Markov Point Process for Multiple Object Detection

Xavier Descombes

INRIA Sophia Antipolis Méditerranée

07/03/2023

Outline

Motivation

The different issues :

- objects
- reference measure
- prior
- data term
- **optimization**

Some results

Du contexte

A la géométrie

Du contexte

A la géométrie

Motivation : from context to geometry

1) Consider prior information (Bayesian approach, Markov Random Fields, interactions)

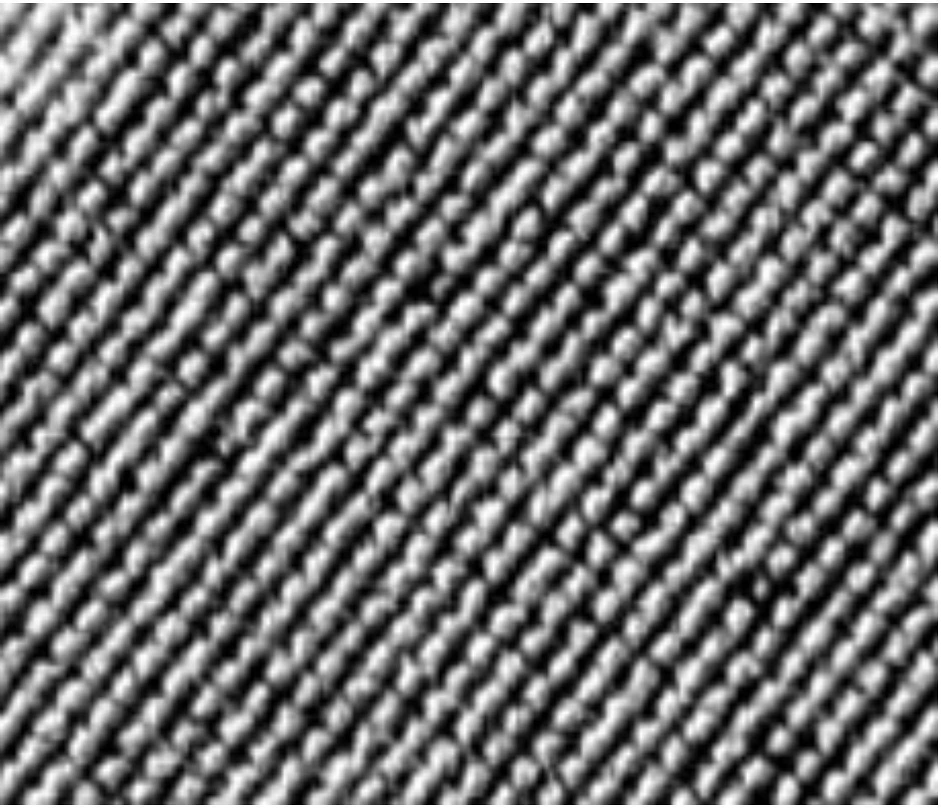
2) Embed geometric information (graph of objects)

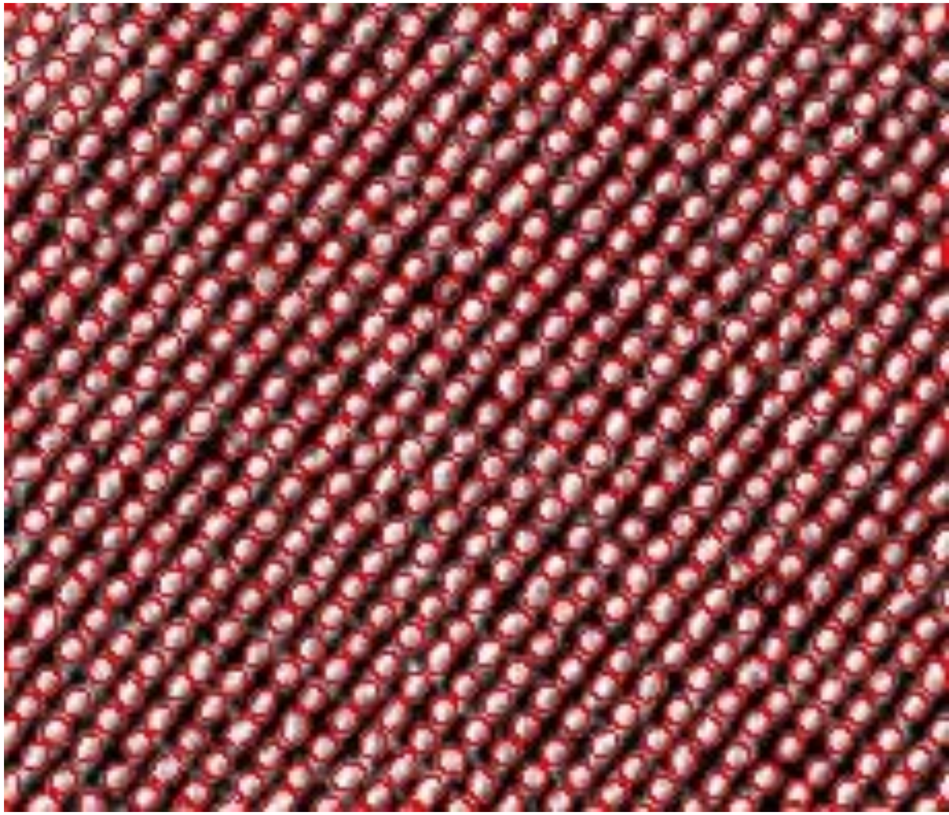
3) Modeling the scene structure (interactions between objects, unknown number of objects)

4) Need algorithms for simulating, optimizing the models

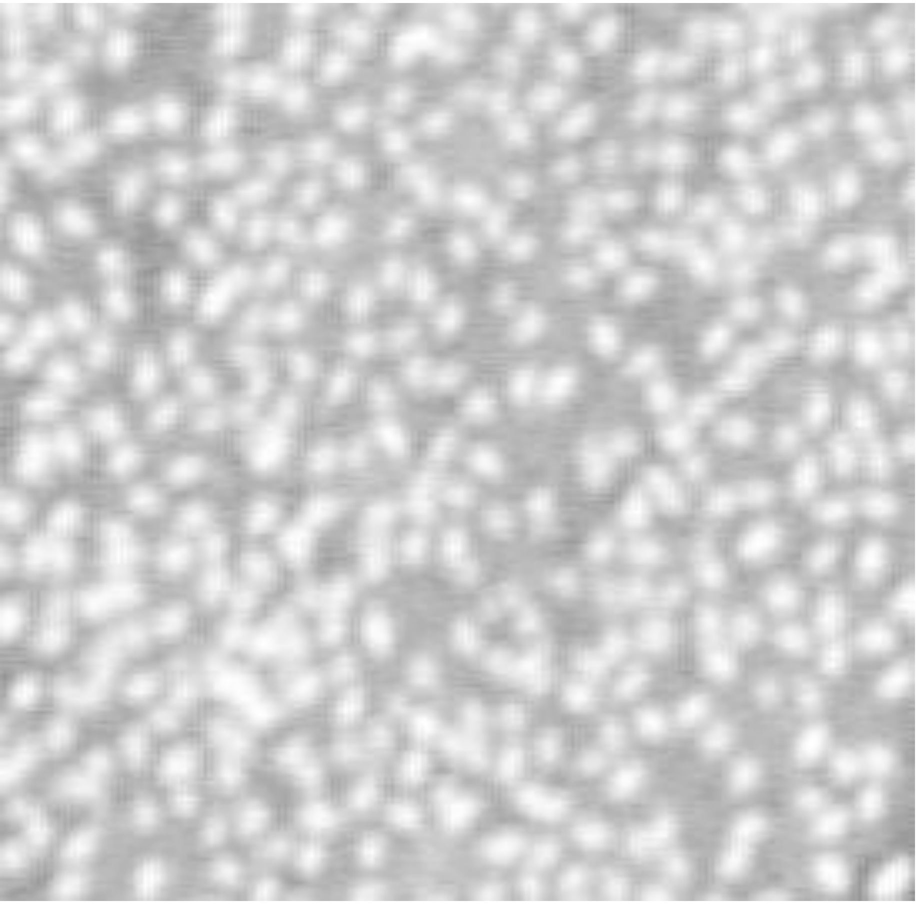
→ Marked point processes

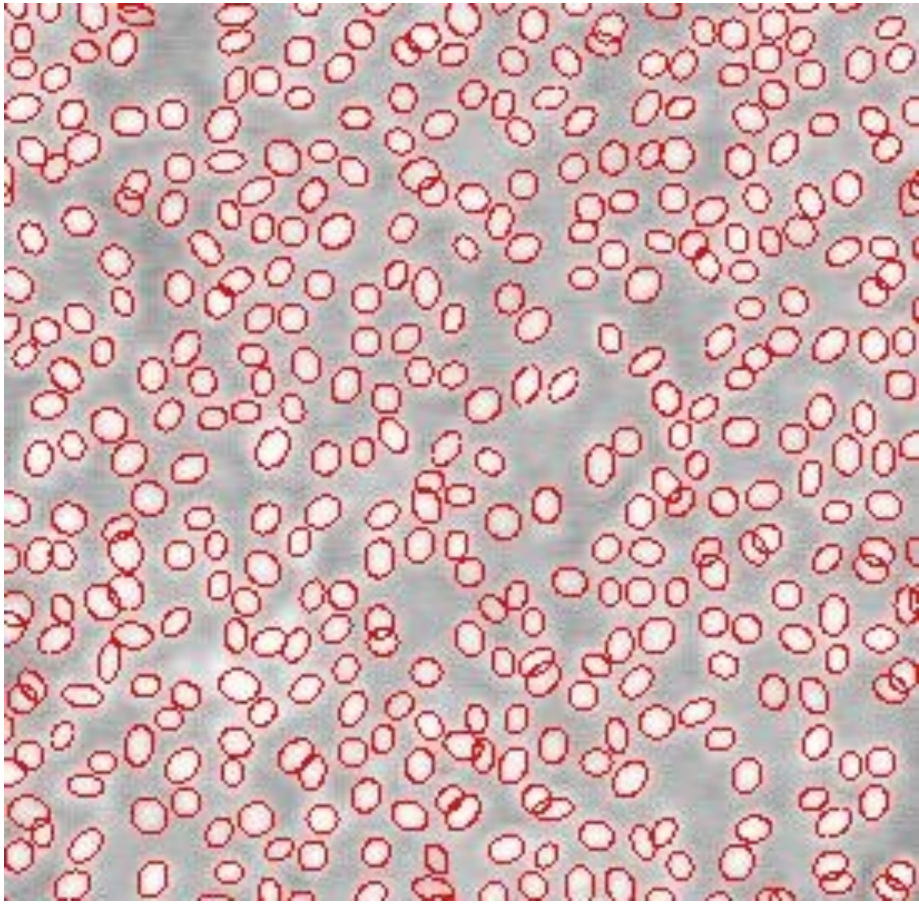
Motivation : a scene as a collection of objects



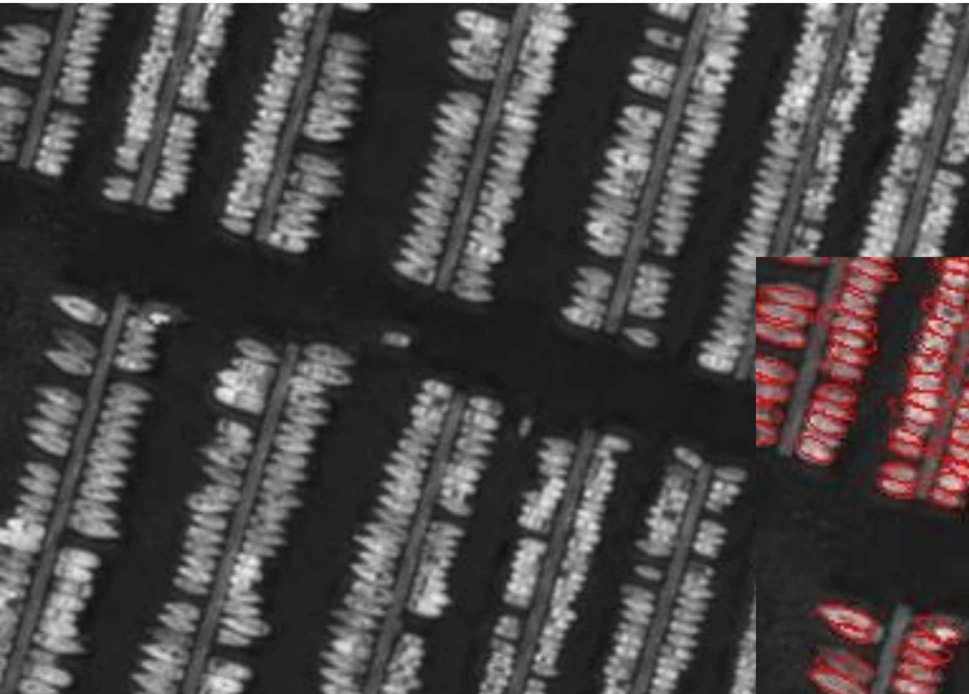


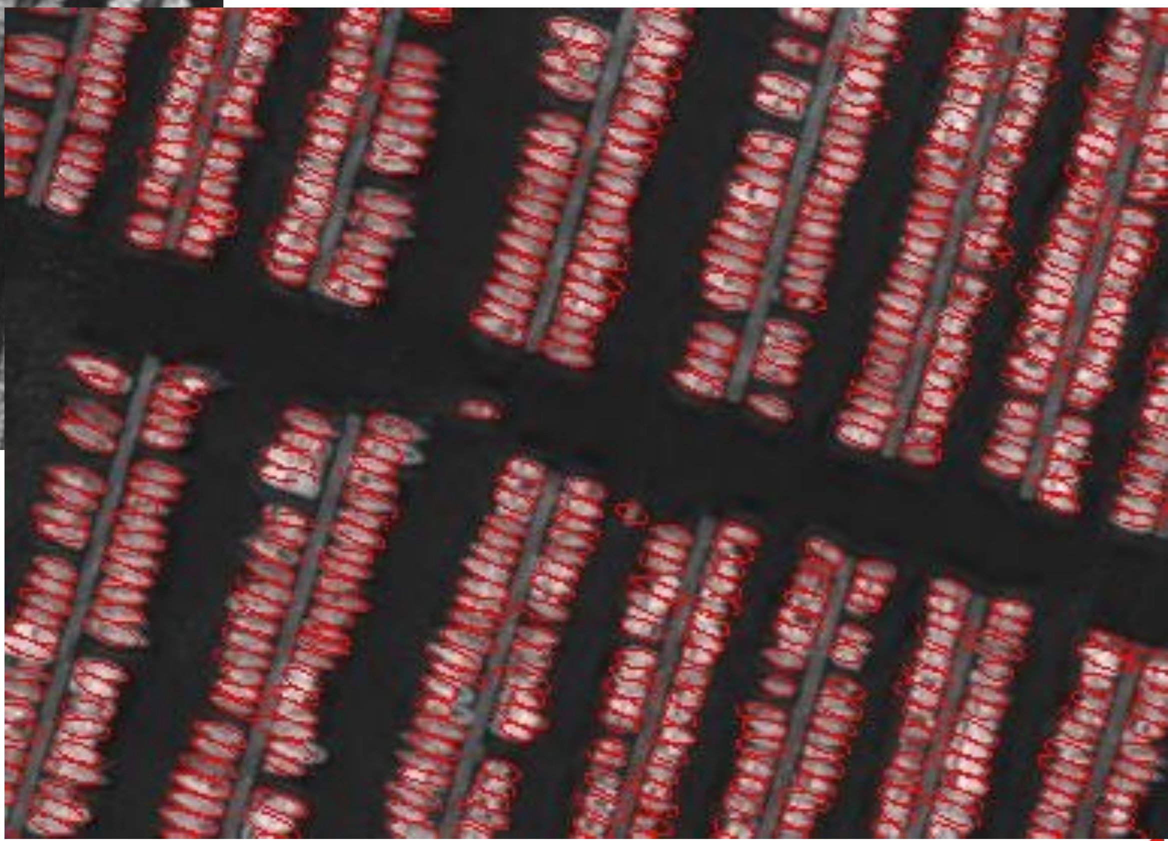
Motivation : a scene as a collection of objects



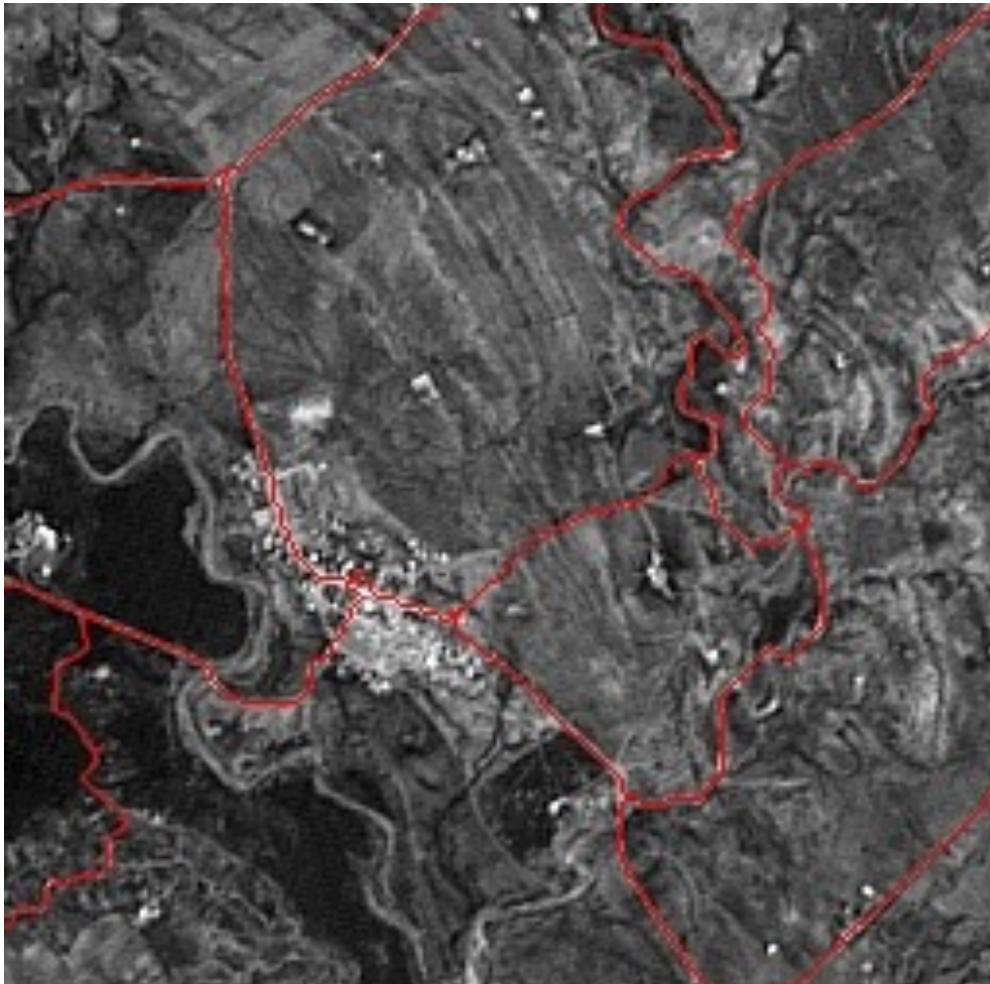


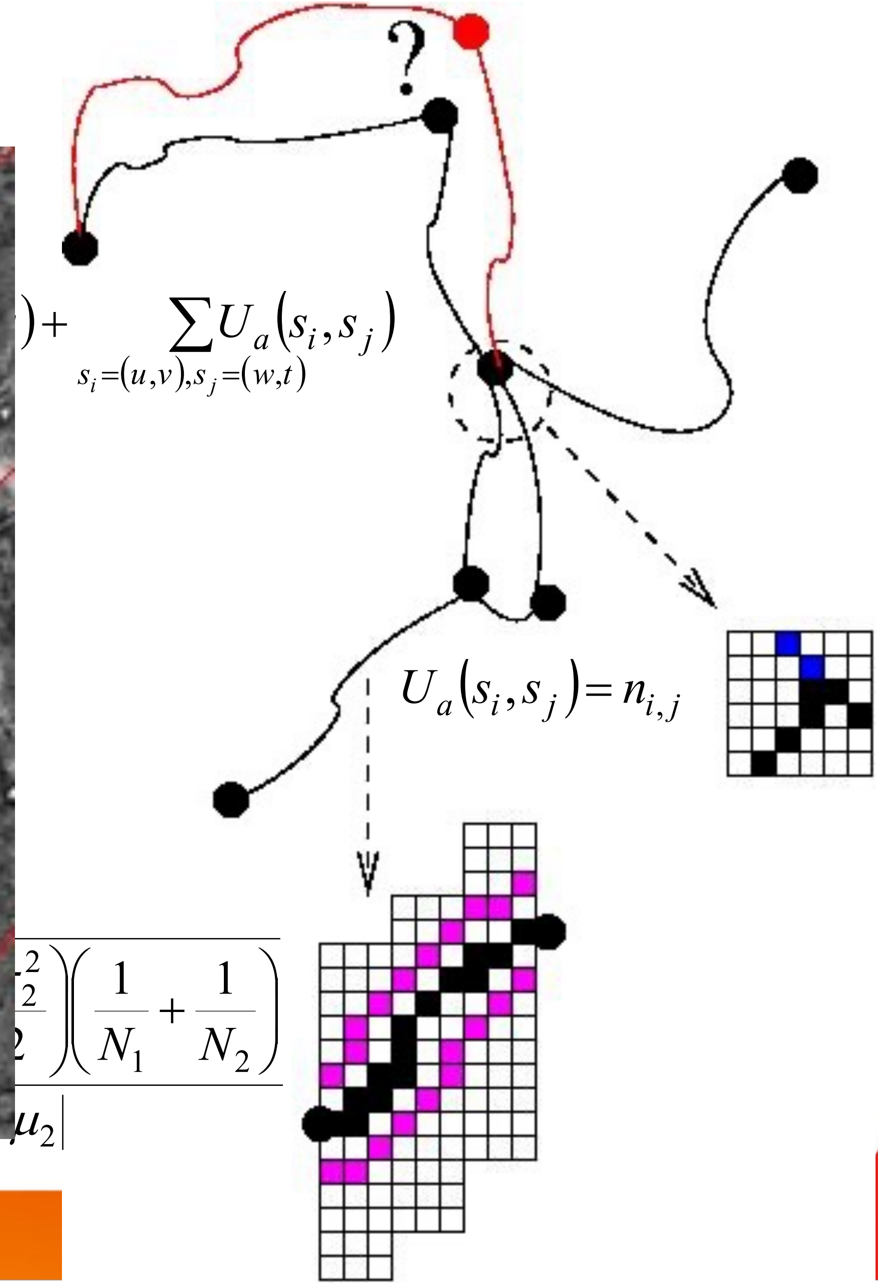
Motivation : a scene as a collection of objects



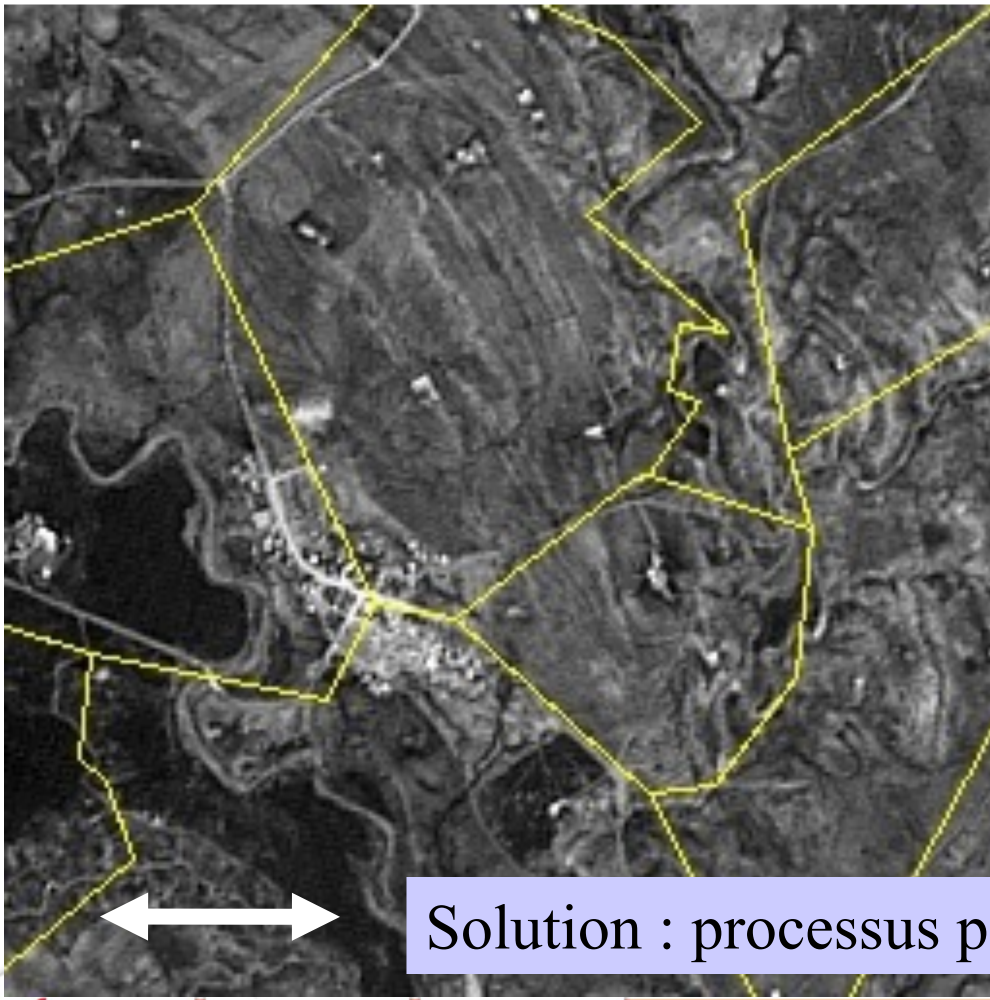


Exemple: détection de réseaux routiers





Problèmes liés à cette approche



1) Combien de noeuds ?

2) Où positionner les arêtes ?

Solution : processus ponctuels marqués

Processus de Poisson

- Un processus ponctuel X sur χ est appelé processus ponctuel de Poisson de mesure d'intensité $\nu(.)$ si :
  - $N_X(A)$ suit une loi de Poisson d'espérance $\nu(A)$ pour tout borélien borné A de χ
  - Pour k boréliens disjoints $A_1,\ldots A_k$, les variables aléatoires $N_X(A_1),\ldots,N_X(A_k)$ sont indépendantes.

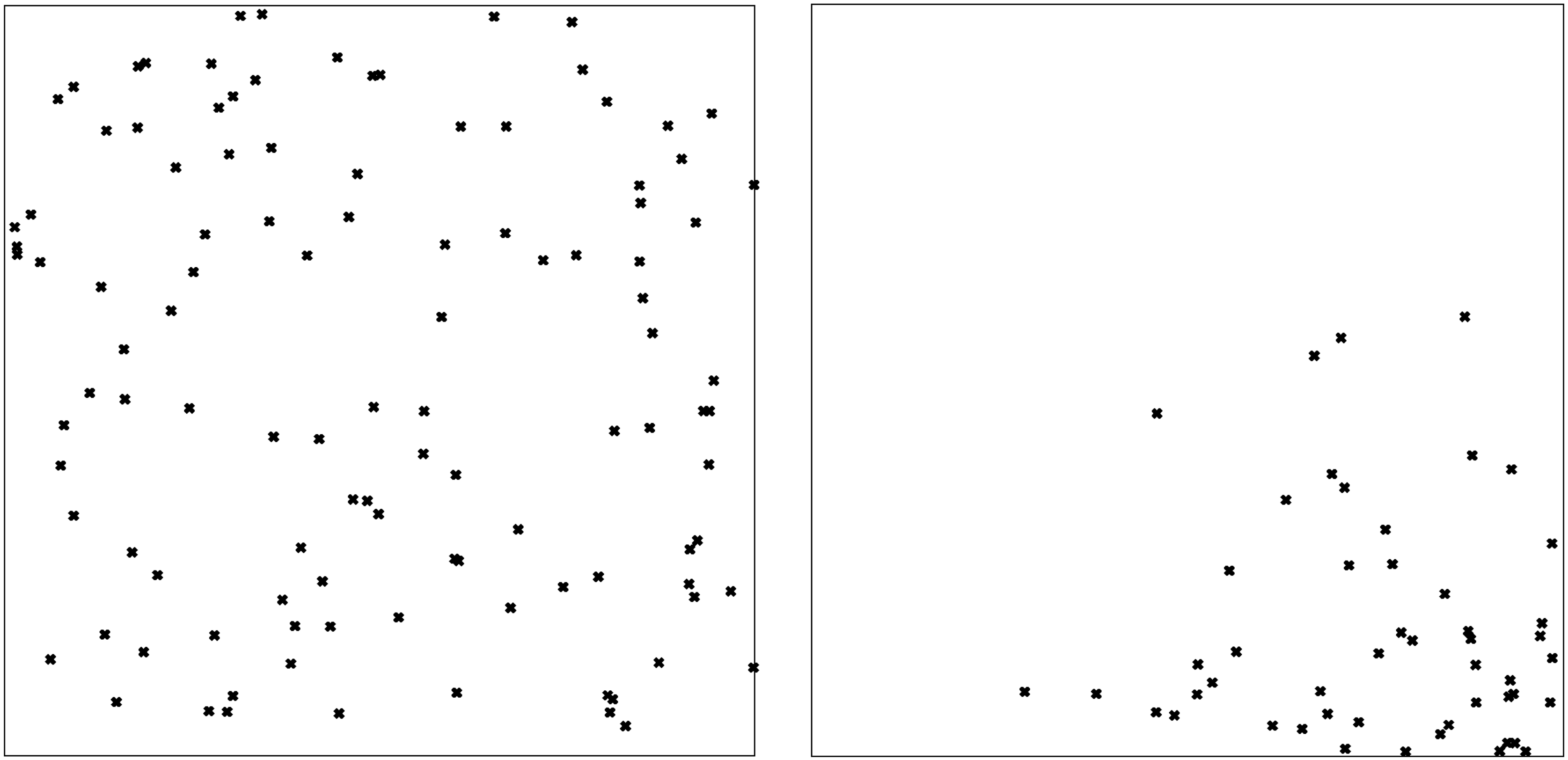
Intensité

- Processus de Poisson homogène : $\nu(.)$ proportionnelle à la mesure de Lebesgue $\Lambda(.)$
- Processus non homogène : fonction d'intensité

  $\lambda(.) > 0$ définie comme la dérivée de Radon Nikodym de $\nu(.)$ par rapport à la mesure de Lebesgue :

$$\nu(A) = \int_A \lambda(x) \Lambda(dx) < \infty$$

Simulations du processus de Poisson



Homogène vs non homogène

Processus ponctuels définis par une densité

- On définit X par sa densité de probabilité f (dérivée de Radon Nikodym) par rapport à la loi $\pi_\nu(.)$ du processus de Poisson dit de référence :

$$f : N^f \to [0,\infty[: \int_{N^f} f(x) d\pi_\nu(x) = 1$$

- On a alors :

$$p_n = \frac{e^{-\nu(\chi)}}{n!} \int_\chi \cdots \int_\chi f(\{x_1,\cdots,x_n\}) d\nu(x_1) \cdots d\nu(x_n)$$

Processus de Strauss

$$f(x) = \alpha \beta^{n(x)} \gamma^{s(x)}$$

n(x) : nombre de points de x

s(x) : nombre de paires de points en relation (nombre de cliques)

u~v si et seulement d(u,v) < R

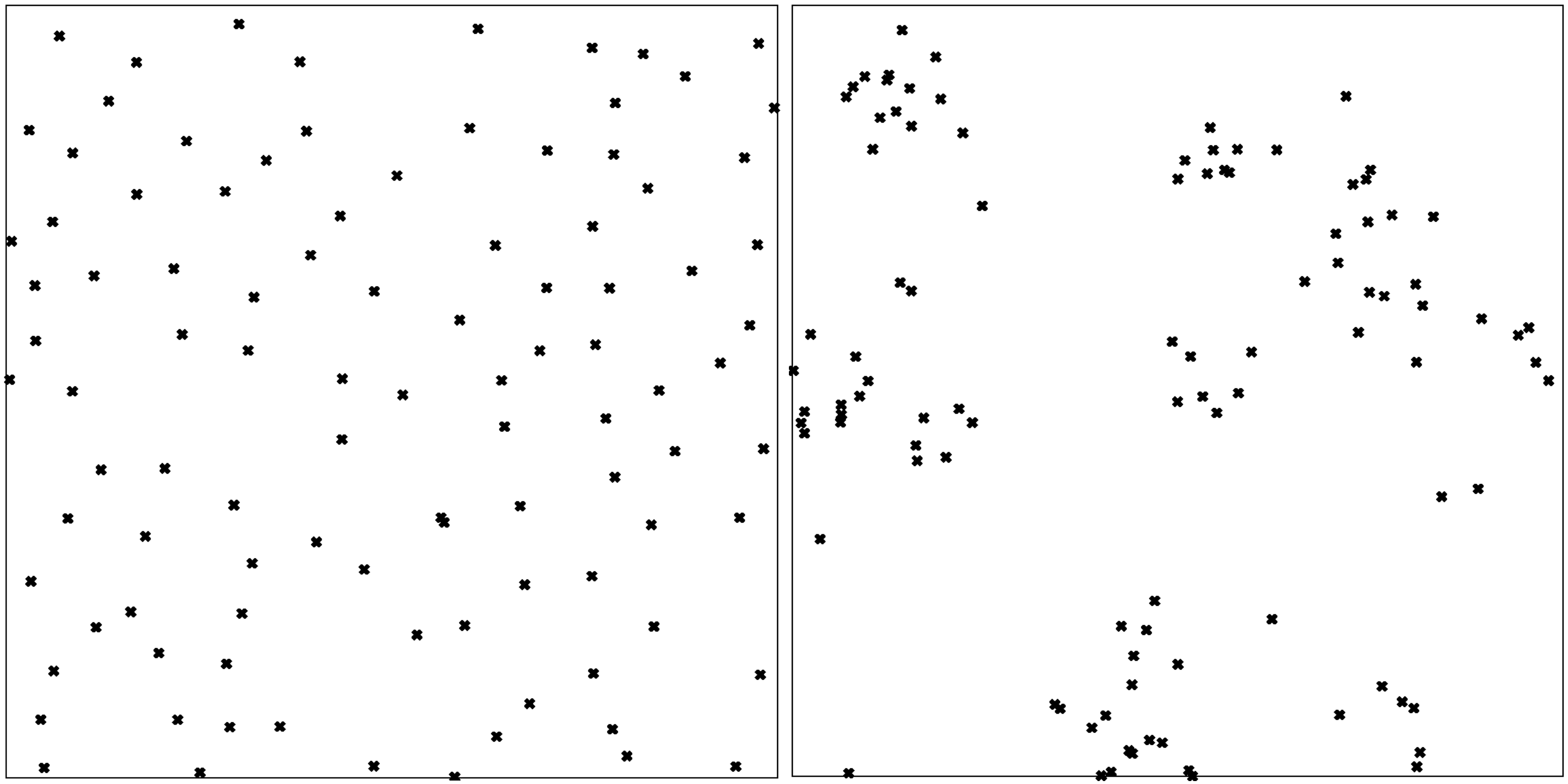
$\gamma = 1$: processus de Poisson d'intensité $\beta\lambda(.)$

$0<\gamma<1$: répulsion entre points proches

$\gamma=0$: interdiction de points proches (hard core process)

$\gamma > 1$: processus attractif ou agrégé si l'on fixe le nombre de points

Simulations du processus de Strauss



Répulsif vs Attractif

The configuration space

« Simple » parametric shapes : $S = \{ s = (x,m), x \in K, m \in M \}$

(x,r) (x,a,b,θ) (x,L,l,θ) (x,L,θ)

$\Omega_n = \{ \{ s_1,\ldots,s_n \}, s_i \in S \}$ $\Omega_0 = \varnothing$

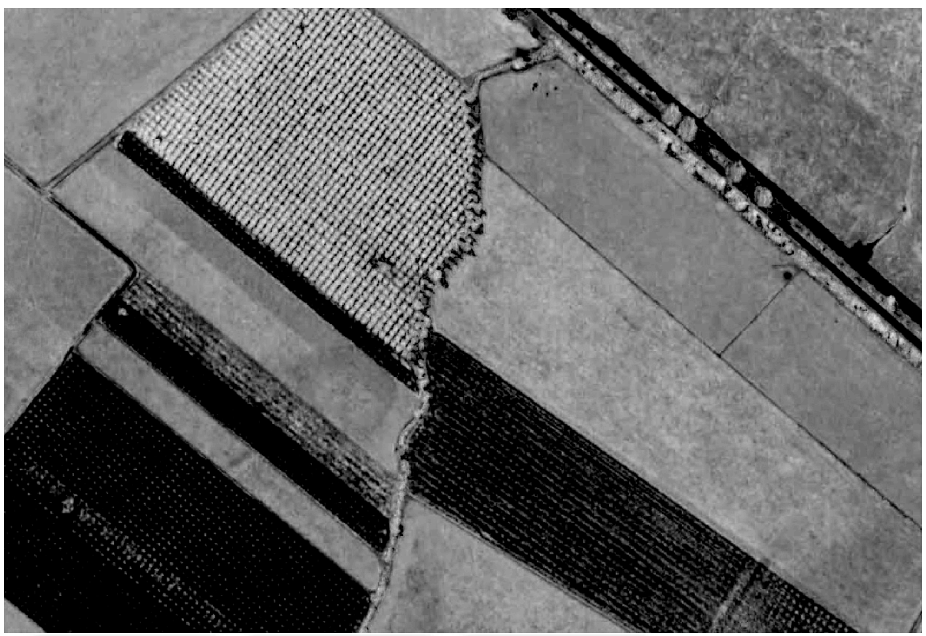
$$\Omega = \bigcup_n \Omega_n$$

The reference measure

Usually the Poisson measure :

$$\pi_\nu(B) = e^{-\nu(\chi)}\left(1_{[\emptyset\in B]} + \sum_{n=0}^{\infty}\frac{\pi_{\nu_n}(B)}{n!}\right)$$

$$\pi_{\nu_n}(B) = \int_\nu \cdots \int_\nu 1_{[\{x_1,\cdots,x_n\}\in B]}\,\nu(dx_1)\cdots\nu(dx_n)$$



Intensity measure: uniform or not

$$\nu(A)\int_A \lambda(x)dx$$

NDVI MAP

The density

The model is defined by a density (usually un-normalised) w.r.t the reference measure:

$$f:\Omega \to [0,\infty[, \int_{\Omega} f(x)\, d\pi_{\nu}(x) < \infty$$

Mimicking the Bayesian approach:

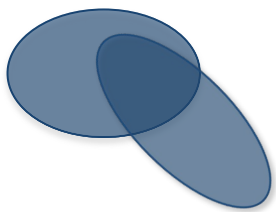
$$f(x) = g(x) h_I(x)$$

Prior

Data (I) term

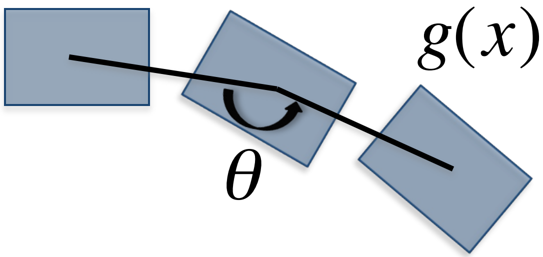
The prior

Overlap penalization (pairwise interaction):



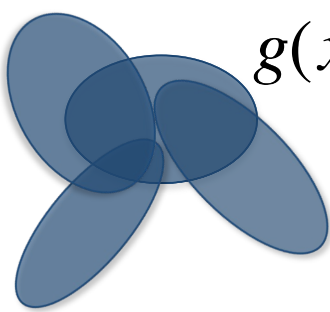
$$g(x) = \prod_{i \sim j} \varphi(x_i, x_j) \qquad \varphi(x_i, x_j) = \Phi\left(\frac{|s_i \cap s_j|}{\min(|s_i|, |s_j|)} \right)$$

Alignment :



$$g(x) = \prod_{i \sim j \sim k} \varphi(x_i, x_j, x_k) \quad \varphi(x_i, x_j, x_k) = \Phi\left(\left|\pi - \theta_{i,j,k}\right|\right)$$

Overlap penalization :



$$g(x) = \prod_{i} \varphi(x_i, x_j, j \sim i)$$

$$\varphi(x_i, x_j, j \sim i) = \Phi\left(\max_{j \sim i} \left(\frac{|s_i \cap s_j|}{\min(|s_i|, |s_j|)} \right) \right)$$

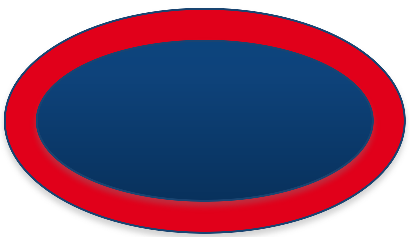
And many more …

The data term

Bayesian approach: likelihood

$$h_Y(x) = \prod_{i \text{ inside objects}} 1_{\text{object}}(y_i) \prod_{i \text{ outside objects}} 1_{\text{background}}(y_i)$$

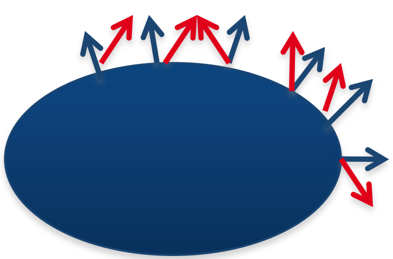
Distance between interior and exterior



$$h_Y(x) = \prod_{\text{objects}} dissimilarity(\text{red pixels, blue pixels})$$

Examples : Bhattacharya distance, Statistical test,…

Geometrical consistency



$$h_Y(x) = \prod_{\text{objects}} \exp - U(\omega)$$

$$U(\omega) = \frac{1}{|\partial\omega|} \int_{\partial\omega} \left\langle \frac{\nabla Y(u)}{\sqrt{\|\nabla Y(u)\|^2 + \varepsilon^2}}, n(u) \right\rangle du$$

$$U_d(\omega, Y) = \psi(U(\omega), t)$$

And many more …

Optimization : RJMCMC

$$(h(x))^{1/T} d\pi(x)$$

- initialize the temperature T and the configuration x (empty set)
- Choose a proposition kernel $Q_m(x,.)$ with probability $p_m(x)$, or let the configuration unchanged probability $1-\Sigma_m p_m(x)$.
- Sample x' according to the chosen kernel
- Compute the acceptation ratio :

$$R_m(x,x') = \frac{D_m(x',x)}{D_m(x,x')} = \frac{(h(x'))^{1/T}\pi(dx')Q_m(x',dx)}{(h(x))^{1/T}\pi(dx)Q_m(x,dx')}$$

- With probability $\alpha = \min(1,R_m)$ set $x_{t+1} = x'$, else reject the proposition : $x_{t+1} = x$.

Some perturbation kernels (proposal)

Adding an object

Removing an object

Modifying an object (translation, rotation, dilation)

Merging/Splitting objects

Optimization : RJMCMC

Pros :

- Generality
- Choice for kernels
- Convergence to the global optimum

Cons :

- Rejection
- Simulated annealing scheme (parameters setting)
- Kernels usually involve one or two objects

Optimization :Multiple births and deaths

Goals :

Avoid rejection
Consider several objects at once

Idea :

Extend Langevin's dynamics (Stochastic Differential Equation : diffusion process)

Optimization :Multiple births and deaths

Consider : $f(x) = \exp - \beta E(x)$

A birth intensity consisting in adding an object u to the configuration x :

$$b(x,u)du = z\,du \text{ if } u \in D(x), \text{ where } D(x) = \left(\bigcup_{v \in x} B_v(\varepsilon)\right) \bigcap K$$

A death intensity consisting in removing an object u from the configuration x :

$$d(x,u) = \exp \beta \left[E(x) - \mathrm{E(x/u)} \right] \text{ if } u \in x,$$

Detailed balance condition holds

Optimization :Multiple births and deaths

A New Approximation Process

$$\text{Markov Chain : } T_{\beta,\delta}(m),\ m = \left[\frac{t}{\delta}\right] = 0,1,2,\cdots\ (\text{discretization of time})$$

$$x_{n+1} = x_1 \cup x_2,\ \ x_1 \subseteq x_n,\ \ x_1 \cap x_2 = \varnothing$$

$$x_2 : \text{Poisson law (with intensity z) distributed}$$

Birth transition :

$$q_{u,\delta} = \begin{cases} z\delta\Delta u, & \text{if } x \to x \cup \{u\} \\ 1 - z\delta\Delta u, & \text{if } x \to x \text{ (no birth in } \Delta y) \end{cases}$$

Death transition :

$$p_{u,\delta} = \begin{cases} \dfrac{\delta \exp\left[E(x) - E(x/u)\right]}{1 + \delta \exp\left[E(x) - E(x/u)\right]}, & \text{if } x \to x/u \\ \dfrac{1}{1 + \delta \exp\left[E(x) - E(x/u)\right]}, & \text{if } x \to x \text{ } (x \text{ survives}) \end{cases}$$

Optimization :Multiple births and deaths

A New Algorithm

1) Precomputing of the data term / birth map

2) Repeat :

2.1) Birth:

For each pixel, add an object with probability :

$$\delta B\left(E_d(u)\right)$$

2.2) Sort the objects with respect to their data term value

2.3) Death:

For each object u taken in the list order, remove it with probability :

$$\frac{\delta \exp\left[\beta\left(E(x)-E(x/u)\right)\right]}{1+\delta \exp\left[\beta\left(E(x)-E(x/u)\right)\right]}$$

Optimization :Multiple births and deaths

<u>Theorem:</u> When $N \rightarrow \infty, \beta \rightarrow \infty, \delta \rightarrow 0$, convergence toward the configuration minimizing the energy

Pros :

No rejection in the birth step
Birth does not depends on the temperature
Convergence to the global optimum

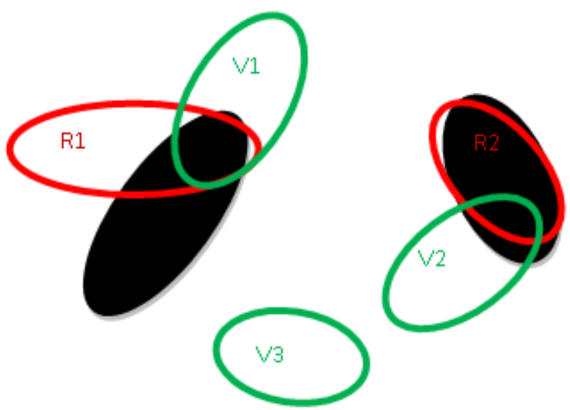
Cons :

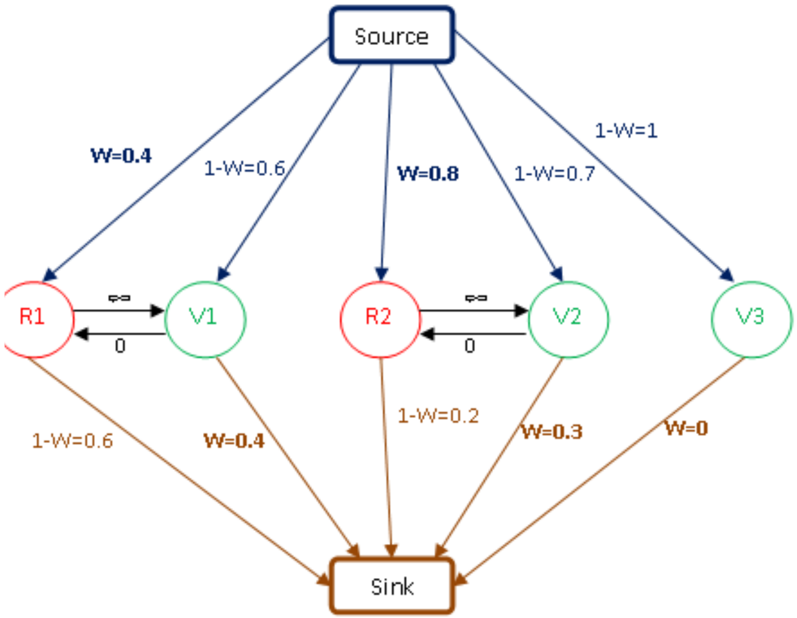
Only births and deaths kernels

Optimization :Multiple births and cut

Idea : Combine multiples births and deaths with graphcut techniques

- **Generate a first configuration x_0 of non overlapping objects and iterate the following steps:**
- ***Birth:***
  - ***generate a new configuration of non overlapping objects x'***
- ***Death:***
  - ***x_{n+1} is defined by $Cut(x_n \cup x')$***





Optimization :Multiple births and cut

Pros :

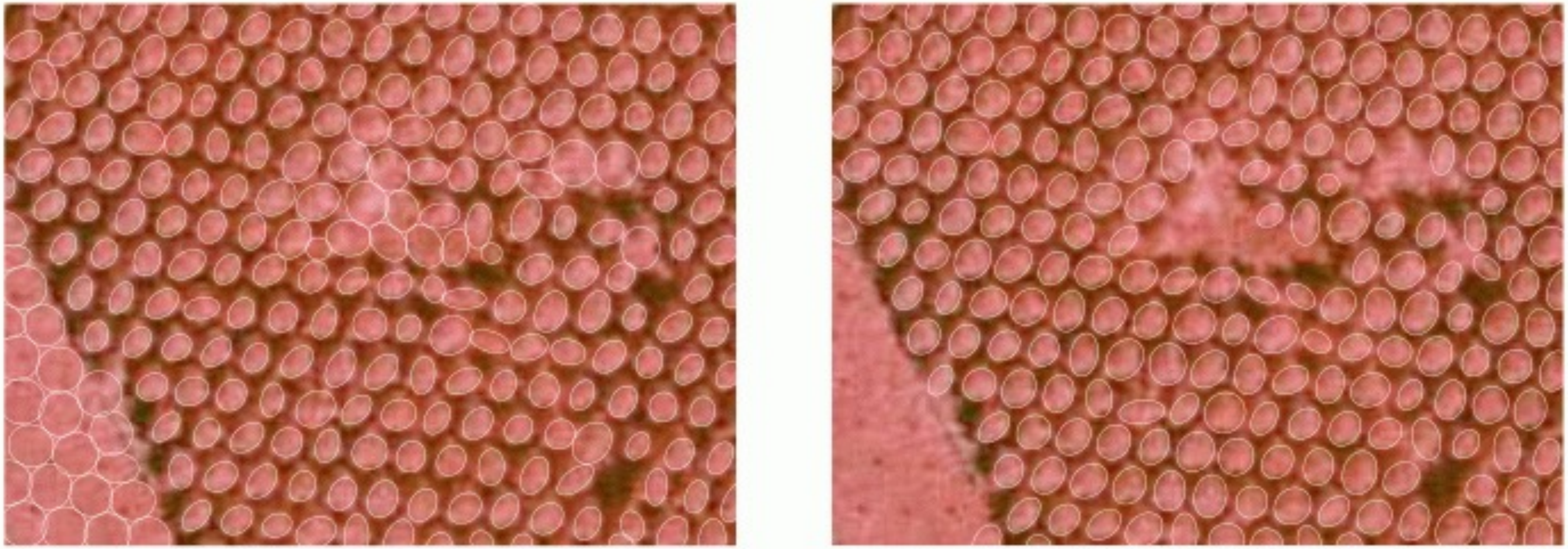
No rejection in the birth step
No cooling schedule

Cons :

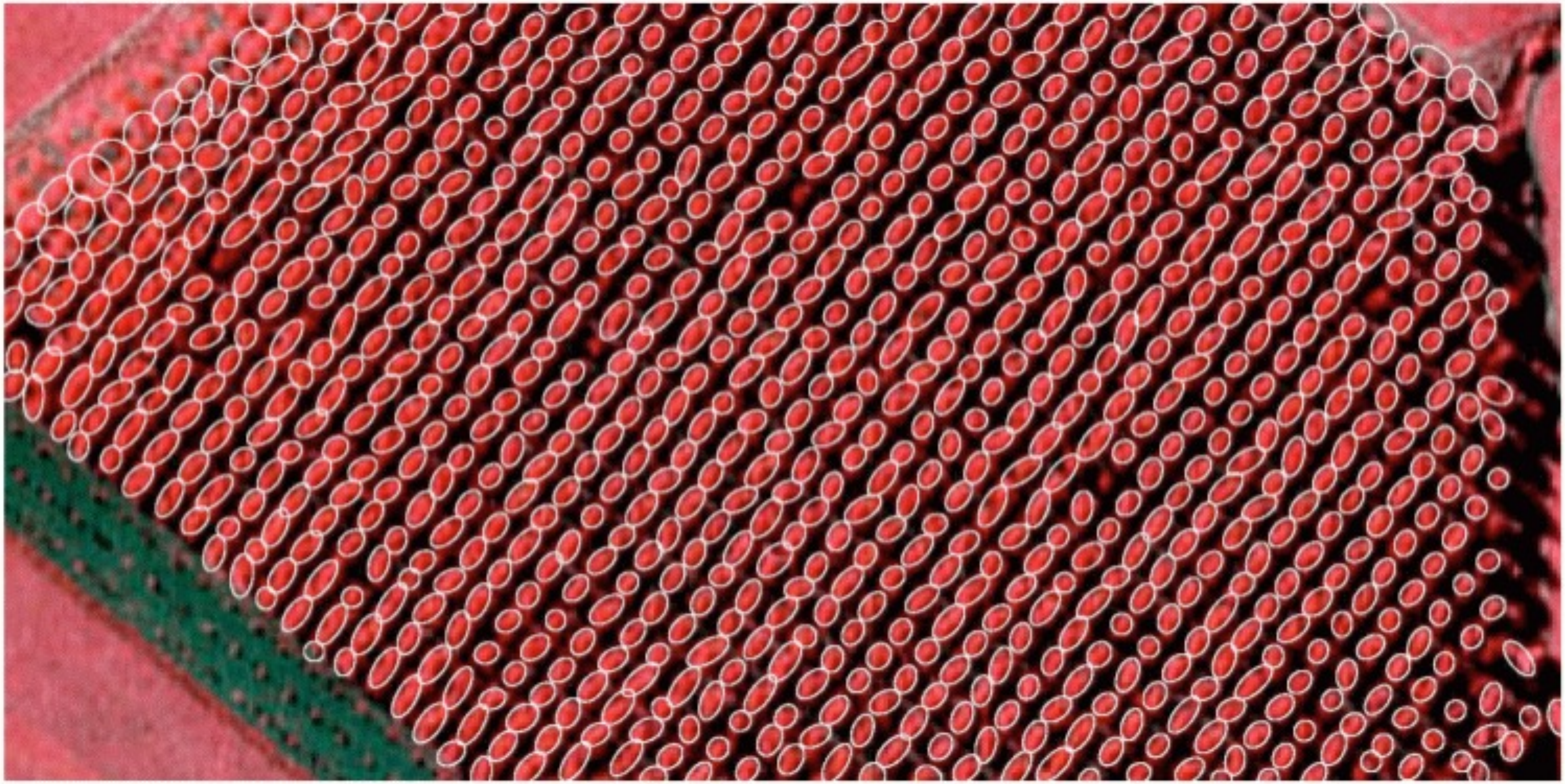
Only births and deaths
No proof of convergence

Result :Trees counting

Result :Trees counting



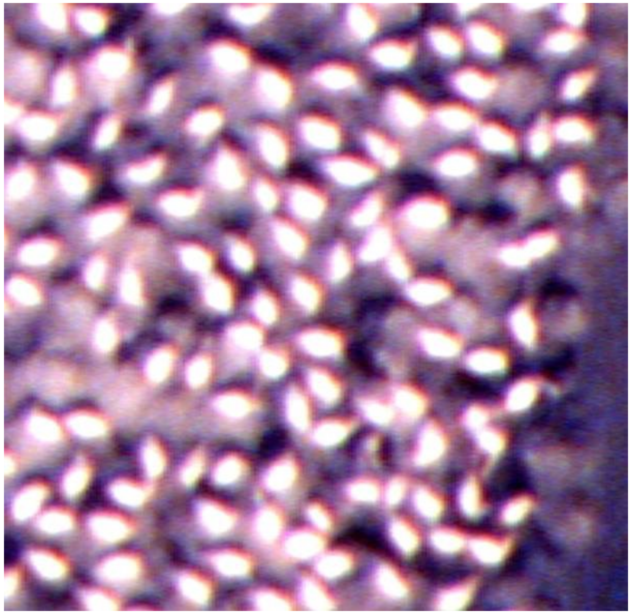
Result :Trees counting

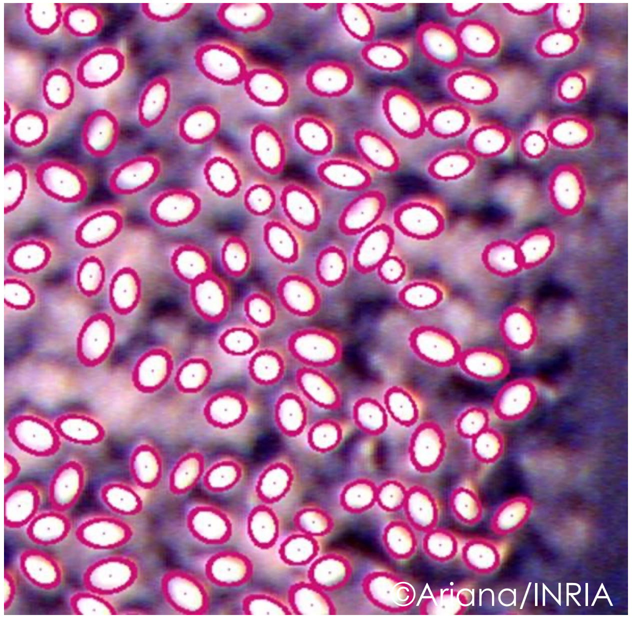


Result : Flamingos counting

<u>Estimation of the size of a colony in Turkey (2004):</u>

♣ **3682 detected flamingos** ***(Tour du Valat: 3684 flamingos)***

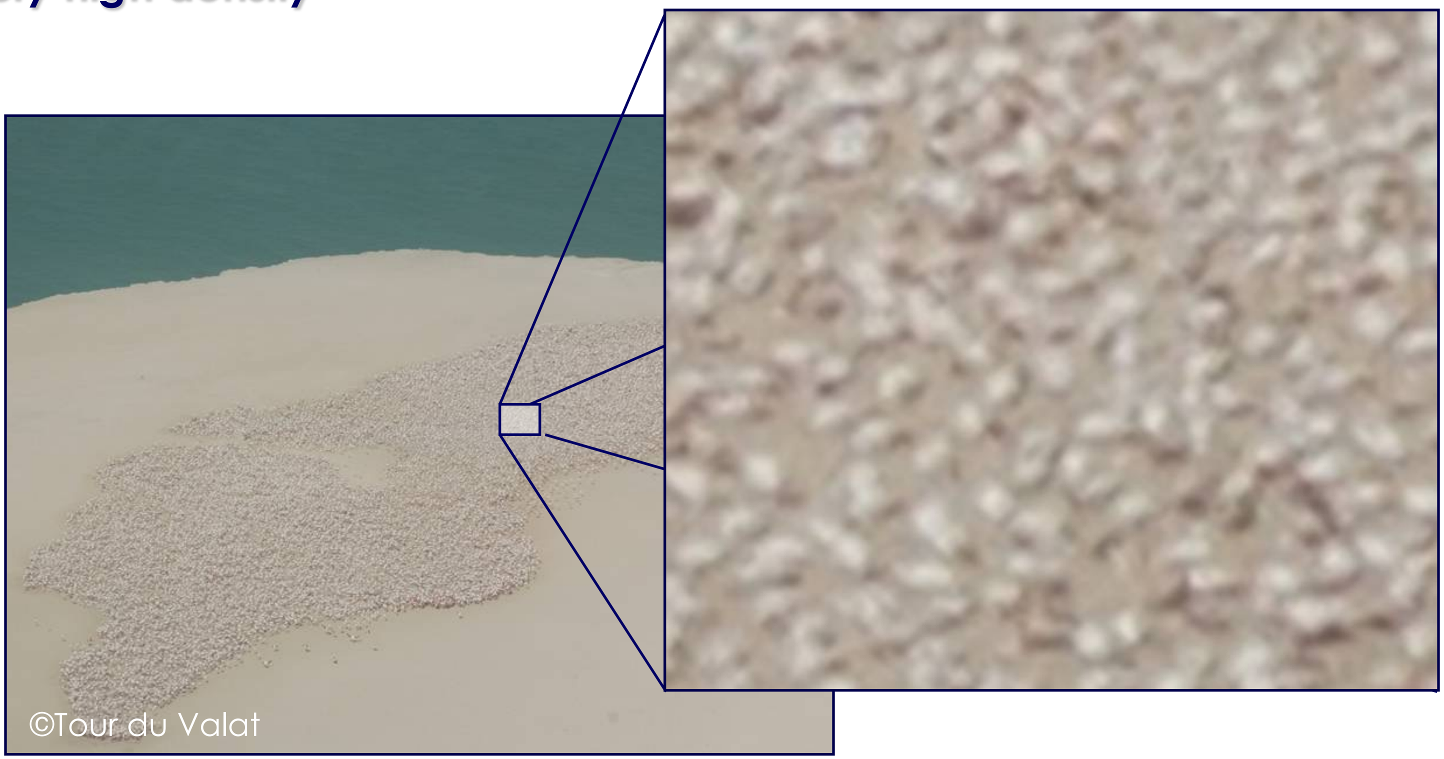




Result : Flamingos counting

Estimation of the size of a colony in Mauritania:

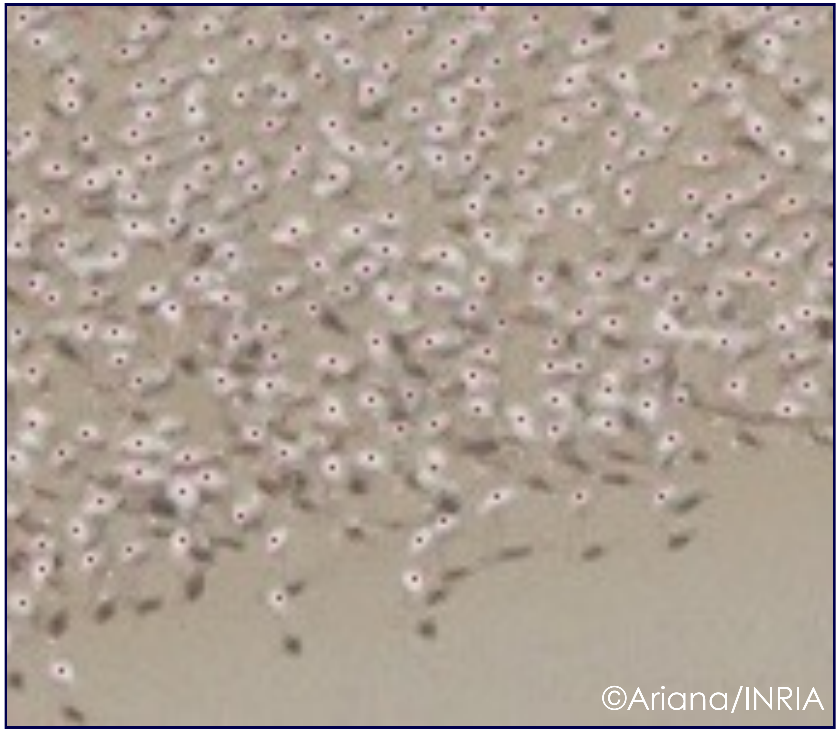
♣ **Very high density**



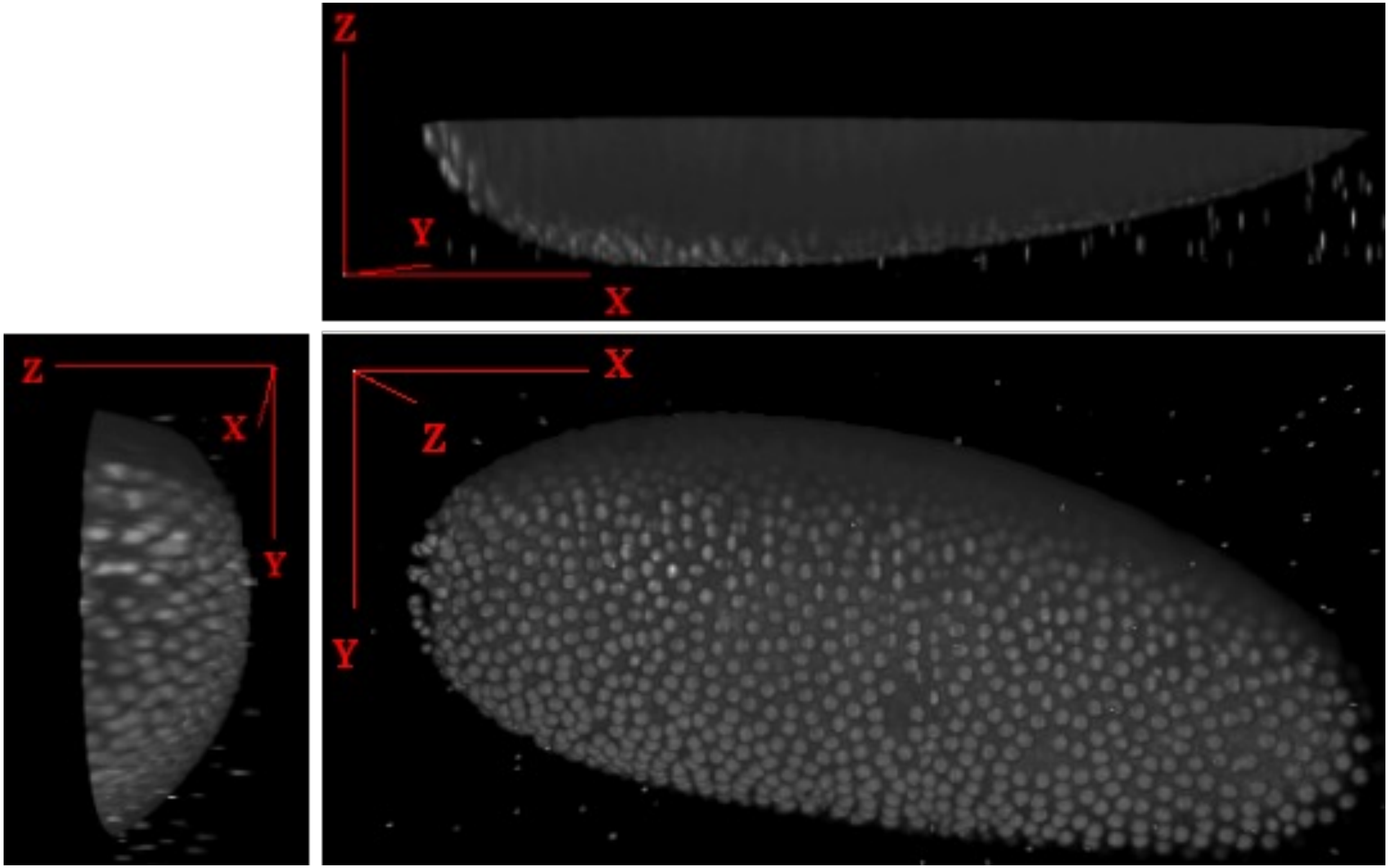
Result : Flamingos counting

Estimation of the size of a colony in Mauritania:

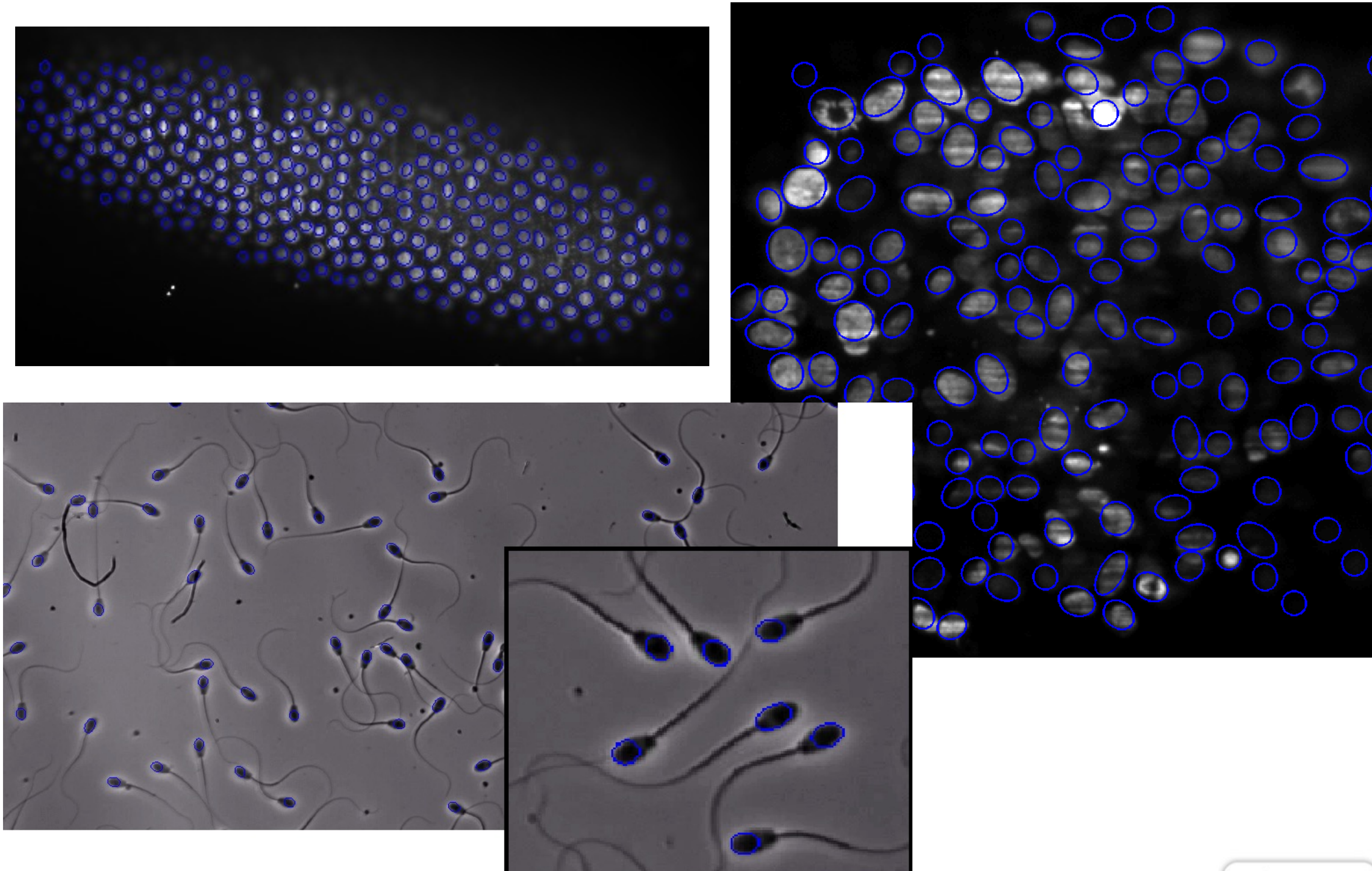
♣ **14595 detected flamingos (*Tour du Valat: 13650 flamingos*)**



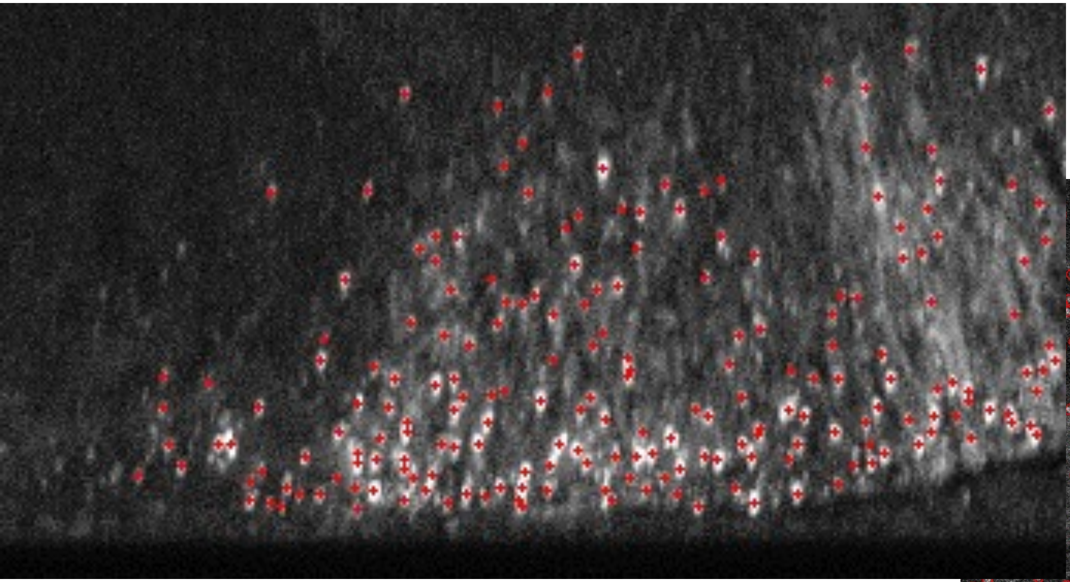
Result : Cells counting

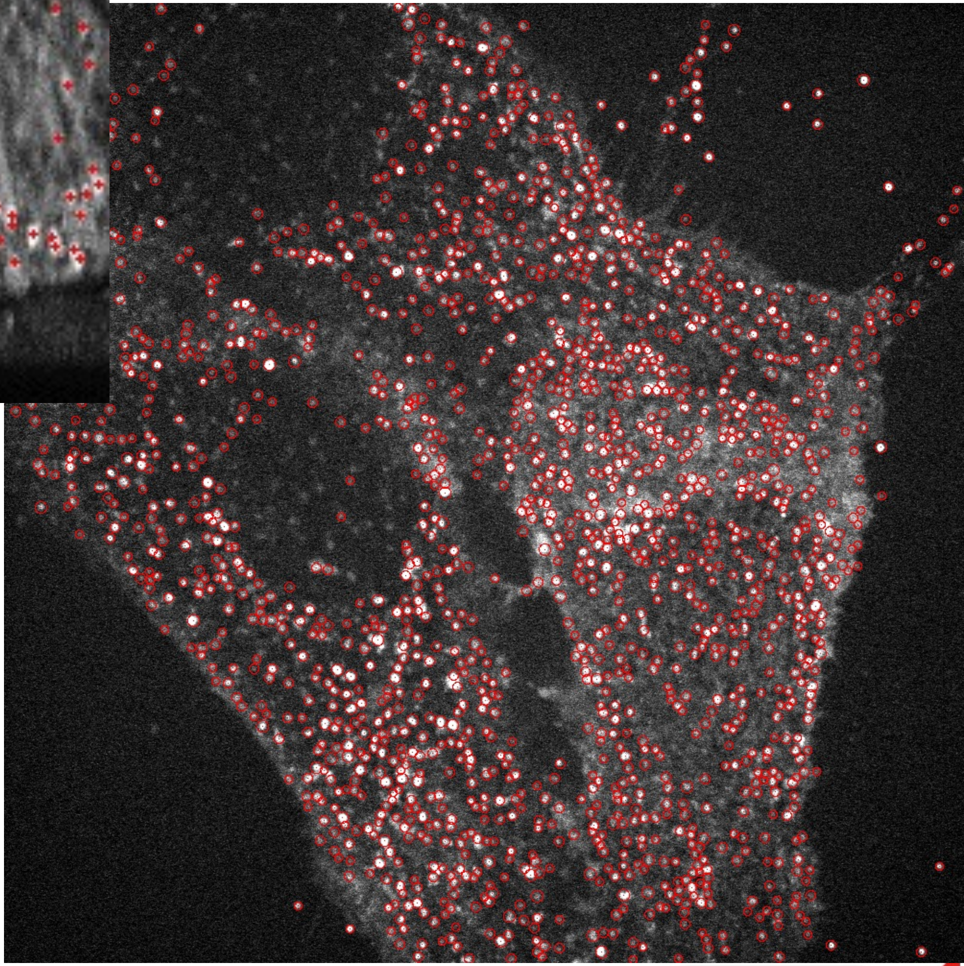


Result : Cells counting



Result : Vesicule (co)localization





Conclusion

Pros :

General framework / Numerous application
Embed strong geometric constraint

Future work :

parallelism
parameter estimation
open source software