Multiresolution Random Fields and Their Application to Image Analysis

Roland Wilson*and Chang-Tsun Li†
Department of of Computer Science,
University of Warwick,
Coventry CV4 7AL, UK
†Department of Electrical Engineering, CCIT, Taiwan, ROC.

August 20, 1999

Abstract

In this paper, a new class of Random Field, defined on a multiresolution array structure, is defined. These combine earlier, tree based models with the more conventional MRF models. The fundamental statistical properties of these models are investigated and it is proved that they can avoid some of the obvious limitations of their predecessors, in terms of modelling realistic image structures. Prediction and estimation from noisy data are then considered and a new procedure: Multiresolution Maximum a Posteriori estimation, is defined. These ideas are then applied to the problem of analysing images containing a number of regions. It is shown that the model forms an excellent basis for the segmentation of such images.

Keywords: Markov random fields, image segmentation, Bayesian estimation.

---

*Corresponding Author

1 Introduction

Among the statistical approaches to image modelling, Markov Random Fields (MRF's) have been around about the longest [28], [6],[14]. Recently, however, they have gained significant attention [8] [10] [18] [22] [23] [25], especially in the segmentation of regions of more or less uniform colour or texture. For example, Geman et al. [9] used the Kolmogorov-Smirnov non-parametric measure of difference between the distributions of spatial features extracted from pairs of blocks of pixel gray levels, with MAP estimation of the boundary. Panjwani et al. [23] adopted an MRF model to characterise textured colour images in terms of spatial interaction within and between colour planes. In a technique which is similar to that used in [27], Bouman and Shapiro used sequential maximum a posteriori (SMAP) estimation in conjunction with a multi-scale random field (MSRF) [5] , which is a sequence of random fields at different scales. Other work exploring the multiresolution approach to MRF's is described in [7], [1],[21],[16] and [24]. The last two of these papers point out the difficulties in preserving the Markovian properties, which require a locality constraint, within a sampling scheme which implies the violation of that constraint. On the other hand, there is ample evidence that multiresolution processing can lead to highly efficient algorithms in many areas - from image restoration to optical flow.

The upsurge in interest in MRF's has been prompted by the work of Besag [2] and the Gemans [10, 9], largely because of the new approaches to Bayesian or Maximum a Posteriori (MAP) estimation. Although expensive computationally, these algorithms provide a relatively simple way to approach an optimal estimate using the principles of stochastic sampling, or Markov Chain Monte Carlo methods [11], as they are sometimes called. Alternatively, it is often possible to get adequate results with deterministic procedures, such as Besag's Iterated Conditional Modes (ICM) algorithm [3]. In any event, MAP estimation based on non-causal MRF's suffers from several drawbacks when applied to images. Apart from the considerable computational burden, the simpler models have 'low energy' states which represent uniform colourings, so that if an MCMC algorithm is allowed to run long enough, it will tend to produce results which reflect this, especially if the data are noisy. On the other hand, deterministic algorithms such as ICM can become trapped in local minima, also an undesirable property. While there has been a lot of work showing the efficacy of multiresolution methods, these have often been justified on heuristic grounds. The notable exceptions, [7],[1], [21], [5], tend to rely on palpably unrealistic models, such as those based on quadtrees, which are prone to the blocking artefacts associated with the sampling scheme. Attempts to avoid these problems tend to remove the link to the model, to a greater or lesser extent, unfortunately.

The work described in this paper presents a new attempt to marry the ideas of the MRF and multiple resolutions. As such, it is more closely related to the work of Bouman than the other multiresolution approaches to the problem. It also starts from a quadtree process, in which the passage from coarse to fine scales can be described as a Markov Chain. It differs fundamentally, however, in the process by which each scale or resolution is conditioned on its immediate predecessor in scale: whereas in the simple model, a pixel at a given scale depends only on its *ancestors* in scale, in the new model, it is also directly dependent on its neighbours at its own scale. In effect, this makes the model a cross between a conventional MRF, applied at a single resolution and the models proposed by Bouman and others [7], [1], [21]. This allows it to model image structures, such as multiple regions having smooth boundaries, in a much more realistic way than previous stochastic image models. The next section presents the background theory of the new model. This is followed by a description of its application to the problem of segmenting images into homogoneous regions from uncertain data. The paper is concluded with a discussion of the new model and its possible extensions.

2 Multiresolution Random Fields

The feature of a Markov Random Field which makes it attractive in applications is that the state of a given site depends explicitly only on interactions with its neighbours [10]. We model an image as a sequence of MRF's, conforming to a quadtree structure, with a nominal top level 0 and N levels below that, level k having $2^k \times 2^k$ *sites* for a finite image of size $2^N \times 2^N$ pixels. Note that we order levels from 0 at the top of the tree to N at the image level. The neighbourhood structure we impose consists of n pixels in an *isotropic* neighbourhood, such as the standard *first* and *second* order MRF models [2]:

$$\mathcal{N}_{ijk} = \{(i-l, j-m, k), l^2 + m^2 \leq R^2, l \neq m \neq 0\} \tag{1}$$

Typically, we have chosen a radius $R = 1$ or $R = \sqrt{2}$, implying the 4 or 8 neighbours commonly used in image processing [12]. In addition, we define the *parent* set, on level $k-1$, which in the simplest case consists of the so called quadtree *father*

$$\mathcal{P}_{ijk} = (\lfloor i/2 \rfloor, \lfloor j/2 \rfloor, k-1) \tag{2}$$

where $\lfloor . \rfloor$ denotes the floor of a real number. In more complex cases, we have used 8 neighbours on the same level and 4 on the 'father' level. The crucial point about any of these neighbourhood systems is that they imply *causality in scale*: in other words, the process at level k is conditioned on that at level $k-1$. This has important consequences for the properties which such fields can display. For example, it follows immediately that the field is not Markov. We shall only consider the case where pairwise interactions are involved, so that the conditional probability defining the RF can be written

$$P(X_{ijk} = m | \{X_{pqr} = n, (p,q,r) \in \mathcal{N}_{ijk} \cup \mathcal{P}_{ijk}\}) = \beta \prod_{(p,q,r) \in \mathcal{N}_{ijk} \cup \mathcal{P}_{ijk}} \rho(X_{ijk} = m | X_{pqr} = n) \tag{3}$$

where $m, 1 \leq m \leq M$ is the label at (i,j,k), β is a normalising factor and the pairwise interactions are given by

$$\rho(X_{ijk} = m | X_{pqr} = n) = e^{a_r + b_r \delta_{mn}} \tag{4}$$

where a_k, b_k are constants and δ_{mn} is the Kronecker-δ. Equivalently, the model can be expressed in terms of Gibbs potentials [15]:

$$U(\omega_k | \omega_{k-1}) = \sum_{i,j} V_{k|k-1}(\omega_{ijk} | \omega_{i/2,j/2,k-1}) + \sum_{(l,m,k) \in \mathcal{N}_{ijk}} V_{k,k}(\omega_{ijk}, \omega_{lmk}) \tag{5}$$

The model encompasses both the quadtree model, for which $b_k = 0$ and the conventional 'flat' MRF, for which $b_{k-1} = 0$. Note that the model is based on *differences* between labels: it specifies that adjacent sites are more likely to have the same label than not. Because of the product form, the conditioning of X_{ijk} depends only on the *number* of labels of different classes among its neighbours. The conventional $2-D$ discrete MRF was much studied in the context of the Ising model for magnetic spin, where it was shown that the model in $2-D$ has a significant feature, namely the occurrence of *criticality*. In the binary case, it was established that for large values of the ratio $p/(1-p)$, where p is the conditional probability $P(X_{ij} = k | X_{mn} = k, (m,n) \in \mathcal{N}_{ij})$, the model will exhibit two distinct limiting distributions, each corresponding to a predominant colouring of the lattice [15]. The interesting feature of the new model, which is not shared by a conventional MRF, is that the 'low energy' (high probability) states of the model are not in general *single coloured*: the fathers exert a force on their children which discourages uniform labellings. For example, if there exists on level k a region of constant colour, of area r^2, then the $4r^2$ pixels on level $k+1$ which represent the same region would incur a penalty of order $4r^2$ for disagreeing with their fathers, via (4); for large r, this will greatly outweigh that incurred by the approximately $4r$ boundary pixels which disagree with one or two of their neighbours on level $k+1$. This is quite different from the situation in a conventional MRF, where *any* colouring which is nonuniform incurs a cost proportional to the boundary length, so that the only way to generate significantly nonuniform labellings is to 'increase the temperature', with the result that the typical sample will have a random textured appearance. Although with the above asymmetrical neighbourhoods, the field is not Markovian, if we consider the state of, say, level k of the field conditioned upon that at level $k-1$

$$P(X_k = \omega_k | X_{k-1} = \omega_{k-1}) = \frac{1}{Z_k} e^{-U(\omega_k|\omega_{k-1})} \tag{6}$$

where ω_i is the configuration of the whole image at level i and Z_k is the so-called *partition function*, then the Markovian/Gibbsian property is restored. In effect, the configuration at level $k-1$ acts as an 'external field' for the lattice at level k. Unlike a typical such field, this one varies at half the sampling rate of the image, imposing the coarse structure on its children at the next level, which adds details, especially in the vicinity of the boundaries. This is demonstrated in the following theorem:

Proposition 1 *Let $P(\omega_k|\omega_{k-1})$ define a conditional Markov Random Field through the potential function in (5) above, where the indexes $i/2, j/2$ are taken as integers and the potentials are, without loss of generality, zero if the two states are identical and satisfy*

$$V_{k|k-1}(n|m) \geq |\mathcal{N}_{ijk}| \max_{n \neq m} V_{k,k}(m,n) > 0, \; m \neq n \tag{7}$$

Then the configuration ω_k^* which maximises $P(\omega_k|\omega_{k-1})$, for a given configuration ω_{k-1}, is given by

$$\omega_{ijk}^* = \omega_{i/2,j/2,k-1} \tag{8}$$

In other words, the most likely configuration on level k, ω_k^*, is just a copy of that on the level above.

Proof: To prove the assertion, note that each site $(i, j, k-1)$ on level $k-1$ has four children on level k, each of which has the same state in configuration ω_k^*. Similarly, each neighbour of each of these sites will have a state equal to that of its father on level $k-1$. Suppose that all of these sites have the same state, m, say. Then any change to another state at $(2i+q, 2j+r, k)$, $q, r \in [0,1]$ must increase the potential at that site, since it will introduce a disagreement between itself and its father and neighbours. It follows that only sites on the boundary between two regions with different states need be considered. Any such site $(i, j, k-1)$ can have at most $|\mathcal{N}_{i,j,k-1}|$ neighbours on level $k-1$ in a different class. Similarly, each child of such a node can have at most $|\mathcal{N}_{i,j,k}| - 2$ neighbours in a different class, since it will have at least 2 siblings which are also in class m. Hence the maximum reduction in potential from changing such a site occurs when all the neighbours have the same state $n \neq m$ and all 4 children are changed to n from m:

$$\triangle U = 4V_{k|k-1}(n|m) - \sum_{(s,t,k) \in \mathcal{N}_{ijk}^s} V_{k,k}(n,m) \tag{9}$$

where $\mathcal{N}_{ijk}^s$ is the set of neighbours of (i, j, k) or any of its 3 siblings. But for any isotropic neighbourhood structure on level k

$$|\mathcal{N}_{ijk}^s| < 4|\mathcal{N}_{ijk}| - 4 \tag{10}$$

Hence if the potentials satisfy (7), then

$$\triangle U > 4V_{k,k}(n,m) > 0 \tag{11}$$

So the change in potential is positive and the state is less likely. Since this is the worst case, in terms of change in potential, it follows that the state ω_k^* is indeed the most likely one, given ω_{k-1}. $\diamond$

Although the result is a simple one, its significance should not be overlooked: it shows that the structure obtained at coarse scales will be preserved as details are added. Without such structure preservation, there would indeed be little point in having the multiresolution model.

With a tighter constraint on the father-child potential, we can establish a stronger result, which is summarised in the following:

Theorem 1 *Let $P(\omega_k|\omega_{k-1})$ define a conditional Markov Random Field through the potential function in (5) above, where the indexes $i/2, j/2$ are taken as integers and the potentials are, without loss of generality, zero if the two states are identical and satisfy*

$$V_{k|k-1} - 2|\mathcal{N}_{ijk}|V_{k,k} > 2k \log \frac{2}{\epsilon} \tag{12}$$

Then the copy configuration ω_k^* defined in (8) has a probability

$$P(\omega_k^*|\omega_{k-1}) > 1 - \epsilon \tag{13}$$

Proof: To prove the result, it suffices to note that *any* configuration differing from ω_k^* has a probability bounded below by that of a Bernouilli process, whose probabilities are defined by the potentials. This process is defined as follows:

$$\lambda_{ijk} = \begin{matrix} 0 & \text{if } \omega_{ijk} = \omega_{i/2,j/2,k-1} \\ 1 & \text{else} \end{matrix} \tag{14}$$

Now the associated potentials are taken from the original process:

$$V^{\lambda}_{k|k-1} = V_{k|k-1}, \quad V^{\lambda}_{k,k} = V_{k,k}. \tag{15}$$

With each configuration ω_k is a corresponding 'error' configuration λ_k. Each 0 in λ_k has a maximum potential $R\ V_{k,k}$, where $R = |\mathcal{N}_{ijk}|$ is the neighbourhood size; this occurs if all of its neighbours are different, while each 1 carries a cost no less than $V_{k|k-1} - |\mathcal{N}_{ijk}|V_{k|k}$, since it must at the very least disagree with its father. It follows that the probabiltity of any configuration having m 0's is bounded below by

$$P_k(m) \geq \frac{1}{Z}\ e^{-mRV_{k,k} - (2^{2k} - m)(V_{k|k-1} - RV_{k,k})} \tag{16}$$

which can be expressed as

$$P_k(m) \geq p^m\ q^{2^{2k} - m} \tag{17}$$

where

$$p = \frac{e^{V_{k|k-1} - 2RV_{k,k}}}{1 + e^{V_{k|k-1} - 2RV_{k,k}}} \tag{18}$$

Hence the probability of the configuration with 0 errors is just

$$P_k(0) \geq \frac{1}{(1 + e^{2RV_{k,k} - V_{k|k-1}})^{2^{2k}}} \tag{19}$$

so, if

$$2^{2k}e^{2RV_{k,k}-V_{k|k-1}} > \epsilon \tag{20}$$

then $P_k(0) > 1-\epsilon$, from which the result follows immediately.$\diamond$

It is perhaps obvious from the construction that this is a weak bound, since the vast majority of error configurations actually increase the boundary length. That has little impact, however, on the general conclusion that the *spread* of the distribution around ω_k^* can be tightly controlled by the choice of interaction potentials.

To illustrate how the father level affects the process, figure 3(a) shows an array of 4×4 sample images generated from the same 16×16 father image using different combinations of father-child and neighbour interactions. Each 32×32 image within this array is the result of 2000 iterations, more than sufficient for an array of this size to approach the stationary distribution. Note that the image at the top left shows complete randomness, as both interactions are 0 in this case, while moving from top to bottom the neighbour interaction strength increases and from left to right the father-child interaction increases. This shows rather clearly the effect of the father over a wide range of interaction strengths: there is a doubling of the interaction potential for each step to the right or downwards. Note that in this and the other sample images, the image boundary pixels are fixed as 'black' (0), so that, in the absence of father-child interactions, for the higher levels of neighbour interaction, the stationary distribution will have a maximum when all pixels are black. It can also be seen that all of the images in the top row have random, isolated pixels. This is because, in the absence of neighbour interactions, the process is Bernouilli, with each pixel's colour being chosen independently; as we move right along the row, however, the probability of its colour being different from its father's decreases geometrically.

Although the $4-$*neighbour* field is simple, it is rather limited in its ability to produce smooth boundaries and so we also examined the use of the $8-$*neighbourhood*. Using the $8-$*neighbours* as conditioning elements produces the result shown in figure 3(b). In comparison with 3(a), the regions are noticeably smoother, with the gaps in the nonconvex shape being filled, but the same general trends can be observed as the interaction strengths increase. For example, with greater potentials to the father level, small regions tend to survive better. This is evident in the images on the right side of the array. Another obvious defect of the above models is that boundaries tend to align with the quadtree, which introduces a 'blockiness' into both the statistical structure and the sample images. A simple way around this problem is to make the influence of the father level $k-1$ zero for any site at a boundary (ie. having a neighbour of a different colour). This is simply accomplished by setting the father-child potential to 0 whenever the father has a neighbour in a different class

$$V_{ijk|i/2,j/2,k-1}(n|m) = 0, \text{ if } \omega_{i/2,j/2,k-1} \neq \omega_{lmk-1}, \text{for some}(l,m,k-1) \in \mathcal{N}_{i/2,j/2,k-1} \tag{21}$$

This is equivalent to extending the set of neighbours of (i,j,k) on level $k-1$ to a larger set of pixels. As a demonstration of the effect of the two changes, it is simple to show that the most likely configuration on level k, ω_k^*, tends to remove corners propagated from level $k-1$:

Proposition 2 *Let ω_{k-1} be a configuration of level $k-1$ of a MMRF and let ω_k^* be the corresponding most likely configuration on level k, based on pairwise potentials using $8-neighbours$ and such that fathers on boundaries have interaction potential 0 with their children. Let $(i,j,k-1)$ be a pixel on level $k-1$, such that*

$$\omega_{ijk-1} \neq m, \; \omega_{pqk-1} = m; \;\; (p,q) \in \mathcal{N}^2_{ijk-1} \tag{22}$$

where $m \in [0,1]$ and the neighbourhood $\mathcal{N}^2_{ijk-1} = \{(i+d,j,k-1),(i,j+e,k-1),(i+d,j+e,k-1),\; d = 1 or \;\; d = -1$ and $e = 1 or \;\; e = -1$, are neighbours of $(i,j,k-1)$ within a block of size 2×2 pixels. Suppose that the configuration ω_k^* has the property that

$$\omega^*_{2p+r,2q+s,k} = m, \;\; r,s \in [0,1], \;\; (p,q,k-1) \in \mathcal{N}^2_{ijk-1} \tag{23}$$

ie. the children of the 3 neighbours have the same class as their fathers. Then

$$\omega^*_{2i+t,2j+u,k} = m, \; t = \frac{1+d}{2}, \; u = \frac{1+e}{2} \tag{24}$$

In other words, the 'corner' pixel $(2i+t,2j+u,k)$ is more likely to change its label than to retain its father's class. The effect is illustrated in figure 1

Proof: First, observe that since each of the pixels within the 2×2 block on level $k-1$ has a neighbour in a different class, the interaction potentials with their children are 0. It follows that only the interactions on level k need be considered.

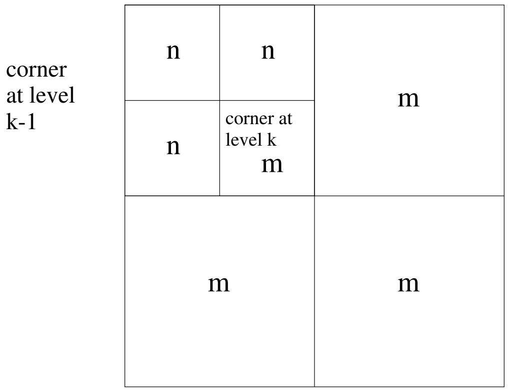


Figure 1: The 'corner effect' in an 8-neighbour MMRF with boundary effect. The corner pixel at level $k-1$ is refined at level k to reduce the boundary energy.

Next consider the pixel $(2i+t, 2j+u, k)$ on level k. Of its 8 neighbours, at most 3 have a label other than m in configuration ω_k^*. It follows that the minimum energy label for $(2i+t, 2j+u, k)$ must be $m.\diamond$

A noteworthy consequence of this is that the approximation of a straight edge does become less 'jaggy' as the resolution increases. As an illustration of the effect of these modifications, figure 4 shows a sample from a binary process using the 8-neighbours on levels $5 \leq k \leq 8$, but a quadtree on levels $0 \leq k \leq 4$. By choosing the model parameters appropriately as functions of the level, different combinations of structure can be obtained. In this case, the coarse structure representing the bottom level of the pure quadtree process is *refined* by the $8 - neighbour$ MMRF, resulting in a single, smooth 'blob' representing the object. Similar results were obtained after $10,000$ iterations, indicating that 4 is in equilibrium. A second example, using a lower correlation coefficient between neighbours, is shown in figure 6. Note that the large scale structure in this case is punctuated by some smaller 'objects'. In both cases, however, the parameters of the field are super-critical and the boundary sites at each level are fixed at 0, so the most likely image is black. The sample autocorrelation from a set of 20 images produced in a similar way is shown in figure 7; not surprisingly, this reflects the large scale structure of these images.

Condition	Level 8	Level 7	Level 6	Level 5
9 Neighbours	0.0093	0.0101	0.0178	0.0898
Father	0.0092	0.0104	0.0171	0.0303

Table 1: Prediction error rates from level $k-1$ to k using all 9 neighbours or copying the father level, for figure 4.

2.1 Prediction and Estimation

The above observations suggest that a very high level of compression could be achieved in representing a sample from such a process. In effect, the configuration ω_k^*, which depends only on the level above, is an excellent predictor for ω_k. This is illustrated by the difference pyramid in figure 5, which shows the absolute error

$$\epsilon_k = ||\omega_k - \omega_k^*|| \tag{25}$$

Indeed, it follows from proposition 1 above that, under the conditions of the proposition, ω_k^* is the Maximum Likelihood predicition of level k from level $k-1$. In more general cases, however, sampling will be necessary to find the best predictor

$$\hat{\omega}_{k|k-1} = \arg\max_{\omega_k} P(\omega_k|\omega_{k-1}) \tag{26}$$

Figure 8 shows the result of simulated annealing over 200 iterations, with a logarithmic schedule, to predict each level of the $8-\textit{neighbour}$ pyramid of figure 4 from its father. While the result is visually quite convincing, the table of error rates 1 shows that there is no improvement over the 'copy' from the father.

Apart from its importance in communications applications, the question of predictability also relates directly to the Gibbs distribution, which is well known to maximise entropy for a given expected energy [15]. The appropriate entropy measure is the entropy conditioned on the neighbours

$$H_k(P) = \sum_{\omega_{ijk},\omega_{lmn},(l,m,n)\in\mathcal{N}_{ijk}\cup\mathcal{P}_{ijk}} P(\omega_{ijk},\omega_{lmn}) \ \log_2 \frac{1}{P(\omega_{ijk}|\omega_{lmn})} \tag{27}$$

Table 2 shows the sample entropies from the bottom 5 levels of the pyramid in 4, conditioned both on the 8 neighbours and on only the father. Although all the entropies are small, representing the very high degree of predictability in this process, there is a significant benefit in using the context on the same

Condition	Level 8	Level 7	Level 6	Level 5	Level 4
8 Neighbours	0.0026	0.0044	0.0037	0.0080	0.0293
Father	0.0758	0.0816	0.1206	0.1967	0.3005

Table 2: Conditional entropies in bits/pixel for various levels of the $8-$ *neighbour* MMRF.

level, over the prediction from the father. More significantly, it should be noted that the entropy per pixel seems to be headed for 0 as the resolution increases. The reason for this is clear from the prediction error images in figure 5: the errors are confined to a narrow region around the edge of the region. Since the edge is tending to a smooth shape, it will be of length $O(2^N)$, as N increases, resulting in an entropy which tends to zero

$$\lim_{k \to \infty} H_k(P) = 0 \tag{28}$$

The other main application of the model is in estimation from noisy data. At its simplest, we might consider the problem of estimating the image at one level, X_k, say, from noisy data Y_k. If we know the image X_{k-1}, we can use the Conditional Maximum A Posteriori (CMAP) estimator

$$\hat{X}_k = \arg \max_{X_k} P(X_k | Y_k, X_{k-1}) \tag{29}$$

where, from (6) above,

$$P(X_k | Y_k, X_{k-1}) = \frac{P(Y_k | X_k) P(X_k | X_{k-1})}{P(Y_k | X_{k-1})}, \tag{30}$$

the first term on the right being the likelihood. While the CMAP estimate is simple, there are few practical applications where the images $X_k, 0 \leq k \leq N$ are available. The obvious alternative is the unconditioned MAP estimate of all the levels

$$\{\hat{X}_i, i \leq k\} = \arg \max_{\{X_i, i \leq k\}} P(X_k, X_{k-1}, ..., X_0 | Y_k) \tag{31}$$

and so on for $X_i, i < k$, which can be expressed via (6) as

$$P(X_k, X_{k-1}, ..., X_0 | Y_k) = \frac{P(Y_k | X_k) P(X_0)}{P(Y_k)} \prod_{i=1}^{k} P(X_i | X_{i-1}) \tag{32}$$

which avoids assumptions about X_{k-1}, but poses another problem: how to select $X_i, i < k$ to simultaneously maximise the posterior with respect to all $k+1$ images. The obvious weakness of this approach is that the data Y_k constrain $X_i, i < k$ quite strongly and this ought to be built into the estimation procedure. This is the goal of the multiresolution MAP (MMAP) estimator. For the above problem, we start again from the left side of (32), but now expand as

$$P(X_k, X_{k-1}, ..., X_m | Y_k) = P(X_m | Y_k) \prod_{i=m+1}^{k} P(X_i | X_{i-1}, Y_k) \qquad (33)$$

where we have used the Markov property of the sequence X_i and we may start the estimation on a level other than 0. The interesting feature of this estimator is that it has a sequential structure: first estimate X_m, then X_{m+1} and so on, up to X_k, using

$$P(X_i | X_{i-1}, Y_k) = \frac{P(Y_k | X_i) P(X_i | X_{i-1})}{P(Y_k | X_{i-1})} \qquad (34)$$

The denominator, for fixed X_{i-1}, is a constant, but there is a gap between the data Y_k and X_i. A short cut to the solution is to use the copy configuration $X_{k|i}$, which for parameters which are supercritical and satisfy the conditions of Theorem 1 represents a good predictor for X_k. This allows sampling to be performed on X_i. In the binary case, an equivalent procedure is to define data $Y_i, i < k$, by simply averaging over the block of $2^{k-i} \times 2^{k-i}$ pixels on level k corresponding to each pixel (p, q, i) on level i. It is not hard to see that in this case, we can replace (33) by

$$P(X_k, X_{k-1}, ..., X_m | Y_k, Y_{k-1}, ..., Y_m) = P(X_m | Y_m) \prod_{i=m+1}^{k} P(X_i | X_{i-1}, Y_i) \qquad (35)$$

As a simple example, consider figure 9, which shows the pyramid obtained from the image at the bottom level of figure 4, corrupted by additive white Gaussian noise of unit variance. The pyramid, which represents the 'raw' data for the estimator, was obtained using simple block averaging, as in a quadtree: each pixel on level k is the average of its 4 children on level $k+1$. The estimation problem is to reconstruct the original binary pyramid from these data. The data are conditionally normal

$$p(Y_{ijk} | l) = N(l, v_k), \quad l \in [0, 1] \qquad (36)$$

where the variance on level k, $v_k = v_{k+1}/4$ and $v_8 = 1$. The estimator we use is an extension of the stochastic (Gibbs Sampling) methods described in [10], which takes account of the conditioning of level k by its father $k-1$. Thus the MMAP estimate is defined by

$$\hat{X}_{ijk} = \arg\max_m P(X_{ijk} = m | Y_{ijk}, \hat{X}_{pqr}, (p,q,r) \in \mathcal{N}_{ijk} \cup \mathcal{P}_{ijk}) \qquad (37)$$

where Y_{lmn} are the noisy data. There are two distinct procedures for implementing the estimator, which we have dubbed MMAP and *sequential* (SMMAP) in the sequel. The MMAP method visits *each* level k on each iteration, thus sampling different scales 'simultaneously'. The sequential method iterates on one level at a time, with level k only being estimated after level $k-1$ converges, making it analogous to the conditional MAP procedure.

The estimates were initialised by thresholding level 4 in the data pyramid and thereafter using conditioned Gibbs sampling. After 100 iterations at each level, the result in figure 10 was obtained. The error rates at the various levels for each estimator are shown in Table 3 below. Although only few iterations were used, the results at high resolutions are significantly better than were obtained by simply copying the initial level or using a conventional $8-$*neighbour* MRF estimator. Of the MMAP estimators, the MMAP algorithm performed better than either CMAP or SMMAP. Closer examination shows that the MMRF estimate gives a more or less constant error rate of 50% per boundary pixel, across a range of scales. This is because the data are uncertain in these areas - averaging across the boundary does not improve this. Figure 11 shows the number of sites changing on each iteration, for each of the 4 levels. It shows that after an initial burst at each level, occupying a few iterations, the sampler settles down to a steady state where only a handful of sites change on any iteration. The next image 12 shows that even with a random initial configuration, the sampler quickly converges to a point where only a few sites change on each iteration. Note that one iteration here refers to a scan-order visit to every pixel on a given level.

We conclude this example by making the following observations:

1. Both MMAP estimators perform well on this problem - better than those based on simple MRF or low-resolution thresholding.

2. The relatively worse performance of the SMMAP algorithm shows the effect of constraining the estimate at level k by a single realisation at level $k-1$, rather than sampling over the whole space simultaneously.

Estimator	Level 8	Level 7	Level 6	Level 5	Level 4
CMAP	0.0044	0.0084	0.0164	0.0332	0.0547
MMAP	0.0036	0.0074	0.0149	0.0332	0.0547
SMMAP	0.0047	0.0083	0.0156	0.0292	0.0547
8-neighbour MRF	0.0126	0.0094	0.0171	0.0332	0.0508
Copy from lev. 3	0.0155	0.0128	0.0178	0.0302	0.0547

Table 3: Error rates in MMAP estimates at various levels, compared with rate from 8-neighbour MRF and from thresholding at level 4 and copying.

3. Both are fast, requiring of the order of 20 iterations to obtain satisfactory estimates.

4. However, since the cost of an iteration at high resolution far outweighs that at low resolution, there is a computational advantage to the SMMAP approach because it allows a tailoring of the annealing schedule to each level separately.

5. A further advantage of SMMAP is that the estimate at level k can be used to initialise model parameter estimation at level $k+1$, eg. using the sampling method described in [17].

A more realistic case is the image shown in figure 13, which again is a binary image with added white Gaussian noise at a standard deviation of 1, ie. equal to the difference between black and white. The estimation error at the highest resolution, using the MMAP algorithm in this case was 1.3%, better than most results reported on comparable problems in the literature [5],[17],[27]. The resulting estimate is shown in figure 14; apart from the corners, where the model does not fit the data, the estimate is visually quite good.

In more general cases, the measurements are not all the result of averaging noisy binary image data, of the form of fig. 9. Instead, let the data be given by the pyramid $\{Y_i, m \leq i \leq k\}$, where

$$P(Y_k, Y_{k-1}, ..., Y_m | X_k, X_{k-1}, ..., X_m) = \prod_{i=m}^{k} P(Y_i | X_i) \tag{38}$$

In other words, the data on level i are the result of applying an independent noise process to the image at that level. We wish to preserve the sequential structure in developing a solution to the MAP problem

$$\{\hat{X}_i, m \leq i \leq k\} = \arg \max_{\{X_i, m \leq i \leq k\}} P(X_k, X_{k-1}, ..., X_m | Y_k, Y_{k-1}, ..., Y_m) \quad (39)$$

Now the posterior in this case is easily obtained with the help of (6) and (38)

$$P(X_k, X_{k-1} ..., X_m | Y_k, Y_{k-1}, ..., Y_m) = \frac{P(X_m) P(Y_m | X_m) \prod_{i=m+1}^{k} P(X_i | X_{i-1}) P(Y_i | X_i)}{P(Y_k, Y_{k-1}, ..., Y_m)} \quad (40)$$

where the denominator is a constant. This has significant implications for how we may obtain a MAP estimate: it leads us directly to the MMAP and SMMAP algorithms described above. Initialisation at level m is readily done if we assume that the father-child potential $V_{m|m-1} = 0$; alternatively, ML estimation can be used at that level (as in the binary example of fig. 9). In the MMAP estimate, (40) can be used to sample simultaneously from the posterior distribution, while in SMMAP, sampling at level k only starts when that on level $k-1$ terminates. This implies that, while SMMAP may give an excellent approximation to the MAP estimate, it is not MAP, but in this case, MAP=MMAP.

2.2 Hidden Models

While there are few segmentation tasks in which this discrete model directly reflects image intensity or colour, it is very useful as a *hidden* model: the state of the site (i, j, k) controls the parameters of a local image model defining the characteristics within the region of $2^k \times 2^k$ pixels associated with that site. In that case, there will be a *measurement vector* $\boldsymbol{Y}_{ijk}$ associated with the site, which depends on the label, ie

$$p(\boldsymbol{Y}_{ijk} | X_{ijk} = m) \neq p(\boldsymbol{Y}_{ijk} | X_{ijk} = n), \text{ if } m \neq n \quad (41)$$

The measurement vector might represent a histogram of intensity or colour or some suitable texture measure, for example. The *Maximum Likelihood* (ML) estimator for the label is then

$$\hat{X}_{ijk} = \arg \max_m \ p(\boldsymbol{Y}_{ijk} | X_{ijk} = m) \quad (42)$$

which is simple, but ignores the prior probability. The MMAP estimator in this case is similar in spirit to the SMAP estimator of [5], but differs in one important respect: sampling is used to obtain the estimate at each level. As in SMAP, we compute the estimates sequentially over scale, starting at some coarse scale k_{min}, for which we use the conventional MAP estimate, obtained by a simulated annealing process

$$\hat{X}_{ijk} = \arg\max_m \; P(m|\boldsymbol{Y}_{ijk}, \{\boldsymbol{Y}_{pqk}, \hat{X}_{pqk}, (p,q,k) \in \mathcal{N}_{ijk}\}) \tag{43}$$

In effect, we are assuming independence of level k_{min} from level $k_{min}-1$. At subsequent levels, the labelling $\hat{X}_{ijk}$ is used to *condition* that at level $k+1$:

$$\hat{X}_{ijk} = \arg\max_m \; P(m|\boldsymbol{Y}_{ijk}, \{\boldsymbol{Y}_{pqr}, \hat{X}_{pqr}, (p,q,r) \in \mathcal{N}_{ijk} \cup \mathcal{P}_{ijk}\}) \tag{44}$$

This gives the estimation a *causal* direction through scale, whilst using the non-causal, iterative process of annealing at each scale. Moreover, the 'copy' configuration is used as the initial labelling at level k.

In many practical applications, using pairwise potentials and difference measurements, we end up with a normal model for the likelihoods, of the general form

$$p(\boldsymbol{Y}_{ijk}-\boldsymbol{Y}_{pqr}|\omega_k, \omega_{k-1}) = N(\mu(\omega_{ijk}, \omega_{pqr}), \Sigma(\omega_{ijk}, \omega_{pqr})), \quad (p,q,r) \in \mathcal{N}_{ijk} \cup \mathcal{P}_{ijk} \tag{45}$$

where the normal mean and covariance parameters depend only on the classes at the two sites. These parameters can be estimated on-line, given the current classification at level k. This illustrates another advantage of using the multiresolution approach: although the equilibrium distribution will in principle be approached from *any* initial configuration, in practice, it will happen sooner if the initial configuration is close to equilibrium. As with the prior, the posterior distribution of ω_k will be a Gibbs distribution conditioned on the configuration on level $k-1$ and so sampling methods can be used to locate the maximum.

2.3 Application to Texture Segmentation

In its application to texture segmentation, the model is hidden, with each site on level k representing a square region of nominal size $2^{N-k} \times 2^{N-k}$ pixels, from which texture measurements are taken, as in [9]. In fact, windows with a 50% overlap are used to reduce estimation artefacts. It is convenient to specify the model in terms of the Gibbs potentials. The interaction potential defining the MRF at level k in the tree is based on pairwise interactions:

$$V_{ijk}(m|n) = a + b\|\boldsymbol{Y}_{ijk} - \boldsymbol{Y}_{pqr}\|^2 \delta_{mn}, \quad (p,q,r) \in \mathcal{N}_{ijk} \cup \mathcal{P}_{ijk} \tag{46}$$

where $\|.\|$ is a suitably chosen norm, such as the Euclidean norm. In other words, there is a cost based on feature similarity associated with sites in the same class. Sampling is then based on the corresponding Gibbs distribution

$$P(m) \propto e^{-\frac{U(m)}{T}} \tag{47}$$

where the position indices have been suppressed and T is the scale parameter, or *temperature*, which is varied using a logarithmic annealing schedule [10] From these definitions, the SMMAP algorithm becomes:
For level $k \geq k_{min}, k \leq N$

1. Sample at every site on level k using measurements $\boldsymbol{Y}_{ijk}$ and a logarithmic annealing schedule, until no change is detected over a number I_k of iterations over the image at that scale.

2. Use labels on level k as the initial labelling on level $k+1$, by copying labels from fathers to children in the quadtree and to condition the simulation on level $k+1$.

The initial labelling at level k_{min} is random.

While the above algorithm provides a general framework for segmentation, its effectiveness depends critically on the texture descriptors used. We have four local measurements, which are based on the 'deterministic+stochastic' decomposition, which is a generalisation of the Wold decomposition of signals [13]. The four components are:

1. The difference between the average gray level in the blocks.

2. Two measures associated with the deterministic component, based on an affine deformation model

$$f_s(\vec{\xi}) = f_{s'}(\boldsymbol{A}^{-1}(\vec{\xi} - \vec{\chi})) + \nu_s(\vec{\xi}) \tag{48}$$

where $f_s(.)$ represents the patch of an image centred at site s, site $s' = (l, m, k)$ is a 4-neighbour of site $s = (i, j, k)$, $\boldsymbol{A}$ is that 2×2 nonsingular linear co-ordinate transform and $\vec{\chi}$ that translation which together give the best fit in terms of total deformation energy between the two patches. These are identified using the method described in [13], which makes use of local Fourier spectra calculated at the appropriate scale using the Multiresolution Fourier Transform (MFT) [26]. The deformation energy consists of:

(a) The deformation term $||\boldsymbol{A}-\boldsymbol{I}||^2$ represents the amount of 'warping' required to match the given patch using its neighbour.

(b) The error term $||\nu_s(\vec{\xi})||^2$ is the average residual error in the approximation.

3. A measure for the stochastic component, based on differences in the spectral energy densities estimated at each site via the MFT, $|\hat{f}(\vec{\xi}, \vec{\omega}, \sigma)|^2$, where

$$\hat{f}(\vec{\xi}, \vec{\omega}, \sigma) = \frac{1}{\sqrt{\sigma}} \int d\vec{x}\ f(\vec{x}) w(\frac{\vec{x} - \vec{\xi}}{\sigma})\ e^{-\jmath \omega \vec{m}.\vec{x}} \tag{49}$$

is the (continuous) MFT at spatial co-ordinate $\vec{\xi}$, frequency $\vec{\omega}$ and scale σ [26], which is approximated by a sampled version in practice. This is similar to many texture classification methods based on local spectra, Gabor filters or autocovariance estimates [27].

Each of these measures is scaled by the corresponding (within-class or between-class) sample variance and the four are added with appropriately chosen weights to give the final interaction energy. Only the gray level difference is used for the father interaction, however.

The neighbour conditional probabilities are estimated directly from the data during the sampling process, as are the within-class and between-class variances. At levels $k > k_{min}$, the priors take into account the classification on the previous level, $k-1$: the prior probability that a child has the same class as its father is approximated by

$$P(X_{ijk} = X_{i/2,j/2,k-1}) = 1 - \rho^{d_{i/2,j/2,k-1}}, \tag{50}$$

where $(i/2, j/2/k-1)$ is the father site, $\rho < 1$ is a constant and d_s is the shortest distance between site s and a site having a different class, ie. it represents distance to the boundary. In the experiments reported below, $\rho = 0.5$, implying that fathers have no effect at the boundary, which ensures that boundaries are not biased by the quadtree.

In addition , a line process has been introduced to increase the accuracy of the segmentations using an assumption of smoothness of the boundary, since texture measurements require a minimum sample size, which we have found in practice to correspond to a sampling interval of 4×4 pixels with the above texture measures. The line process is also based on pairwise interactions between neighbouring boundary blocks, based on the oriented line joining the estimated positions of the putative boundary in each block. Boundary processing is also a simulation designed to find the Bayesian estimate, but occurs after the regions have been identified on a given level. Only region sites having neighbours which belong to a different class are identified as potentially boundary-containing and the process is run on those alone. From these sites, a subset is selected by stochastic labelling, using a potential function which penalises curvature in the line joining the estimated centroids of the putative boundary segment in each block. The potential has the form

$$V(\boldsymbol{Y}, \boldsymbol{Z}) = (\sin Y_3 + \sin Z_3) \sum_{i=1}^{2} (Y_i - Z_i)^2 \tag{51}$$

where the first two vector components represent the centroid position (X_1, X_2) and the third component is the angular difference between the boundary angle at (X_1, X_2) and the line joining the two centroids, as illustrated in figure 2. The centroid position and boundary angle at a site are estimated using the MFT-based technique first described in [26]. In this way, both texture and boundary features can be computed within the same framework. Full details can be found in [19]. A summary of the boundary labelling algorithm follows:

At each temperature T:

1. For each site $i \in \mathcal{B}$
2. Calculate the potential $V(\boldsymbol{Y}_i, \boldsymbol{Y}_j),\ j \in \mathcal{N}_{B,i}$
3. Sample from the Gibbs distribution to determine the label γ_i

A logarithmic annealing schedule is again followed for the boundary processing, which runs after the region processing is complete at a given level. In the present scheme, no information is propagated from 'boundary fathers' to their children and sites in the boundary set $\mathcal{B}$ are labelled as either B or $\bar{B}$. This is a significantly different model from the classic line schemes based on pixel labelling (eg. [10], as it is designed to fulfil a different role.

The experiments we have used to test the model demonstrate its ability to segment textured images of various types, as can be seen from figures 15 and 16. In figure 15, the refinement of the segmentation through the MMAP procedure is evident, as is the improvement due to the boundary process. Table 4 summarises the performance of the algorithm on this data. The

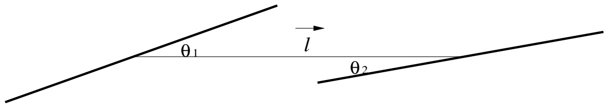


Figure 2: 'Distance' between boundary segments

Table 4: Segmentation error rates and number of iterations per pixel (# i/p) for image of fig. 15.

level k	**C**	**Region Process**		**Boundary Process**	
		Error rate (%)	**# i/p**	**Error rate (%)**	**# i/p**
3	8	7.053	0.189	3.079	0.018
4	6	1.640	0.074	1.059	0.009
5	4	1.265	0.349	0.485	0.016
6	3	0.716	2.191	0.365	0.045
Total # i/p			2.803		0.088

error rate drops to less than 1% with the boundary process at the highest resolution and this is achieved at a normalised number of iterations per pixel of only 2. This figure is the sum of contributions from the various levels, each weighted by the number of pixels on that level. Note that the algorithm terminates 2 levels above the image level because this is the highest resolution for which we can obtain meaningful texture and boundary estimates.

In the second figure, a summary of the high resolution segmentations is shown, for several combinations of two or more textures. It should be noted that no additional information on the number of textured regions is required by the algorithm - it is completely unsupervised. These pictures illustrate the effectiveness of the overall technique and the utility of the boundary process, which both improves the subjective quality and lowers the misclassification rates to be among the best reported in the literature - typically of the order of $1-2\%$. The test images were 256×256 pixels, with the textures taken from Brodatz's book. Because of the multiresolution estimation, the overall number of iterations required to attain convergence was low - in the examples shown in figure 16, the number of iterations/pixel was of the order of 4.

We have compared these results with those presented by a number of authors, including [16] [4],[5], [17],[27] and [20]. The results presented here are superior in terms of error rates to those and compare well with any we have seen in the literature on image segmentation.

3 Conclusions

In this paper, we have presented a new model for image analysis, which combines the notions of multiple resolutions and MRF's to provide a powerful way of describing image structure statistically. Correspondingly, a new form of MAP estimator - the Multiresolution MAP estimator - was presented. The model was illustrated with examples of image segmentation, in which it has been shown to be among the most effective methods yet described for the task. The advantages of the new model may be summarised as:

1. By conditioning the MRF at level k by that on level $k-1$, uniform labellings are no longer the 'ground' state of the model. This avoids one of the most obvious weaknesses of conventional MRF models. By using an appropriate neighbourhood and conditioning the father-child interactions on the presence of boundaries, it is possible to trade off boundary smoothness against the degree of structure preservation. This is a completely new feature of the model, which it does not share with previous image models.

2. The *final* state at level k, as well as conditioning the MRF at level $k+1$, can be used as an *initial state* at level $k+1$, simply by copying labels from fathers to their four quadtree children. Although the final MAP estimate should be independent of the initial state, the time taken to get there is affected by the initialisation. Using the labelling in this way speeds up computation.

3. By appropriately combining the spatially invariant MRF structure with the quadtree, the blocking and non-stationarity artefacts of that model are greatly reduced.

4. The model parameters, which generally are unknown, can be estimated for the higher resolutions by using the segmentation obtained at the coarser scales.

5. Again because of the conditioning by coarse scales, the results are not critically dependent on the number of labels M.

Although the work reported here is encouraging, much remains to be done before it can be considered complete. For example, the line process which was used to improve the estimate of the boundary, does not interact with the region labelling. Similarly, the segmentation model has not been tested with other image features. Work is currently under way to address these issues.

Acknowledgments

The authors wish to thank Chung-Cheng Institute of Technology in Taiwan for supporting this study.

References

[1] M. Basseville, A. Benveniste, K.C. Chou, S.A. Golden, R. Nikoukah, and A.S. Willsky. Modelling and estimation of multiresolution stochastic processes. *IEEE Transactions on Information Theory*, 38(2):766–784, 1992.

[2] J. Besag. Spatial Interaction and the statistical Analysis of Lattice Systems. *Journal of the Royal Statistical Society (Series B)*, 36:192–236, 1974.

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

[4] C. A. Bouman and B. Liu. Multiple Resolution Segmentation of Textured Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(2):99–113, 1991.

[5] C. A. Bouman and M. Shapiro. A Multiscale Random Field Model for Bayesian Image Segmentation. *IEEE Transactions on Image Processing*, 3:162–176, 1994.

[6] R. Chellappa and R. L. Kashyap. Texture Synthesis using 2-D noncausal autoregressive models. *IEEE Transactions on Acoustics, Speech, Signal Processing*, 33:194–203, 1985.

[7] S. Clippingdale. *Multiresolution Image Modelling and Estimation*. PhD thesis, Department of Computer Science, The University of Warwick, UK, September 1988.

[8] F. S. Cohen and Z. Fan. Maximum Likelihood Unsupervised Textured Image Segmentation. *Computer Vision, Graphics, and Image Processing*, 54:239–251, 1992.

[9] D. Geman, C. Graffigne S. Geman, and P Dong. Boundary Detection by Constrained Optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:609–628, 1990.

[10] S. Geman and D. Geman. Stochastic relaxation, Gibbs distribution, and Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

[11] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter. *Markov Chain Monte Carlo in Practice*. Chapman and Hall, 1996.

[12] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Addison Wesley, 1992.

[13] T. I. Hsu and R. G. Wilson. A Two-component Model of Texture for Analysis and Synthesis. *IEEE Transactions on Image Processing*, 7(10):1466–1476, 1998.

[14] A. K. Jain. *Fundamentals of Digital Image Processing*. Prentice Hall, 1989.

[15] R. Kinderman and J. L. Snell. *Markov Random Fields and Their Applications* . American Math. Society, 1980.

[16] S. Krishnamachari and R. Chellappa. Multiresolution Gauss-Markov Random Field Models for Texture Segmentation. *IEEE Transactions on Image Processing*, 6(2):251–267, 1997.

[17] S. Lakshmanan and H. Derin. Simultaneous Parameter Estimation and Segmentation of Gibbs Random Fields Using Simulated Annealing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11:799–813, 1989.

[18] S. M. Lavalle and S. A. Hutchinson. A Bayesian Segmentation Methodology For Parametric Image Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17:211–217, 1995.

[19] Chang-Tsun Li. *Unsupervised Image Segmentation Using Multiresolution Markov Random Fields"*. PhD thesis, University of Warwick, UK, 1998.

[20] C. S. Lu, P. C. Chung, and C. F. Chen. Unsupervised Texture Segmentation via wavelet Transform. *PR*, 30(5):729–742, 1997.

[21] M.R. Luettgen, W.C. Karl, and A. Willsky. Multiscale representations of Markov Random Fields. *IEEESP*, 41:3377–3395, 1993.

[22] B. S. Manjunath and R. Chellappa. Unsupervised Texture Segmentation Using Markov Random Field Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:478–482, 1991.

[23] D. K. Panjwani and G. Healey. Markov Random Field Models For Unsupervised Segmentation Of Textured color Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17:939–954, 1995.

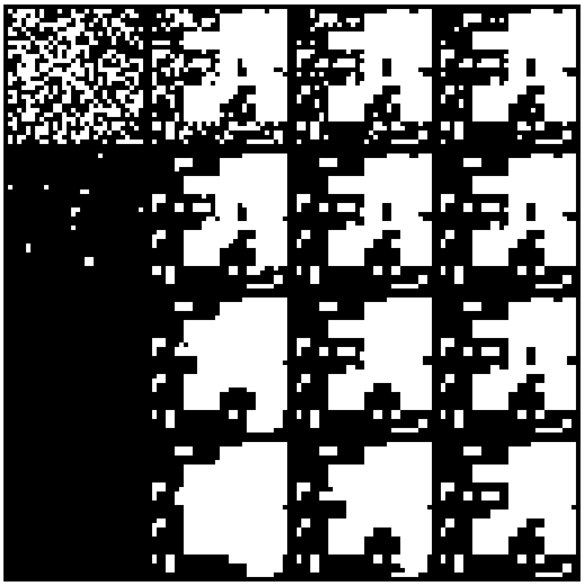
[24] P. Perez and F. Heitz. Restriction of a Markov Random Field on a Graph and Multiresolution Statistical Image Modeling. *IEEE Transactions on Information Theory*, 42(1):180–190, 1996.

[25] R. Szeliski. *Bayesian Modelling of Uncertainty in Low-level Vision.* Kluwer Academic Publishers, 1989.

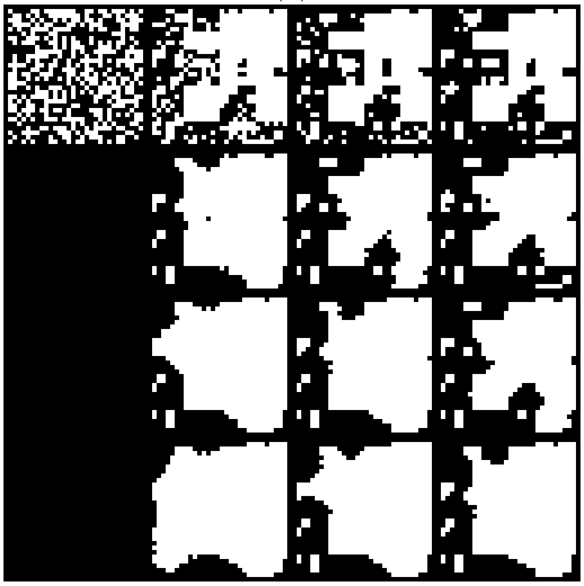
[26] R. Wilson, A. Calway, and E.R.S. Pearson. A Generalised Wavelet Transform for Fourier Analysis: The Multiresolution Fourier Transform and Its Application to Image and Audio Signal Analysis. *IEEE Transactions on Information Theory*, 38(2), March 1992.

[27] R. Wilson and Michael Spann. *Image Segmentation and Uncertainty.* Research Studies Press, 1988.

[28] J. W. Woods. Two-dimensional Discrete Markovian Fields. *IEEE Transactions on Information Theory*, 18:232–240, 1972.



(a)



(b)

Figure 3: Illustrating the effect of father-child interactions. Each of the 16 sub-images is sampled at 32*32 pixels using (a) 4-neighbours and the father as the conditioning elements and (b) using 8-neighbours and father. From top to bottom, the neighbour interaction energy increases, while from left to right the father-child interaction increases. Each image is the outcome of 2000 iterations using a common seed.

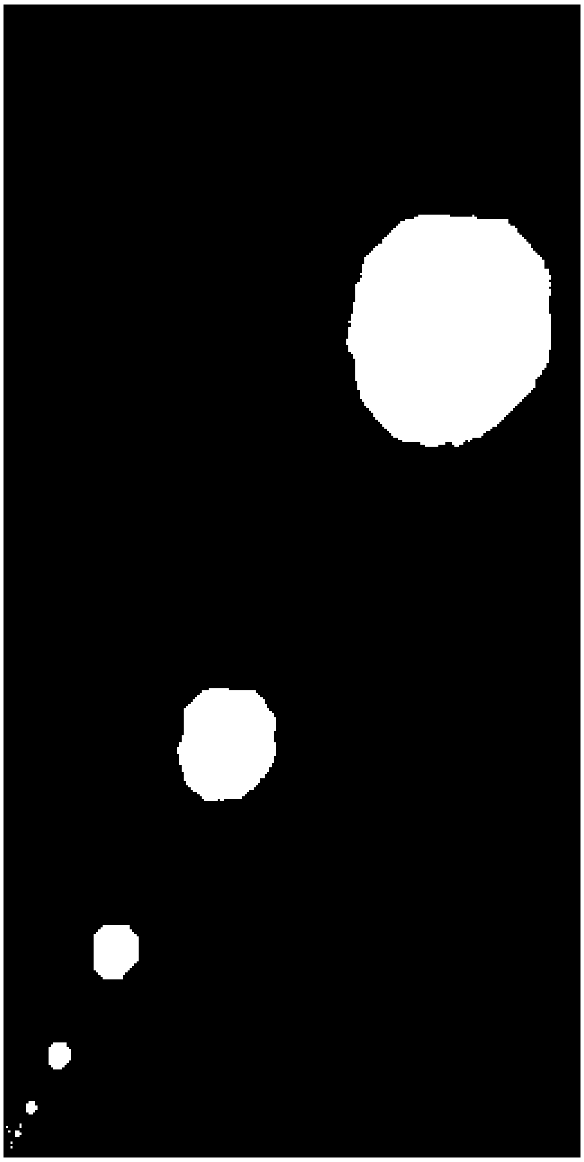


Figure 4: Sample from a MMRF process. The bottom level of the pyramid, k=8, is 256 by 256 pixels. 1000 iterations were used at each level.

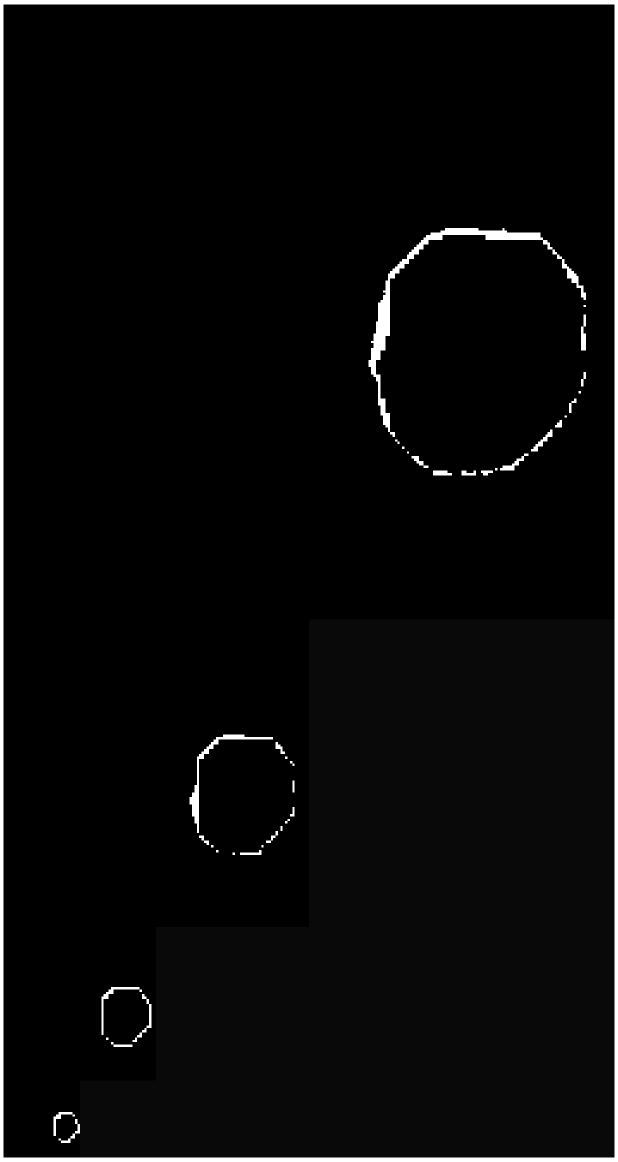


Figure 5: Illustrating the dependence between levels in the process: the difference between each level of the pyramid and the 'copy' from the level above. With supercritical parameters, the MRF acts to refine the existing structure.

Figure 6: Sample a second MMRF process with lower interaction potentials. The number of scales is 8, with the bottom level image being 128 by 128 pixels. The neighbourhood size was 8 for the bottom 3 levels of the pyramid and 500 iterations were run at each of these scales. The top 4 levels were generated with a pure quadtree model.

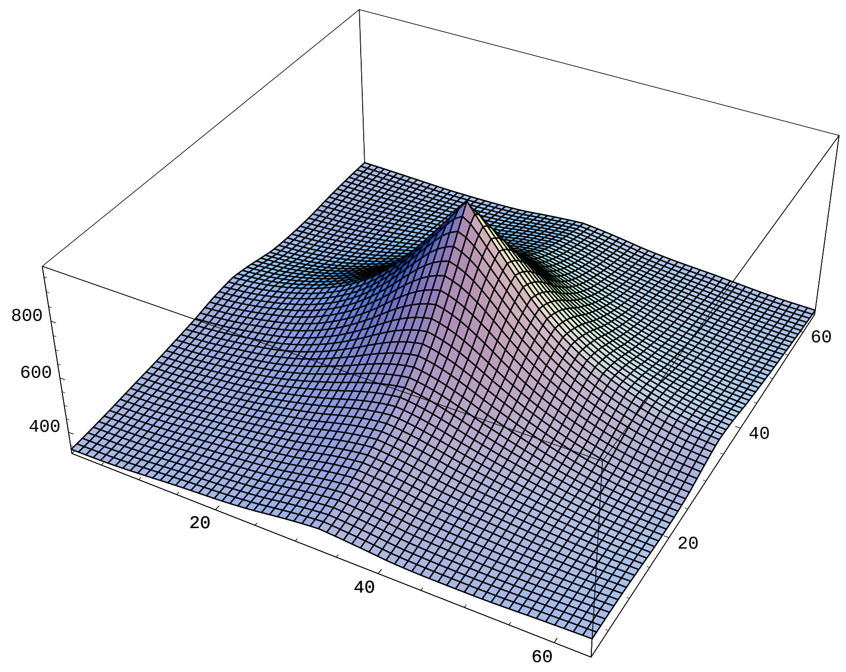


Figure 7: Sample autocorrelation from 20 samples of a binary MMRF. The number of scales is 8, with the bottom level image being 128 by 128 pixels. The neighbourhood size was 8 for the bottom 3 levels of the pyramid and 500 iterations were run at each of these scales. The top 4 levels were generated with a pure quadtree model.

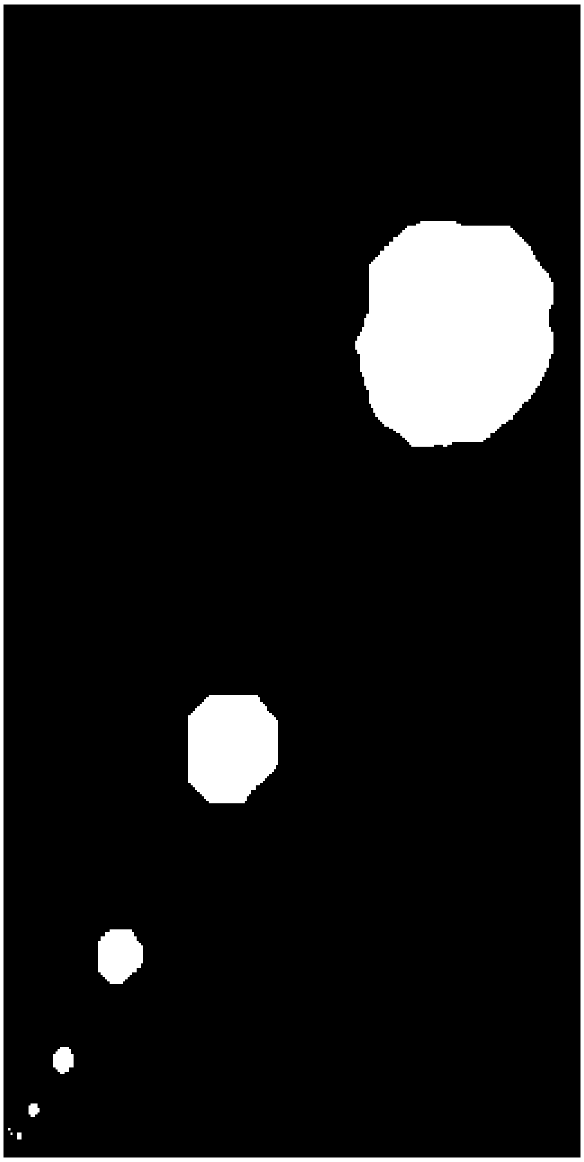


Figure 8: Prediction of level k from level $k-1$ in the image of figure 4, using simulated annealing over 200 iterations.

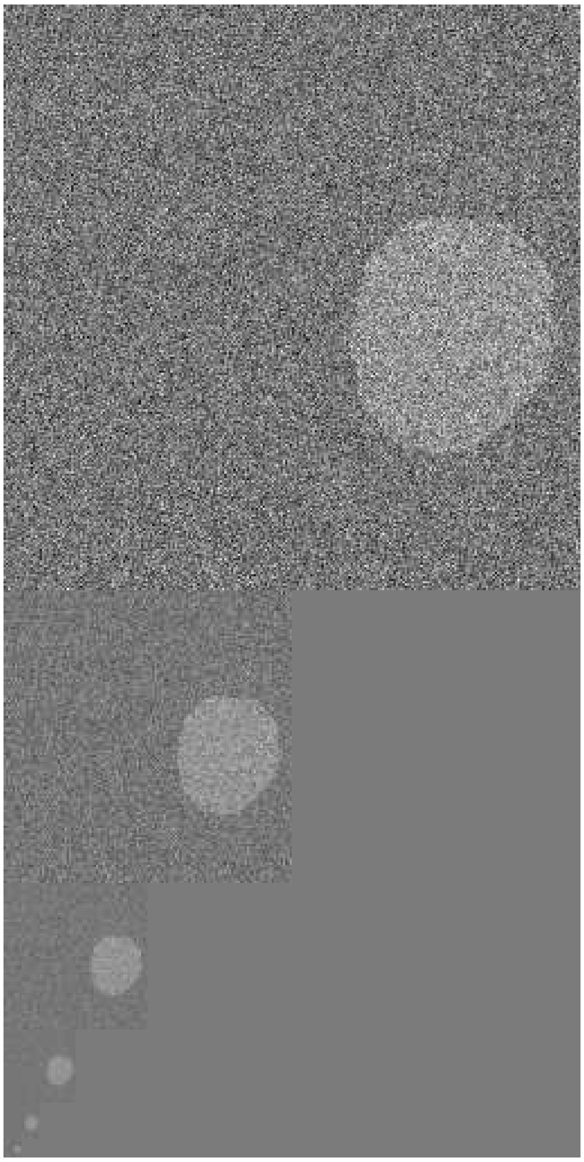


Figure 9: Noisy data pyramid obtained by quadtree averaging of image at bottom level, which has unit variance additive white Gaussian noise.

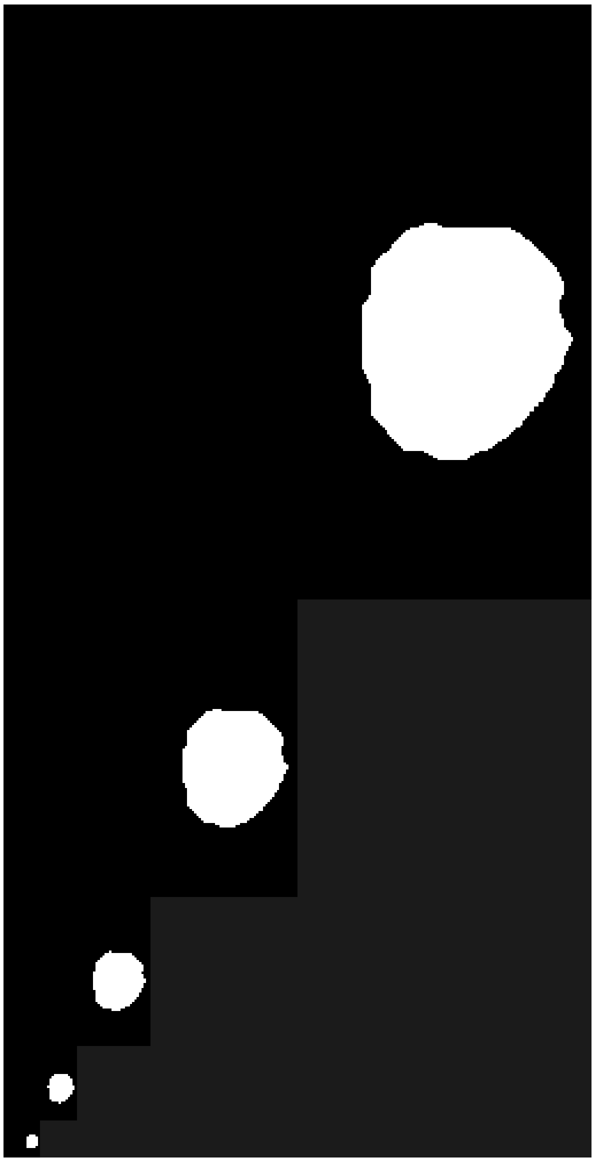


Figure 10: MMAP estimates of image in figure 4 from the noisy data of figure 9.

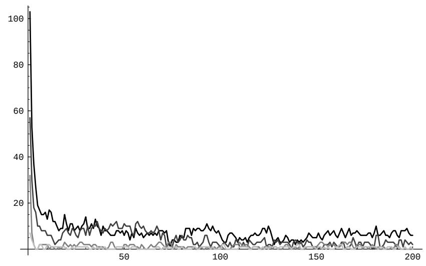


Figure 11: Plot of the number of changes on each iteration of the sampler, for levels 8 (top curve) up to 5(bottom curve).

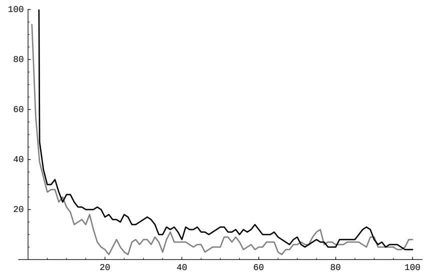


Figure 12: Number of changes on each iteration, for random initial configuration (top) and state copied from father (bottom) on level 8 of the pyramid.

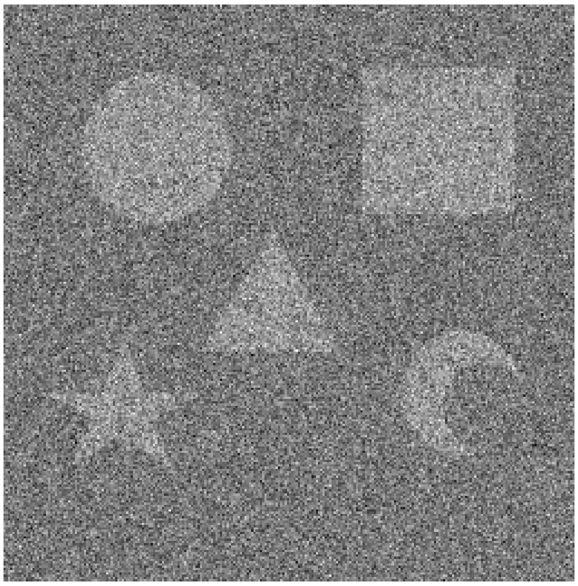


Figure 13: Noisy 'shapes' image with SNR=0dB.

Figure 14: Full resolution MMAP estimate of image in figure 13.

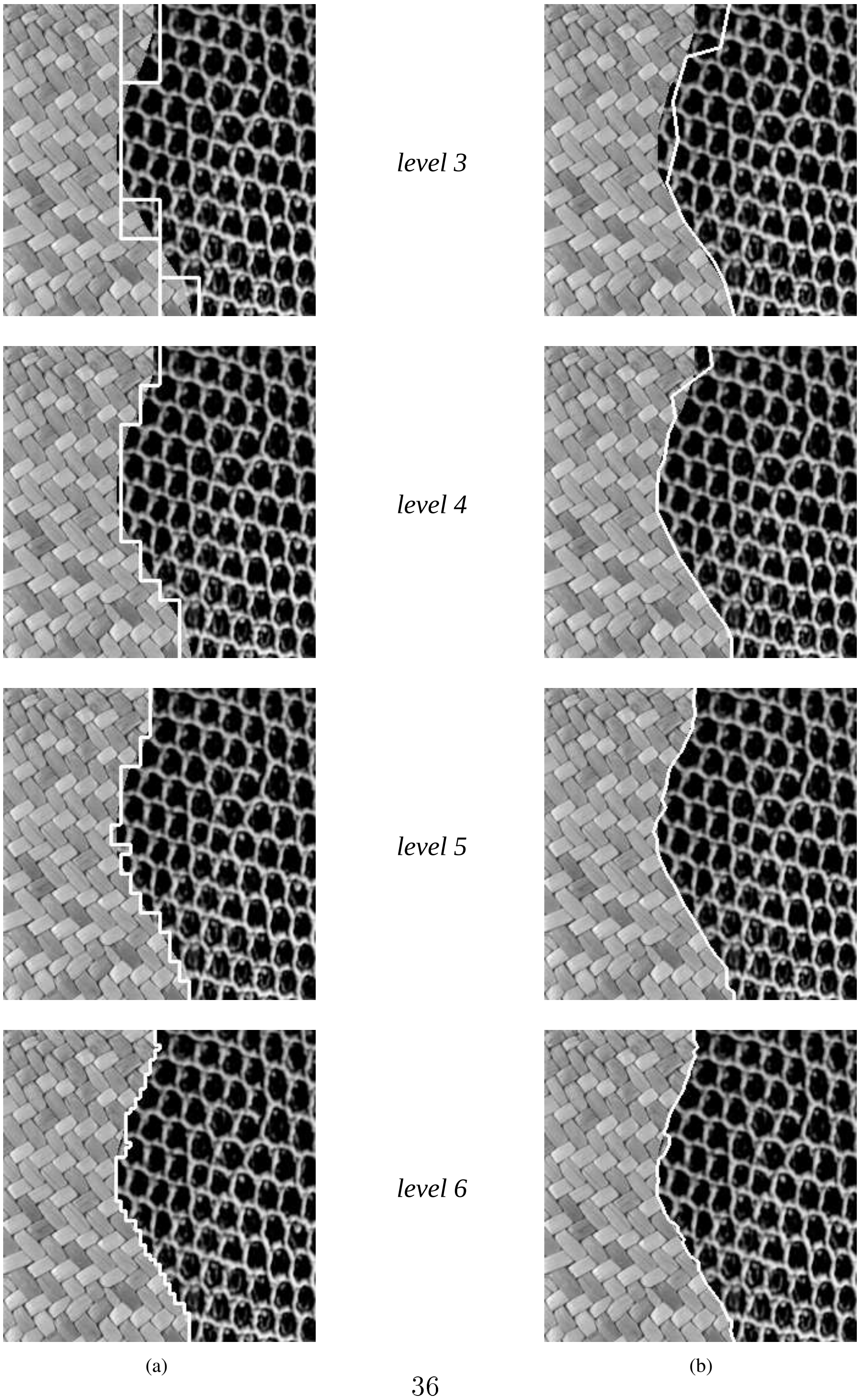


Figure 15: Segmentation results of *Image I*. (a) The results *before* the boundary process is executed at level 3 to 6 respectively. (b) The results *after* the boundary process is executed at level 3 to 6 .

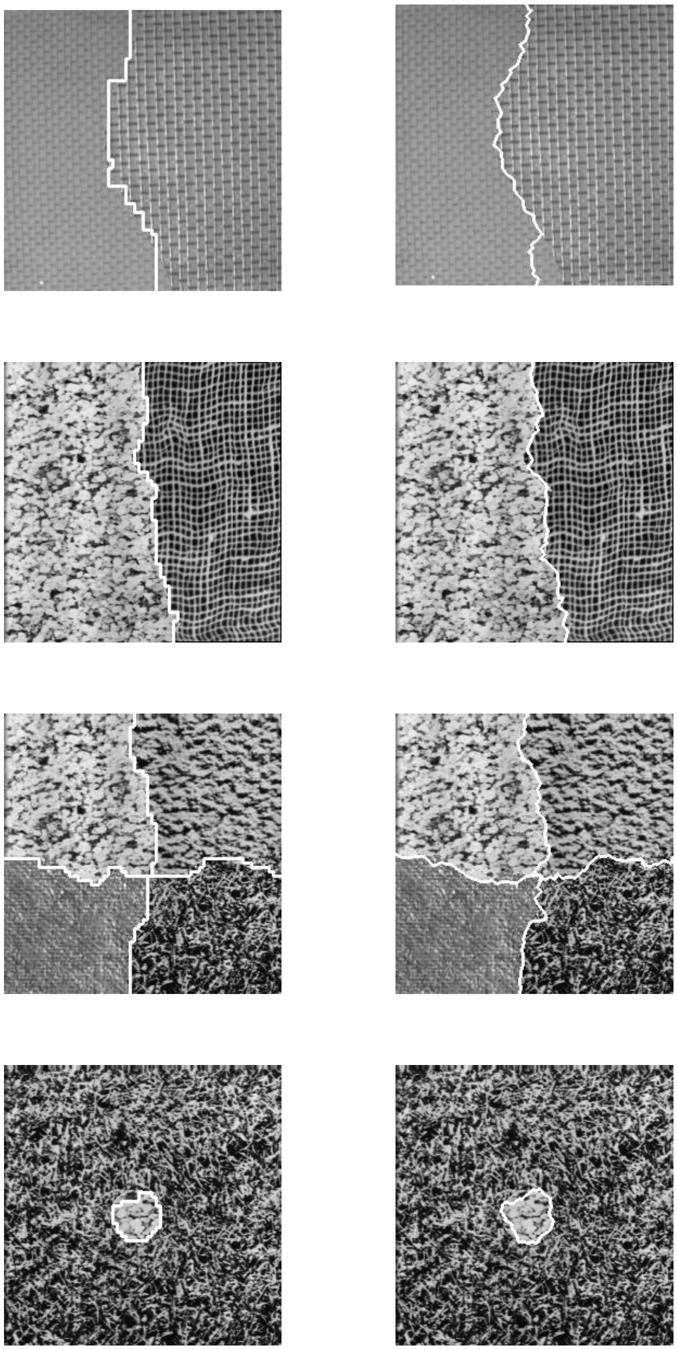


Figure 16: Summary of final segmentation results on various texture combinations. Left images: without boundary process; right images: with boundary process.