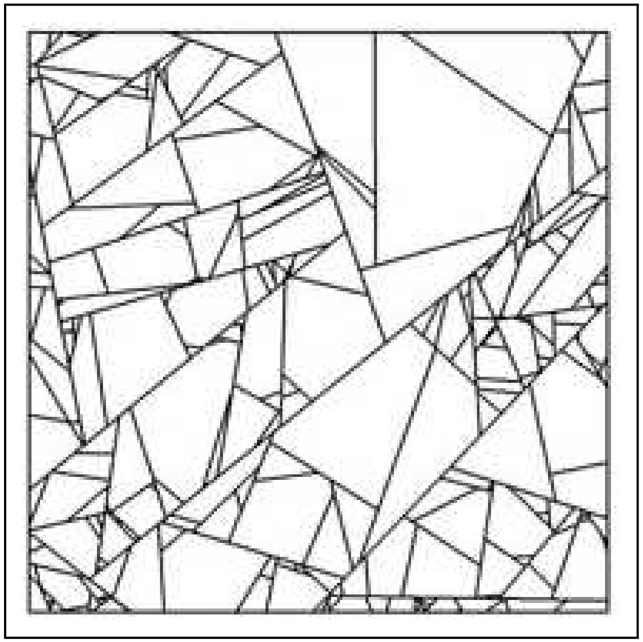
and drop the symmetry requirement on $\sim$. Useful in motion tracking.

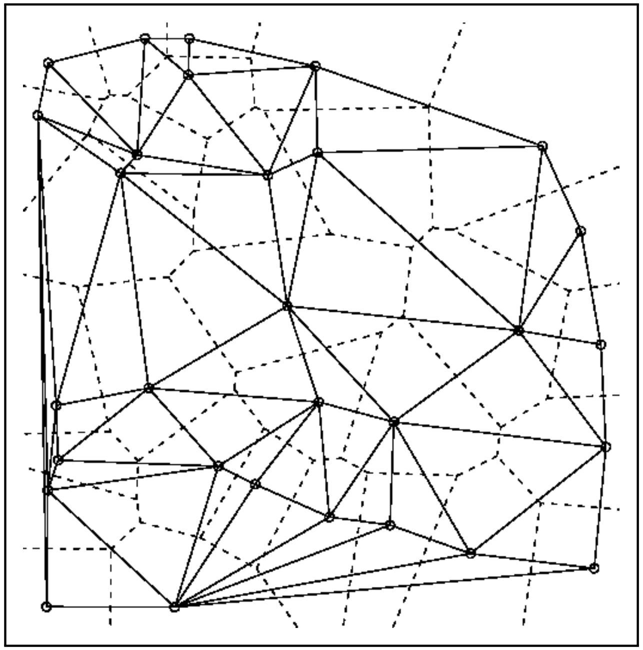


Intermediate level modelling

Idea: Regard an image scene as a (coloured) tessellation. Thus,

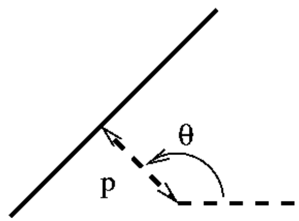
- global aspects of the image are captured;
- no need to model all objects in the image.

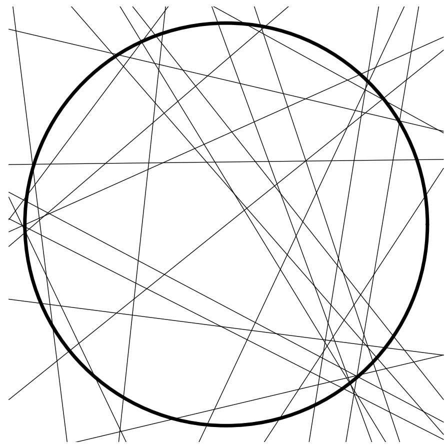




Left: STIT (Nagel and Weiss, 2003); Right: Voronoi and Delaunay.

Poisson line tessellation: Definition





A Poisson point process has intensity $\lambda > 0$ w.r.t. $dp d\theta$. Equivalently, the heads of the perpendiculars form a Poisson point process in $\mathbb{R}^2$ with intensity function

$$\lambda / ||(x, y)||.$$

Poisson line tessellation - Properties

The Poisson line process is

- **well-defined:** Any bounded set B is hit by finitely many lines;
- isotropic and consistent.
- **independent:** Conditionally on n lines hitting B, they are i.i.d.

Line transects: The intersection with a fixed line ℓ

- is a Poisson point process on l with rate 2λ,
- the intersection angles are i.i.d. with pdf

$$\sin\theta/2, \quad \theta \in [0, \pi).$$

George, 1987.

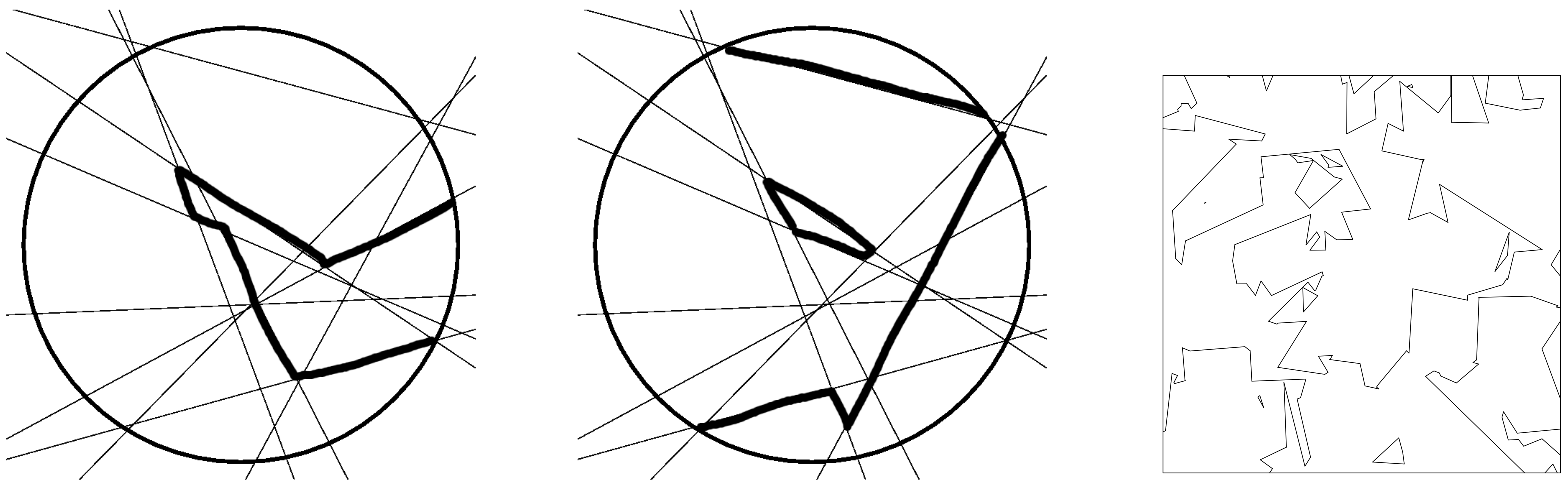
CWI

Polygonal Markov field: Definition

Note: Many line tessellations yield **convex** polygonal cells.

Idea: To obtain other shapes, draw polygonal contours γ on a Poisson line process **using each line only once** (hence no X- or T-shapes) and give weight proportional to

$$\exp\left[-2\ \text{length}(\gamma)\right], \qquad \text{Arak, 1982.}$$



Polygonal Markov field: Properties

The Arak model $\mathcal{A}_D$ on $D \subset \mathbb{R}^2$ is

- isotropic;
- consistent: for $D' \subseteq D$, $\mathcal{A}_{\Phi_D} \cap D' =^d \mathcal{A}_{\Phi_{D'}}$;
- admits two equally likely colourings such that no adjacent regions share the same colour;
- there is a dynamic representation that allows for easy simulation.

Line transects: The intersection with a fixed line ℓ is a Poisson point process on l with rate 2λ and

- the two feasible colourings are equally likely,
- the intersection angles are i.i.d. with pdf $\sin\theta/2, \quad \theta \in [0, \pi)$.

Markov: the conditional behaviour in an open bounded domain depends on the exterior configuration only through arbitrarily close neighbourhoods of the boundary.

Arak and Surgailis, 1989.

CWI

Foreground/background segmentation

Hierarchical approach: regularise towards short lengths. Then the MAP classifier is

$$\hat{\gamma} = \text{argmax} \{-\beta \, \text{length}(\gamma) - \sum_{i=1}^{m} |y_i - \theta(\gamma)_i|\}$$

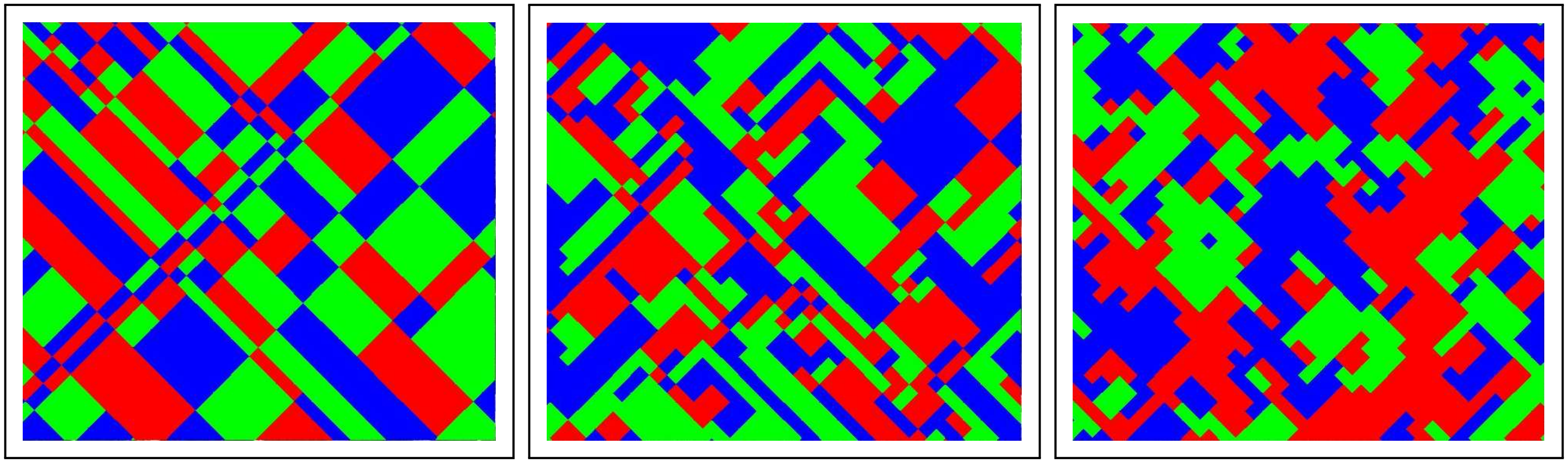
where $\theta(\gamma)$ is the segmentation image defined by γ.

Kluszczyński, Van Lieshout and Schreiber, 2007.

Remarks and extensions

- More than two colours;
- Draw polygonal contours on **fixed** finite collection of lines $\mathcal{T}$ (Schreiber and Van Lieshout, 2010; Van Lieshout, 2013);

Note: The discrete Arak mosaic is dual to a Markov random field.



Realisations with three colours. Left: no V-junctions; Right: no X-junctions.

Network extraction

Goal: Reconstruct the network of tracks that run between adjacent fields.

Network extraction - Hierarchical approach

Reference model: Arak mosaic with four colours.

Regularised goodness of fit:

$$\beta \sum_{e \in E(\gamma)} \left[f(e) - c(e) \right],$$

with $\beta > 0$, and

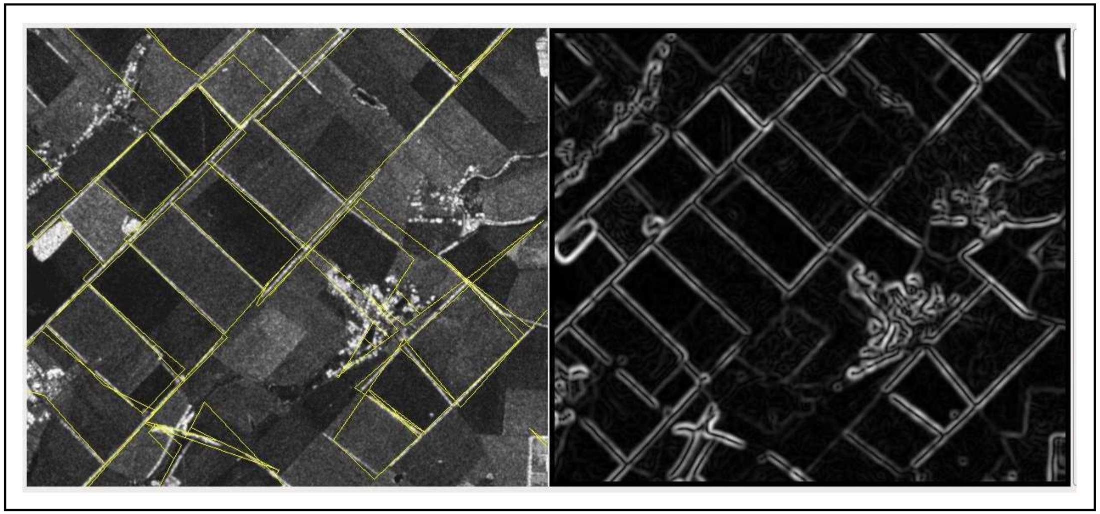
- $f(e)$: integrated absolute gradient flux along edge e:

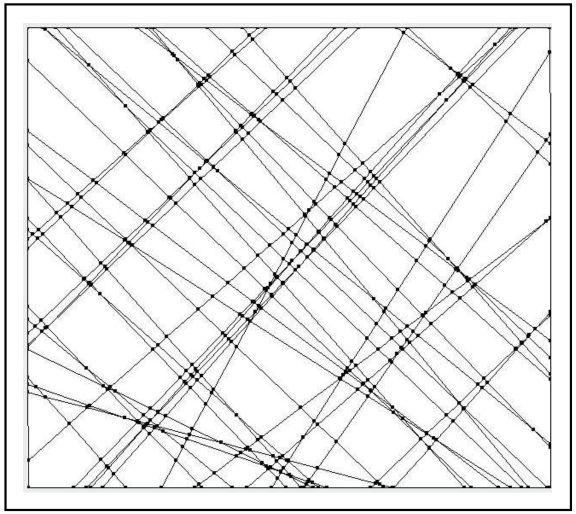
  $$f(e) = \int_e |\nabla(\vec{\mathbf{y}})_1(p) n_1(p) + \nabla(\vec{\mathbf{y}})_2(p) n_2(p)| dp$$

  where (n_1, n_2) is the unit normal to e;
- $c(e)$: twice the number of segments along the edge to discourage spurious edges.

Van Lieshout, 2013.

Results





Left: extracted network; Right: gradient image. $\mathcal{T}$: lines of big gradient.