BAYESIAN OBJECT IDENTIFICATION

HÅVARD RUE*
DEPARTMENT OF MATHEMATICAL SCIENCES
NTNU, NORWAY

MERRILEE A. HURN
SCHOOL OF MATHEMATICAL SCIENCES
UNIVERSITY OF BATH, U. K.

MAY, 1997

SUMMARY

This paper addresses the image analysis problem of object recognition - locating and identifying an unknown number of objects of different types in a scene. The particular application in mind is the automatic labelling of cells in a microscope slide. High-level statistical image analysis has been the subject of much recent research activity (Baddeley & Van Lieshout, 1993; Grenander & Miller, 1994). The former of these approaches advocates marked point processes as object priors; the latter approach is built around the use of deformable template models. In this paper elements of both approaches are combined to handle scenes containing variable numbers of objects of different types. The complexity of the posterior distribution of interest, together with the variable dimension of the parameter space, mean that reversible jump Markov chain Monte Carlo methods are required (Green, 1995). The naive application of these methods here leads to slow mixing; we propose three strategies to deal with this. This first two expand the model space by introducing an additional "unknown" object type and the idea of a variable resolution template. The third strategy is to include classes of updates which provide intuitive transitions between realisations containing different numbers of cells by splitting or merging nearby objects. A novel point estimator for the number of objects together with their locations, shapes and types is suggested and applied to an example of microscopy data.

SOME KEY WORDS: Bayesian inference; Deformable templates; Image analysis; Loss functions; Marked point processes; Markov chain Monte Carlo; Object recognition; Variable dimension distributions.

*ADDRESS FOR CORRESPONDENCE: Department of Mathematical Sciences, Norwegian University of Science and Technology, N-7034 Trondheim, Norway.
E-MAIL: Havard.Rue@math.ntnu.no and M.A.Hurn@maths.bath.ac.uk

1 INTRODUCTION

The processing of information carried in the form of images is often divided into "low-level" and "high-level" tasks. Statistical analyses in the former category generally model the image on a discrete lattice and operate at this pixel level, for example the use of Markov random field models in the removal of noise and blur. High-level imaging tasks such as object recognition require models and algorithms which deal with the components of the image on a global scale. High-level imaging is an important area in which there is much current research, images generally contain a huge amount of data and there is an important role for low dimensional models to ease interpretation and understanding. This paper deals with the object recognition task of locating and identifying cells of different types in microscope images.

The pioneering work of Ulf Grenander on pattern theory has been developing since the late 1960s and is now collected together in Grenander (1993). Pattern theory is a framework for representing knowledge in complex systems. The theory gives an algebraic framework (image algebras) for describing patterns as structures regulated by various rules. Natural variability of a pattern can be obtained by superimposing a probability measure on the image algebras. This variability is often described by probability measures on the parameters in low-dimensional Lie groups describing the deformation of an initial or typical configuration known as the template. The template is a closed polygon representing the outline of a typical object. The HANDS-study (Grenander, Chow & Keenan, 1991) describes early experiments, and the discussion-paper by Grenander & Miller (1994) summarises the current status of the research within this framework. Another approach to defining a high-level image model was proposed by Baddeley & Van Lieshout (1993), representing images as processes of geometrical objects. Marked point processes are advocated to handle the varying dimensionality of the number of objects, with the points representing the locations of objects and the marks being the other variables required to describe them. Other work in estimating the closed (discrete or continuous) boundaries of objects in an image, often with only vague knowledge of object shape, has been done by Blake & Yuille (1992), Storvik (1994), Helterbrand, Cressie & Davidson (1994), Pievatolo & Green (1995), Qian & Mardia (1995), Qian, Titterington & Chapman (1996) and Hurn (1996). Amit, Grenander & Piccioni (1991), Amit & Manbeck (1993), Mardia & Hainsworth (1993) and Jain, Zhong & Lakshmanan (1996) use a typical grey-level or black-and-white image as a template for the scene, and allow deformations of these shapes to match the observed data. Tjelmeland & Besag (1996) use a Markov random field model but incorporate higher order interactions tuned in such a way as to model compact objects. Clifford & Nicholls (1994) and Nicholls (1997) move away from models based on square pixellation of the lattice and use polygonal and triangulation based representations. See also Grenander & Miller (1994) and Jain et al. (1996) for more references.

In this paper we address the question of identifying an unknown number of object of potentially different types from an image such as that shown in Figure 7 (a). This confocal microscopy image of cartilage growth plate contains cells belonging to one of two classes. The cell shapes vary, both between and within cell types. Our goal is to estimate all of the following simultaneously: the number of cells; the cell shapes; the cell locations; and the type of each cell. The approach we shall take combines the methods advocated by Grenander & Miller (1994) and Baddeley & Van Lieshout (1993). To describe the shape of each type of cell we will use a deformable template model. We model the natural shape variations of a particular type of cell using stochastic deformations of its template. Section 2 begins by introducing this template model and the stochastic deformation mechanism. In our application, the number of cells present is unknown. Following the approach of Baddeley & Van Lieshout (1993), we handle the unknown dimensionality using a marked point process, modelling each cell as a point representing its location, and a mark variable representing its other characteristics. Different types of cell can be modelled using different templates. The mixture of different cell types is then handled by using a mixture distribution of different templates as this mark distribution. Reversible jump Markov chain Monte Carlo methods (Green, 1995) make it possible to analyse models such as this based on simulations. Section 2 concludes by describing the marked point process used as an image prior and the algorithms for simulating from the resulting posterior distribution which allow dimension changes via "births" and "deaths" of cells.

Although in theory the approach described in Section 2 is sufficient, in practice it may not be usable as the naive update proposals used in the MCMC algorithm may not result in adequate mixing, with too long waiting times between moves between cell types or configurations with different numbers of cells. In Section 3 we expand the parameter space by introducing resolution varying templates and an additional unlabelled cell type. The aim of this extension is to improve transitions between the labelling of the cells. However it also allows us to improve the transitions in the numbers of cells by including proposals which split or merge cells.

Inference is discussed in Section 4 where we extend the ideas in Rue & Syversveen (1995) to allow simultaneous estimation of the number of cells, their outline, type and location, using a loss function which penalises spatial errors. Examples and results so far are given in Section 5.

2 The High-level Image Model

This section defines a high-level image model to be used as a prior in our object recognition application. We begin by describing the basic template model for a single cell (Grenander, 1993), before embedding these cell representations into a marked point process framework (Baddeley & Van Lieshout, 1993). Simulation from this prior model, and from the corresponding posterior density, are then discussed.

2.1 A Deformable Template Model for a Single Cell

Suppose that the shape of a cell can be approximated by an n-sided simple polygon. The deformable template approach of Grenander and coauthors defines the template for an object to be the polygon representing its typical shape; the shape variability is then modelled by allowing deformations of this typical polygon, with the parameters of this deformation following a specified density. As a simple illustrative example, consider the square template (an $n = 4$ sided polygon) in Figure 1. One vertex is identified as the initial vertex (marked with a black dot in Figure 1). The vectors joining the vertices, known as the edges, are denoted $\{g_i\}$, $i = 0, 1, 2, 3$. In order to locate the template in the window $\Lambda \subset \Re^2$, the initial vertex is positioned at the point c; the positions of the four vertices are then c, $c+g_0$, $c+g_0+g_1$ and $c+g_0+g_1+g_2$. The closure of the polygon imposes the constraint that $\sum_{i=0}^{3} g_i = 0$. To model deformations of the square, each edge g_i is allowed to be deformed stochastically. We write the deformed column vector g_i as the new vector $s_i g_i$ defined by Fig.1 here

$$s_i g_i = g_i + r_i \begin{bmatrix} \cos(\theta_i) & \sin(\theta_i) \\ -\sin(\theta_i) & \cos(\theta_i) \end{bmatrix} g_i = \begin{bmatrix} 1 + t_i^{(0)} & t_i^{(1)} \\ -t_i^{(1)} & 1 + t_i^{(0)} \end{bmatrix} g_i. \tag{1}$$

The variables r_i and θ_i control the scale and angle of the change in the edge, $s_i g_i - g_i$. The alternative parameterisation in terms of $t_i^{(0)}$ and $t_i^{(1)}$ is used for the following reason: If r_i is proportional to a Rayleigh random variable and θ_i is uniformly distributed in $[0, 2\pi]$, then $t_i^{(0)}$ and $t_i^{(1)}$ are independent and identically distributed Gaussians with zero means. Following Grenander et al. (1991), we set the density of $\boldsymbol{t}^{(0)} = (t_i^{(0)})$ to be Gaussian with zero mean and inverse covariance matrix $\boldsymbol{\Sigma}^{-1} = (\Sigma^{i,j})$ in such a way that $\boldsymbol{t}^{(0)}$ is a first order cyclic Markov random field. In that case, the only nonzero terms in $\boldsymbol{\Sigma}^{-1}$ are then

$$\Sigma^{i,i} = a_0/n + 2a_1 n^2, \quad \text{and} \quad \Sigma^{i,i+1} = \Sigma^{i,i-1} = -a_1 n^2, \tag{2}$$

where all indices are modulo the number of vertices n. The parameterisation in (2) into a_0, a_1 and n is proposed by Kent, Mardia & Walder (1996). Eq. (2) has the nice interpretation that

$$\text{Var}(t_i^{(0)} \mid \text{rest of } \boldsymbol{t}^{(0)}) = 1/\Sigma^{i,i}, \quad \text{and} \quad \text{Corr}(t_i^{(0)}, t_{i+1}^{(0)} \mid \text{rest of } \boldsymbol{t}^{(0)}) = -\Sigma^{i,i+1}/\Sigma^{i,i}. \tag{3}$$

For the moment ignoring the closure constraint that $\sum_i s_i g_i = 0$, $\boldsymbol{t}^{(1)}$ is taken to be independent of $\boldsymbol{t}^{(0)}$ and identically distributed. The unconditioned joint density of $\boldsymbol{s}$ where $s_i = (t_i^{(0)}, t_i^{(1)})$ is then Gaussian and will be denoted $\pi(\boldsymbol{s})$. Imposing the constraint that $\sum_i s_i g_i = 0$ reduces the parameter dimension to $2n-2$ but with $\boldsymbol{t}^{(0)}$ and $\boldsymbol{t}^{(1)}$ now dependent. $\boldsymbol{s}$ still has a Gaussian density here, which we will denote $\pi^*(\boldsymbol{s})$; we will later use both $\pi(\boldsymbol{s})$ and $\pi^*(\boldsymbol{s})$ as appropriate. Figure 1 (b) shows an example of such a deformed template.

Notice that no mention has been made so far of the requirement that the polygon constructed in this way is simple (non-self-intersecting). Imposing this additional constraint, the density of $\boldsymbol{s}$, $\pi^{**}(\boldsymbol{s})$, is proportional to $\pi^*(\boldsymbol{s})$. The constant of proportionality, which will depend on n, is unknown and is difficult to evaluate analytically. However for reasons which we will elaborate in Section 2.3, this will not be a problem.

2.2 A Model for an Unknown Number of Objects

Extending the model presented in Section 2.1 to describe an object other than that represented by the simple square template is straightforward. Different templates can be defined for the different object types, possibly not using the same numbers of vertices or the same parameters controlling the typical deformations. Of more interest is whether these templates can be combined into a model which handles an unknown number of unidentified objects. Certainly one way to achieve this is to embed the deformable template models into a marked point process framework (Baddeley & Møller, 1989; Baddeley & Van Lieshout, 1993). Within this framework, each object x_i is represented by a point which gives its location and a set of marks which then specify its other attributes, such as its type and outline. The points are required to lie in the window Λ, and the marks in some space M, so then x_i lies in $U = \Lambda \times M$. A configuration of cells is described as a finite unordered set $\boldsymbol{x} = \{x_1, \ldots, x_k\}$ of objects which follow an object process, i.e. a marked point process on U with a Poisson object process on U as the basic reference process. Let $\mu = \lambda \otimes \nu$ be the finite non-atomic measure on U, where λ is Lebesgue measure on Λ and ν is the probability measure on M. Under the Poisson model, the mean number of objects is $\mu(U)$ and, conditional on the number of objects k, these objects are uniformly distributed in U. The joint density of $\{x_1, \ldots, x_k\}$ is defined by a density $f(\boldsymbol{x}, k)$ relative to this Poisson object process. For example, to model pairwise interactions between objects which are defined to be neighbours by some relation $\sim$, the function

$$f(\boldsymbol{x}, k) \propto \beta^k \prod_{i \sim j} h(x_i, x_j), \quad \beta > 0, \tag{4}$$

could be used; Van Lieshout (1995) discusses various interaction models. To model a situation where objects are not allowed to overlap (or perhaps to be closer than some distance apart), all objects are defined to be neighbours and the interaction function $h(x_i, x_j)$ is taken to be zero if objects x_i and x_j overlap and one otherwise. This model is known as the hard core object process, and is the one we will use in the examples in Section 5. We take the location of the first vertex c as the point for each object. In situations where more than one object type is present, the first mark each point carries is a categorical variable defining which type of object it is. Then, given the object type, the remaining marks are the deformations $\boldsymbol{s}$ of the standard template for this type. The mark density is a mixture distribution formed with the $\pi^{**}(\boldsymbol{s})$ for the different object types, as described in Section 2.1.

2.3 Simulation from the Posterior using this Model

Throughout this paper we will primarily be concerned with using the marked point process model described in Section 2.2 as a high-level image prior. Inference about the underlying scene $\boldsymbol{x}$ will be made from its posterior density given the data $\boldsymbol{y}$, $\pi(\boldsymbol{x}|\boldsymbol{y}) \propto \pi(\boldsymbol{x})\pi(\boldsymbol{y}|\boldsymbol{x})$. In this application, the data are recorded on a discrete lattice of pixels. Following Baddeley & Van Lieshout (1993), we will define the likelihood using the pixel silhouette of the configuration $\boldsymbol{x}$ (a pixel is said to be in the foreground if it lies within some polygon x_i, otherwise it is in the background). We will make the modelling assumption that background and foreground pixels have typical values μ_0 and μ_1 respectively, and that these values are recorded with additional pixel-wise independent Gaussian noise with variance σ^2.

For reasons of complexity, inference from the posterior (6) has to be based on sampling using Markov chain Monte Carlo methods. There is now an extensive literature on MCMC methods and Gilks, Richardson & Spiegelhalter (1996) provide a comprehensive overview of theory and applications. In brief, in order to generate realisations from some density $f(d\boldsymbol{x})$ we construct a Markov chain of $\boldsymbol{x}$ values using an irreducible aperiodic transition kernel which has $f(d\boldsymbol{x})$ as its equilibrium distribution. When the variable $\boldsymbol{x}$ is of fixed dimension, the algorithm is as follows: Suppose the current state of the Markov chain is $\boldsymbol{x}$, and we propose a move of type j that moves $\boldsymbol{x}$ to $d\boldsymbol{x}'$ with probability $q_j(\boldsymbol{x}, d\boldsymbol{x}')$. The move to $\boldsymbol{x}'$ is accepted with probability $\alpha_j(\boldsymbol{x} \to \boldsymbol{x}')$, where

$$\alpha_j(\boldsymbol{x} \to \boldsymbol{x}') = \min\left(1, \frac{f(d\boldsymbol{x}')\, q_j(\boldsymbol{x}', d\boldsymbol{x})}{f(d\boldsymbol{x})\, q_j(\boldsymbol{x}, d\boldsymbol{x}')}\right), \tag{5}$$

otherwise we stay in the original state $\boldsymbol{x}$. The situation is more complicated when $\boldsymbol{x}$ is not of fixed dimension, as is the case here for scenes containing different numbers of cells. Green (1995) describes the reversible jump MCMC method for jumping between $\boldsymbol{x}$ of different dimensions. In these cases, it is necessary to "dimension match" $\boldsymbol{x}$ and $\boldsymbol{x}'$, including an additional Jacobian term in the acceptance ratio. Both types of move are required in sampling our posterior density; moves which keep the dimension of $\boldsymbol{x}$ fixed, for example altering the location of shape of one of the cells, and moves which alter the dimension by altering the number of cells. We will concentrate on the first category initially, giving a simple illustrative example.

Consider again the square template shown in Figure 2 (c). We will use this template in locating the object observed in the data shown in Figure 2 (b). This data have been generated from the underlying pixellated image $\boldsymbol{x}$ in Figure 2 (a) by setting the black background pixels to have value $\mu_0 = 0$ and the white foreground pixels to have value $\mu_1 = 1$. Pixel-wise independent zero mean Gaussian noise with variance $\sigma^2 = 0.5^2$ had then been added. Using the prior density for location c and deformations $\boldsymbol{s}$ described in Section 2.1 conditioned on the closure of the polygon, and this likelihood $L(\boldsymbol{y}\,|\boldsymbol{s}, c)$, the posterior density for c and $\boldsymbol{s}$ becomes Fig.2 here

$$\pi^*(\boldsymbol{s}, c \mid \boldsymbol{y}) \propto \pi^*(\boldsymbol{s}) \prod_{i \in S} \exp\left(-\frac{1}{2\sigma^2}(y_i - \mu(i;\, \boldsymbol{s}, c))^2\right), \tag{6}$$

where S indexes the pixel lattice, and $\mu(i;\, \boldsymbol{s}, c)$ is equal to μ_1 if pixel i is inside the deformed template, and equal to μ_0 if it is outside.

There are various ways in which we can propose a change to the current realisation of $\boldsymbol{x} = \{c, \boldsymbol{s}\}$. Proposing a change in c alters the location of the current polygon without altering its shape; in the early stages if the object is not well located, moves of this type are quite useful. By proposing symmetric changes in c, the acceptance ratio (5) will depend only on the likelihood ratio of the new and old states. How can the shape of the polygon be changed? Proposed changes can be made either by altering $\boldsymbol{s}$ itself, or by altering the position of one or more of the vertices directly. Notice that the vertex locations can be written as a one-to-one linear transformation of c and $\boldsymbol{s}$ (subject to keeping the labelling of the vertices constant). This means we can propose to move a randomly chosen vertex, perhaps uniformly in some small disc around its existing value (a symmetric proposal). Recall that $\pi^*(\cdots)$ is the density of the closed but not necessarily simple polygons. The non-self-intersection constraint is incorporated in the MCMC method without knowing the necessary normalising constant by simply rejecting all proposed non-simple polygons and accepting the others using $\pi^*(\cdots)$ in (5). This follows from the fact that the prior density for the simple polygons is proportional to $\pi^*(\cdots)$ over such polygons, noting that only the ratio of this density at two configurations is required in the acceptance probability. Because the transformation from c and $\boldsymbol{s}$ to the vertices is linear, the Jacobian is a constant, which will therefore cancel in the acceptance ratio which becomes

$$\alpha_j((\boldsymbol{s}, c) \to (\boldsymbol{s}', c')) = \min\left(1, \frac{\pi^*(\boldsymbol{s}', c' \mid \boldsymbol{y})}{\pi^*(\boldsymbol{s}, c \mid \boldsymbol{y})}\right) = \min\left(1, \frac{\pi(\boldsymbol{s}', c' \mid \boldsymbol{y})}{\pi(\boldsymbol{s}, c \mid \boldsymbol{y})}\right) \tag{7}$$

where $\pi(\boldsymbol{s}, c|\boldsymbol{y})$ denotes the posterior density using the unconditioned $\pi(\boldsymbol{s})$ rather than $\pi^*(\boldsymbol{s})$ in (6). The second equality above uses the general fact that $f(\boldsymbol{u}')/f(\boldsymbol{u}) = f(\boldsymbol{u}'|\boldsymbol{A}\boldsymbol{u}' = \boldsymbol{b})/f(u|\boldsymbol{A}\boldsymbol{u} = \boldsymbol{b})$, for some matrix $\boldsymbol{A}$ and vector $\boldsymbol{b}$. This allows a fast evaluation of the density $\pi(\boldsymbol{s})$ by using its first order Markov property. Figure 2 (d) shows some samples of the posterior density using the template in Figure 2 (c) and parameters $a_0 = 50$ and $a_1 = 0.5$ in (2).

Extending this example to handle an unknown number of objects requires modifications of the MCMC algorithm to accommodate the dimensionality changes. Essentially, as well as proposing fixed dimension changes to the current cells, the method has to allow for births and deaths of cells. A framework has been provided for this by Geyer & Møller (1994), Green (1995) and Geyer (1996). We will not concentrate too heavily on these move types for reasons which will be elaborated in Section 3. However, one way to decrease the number of cells by one is to delete a randomly selected cell. The complimentary move, increasing the number of cells by one, is to propose a new cell by simulating a location uniformly in the window, and a set of marks at random from the mixture model of template types. The corresponding acceptance probabilities are given in the Appendix A.1.

3 Variable Resolution Templates and Improved Mixing

In theory the move types described in Section 2 are sufficient for sampling the posterior density of interest. However consider the following two scenarios: In the first case, an object has been located but there is some uncertainty over its type; using birth and death moves to change the object label, the sampling algorithm must first remove the object altogether and then a new object has to be born in roughly the same location but with a different label. In the second scenario, it is not clear whether one detected "object" is actually two objects close together. Again using births and deaths, the algorithm may have to move through a low probability configuration containing no object in order eventually to reach the other high probability state. In this section, we introduce various enhancements both of the model and of the range of proposal move types in order to construct good and intuitive transitions around the posterior density. We begin by introducing the idea of variable resolution templates. This extension of the model brings two benefits. The first is a way to change label types readily. The second benefit is in formulating a class of split and merge transitions which allow nice intuitive moves between configurations with different numbers of objects.

3.1 Variable Resolution Templates

The challenge in designing a move type which alters the label of an object, thereby describing it by the deformed version of a different template, is to do it in such a way that the acceptance rate is not too low. Recall that templates need not have the same number of vertices, perhaps one object type is modelled as a deformed square, another object type as a deformed triangle. Following previous updating steps, the existing vertices should be well positioned to describe the object, subject to the constraints of the polygonal template. How should a change of type be proposed which will preserve this advantage? If the templates had the same number of sides, then one option would be to propose an object of the new type which shared the existing vertex positions, but whose interpretation was now as a deformed version of the new template. With this in mind, we introduce the idea of a variable resolution template.

The basic idea is simple, and we begin by describing the variable resolution template for a single template, before describing moves between different templates. Consider again the example in Section 2, where the template is a square with $n = 4$ edges. Although four edges are sufficient to describe a square, Figure 3 (b) and (c) show two alternative templates using five and six edges (the edges are denoted as vectors). We will include this feature in the deformable template model by allowing the resolution n to vary, Fig.3 here

$$\pi(\boldsymbol{s}, n) = \pi(\boldsymbol{s} \mid n)\ \pi(n), \qquad n \in \mathcal{N}, \tag{8}$$

where $\mathcal{N}$ is the set of possible resolutions. We add a superscript "(n)" to variables to indicate the resolution of the corresponding template. A preference for $n = 4$ can be included in the prior density $\pi(n)$ which should be concentrated on parsimonious shape descriptions. The particular choice of where the additional vertices should be placed in the template has been made to minimise the spread of the length of the edges $g_i^{(n)}$'s. Once the range of allowable resolutions for a particular object type has been decided by specifying $\pi(n)$, this feature can be included in the mixture model of objects.

3.2 Altering the Resolution of a Template

We need to extend our MCMC algorithm in Section 2.2 to include a move type which alters the resolution of the template. This type of move alters the dimension of the parameter space and so the reversible jump MCMC of Green (1995) is required. In increasing the resolution of the current template from n to $n + 1$, $n - 1$ of the vertices will be held fixed, while the remaining one is replaced by two new vertices. Figure 4 displays this operation when we propose to increase the resolution of a template from $n = 5$ to $n = 6$. First we set the location $c^{(n+1)} = c^{(n)}$, and then we sample one of the n available vertices at random, we denote this vertex v_u (displayed as a circle in Figure 4). It is not this vertex which will be altered, but rather the deformations starting from this vertex. We sample these $n_u = 3$ new deformations starting from v_u conditional on the remaining $n - n_u$ deformations using the Gaussian prior density described in Section 2.1 (these new edges are shown as dashed lines in Figure 4). (More generally, we could alter more edges by setting n_u to be higher than 3.) Fig.4 here

Some edges of the polygon are held fixed during the move, but the corresponding deformations s_i are not necessarily fixed because when the underlying template changes, the interpretation of these edges is now as deformations of the new template. Let $F^{(n)}$ and $F^{(n+1)}$ be the ordered set of indices where the corresponding edge is fixed during the proposed move, and let $U^{(n)}$ and $U^{(n+1)}$ be the ordered indices of the remaining updated edges. (In our example, $F^{(5)} = (0, 3, 4)$ and $F^{(6)} = (0, 4, 5)$, while $U^{(5)} = (1, 2)$ and $U^{(6)} = (1, 2, 3)$.) Denote the j^{th} element in $F^{(n)}$ by $F_j^{(n)}$, and so on. Then the relationship to be satisfied is

$$s_{F_j^{(n)}}^{(n)} g_{F_j^{(n)}}^{(n)} = s_{F_j^{(n+1)}}^{(n+1)} g_{F_j^{(n+1)}}^{(n+1)}, \quad j = 1, \dots, n - n_u. \tag{9}$$

The reverse move, decreasing the resolution of the template by one vertex, is proposed in a similar way: select a vertex v_u at random, then simulate the following two deformations conditional on the vertices other than the two immediately following v_u. The acceptance rate for the proposed increase in resolution can then be calculated, and this ratio is given in Appendix A.2

3.3 ALTERING THE LABEL OF AN OBJECT

Our next step is to implement moves which alter the object label; such moves will only be proposed between templates of the same resolution. As a simple example, Figure 3 (d-g) display a triangular template with between three and six vertices; we want to combine these templates with the square template in Figure 3 here using between four and six vertices. The joint mixture distribution for the deformations $\boldsymbol{s}$, the number of vertices n and the model type m is

$$\pi(\boldsymbol{s}, n, m) = \pi(\boldsymbol{s}^{(n,m)} \mid n, m)\, \pi(n \mid m)\, \pi(m), \qquad n \in \mathcal{N}^{(m)}, \quad m \in \mathcal{M}, \tag{10}$$

where $m \in \mathcal{M}$ indicates the model-type, for example square ($m = 1$) or triangle ($m = 2$). The set of allowable resolutions $\mathcal{N}^{(m)}$ can depend on m. The move then becomes a change in the *interpretation* of the polygon, for example from a deformed square to a deformed triangle. Before we give details of this move, we note the following: The vertex positions are held fixed during the move. If the likelihood model depends only on the outline of the polygon (as described in Section 2.2), then the value of the likelihood is unchanged by the move. This feature is very important for preserving the work already done in the template matching to the data, and so in generating good proposals.

For the models in Figure 3, we permit jumps between (a) and (e), (b) and (f), (c) and (g), simply by changing the interpretation of the polygon. A typical scenario is the following: Suppose the true object a square and the deformed template initially is of the triangle type with $n = 3$. The resolution will tend to increase so that the polygon fits the data better (weighed against the cost of decreased density of the required deformations). The polygon will soon start to be more like a square than a triangle, and then the object-type may well change to a square. The resolution will now typically reduce to the minimum number of edges required to describe a deformed square. The acceptance rate for proposing a move to object-type m' from m (keeping the position of the vertices fixed) is given in Appendix A.3.

3.4 Split and Merge Transitions

In this section, we describe another very useful class of proposed moves, those which split or merge objects. Such moves might be used when there is uncertainty whether an existing object actually covers two objects in the scene, or vice versa. If the data are reasonably good, the existing object will deform to match the data, and using birth and death moves alone the intermediate configuration between the one and the two object states will be of prohibitively low probability. See also the discussion in Grenander & Miller (1994) and Rue & Syversveen (1995). Resolution varying templates provide a natural framework for constructing good split and merge transitions, together with simple expressions for the acceptance probabilities (compare this to the very complicated procedure in Rue & Syversveen (1995)). We will begin by discussing split and merge transitions when there is only one template type present. We will then go on to describe the generalisation which allows for various templates; this generalisation also introduces a new object type, labelled "unknown".

A simple and desirable way to merge two objects, is to let them "melt" together where they are close, preserving their other vertices; Figure 5 (a) illustrates this idea. Label the two objects in Figure 5 as A and B, and the new merged object as C. We will restrict the merging proposals to cases where the two objects are close (the Euclidean distance between the merging vertices must be less than a specified value ϵ). The merging operation must be done in such a way that a reverse splitting move can be constructed. We propose to merge A and B into C by replacing the two pairs of merging vertices each by a new vertex in the centre of these two ϵ-close pairs (see Figure 5). Object C has resolution $n_C = n_A + n_B - 2$. The use of resolution varying templates makes it easy to allow parts of A and B to remain unchanged during the merge (necessary to obtain efficient transitions, the fit in the fixed parts will most often be good already). In the reverse splitting step, we need to replace two vertices belonging to C with two pairs of new ϵ-close vertices belonging to A and B. To do this, for each pair we draw one vertex uniformly in a disc with diameter ϵ, and let this determine the other vertex by fixing their centre to be the vertex of C. Note that it is possible to extend these split and merge steps to allow a stochastic number of vertices between the merging p_1 and p_2 in Figure 5 (b). In splitting, the deformations between p_2 and p_1 for object A can be updated (from the prior density) conditional on p_1, p_2 and the rest of object A, and similarly with object B. The acceptance rate is given in Appendix A.5.

Fig.5 here

In either the splitting or the merging stage, notice that any one of the vertices of the new object created can be labelled as the initial vertex. This will not be an issue if the template in question is rotationally invariant. However if this is not the case, then many potentially good proposed moves may be rejected simply because of the specified order of vertex labelling. An additional complication if more than one object type is present, is that we need to sample the object types for the new configuration. For this reason, we introduce an additional object type, labelled as "unknown" which has a rotationally invariant template at all resolution levels. This object type has a large variance in the density controlling its deformations (to allow large shape variability) but a low occurrence rate in the mixture distribution of object types. Splitting and merging will only be proposed for objects of this "unknown" class. The idea behind this being that objects on the verge of splitting or merging will begin to be unlikely deformations of the standard templates, and so may be more readily labelled as "unknown". The arbitrary labelling of the resulting objects will not be a problem for the rotationally invariant “unknown” template, nor will the fact that initially the shape may be complicated. As the new object’s shape evolves, it should become recognizable as a properly labelled object type. A change of label and ordering is a cheap proposal in comparison to the split or merge step since all that is required is a reinterpretation of the template as described in the previous section. The acceptance probability for relabelling the vertices is given in Appendix A.4. Notice that the “unknown” type may of course also be useful in identifying objects which have not been specified in the list of template types.

4 INFERENCE

Inference for the posterior density has to be based on samples generated by our MCMC algorithm. Global properties such as the mean number of objects, or the number of objects of a specific type, together with mean areas of typical objects of a specific type can be estimated by averaging the statistics of interest over samples from the posterior density. However, it becomes more complex to estimate simultaneously the number of objects, their locations, outlines and types. The estimator should be based on a suitable loss function such as the distance measure we would have used to compare an estimate to the true (unknown) configuration (Rue & Syversveen, 1995; Frigessi & Rue, 1997). In practise, it can be infeasible to compute the optimal Bayes estimator corresponding to a desired loss function. This is one of the reasons for the use of the common zero-one loss function which carries a loss of 1 for all configurations other than the true one; it results in the maximum a posteriori (MAP) estimate and this mode of the posterior density may be approximated using simulated annealing for example. Despite this, Rue & Syversveen (1995) demonstrate that it is still possible to utilise good distance measures between configurations of objects as workable loss functions. They propose using Baddeley’s delta metric as a loss function for Bayesian object recognition, and they demonstrate how the two-step MCMC and simulated annealing algorithm of (Rue, 1995; Rue, 1997) can be applied within the marked point process framework to estimate the corresponding optimal Bayes estimator. Baddeley’s delta metric is a distance measure between two configurations of objects (Baddeley, 1992; Frigessi & Rue, 1997). For our application where the two configurations $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ are defined in a window Λ

$$\Delta_c^2(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \left(\frac{1}{|T|} \sum_{t \in T} |\min(c, d(t, \boldsymbol{x})) - \min(c, d(t, \hat{\boldsymbol{x}}))|^2 \right)^{1/2}, \qquad c > 0, \tag{11}$$

where T is a lattice covering Λ finely enough to be a good approximation to a continuous version of (11), $d(t, \boldsymbol{x})$ is the Euclidean distance from a point t to the nearest object in configuration $\boldsymbol{x}$ (with the convention that $d(t, \boldsymbol{x}) = 0$ if t is within an object), and the constant c determines the range within which each point t looks for objects.

The delta metric implicitly gives the loss for the number of objects and their outline, but not for properties of the objects such as type. To estimate the type of each of the objects in a scene, it is not possible simply to add terms into the loss function which specify the loss for object type a if we estimate it by type $\hat{a}$. This is a result of the variable dimension aspect of the model, the number of objects is unknown and the model is defined for an unordered set of objects $\{x_1, x_2, \ldots\}$. To avoid this problem, we propose to associate the type–property with the discretised lattice Λ, specifying the loss for mislabeling pixels i in Λ rather than the objects themselves. Properties such as type can then be estimated by adding an extra term to (11) which gives the loss for mislabeling a point in T,

$$L_{\text{type}}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \frac{1}{|T|} \sum_{t \in T} 1_{\text{type}}(t, \boldsymbol{x}, \hat{\boldsymbol{x}}), \tag{12}$$

where $1_{\text{type}}(t, \boldsymbol{x}, \hat{\boldsymbol{x}})$ is the indicator function taking the value 1 if the type at t is wrong and zero otherwise (with the natural convention that all points within an object have the same type as the object). We could have similar terms to (12) for each property of interest for the object, for example the grey-level of an object may be included by using a sum of squared differences of the mean levels in the objects. We then take our total loss function as a weighted sum of (11) squared and divided by c^2, plus γ times (12). The posterior risk becomes (up to an additive constant)

$$\begin{aligned} \mathrm{E}_{\boldsymbol{x}|\boldsymbol{y}} L(\boldsymbol{x}, \hat{\boldsymbol{x}}) =& \frac{1}{|T|c^2} \sum_{t \in T} \min(c, d(t, \hat{\boldsymbol{x}})) \left(\min(c, d(t, \hat{\boldsymbol{x}})) - 2\mathrm{E}_{\boldsymbol{x}|\boldsymbol{y}} \min(c, d(t, \boldsymbol{x}))\right) \\ &+ \frac{\gamma}{|T|} \sum_{t \in T} \mathrm{pr}_{\boldsymbol{x}|\boldsymbol{y}}(1_{\text{type}}(t, \boldsymbol{x}, \hat{\boldsymbol{x}}) \neq 0), \qquad \gamma > 0. \end{aligned} \tag{13}$$

These posterior expectations are estimated from samples from the posterior as the first step in the algorithm. The optimal Bayes estimator is then the $\hat{\boldsymbol{x}}$ which minimises (13), and it can be estimated by a simulated annealing algorithm as described in Rue & Syversveen (1995).

5 Examples and further work

5.1 Artificial test example

We will begin this section with a small example of object recognition in a situation where there is little uncertainty. The primary aim of this example is to illustrate the use of simple templates and to demonstrate the strength of the sampling behaviour with our proposed approach.

Figure 6 (a) shows a set of artificial data generated from a pixellated image where background and foreground take mean levels $\mu_0 = 0$ and $\mu_1 = 1$ respectively; the variance of the pixel-wise independent Gaussian noise is $1/4^2$. To identify the objects we use three templates, a circle approximation, a rectangle and an "unknown" object-type. Figure 6 (a) shows the sampler's starting configuration containing two rectangles both covering parts of both objects. Here the relatively low noise level is not an advantage, and without our proposed methods for improving the mixing, progress would be slow in moving away from this poor starting point. Monitoring the sampler, the resolution of each template increases as it tries to match the data, and at some stage both appear sufficiently deformed that they are labelled "unknown". As such, they are now candidates for split or merge transitions. Figure 6 (b) and Figure 6 (c) capture first the merge into a single object and then a subsequent split back into two (the configuration prior to the change is shown as the fainter lines). Now with the two objects better Fig.6 here located, changes in the object labels from “unknown” to the correct labels occur, and the resolution of each template reduces. Finally Figure 6 (d) shows 50 overlaid samples from the posterior soon after the split step.

5.2 IDENTIFYING CELLS IN IMAGES OF CARTILAGE GROWTH

Our second example provides a more challenging test of the method in identifying cells, and also illustrates the estimator proposed in Section 4. The data shown in Figure 7 (a) been collected by imaging an area of cartilage growth using a confocal microscope. Within the proliferative and hypertropic zones of such growth plates two types of cells are usually present. These cells are differentiated on the basis of shape and size, one type being roughly spherical while the other type are generally smaller and more ellipsoid. Viewing the data in Figure 7 (a), is clear that identification is a difficult task, complicated by the fact that the object under study is truly 3-D and not physically sectioned (in fact these data form just one of a stack of optical sections through the specimen). Fig.7 here

Three object types are defined, approximations to a circle and an ellipse, and an “unknown” category (in this case, this is another approximation to a circle, but with a large variability permitted in the deformations). Parameters for the deformations have been chosen in an ad-hoc manner, simulating from the prior distribution to identify a reasonable parameter combination. For both the circle and the ellipse, $a_0 = 60$ and $a_1 = 0.03$ in (2), for the “unknown” type $a_0 = 10$ and $a_1 = 0.001$. The templates are allowed between 6 and 20 vertices, with n proportional to a Poisson distribution with mean 8 for circles and ellipses, and mean 1 for the “unknown”. In the mark mixture distribution, all object types have been assigned equal probability.

Figure 7 (b) and (c) show the delta estimate of the cell outlines and cell types respectively (with $\gamma = 1$ and $c = 10$ pixels). The different types of cell are denoted by different colours; in this case all cells are labelled as one of the true cell types and there are no “unknown”s. Notice that it is possible to estimate partially observed cells at the edges of the window by providing a guard boundary within which the data are assumed to be missing (points c are constrained to lie within the observed data window in order to avoid additional completely unobserved cells).

These two examples demonstrate the use of deformable template models in Bayesian object recognition. In particular, attention has focussed on ensuring efficient sampling via a combination of template types and complimentary MCMC updates. Resolution varying templates have provided an attractive tool in achieving this efficiency. A new estimator has been used based on a more appealing loss function than the usual MAP estimator. Work remains to be done in a number of areas, the most pressing being the issue of parameter estimation for the template models.

A Details of the MCMC algorithm

In this Appendix, we give the acceptance rates for the proposed update types. Since these proposed moves involve a change in the dimension of the variables involved, we follow the arguments given in Green (1995) to dimension match the current state $\boldsymbol{x}$ with the proposed state $\boldsymbol{x}'$.

A.1 Acceptance probabilities for births and deaths

Moves of the birth and death type for point processes are described by Geyer & Møller (1994) and Geyer (1996). Consider a proposed change which adds an additional cell to the configuration $\boldsymbol{x}$ taking it from k to $k+1$ cells. This new cell is born at a random location c in the window Λ with a type m, a resolution n and deformations $\boldsymbol{s}$ which are distributed according to $\pi(\boldsymbol{s}, n, m)$ (Eq. (10)). Notice that simulation of the required simple polygons is achieved by rejection sampling from the deformation model described in Section 2.1 without the non-self-intersecting constraint. The reverse move which would take this $k+1$ cell state $\boldsymbol{x}'$ back to a k cell state is that a randomly selected cell is deleted from the configuration. The components of $\boldsymbol{x}'$ are all the components of $\boldsymbol{x}$ (the parameters describing the common k cells) plus the additional random variables belonging to the new-born cell (its c, $\boldsymbol{s}$, n and m).

When the probabilities of proposing a birth move or a death move are equal, then following Eq. (8) of Green (1995) the acceptance probability for a proposed birth involves the product of the following terms: the ratio of the likelihood of the data given $\boldsymbol{x}'$ to that given $\boldsymbol{x}$, the ratio of the object prior density at $\boldsymbol{x}'$ and $\boldsymbol{x}$, the ratio of the probability of picking the new born cell to be deleted to the density of the new random variables $\boldsymbol{u}$ used in its birth, and a Jacobian term for the change of variable from $\boldsymbol{x}'$ to $\boldsymbol{x} \cup \boldsymbol{u}$. In this case, many of these terms cancel; since the variables $\boldsymbol{u}$ used in moving from $\boldsymbol{x}$ to $\boldsymbol{x}'$ are the parameters of the new cell, their proposal density cancels with the corresponding terms in the mark density. Similarly, the Jacobian term equals 1. Notice that it is possible to propose a new cell which overlaps the existing ones, but that (14) will always reject this move. The acceptance rate becomes

$$\alpha_{\text{birth}}(\boldsymbol{x} \to \boldsymbol{x}') = \min\left\{1, \frac{\pi(\boldsymbol{y} \mid \boldsymbol{x}')}{\pi(\boldsymbol{y} \mid \boldsymbol{x})} \times \beta I_{[\text{No cell overlap in } \boldsymbol{x}']} \times \frac{1}{k+1}\right\}. \tag{14}$$

A.2 Acceptance probabilities for resolution changes

A resolution move may increase or decrease the current resolution n by 1 (subject to each template type having a minimum or maximum allowable number of vertices). We will describe the acceptance probability for the move from n to $n' = n+1$ (assuming equal probabilities of proposing a move from n to $n+1$ as from $n+1$ to n).

$$\begin{aligned}\alpha_{\text{increase } n}(\boldsymbol{x} \to \boldsymbol{x}') = \min\Bigg\{1, &\frac{\pi(\boldsymbol{y} \mid \boldsymbol{x}')}{\pi(\boldsymbol{y} \mid \boldsymbol{x})} \times \frac{\pi^{**}(\boldsymbol{s}'|n', m)}{\pi^{**}(\boldsymbol{s}|n, m)} \times \frac{\pi(n'|m)}{\pi(n|m)} \times \frac{1/(n+1)}{1/n} \\ &\times \frac{\pi^{*}(\boldsymbol{s}_j^{(n)}, j \in U^{(n)} \mid \boldsymbol{s}_j^{(n)}, j \in F^{(n)})}{\pi^{*}(\boldsymbol{s}_j^{(n+1)}, j \in U^{(n+1)} \mid \boldsymbol{s}_j^{(n+1)}, j \in F^{(n+1)})} \\ &\times \prod_{j \in F^{(n)}} \left|g_j^{(n)}\right|^2 \Big/ \prod_{j \in F^{(n+1)}} \left|g_j^{(n+1)}\right|^2 \Bigg\},\end{aligned} \tag{15}$$

The first three terms are the ratios of the posterior density terms. The next term is the ratio of the probability of choosing the same vertex v_u for both the increase and the decrease steps. Then follows the ratio of the conditional Gaussian densities for generating the new deformations, and finally a Jacobian term relating the old and the new random variables. Notice that self-intersecting polygons are rejected, and that the ratio of the normalising constants for $\pi^{**}(\boldsymbol{s}, n, m)$ for the different n values must be evaluated; this has to be estimated off-line by simulating from the unconstrained models.

A.3 ACCEPTANCE PROBABILITIES FOR OBJECT TYPE CHANGES

The acceptance rate for proposing a move to object-type m' from m (keeping the position c and the vertices fixed, and hence the likelihood fixed) is based on the ratio

$$\alpha_{\text{model}}(\boldsymbol{x} \to \boldsymbol{x}') = \min\left\{1, \frac{\pi^{**}(\boldsymbol{s}|n, m')}{\pi^{**}(\boldsymbol{s}|n, m)} \times \frac{\pi(n|m')}{\pi(n|m')} \times \frac{\pi(m')}{\pi(m)} \times \left|\frac{\partial \boldsymbol{s}^{(n,m')}}{\partial \boldsymbol{s}^{(n,m)}}\right|\right\}, \tag{16}$$

where $\boldsymbol{s}^{(n,m')}$ indicates the deformations of the template interpreted under model m'. Again notice that in evaluating the ratio involving $\pi^{*}(\boldsymbol{s}|n, m)$, there is an unknown normalising constant which can be estimated off-line. To evaluate the Jacobian in (16), we transform $\boldsymbol{s}$ into the $2n-2$-dimensional subspace of independent identically distributed standard Gaussian variables $\boldsymbol{z}$, using the singular value decomposition of the covariance matrix $\boldsymbol{\Sigma}^{(n,m)} = \boldsymbol{V}^{(n,m)}\boldsymbol{\Lambda}^{(n,m)}\boldsymbol{V}^{(n,m)'}$, writing $\boldsymbol{s}^{(n,m)} = \boldsymbol{V}^{(n,m)}\boldsymbol{\Lambda}^{(n,m)^{1/2}}\boldsymbol{z}^{(n,m)} = \boldsymbol{A}^{(n,m)}\boldsymbol{z}^{(n,m)}$, and similarly $\boldsymbol{s}^{(n,m')} = \boldsymbol{A}^{(n,m')}\boldsymbol{z}^{(n,m')}$. Since the polygon and its location is held fixed, we have that $s_i^{(n,m)}g_i^{(n,m)} = s_i^{(n,m')}g_i^{(n,m')}$ for $i = 1, \ldots, n$, or $\boldsymbol{s}^{(n,m')} = \boldsymbol{G}^{(n,m,m')}\boldsymbol{s}^{(n,m)}$ for short. Furthermore,
$\pi(\boldsymbol{s}^{(n,m)}) = \pi(\boldsymbol{z}^{(n,m)})|\boldsymbol{A}^{(n,m)-}|$ where "$-$" indicates the pseudo-inverse of the matrix. We thus obtain the following expression for the Jacobian:

$$|\boldsymbol{A}^{(n,m')}|\,|\boldsymbol{A}^{(n,m')-}\boldsymbol{G}^{(n,m,m')}\boldsymbol{A}^{(n,m)}|\,|\boldsymbol{A}^{(n,m)-}|. \tag{17}$$

The determinant $|\cdot|$ is the product of all non-zero eigen-values of the matrix. Notice that the value of (17) is 1 if $m = m'$, and that (17) can be precomputed. The additional complexity results from the polygon closure constraint, since without this the determinant terms in (16) reduce to $\prod_{i=1}^{n}|g_i^{(n,m)}|^2/|g_i^{(n,m')}|^2$, as was the case with the Jacobian in (15).

A.4 A GIBBS STEP FOR REORDERING THE VERTICES

Not all template types exhibit rotational symmetry. For this reason, it may be useful to reorder vertices since different models may have different interpretations of the position of the first vertex. This type of move would be particularly useful in conjunction with model changes and split or merge steps. We propose to use a Gibbs sampler step to alter the start vertex by relabelling the vertices cyclically by an increment $o \in \{0, \ldots, n-1\}$ (note that this operation is a fixed dimension move). The vertex locations (and hence the likelihood), and all other variables do not change, with $s_i'g_i' = s_{i+o}g_{i+o}$ for all i and for increment $o \in \{0, \ldots, n-1\}$. Thus the conditional density for the Gibbs step is $\pi(o \mid \ldots) \propto \pi(\{s_{i+o}\} \mid n, m)$.

A.5 ACCEPTANCE PROBABILITIES FOR SPLIT AND MERGE MOVES

The moves which either merge objects A and B into a single object C, or split C into two objects A and B, involve two sources of dimension change. Firstly the number of cells alters by 1, giving a contribution to the acceptance probability from the prior on number of objects. Second, the numbers of vertices for C, n_C equals the total number of vertices for A and B minus 2 (Figure 5).

Denote the ordered sets of fixed indices F_A, F_B, F_C^A and F_C^B, and the ordered sets of updated indexes U_A, U_B. The sets F_A and F_C^A map onto each other, as do F_B and F_C^B. In splitting, let p_1 denote the vertex of A in the upper disc in Figure 5 (b), and let p_2 be the corresponding vertex in the lower disc. Both p_1 and p_2 are uniformly distributed in the disc which has diameter ϵ. Under the closure constraint there is a linear one-to-one mapping between $\boldsymbol{p} = [p_1, p_2]'$ and $\{s_j^{(n_A)}, j \in U_A\}$ and $\{s_j^{(n_B)}, j \in U_B\}$, which we write as $\boldsymbol{s}_U = \boldsymbol{D}\boldsymbol{p} + \boldsymbol{d}$. Assuming equal probabilities of proposing a split or a merge, the acceptance rate becomes[1]

$$\alpha_{\text{merge}}(\boldsymbol{x} \to \boldsymbol{x}') = \min\left\{1, \frac{\pi^{**}(\boldsymbol{x}' \mid \boldsymbol{y})}{\pi^{**}(\boldsymbol{x} \mid \boldsymbol{y})} \times \frac{\text{pr(propose to split object } C \text{ into } A \text{ and } B)}{\text{pr(propose to merge objects } A \text{ and } B \text{ into } C)} \times \right.$$
$$\left. \frac{|\boldsymbol{D}^-|}{(\pi(\epsilon/2)^2)^2} \times \frac{n_C}{n_A n_B} \times \frac{\prod_{j \in F_A} \left|g_j^{(n_A)}\right|^2 \prod_{j \in F_B} \left|g_j^{(n_B)}\right|^2}{\prod_{j \in F_C^A \cup F_C^B} \left|g_j^{(n_C)}\right|^2} \right\}. \tag{18}$$

The first term is the ratio of the posterior densities (this takes into account overlapping or self-intersecting objects, giving them a zero acceptance probability); again note the need for normalising constants to be estimated off-line. The next term is the ratio of the probability of proposing to split C and that of merging A and B. These terms take into account the numbers of pairs of cells which might be selected for merging, plus the number of ways the cell C could be split. The next term is the density for $\boldsymbol{p}$ reparameterised as the density for $\boldsymbol{s}_U$, followed by the ratio of the probability for proposing the particular labelling of objects A, B and C (there are n_A ways of selecting a vertex of A as the first vertex etc). The last term is the Jacobian for the change of variables. Note that it is straightforward to extend these split and merge steps to allow a stochastic number of vertices to be lost or gained in these move types.

ACKNOWLEDGEMENTS

The authors are grateful to the Research Council of Norway and the British Council for funding under their scholarship scheme for collaborative research. The second author also acknowledges the financial support of the Nuffield Foundation. Thanks to Dr Nick White, Dr Rachel Errington and the Plant Science Department at Oxford University for describing the application and providing data.

REFERENCES

AMIT, Y. & MANBECK, K. M. (1993). Deformable template models for emission tomography, *IEEE Transaction on Medical Imaging* 12: 260–269.

AMIT, Y., GRENANDER, U. & PICCIONI, M. (1991). Structural image restoration through deformable templates, *Journal of the American Statistical Association* 86: 376–387.

BADDELEY, A. J. (1992). Errors in binary images and a L^p version of the Hausdorff metric, *Nieuw Archief voor Wiskunde* 10: 157–183.

[1]There is a misprint in the previous version of this report where $\frac{n_A n_B}{n_C}$ occured in (18). The correct term is $\frac{n_C}{n_A n_B}$.

BADDELEY, A. J. & MØLLER, J. (1989). Nearest–neighbour Markov point processes and random sets, *International Statistics Review* 57(2): 89–121.

BADDELEY, A. J. & VAN LIESHOUT, M. N. M. (1993). Stochastic geometry models in high–level vision, *in* K. V. Mardia & G. K. Kanji (eds), *Statistics and Images*, Vol. 20, Abingdon: Carfax Publishing, chapter 11, pp. 235–256.

BLAKE, A. & YUILLE, A. (1992). *Active Vision*, Massachusetts Institute of Technology Press.

CLIFFORD, P. & NICHOLLS, G. (1994). A Metropolis sampler for polygonal image reconstruction, *Technical report*, Department of Statistics, Oxford University.

FRIGESSI, A. & RUE, H. (1997). Bayesian image classification with Baddeley's delta loss, *Journal of Computer Graphics and Statistics* 6(1): 55–73.

GEYER, C. (1996). Likelihood inference for spatial point processes, *in* W. S. Kendall (ed.), *Proc. Seminaire Européen de Statistique Tolouse 1996, Stochastic Geometry: Theory and Applications*, Springer Lecture Notes.

GEYER, C. J. & MØLLER, J. (1994). Simulation procedures and likelihood inference for spatial point processes, *Scandinavian Journal of Statistics* 21: 359–373.

GILKS, W. R., RICHARDSON, S. & SPIEGELHALTER, D. J. (1996). *Markov Chain Monte Carlo in Practice*, Chapman & Hall.

GREEN, P. J. (1995). Reversible jump MCMC computation and Bayesian model determination, *Biometrika* 82(4): 711–732.

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

GRENANDER, U., CHOW, Y. & KEENAN, D. M. (1991). *Hands: a Pattern Theoretic Study of Biological Shapes*, Research Notes on Neural Computing, Springer, Berlin.

GRENANDER, U. & MILLER, M. I. (1994). Representations of knowledge in complex systems (with discussion), *Journal of the Royal Statistical Society, Series B* 56(4): 549–603.

HELTERBRAND, J. D., CRESSIE, N. & DAVIDSON, J. L. (1994). A statistical approach to identifying closed object boundaries in images, *Advances in Applied Probability* 26(4): 831–854.

HURN, M. A. (1996). Bayesian image analysis in confocal fluorescence microscopy, *Statistical Research Report 96:01*, School of Mathematical Sciences, Univeristy of Bath.

JAIN, A. K., ZHONG, Y. & LAKSHMANAN, S. (1996). Object matching using deformable templates, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18(3): 267–278.

KENT, J. T., MARDIA, K. V. & WALDER, A. N. (1996). Conditional cyclic Markov random fields, *Advances in Applied Probability (SGSA)* 28: 1–12.

MARDIA, K. V. & HAINSWORTH, T. J. (1993). Image warping and Bayesian reconstruction with grey–level templates, *in* K. V. Mardia & G. K. Kanji (eds), *Statistics and Images*, Vol. 20, Abingdon: Carfax Publishing, chapter 12, pp. 257–280.

NICHOLLS, G. (1997). Bayesian image analysis with Markov chain Monte Carlo and colored continuum triangulation models, *Journal of the Royal Statistical Society, Series B.* (to appear).

PIEVATOLO, A. & GREEN, P. J. (1995). Object restoration through dynamic polygons, *Technical Report 12*, University of Bristol.

QIAN, W. & MARDIA, K. V. (1995). Recognition of multiple objects with occlusions, *Stat 95/01*, University of Leeds.

QIAN, W., TITTERINGTON, D. M. & CHAPMAN, J. N. (1996). An image analysis problem in electron microscopy, *Journal of the American Statistical Association* 191(435): 944–952.

RUE, H. (1995). New loss functions in Bayesian imaging, *Journal of the American Statistical Association* 90: 900–908.

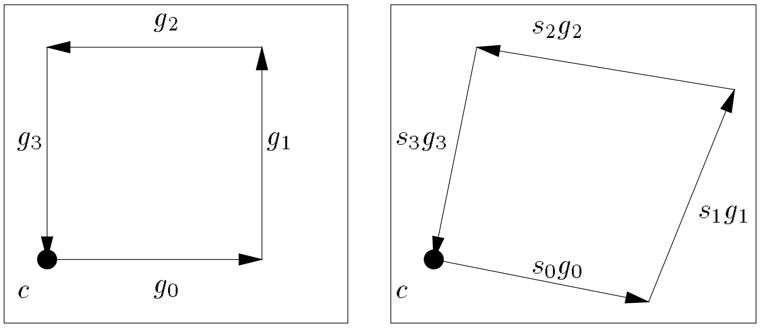
RUE, H. (1997). A loss function model for the restoration of grey level images, *Scandinavian Journal of Statistics* 24(1): 103–114.

RUE, H. & SYVERSVEEN, A. R. (1995). Bayesian object recognition with Baddeley's delta loss, *Statistics no. 8*, Departments of Mathematics, The University of Trondheim, Trondheim, Norway.

STORVIK, G. (1994). A Bayesian approach to dynamic contours, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16(10): 976–986.

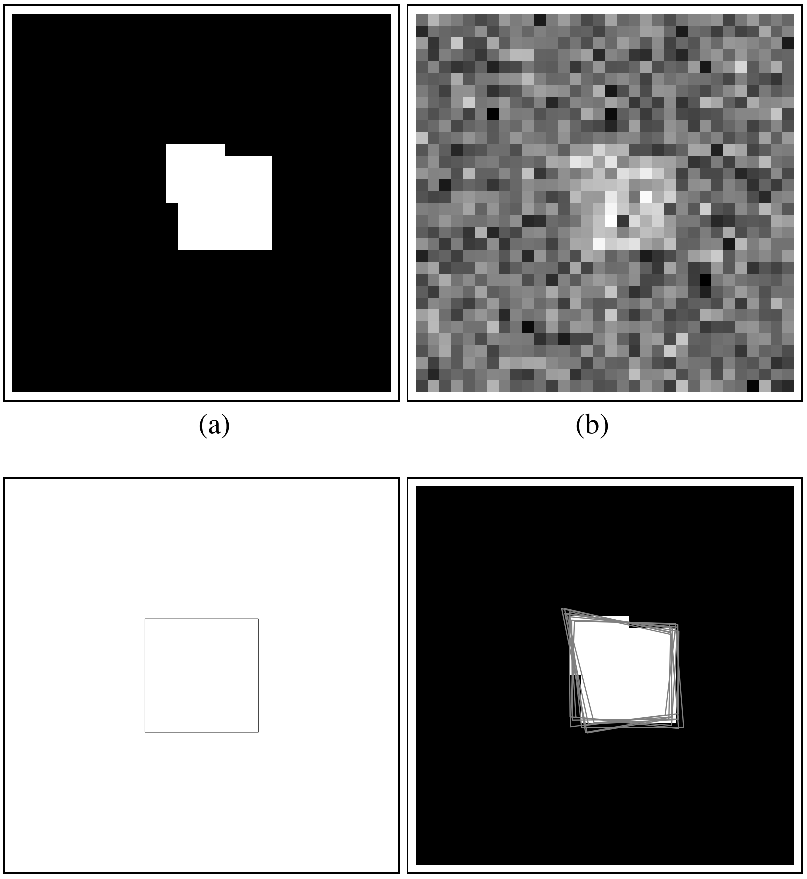
TJELMELAND, H. & BESAG, J. (1996). Markov random fields with higher order interactions, *Statistics no. 1*, Departments of Mathematics, Norwegian University of Science and Technology, Trondheim, Norway.

VAN LIESHOUT, M. N. M. (1995). Markov point processes and their applications in high-level imaging (with discussion), *Bulletin of the International Statistical Institute* LVI, BOOK 2: 559–576.



(a) (b)

FIGURE 1: (a) The undeformed square template, (b) a deformation holding the location c fixed.



(a) (b)

(c) (d)

FIGURE 2: (a) The true pixellated image, (b) the data, (c) the square template, (d) samples of the deformed template overlaid on the true scene.

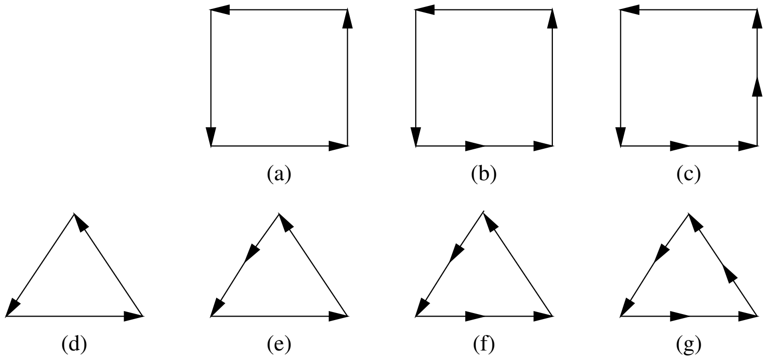


FIGURE 3: Examples of templates with variable resolutions: (a)-(c) Square templates, (d)-(g) triangular templates.

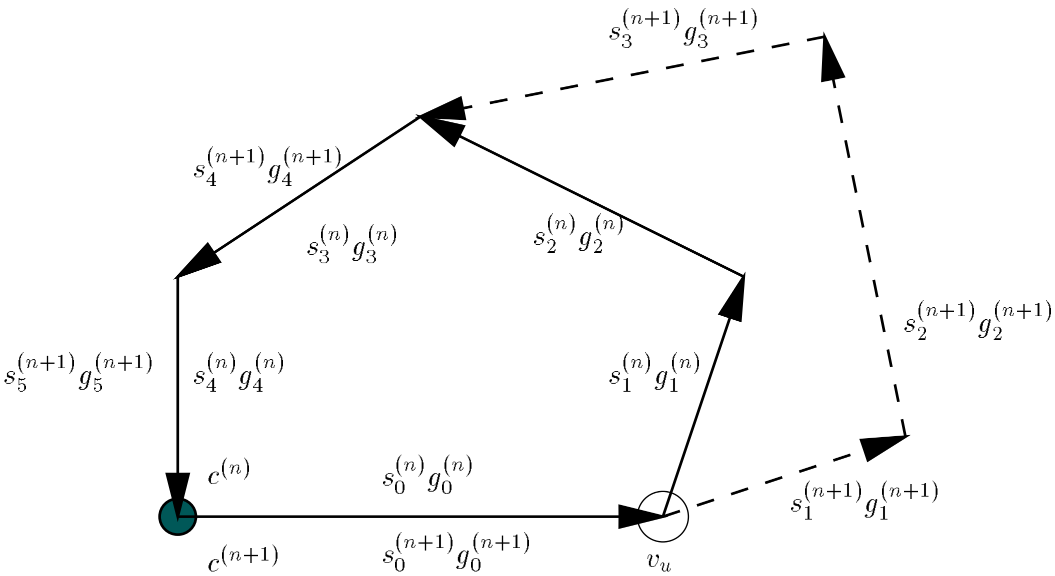
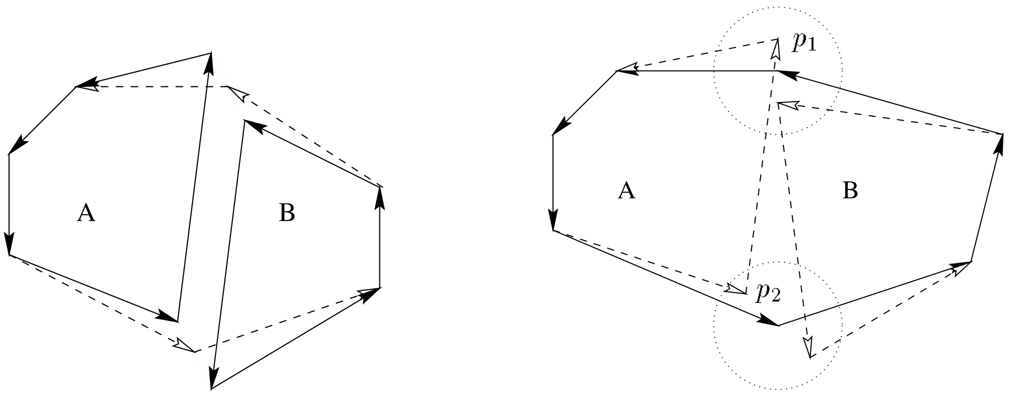
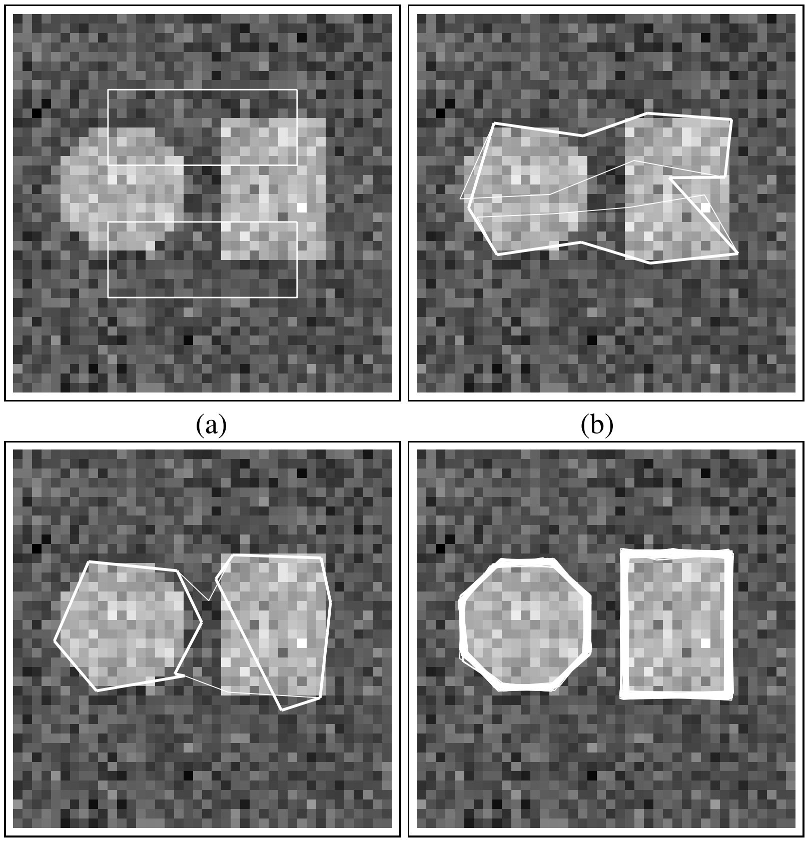


FIGURE 4: Altering the template resolution.



(a) (b)

FIGURE 5: (a) Merging two deformed templates, (b) the reverse split move.

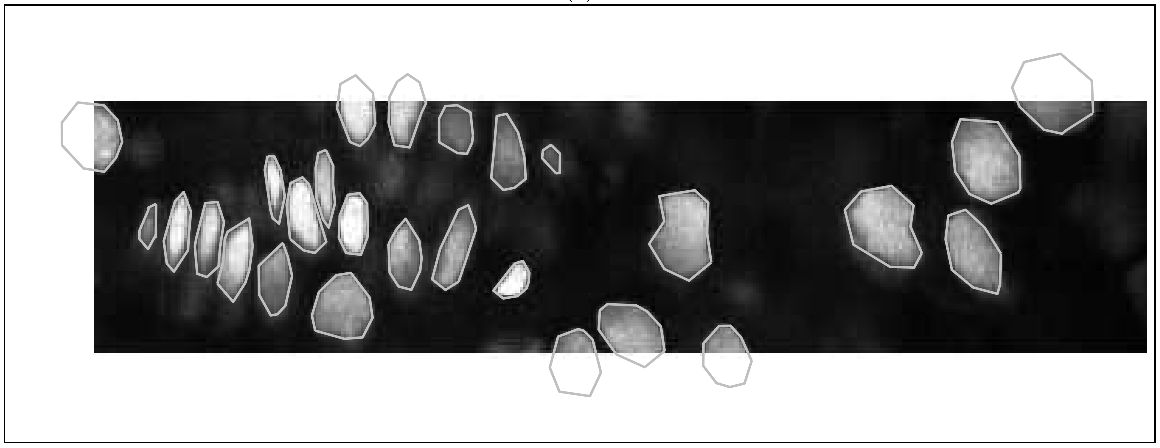


(a) (b)

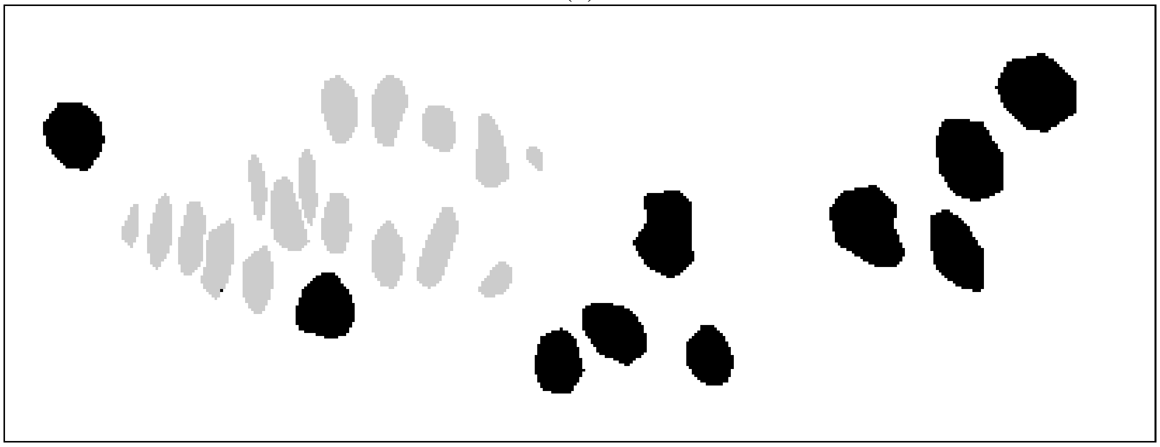
(c) m (d)

FIGURE 6: (a) The initial configuration, (b) an intermediate merge step, (c) an intermediate split step, (d) samples from the posterior overlaid on the data.

(a)



(b)



(c)

FIGURE 7: (a) A section of confocal data, (b) the estimated cell outlines, (c) the estimated cell labels.