Bayesian Object Recognition with Baddeley's Delta Loss

Håvard Rue* and Anne Randi Syversveen
Department of Mathematical Sciences
NTH, Norway

October 30, 1995

Abstract

A common problem in Bayesian object recognition using marked point process models is to produce a point estimate of the true underlying object configuration. In the Bayesian framework we could use decision theory and the concept of loss functions to design a more reasonable estimator for the true underlying object configuration than the common zero–one loss, which corresponds to the *maximum a–posteriori* estimator. We propose to use the Δ–metric of Baddeley (1992) as our loss function. It is demonstrated that the optimal Bayesian estimator corresponding to the Δ–metric can be well approximated by combining Markov chain Monte Carlo methods with Simulated Annealing into a two–step algorithm. The proposed loss function is tested using a marked point process model developed for locating cells in confocal microscopy images. In order to obtain reliable results, we must include moves that split and fuse objects within the Markov chain Monte Carlo framework. The experimental results obtained with the proposed loss function seem to yield more stable and visually better estimates of the shape of each object and the total number of objects, compared to the common *maximum a–posteriori* estimator. Finally, the ideas are applied to locate cells in a confocal microscopy image.

Keywords: Bayesian inference; Unsymmetric loss functions; Object Recognition; Template models; Markov chain Monte Carlo methods; Marked point processes; Distance between images; Confocal Microscopy Images

*Address for correspondence: Department of Mathematical Sciences, The Norwegian Institute of Technology, N–7034 Trondheim, Norway.
E–mail: havard.rue@imf.unit.no and annerand@imf.unit.no

1 Introduction

Bayesian object recognition is the problem of how to estimate the number of objects and their location in a non–ideal environment. The difficulty varies with the degree of variation in size and shape of the objects, the number of types of objects present, and if they are allowed to overlap or not. One relatively easy example is to locate nearly circular cells on a slowly varying background, as in the confocal microscopy images we will study later. A more complex example is the estimation of location, orientation and shape of mitochondria and membranes in electron micrographs of cardiac muscle cells, studied by Grenander and Miller (1994). The objects have large variability in shape, and the images contain several types of objects. Industrial and practical applications can be found, within document reading and robot vision.

Object recognition belongs to the field of high–level imaging, for which the image modeling is on a more global scale and directly connected to the object types in mind compared to low–level imaging which deals with (smoothing) prior models on a pixel level. Over the last years, there has been a growing interest in this field; especially along the guidelines of Grenander's general pattern theory using deformable templates. The original study is available in Grenander, Chow, and Keenan (1991). The recent discussion paper by Grenander and Miller (1994) represents the "state of the art" of this approach, and points out future directions for the practical use of general pattern theory, now collected in a monograph (Grenander 1993).

A template can be thought of as a continuous model for a single object; typically a polygonal silhouette defining the contour of the object. The prior model focuses on how the object can be deformed from an ideal or typical outline, by effecting the length and orientation of the boundary segments in case of a polygon. The observed data is most often an image with discrete pixel values available on a lattice. Even for perfect data, the (continuous) object of interest is only available in a discretized version. Combining the prior model and the likelihood, which contains the information from the data, we obtain the posterior distribution for the object. The posterior distribution is used in any further inferential questions concerning the object within the Bayesian paradigm. Often, this step relies heavily on iterative simulation methods like Markov chain Monte Carlo (MCMC) methods or diffusion simulation (Grenander and Miller 1994). (Refer to Tierney (1994) or Besag *et al.* (1995) for a general discussion of MCMC methods and to Geyer and Møller (1994) for application to point processes.) Examples include a study of the silhouette of a hand (Grenander, Chow, and Keenan 1991), a more challenging situation with a grey–level image of a hand (Amit, Grenander, and Piccioni 1991), detection of defects in potatos (Grenander and Manbeck 1993), finding spiral structures in images of galaxies (Ripley and Sutherland 1990), and modeling human faces (Phillips and Smith 1994).

Lately, template models have been extended to handle the case when the number of objects is not known *a priori*; that is, the parameter space is a union of subspaces with varying dimension. Two different approaches exist. Grenander and Miller (1994) suggest using multiple–graph deformable templates and jump–diffusion simulation. Baddeley and van Lieshout (1991, 1993) and Baddeley (1994) argue for using marked point processes (mpp), and MCMC or birth–death methods for simulation. The shape of each object is described by a mark, and the location of the object by a point. Markov point processes model interaction between objects, and a deformable template models each object. The two approaches seem to differ in complexity with an order of magnitude. In this paper we will therefore concentrate on the mpp–approach which seems well suited for our main objective: To obtain a good point estimate for object recognition models, by using decision theory and the concept of loss functions.

Why is the concept of loss functions important within Bayesian object recognition? Assume for a moment that we will perform a simulation study for our object recognition model with a known true scene. After discretizing the continuous objects and adding noise to the image, we want to obtain a point estimate for the shape, size, location and the number of objects. Suppose that some estimates are available; for example, the *maximum a–posterior* (MAP) estimate, approximated by the mode of the posterior distribution determined by a stochastic search procedure like Simulated Annealing, and estimates found by applying other variants of steepest ascent and coordinatewise steepest ascent algorithms. We want to quantify the quality of the estimates by comparing to the known true scene. For this purpose, we use a distance measure or metric d. When the true scene is not known, we can compare the posterior expectation of d for the different estimates. We then use d implicitely as our loss function, since this is how we have chosen to quantify the distance between different scenes. By minimizing the posterior expectation of the loss d (the risk), we obtain the *optimal Bayes estimator* (OBE) with respect to d. This gives a lower value for the risk than using for example the MAP–estimate which is OBE with respect to the 0–1 loss function. It should be noted that an important assumption is made: We implicitly assume that we are able to compute the OBE for d. This may not always be possible. An example of the described scenario, is the simulation examples in Baddeley and van Lieshout (1991) and van Lieshout (1994), where different estimates are computed and the quality measured by using Baddeley's Δ–metric (Baddeley 1992). The main contribution in our work is actually to demonstrate how we can compute (approximately) the OBE with the Δ–metric as our loss function. In this way, we will then be able to do a more coherent inference in object recognition models within the Bayesian framework.

The experience from low–level imaging is that more sensible loss functions than those leading to MAP, the posterior marginal mean and mode may give more reasonable estimates. See for example the results in Rue (1995b) and Frigessi and Rue (1995) for image classification (which seems most promising) and Rue (1995a) for grey–scale images. The proposed loss functions have a local structure, for which the OBE can be well approximated by a two–step

MCMC and Simulated Annealing algorithm described in Rue (1995b).

Baddeley's Δ–metric is of particular interest for object recognition, and has recently been applied as a loss function in image classification (Frigessi and Rue 1995). The Δ–metric measures the distance between images of (discrete or continuous) objects, and has several appealing theoretical properties. It is a good distance measure between images of objects and (surprisingly) the computation of the OBE is relatively easy. Even if the inferential part is the main objective in this paper, we will postpone the discussion of the Δ–metric and inference to Section 4 because we find that it is easier to describe the algorithm when we have one particular mpp–model in mind. To start with, we will therefore in Section 2 present a simple mpp–model for location of certain cells in confocal microscopy images. In Section 3 we discuss how to efficiently jump between different parameter subspaces, by allowing the objects to split and fuse within the MCMC framework. This is important in the application of our model to confocal images. In Section 4 we discuss inference, and in Section 5 we present experiments with the proposed mpp–model, and compare the MAP–estimator with the Δ–estimator.

2 The Model

This section describes the marked point process (mpp) model we propose to use for locating cells in confocal microscopy images. The model is motivated from the cells or objects similar to those in the 365×387 image shown in Figure 1. The model has some similarity with the mitochondria–model in (Grenander and Miller 1994), but is simpler both in form and in the interpretation of the parameters. This simplification is possible because the cells are roughly circular. We will first describe the model for one object and then show how to integrate this into a mpp–model to handle an unknown number of objects with different size, shape, and location.

To describe our prior model for one object, choose hte origin as the center of the object. As a first approximation, the object is assumed to be circular with stochastic radius $r > 0$, where the density of r is proportional to a Gaussian density with mean μ_r and variance σ_r^2. To allow for local deviation from the circle we divide the circle into n_z line segments as displayed in Figure 2 and introduce $\mathbf{z} = (z_1, \ldots, z_{n_z})$. Each z_i describes deviation from the assumed circle along a line with angle $2\pi(i-1)/n_z$ starting from the center of the object. We take $\mathbf{z}$ to be a Gaussian vector conditionally on r, that is

$$\mathbf{z}|r \sim \mathrm{N}(\mathbf{0}, \sigma_z^2(r)\mathbf{\Sigma}).$$

It is natural to choose $\mathbf{\Sigma}$ to be a Toeplitz circulant matrix with ones on the diagonal, to reflect invariance and stationary properties of a cell. The deviation of an object from a circle is slowly varying with an exponentially decaying correlation function $\rho_z(\tau)$ for $0 \leq \tau < 1/2$, and $\rho_z(1-\tau)$ for $1/2 \leq \tau < 1$, i.e.

$$(\mathbf{\Sigma})_{1j} = \rho_z\left(\min(j-1, n_z-j+1)/n_z\right), \quad j = 1, \ldots, n_z\,.$$

The function $\rho_z(\tau)$ will be given later when needed. Let $\nu(\mathbf{z}, r) = \nu(\mathbf{z}|r)\,\nu(r)$ be the density of the object. In order to separate the interpretation of r and $\mathbf{z}$, we choose

$$\sigma_z^2(r) = \sigma_z^2\, r^2\,.$$

such that the radius r is a hyperparameter that controls the overall size and shape, and $\mathbf{z}$ describes the local variability. This choice ensures that a scaling of an object is invariant in $\mathbf{z}$, since

$$\frac{\nu(\mathbf{z}, r)}{\nu(\kappa\mathbf{z}, \kappa r)} = \kappa^{n_z/2}\frac{\nu(r)}{\nu(\kappa r)}, \quad \kappa > 0,$$

only depends on r and κ.

The next step is to include the model for one object into our mpp–model; refer to Baddeley and Møller (1989), Baddeley and van Lieshout (1993) or van Lieshout (1995) for further details. Each object x has a location $s = (s_x, s_y) \in S$, size and shape $m = (r, \mathbf{z}) \in M$. An object configuration is a finite unordered set $\mathbf{x} = \{x_1, \ldots, x_n\}$ of objects $x_i = (s_i, m_i) \in U$, and let Ω denote the set of all configurations. For our case, $U = S \times M$, where S is a finite subset of the plane $\mathcal{R}^2$ and $M = \mathcal{R}^+ \times \mathcal{R}^{n_z}$. A marked point process, or object process, is a point process on the class of objects U. The basic reference model is the Poisson object process in U. Let $\mu = \lambda \otimes \nu$ be a finite non–atomic measure on U, where λ is the Lebesgue measure on S with $\lambda(S) < \infty$ and ν is the probability measure on M with $\nu(M) = 1$. Under the Poisson model, the number of objects has a Poisson distribution with mean $\mu(U)$. Conditionally on n objects, each object is distributed uniformly in U. The joint probability density of $(x_1, \ldots, x_n)$ and n is defined by the density f relatively to the Poisson object process, for instance by the pairwise interaction model

$$f(\mathbf{x}, n) = \alpha\beta^n \prod_{i<j} g(x_i, x_j)\,.$$

The normalization constant is α and $g(x_i, x_j)$ is the interaction function. van Lieshout (1995) gives an overview and discusses suitable interaction models.

We will try to keep the model simple to make the interpretation of the examples in Section 5 with the new loss function as clear as possible. We do not model any interaction between objects apart from the non–overlapping property of the cells in Figure 1. We choose

$$g(x_i, x_j) = \begin{cases} 0, & \text{if object } x_i \text{ and } x_j \text{ overlap,} \\ 1, & \text{otherwise,} \end{cases}$$

so our prior model becomes a hard core object process. We will comment on the importance of the interaction between objects in Section 6. To summarize, the prior model for $\mathbf{x}$ (and n) is

$$p(\mathbf{x}) = \alpha\beta^n \exp(-\lambda(S)) \prod_i \nu(m_i) \prod_{i>j} g(x_i, x_j)\,.$$

Our data is an imperfect image of discretized objects, $\{y_t, t \in T\}$, where T is a lattice and y_t is the pixel–value at pixel t, typically binary or byte valued. We assume that y_t are conditionally independent given $\mathbf{x}$, so the likelihood reads

$$l(\mathbf{y}|\mathbf{x}) = \prod_{t \in T} l_t(y_t|\mathbf{x}). \tag{1}$$

For object recognition models, the likelihood is not uniquely defined. This is mainly because our model is defined on properties of the images, such as the number of objects and a continuous object shape, and not on a pixel basis where the data is recorded. One common approach is to define the silhouette $\{R(t;\mathbf{x}), t \in T\}$ to be a binary image, and $R(t;\mathbf{x})$ is zero if t is outside all objects and one if t is inside one object. The silhouette can be used to define the likelihood. Assume y_t is a Gaussian variable with mean ξ_0 and variance σ_0^2 in the background area, and mean ξ_1 and variance σ_1^2 in the foreground area. Then the likelihood is

$$l(\mathbf{y}|\mathbf{x}) = \prod_{t \in T} \frac{1}{\sqrt{2\pi}\sigma_{R(t;\mathbf{x})}} \exp -\frac{1}{2}\left(\frac{y_t - \xi_{R(t;\mathbf{x})}}{\sigma_{R(t;\mathbf{x})}}\right)^2. \tag{2}$$

We have used this model in the first two simulation experiments in Section 5. For the confocal images like Figure 1 this likelihood model is a little crude, since the mean level is reasonably constant within each cell, but seems to vary between cells. A nice approach is to expand the mark model to carry an additionally variable ξ on each object, $m = (\mathbf{z}, r, \xi)$, where the density $p(\xi)$ has support in D_ξ say. Since we cannot distinguish cells with level near ξ_0 from the background, we choose D_ξ and a (not too small) positive constant δ, so $\xi' - \delta > \xi_0$, $\forall \xi' \in D_\xi$. We redefine the silhouette in the foreground area so $R(t;\mathbf{x})$ equals one in the interior of x_1, two in the interior of x_2, and so on, to make the likelihood (2) still valid. We have used this approach on the confocal images in Section 5.

3 Simulation from the Model

Markov chain Monte Carlo (MCMC) methods have been around for a while as the main tool for (iteratively) generating (dependent) samples from and exploring high dimensional (posterior) distributions. We will not focus on standard application of the MCMC methodology, like how to generate samples from our posterior, but instead concentrate the practically necessity of moving between different parameter subspaces by splitting and fusing objects in an efficient way within the MCMC framework.

Let $\pi(dx)$ be the target distribution of interest. In the MCMC methodology, we can randomly propose different ways to change the current state of the Markov chain x, and accept the move with a certain probability to maintain the correct equilibrium distribution. We construct a transition kernel $P(x, dx')$ that is irreducible and aperiodic, so that $\pi(dx)$ is the equilibrium distribution for the chain. It is common to use transition kernels that are reversible with respect to π, i.e., $P(x, dx')$ satisfies the detailed balance (for all appropriate A and B)

$$\int_A \int_B \pi(dx)\, P(x, dx') = \int_B \int_A \pi(dx')\, P(x', dx)\,. \tag{3}$$

The detailed balance restricts P more than actually needed, but makes the design of P more tractable. The algorithm goes as follows. Let x be the current state of the chain. We then propose a move of type j that moves x to dx' with probability $q_j(x, dx')$. This move is accepted with probability $\alpha_j(x, x')$ (Hastings 1970),

$$\alpha_j(x, x') = \min\left(1, \frac{\pi(dx')\, q_j(x', dx)}{\pi(dx)\, q_j(x, dx')}\right)\,, \tag{4}$$

otherwise we stay in the old state x. This chain can then be simulated to obtain a (dependent) sample from (approximately) the target distribution.

For our mpp–model we use the following types of moves: move and scale an object, change r_i and parts of $\mathbf{z}_i$, and finally, birth and death of objects. These types of moves are sufficient to be able to move between two arbitrary configurations in Ω. In practice, however, we need more possible moves when we have good quality images like the one shown in Figure 1. Assume one object is on its way to cover two cells lying close to each other. The strong likelihood will in practice prevent the necessary series of single steps needed to get out of this deadlock: remove the object, and then give birth to two new objects (one in each cell). The solution is to do the necessary series of single steps in one new step. An example of this scenario is provided in Figure 7b, which shows an example of an accepted split of an object covering two cells, using the guidelines below. (See also Grenander and Miller (1994) for examples of fusing and splitting of objects with a different model.)

When we move between subspaces giving simultaneous death to one object and birth to two new objects, we must be careful to maintain the correct equilibrium distribution. Green (1995) discusses reversible jumps between parameter subspaces in general, how to meet the reversible requirement and to adjust the acceptance probability in (4) by an appropriate Jacobian (in the last term) for dimension–matching. We will give a brief outline on how we propose to do efficient fusing and splitting of objects for our model. Figure 3 shows schematically how we propose to split an object into two objects. Let $(s, (r, \mathbf{z}))$ be the parameters of the original object, and $(s_1, (r_1, \mathbf{z}_1))$ and $(s_2, (r_2, \mathbf{z}_2))$ be the parameters for the two new objects. The required steps are as follows:

1. We propose an angle u_1 uniformly in $[0, 2\pi)$, and let L be the line with this angle through s. Then we select two new centers s_1 and s_2 uniformly on the line L in the interval $s \pm r$, with the constraint $|s_1 - s_2| = r$. Figure 3 shows the case when $u_1 = 0$.

2. The radii for the two new objects (r_1 and r_2) are drawn from a truncated Gaussian distribution, say, where the parameters depend on the distances $|s_1 - s|$ and $|s_2 - s|$ to

reflect the different sizes of object 1 and 2 (see Figure 3). It is important that the radii are not too large to prevent overlap of the two new objects.

3. We propose to handle the deviations $\mathbf{z}_1$ and $\mathbf{z}_2$ in the following way. First draw two lines orthogonal to L through s_1 and s_2 (see Figure 3). This divides the object into three regions; the left, interior and right part. As the outer boundary is often well placed, it is of vital importance to keep the left and right part approximately unchanged after the split. We let $\mathbf{z}_1$ take values to fit the polygon in the left part in Figure 3, and similarly for $\mathbf{z}_2$. To ensure reversibility, we must add some random noise to $\mathbf{z}_1$ and $\mathbf{z}_2$. The parts of $\mathbf{z}_1$ and $\mathbf{z}_2$ in the interior of the old object are updated from their Gaussian prior conditionally on the left and right part, respectively. (See also the comments in Section 5.3.)

4. For the confocal microscopy images the mean level of each object ξ is a stochastic variable, with $p(\xi) > 0$ in D_ξ. The new levels ξ_1 and ξ_2 then become $\xi \pm v$ where v is a random variable with density truncated so that $\xi \pm v \in D_\xi$.

This split–move has shown to be quite useful in the following case as well. Assume an object is born that covers only parts of a cell, and has a large part in the background area. This object is now likely to first be splitted into two objects, and then the object in the background area will be removed. Figure 7c shows an example.

We found the Jacobian needed in (4) for dimension–matching, to be r or $2r$ for fixed and stochastic level of each object, respectively. Refer to (Green 1995) for details.

To fuse two objects, we do essentially the same as described above with obvious adjustments like $r = |s_1 - s_2|$, and for the confocal microscopy images $\xi = (\xi_1 + \xi_2)/2$.

The effort to code the proposed splitting and fusing of objects is significant, and required (for us) much more work than all the other basic steps in the MCMC algorithm altogether. However, the coding effort is rewarded by the improved convergence.

4 Inference

This section discusses the inference for our model. Recall that our goal is to obtain a point estimate of $\mathbf{x}$ that includes the number of objects n as well. We will first review the definition and some properties of Baddeley's Δ–metric. Then we explain how to obtain the corresponding OBE approximately by using the ideas of the two–step MCMC and Simulated Annealing of Rue (1995b).

Recall that S is a finite portion of the plane, and that T is the finite lattice for which our data is observed. Assume for simplicity that S covers T. Let $\rho(u_1, u_2)$, $u_1, u_2 \in S$, be a metric on S, for instance the Euclidean metric. Define $d(u_1, \mathbf{x})$ to be the shortest distance from a point u_1 to the (contour defined by) the objects $\mathbf{x}$ measured by ρ. If u_1 is inside, or on the boundary of an object, the distance is zero. Baddeley (1992) suggests to use the Δ-metric to measure the distance between two configurations $\mathbf{x}$ and $\mathbf{x}'$,

$$\Delta_w^p(\mathbf{x}, \mathbf{x}') = \left(\frac{1}{|T|} \sum_{t \in T} |w(d(t, \mathbf{x})) - w(d(t, \mathbf{x}'))|^p \right)^{1/p} . \tag{5}$$

Here, $w(\cdot)$ is a concave and strictly increasing function, satisfying $w(v) = 0$ when $v = 0$. Furthermore, $w(\cdot)$ is eventually constant, i.e. there is a value v_c such that $w(v) = c$, for all $v \geq v_c$. Similar to Baddeley and van Lieshout (1991) and van Lieshout (1994) and for reasons discussed in Frigessi and Rue (1995), we choose $p = 2$ and

$$w(v) = \min\{c, v\}, \quad c > 0 . \tag{6}$$

We write $d_c(u_1, \mathbf{x})$ for $\min\{c, d(u_1, \mathbf{x})\}$ and $\Delta_c^2(\mathbf{x}, \hat{\mathbf{x}})$ when we use (6). In (6), c is a truncation parameter that determines the degree of localness in $\Delta_c^2(\mathbf{x}, \hat{\mathbf{x}})^2$. The contribution to one of the terms in the sum (5) is only influenced by the object configuration within a circle with radius c centered at t.

Refer to Baddeley (1992) for a discussion of general topological properties of the Δ–distance that are important in the context of image analysis described in Serra (1982).

The Δ–metric is particularly suitable for our case, as it is natural to consider our objects as sets for which the Δ–metric was derived. The argumentation in Section 1 is our motive for adopting (5) as loss function for configurations in Ω,

$$L_c(\mathbf{x}, \hat{\mathbf{x}}) = \begin{cases} \Delta_c^2(\mathbf{x}, \hat{\mathbf{x}})^2, & \text{if } \hat{\mathbf{x}} \in \Omega, \\ +\infty, & \text{otherwise.} \end{cases} \tag{7}$$

The loss function $L_c(\mathbf{x}, \hat{\mathbf{x}})$ describes the loss, or cost, if we estimate $\mathbf{x}$ as $\hat{\mathbf{x}}$. Note that the loss for estimating $\hat{n}$ objects when there is n objects, is *implicitely* present in (7). The optimal Bayesian estimator (OBE) minimizes the posterior expectation of the loss,

$$\mathbf{x}_\Delta = \arg\min_{\hat{\mathbf{x}} \in \Omega} \mathrm{E}_{\mathbf{x}|\mathbf{y}} \, L_c(\mathbf{x}, \hat{\mathbf{x}}) , \tag{8}$$

and we call it the Δ–estimator. In Section 5 we will study how the estimates using this loss function compare with those obtained with the commonly used 0–1 loss,

$$L_{\mathrm{MAP}}(\mathbf{x}, \hat{\mathbf{x}}) = \begin{cases} 0, & \text{if } \hat{\mathbf{x}} = \mathbf{x}, \\ 1, & \text{otherwise.} \end{cases} \tag{9}$$

The OBE for (9) is the posterior mode, $\mathbf{x}_{\mathrm{MAP}}$. The Δ–estimator has a nice property in common with the MAP–estimator, it is invariant for reparametrizations of the model parameters. This is obvious, since the loss only depends on the shape of the objects and then only implicitely on the underlying parameters. It seems for us very reasonable to define the loss on the shape of the objects, since we only see the objects and not the parameters.

To get a rough feeling of $\mathbf{x}_\Delta$, consider a one–dimensional object with the only edge at position θ with posterior distribution $\pi(\theta)$. The inside of the object is defined to be to the left of θ and the outside to the right of θ. This is a simple model for one (isolated) node–value in Figure 2, $r + z_i$ say, where the center is fixed. For a continuous lattice T, we find that for small c, the OBE for θ is

$$\theta_\Delta = \text{median}(\theta) - c + \mathcal{O}(c^2)\,. \tag{10}$$

In comparison, the MAP–estimator is the mode of $\pi(\theta)$. We can expect the mode of $\pi(\theta)$ to have more variability than the median. Thus we expect a smoother boundary for our objects using the new loss function, compared to the posterior mode, even if the real situation is considerably more complex than in this example. Another observation is, that (10) shifts the OBE in the direction to the *interior* of the object. This in agreement with the more complex situation in Figure 4 showing three objects A, A^+ and A^-. The Δ–distance between A and A^+ is (much) larger than the distance between A and A^-, as the extra part in A^+ influence the distance from far more sites t compared to the reversed situation in A^-. Assume that our prior knowledge prefer smooth objects similar to A. If the true object is A^+, then the estimated object will most likely contain the additional part in A^+ but slightly diminished. If the true object is A^-, then the estimated object can be more similar to the prior A. Refer also to Baddeley (1992) for more examples.

Let us now turn to the problem of how to compute, at least approximately, the Δ–estimate. We follow the ideas in Rue (1995b), but refer also to Rue (1995a) for a general discussion of how to obtain the OBE and to Frigessi and Rue (1995) for a more detailed discussion using (7) in image classification. First, suppose that

$$h(\hat{\mathbf{x}}) = \alpha \exp\left(-\mathrm{E}_{\mathbf{x}|\mathbf{y}}\, L_c(\mathbf{x}, \hat{\mathbf{x}})\right)\,, \quad \alpha > 0\,, \tag{11}$$

is known and practically computable. We will in a moment explain how to meet this requirement. For technical reasons we consider only finite objects, i.e. M is finite. In general, define M^+ as the subset of M with a positive measure, and denote by η the uniform probability measure on M^+. Define a Poisson object process with measure $\mu' = \lambda' \otimes \eta$, where λ' is the Lebesgue measure and $\lambda'(S) = 1$. Interpret $h(\hat{\mathbf{x}})$ as the density with respect to μ', and then α as a normalization constant. Then $\mathbf{x}_\Delta$ is the mode of the corresponding probability distribution $p_\Delta(\hat{\mathbf{x}})$. As (11) is practically computable, the mode, or at least good approximation of it, can be found using the standard technique of Simulated Annealing. Refer to van Lieshout (1994) for a proof of convergence and technical details for objects models. The annealing is to generate samples using MCMC methods from the tempered distribution $p_\Delta(\hat{\mathbf{x}})^{1/b}$, as we let the temperature b tend to zero sufficiently slowly. At the end, we will observe samples uniformly on the set of solutions of (8). In practice, we let b tend to zero faster than theoretically required, with the consequence that we are not ensured to reach the global mode of $p_\Delta(\hat{\mathbf{x}})$ or equivalently the minimum of (8). Our experience is that good estimates of $\mathbf{x}_\Delta$ can be obtained even with fast cooling or some steepest ascent algorithm described in (Baddeley and van Lieshout 1993), meaning that this step need not to be the most computationally demanding step in our algorithm.

The next step is to ensure that (11) is practically computable. We expand the square in $L_c(\mathbf{x},\hat{\mathbf{x}})$ to obtain

$$h(\hat{\mathbf{x}}) = \alpha \exp\left(-\frac{1}{|T|}\,\mathrm{E}_{\mathbf{x}|\mathbf{y}}\sum_{t\in T}\left(d_c(t,\hat{\mathbf{x}}) - d_c(t,\mathbf{x})\right)^2\right) \tag{12}$$

$$= \alpha' \exp\left(-\frac{1}{|T|}\sum_{t\in T} d_c(t,\hat{\mathbf{x}})\left(d_c(t,\hat{\mathbf{x}}) - 2\,\zeta_t\right)\right), \tag{13}$$

where

$$\zeta_t = \mathrm{E}_{\mathbf{x}|\mathbf{y}}\, d_c(t,\mathbf{x}), \tag{14}$$

and the constants α and α' are of no interest regarding the OBE. To estimate $\{\zeta_t, t\in T\}$ we make use of MCMC methods to estimate ζ_t, by averaging over states $\mathbf{x}^{(1)},\mathbf{x}^{(2)},\ldots,\mathbf{x}^{(K)}$ of the Markov chain, with the posterior as equilibrium distribution,

$$\overline{\zeta_t} = \frac{1}{K}\sum_k d_c(t,\mathbf{x}^{(k)})\,. \tag{15}$$

The law of large numbers still holds, even if the samples are weakly dependent, thus $\overline{\zeta_t}$ converges to the correct value. So (11), with a finite value of c, is practically computable. Note that the computational cost will increase for increasing values of c. In practice, values of c in the range $3-7$ seems reasonable.

Let us comment on the practical issue on how samples from $p_\Delta(\hat{\mathbf{x}})$ can be generated. We suggest to use MCMC with the same kind of proposals $q_j(x,dx')$ as for generating samples from the posterior distribution. This, because $h(\hat{\mathbf{x}})$ contains a lot of information about the posterior distribution through the ζ_t's. The practical implementation to compute $\mathbf{x}_\Delta$ is now easy, since we only need to replace $p(\mathbf{x}|\mathbf{y})$ by $p_\Delta(\mathbf{x})$, when calculating the acceptance rate $\alpha_j(x,x')$, after, for instance a proposed fusing of two objects. An efficient algorithm for computing $d_c(t,\mathbf{x})$ can make use of the fast algorithm (for discrete objects) in (Borgefors 1986) with some modifications for our continuous objects.

To summarize, the algorithm proceeds as follows. First, run a Markov chain with the posterior as the equilibrium distribution and estimate $\{\zeta_t, t\in T\}$. Secondly, run a Simulated Annealing algorithm on $p_\Delta(\hat{\mathbf{x}})$, to obtain a good approximation to $\mathbf{x}_\Delta$. Note the absence of the mpp–model itself, which makes the algorithm trivial to apply to other mpp–models.

5 Examples

This section contains some simulation experiments with the proposed mpp–model and a comparison between the Δ–estimator and the MAP–estimator. We start with an example with a single object and then move on to an example with an unknown number of objects. Finally, we show the result of applying our model to a real confocal microscopy image and show examples of the fusing and splitting of objects discussed in Section 4.

Our MCMC algorithm is implemented as described in Section 4. We have tuned the different parameters, for instance how often we shall propose to give birth to a new objects etc., to obtain a reasonable balance between the different kinds of moves as it appears on the screen. The $\{\zeta_t, t \in T\}$ were estimated using (15) and a large number of samples from the Markov chain. We got essentially the same results when we increased the number of samples. The Simulated Annealing algorithm used for computing the MAP and the Δ–estimator (with an Euclidean metric ρ) used an exponential cooling schedule. In all the examples, we used as S the continuous lattice T normalized to $\lambda(S) = 1$.

5.1 Simulation Experiment 1

The purpose of this first experiment is to show a typical example of how the Δ–estimator estimates the shape of an object in comparison to the MAP–estimator. We assume it is known in advance that the image contains only one object.

Our image has size 32×32, with a near circular object of pixel value 1 on a background with pixel value 0 (Figure 5a). We add iid Gaussian noise with mean zero and variance 1 (Figure 5b). We choose the parameters in the model as: $\mu_r = 5$, $\sigma_r^2 = 0.5^2$, $n_z = 30$, $\rho_z(\tau) = \exp(-\tau/\tau_c)$ where $\tau_c = 1/4$ and $\sigma_z^2 = 0.5/5^2$. Figure 5c and Figure 5d show the estimated MAP and Δ–estimates using $c = 5$.

The results from this example show a typical behavior. The contour of the object for the MAP–estimate is more influenced by the noise resulting in a more wiggly contour of the object. The Δ–estimate is more stable and gives a smoother estimate. Our experience with similar kind of examples, is that the Δ–estimator gives better, or equally good estimates of the contour of the objects, most prominent for moderate to bad data. The MAP–estimator gives (too) often an impression of capturing non–existing details in the object, that are not present in the Δ–estimate.

5.2 Simulation Experiment 2

In this experiment, the number of objects is unknown *a priori*, and we want to study how the number of objects are estimated using the MAP and the Δ–estimator.

Our image is of size 64×64 with three nearly circular objects of pixel value 1 on a background with pixel value 0 (Figure 6a). Figure 6b shows the result after adding iid Gaussian noise with zero mean and variance 1. The parameters in the model were: $\beta = 20$, $\mu_r = 4.0$, $\sigma_r^2 = 0.5^2$,

$n_z = 30$, $\rho_z(\tau) = \exp(-\tau/\tau_c)$ where $\tau_c = 1/4$ and $\sigma_z^2 = 0.4^2/4^2$. Figure 6c and Figure 6d show the estimated MAP and Δ–estimate using $c = 5$.

The MAP–estimate shows four objects, while the Δ–estimate shows the correct three objects, even if the prior model through the parameter β suggests more objects. For small objects measured as bad data, it is likely that there are parts in the background area that might seem like a small object. Our experience with similar kind of examples, is that the MAP–estimator tends to estimate more objects than there actually are, while the Δ–estimate is more stable with respect to the number of objects and seldom overshoots. If the MAP–estimate fails to locate an object, the Δ–estimate is likely to fail too. Even if it is easier to estimate too many objects when the size of the objects is small, the same will happen for larger objects with a proper scaling of the noise variance.

5.3 Application to a Confocal Microscopy Image

Finally, we apply our mpp–model to the real confocal microscopy image shown in Figure 1. It appears to be an easy task, since all the cells are clearly visible. However, our mpp–approach required the splitting and fusing steps described in Section 3 to obtain reliable results. We have displayed some of the moves that appeared in our trials to check for reliability of the approach. Figure 7a shows two misplaced objects that fuse together and after a short while and further developing, they split as shown in Figure 7b. The situation in Figure 7c is the one occurring most frequently. Here a misplaced object is divided in two, and soon the object in the background area meets its death. Two objects may be born into the same cell and after a short while they fuse into one object (Figure 7d).

An estimate of the location and shape of the cells in Figure 1 is shown in Figure 8. We estimated the parameters in prior distribution from a set of similar images by hand–tracking the cells. The correlation function ρ_z was found to be well approximated by

$$\rho_z(\tau) = \exp(-\tau/\tau_c)\,\cos(4\pi\tau)\,. \tag{16}$$

We choose $\tau_c = 1/3$ as a conservative estimate to avoid introducing too strong dependency structure in $\mathbf{z}$ for those of the cells with a slightly different correlation structure than (16). When we split an object as described in Section 3, we obtain better splits using the correlation function $\rho_z(\tau) = \exp(-\tau/\tau_c)$ instead of (16) to propose new values for $\mathbf{z}_1$ and $\mathbf{z}_2$ in the interior part of the object. This is because (16) fits well for one cell but is not always appropriate for the two new objects (see Figure 3) that may not (yet) have a fully developed shape as a cell.

The proposed model seems to do a reasonable job. But for so good data, samples of the Markov chain after an initial period and often some crucial moves like those displayed in Figure 7 are very good. Thus, the MAP–estimate, the Δ–estimate and simply one late sample of the Markov chain are all nearly equal to Figure 8.

6 Discussion

We end this paper by some comments on what the important parts in our approach are. For real applications with good images like Figure 1, it is simply a reasonable model for an object combined with a good simulation algorithm that does the correct types of moves within the MCMC framework. Figure 8 shows an example of good data where different estimators will not differ much from a late (perhaps cooled) sample from the posterior distribution. When the quality of our data decreases, the object model is again important, but also what kind of loss function we use. Based on our experience, the Δ–estimator is to be preferred to the MAP–estimator in such situations, even if the computational and coding cost slightly increase.

There is one part missing in our mpp–model, that is, an object interaction term in addition to the hard core part. We think that for situations where the objects are small (and more like points), where we have bad data, or we have strong prior knowledge of how the objects interact, an interaction term can give significantly improved results. Otherwise, we can concentrate on the object model and a MCMC algorithm that handle jumps between parameter subspaces in a good way, as the basis for doing inference and producing point estimates with the Δ–estimator.

Acknowledgements

The authors thank Henning Omre and Arnoldo Frigessi for helpful discussions. The second author was funded by a Ph.D grant from the Norwegian Research Council.

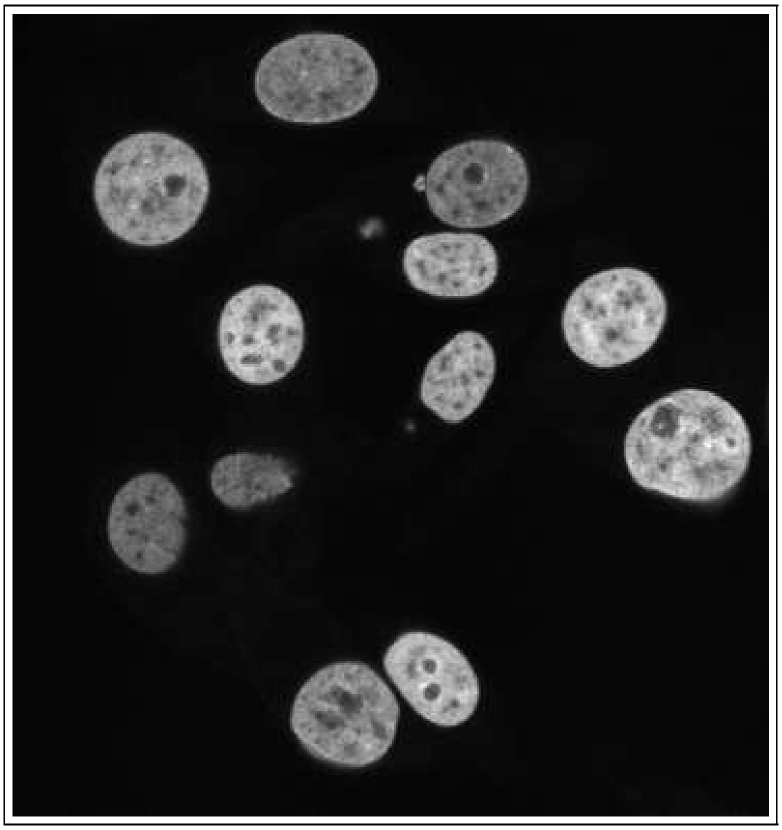


FIGURE 1: The image shows a typical confocal microscopy image of cells which is our motivation for the proposed mpp–model.

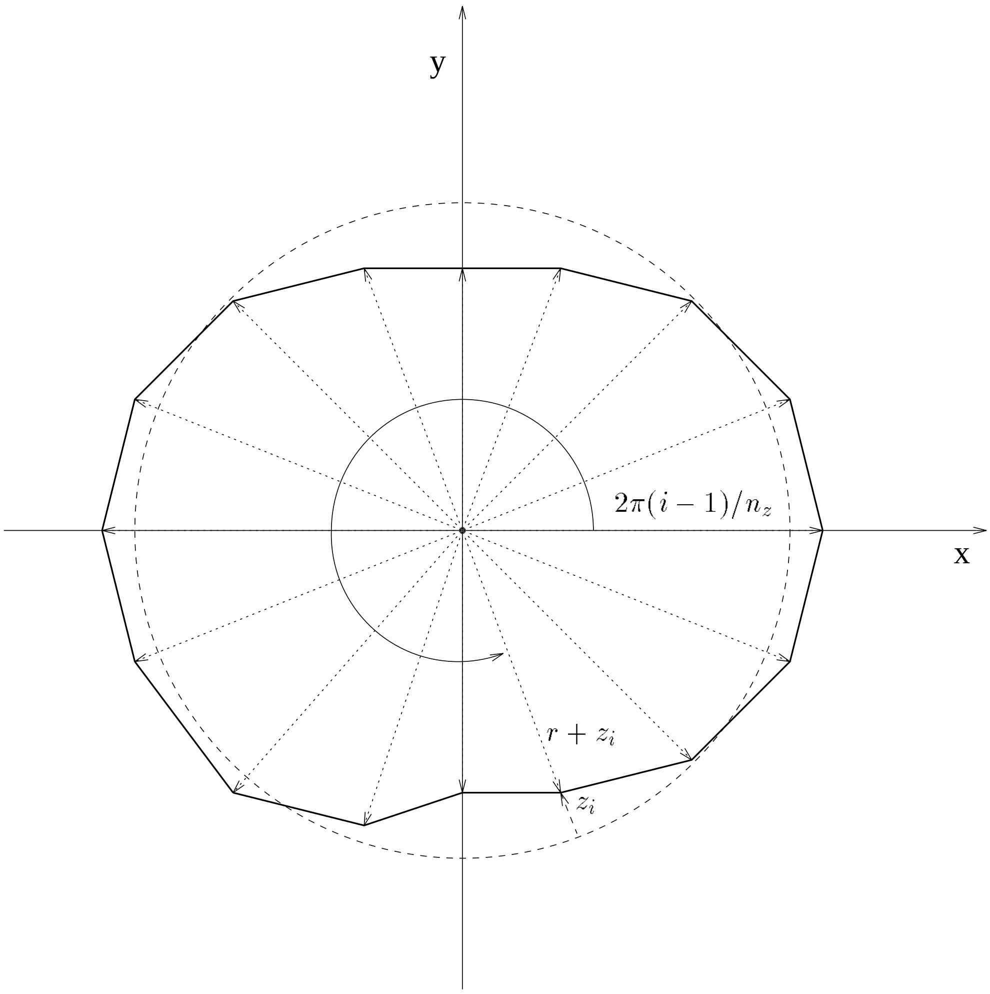


FIGURE 2: The figure displays schematicly our model for one object. The dotted circle with radius r is the overall shape. Local variability is modeled through the Gaussian vector $\mathbf{z} = (z_1, \ldots, z_{n_z})$, where z_i denotes the deviation from the circle along the line with angle $2\pi(i-1)/n_z$.

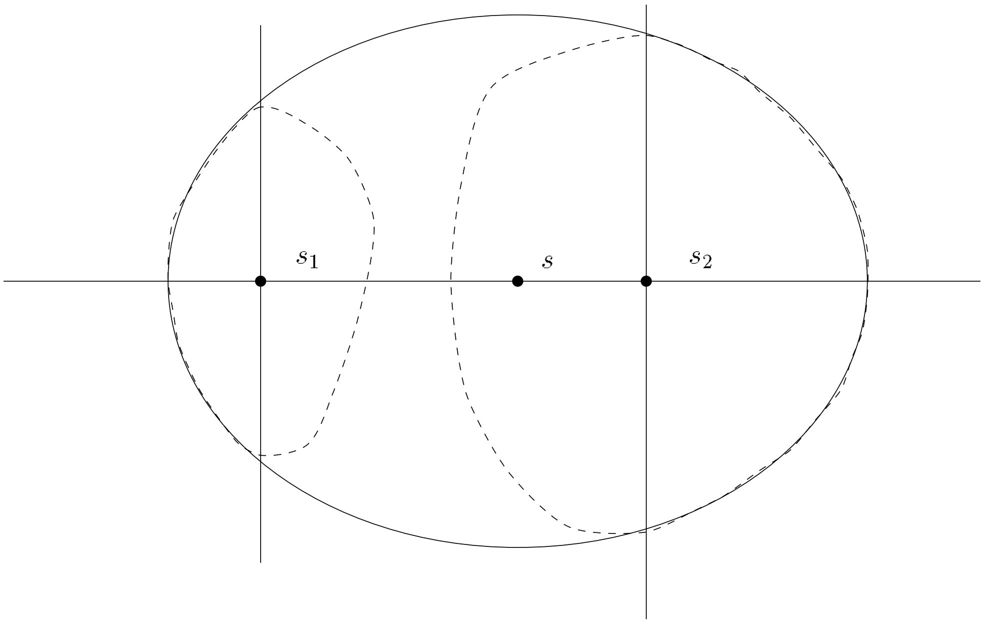


FIGURE 3: The figure shows how we split one object m (solid line) into two objects m_1 and m_2 (dotted lines). The two vertical lines divide the object into a left, interior and right part. The different steps are explained in Section 4.

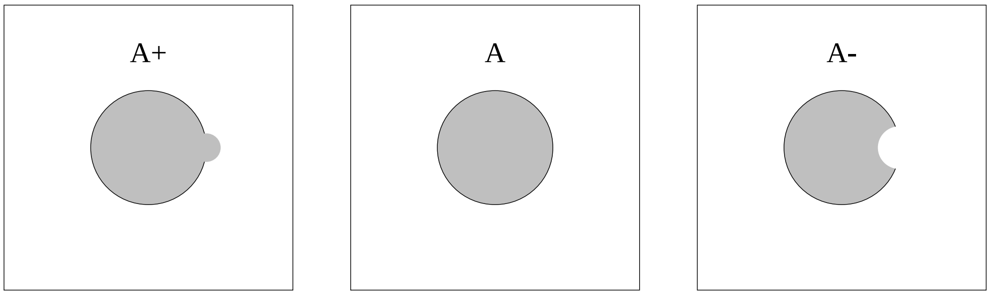


FIGURE 4: The figure shows three different objects, A^+, A and A^-. The distance between A and A^+ measure by the Δ–metric is (much) larger than the corresponding distance between A and A^-. This is because the extra part in A^+ influence $d_c(t, A)$ for far more sites t compared to the reversed situation in A^-.

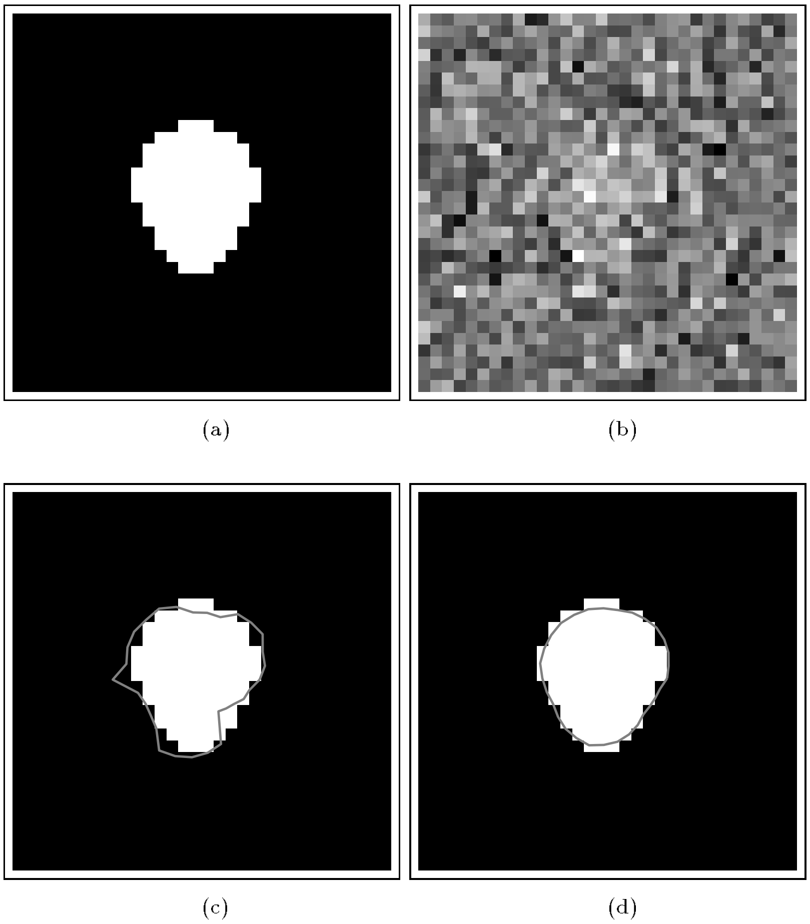


FIGURE 5: The images are the results from simulation experiment one. Image (a) shows the true discretized object with leaves zero and one. Image (b) shows the result adding Gaussian noise with zero mean and variance one. The MAP and Δ–estimate (with $c = 5$) are shown in image (c) and (d) respectively.

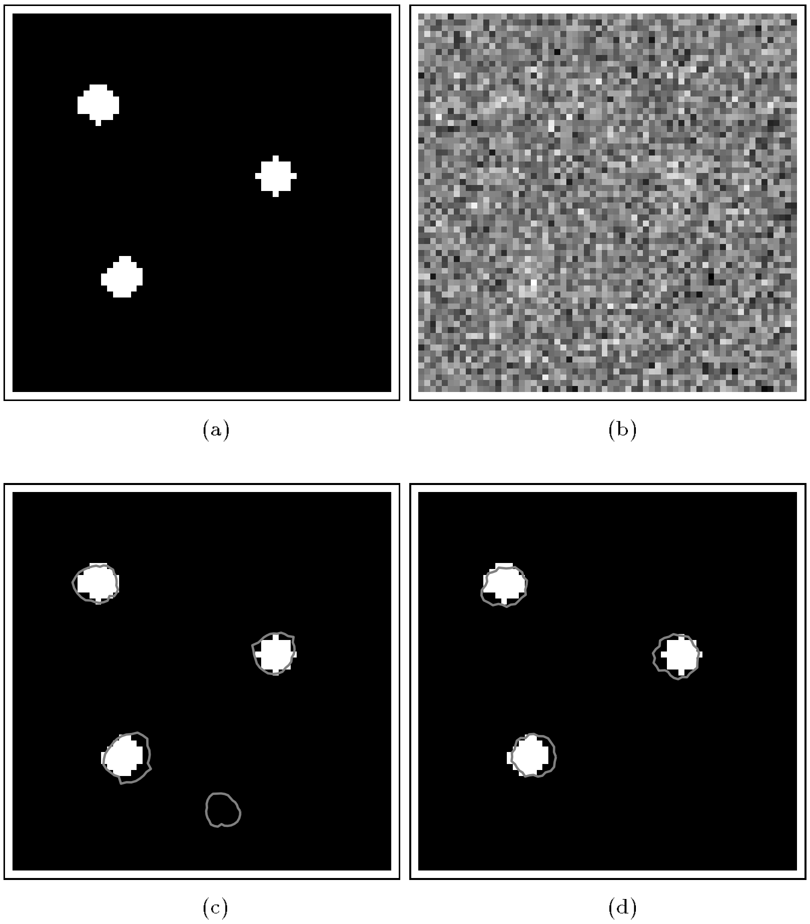
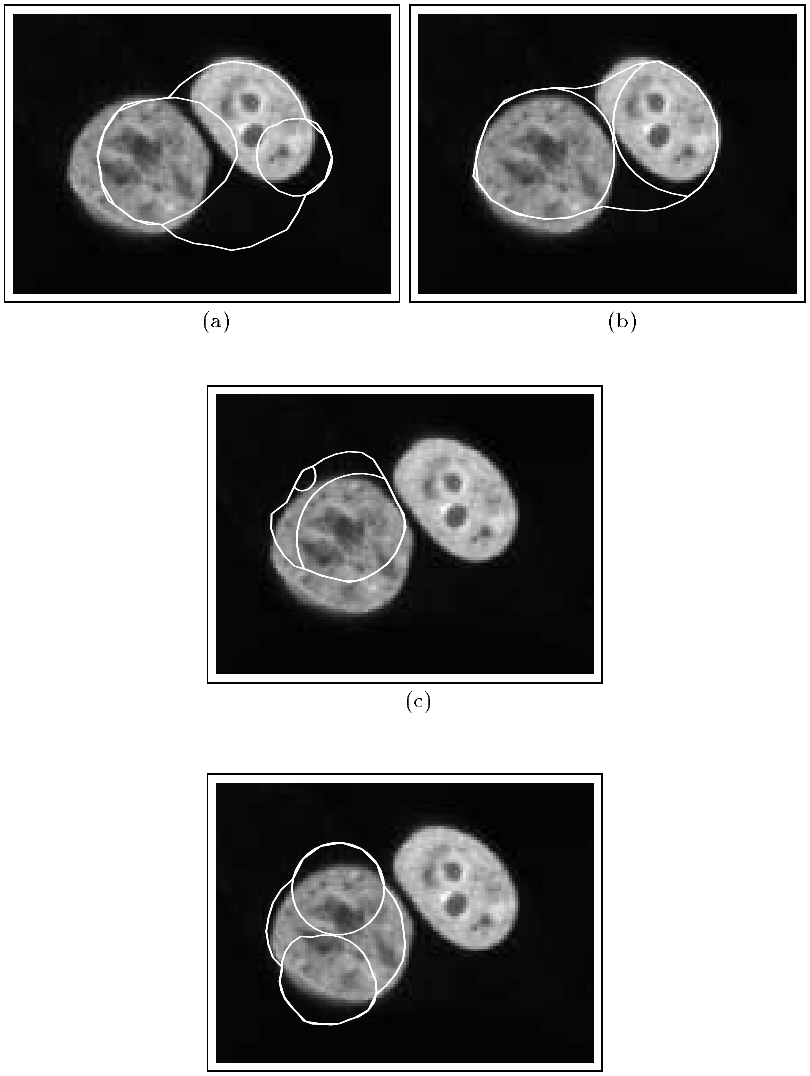


FIGURE 6: The figure shows the images from simulation experiment two. Image (a) shows the true discretized object with levels zero and one. Image (b) shows the result adding Gaussian noise with zero mean and variance one. The MAP and Δ–estimate (with $c = 5$) are shown in image (c) and (d) respectively.



(a) (b)

(c)

(d)

FIGURE 7: The images show a detail of Figure 1 and different moves happened when running our MCMC algorithm on the larger image. In image (a) two misplaced objects first fuse together and after a short while they split again as shown in image (b). Image (c) shows s splitting of one object into two objects, and the small object in the background area will soon meet the death. Image (d) shows fusing of two objects early in the MCMC run.

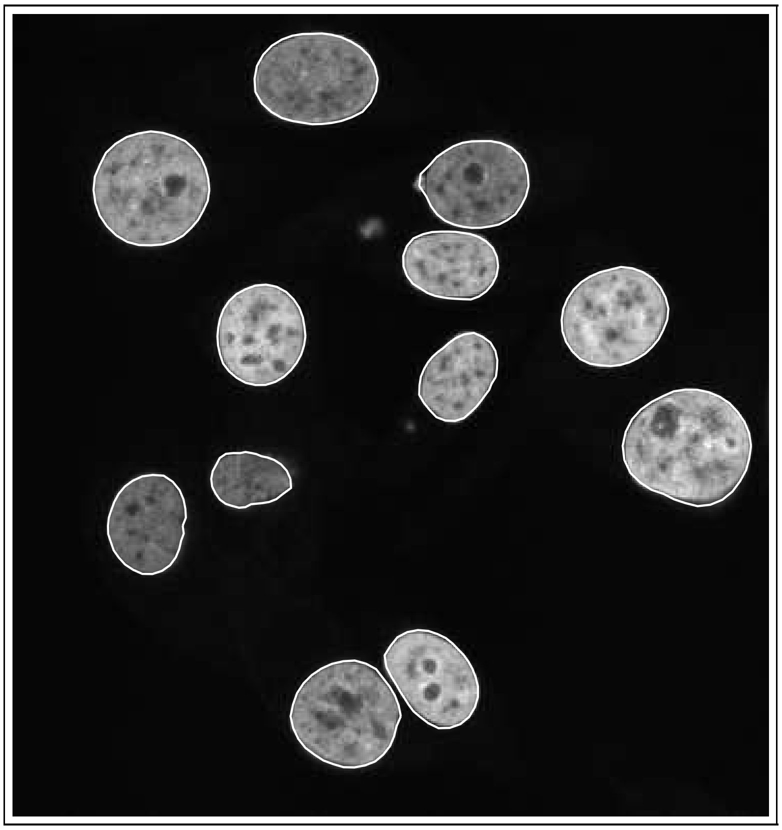


FIGURE 8: The image shows a typical result of applying our mpp–model to a real confocal microscopy image.

References

Amit, Y., U. Grenander, and M. Piccioni (1991). Structural image restoration through deformable templates. *J. Am. Stat. Assoc. 86*, 376–387.

Baddeley, A. (1992). Errors in binary images and a L^p version of the Hausdorff metric. *Nie. Arch. c. Wisk. 10*, 157–183.

Baddeley, A. J. (1994). Contribution to the discussion of paper by Grenander and Miller (1994). *J. R. Stat. Soc. Ser. B 56*(4), 584–585.

Baddeley, A. J. and J. Møller (1989). Nearest–neighbour markov point processes and random sets. *Int. Stat. Rev. 57*(2), 89–121.

Baddeley, A. J. and M. N. M. van Lieshout (1991). Recognition of overlapping objects using markov spatial processes. Technical Report Research Report BS–R9109, Centrum voor Wiskunde en Informatica.

Baddeley, A. J. and M. N. M. van Lieshout (1993). Stochastic geometry models in high–level vision. In K. V. Marida and G. K. Kanjii (Eds.), *Statistics and Image*, Volume 20, Chapter 11, pp. 235–256. Abingdon: Carfax Publishing Co. Special issue Journal of Applied Statistics.

Besag, J., P. Green, D. Higdon, and K. Mengersen (1995). Bayesian computation and stochastic systems (with discussion). *Stat. Sci. 10*(1), 3–66.

Borgefors, G. (1986). Distance transformations in digital images. *Comp. vis. graph. im. proc. 34*, 344–371.

Frigessi, A. and H. Rue (1995). Bayesian image classification with Baddeley's delta loss. Technical Report Statistics no. 5/1995, Departments of Mathematics, The University of Trondheim, Trondheim, Norway.

Geyer, C. J. and J. Møller (1994). Simulation procedures and likelihood inference for spatial point processes. *Scand. J. Stat. 21*, 359–373.

Green, P. J. (1995). Reversible jump MCMC computation and bayesian model determination. *Biometrika 82*(4), ???–???

Grenander, U. (1993). *General Pattern Theory.* Oxford University Press.

Grenander, U., Y. Chow, and D. M. Keenan (1991). *Hands: a Pattern Theoretic Study of Biological Shapes.* Research Notes on Neural Computing. Springer–Verlag.

Grenander, U. and K. M. Manbeck (1993). A stochastic model for defect detection in potatoes. *J. Comp. Graph. Stat.*. to appear.

Grenander, U. and M. I. Miller (1994). Representations of knowledge in complex systems (with discussion). *J. R. Stat. Soc. Ser. B 56*(4), 549–603.

Hastings, W. K. (1970). Monte carlo simulation methods using markov chains and their applications. *Biom. 57*, 97–109.

Phillips, D. B. and A. F. M. Smith (1994). Bayesian faces via hierarchical template modeling. *J. Am. Stat. Assoc. 89*(428), 1151–1163.

Ripley, B. D. and A. I. Sutherland (1990). Finding spiral structures in images of galaxies. *Phil. Trans. R. Soc. L. A 332*, 477–485.

Rue, H. (1995a). A loss function model for the restoration of grey level images. *In Revision.*

Rue, H. (1995b). New loss functions in Bayesian imaging. *J. Am. Stat. Assoc. 90*, 900–908.

Serra, J. (1982). *Image Analysis and Mathematical Morphology.* Academic press.

Tierney, L. (1994). Markov chains for exploring posterior distributions (with discussion). *Ann. Stat. 22*(4), 1701–1762.

van Lieshout, M. N. M. (1994). Stochastic annealing for nearest–neighbour point processes with application to object recognition. *Adv. App. Prob. 26*, 281–300.

van Lieshout, M. N. M. (1995). Markov point processes and their applications in high-level imaging (with discussion). *Bul. Int. Stat. Inst. LVI, Book 2*, 559–576.