Object Tracking in Microscopic Images

Xavier Descombes
Morpheme team
INRIA/I3S/iBV

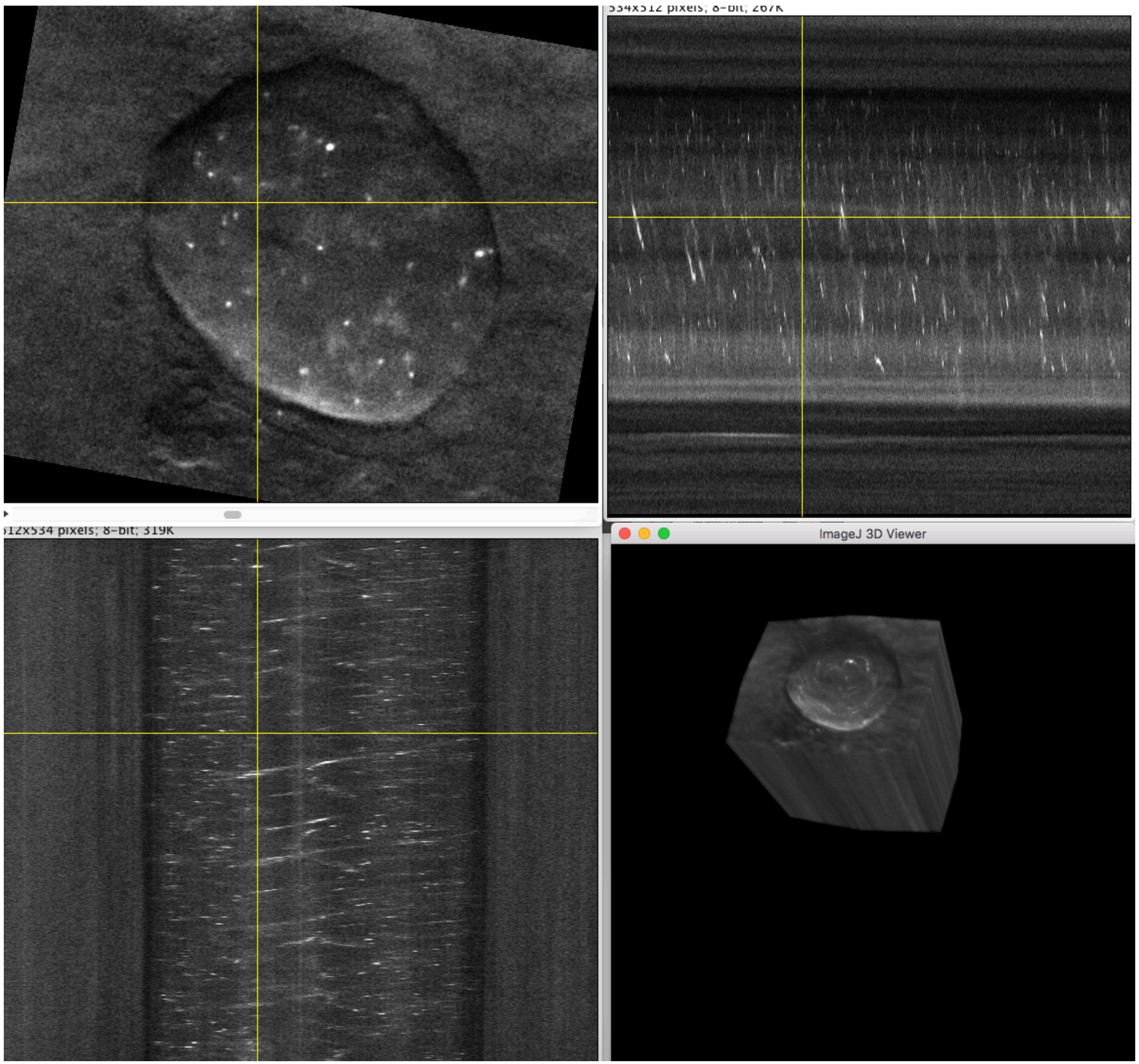
Multiple objects tracking

- Goal : follow a population, classification from movement, movement statistics,…
- Huge datasets : one more dimension !
- Requirements: robustness, efficiency, ergonomic
- Special case : particles tracking (an object is given by a point)

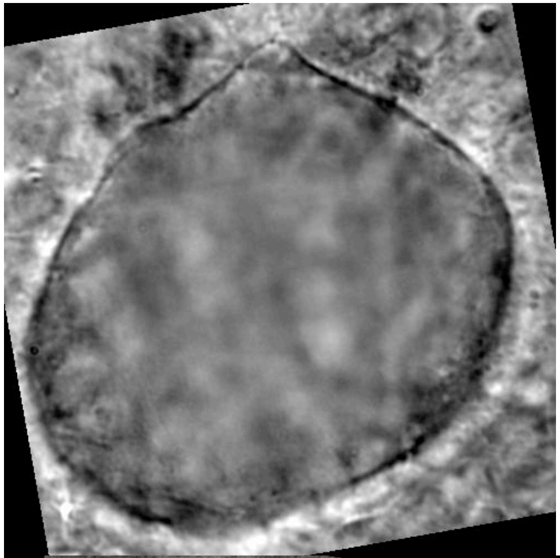
Multiple particles tracking

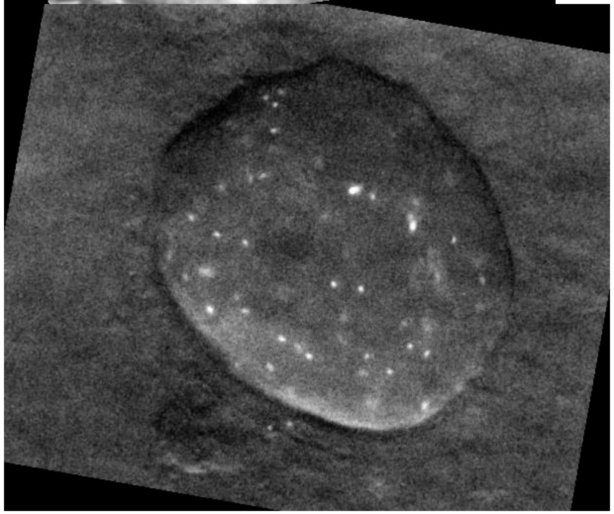
- Main approaches in two steps :
  1) Objects (particles) detection
  2) Objects (particles) linking
- Particular case : (for low speed)
  - » Trajectories detection in (Space + Time) domain

Multiple particles tracking

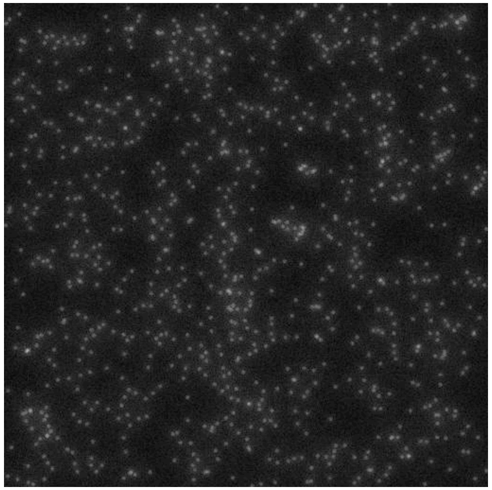


Particles tracking :Examples

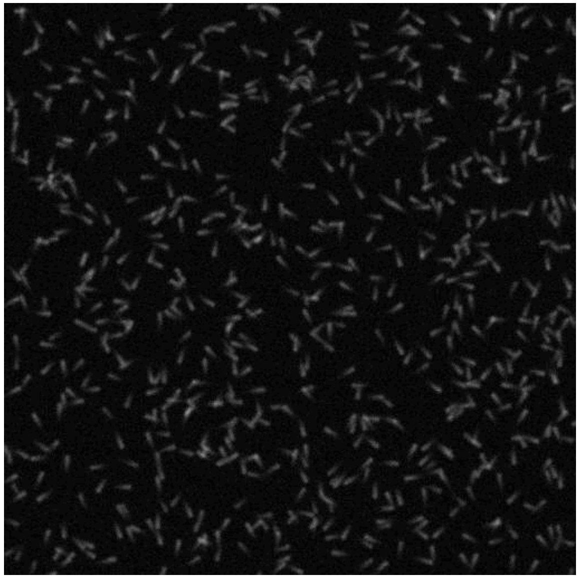




- Vesicles :



- Microtubules :



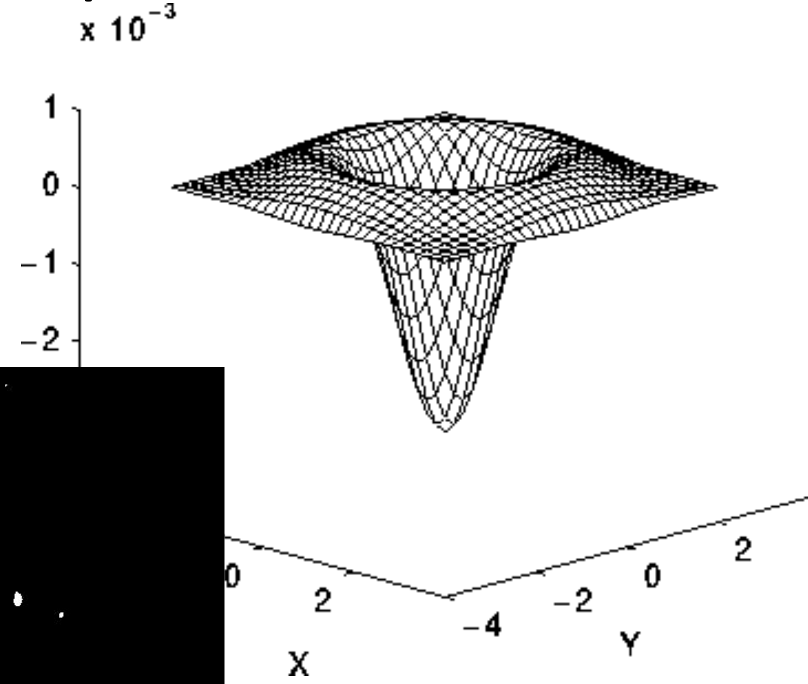
The ImageJ (Fiji) approach

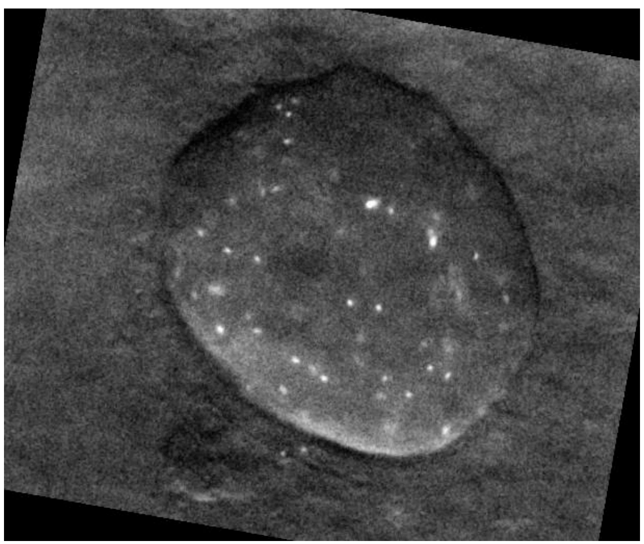
- https://imagej.net/MTrack2
- https://imagej.net/TrackMate
- https://imagej.net/Particle_Tracker
- …

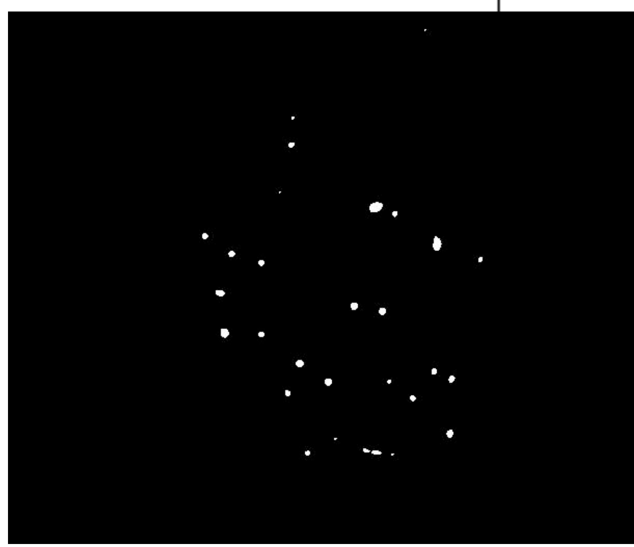
Trackmate

- Detection using LoG filter (Laplacian of Gaussian) :

$$Log(x,y) = -\frac{1}{\pi\sigma^4}\left[1 - \frac{x^2+y^2}{2\sigma^2}\right]\exp - \left[\frac{x^2+y^2}{2\sigma^2}\right]$$

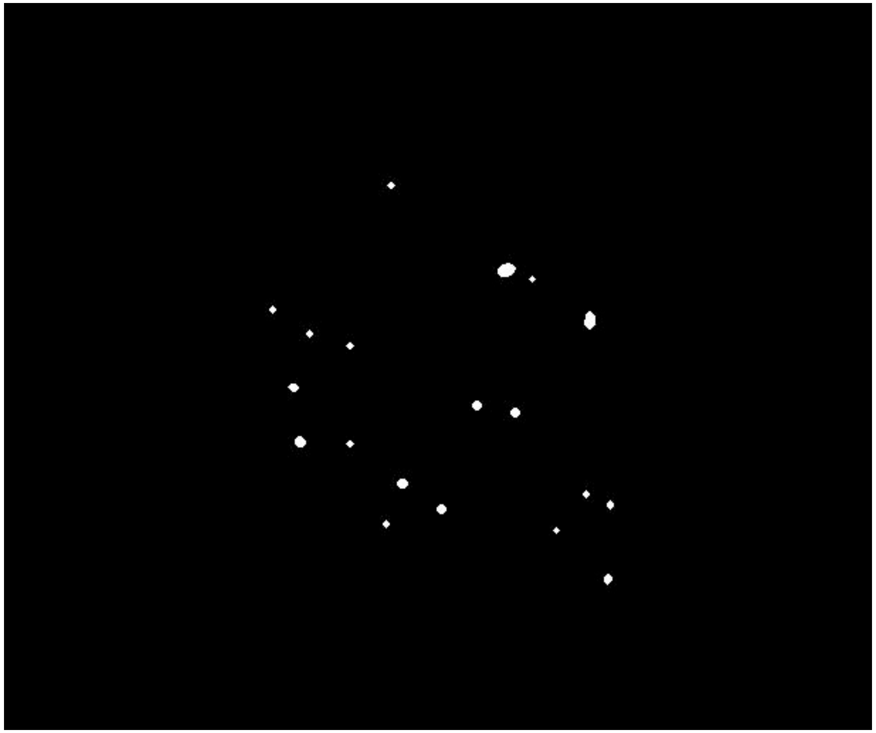






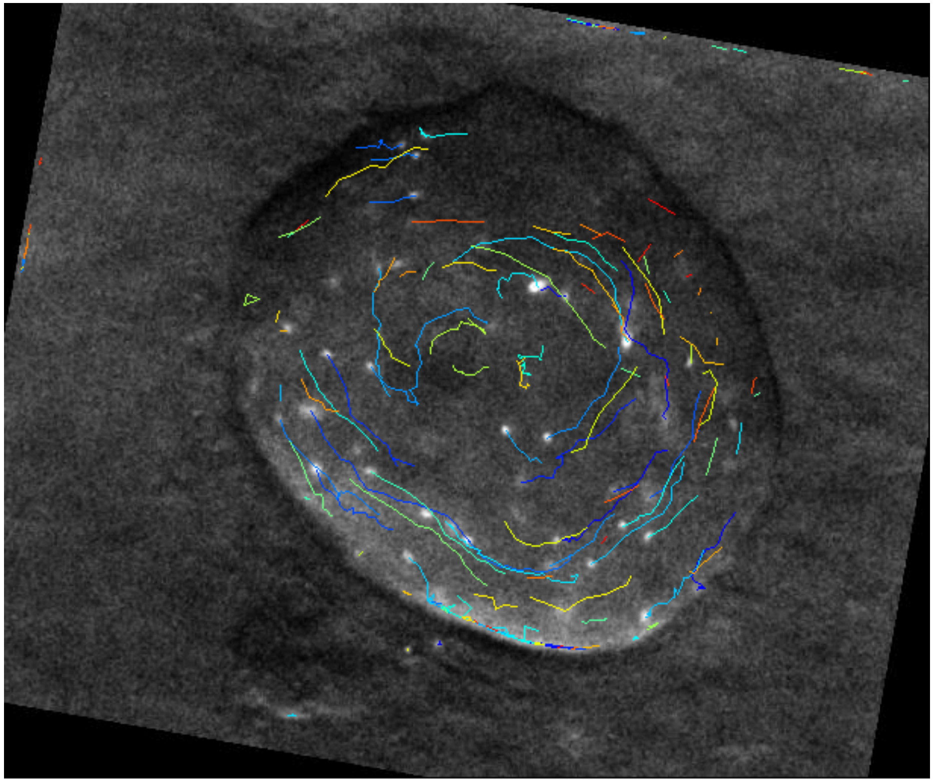
Trackmate

- Particle selection from size (min and max)



Trackmate

- Matching : nearest neighbor
- Maximum speed (distance) between consecutive frame,
- gap filling



TrackM2

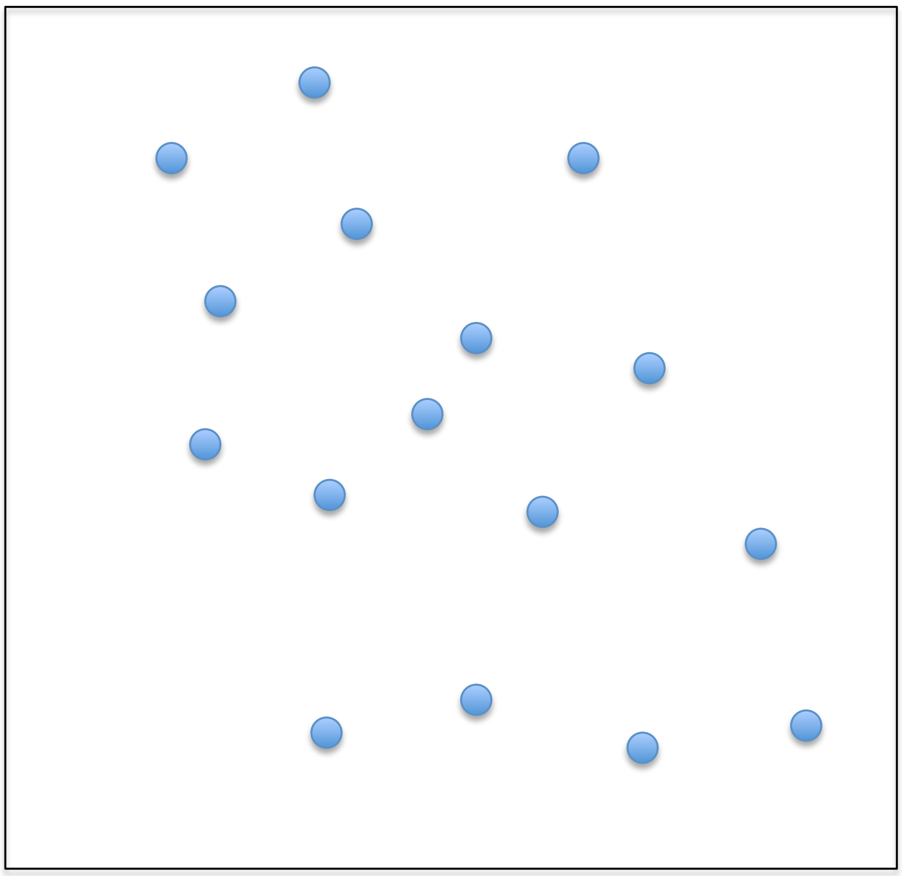
- Influence of the maximum velocity

The challenges

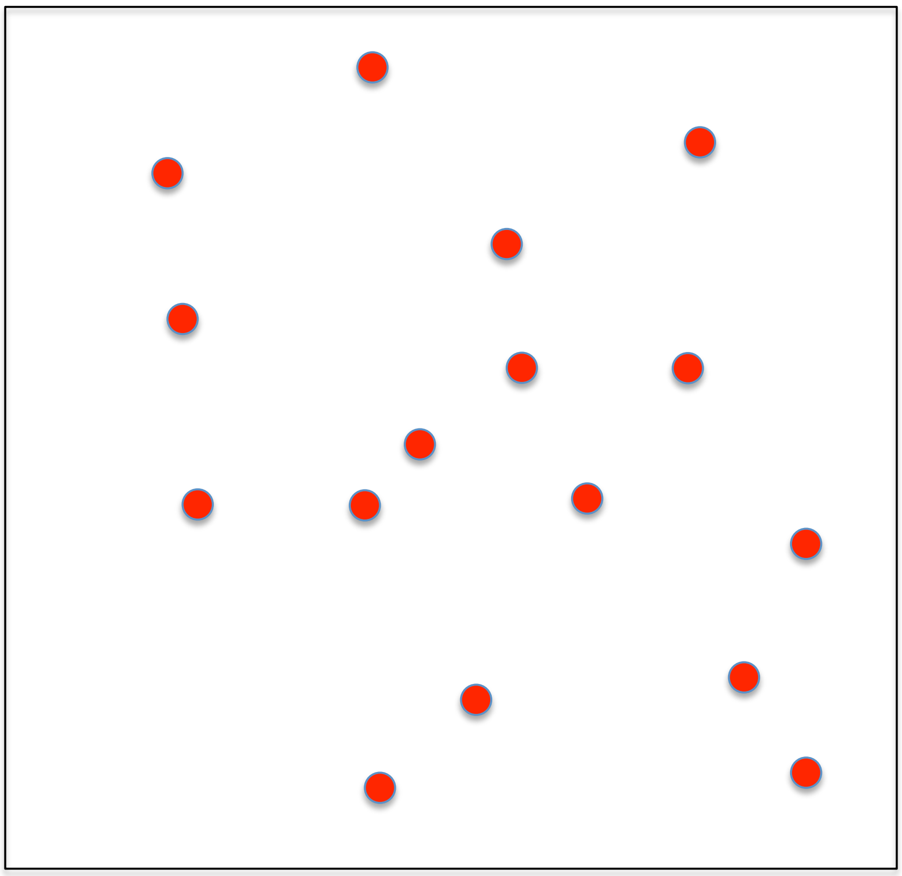
- Detection (see corresponding course)
- Appearance / Disappearance of particles
- Crossing
- Occlusion
- Noise
- Location VS shape descriptor

Detect and Match

t



t + 1

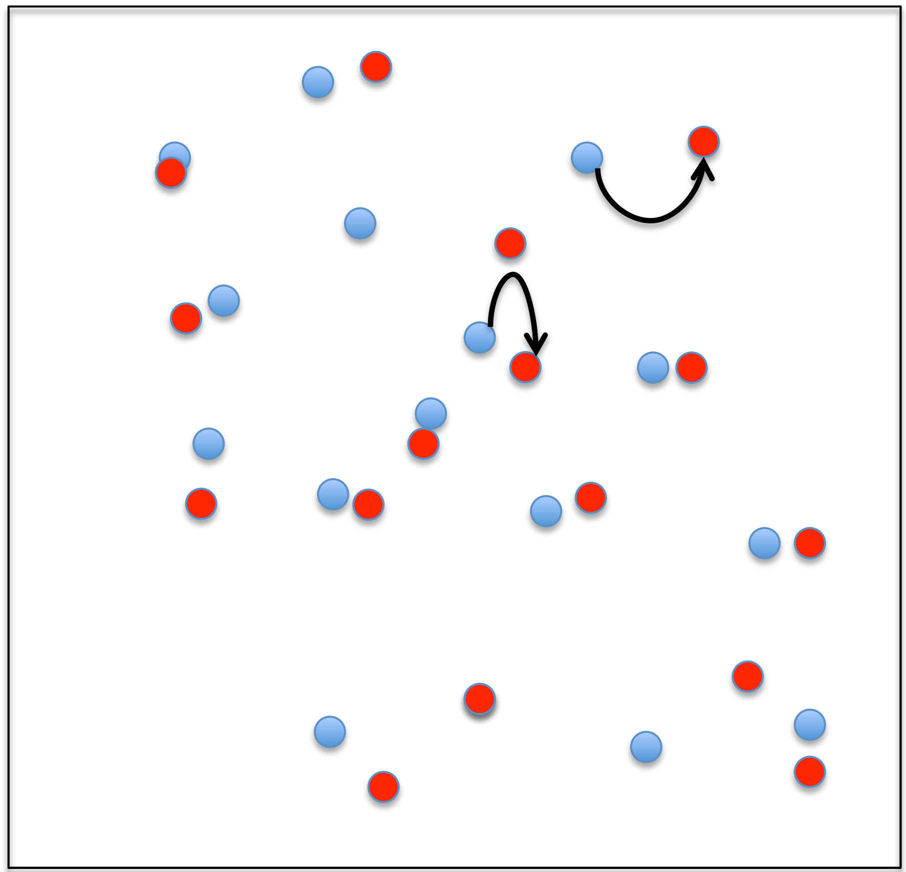


Match : nearest neighbor

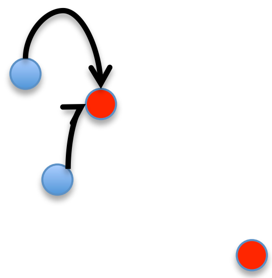
x_i

y_j

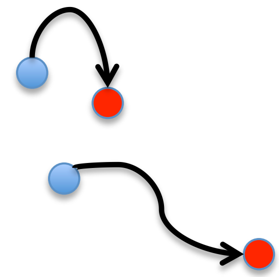
$$M(x_i) = \arg\min_{y_j} d(x_i, y_j)$$



Unicity constraint

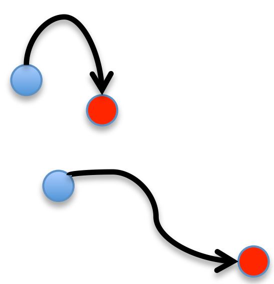


versus

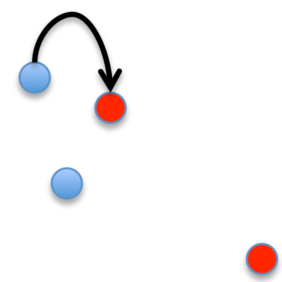


Matching matrix : $M(i,j) = 0 \text{ or } 1$ $\quad \sum_i M(i,j) = 0 \text{ or } 1$

Maximum velocity



versus



$$d(x_i, y_j) > V_{max} \implies M(i,j) = 0$$

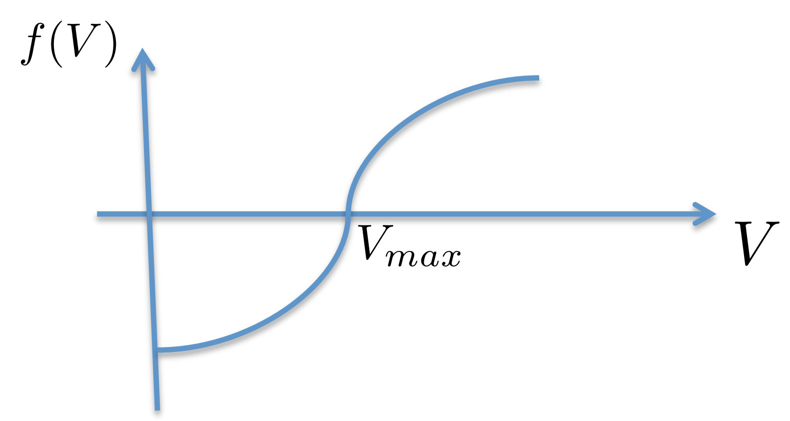
Global optimization

$$argmin_M \sum_{i \in I, j \in J} d(x_i, y_j) M(i,j)$$

$$\boxed{M(i,j) = 0 \text{ or } 1}$$

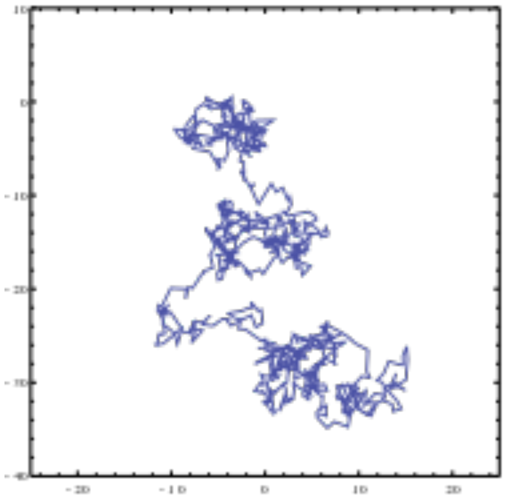
$$\forall i \sum_j M(i,j) = 0 \text{ or } 1$$

$$d(x_i, y_j) = f(||y_j - x_i||)$$



Movement modeling

- Brownian motion : random movement (big particle in a fluid)



$$P(x_{t+1}|x_t) = \frac{1}{2\pi\sqrt{\sigma^2}} \exp\left[-\frac{(x_{t+1}-x_t)^2}{2\sigma^2}\right]$$

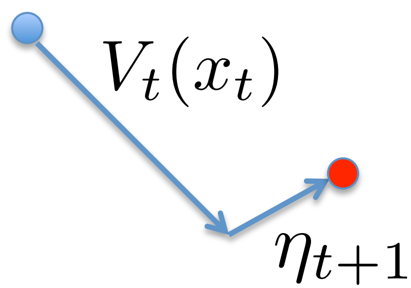
https://fr.wikipedia.org/wiki/Mouvement_brownien#Processus_d%E2%80%99Ornstein-Uhlenbeck

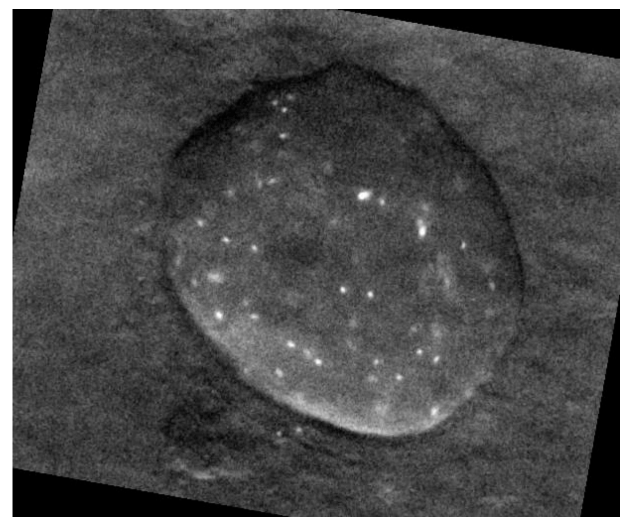
Deterministic speed model

$$x_{t+1} = x_t + V_t(x_t)dt + d\eta_{t+1}$$

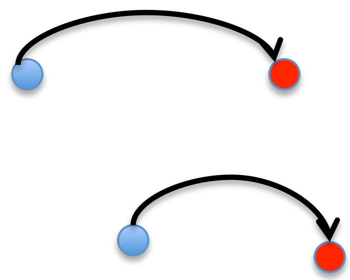
Deterministic model

Fluctuation

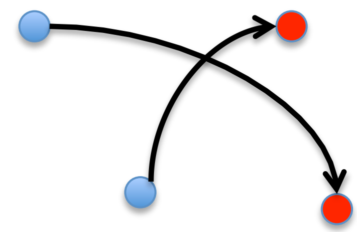


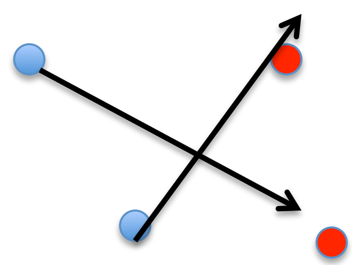


Advantage of a model



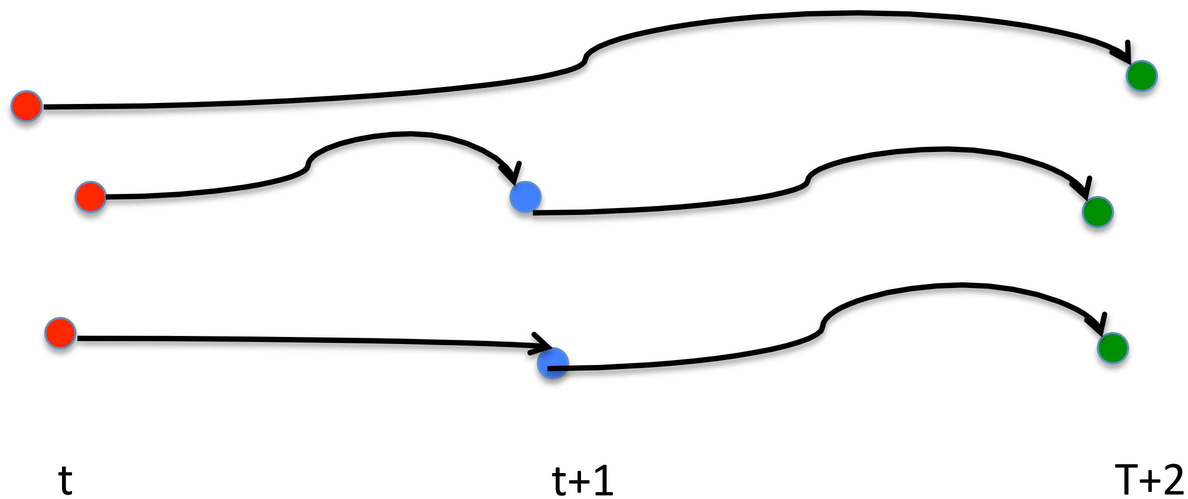
versus



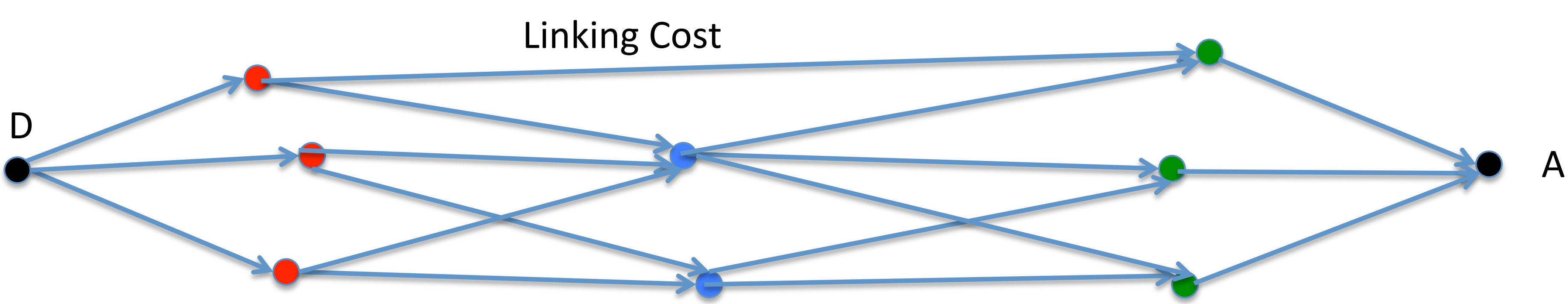


Speed : learnt from a model or estimated from past steps

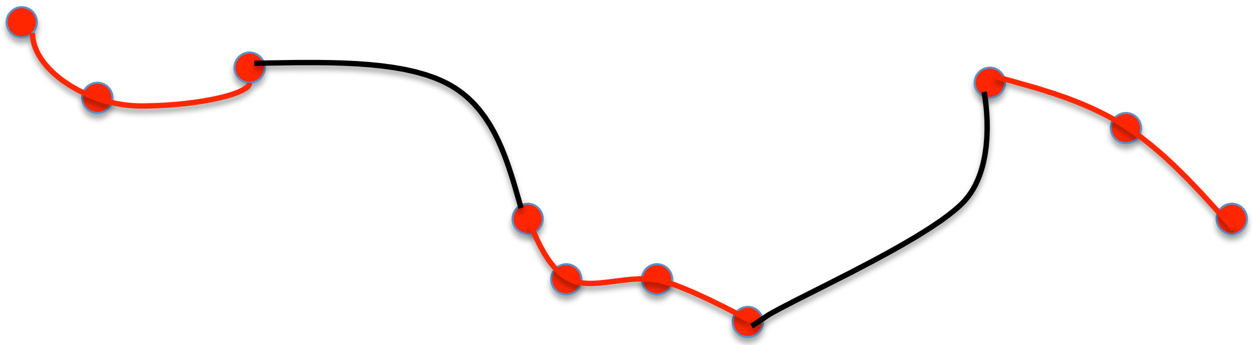
Gap filling



Graph model : Minimal Path



Tracklets



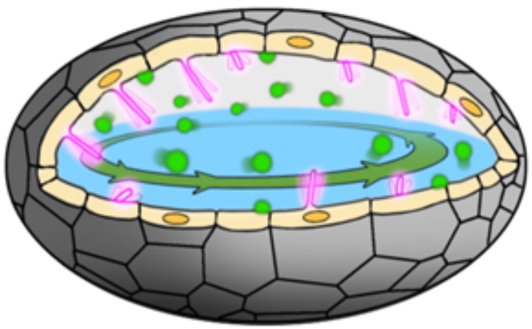
Two steps : Local (tracklets detection), tracklets merging

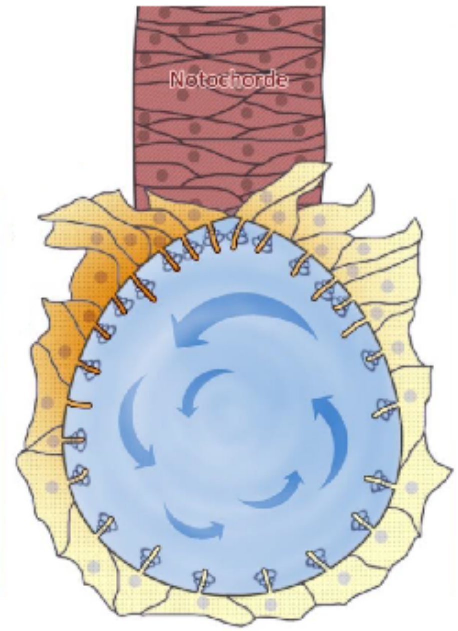
Pros : consider trajectory and/or speed models

3D Trajectory : a case study

Left/Right asymmetry in zebrafish

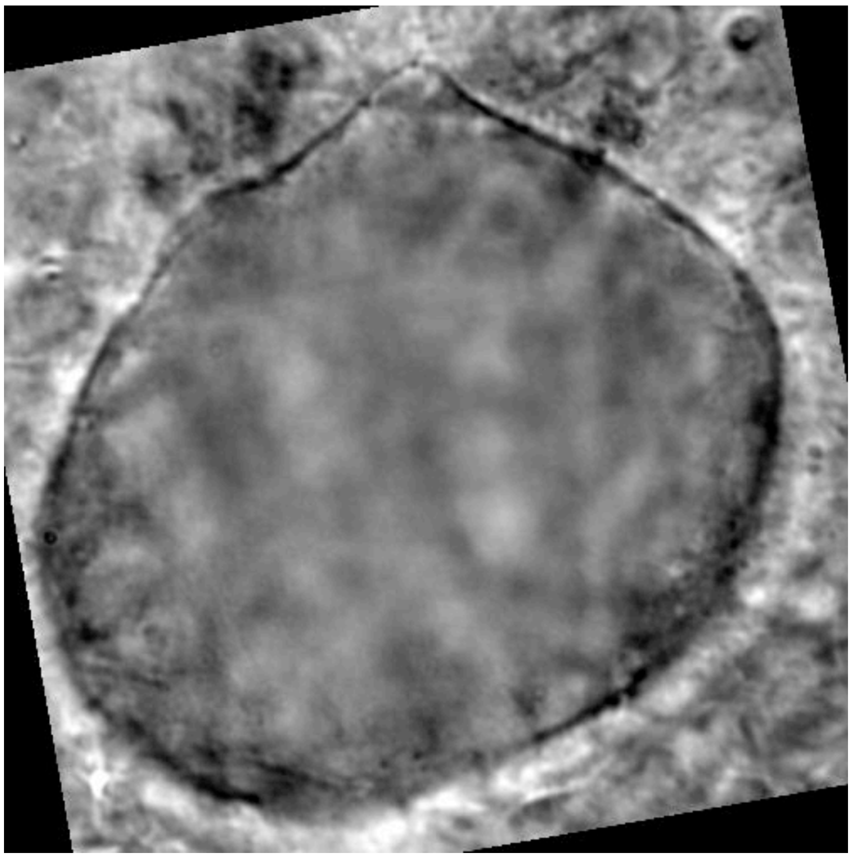
Kupffer vesicle

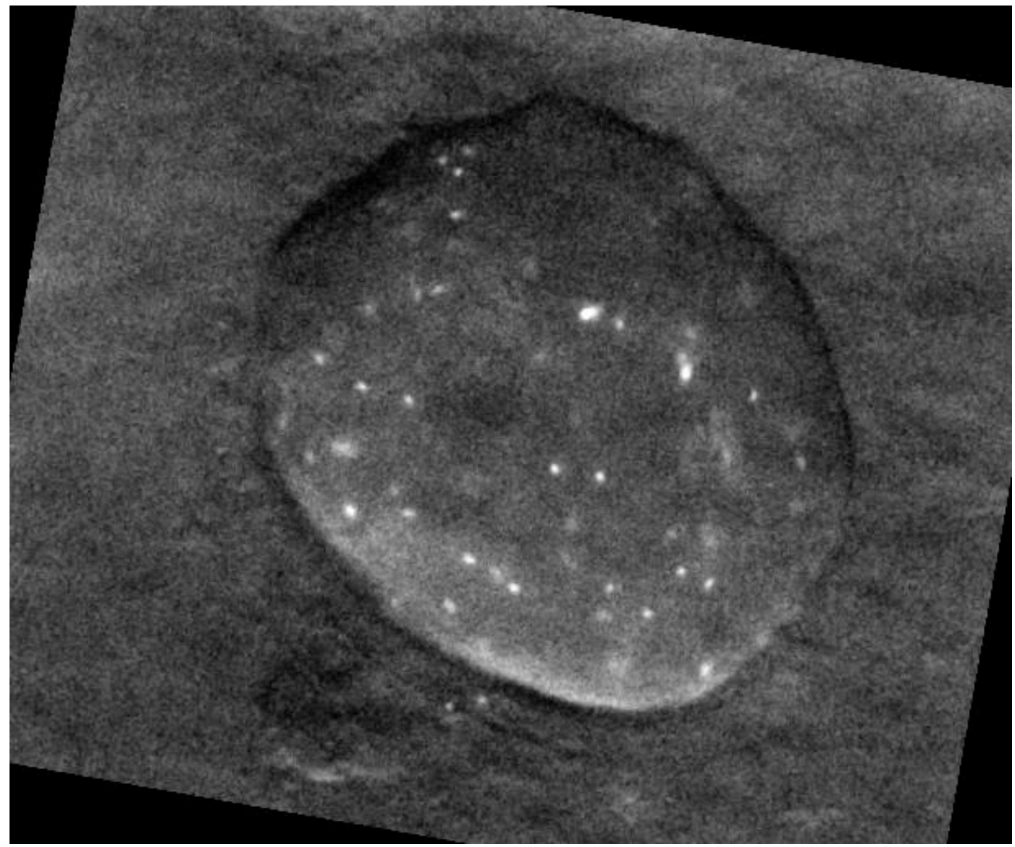




Left/Right organizer

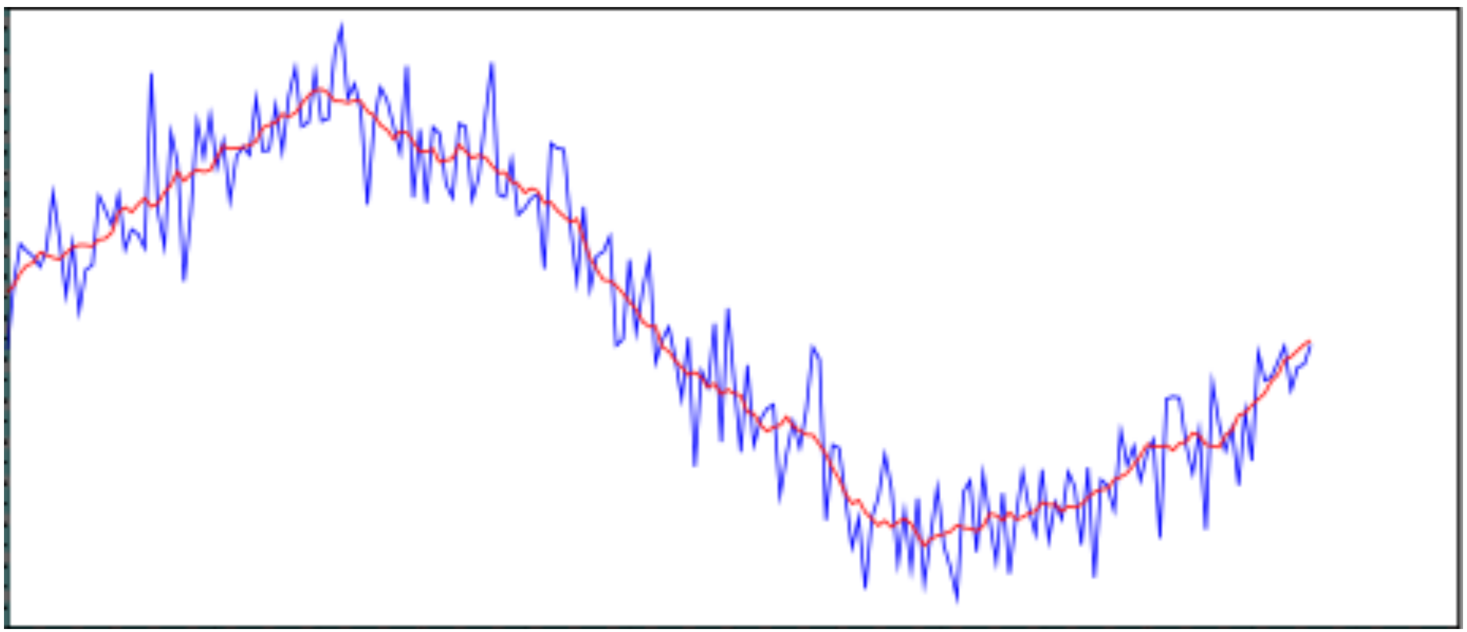
Data : endogeneous and external particles



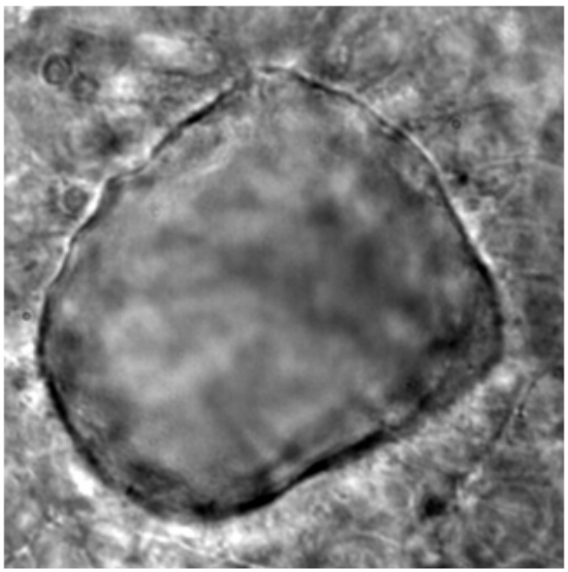


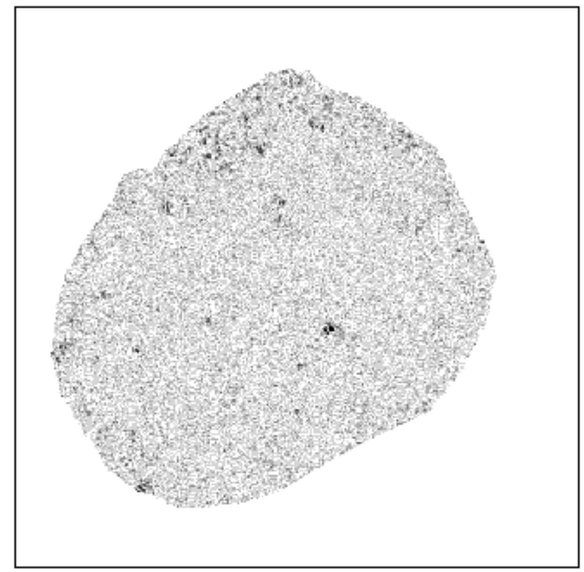
Background substraction

Compute the mean over a sliding window and substract it

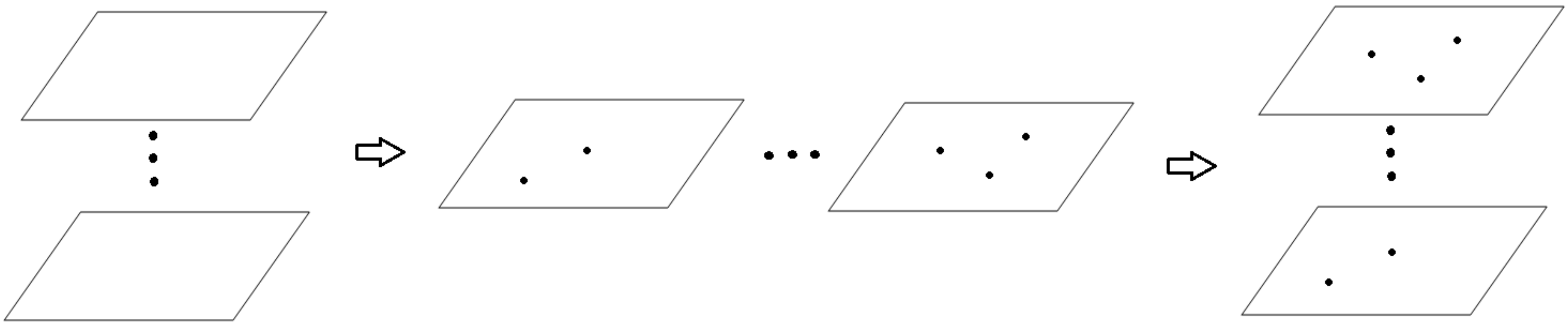


Background substraction: result

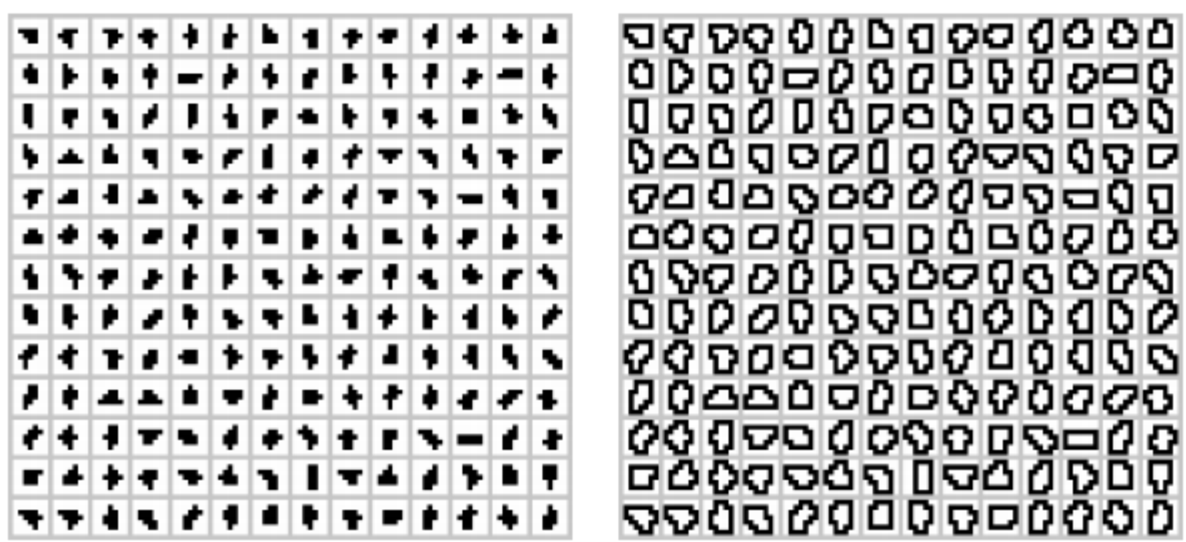




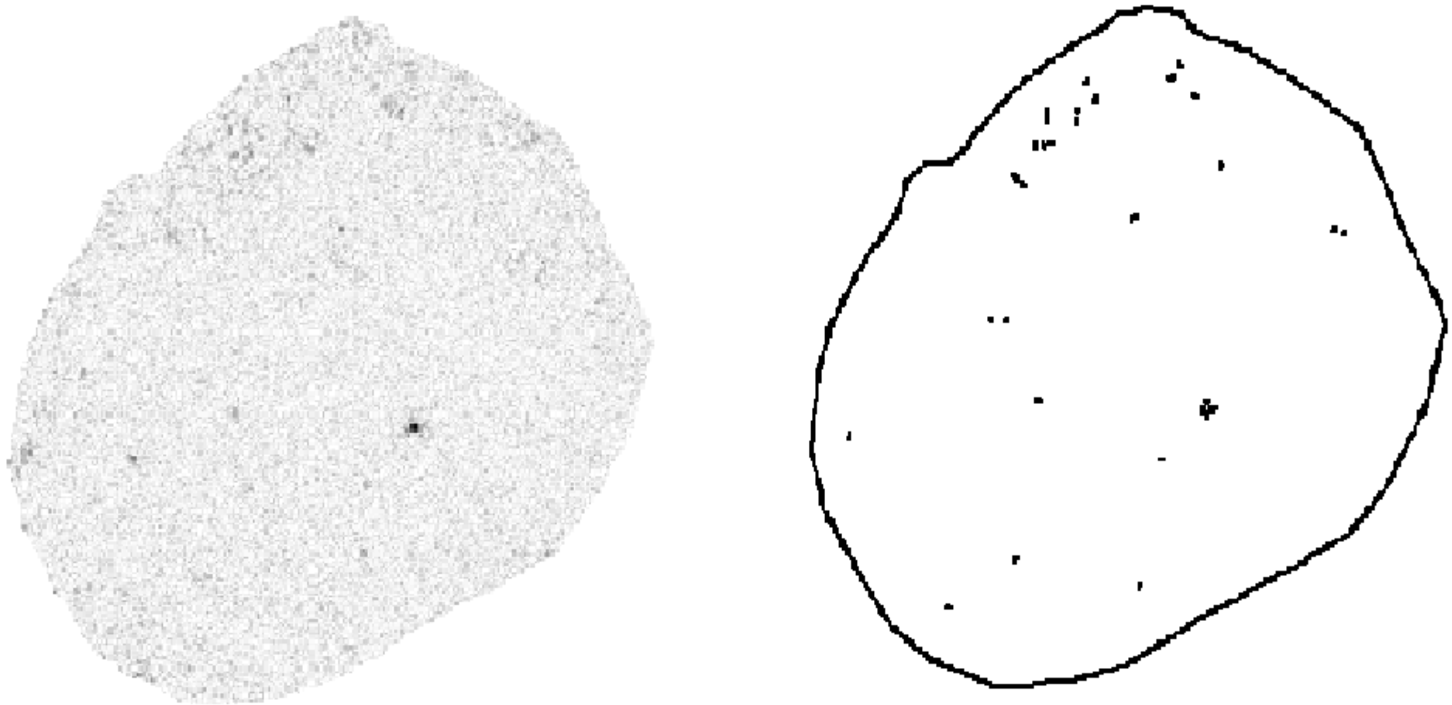
Particles detection : SPADE



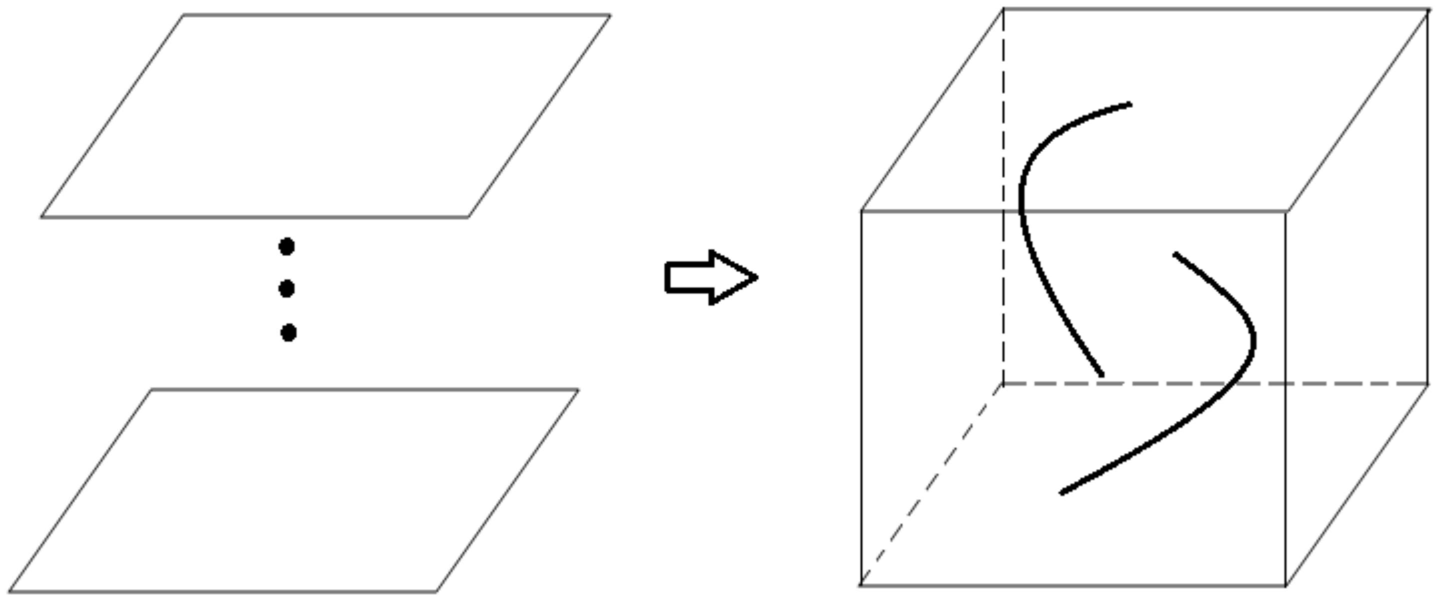
Shape dictionnary :



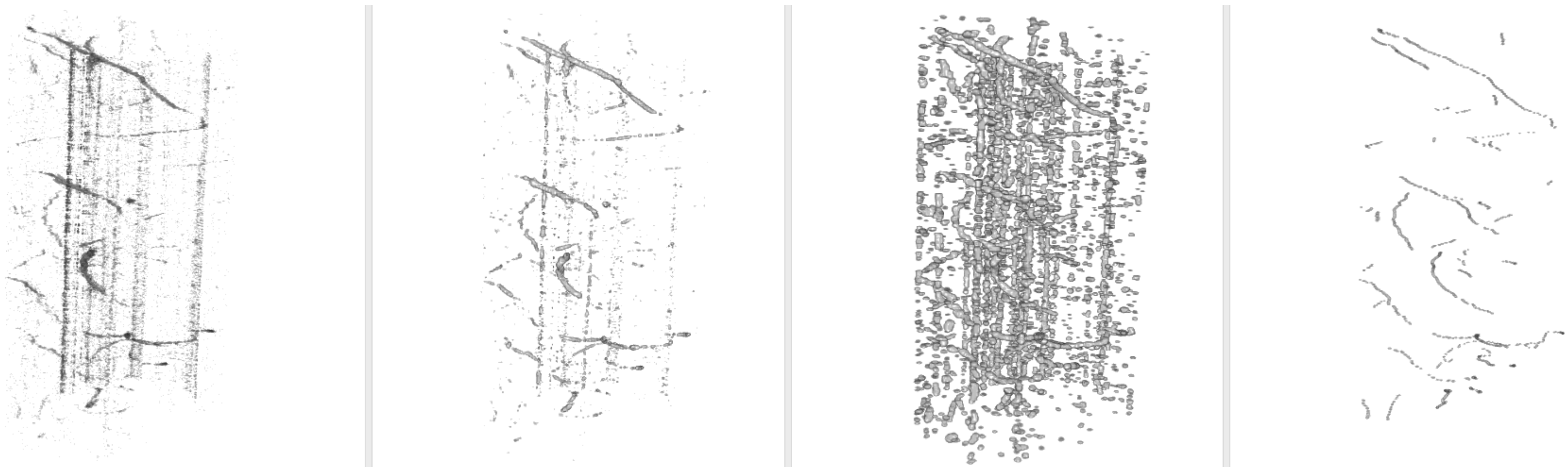
SPADE result



2nd approach : trajectories in 3D



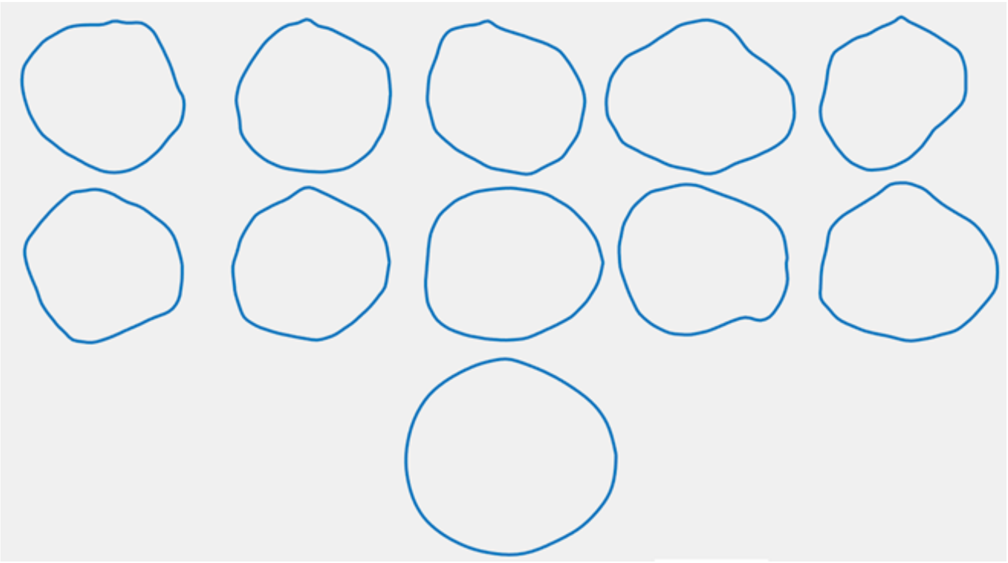
Lines enhancement using Frangi filter



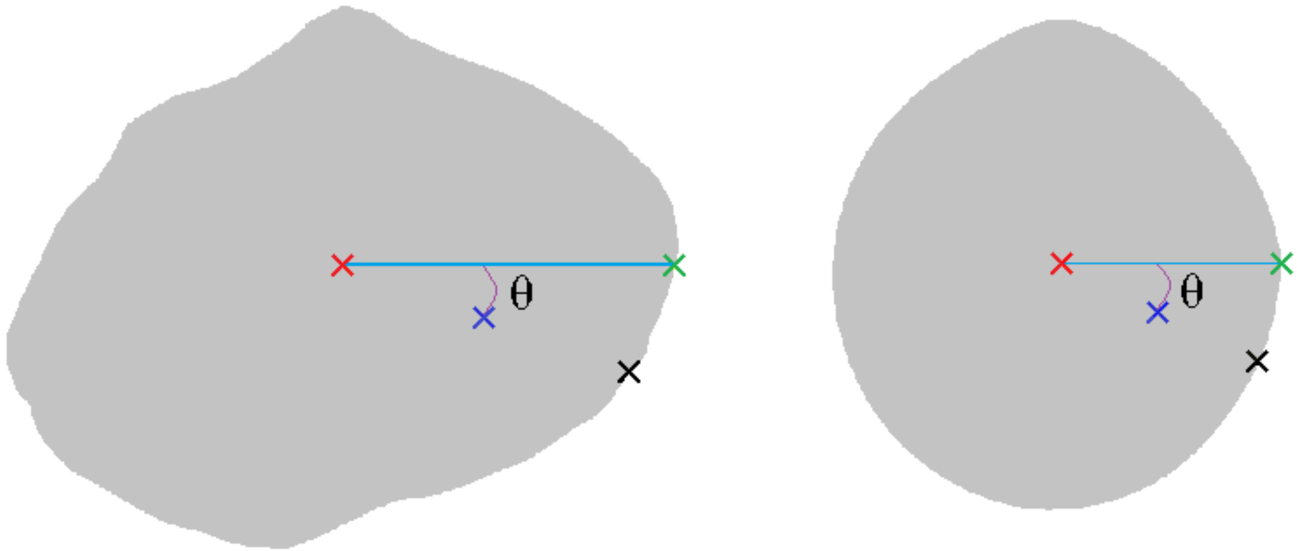
Population study: mean shape

- Karcher mean :

$$\hat{x} = arg \min_{x \in \mathcal{S}} \sum_{i=1}^{n} d^2(x_i, x)$$



Computation of a common reference space



Speed extrapolation