A 3D segmentation algorithm for ellipsoidal shapes. Application to nuclei extraction.

Emmanuel Soubiès, Pierre Weiss, Xavier Descombes

A 3D SEGMENTATION ALGORITHM FOR ELLIPSOIDAL SHAPES.
Application to nuclei extraction.

Emmanuel Soubies [1], Pierre Weiss [1], Xavier Descombes [2]
[1] *ITAV-UMS3039, Université de Toulouse, CNRS, Toulouse, France.*
[2] *MORPHEME team, INRIA/I3S/iBB, Sophia-Antipolis, France.*
esoubies@gmail.com, pierre.armand.weiss@gmail.com, xavier.descombes@inria.fr

Keywords: Nuclei segmentation, 2D and 3D images, graph-cuts, marked point processes, ellipses and ellipsoids, multiple object detection, multiple birth and cut, bio-imaging.

Abstract: We propose some improvements of the Multiple Birth and Cut algorithm (MBC) in order to extract nuclei in 2D and 3D images. This algorithm based on marked point processes was proposed recently in (Gamal Eldin et al., 2012). We introduce a new *contrast invariant* energy that is robust to degradations encountered in fluorescence microscopy (e.g. local radiometry attenuations). Another contribution of this paper is a fast algorithm to determine whether two ellipses (2D) or ellipsoids (3D) intersect. Finally, we propose a new heuristic that strongly improves the convergence rates. The algorithm alternates between two birth steps. The first one consists in generating objects uniformly at random and the second one consists in perturbing the current configuration locally. Performance of this modified birth step is evaluated and examples on various image types show the wide applicability of the method in the field of bio-imaging.

1 INTRODUCTION

Cell or nuclei segmentation in 2D and 3D is a major challenge in bio-medical imaging. New microscopes provide images at higher resolutions, deeper into biological tissues, leading to an increasing need for automatic cell delineation. This task may be easy in certain imaging modalities where images are well resolved and contrasted, but it remains mostly unresolved in emerging fluorescent microscopes dedicated to live imaging such as confocal, bi-photon, or selective plane illumination microscopes. These modalities suffer from multiple degradations such as light attenuation in the sample, heavy noise and spatially varying blur that make the segmentation task hard even for human experts

Our aim in this work is to propose a segmentation algorithm robust to such situations. Since images are heavily deteriorated, standard methods aiming at finding contours based on a sole regularity assumption such as active contours or Mumford-Shah derivatives fail for the segmentation. This observation led us to introduce strong shape priors: cells are modelled as ellipses or ellipsoids that should fit the image contents. Unfortunately, adding geometrical constraints makes the optimization problems highly non convex and appeal for the development of new global optimization methods.

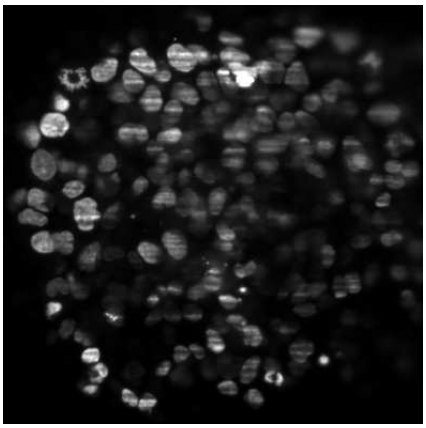

Figure 1: Example of a SPIM image (*Multicellular tumor spheroid*).

Following recent works (Descombes et al., 2009; Descombes, 2011; Gamal Eldin et al., 2012), we use randomized algorithms that allow to escape from local minima. These algorithms are based on marked point processes. The Marked Point Process (MPP) approach (Baddeley and Van Lieshout, 1993; Dong and Acton, 2007) consists in estimating a configuration of geometric objects (in our case ellipses or ellipsoids) whose number, location and shape are un-

known. It has proved to be very efficient in numerous image analysis applications as it allows the combination of radiometric information with strong geometrics constraints on the objects but also at a global scale. Defined by a density against the Poisson process measure, its main advantage is to consider a random number of objects and can be considered as an extension of the Markov Random Field approach. A review of this approach and its applications can be found in (Descombes, 2011).

The objects are defined on a state space $\chi = I \times M$ by their location and their marks (i.e. geometric attributes). The associated marked point process X is a random variable whose realisations are random configurations of objects. Considering a Gibbs process, the modeling consists of an energy construction. Similarly to the Bayesian framework, this energy can be written as the sum of a data term and a prior. In this paper we consider a pairwise interactions prior that forbids intersections between objects. Once the model defined, the solution is obtained by minimizing the energy. This energy being highly non-convex requires stochastic dynamics, such as MCMC methods, to be minimized. The Reversible Jump MCMC embeded in a simulated annealing framework is a natural candidate for this task (Green, 1995). However, in case of simple constraints such as non overlap, the recently proposed multiple birth and death algorithm is preferable (Descombes et al., 2009). To avoid the fastidious calibration of annealing parameters, we propose to revisit the combination of the multipe births principle with the graph cut paradigm proposed by (Gamal Eldin et al., 2012).

The paper is organized as follows. We formalize the segmentation problem as a minimization problem in section 2. Section 3 begins by a global algorithm description and is followed by a precise description of each algorithm step. We finish by presenting numerical results in section 4.

2 PROBLEM STATEMENT

Figure [1] contains typical examples of images encountered in biology. It is readily seen from these images that most nuclei contours can be well approximated by ellipses or ellipsoids, at least at a coarse scale. Moreover these nuclei cannot overlap due to obvious physical considerations. We thus formulate our segmentation problem as that of finding a set of non overlapping ellipsoids that fit the image contents. We formalize this statement in the latter.

Let $\mathcal{C}_n$, $n \in \mathbb{N}$ denote the set of configurations containing n objects that do not overlap. An element $\mathbf{x} \in \mathcal{C}_n$ is a set of n non overlapping objects. Since the number of nuclei in the configuration is unknown, we aim both at finding this number n^* and the best configuration $\mathbf{x} \in \mathcal{C}_{n^*}$ with respect to a certain data fidelity term $f(\mathbf{x})$. Our optimization problem can thus be formulated as follows. Let

$$g(n) = \min_{\mathbf{x} \in \mathcal{C}_n} f(\mathbf{x})$$

denote the minimum value of f in the set $\mathcal{C}_n$. We wish to find both

$$n^* = \arg\min_{n \in \mathbb{N}} g(n)$$

and

$$\mathbf{x}^* = \arg\min_{\mathbf{x} \in \mathcal{C}_{n^*}} f(\mathbf{x}).$$

By convention, we assume that $\mathcal{C}_0 = \emptyset$ and that $\min_{\mathbf{x} \in \mathcal{C}_0} f(\mathbf{x}) = 0$. The data term f should thus be negative for configurations that are likely to represent the nuclei parameters and positive otherwise. We detail how the ellipses are parametrized and the construction of such a function in the following paragraphs.

Object modelling In 2 dimensions, ellipses are parameterized using 5 parameters (see Figure 2):

- $(x,y) \in \Omega$: center coordinates which should belong to the image domain Ω.
- $\theta \in [0, 2\pi[$: angle with the horizontal direction.
- $0 < \lambda_- < b < a < \lambda_+$: describe the ellipses minor and major axes size. λ_- and λ_+ are user defined parameters that describe the nuclei maximal size and ellipticy.

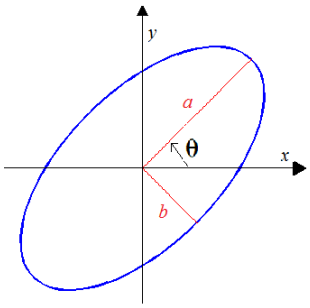

Figure 2: Parameters of the ellipse.

In 3 dimensions, nuclei are parameterized using 9 parameters:

- $(x,y,z) \in \Omega$: center coordinates.
- $\phi, \theta, \psi \in [0, 2\pi[^3$: Euler angles to define the ellipsoids orientations.
- $0 < \lambda_- < c < b < a < \lambda_+$: axes lengths.

Overall, it can be seen that objects belong to a state space χ defined as a parallelepiped:

$$\chi = \Omega \times [0, 2\pi[\times [\lambda_-, \lambda_+]^2 \quad (1)$$

in 2D and

$$\chi = \Omega \times [0, 2\pi[^3 \times [\lambda_-, \lambda_+]^3 \quad (2)$$

in 3D.

In this paper, the objects are denoted ω and their boundary is denoted $\partial\omega$.

Data term Let $u : \Omega \to \mathbb{R}$ denote a grayscale image. In order to define the data term $f(\mathbf{x})$, we associate an elementary energy $U_d(\omega, u)$ to each element $\omega \in \mathbf{x}$ and set:

$$f(\mathbf{x}) = \sum_{\omega \in \mathbf{x}} U_d(\omega, u). \quad (3)$$

The function $U_d(\omega, u) \in [-1, 1]$ should be negative if the object ω is well positioned on the image and positive otherwise.

In fluorescence microscopy, nuclei are usually characterized by bright region surrounded by a dark background since they are stained or genetically modified in order to express a fluorescent protein. Unfortunately, their radiometry is not constant due to local bleaching or light attenuation in the deepest layers. We thus need to construct an energy that is *contrast invariant*, meaning that local modifications of the radiometry shall not affect the energy. Such an energy can be constructed easily by considering the normal to the image level lines $\frac{\nabla u}{|\nabla u|}$ where ∇u denotes the usual gradient in $\mathbb{R}^d$ and $|\nabla u|$ denotes the gradient magnitude in the standard Euclidean norm. This tool is well known to be contrast invariant. Let us define an energy U for a given object ω as:

$$U(\omega) = \frac{1}{|\partial\omega|} \int_{\partial\omega} \langle \frac{\nabla u(x)}{\sqrt{|\nabla u(x)|^2 + \varepsilon^2}}, n(x) \rangle dx \quad (4)$$

where $\langle \cdot, \cdot \rangle$ denotes the standard scalar product, $|\partial\omega|$ denotes the length of the object boundary, $n(x)$ denotes the outward normal to ω at location $x \in \partial\omega$ and ε is a regularization parameter that discard faint transitions. The behavior of this energy is illustrated on Fig. 3. Overall, it does what is expected, but as can be seen on the illustration b) and d) in Fig. 3, badly located ellipses might have a negative energy and be kept in the final configuration. It is thus necessary to modify U in order to promote well located objects only. A simple way to do so consists in setting:

$$U_d(\omega, u) = \psi(U(\omega), s)$$

where $s \in]-1, 0]$ is an acceptance threshold for the objects. and

$$\psi(\alpha, s) = \min(\frac{1}{s+1}\alpha - \frac{s}{s+1}, 1).$$

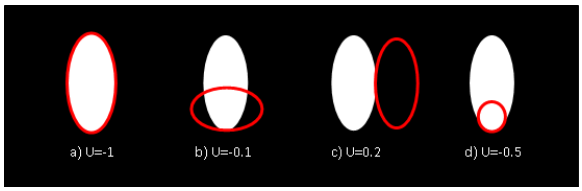

Figure 3

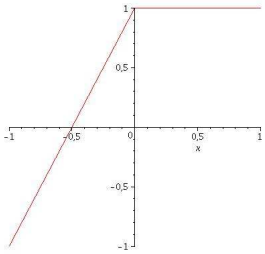

Figure 4: Graph of the function $\psi(\alpha, s)$ with respect to α for $s = -0.5$. Note that the function becomes positive for values of $\alpha > s$.

This function is illustrated on Figure 4

Other data terms based on the contrast between the object interior and the background as presented by (Gamal Eldin et al., 2012) (in dimension 2) could also be used but present two drawbacks: first they require to compute an integral over the interior of the domain while the proposed approach consist in computing a boundary integral which is faster. Second, such measures might be inaccurate in the case of very dense media, where the background can be difficult to extract. Finally our measure is contrast invariant, which is central for the targeted applications.

3 MULTIPLE BIRTH AND CUT ALGORITHM (MBC)

The Multiple Birth and Cut algorithm (MBC) has been proposed by (Gamal Eldin et al., 2012) for counting flamingos in a colony. In this section, we describe the different steps of the MBC algorithm (Algorithm 1).

The main idea consists in generating two random configurations of non-overlapping objects $\mathbf{x}$ and $\mathbf{x}'$ (birth step) and then keep the subset of objects in $\mathbf{x} \cup \mathbf{x}'$ that minimizes f (cut step). This process is iterated and decreases f at each iteration. The cut step can be performed efficiently using a Graph Cut algorithm (Boykov et al., 2001; Kolmogorov and Zabih, 2004). We describe this algorithm more formally below:

Interestingly, this algorithm contains only one parameter N (the number of objects generated in a con-

Algorithm 1 Multiple Birth & Cut algorithm

Require: N
1: Generate a configuration $\mathbf{x}_{[0]}$ with Algorithm 2
2: $n \longleftarrow 0$
3: **while** (Not converged) **do**
4: Generate of a new configuration $\mathbf{x}'$ using Algorithm 2.
5: $\mathbf{x}_{[n+1]} \longleftarrow Cut(\mathbf{x}_n \cup \mathbf{x}')$
6: $n \longleftarrow n+1$
7: **end while**

figuration). We observed experimentally that this parameter might affect slightly the speed of convergence but not the segmentation accuracy. This algorithm is thus much easier to tune than more standard RJMCMC based dynamics.

3.1 Birth step

A new configuration $\mathbf{x}'$ of non-overlapping objects is generated. Note that only objects which are in the same configuration have to respect the non-overlapping constraint, but two objects in different configurations can intersect as can be seen on Figure 5.

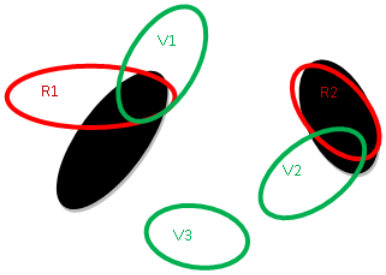

Figure 5: Two configuration on an image (the black ellipses are the object to detect)

The birth step is detailed in Algorithm 2. The fourth step of this algorithm can be efficiently implemented using a lookup table and the fast intersection algorithm proposed in the latter.

3.2 Cut step

This step consists in selecting the best configuration of non-overlapping objects in $(\mathbf{x}_{[n]} \cup \mathbf{x}')$. To perform this optimization, a weighted graph is constructed. The nodes of this graph are the objects ω of the two configurations $\mathbf{x}_{[n]}$ and $\mathbf{x}'$. This graph also possesses

Algorithm 2 Birth step

Require: N, n_{max}.
1: Set $k = 0$, $n = 0$, $\mathbf{x}' = \emptyset$.
2: **while** $k < N$ and $n < n_{max}$ **do**
3: Construct an object ω' by generating a random vector uniformly in χ.
4: If ω' intersects an object in $\mathbf{x}'$, set $n = n+1$ and go back to 3.
5: Otherwise set $\mathbf{x}' = \mathbf{x}' \cup \{\omega'\}$, $k = k+1$, $n = 0$ and go back to 3.
6: **end while**

two special nodes, the source '**s**' and the sink '**t**'. The weights should belong to $[0,1] \cup \{+\infty\}$ and are defined using the data term $U_d(\omega, u)$ by:

$$W(\omega) = (1 - U_d(\omega, u))/2.$$

Graph construction

Each object of the configuration $(\mathbf{x}_{[n]} \cup \mathbf{x}')$ is linked to the source and the sink. The difference between the objects $\omega_i \in \mathbf{x}_{[n]}$ and the objects $\omega_j \in \mathbf{x}'$ is that the objects $\omega_i \in \mathbf{x}_{[n]}$ are linked to the source with a weight equal to the data energy $W(\omega)$ and to the sink with a weight equal to $1 - W(\omega)$, while it is the reverse for the objects $\omega_j \in \mathbf{x}'$.

The weights associated to edges linking two objects are non zero only when two objects intersect. If $\omega_1 \in \mathbf{x}_{[n]}$ (current configuration) intersects with $\omega_2 \in \mathbf{x}'$ (new configuration), the link from ω_1 to ω_2 is set to ∞ and the link from ω_2 to ω_1 is set to zero [1]. This ensures that the cut step generates an admissible configuration (with no overlapping objects). Figure 6 summarises the graph construction of the configurations on Figure 5. The nuclei to detect are represented by black ellipses.

Cut

Once the graph is constructed, we perform a cut that consists in partitioning the vertices into two disjoint subsets. One contains the source and the other the sink. The cut realized is the one with minimal cost (the one minimizing the sum of the weights of the removed edges).

After the cut step, if $\omega_i \in \mathbf{x}_{[n]}$ is in the sub-graph containing the source, we keep it, otherwise we remove it. On the contrary the objects $\omega_j \in \mathbf{x}'$ are

[1] When two objects intersect the link affected by a weight of ∞ is always the link from the object of the current configuration to the object of the new configuration.

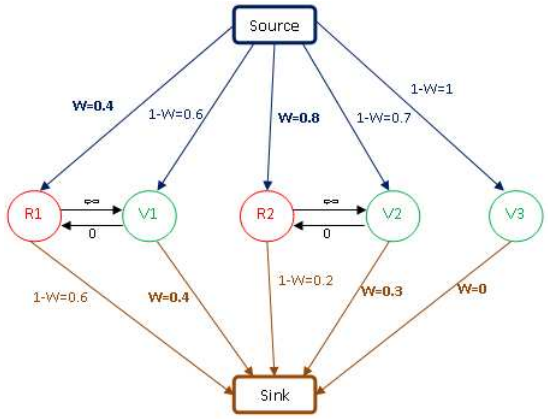

Figure 6: Graph corresponding to the figure 5

kept only if they belong to the sub-graph that contains the sink. This difference of interpretation between the two configurations combined with the different weights to the source and the sink, ensure that in case where an object of $\mathbf{x}_{[n]}$ and an object of $\mathbf{x}'$ intersect, only one can be kept.

The cut step is implemented using the graph-cut code developed by Yuri Boykov and Vladimir Kolmogorov in (Boykov et al., 2001; Boykov and Kolmogorov, 2004; Kolmogorov and Zabih, 2004).

3.3 A FAST DETERMINATION OF ELLIPSES INTERSECTION

One of the proposed algorithm bottleneck is the fast determination of whether two ellipsoids intersect or not. In this section, we present a fast algorithm to answer that question and prove theoretically that only a few arithmetic operations suffice to provide the answer with a low error rate.

Let ω be an ellipse or an ellipsoid. It can be defined using a quadratic function Q_ω as $\omega = \{x \in \mathbb{R}^d, Q_\omega(x) = 1\}$. The quadratic function Q can be defined by:

$$Q_\omega(x) = \langle A(x-c),(x-c)\rangle \tag{5}$$

where c denotes the object center and A is positive definite matrix defined by:

$$A = P^{-1}DP = P^T DP.$$

where P is a rotation matrix and D is a positive diagonal matrix. In 2D, P is defined by:

$$P = \begin{pmatrix} cos(\theta) & sin(\theta) \\ -sin(\theta) & cos(\theta) \end{pmatrix}$$

and

$$D = \begin{pmatrix} \frac{1}{a^2} & 0 \\ 0 & \frac{1}{b^2} \end{pmatrix}.$$

In 3D, the notation become cumbersome and we leave them to the reader.

Let ω_1 and ω_2 be two ellipses or ellipsoids. In order to know whether they intersect or not, we can find the minimal level set of Q_{ω_2} which intersects the boundary of ω_1. If this level set is associated to a value greater than 1, the ellipses are separated, otherwise they overlap. This idea can be formulated as the following minimization problem:

$$\min_{x\in\mathbb{R}^d, Q_{\omega_1}(x)\leq 1} Q_{\omega_2}(x) \tag{6}$$

This problem consists of minimizing a quadratic function over convex set. Projected descent methods can thus be used. Unfortunately, there exists no closed form solution to the problem of projection of a point on an ellipse. We thus need to simplify the constraint set:

$$\begin{aligned}
&\min_{Q_{\omega_1}(x)\leq 1} Q_{\omega_2}(x) \\
&= \min_{\langle A_1(x-c_1),(x-c_1)\rangle\leq 1} \langle A_2(x-c_2),(x-c_2)\rangle \\
&= \min_{\langle \sqrt{A_1}(x-c_1),\sqrt{A_1}(x-c_1)\rangle\leq 1} \langle A_2(x-c_2),(x-c_2)\rangle \\
&\min_{\|y-\sqrt{A_1}c_1\|_2^2\leq 1} \langle A_2(A_1^{-\frac{1}{2}}y-c_2),(A_1^{-\frac{1}{2}}y-c_2)\rangle.
\end{aligned}$$

where $y = \sqrt{A_1}x$. In this reformulation, the constraint set $Y = \{y \in \mathbb{R}^d, \|y-\sqrt{A_1}c_1\|_2^2 \leq 1\}$ is a simple l^2-ball and the function $F(y) = \langle A_2(A_1^{-\frac{1}{2}}y - c_2),(A_1^{-\frac{1}{2}}y - c_2)\rangle$ is a strongly convex differentiable function. We can thus use a projected gradient descent that writes:

Algorithm 3 Detection of overlapping ellipsoids

Require: $Q_{\omega_1}, Q_{\omega_2}, \varepsilon > 0$.
1: Set $k = 0$, $y_0 = \frac{c_1+c_2}{2}$.
2: Set $\mu = \frac{b_1^2}{a_2^2}$, $L = \frac{a_1^2}{b_2^2}$.
3: Set $\tau = \frac{2}{\mu+L}$.
4: **while** $\|y_{k+1} - y_k\| \geq \varepsilon$ **do**
5: $\quad y_{k+\frac{1}{2}} = y_k - \tau\nabla F(y_k)$.
6: $\quad y_{k+1} = \Pi_Y\left(y_{k+\frac{1}{2}}\right)$.
7: $\quad k = k+1$.
8: **end while**
9: If $F(y_k) >= 1$ return 0 (the ellipsoids do not intersect with high probability).
10: If $F(y_k) < 1$ return 1 (the ellipsoids intersect).

Let y^* denote the solution of the above problem. The previous algorithm comes with the following guarantees:

Theorem 1. *After k iterations, y_k satisfies:*

$$F(y_k) - F(y^*) \leq \frac{\mu}{2}\|y_0 - y^*\|_2^2 \left(\frac{Q_F - 1}{Q_F + 1}\right)^{2k}$$

$$\|y_k - y^*\|_2^2 \leq \|y_0 - y^*\|_2^2 \left(\frac{Q_F - 1}{Q_F + 1}\right)^{2k}$$

where

$$Q_F = \frac{a_1^2}{b_2^2}\frac{a_2^2}{b_1^2} \leq \frac{\lambda_+^4}{\lambda_-^4}.$$

Proof The Hessian of F is $H_F(y) = 2A_1^{-\frac{1}{2}} A_2 A_1^{-\frac{1}{2}}$. Since A_1 and A_2 are products of orthogonal and diagonal matrices ($A = P^T DP$), the eigenvalues of $H_F(y)$ can be easily bounded:

$$\lambda_{min}[H_F(y)] \geq \frac{b_1^2}{a_2^2} \qquad \lambda_{max}[H_F(y)] \leq \frac{a_1^2}{b_2^2}$$

The function F is thus μ-strongly convex with $\mu \geq \frac{b_1^2}{a_2^2}$ and its gradient is L-Lipschitz with $L \leq \frac{a_1^2}{b_2^2}$. Using standard convergence theorems in convex analysis (Bertsekas, 1999), we obtain the announced result. □

The conditioning number Q_F depends solely on the ratio between the major axis and the minor axis sizes and not on the dimension d. This algorithm will thus be as efficient in 3D as in 2D. For two circles the ratio Q_F is equal to $\frac{a}{b} = 1$ and the algorithm provides the exact answer after one iteration. For elliptic ratios of 2, $Q_f = 16$ and in the worst case, after 18 iterations, the algorithm returns a point y^k that is 100 times closer to the intersection than y_0. We also tested an accelerated algorithm by (Nesterov, 2004), where the convergence rate is of order $\left(\frac{\sqrt{Q_F}-1}{\sqrt{Q_F}+1}\right)^{2k}$ but it did not improve the computing times.

In our problems the ratio between a and b is always less than 2 and the algorithm usually converges in just a few iterations (2 to 10 depending on the problem).

3.4 Acceleration by local perturbations

When the objects variability is important, the state space size increases and affects the convergence speed of the MBC algorithm. This problem is particulary important in 3D since ellipsoids are defined by 9 parameters instead of 5 for the 2 dimensional case.

In order to improve the convergence speed, (Gamal-Eldin et al., 2011) proposed to insert a *selection phase* in the birth step. This selection phase consists in generating a dense configuration of objects at similar locations and to keep the best ones using Belief Propagation in order to form the new configuration.

In this paper, we propose another heuristic in order to increase the convergence speed. We propose to alternate between two different kinds of birth steps. The first one is that proposed in algorithm 2. The second one consists in perturbating locally the current configuration. This principle mimics the proposition kernels used in RJMCMC algorithms (Perrin et al., 2005). The idea behind this modification is that after a while, most objects are close to their real location and that local perturbations may allow much faster convergence than fully randomized generation. This algorithm is described in details in Algorithm 4.

Algorithm 4 MBC algorithm with local perturbations

Require: N
Generate a configuration $\mathbf{x}_{[0]}$ using Algorithm 2.
$n \longleftarrow 0$
while (Not converged) **do**
  Generate a uniformly distributed random number $r \in [0,1]$.
  if $r <$ **p then**
    Generate a new configuration $\mathbf{x}'$ using Algorithm 2.
  else
    Generate a new configuration $\mathbf{x}'$ using Algorithm 5.
  end if
  $\mathbf{x}_{[n+1]} \longleftarrow Cut(\mathbf{x}_n \cup \mathbf{x}')$
  $n \longleftarrow n+1$
end while

Algorithm 5 Birth step with local perturbation

Require: $\mathbf{x}_{[n]}$.
while $k < size(\mathbf{x}_{[n]})$ **do**
  Construct an object ω' by local perturbation of $\omega_k \in \mathbf{x}[n]$.
  If ω' intersects an object in $\mathbf{x}'$, set $k = k+1$ and go back to 2.
  Otherwise set $\mathbf{x}' = \mathbf{x}' \cup \{\omega'\}$, $k = k+1$ and go back to 2.
end while

Local perturbations

A given object ω in $\mathbf{x}_{[n]}$ is described by a set of parameters $\lambda \in \chi$ (see equations 1 and 2). We generate the new object ω' by setting its parameters $\lambda' = \lambda + z$ where z is the realization of a random vector Z dis-

tributed uniformly in χ_ε where :

$$\chi_\varepsilon = [-\delta_{xy}, \delta_{xy}]^2 \times [-\delta_{ab}, \delta_{ab}]^2 \times [0, 2\pi[$$

in 2D and

$$\chi_\varepsilon = [-\delta_{xyz}, \delta_{xyz}]^3 \times [-\delta_{abc}, \delta_{abc}]^3 \times [0, 2\pi[^3$$

in 3D.

The value of the different δ describes the perturbation extent. We observed that small values accelerates the convergence speed.

Comparison of the convergence speed

We have tested this method in order to compare the speed of convergence of the MBC algorithm and the MBC algorithm with local perturbation. Figure 7 presents the energy evolution with respect to time for both MBC and MBC with local perturbations (denoted MBC with LP) on the same image (the 3D nuclei of Drosophila embryo). The segmentation result is presented on Figure 14 (image size $700 \times 350 \times 100$). These results show that the MBC with LP algorithm strongly improve the MBC algorithm.

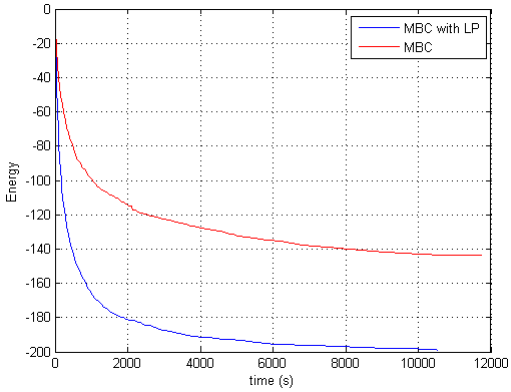

Figure 7: Comparison of the MBC and MBC with LP algorithms

4 RESULTS

In this section, we present some practical results in 2D and 3D. Figure 9 shows the segmentation result on a Drosophila embryo obtained using SPIM imaging. This is a rather easy case, since nuclei shapes vary little. The images are impaired by various defects: blur, stripes and attenuation. Despite this relatively poor image quality, the segmentation results are almost perfect. The computing time is 5 minutes using a c++ implementation. The image size is 700×350.

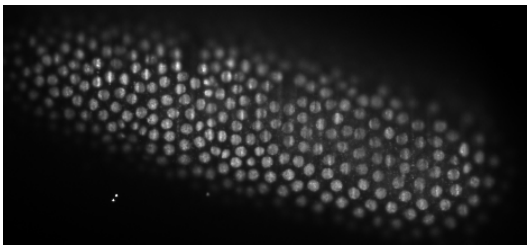

Figure 8: 2D nuclei of Drosophila embryo

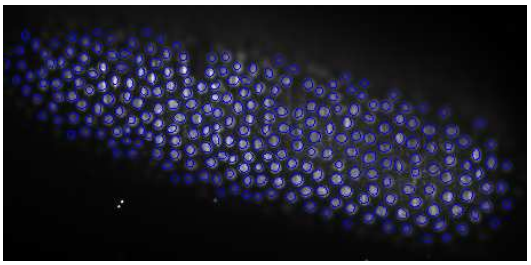

Figure 9: 2D segmentation of a nuclei of Drosophila embryo (Fig 8)

Figure 10 presents a more difficult case, where the image is highly deteriorated. Nuclei cannot be identified in the image center. Moreover, nuclei variability is important meaning that the state space size χ is large. Some nuclei are in mitosis (see e.g. top-left). In spite of these difficulties, the MBC algorithm provides acceptable results. They would allow to make statistics on the cell location and orientation, which is a major problem in biology. The computing times for this example is 30 minutes.

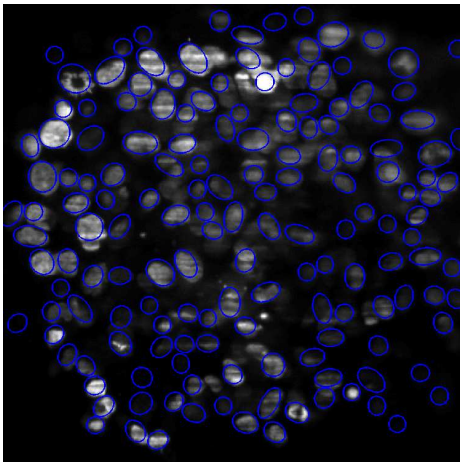

Figure 10: 2D segmentation of a multicellular tumor spheroid (Fig 1)

Nuclei segmentation is a major open problem with a large number of other applications. In Figure 11, we attempt to detect the spermatozoon heads. The foreseen application is tracking in order to understand

their collective motion. Figure 12 presents a multicellular spheroid surrounded by a circle of pillars. A typical problem consists in determining whether the pillars have moved afetr a while, due to the tumor dynamic. They thus need to detect the pillars and evaluate their motion. Figure 11 and Figure 12 show that the segmentation results are extremely precise.

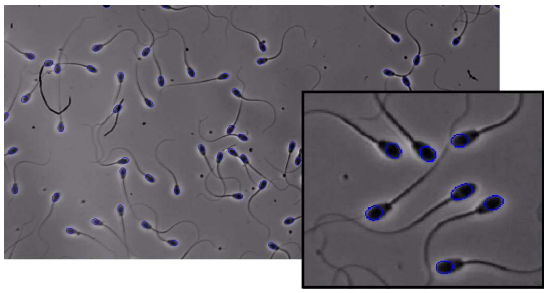

Figure 11: Segmentation of a spermatozoon colony (5 minutes). Image size: 2000 x 1024.

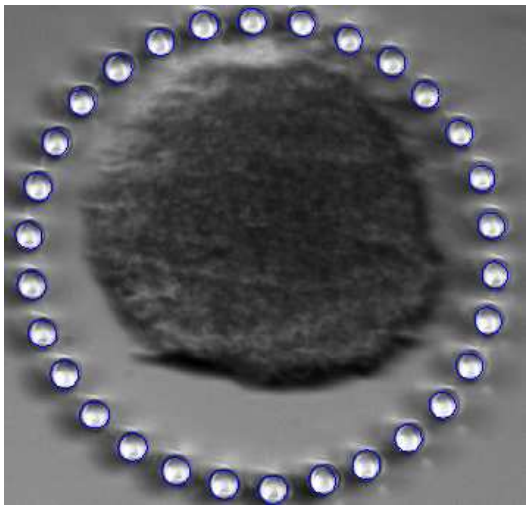

Figure 12: Nano pillars detection (less than 1 minute). Image size: 840 x 800.

3D results are presented in Figure 14 and 16. For the Drosophila embryo, the segmentation is very close to what a human expert would do. The computing times are 2 hours and the image size is $700 \times 350 \times 100$. The curves in Figure 14 correspond to this image.

The spheroid segmentation presented in Figure 16 is less precise than the previous ones due to an important cell variability and to the fact that the images are extremly blurry in the Z direction. For that case, image restoration algorithms and the design of new energies robust to strong perturbations seem important.

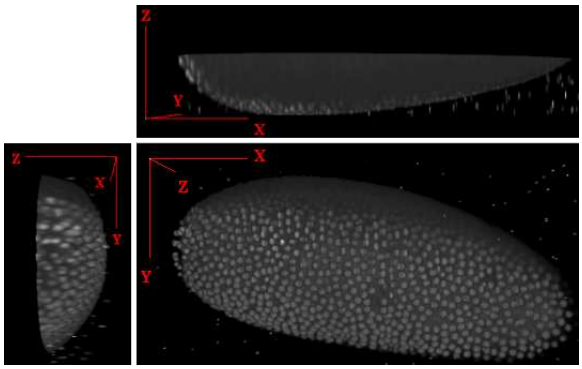

Figure 13: 3D Drosophila embryo nuclei

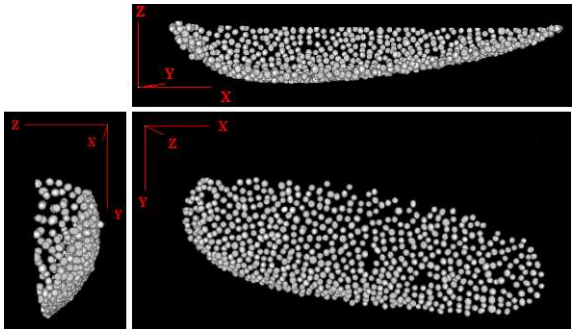

Figure 14: 3D segmentation of the Drosophila embryo nuclei (Fig 13)

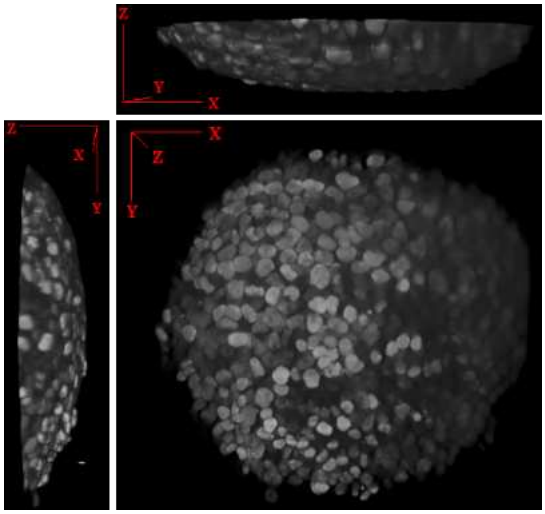

Figure 15: 3D multicellular tumor spheroid

ACKNOWLEDGEMENTS

This work was partially funded by the Mission pour l'interdisciplinarité from CNRS, Région Midi Pyrénées, PEPII CASPA3D and ANR SPHIM3D.

The authors wish to thank F. Malgouyres and J. Fehrenbach for interesting discussions. They also thank Valérie Lobjois, Charlotte Emery, Jacques

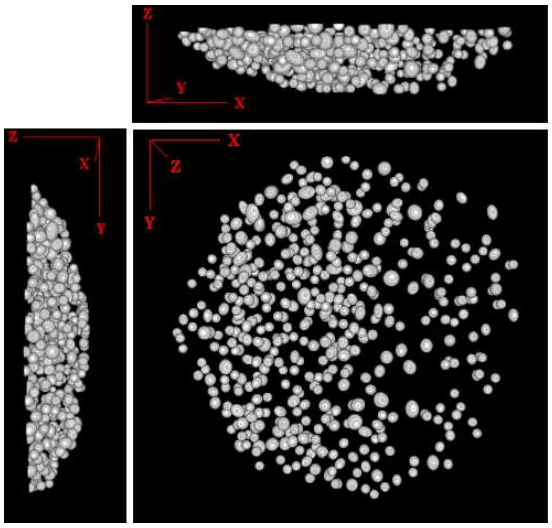

Figure 16: 3D segmentation of the multicellular tumor spheroid (Fig 15)

Thomazeau, Paul Escande and Bernard Ducommun for their help in this project. They thank all the ITAV staff for their warm welcome in a biology laboratory.

REFERENCES

Baddeley, A. and Van Lieshout, M. (1993). Stochastic geometry models in high-level vision. *Journal of Applied Statistics*, 20(5-6):231–256.

Bertsekas, D. (1999). Nonlinear programming.

Boykov, Y. and Kolmogorov, V. (2004). An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(9):1124–1137.

Boykov, Y., Veksler, O., and Zabih, R. (2001). Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23:1222–1239.

Descombes, X. (2011). *Stochastic geometry for image analysis*. Wiley/Iste, x. descombes edition.

Descombes, X., Minlos, R., and Zhizhina, E. (2009). Object extraction using a stochastic birth-and-death dynamics in continuum. *Journal of Mathematical Imaging and Vision*, 33(3):347–359.

Dong, G. and Acton, S. (2007). Detection of rolling leukocytes by marked point processes. *Journal of Electronic Imaging*, 16:033013.

Gamal-Eldin, A., Descombes, X., Charpiat, G., and Zerubia, J. (2011). A fast multiple birth and cut algorithm using belief propagation. In *Image Processing (ICIP), 2011 18th IEEE International Conference on*, pages 2813–2816. IEEE.

Gamal Eldin, A., Descombes, X., Charpiat, G., Zerubia, J., et al. (2012). Multiple birth and cut algorithm for multiple object detection. *Journal of Multimedia Processing and Technologies*.

Green, P. J. (1995). Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 82:711–732.

Kolmogorov, V. and Zabih, R. (2004). What energy functions can be minimized via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:65–81.

Nesterov, Y. (2004). *Introductory lectures on convex optimization: A basic course*, volume 87. Springer.

Perrin, G., Descombes, X., and Zerubia, J. (2005). A marked point process model for tree crown extraction in plantations. In *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, volume 1, pages I–661. IEEE.