A DATA–DRIVEN BAYESIAN SAMPLING SCHEME FOR UNSUPERVISED IMAGE SEGMENTATION

E. Clark and A. Quinn

Department of Electronic and Electrical Engineering, University of Dublin
Trinity College, Dublin 2, Ireland.
eclark@ee.tcd.ie ; aquinn@tcd.ie

ABSTRACT

A Bayesian scheme for fully unsupervised still image segmentation is described. The likelihood function is constructed by assuming that the grey level at each pixel site is a realization of a Gaussian random variable of unknown parameters, there being an uncertain number of distinct Gaussian classes in the image. Spatial connectivity between pixels is encouraged via a Markov random field prior. The task of identifying the model parameters and recovering the underlying class label at each site (i.e. segmentation) is accomplished using a novel reversible jump Markov chain Monte Carlo (MCMC) scheme. This scheme explores the space of possible segmentations via proposals that are driven by the actual image realization—so-called data-driven proposals. The aim is to (i) induce good mixing in regions of high probability, and (ii) to optimize the acceptance probability of the proposals. A key development is a stochastic version of a recursive labeling algorithm which has been used in previous work for fast image region splitting. In the current stochastic context, it yields fast and effective split and merge proposals. The performance of the novel MCMC scheme is illustrated in simulation.

1. INTRODUCTION

Segmentation may be defined as the task of dividing an image into regions of strict stationarity. Each distinct model from which pixels are realized is referred to as a class. In the most general case, neither the number of classes, nor the paramters of each class, is known *a priori*. Inference, then, of these unknowns, along with a class label at every pixel site constitutes the *unsupervised* image segmentation problem.

Segmentation constitutes an ill–posed inverse problem, possessing no unique solution [1]. A fully Bayesian approach is adopted. It explores, in a regularized manner, the space of possible segmentations. Specifically, probability is distributed over the segmentation space in the light of data (i.e. the image realization) and the modelling assumptions. Because of the high dimensionality of the space and the complexity of the posterior probability distribution, analytical evaluation or optimization are infeasible. It is for this reason that stochastic sampling techniques have become important in image segmentation [2, 3]. This paper contributes to the literature on the design of effective samplers for such problems. A Markov random field (MRF) prior (the Pott's model) on the class labels is adopted to encourage spatial connectivity. The pixels themselves are modelled as being conditionally iid Gaussian, an appropriate model for many measurement and degradation [1] processes. Note that, in the case of this model, segmentation is interpretable as image denoising or restoration, since the inference of a label field is equivalent to recovering the mean field in the presence of additive noise. Note, also, that this problem is closely related to order-uncertain Gaussian mixture model clustering [4], but with the class generation process modelled as a (non-causal) Markov process, rather than as iid.

This work was supported by the Forbairt Grant FR/97/012.

2. SEGMENTATION MODEL

2.1. Basic Definitions

Let $\Xi = \{i=(a,b) : 0 \leq a < N\,,\ 0 \leq b < M\}$ be the site lattice for an image of dimension $N \times M$, and $\eta_i = \{(i \pm j) \cap \Xi : j \in \{(1,0),(0,1)\}\}$ be the index set to a first order neighbourhood of site $i \in \Xi$. We write $y = \{y_i : y_i \in [0,1)\,,\ i \in \Xi\}$ for the observed image data. Let class parameter $k \in \{1,2,\ldots,P\}$ denote the (unknown) number of classes, and $x \in \mathcal{D}_k = \{1,2,\ldots,k\}^{M\times N}$ be a realization of a k–class label field over Ξ. The data model is

$$\Pr(y_i|x_i,\mu_{x_i},\sigma^2_{x_i}) = \frac{1}{Z_L(x_i)\sqrt{2\pi\sigma^2_{x_i}}} \exp\left[-\frac{(y_i-\mu_{x_i})^2}{2\sigma^2_{x_i}}\right], \tag{1}$$

where $\mathrm{Z_L}(\cdot)$ accounts for the truncation of the Gaussian outside $[0,1)$. Under the asssumption that the density is concentrated away from the boundaries, it is assumed forthwith that $Z_L(\cdot) \approx 1$.

All the Gaussian parameters for the k-class problem are denoted $\theta = (\mu,\sigma^2) \in \mathcal{C}_k = \mathbb{R}^k \times \mathbb{R}^k_+$, where μ and σ^2 are k-length vectors. Finally, $z = (k,x,\theta)$ is a point in the k-class segmentation subspace $\Omega_k = \{k\} \times \mathcal{D}_k \times \mathcal{C}_k$, and $\Omega = \Omega_1 \oplus \Omega_2 \ldots \oplus \Omega_P$, being the non–intersecting sum of all k–class segmentation subspaces, is the complete segmentation space over which inference is perfromed.

Some useful operators can now be defined : $\omega_l(x) = \{i : x_i = l\,,\ i \in \Xi\}$ is the support of label l in label field x ; $\mathcal{K}(z) = \{1,2,\ldots,k\}$ is the set of class labels realizable in state z ; $\mathcal{U}(\cdot)$ denotes a uniform probability mass assignment over the elements of finite set $\cdot$; $|\cdot|$ denotes the cardinality of a countable set, unless specified otherwise.

2.2. The Prior, Likelihood and Posterior Distributions

The prior may be expressed as a product of distributions,

$$\Pr(z) = \Pr(x|k)\Pr(\theta|k)\Pr(k), \tag{2}$$

where x and θ are conditionally independent, given k. A Pott's model (colour-blind MRF) is adopted *a priori* for the label field :

$$\Pr(x|k) = \frac{1}{Z_P(k\,;\beta)} \exp\left\{\frac{\beta}{2} \sum_{i\in\Xi} \sum_{j\in\eta_i} (2\delta_{[x_i,x_j]} - 1)\right\},$$

where $\mathrm{Z_P}(k\,;\beta) = \sum_{x\in\mathcal{D}_k} \exp(\cdot)$ is the normalization constant (partition function), and $\delta_{[\cdot,\cdot]}$ is the Kronecker delta function. The hyper–parameter β controls the degree of connectivity, encouraging clustering of similar labels if greater than 0, or anti-clustering if less than 0. In our experiments, we take β to be 1.5. Explicit calculation of $\mathrm{Z_P}(k\,;\beta)$ is not possible as this would involve summing $|\mathcal{D}_k| = k^{N\times M}$ terms. Thus, where the need arises, we adopt an approximation, as motivated by the pseudo-likelihood in [3] :

$$\tilde{\mathrm{Z}}_{\mathrm{P}}(z\,;\beta) = \prod_{i\in\Xi} \sum_{l\in\mathcal{K}(z)} \exp\left\{\beta \sum_{j\in\eta_i} (2\delta_{[l,x_j]} - 1)\right\}.$$

The distribution on parameters θ is given by

$$\Pr(\theta|k) = \prod_{i=1}^{k} \Pr(\mu_i)\Pr({\sigma^2}_i),$$

where $\Pr(\mu_i)$ is taken to be uniform on $[0,1)$, and $\Pr({\sigma^2}_i)$ to be Jeffreys' (i.e. proportional to ${\sigma_i^{-2}}$) on $(0,1)$. It is assumed that $K = k$ is distributed as $\mathcal{U}(\{1,2,\ldots,P\})$, with $P = 20$. From (1), the likelihood function for the image is

$$\Pr(y|z) = \prod_{i\in\Xi} \frac{1}{\sqrt{2\pi\,{\sigma^2}_{x_i}}} \exp\left\{-\frac{(y_i - \mu_{x_i})^2}{2{\sigma^2}_{x_i}}\right\}. \qquad (3)$$

Bayes theorem provides the relationship between the posterior distribution and equations (2) and (3):

$$\Pr(z|y) \propto \Pr(y|z)\Pr(z). \qquad (4)$$

3. SAMPLING FROM THE POSTERIOR DISTRIBUTION

The Metropolis-Hastings (MH) algorithm [4, 5, 6] is a procedure for sampling from arbitrary distributions that need only be available up to a multiplicative constant. It is this feature of the algorithm that greatly simplifies the problem of sampling from the posterior (4). For example, a realization of a truncated Markov chain, yeilding samples from a target distribution, $\pi(z)$, is obtained by repeated iteration of the MH algorithm, which is fully specified in the following two-step process :

- `From current state` z`, generate a proposed state` z' `by sampling from proposal distribution` $g(z'|z)$
- `Accept proposed state` z' `with probability` $\alpha(z'|z) = \min\{1, A\}$`, else accept` z

Here the acceptance ratio is

$$A = \left\{\frac{\pi(z')}{\pi(z)} \frac{g(z|z')}{g(z'|z)}\right\}. \qquad (5)$$

Note that when $\pi(z) = \Pr(z|y)$ in (4), the normalization constant need not be calculated, as it cancels in (5). This represents a key advantage over other rejection-based sampling techniques. One is free to choose the proposal distribution $g(\cdot|\cdot)$ providing that it meets the technical requirements for Markov chain convergence. Further detail may be found in [5], but we emphasize the following: (i) *detailed balance (reversibility)*, which demands that, for all pairs of states, z and z', each of non-zero posterior probability, we must have $g(z|z') > 0$ iff $g(z'|z) > 0$; (ii) *irreducibility* over Ω, which demands that the proposal distribution must, with non-zero probability, potentially allow the chain to reach all states of non–zero target density, independently of the starting state. Gibbs' samplers, which sample random variables, without rejection, from their full conditional, have also been employed in image segmentation [2] with a known number of classes, but are intractable when proposing jumps between models of different order, such as is necessary in the unsupervised case. The technical demands placed on the MH proposals in this latter case are satisfied by the reversible jump [6] techniques considered in Section 4.4.

4. MH PROPOSALS FOR SEGMENTATION

We present four proposal algorithms below. Algorithms L, M and V within themselves satisfy the reversibility condition, and as a family provide irreducibility over each Ω_k. The algorithm pair $(\mathrm{J_M}, \mathrm{J_D})$ provides the dynamic needed to step between segmentation subspaces. The first operation in generating a MH proposal for sampling from (4) is to select stochastically one of the algorithms from {L, M, V, $(\mathrm{J_M}, \mathrm{J_D})$} with respective probabilities $\{0.99, 0.004, 0.004, (0.001, 0.001)\}$, and then to pass control to that algorithm. Each proposal algorithm is now presented.

4.1. L : **Propose Label**

Algorithm :

- `Sample label index` $i \sim \mathcal{U}(\{1,2,\ldots,|\Xi|\})$
- `Sample proposed label variable from full conditional` $x_i' \sim \Pr(\cdot|y, z\setminus\{x_i\})$`, where, from (4):`

$$\Pr(X_i = x_i'|y, z\setminus\{x_i\}) \propto \frac{1}{\sqrt{{\sigma^2}_{x_i'}}} \exp\left\{-\frac{(y_i - \mu_{x_i'})^2}{2{\sigma^2}_{x_i'}}\right\} \times \exp\left\{\beta \sum_{j\in\eta_i} (2\delta_{[x_i',x_j]} - 1)\right\}.$$

Here the proposal is both *data–driven* and *state–driven*. Since the proposal distribution and target distribution from which we are aiming to sample are the same in this case (i.e. the full conditional of X_i), then $A = 1$ (5). Hence, the proposal is always accepted.

4.2. M : **Propose Mean**

Algorithm :

- `Sample parameter index` $i \sim \mathcal{U}(\mathcal{K}(z))$
- `Propose new mean` $\mu_i' = \mu_i + \mathcal{N}(0, 25)$

Here the proposal is state–driven. As the Gaussian density associated with the perturbation term is an even function, the ratio of generation probabilities is one. The acceptance ratio is then given by

$$A = \prod_{j\in\omega_i(x)} \exp\left\{\frac{(y_j - \mu_i)^2 - (y_j - \mu_i')^2}{2\,{\sigma^2}_i}\right\}$$

for $\mu_i' \in [0,1)$. Otherwise, $A = 0$.

4.3. V : **Propose Variance**

The algorithm for proposing a new variance is analogous to that for the mean. The acceptance ratio is now given by

$$A = \frac{\Pr({\sigma^2}'_i)}{\Pr({\sigma^2}_i)} \prod_{j \in \omega_i(x)} \sqrt{\frac{{\sigma^2}_i}{{\sigma^2}'_i}} \exp\left\{ \frac{(y_j - \mu_i)^2 \, ({\sigma^2}'_i - {\sigma^2}_i)}{2\,{\sigma^2}_i {\sigma^2}'_i} \right\}.$$

4.4. $(\mathrm{J_M}, \mathrm{J_D})$: **Propose Reversible Jump**

The reversible jump proposal comprises two algorithms: (i) a *merge* algorithm, $\mathrm{J_M}$, which selects two classes and proposes to merge them into one new class; (ii) a *divide* algorithm, $\mathrm{J_D}$, which selects one class and proposes to divide it into two new classes. As the original and proposed states are from segmentation subspaces of differing dimension, the required reversible jump MH acceptance ratio is [6]

$$A = \left\{ \frac{\pi(z')}{\pi(z)} \, \frac{g(z|z',n')q'(n')}{g(z'|z,n)q(n)} \, |J| \right\}. \tag{6}$$

Here q and q' are the respective distributions on auxiliary variables n and n', and $|J|$ is the magnitude of the determinant of the Jacobian for some bijective transformation of between the continuous variables of (z, n) and (z', n'). It is worthwhile noting that the generality of equation (6) reflects the degree of choice open to the designer of a reversible jump process. The generation distribution, auxiliary variables, and their respective distributions are unspecified, and it is a challenge to the designer to define these in a way which addresses the particular reversible jump problem, while, at the same time, satisfying the bijective requirement and the requirements for Markov chain convergence, set out in Section 3. When setting up a reversible jump for the segmentation problem, we seek to maintain reversibility between the merge and divide proposals. For this, we need stochastic schemes for the reversible merging and division of parameters, and for the binary labelling of sites.

State-driven proposals were adopted for algorithms M and V above, and since each is exploring in a single dimension, the proposal schemes are (i) fast, and (ii) yield good acceptance probabilities. In the case of the high dimensional jump proposal J_D however, there is very little state information available *a priori* to base satisfactory proposals on. Further more, very low generation probabilities are encountered from independently sampling each dimension of the multi–dimensional proposal space, an effect we will call *dilution*. To overcome the first of these problems, we employ proposal schemes which combine a deterministic (data–driven) forcing terms with a stochastic perturbation. The inclusion of a forcing term makes it possible to concentrate generation probability around data respecting states, thereby encouraging 'sensible' proposals. Good deterministic schemes for parameter inference—such as the EM or K-means [7] clustering algorithms—are readily available. So also is a fast binary labelling scheme—the Recursive Unanimity Rule (RUR) [7]—which samples on sparse grids of pixel sites, massively reducing dilution effects. The proposal algorithms are now presented, followed by details of the 'stochasticized' schemes for parameter selection (via a stochastic clustering algorithm (CA)) and relabelling (via the stochastic RUR (i.e. SRUR) algorithm).

$\mathrm{J_D}$: Divide Class Proposal

This algorithm divides a single class d in $z = (k, x, \theta)$, giving classes d_+ and d_- in the proposal $z' = (k' = k + 1, x', \theta')$.

Algorithm :

- `If` $k = K$, `set` $A = 0$ `and return`
- `Sample class index` $d \sim \mathcal{U}\{\mathcal{K}(z)\}$
- `Calculate effective values of reverse proposal auxiliary variables` $n_\uparrow{}' = \theta_d - \mathrm{CA1}(d, y, z)$
- `Sample forward proposal auxiliary variables` $n_\downarrow \sim (\mathcal{N}(0,5), \mathcal{N}(0,5), \mathcal{N}(0,5), \mathcal{N}(0,5))$
- `Calculate proposed parameters` $(\theta'_{d_+}, \theta'_{d_-}) = \mathrm{CA2}(d, y, z) + n_\downarrow$
- `Sample the proposed labelling via SRUR scheme` $x' \sim \Pr(\cdot|x, y, d, d_+, d_-, \theta')$

From (6), the acceptance ratio is given by

$$\begin{aligned} A \;=\; & \prod_{i \in \omega_d(x)} \sqrt{\frac{{\sigma^2}_d}{{\sigma^2}'_{x'_i}}} \exp\left\{ \frac{(y_i - \mu_d)^2}{2\,{\sigma^2}_d} - \frac{(y_i - \mu'_{x'_i})^2}{2\,{\sigma^2}'_{x'_i}} \right\} \\ & \times \left(\frac{\tilde{Z}_{\mathrm{P}}(z\,;\beta)}{\tilde{Z}_{\mathrm{P}}(z'\,;\beta)} \right) \exp\left\{ \beta \sum_{i \in \omega_d(x)} \sum_{j \in \eta_i \cap \omega_d(x)} (\delta_{[x'_i, x'_j]} - 1) \right\} \\ & \times \frac{\Pr(\mu'_{d+}) \Pr({\sigma^2}'_{d+}) \Pr(\mu'_{d-}) \Pr({\sigma^2}'_{d-})}{\Pr(\mu_d) \Pr({\sigma^2}_d)} \\ & \times \frac{2\,q'({N_\uparrow}' = {n_\uparrow}')}{(k+1)\Pr(X' = x'|x, y, d, \theta'_{d+}, \theta'_{d-})\, q(N_\downarrow = n_\downarrow)}. \end{aligned} \tag{7}$$

Here, the ratio of terms separated by multiplication signs are due to the likelihood, the Pott's prior, the parameter prior, and the generation probabilities. Note that $|J| = 1$ for the transformation $(\theta, n) \leftrightarrow (\theta', n')$.

$\mathrm{J_M}$: Merge Classes Proposal

This algorithm merges classes in m_+ and m_- in $z = (k, x, \theta)$ to give class m in the proposal $z' = (k' = k - 1, x', \theta')$.

Algorithm :

- `If` $k = 1$, `set` $A = 0$ `and return`
- `Sample two class indices,` $m_+ \sim \mathcal{U}\{\mathcal{K}(z)\}$ `repeat` $m_- \sim \mathcal{U}\{\mathcal{K}(z)\}$ `while` $[m_+ = m_-]$
- `Reorder indices` m_+, m_- `such that` $\mu_{m_+} > \mu_{m_-}$
- `Calculate effective values of reverse proposal auxiliary variables` $n_\downarrow{}' = (\theta_{m_+}, \theta_{m_-}) - \mathrm{CA2}(m, y, z')$
- `Sample forward proposal auxiliary variables` $n_\uparrow \sim (\mathcal{N}(0,5), \mathcal{N}(0,5))$
- `Calculate proposed parameters` $\theta'_m = \mathrm{CA1}(m, y, z') + n_\uparrow$
- `Calculate SRUR generation probability for current labelling` x, $\Pr(X = x|x', y, m, m_+, m_-, \theta)$

As the divide and merge routines are the reversible counterparts of each other, the acceptance ratio for $\mathrm{J_M}$ is the reciprocal of that for $\mathrm{J_D}$ (7).

Clustering Algorithms : CA1 , CA2

$\mathrm{CA1}(d, y, z)$ and $\mathrm{CA2}(d, y, z)$ denote respectively one and two centroid clustering algorithms as run over data set $\{y_i : i \in \omega_d(x)\}$. CA1 simply returns the mean and variance for the whole data set. CA2, with initial centroid positions set at the supremum and infimum of the data set, performs a two-partition of the data, and returns the mean and variance of the data in each of the resulting partitions. In adding an auxiliary variable to the data-driven output of the CA algorithms, the stochastic diffusion behaviour necessary for reversibility and irreducibility of the parameter splitting scheme is ensured.

Binary Relabelling: the SRUR algorithm.

In the $\mathrm{J_D}$ (divide) algorithm above, pixels in $\omega_d(x)$ are relabelled via a stochastic binary labelling process. The key requirement of this process, as dictated by reversibility, is that for all $\Lambda \subseteq \Xi$, the process must be able to generate all binary labellings over Λ with non–zero probability. Furthermore, it is necessary that the forward generation probability of any such binary labelling proposal be attainable. A simple scheme which meets both these needs is the independent Bernoulli sampling of each label. However, this independent sampling over a large set of sites induces dilution effects. Specifically, the generation probability, $g(z'|z, n)$, for any proposed split is small in comparison to the probability, $g(z|z', n')$, of the reverse merge proposal. From (6), the second ratio dominates the first, leading to far higher acceptance probabilities for a split, than for a merge, proposal, *irrespective* of $\pi(z')/\pi(z)$.

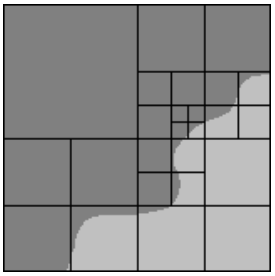
The Stochastic Recursive Unanimity Rule (SRUR) is a stochastic extension of a deterministic data-driven scheme for *sparse* labelling, leading to fast and accurate segmentations [7]. While details are not provided here owing to lack of space, the essential feature is that SRUR can reach—with non-zero probability—any binary labelling pattern for a square window of size $w_0 = 2^n$. A ternary decision among the following actions is made: (i) unanimous labelling with d_+ (probability p_{d_+}); (ii) unanimous labelling with d_- (probability p_{d_-}); (iii) split window into four sub-windows, and recurse the procedure (probability p_s). Fig. 1 conceptually illustrates this idea. The algorithm follows a pseudo-unique trajectory to each possible labelling outcome, thereby greatly simplifying the problem of calculating the generation probability. The action probabilities above are generated from independent Bernoulli trials (based on binary class membership in d_+ and d_-) at each of the four corners of the window. Then:

$$\begin{aligned} p_{d_+} &= \Pr(\text{4 or 3 corners labelled } d_+), \\ p_s &= \Pr(\text{2 corners labelled } d_+), \\ p_{d_-} &= \Pr(\text{1 or 0 corners labelled } d_+). \end{aligned}$$

The label sampling step of $\mathrm{J_D}$, i.e. $x' \sim \Pr(\cdot|x, y, d, d_+, d_-, \theta')$, invokes the SRUR algorithm on the windows tessellating $\omega_d(x)$.

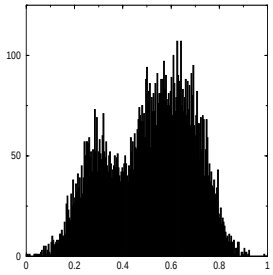
5. SIMULATION AND DISCUSSION

Fig. 2 shows a synthetic 100×100-pixel image, composed of regions drawn from a palette of six grey levels, and corrupted by iid (i.e. mean-independent) noise, such that the average SNR over the entire image is +33 dB. Fig.3 shows the corresponding gray level histogram. Starting from a random initial configuration, the segmentation result presented in Fig. 4 is a sample from the posterior distribution obtained after 5×10^6 MH steps.



1. SRUR splitting.

2. Synthetic data.



3. Grey level histogram.

4. Segmentation.

There is clear evidence of convergence to an excellent segmentation (restoration), even without the need for annealing. A full consideration of the convergence speed-ups achieved by way of the data-driven proposals developed in this paper will be reported shortly. The viability of reversible jump MCMC algorithms in high dimensional or speed-sensitive applications, such as the one described here, will be enhanced by data-driven proposal design.

6. REFERENCES

[1] Haluk Derin and Howard Elliot. Modeling and Segmentation of Noisy and Textured Images Using Gibbs Random Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(1), January 1987.

[2] Stuart Geman and Donald Geman. Stochastic Relaxation, Gibbs distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6), November 1984.

[3] J. Besag. On the Statistical Analysis of Dirty Pictures. *Journal of the Royal Statistical Society, Series B*, 48:259–302, 1986.

[4] Sylvia Richardson and Peter J. Green. On Bayesian Analysis of Mixtures with an Unknown Number of Components. *Journal of the Royal Statistical Society*, 59(4):731–792, 1997.

[5] Alan D. Sokal. Monte Carlo Methods in Statistical Mechanics: Foundations and New Algorithms. Lecture Notes, Department of Physics, New York University, 1989.

[6] Peter J. Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732, 1995.

[7] Julien Reichel and Anthony Quinn. A Fast and Fully Unsupervised Scheme for Model–Based Image Segmentation. In *Bayesian inference for Inverse Problems*. SPIE Conference, 1998.