

Texture Synthesis and Unsupervised Recognition with Nonparametric Multiscale Markov Random Field Models

Rupert Paget, *Member, IEEE* and I. D. Longstaff, *Senior Member, IEEE*

Abstract

In this paper we present noncausal, nonparametric, multiscale, Markov Random Field (MRF) models for synthesising and recognising texture. The models have the ability to capture the characteristics of a wide variety of textures, varying from the structured to the stochastic. For texture synthesis, we use our own novel multiscale approach, incorporating local annealing, allowing us to use large neighbourhood systems to model some complex textures. The new multiscale texture synthesis algorithm also produces synthetic textures with few phase discontinuities. The power of our modelling technique is evident in that only a small training image is required to synthesis representative examples of the training texture. We also show how the high dimensional model of the texture may be modelled with lower dimensional statistics without over compromising the integrity of the representation. We then show how these models can be used for the unsupervised texture segmentation and recognition of images containing previously unseen textures; a technique useful in the practical application of recognising different terrain types from Synthetic Aperture Radar (SAR) images.

Keywords

Markov random fields, Nonparametric estimation, Multiscale, Texture synthesis, Unsupervised texture segmentation, Unsupervised texture recognition.

I. INTRODUCTION

IN the image processing literature, texture is usually defined in terms of the spatial interactions between pixel values. The aim of texture analysis is to capture the visual characteristics of texture in an analytical form by mathematically modelling these spatial interactions [26]. If these spatial characteristics are uniquely modelled, then different examples of textures from one source (population) can be associated analytically, and textures from other sources can be discriminated against. This allows segmentation of an image into its various textural components with each component being classified according to how well it fits the mathematical model of a particular texture. Although this approach is theoretically plausible, in practice current techniques require the number and type of textures to be a priori known. That is, they use a set of training textures to formalise the criteria by which the texture models become unique from each other, but not necessarily unique from other textures not included in the training set [12], [10], [7], [29]. These conventional models need only capture enough textural characteristics to classify the set of textures via discriminant analysis [17]. This approach is adequate if the image undergoing texture segmentation and classification is known to contain only textures which were modelled.

If a texture is to be recognised in a scene containing previously unseen textures, then a new approach is required. The texture models needs to capture more than just the characteristics required to distinguish it from other known textures – it needs to capture *all* the unique characteristics of that texture. When segmenting and classifying an image, the texture model could then be used to determine the probability a segmented area of an image had the same unique characteristics as the modelled texture without having to measure the probability against other possible textures. This would solve the problem of

The authors are with the Department of Electrical and Computer Engineering, University of Queensland, QLD 4072 AUSTRALIA. E-mail: {paget,idl}@elec.uq.edu.au

This work was supported by the Cooperative Research Centre for Sensor, Signal, and Information Processing.

previously unseen textures being present in the image. Images susceptible to this type of texture recognition problem are the Synthetic Aperture Radar (SAR) images of Earth's terrain obtained via satellites or airborne reconnaissance. These types of images contain a myriad of textures. It would be unreasonable to expect a conventional texture classification scheme to have previously identified and modelled all the different types of textures possibly present in such images.

Unfortunately, with the present knowledge of texture, obtaining a model that captures *all* the unique characteristics specific to a particular texture is an open problem [20]. Texture is not fully understood, and therefore, what constitutes the unique characteristics has not been defined. However, a reasonable way to test whether a model has captured all the unique characteristics is to use the same model to synthesise texture and subjectively judge the similarity of the synthetic texture to the original.

Conventional texture models, like the auto-models [2], autoregressive (AR) models [11], moving average (MA) models [26], or combination of both (ARMA) models [31], have not been found to provide a basis for realistically synthesising natural textures [26]. However recent advances in texture synthesis have produced models that are capable of synthesising natural textures [8], textures that contain both structural and statistical elements. These models are based on the stochastic modelling of various multi-resolution filter responses [5], [45], [28], [37], but they do not use third or higher order statistics, and it is undetermined whether the chosen filters are globally optimal for all textures. Julesz [30] suggested there was textural information in the higher order statistics, and Gagalowicz *et al.* [19] used third order statistics to generate some natural textures. Popat and Picard [39] successfully used a high-order, causal, nonparametric, multiscale MRF model to synthesise some structured natural textures.

In this paper, we present a noncausal, nonparametric, multiscale, Markov Random Field (MRF) model capable of synthesising both stochastic and structured textures. We use a multiscale synthesis algorithm [24] in which we incorporate a novel pixel temperature function for local annealing.

Although the synthesis test may indicate if a model has captured the specific characteristics of a texture, it does not determine whether the model is suitable for segmentation and classification. Using the same philosophy as in [45], a texture model should maximise its entropy while retaining the unique characteristics of the texture. In terms of the nonparametric MRF this is equivalent to reducing the statistical order of the model while retaining the integrity of the synthesised textures.

In this paper, we also present a method for reducing the statistical order of the nonparametric MRF model to a set of lower order statistical properties based on the *cliques* of the MRF. We show that this reduced model still contains the unique characteristics required for synthesising representative texture, but due to the lower order statistics is able to perform better segmentation and classification. By adjusting the extent of statistical reduction, the model can be optimised to capture the most unique characteristics while retaining the integrity of the synthesised textures, thereby producing a model suitable for unsupervised recognition (and hence labelling) of matching segments in an image.

Unsupervised recognition should not be confused with the term *unsupervised classification*, used [32], [40] to describe the selection of like-textures in an image without any prior knowledge of the number or types of textures present. Unsupervised classification can not be used to give anything but arbitrary labels to segments in a scene.

We have structured the paper so as to first outline the fundamentals of Markov Random Fields in section II. Then we move onto the nonparametric model, section III, which is immediately followed by our improved nonparametric model that is based on the cliques of the MRF, section IV. We then look at how we were able to use a multiscale representation to synthesise visually similar texture from a source texture, section V, and how the models

were used in the unsupervised segmentation and recognition of textured images, section VI. Results are presented in section VII. Finally we discuss how these methods may be used in a practical application, section VIII, and conclude with a summary and conclusion, section IX.

II. GENERAL MARKOV RANDOM FIELD MODEL

To model a digital image as a MRF, consider each pixel in the image as a site s on a lattice $S = \{s_1, s_2, \dots, s_N\}$, and the grey scale value associated with the pixel as equal to the value x_s . The site value x_s is then contained in the *state space* $\Lambda \doteq \{0, 1, 2, \dots, L-1\}$, where L is the number of *grey levels*. The *configuration space* for the set of variables $x = \{x_s, s \in S\}$ is given by Ω and is the set of all possible images, where $\Omega = \Lambda^N$.

Let Π be the (joint) probability measure on Ω with $\Pi(x) > 0, \forall x \in \Omega$. Besag [2, p 195] proved that the joint distribution $\Pi(x)$ is uniquely determined by its Local Conditional Probability Density Functions (LCPDF), $\Pi_s(x_s|x_r, r \neq s)$.

The property of a MRF is that the LCPDF is only dependent on a subset of sites $\{r \in \mathcal{N}_s \subset S\}$, where \mathcal{N}_s is the set of sites neighbouring s . The set \mathcal{N}_s is referred to as the *neighbourhood* of s , and $\mathcal{N} = \{\mathcal{N}_s \subset S, s \in S\}$ as the *neighbourhood system* which is the set of neighbourhoods on S . That is,

$$\Pi_s(x_s|x_r, r \neq s) = P(x_s|x_r, r \in \mathcal{N}_s) \quad s \in S, x \in \Omega. \quad (1)$$

An image is modelled by estimating the LCPDF with respect to a neighbourhood system \mathcal{N} .

The *Hammersley-Clifford theorem* [27], also referred to as the *MRF-Gibbs equivalence theorem*, and proved in [2], [20], [25], [36], gives form to the LCPDF $P(x_s|x_r, r \in \mathcal{N}_s)$ [22], [20] such that the respective Π defines a valid joint distribution.

Given a neighbourhood system \mathcal{N} , a *clique* is a set $C \subseteq S$ such that $s, r \in C, s \neq r$ implies $s \in \mathcal{N}_r$. That is, all pairs of distinct sites in a clique are neighbours. The single site subset is also a clique. The set of cliques defined on S with respect to \mathcal{N} is denoted by \mathcal{C} . The *local clique set* for the site $s \in S$ is denoted by $\mathcal{C}_s = \{C \in \mathcal{C}, s \in C\}$. A scheme for extracting the local clique set is given in [38]. A valid LCPDF is defined with respect to the cliques $C \in \mathcal{C}_s$ and the *potential functions* $V_C(x)$ defined on each clique [22], [20]:

$$P(x_s|x_r, r \in \mathcal{N}_s) = \frac{1}{Z_s} \exp \left\{ \sum_{C \in \mathcal{C}_s} V_C(x) \right\}, \quad (2)$$

where the *local normalising constant* $Z_s = \sum_{x_s \in \Lambda} P(x_s|x_r, r \in \mathcal{N}_s)$.

The essential ingredients of the *potential functions* $V_C(x)$ defined on each clique are: $V_C(x) = 0$ if $C \notin \mathcal{C}$; $V_C(x) \in \mathbb{R}$; and $V_C(x) = V_C(x')$ if $x_s = x'_s, \forall s \in C$. V is *normalised* if $V_C(x) = 0$ whenever $x_s = 0$ for any $s \in C$, where we assume $0 \in \Lambda$, although any other consistent value for x_s would do equally well. Normalised potentials ensure unique representation, but have no practical importance. For later reference, potential functions which fulfill all of the above criteria will be labelled \mathcal{N} -potentials.

The MRF-Gibbs equivalence theorem implicitly requires the neighbourhood system to adhere to the criterion $s \in \mathcal{N}_r \Leftrightarrow r \in \mathcal{N}_s$. This in turn implies that the neighbourhoods must be symmetrical for homogeneous MRFs. The symmetrical neighbourhood systems used in this paper are the same as in [20], [22] for which the neighbourhood system $\mathcal{N}^o = \{\mathcal{N}_s^o, s \in S\}$ is defined as

$$\mathcal{N}_s^o = \{r \in S : 0 < |s - r|^2 \leq o\}, \quad (3)$$

where $|s - r|$ is the Euclidean distance between two points $s, r \in S$, and o is the order of the neighbourhood system, (see Fig. 1).

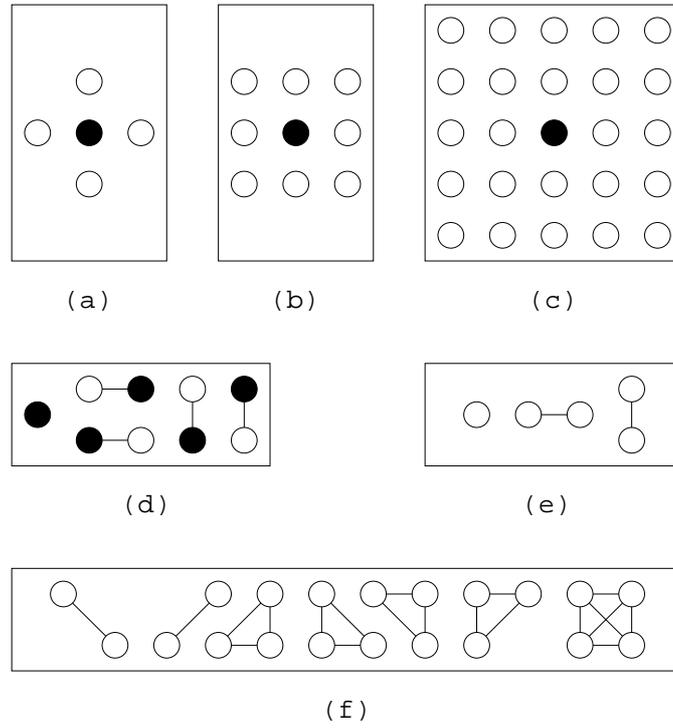


Fig. 1. Neighbourhoods and cliques: (a) The first order neighbourhood $o = 1$ or *nearest-neighbour* neighbourhood for the site $s = \bullet$ and $r = \circ \in \mathcal{N}_s$; (b) second order neighbourhood $o = 2$; (c) eighth order neighbourhood $o = 8$. (d) local clique set for nearest-neighbour neighbourhood; (e) clique types for nearest-neighbour neighbourhood; (f) additional clique types for second-order neighbourhood.

A. A Representation for \mathcal{N} -Potential V

The following construction for the \mathcal{N} -Potential is due to Grimmett [25], which first requires the Möbius inversion theorem [41].

Theorem 1: Möbius inversion theorem for arbitrary real functions F and G defined on the subsets B and C of some finite set A :

$$F(A) = \sum_{B \subseteq A} G(B) \quad \text{iff} \quad G(B) = \sum_{C \subseteq B} (-1)^{|B|-|C|} F(C) \quad (4)$$

or, equivalently,

$$F(A) = \sum_{B \subseteq A} \sum_{C \subseteq B} (-1)^{|B|-|C|} F(C), \quad (5)$$

where $|A|$ = number of sites in set A . Moussouris [36] developed an elegant proof for this construction.

Given an image $x \in \Omega$ and a set $A \subset S$, denote

$$x^A = \{x_s^A, s \in S\}, \quad x_s^A = \begin{cases} x_s, & s \in A \\ 0, & s \notin A. \end{cases} \quad (6)$$

For ease of notation, denote $x_{(s)} = \{x_r, r \neq s\}$.

Then, any $\Pi > 0$ is a Gibbs distribution with respect to \mathcal{N} -potentials:

$$V_C(x) = \sum_{C' \subseteq C} (-1)^{|C|-|C'|} \log \Pi(x^{C'}), \quad C, C' \in \mathcal{C}. \quad (7)$$

Moreover from [20], for any element $s \in C$,

$$V_C(x) = \sum_{C' \subseteq C} (-1)^{|C|-|C'|} \log \Pi_s(x_s^{C'} | x_{(s)}^{C'}), \quad C, C' \in \mathcal{C}. \quad (8)$$

The re-expression for the joint distribution $\Pi(x)$, as given by Geman [20] from equation (7), can be similarly obtained for the LCPDF by the Möbius inversion of equation (8), giving

$$\log P(x_s|x_r, r \in \mathcal{N}_s) = \sum_{C \in \mathcal{C}_s} \sum_{\substack{C' \subseteq C, \\ s \in C'}} (-1)^{|C|-|C'|} \log \Pi_s(x_s^{C'}|x_{(s)}^{C'}). \quad (9)$$

Moussouris [36] gave a further re-expression of the clique decomposition formulae of $\Pi(x)$. The equivalent re-expression of the LCPDF clique decomposition from equation (9) is

$$P(x_s|x_r, r \in \mathcal{N}_s) = \prod_{C \subseteq \mathcal{C}_s} \Pi_s(x_s^C|x_{(s)}^C)^{n_{C_s C}}, \quad (10)$$

where $\mathcal{C}_s = \{C \in \mathcal{C}, s \in C\}$ is the local clique set and

$$n_{C_s C} = (-1)^{|C|} \sum_{C' \subseteq C' \in \mathcal{C}_s} (-1)^{|C'|}. \quad (11)$$

III. NONPARAMETRIC MRF MODEL

The nonparametric MRF model is based on estimating the LCPDF from a multi-dimensional histogram of the neighbourhood over a homogeneous textured image. Each dimension of the histogram represents a site from the neighbourhood of the LCPDF with one dimension for the site itself. The total number of dimensions is the statistical order of the model and is equal to the neighbourhood size plus one. Although it would be informative to test larger and larger neighbourhoods for modelling texture, there is a limit to the size of which may be successfully modelled. Silverman [43, p 94] showed that to maintain accuracy in a model, the amount of sample data needs to grow almost exponentially with the dimensionality of the histogram. As we are dealing with a limited amount of sample data – approximately equal to the number of pixels in a source texture image – the accuracy of the model will rapidly decrease as the dimensionality of the histogram increases. In such cases, when the sample data is sparsely dispersed over the multi-dimensional histogram domain, nonparametric estimates of the LCPDF tend to be more reliable than their parametric counterparts if the underlying true distribution is unknown [43]. This is because nonparametric estimation only tries to model those areas of the multi-dimensional histogram that contain the data rather than trying to fit a model over the whole of the histogram domain as with parametric estimation [13], [42].

Density estimation is an attempt to discern a representative probability density function from the sampled data. However in using nonparametric estimation the LCPDF may no longer define a valid joint distribution Π , but the underlying true shape of the distribution will not be compromised by trying to fit the shape of an assumed parametric distribution to the data.

The most common nonparametric density estimator is the Parzen-window estimator [43], [17]. We chose a standard multi-dimensional Gaussian density function as the kernel function for the Parzen-window estimator. Given a source image $y \in \Omega$ of a homogeneous texture and a predefined neighbourhood system \mathcal{N} defined on a lattice S_y , the sample data $\mathbf{Z}_p = [y_p, y_q, q \in \mathcal{N}_p]^T$ is taken from all sites $p \in S_y$ for which $\mathcal{N}_p \subset S_y$. Denote the variable n as the number of sample data \mathbf{Z}_p , *i.e.*, the number of sites $\{p \in S_y, \mathcal{N}_p \subset S_y\}$. Equate $d = |\mathcal{N}_p| + 1$ the number of elements in the vector \mathbf{Z}_p , *i.e.*, d equals the dimensionality of the LCPDF. Finally the Parzen-window estimator requires a window parameter h . Thus the estimated LCPDF, defined with respect to the Parzen-window density estimator [43], [17], is

$$\hat{P}(x_s|x_r, r \in \mathcal{N}_s) = \frac{\sum_{p \in S_y, \mathcal{N}_p \subset S_y} \exp \left[-\frac{1}{2h^2} (\mathbf{z} - \mathbf{Z}_p)^T (\mathbf{z} - \mathbf{Z}_p) \right]}{\sum_{x_s \in \Lambda} \sum_{p \in S_y, \mathcal{N}_p \subset S_y} \exp \left[-\frac{1}{2h^2} (\mathbf{z} - \mathbf{Z}_p)^T (\mathbf{z} - \mathbf{Z}_p) \right]}, \quad (12)$$

where $\mathbf{z} = [x_s, x_r, r \in \mathcal{N}_s]^T$.

The window parameter h determines the amount of smoothing applied to the histogram. The aim is to choose h so as to obtain the best estimate of the LCPDF. If h is too small, the LCPDF will not be general enough to model all textures subjectively judged similar to the modelled texture. If h is too large, then the LCPDF will be too general, and some detail associated with the texture may be lost. We chose the window parameter h to be the optimal window parameter as specified by Silverman [43, p 85].

$$h = \sigma \left\{ \frac{4}{n(2d+1)} \right\}^{1/(d+4)}, \quad (13)$$

where σ^2 is the average marginal variance. In our case, the marginal variance is the same in each dimension of the multi-dimensional histogram and, therefore, σ^2 equals the variance associated with the one-dimensional histogram of the source image y .

It was Moussouris [36] who suggested that the Markovian system could be simplified by imposing stronger conditions on the LCPDF. In the next section we redefine the MRF with a stronger assumption so as to use the *cliques* of the MRF in estimating the LCPDF. This will produce a consistent LCPDF with a valid joint distribution Π , and lower the statistical order of the model.

IV. STRONG NONPARAMETRIC MRF MODEL

Here we present our new version of the nonparametric model. It is still estimated over the same neighbourhood, but the Parzen-window estimation is performed over a set of smaller domains. We estimate the LCPDF as a function of its marginal distributions by assuming that there is conditional independence between non-neighbouring sites for any subset of S . This is a much stronger assumption than made for a normal MRF which defines a site as being conditionally independent upon its non-neighbouring sites given all of the neighbouring sites. We show that this strong MRF model is equivalent to the Analysis-of-variance (ANOVA) construction [18]. This equivalence allows us to use the theorems from the ANOVA construction to estimate the LCPDF of the strong MRF model.

MRF condition

$$\Pi_s(x_s|x_r, r \neq s) = P(x_s|x_r, r \in \mathcal{N}_s), \quad \forall s \in S, x \in \Omega. \quad (14)$$

Strong MRF condition

$$\Pi_s(x_s|x_r, r \neq s, r \in A \subseteq S) = P(x_s|x_r, r \in A \cap \mathcal{N}_s), \quad \forall s \in S, x \in \Omega, A \subseteq S. \quad (15)$$

The strong MRF condition states that the LCPDF $\Pi_s(x_s|x_r, r \in A \subseteq S)$, for any subset $A \subseteq S$, is only dependent on those sites $\{x_r, r \in A \cap \mathcal{N}_s\}$, regardless of whether $\mathcal{N}_s \subseteq A$. This is contrary to a standard MRF for which, when given only some of the pixels $\{x_r, r \in \mathcal{N}_s\}$, the conditional probability $\Pi_s(x_s|x_{(s)})$ will, in general, be no longer conditionally independent of those sites $r \notin \mathcal{N}_s$.

If an image can be modelled as a MRF, it does not necessarily follow that it can also be modelled as a strong MRF. However, a common approach to simplifying complex mathematical problems is to assume a degree of independence where none may exist. We therefore assume an extra degree of conditional independence to simplify the MRF model as a strong MRF model.

We denote $P_A(x) = P(x_s, s \in A)$, where $A \subseteq S$. The probability $P_A(x)$ is then the joint probability for the domain $A \subseteq S$:

$$P_A(x) = \sum_{\substack{y \in \Omega, \\ y^A = x^A}} \Pi(y), \quad A \subseteq S. \quad (16)$$

where x^A is defined as in equation (6). The null probability $P_\emptyset(x)$ is therefore defined by equation (16) as $P_\emptyset(x) = 1$.

The strong MRF condition may be expressed in the form of the following identity. Given two sites $s, t \in S$ for which neither is a neighbour of the other, *i.e.*, $t \notin \mathcal{N}_s \Leftrightarrow s \notin \mathcal{N}_t$, then from the strong MRF condition defined by equation (15),

$$P(x_s|x_t, x_r, r \in B) = P(x_s|x_r, r \in B), \quad (17)$$

giving

$$\frac{P_{B+s+t}(x)}{P_{B+t}(x)} = \frac{P_{B+s}(x)}{P_B(x)}, \quad (18)$$

where $B \subseteq S, s \notin B, t \notin B$. The notation $B + s$ represents a set of sites B plus the site s , whereas $B - s$ is the same set B excluding the site s .

For any $A \subseteq S, s \in A$, denote

$$P_A(x_s|x_{(s)}) = P_A(x)/P_{A-s}(x) = P(x_s|x_r, r \in A - s), \quad (19)$$

Proposition 1: Given a neighbourhood system \mathcal{N} , the LCPDF of a strong MRF is

$$\log P(x_s, x_r, r \in \mathcal{N}_s) = \sum_{C \in \mathcal{C}_s} \sum_{\substack{C' \subseteq C, \\ s \in C'}} (-1)^{|C|-|C'|} \log P_{C'}(x), \quad (20)$$

or equivalently,

$$\log P(x_s|x_r, r \in \mathcal{N}_s) = \sum_{C \in \mathcal{C}_s} \sum_{\substack{C' \subseteq C, \\ s \in C'}} (-1)^{|C|-|C'|} \log P_{C'}(x_s|x_{(s)}) \quad (21)$$

which via the Moussouris [36] decomposition may be expressed in the form,

$$P(x_s, x_r, r \in \mathcal{N}_s) = \prod_{C \in \mathcal{C}_s} P_C(x)^{n_{C_s C}}, \quad (22)$$

and

$$P(x_s|x_r, r \in \mathcal{N}_s) = \prod_{C \in \mathcal{C}_s} P_C(x_s|x_{(s)})^{n_{C_s C}}, \quad (23)$$

respectively, where

$$n_{C_s C} = (-1)^{|C|} \sum_{C' \subseteq C, s \in C'} (-1)^{|C'|}, \quad (24)$$

and C, C' are cliques from the local clique set $\mathcal{C}_s = \{C \in \mathcal{C}, s \in C\}$.

We provide two proofs for Proposition 1. The first method relies on the Möbius inversion formula (4), and follows Grimmett's [25] and Moussouris's [36] construction for the \mathcal{N} -potential V , Appendix A. The second proof is based on the ANOVA construction [18] for testing variable independence in a distribution, Appendix B. The proofs show that the strong MRF model is equivalent to the ANOVA construction.

Even though equation (22) represents the general clique decomposition for $P(x_s, x_r, r \in \mathcal{N}_s)$, it is subject to condition (43). Bishop *et al.* [4] did not derive the general formula (22) for the ANOVA construction, but do define under what conditions it may exist. Given a set of cliques over which the formula (22) is derived, Bishop *et al.* [4, p 76] outlined rules for determining when formula (22) is valid.

A. Estimation of the strong LCPDF

The strong LCPDF, $P(x_s|x_r, r \in \mathcal{N}_s)$, is estimated by first calculating the set of marginal probabilities $\{P_C(x_s|x_{(s)}), C \in \mathcal{C}_s\}$ and then combining them to obtain $P(x_s|x_r, r \in \mathcal{N}_s)$ via equation (23). Each marginal probability $P_C(x_s|x_{(s)})$, $C \in \mathcal{C}_s$ is estimated in the same way as the nonparametric LCPDF (12). However, in the case of the marginal estimates, equation (12) is calculated with respect to C rather than \mathcal{N}_s . The set of sample vectors, $\{Z_p, p \in S_y, \mathcal{N}_p \subset S_y\}$, is therefore modified to $\{Z_p, p \in S_y, C \subset S_y\}$. The dimensionality of z is modified to $d = |C|$, and the optimal window parameter, equation (13), is recalculated for each $C \in \mathcal{C}_s$.

Equations (22) and (23) give the *direct estimate* for determining $P(x_s, x_r, r \in \mathcal{N}_s)$ and $P(x_s|x_r, r \in \mathcal{N}_s)$ from its marginal probabilities, respectively. However for the neighbourhood systems defined by equation (3), the existence rules [4, p 76] preclude equations (22) and (23) for other than the nearest-neighbour neighbourhood system \mathcal{N}^1 with pairwise clique decomposition. Alternatively, the *iterative proportional fitting technique* [4, p 83] and [18, p 37] is able to be used for the evaluation of the strong MRF model for all neighbourhood systems with any clique decomposition. However, regardless of the number of grey levels and which marginals are used, the minimum memory space required to iteratively calculate the LCPDF is $2^{|\mathcal{N}_s|+1}$ floating point values. Our experiments were run on a massive parallel processor machine, the MasPar¹. Although we obtained the necessary increase in processing speed, it was at the expense of available memory. Because of this we found that we could not estimate the LCPDF for neighbourhoods larger than ten sites. This meant that the iterative proportional fitting technique could only be used for neighbourhood systems \mathcal{N}^1 and \mathcal{N}^2 .

Our variation on the direct estimate (22) is the *simple estimate*,

$$P(x_s, x_r, r \in \mathcal{N}_s) = \prod_{\substack{C \in \mathcal{C}_s, \\ C \not\subset C' \in \mathcal{C}_s}} P_C(x). \quad (25)$$

The simple estimate can be calculated on the MasPar computer for large neighbourhood systems with various clique decompositions. The simple estimate is similar to the direct estimate, except it only incorporates those marginal probabilities $P_C(x)$ defined on the major cliques contained in the local clique set, $\{C \in \mathcal{C}_s, C \not\subset C' \in \mathcal{C}_s\}$, [38]. The clique decomposition summation (25) is performed over those cliques contained in the local clique set which are *not* subsets of other cliques contained in the local clique set. Intuitively, the LCPDF will be more “peaked” around the modes. For texture synthesis purposes, a “peaked” LCPDF means that the *Gibbs sampler* [22] will behave more like the *Iterative conditional modes (ICM) algorithm* [3].

V. MULTISCALE TEXTURE SYNTHESIS

Texture synthesis is a means of testing whether the LCPDF has captured the textural characteristics required to model a particular texture. How specific the required texture characteristics need to be is governed by the intended application of the texture model. The aim of our work was to develop a texture model suitable for unsupervised texture recognition, therefore, able to capture *all* textural characteristics unique to a source texture. We propose that if the model is capable of synthesising texture that is visually indistinguishable from the source texture, then it has captured *all* the visual characteristics of that texture.

There have been quite a few attempts at synthesising textures, but none of the conventional techniques have produced a general model for natural textures [26]. However

¹DEC mpp 12000 with 16,384 processors each with 64 kb of memory, which can yield speeds of up to 60 Giga instructions per second

new methods based on stochastic modelling of various multi-resolution filter responses have produced impressive results [5], [45], [28], [37]. Alternatively Popat and Picard [39] successfully used a high order causal nonparametric multiscale MRF model to synthesis structured natural textures. In fact our approach is indicative of theirs, but where they suffered from *phase discontinuity* we used our method of *local annealing* to synthesis highly representative examples of natural textures.

We perform texture synthesis via a multiscale synthesis algorithm incorporating our novel pixel temperature function. As part of the synthesis process, the pixel temperature function actually reduces the dimensionality of the multi-dimensional histogram which, in turn, alleviates the problem associated with estimating the model in a high-dimensional space. This means we are able to use large neighbourhood systems to model texture.

We can synthesise a texture from a MRF model by a method known as stochastic relaxation (SR) [14], [21], [22]. We start with an image and iteratively update pixels in the image with respect to the LCPDF. This generates a sequence of images $\{x(0), x(1), \dots, x(n)\}$ with the property,

$$\lim_{n \rightarrow \infty} P(x(n)|x(0)) = \Pi(x) \quad \forall x \in \Omega. \quad (26)$$

Two well-known SR algorithms are the Metropolis algorithm [35] and the Gibbs sampler [22]. Besag [3] introduced deterministic relaxation algorithm called the *Iterative conditional modes (ICM) algorithm*. Synthesis by the Gibbs sampler tends to converge a texture defined by the equilibrium condition (26), whereas for the ICM algorithm, the synthesis tends to a texture more conditional on the starting image $x(0)$. Both algorithms are adequate for synthesising texture.

A. Multiscale Relaxation

A problem with the single scale relaxation process is that global image characteristics evolve indirectly in the relaxation process [24], [44]. Global image characteristics are typically only propagated across the image lattice by local interactions and therefore evolve slowly, requiring long relaxation times to obtain equilibrium, as defined by equation (26). With multiscale relaxation (MR), we attempt to overcome this problem by implementing stochastic relaxation (SR) at various resolutions; first at a low resolution and then at progressively higher resolutions [24], [5], [34], [7], [1]. The information obtained from SR at one resolution is used to constrain the SR at the next highest resolution. By this method, global image characteristics that have been resolved at a low resolution are infused into the relaxation process at the higher resolutions. This helps reduce the number of iterations required to obtain equilibrium [44]. Multiscale relaxation also helps the ICM algorithm converge to an image that is closer to the global maximum of the joint distribution Π [6], [16].

The multiscale model may be described by a multigrid representation of the image, as shown in Fig. 2. The grid at level $l = 0$ represents the image at the original resolution, where each intersection point ‘•’ is a site $s \in S$. The lower resolutions, or higher grid levels $l > 0$, are decimated versions of the image at level $l = 0$. This multigrid representation is also used by Popat and Picard [39].

Given an $N \times M$ image x , we define the rectangular lattice on which to represent this image at grid level $l = 0$ as

$$S = \{s = (i, j) : 0 \leq i < N, 0 \leq j < M\}. \quad (27)$$

The multigrid representation of the image x is then the set of images x^l , for grid levels $l \geq 0$. The image x^l is defined on the lattice $S^l \subset S$, where

$$S^l = \left\{ s = (2^l i, 2^l j) : 0 \leq i < \frac{N}{2^l}, 0 \leq j < \frac{M}{2^l} \right\}. \quad (28)$$

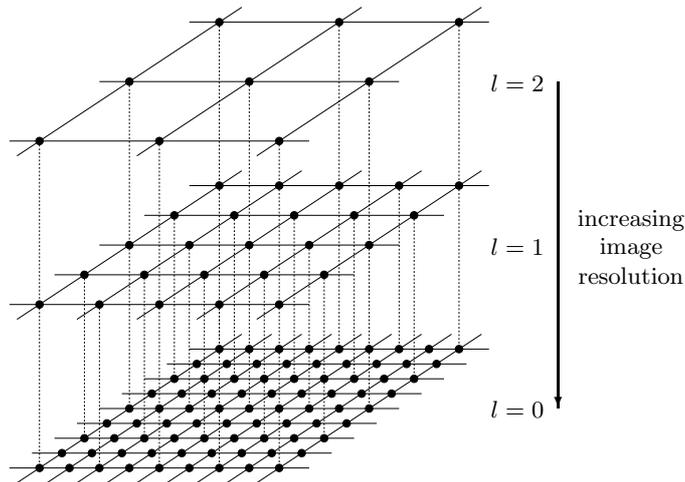


Fig. 2. Grid organisation for MR via decimation.

The set of sites S^l at level l represents a decimation of the previous set of sites S^{l-1} at the lower grid level $l-1$. On this multiscale lattice representation, we need to redefine the neighbourhood system for each grid level $l \geq 0$. Therefore, we define neighbourhood \mathcal{N}_s^l , $s \in S^l$ with respect to order o as

$$\mathcal{N}_{s=(2^l i, 2^l j)}^l = \{r = (2^l p, 2^l q) \in S^l : 0 < (i-p)^2 + (j-q)^2 \leq o\}. \quad (29)$$

We use the MR algorithm proposed by Gidas [24], which maintains the constraint imposed by the image x^{l+1} through the entire SR process at level l and successive levels $k < l$. The constraint is such that, at any point through the SR process at level l , the image x^{l+1} may still be obtained from x^l by the multigrid representation. As we have used decimation to obtain x^{l+1} from x^l , the MR constraint is maintained at level l by *not* performing SR on those sites $s \in S^{l+1} \subset S^l$, as defined by equation (28).

To better incorporate the MR constraint into the SR process, we introduce a novel pixel temperature function, which directly defines how the MR constraint is imposed on the SR process. The pixel temperature function also produces an equilibrium state which may be used to determine when the SR process can be terminated at one level and started at the next level. The multiscale relaxation algorithm incorporating our novel pixel temperature function is also capable of synthesising texture with minimal phase discontinuities.

B. Pixel Temperature Function

The aim of our pixel temperature function is to define a degree of “confidence” in a pixel having the correct value. Each pixel has its own temperature t_s , representing the confidence associated with the pixel x_s . The confidence is expressed as a value $0 \leq t_s \leq 1$, where 0 represents complete confidence, and 1 none at all. We have defined the pixel temperature function so as to relate to the global temperature function T used in stochastic annealing [22]. In fact, the function of our local pixel temperature may be regarded as an implementation of *local annealing* in the relaxation process.

In the MR algorithm, the confidence or temperature associated with each pixel is used to modify the dimensionality of the LCPDF. This is done so that the conditional dependence of the LCPDF is strongest on those pixels with $t_s \rightarrow 0$, and weakest for those with $t_s \rightarrow 1$. The pixel temperature is incorporated into the LCPDF by modifying the form of $(\mathbf{z} - \mathbf{Z}_p)$ in equation (12). Given that the estimate of the LCPDF is for an image x at the site $s \in S$, the vector $\mathbf{z} = [x_s, x_r, r \in \mathcal{N}_s]^T$. The sample data \mathbf{Z}_p is from a source image y defined on

the lattice S_y , where $\mathbf{Z}_p = [y_p, y_q, q \in \mathcal{N}_p]^T$ $p \in S_y$, and

$$(\mathbf{z} - \mathbf{Z}_p) \Rightarrow [x_s - y_p, (x_r - y_{r-s+p})(1 - t_r), r \in \mathcal{N}_s]^T, \quad (30)$$

where the pixel temperature t_r is from the same site as the pixel value x_r , $r \in S$ for the image x .

Before the SR algorithm starts at level l , those pixels which were relaxed at the previous level, $l + 1$, are given a pixel temperature $t_s = 0 \forall s \in S^{l+1}$, *i.e.*, complete confidence. The other pixels have their pixel temperatures initialised to $t_s = 1 \forall s \notin S^{l+1}$, *i.e.*, no confidence. After a pixel is relaxed, it has its temperature updated. We relate pixel temperature to pixel confidence, where pixel confidence is associated with the probability that x_s is the correct pixel value for the site s . Full pixel confidence occurs when x_s is sampled from an LCPDF at equilibrium, or when the LCPDF is completely conditional on its neighbouring pixel values. This occurs when $t_r = 0, \forall r \in \mathcal{N}_s$. The confidence associated with the pixel value x_s is then dependent on the pixel temperatures $t_r, r \in \mathcal{N}_s$. Therefore we use the formula

$$t_s = \max \left\{ 0, \frac{-1 + \sum_{r \in \mathcal{N}_s} t_r}{|\mathcal{N}_s|} \right\} \quad (31)$$

to define the update schedule of the pixel temperature t_s for each time the pixel x_s is updated.

Initially, only those sites that had their values relaxed at the previous grid level are used in the LCPDF. However, as the SR iterations progress, more sites gain a degree of confidence. When $t_s = 0, \forall s \in S^l$, we can say the SR process has reached an equilibrium state, indicating that the image can be propagated to the next lower grid level. The process is then repeated at the next grid level $l - 1$ and so on until the final grid level $l = 0$ has undergone constrained SR.

C. Synthesised Textures

We used source images of size 128×128 pixels to estimate the LCPDF from which images of size 256×256 were synthesised via algorithm 1, Fig. 3. A subjective comparison of the source and resulting synthetic textures, Fig. 4, show that the noncausal nonparametric multiscale Markov random field texture models form highly representative models of natural textures. This confirms that the characteristics of the source texture have indeed been captured by the models.

The results obtained for the strong MRF model, Figs. 4 (a.2) (b.2) and (c.2), suggest that these textures may be successfully modelled with just third order statistics. Although the plain nonparametric MRF model has produced better synthesis results, the strong MRF model is more likely to be the better model for the segmentation and recognition of texture. This is because it uses lower order statistics, thereby increasing its entropy while retaining the unique characteristics of the texture [45].

VI. MULTISCALE UNSUPERVISED TEXTURE RECOGNITION

MRF models have mainly been used for the *supervised* classification of texture, for which a library of pre-modelled textures must exist in order for discriminant analysis to be used. [10], [15], [23], [12], [9]. However, this approach would be cumbersome for SAR images of the Earth's terrain as they contain a myriad of different texture types, too many to be able build a library of pre-modelled textures such that discriminant analysis maybe performed on an arbitrary image.

We present a new approach to this problem by using our multiscale nonparametric MRF model to model just one source texture from which we map the probability over an image of a pixel being of that source texture. We have shown that our MRF model, assessed by

human vision, is able to synthesise highly representative textures. On this basis, we believe that the model captures sufficient unique textural characteristics to identify the probability of any pixel in an image as being of the modelled texture, without the use of discriminant analysis. This capacity permits segmentation and classification of images with undefined texture types, *i.e.*, it permits unsupervised texture recognition.

For cases when x is not all of the one texture, Geman and Graffigne [23] classified small areas of the image assuming they were homogeneous. The classification was made on the basis that the product of the neighbourhood probabilities over the area of concern resembled the joint probability for that area, *i.e.*,

$$\Pi(x_r, r \in W_s) \simeq \prod_{\mathcal{N}_r \subseteq W_s} P(x_r, x_t, t \in \mathcal{N}_r), \quad (32)$$

where W_s is the window of sites, centred at s , which are to be used for the classification of x_s .

A. Probability measurement

We found the probability $\Pi(x_r, r \in W_s)$ as defined by equation (32) to be unwieldy. This was because our nonparametric LCPDF tended to give low probabilities for the neighbourhood configurations in the classification window, which resulted in $\Pi(x_r, r \in W_s)$ being too susceptible to minor fluctuations in these neighbourhood probabilities. Instead, we used the set of probabilities defined by the LCPDF for the window W_s , and compared them directly to the set of probabilities from the source texture.

The probability $P(x_r, x_t, t \in \mathcal{N}_r)$ is calculated from equation (12) as

$$P(x_r, x_t, t \in \mathcal{N}_r) = \frac{1}{nh^d(2\pi)^{d/2}} \sum_{\substack{p \in S_y, \\ \mathcal{N}_p \subseteq S_y}} \exp \left[-\frac{1}{2h^2} (\mathbf{z} - \mathbf{Z}_p)^T (\mathbf{z} - \mathbf{Z}_p) \right], \quad (33)$$

where $\mathbf{z} = [x_r, x_t, t \in \mathcal{N}_r]^T$ and \mathbf{Z}_p are samples taken from the source texture y defined on the set of sites S_y . Note, the pixel temperature t_s , $s \in S$ is not used, and the LCPDF is not normalised. For the strong nonparametric MRF model, we use the simple estimate of equation (25), where each $P(x_r, x_t, t \in C)$ is calculated in the same way as $P(x_r, x_t, t \in \mathcal{N}_r)$ in equation (33), but based on the clique C rather than the neighbourhood \mathcal{N}_r . The set of cliques is dependent on the neighbourhood size and maximum statistical order requested [38].

The samples of the LCPDF, taken from the window $W_s \subset S$, are the set of probabilities $\{P(x_r, x_t, t \in \mathcal{N}_r), \mathcal{N}_r \subseteq W_s\}$. We compare these samples directly with the set of probabilities obtainable from the source texture y . For every site $q \in S_y$, $\mathcal{N}_q \subset S_y$ in the source texture, a vector $\mathbf{z} = [y_q, y_t, t \in \mathcal{N}_q]^T$ maybe used to calculate $P(y_q, y_t, t \in \mathcal{N}_q)$ via equation (33), where again the sample vectors \mathbf{Z}_p are from the source texture y . However, the probability $P(y_q, y_t, t \in \mathcal{N}_q)$ is then biased since the sample $\mathbf{Z}_p = \mathbf{z}$ is included in the estimation. This bias is removed by excluding that site $p = q$ from the calculation of $P(y_q, y_t, t \in \mathcal{N}_q)$. The set of probabilities $\{P(y_q, y_t, t \in \mathcal{N}_q), \mathcal{N}_q \subset S_y\}$ from the source texture y is therefore calculated using the modified formula,

$$P(y_q, y_t, t \in \mathcal{N}_q) = \frac{1}{nh^d(2\pi)^{d/2}} \sum_{\substack{p \in S_y - q, \\ \mathcal{N}_p \subseteq S_y}} \exp \left[-\frac{1}{2h^2} (\mathbf{z} - \mathbf{Z}_p)^T (\mathbf{z} - \mathbf{Z}_p) \right]. \quad (34)$$

Given the set of probabilities $\{P(x_r, x_t, t \in \mathcal{N}_r), \mathcal{N}_r \subseteq W_s\}$ from the window to be classified, and the set of probabilities $\{P(y_q, y_t, t \in \mathcal{N}_q), q \in S_y, \mathcal{N}_q \subset S_y\}$ from the source texture, we can now determine the recognition probability. The null hypothesis is that

the distribution of probabilities from the window is the same as the distribution from the source texture. We use the nonparametric Kruskal-Wallis test [33] to test the hypothesis.

The Kruskal-Wallis test is the nonparametric version of the F test [33]. We use the Kruskal-Wallis test to make inferences about treatment populations, accepting or rejecting the null hypothesis that the populations come from the same distribution, primarily by comparing the means. The Kruskal-Wallis statistic K is calculated in terms of the *ranks* of the observations rather than their nominal values. Given c populations, each with n_j observations, and sum of ranks equal to T_j , the Kruskal-Wallis statistic K is expressed as

$$K = \frac{12}{n_T(n_T + 1)} \sum_{j=1}^c \left(\frac{T_j^2}{n_j} \right) - 3(n_T + 1), \quad (35)$$

where n_T is the sum of the sample sizes.

The sampling distribution of K is approximately chi-squared with $c - 1$ degrees of freedom. The accepted practice is to reject the null hypothesis if K is greater than a particular confidence level α . Our approach is to instead calculate the *confidence* associated with accepting the null hypothesis. This confidence P_{W_s} for a particular window W_s , is

$$P_{W_s} = P(k \geq K), \quad (36)$$

where K is calculated from equation (35) and k is chi-squared-distributed with one degree of freedom. We use this confidence P_{W_s} to plot our probability map.

The multiscale segmentation and recognition algorithm differs from the multiscale synthesis algorithm, Fig. 3, because we do not constrain the calculation of P_{W_s} for each grid level. Instead, we simply label the site s with the combined Kruskal-Wallis probabilities P_{W_s} obtained for each grid level over each site. Since we do not use the Gidas constraint [24] for segmentation and recognition we may use the alternative multigrid representation of *local averaging*. We express the labelling of the site $s \in S$ with the combined Kruskal-Wallis probabilities,

$$P_{W_{s=(i,j)}} = \prod_{l \geq 0} P_{W_r^l}, \quad r = \left(\frac{i}{2^l}, \frac{j}{2^l} \right) \in S^l, \quad (37)$$

VII. SEGMENTED AND CLASSIFIED TEXTURES

To prove the performance of our segmentation/recognition algorithm, Fig. 5, we tested it on images containing a mosaic of sub-images with similar grey levels (see Fig. 6(a) (b)). A conventional application of a (first order) histogram technique would not be able to segment these. Also a mix of structured and stochastic sub images were chosen to illustrate how our non parametric technique is able to recognise all types of textures.

The results of the unsupervised segmentation/classification of Fig. 6(a), with respect to the source textures Figs. 6(.1) (.2) and (.3), are shown as the probability maps Figs. 6(a.1) (a.2) and (a.3), respectively. Fig. 6(b) was segmented and classified with respect to the source textures Figs. 6(.4) (.5) and (.6), and the resulting probability maps are Figs. 6(b.4) (b.5) and (b.6), respectively. The probability maps are grey scale images, with white (grey level 255) representing a probability of one, and black (grey level 0) representing zero probability. Only one source texture was used at any one time to form the probability maps.

The probability maps of Fig. 6 show that with the appropriate texture model it is possible to segment and recognise windows of texture with respect to just one source texture and without prior knowledge of the other types of textures in the image. For these maps we used the strong MRF model with a 3×3 neighbourhood and pairwise cliques, as this was identified as being the optimal model for recognition, Table VII. The maps show that the

TABLE I

PERCENTAGE ERROR FOR UNSUPERVISED TEXTURE RECOGNITION OF 100 VISTEX TEXTURE MOSAICS
 = PERCENTAGE AREA OF FALSE NEGATIVES + PERCENTAGE AREA OF FALSE POSITIVES. VISTEX
 TEXTURE MOSAICS COURTESY OF COMPUTER VISION GROUP, COMPUTER SCIENCE III, UNIVERSITY
 BONN; AND VISION TEXTURE ARCHIVE OF THE MIT MEDIA LAB

<i>Quadtree Height</i>	<i>Square Neighbourhood Size</i>	<i>Clique Size</i>	<i>Percentage Error</i>
0	3×3	-	24.04
0	3×3	2	15.67
0	3×3	3	23.70
0	5×5	-	25.54
0	5×5	2	14.69
0	5×5	3	21.45
1	3×3	-	19.45
1	3×3	2	$\Rightarrow 12.94 \Leftarrow$
1	3×3	3	18.58
1	5×5	-	24.38
1	5×5	2	13.48
1	5×5	3	18.74
2	3×3	-	18.40
2	3×3	2	13.85
2	3×3	3	17.62
2	5×5	-	23.98
2	5×5	2	15.22
2	5×5	3	19.46
3	3×3	-	21.79
3	3×3	2	18.33
3	3×3	3	21.80
3	5×5	-	30.33
3	5×5	2	21.55
3	5×5	3	25.48

segmentation/recognition algorithm, Fig. 5, is able to discriminate between textures similar to the source texture and those that are dissimilar.

VIII. PRACTICAL APPLICATION

The final goal of this research was to produce a method by which an operator may be able to take a radar satellite image; segment a small portion from the image where the terrain was known; use this as the source to a texture model; then with respect to the texture model, find where other similar terrain types lie within the image. Such a method of unsupervised segmentation and unsupervised recognition would be ideal for terrain mapping of Synthetic Aperture Radar (SAR) images, as it does not require a complete library of textures as for discriminant analysis. With our method, any operator may choose which type of texture they wish to model, without the need for a pre-modelled version existing as part of a library. The nonparametric MRF model is suited to this type of approach as there is no exhaustive training required to match the model to the texture. However, as the probability maps are pre-normalised, best results maybe obtained if the user was to iteratively update the probability maps for each source texture used.

The practical application of segmenting and classifying a SAR image of Cultana, Fig. 7, shows the two results if: 1) the operator chose a 64×64 patch of trees from the bottom

left corner, Fig. 8; or 2) the operator chose a 64×64 patch of grass from the bottom right corner, Fig. 9. Again we have used the same strong MRF texture model as applied in the unsupervised segmentation and recognition of the texture mosaics Fig. 6. In both cases the resulting probability maps have been superimposed on top of the original SAR image. This gives a clear indication of how the segmentation/recognition algorithm has performed. The results show the feasibility of such an approach to unsupervised segmentation and unsupervised recognition of texture for terrain mapping of SAR images.

IX. SUMMARY AND CONCLUSION

The nonparametric Markov random field (MRF) model has been shown to synthesise representative examples of both structured and stochastic texture. With the multiscale texture synthesis incorporating our novel pixel temperature function, we were able to use this model to synthesise realistic realisations of a source texture with minimal phase discontinuities. It was with this evidence that we concluded that the nonparametric MRF model had captured all the unique characteristics specific to a particular texture. It was considered that with such a model it would be feasible to recognise other similar texture from an image containing multiple unknown textures. The model was used to classify those image segments that had similar textural characteristics to a source texture, thereby performing unsupervised texture recognition without prior knowledge of other textures types present in the image. Such a technique was considered valuable to the practical application of terrain mapping of SAR images.

A second nonparametric MRF model was proposed, one that was based on the strong MRF model of Moussouris [36]. This model was shown to be equivalent to the ANOVA construction [18], from which we were able to derive the general ANOVA construction equation (22). The strong MRF model was also shown to be able to synthesise representative versions of a source texture. With this model we were able to limit the statistical order required to uniquely represent a texture, thereby increasing the entropy of the LCPDF, which in turn produced a better classification model for unsupervised texture recognition.

The advantage of this unsupervised recognition technique was that it required virtually no training of the texture models, thereby allowing an end user to specify their own type of texture to segment and classify on an undetermined image. Also the resulting probability maps are pre-normalised, allowing the end user to iteratively improve the probability map with further results from the unsupervised recognition of other source textures. Although the choice of the strong MRF model should be based on the synthesis results, the texture synthesis algorithm, Fig. 3, is computationally intensive. However we found the strong MRF model with a 3×3 square neighbourhood and second order cliques was a good general classification model. As a second order model was found to be the most versatile, in future work it might be worthwhile to change the model used in the unsupervised texture recognition to another type of second order model, *e.g.*, like those models based on the stochastic modelling of various multi-resolution filter responses [5], [45], [28], [37].

APPENDIX

I. PROOF 1 OF PROPOSITION 1

This proof is based on the Grimmett [25] and Moussouris [36] equivalence proof for a standard MRF and a Gibbs distribution. Following the layout presented by Geman [20], we show that for a strong MRF, Π is a Gibbs distribution with respect to the strong \mathcal{N} -potential,

$$V_A(x) = \sum_{B \subseteq A} (-1)^{|A|-|B|} \log P_B(x), \quad \forall x \in \Omega. \quad (38)$$

Moreover, for any element $s \in A$,

$$V_A(x) = \sum_{B \subseteq A, s \in B} (-1)^{|A|-|B|} \log P_B(x_s | x_{(s)}), \quad \forall x \in \Omega. \quad (39)$$

This representation is unique among normalised potentials.

1. Π is Gibbs w.r.t. V : Assuming equation (38) and using the Möbius inversion formula (4),

$$P_A(x) = \exp \sum_{B \subseteq A} V_B(x), \quad (40)$$

where A is any subset $A \subseteq S$. This is the second condition imposed on the two functions P and V . The first condition is implied from equation (16) for which, given two subsets A and B of S such that $B \subseteq A \subseteq S$, then

$$\begin{aligned} \log P_B(x) &= \log \sum_{\substack{y^A \in \Omega, \\ y^B = x^B}} P_A(y) = \log \sum_{\substack{y^A \in \Omega, \\ y^B = x^B}} \exp \sum_{C \subseteq A} V_C(y) \\ &= \sum_{C \subseteq B} V_C(x) + \log \sum_{\substack{y^A \in \Omega, \\ y^B = x^B}} \exp \left[\sum_{\substack{C \subseteq A, \\ C \not\subseteq B}} V_C(y) \right]. \end{aligned} \quad (41)$$

Now, since equation (40) is a function defined for all $A \subseteq S$,

$$\log P_B(x) = \sum_{C \subseteq B} V_C(x), \quad (42)$$

which implies equation (40) and (41) are true if the condition,

$$\sum_{\substack{y^A \in \Omega, \\ y^B = x^B}} \exp \left[\sum_{\substack{C \subseteq A, \\ C \not\subseteq B}} V_C(y) \right] = 1 \quad \forall x \in \Omega, \quad (43)$$

holds for $B \subseteq A \subseteq S$.

2. V is normalised: The potential V , defined by equation (38), is not normalised in the conventional manner. In the original proof for a standard MRF, the potentials were said to be normalised if $V_A(x) = 0$ if $x_s = 0$ for any $s \in A$. For the potential defined by equation (38), this is not the case. However,

$$V_\emptyset(x) = \sum_{B \subseteq \emptyset} (-1)^{|\emptyset|-|B|} \log P_B(x) = \log P_\emptyset(x) = \log 1 = 0. \quad (44)$$

Therefore, although the potential V is not normalised in the conventional manner, there is a degree of uniformity in the construction of the potential functions.

3. (38) \Leftrightarrow (39): For any $s \in A$,

$$\begin{aligned} V_A(x) &= \sum_{B \subseteq A, s \in B} (-1)^{|A|-|B|} \log P_B(x) + \sum_{B \subseteq A, s \notin B} (-1)^{|A|-|B|} \log P_B(x) \\ &= \sum_{B \subseteq A, s \in B} (-1)^{|A|-|B|} (\log P_B(x) - \log P_{B-s}(x)) \\ &= \sum_{B \subseteq A, s \in B} (-1)^{|A|-|B|} \log P_B(x_s | x_{(s)}). \end{aligned} \quad (45)$$

4. V is a strong \mathcal{N} -potential: Given that x is defined on a strong MRF with respect to \mathcal{N} , then V is a strong \mathcal{N} -potential if $V_A(x) = 0 \forall A \notin \mathcal{C}$.

We choose $A \notin \mathcal{C}$. Then $\exists s, t \in A$ such that $t \notin \mathcal{N}_s \Leftrightarrow s \notin \mathcal{N}_t$.

$$\begin{aligned}
 V_A(x) &= \sum_{B \subseteq A} (-1)^{|A|-|B|} \log P_B(x) \\
 &= \sum_{B \subseteq A-s-t} (-1)^{|A|-|B|} \log P_B(x) + \sum_{B \subseteq A-s-t} (-1)^{|A|-|B+s|} \log P_{B+s}(x) \\
 &+ \sum_{B \subseteq A-s-t} (-1)^{|A|-|B+t|} \log P_{B+t}(x) + \sum_{B \subseteq A-s-t} (-1)^{|A|-|B+s+t|} \log P_{B+s+t}(x) \\
 &= \sum_{B \subseteq A-s-t} (-1)^{|A|-|B|} \log \left[\frac{P_B(x) P_{B+s+t}(x)}{P_{B+s}(x) P_{B+t}(x)} \right] \\
 &= 0,
 \end{aligned} \tag{46}$$

from identity (18) for a strong MRF. Therefore,

$$\log P_A(x) = \sum_{\substack{C \subseteq A, \\ C \in \mathcal{C}}} V_C(x). \tag{47}$$

This means that for a strong MRF, the Gibbs distribution Π may be expressed with respect to \mathcal{N} -potentials (38) or (39). □

The first part of Proposition 1, the probability decomposition of equation (20) is proved by applying Möbius inversion formula (5) to equation (38) over the set of sites $\{s, r \in \mathcal{N}_s\}$:

$$\log P(x_s, x_r, r \in \mathcal{N}_s) = \sum_{C \subseteq \mathcal{N}_s+s} \sum_{C' \subseteq C} (-1)^{|C|-|C'|} \log P_{C'}(x). \tag{48}$$

As the probability decomposition is over the set of cliques defined on the sites $\{s, r \in \mathcal{N}_s\}$, any clique $C' \subseteq \mathcal{N}$ is always contained in a larger clique $C, s \in C$ for which $C' \subset C$. It is easy to prove that all the probabilities defined on cliques C' that do not contain the site s are cancelled out in the probability decomposition summation (48) giving

$$\log P(x_s, x_r, r \in \mathcal{N}_s) = \sum_{C \in \mathcal{C}_s} \sum_{\substack{C' \subseteq C, \\ s \in C'}} (-1)^{|C|-|C'|} \log P_{C'}(x). \tag{49}$$

However, this is only true if the cliques are *not* arbitrarily restricted to being of a certain type (*i.e.*, say only those cliques $C \in \mathcal{C}_s$ for which $|C| \leq 3$). Otherwise only equation (48) is valid.

The probability decomposition of equation (21), is proved by applying the Möbius inversion formula (5) to equation (39) giving

$$\log P(x_s | x_r, r \in \mathcal{N}_s) = \sum_{C \in \mathcal{C}_s} \sum_{\substack{C' \subseteq C, \\ s \in C'}} (-1)^{|C|-|C'|} \log P_{C'}(x_s | x_{(s)}) \tag{50}$$

□

II. PROOF 2 OF PROPOSITION 1

This proof is based on the ANOVA construction [18] for testing independence in a distribution. In ANOVA-type notation, the probability $P(x_s, x_r, r \in \mathcal{N}_s)$ is decomposed into the *general log-linear model* [18, p 29]:

$$\log P(x_s, x_r, r \in \mathcal{N}_s) = \log P_A(x) = \sum_{B \subseteq A} U_B(x). \quad (51)$$

We now make a slight change in notation to make the proceeding expressions easier to comprehend. First, let $P_A(x) = P(x_s, x_r, r \in \mathcal{N}_s)$, where A is specifically defined as $A = \{s, r \in \mathcal{N}_s\}$, and then redefine x to be on the smaller lattice A rather than S , such that

$$x \in \Omega_A, \text{ where } \Omega_A = \Lambda^{|A|}. \quad (52)$$

In the ANOVA log-linear model, $\sum_{C \subseteq B} U_C(x)$ represents the mean of the logarithmic probabilities $\log P_A(y)$, $y \in \Omega_A$ for which $y^B = x^B$:

$$\sum_{C \subseteq B} U_C(x) = \sum_{\substack{y \in \Omega_A, \\ y^B = x^B}} \frac{\log P_A(y)}{|\Omega_A|}. \quad (53)$$

Then, U_\emptyset is the grand mean of the logarithmic probabilities:

$$U_\emptyset(x) = \sum_{y \in \Omega_A} \frac{\log P_A(y)}{|\Omega_A|}. \quad (54)$$

Note that the value of $U_\emptyset \neq V_\emptyset$ from equation (44).

From equation (53), it is apparent that $U_B(x)$ represents the deviation from the mean $\sum_{C \subseteq B} U_C(x)$ for those $y \in \Omega_A$ for which $y^B = x^B$. Therefore,

$$\sum_{x_s \in \Lambda} U_B(x_s, x_{(s)}) = 0 \quad \forall s \in B. \quad (55)$$

Given that x is defined on a strong MRF with respect to \mathcal{N} , then U is a strong \mathcal{N} -potential if $U_B(x) = 0 \quad \forall B \notin \mathcal{C}$. For $B \notin \mathcal{C} \exists s, t \in B$ such that $t \notin \mathcal{N}_s \Leftrightarrow s \notin \mathcal{N}_t$.

$$\begin{aligned} U_B(x) &= \sum_{\substack{y \in \Omega_A, \\ y^B = x^B}} \frac{\log P_A(y)}{|\Omega_A|} - \sum_{C \subseteq B} U_C(x) \\ &= \sum_{\substack{y \in \Omega_A, \\ y^B = x^B}} \frac{\log P_A(y)}{|\Omega_A|} - \sum_{C \subseteq B-s} U_C(x) - \sum_{C \subseteq B-t} U_C(x) + \sum_{C \subseteq B-s-t} U_C(x) \\ &= \sum_{\substack{y \in \Omega_A, \\ y^B = x^B}} \frac{\log P_A(y)}{|\Omega_A|} - \sum_{\substack{y \in \Omega_A, \\ y^{B-s} = x^{B-s}}} \frac{\log P_A(y)}{|\Omega_A|} \\ &\quad - \sum_{\substack{y \in \Omega_A, \\ y^{B-t} = x^{B-t}}} \frac{\log P_A(y)}{|\Omega_A|} + \sum_{\substack{y \in \Omega_A, \\ y^{B-s-t} = x^{B-s-t}}} \frac{\log P_A(y)}{|\Omega_A|} \\ &= \frac{1}{|\Omega_A|} \sum_{\substack{y \in \Omega_A, \\ y^{B-s-t} = x^{B-s-t}}} \log \left[\frac{P_A(x_s, x_t, y_{(s,t)}) P_A(y)}{P_A(x_s, y_{(s)}) P_A(x_t, y_{(t)})} \right] \end{aligned}$$

$$\begin{aligned}
&= \frac{1}{|\Omega_A|} \sum_{\substack{y \in \Omega_A, \\ y^{B-s-t} = x^{B-s-t}}} \log \left[\frac{P(x_s|x_t, y_{(s,t)})P(x_t, y_{(s,t)})P(y_s|y_{(s)})P(y_{(s)})}{P(x_s|y_{(s)})P(y_{(s)})P(y_s|x_t, y_{(s,t)})P(x_t, y_{(s,t)})} \right] \\
&= 0
\end{aligned} \tag{56}$$

From equation (56), $U_B(x) = 0 \forall B \notin \mathcal{C}$ since, for a strong MRF, $P(x_s|x_t, y_{(s,t)}) = P(x_s|y_{(s)})$ and, similarly, $P(y_s|x_t, y_{(s,t)}) = P(y_s|y_{(s)})$ from the strong MRF identity (18). Therefore, for a strong MRF, the ANOVA log-linear model may be rewritten as

$$\log P(x_s, x_r, r \in \mathcal{N}_s) = \log P_A = \sum_{\substack{C \subseteq A, \\ C \in \mathcal{C}}} U_C(x). \tag{57}$$

The ANOVA log-linear model, like the strong MRF model, specifies the probability $P_A(x)$ as being constructed from potentials defined on interacting subsets $C \subseteq A$, $C \in \mathcal{C}$. However, in the ANOVA log-linear model, the functions $U_C(x)$ are not potentials, but represent successive deviations from the mean. The similarity of the two models is that both build their estimate of the probability $P_A(x)$ from information obtained from the same interacting subsets $C \subseteq A$, $C \in \mathcal{C}$.

Comparing equations (47) and (57), both the strong MRF model and the ANOVA log-linear model restrict the construction of $P_A(x)$ over functions defined on cliques. However, the functions themselves are not the same. This is evident when the Möbius inversion formula (4) is applied to equation (53) to obtain

$$U_C(x) = \sum_{C' \subseteq C} (-1)^{|C|-|C'|} \sum_{\substack{y \in \Omega_A, \\ y^{C'} = x^{C'}}} \frac{\log P_A(y)}{|\Omega_A|}. \tag{58}$$

Recalling from equation (41) that for any set of functions U where equation (57) is true,

$$\log P_B(x) = \sum_{C \subseteq B} U_C(x) + \log \sum_{\substack{y \in \Omega_A, \\ y^B = x^B}} \exp \left[\sum_{\substack{C \subseteq A, \\ C \not\subseteq B}} U_C(y) \right]. \tag{59}$$

Again, if condition (43) holds, and substituting equation (53) into (59),

$$\log P_B(x) = \sum_{C \subseteq B} U_C(x) = \sum_{\substack{y \in \Omega_A, \\ y^B = x^B}} \frac{\log P_A(y)}{|\Omega_A|}. \tag{60}$$

Therefore equation (58) may be rewritten as

$$U_C(x) = \sum_{C' \subseteq C} (-1)^{|C|-|C'|} \log P_{C'}(x). \tag{61}$$

Substituting equation (61) into (57), the ANOVA log-linear model forms the same probability decomposition as for a strong MRF:

$$\log P_A(x) = \sum_{\substack{C \subseteq A, \\ C \in \mathcal{C}}} \sum_{C' \subseteq C} (-1)^{|C|-|C'|} \log P_{C'}(x). \tag{62}$$

Since equation (62) is equivalent to (48), the rest of the proof follows for equation (21) and (20) as by the first proof of Proposition 1. \square

ACKNOWLEDGEMENT

The authors would like to thank the University of Melbourne for the long computer times on the MasPar. Also Project Ingara, Microwave Radar Division, Electronics and Surveillance Research Laboratory at DSTO for providing the Airborne SAR image of Cultana. The authors would also like to thank the following people for their contribution to research: Christine Graffigne, from the Université de Paris-Sud, for advise on Multiscale analysis; Garry Newsam and David Howard, from the Centre for Sensor Signal and Information Processing (CSSIP), for advice on Markov random fields; Andrew Bradley and Andrew Mehnert, from CSSIP, for advise on nonparametric statistical classification.

REFERENCES

- [1] Michele Basseville, Albert Benveniste, Kenneth C. Chou, Stuart A. Golden, Ramine Nikoukhah, and Alan S. Willsky, "Modeling estimation of multiresolution stochastic processes," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 766–784, 1992.
- [2] Julian E. Besag, "Spatial interaction and the statistical analysis of lattice systems," *Journal of the Royal Statistical Society, series B*, vol. 36, pp. 192–326, 1974.
- [3] Julian E. Besag, "On the statistical analysis of dirty pictures," *Journal Of The Royal Statistical Society*, vol. B–48, pp. 259–302, 1986.
- [4] Yvonne M. M. Bishop, Stephen E. Fienberg, and Paul W. Holland, *Discrete Multivariate Analysis: Theory and Practice*, Cambridge: MIT Press, 1975.
- [5] J. S. De Bonet, "Multiresolution sampling procedure for analysis and synthesis of texture images," in *Computer Graphics*. ACM SIGGRAPH, 1997, pp. 361–368, <http://www.ai.mit.edu/~jsd>.
- [6] Charles Bouman and Bede Liu, "Multiple resolution segmentation of textured images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 2, pp. 99–113, 1991.
- [7] Charles A. Bouman and Michael Shapiro, "A multiscale random field model for Bayesian image segmentation," *IEEE Transactions on Image Processing*, vol. 3, no. 2, pp. 162–177, 1994.
- [8] P. Brodatz, *Textures – a photographic album for artists and designers*, Dover Publications Inc., New York, 1966.
- [9] Shankar Chatterjee, "Classification of natural textures using Gaussian Markov random field models," in *Markov Random Fields Theory And Application*, Rama Chellappa and Anil Jain, Eds., pp. 159–177. Academic Press, Boston, Sydney, 1993.
- [10] Rama Chellappa and Shankar Chatterjee, "Classification of textures using Gaussian Markov random fields," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP–33, no. 4, pp. 959–963, 1985.
- [11] R. Chellappa and R. L. Kashyap, "Texture synthesis using 2–D noncausal autoregressive models," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP–33, no. 1, pp. 194–203, 1985.
- [12] R. Chellappa, R. L. Kashyap, and B. S. Manjunath, "Model-based texture segmentation and classification," in *Handbook of Pattern Recognition and Computer Vision*, C. H. Chen, L. F. Pau, and P. S. P. Wang, Eds., pp. 277–310. World Scientific, Singapore, 1993.
- [13] Chaur-Chin Chen, *Markov random fields in image processing*, Ph.D. thesis, Michigan State University, 1988.
- [14] G. C. Cross and A. K. Jain, "Markov random field texture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 5, pp. 25–39, 1983.
- [15] Haluk Derin and Howard Elliott, "Modelling and segmentation of noisy textured images using Gibbs random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI–9, no. 1, pp. 39–55, 1987.
- [16] Haluk Derin and Chee-Sun Won, "A parallel image segmentation algorithm using relaxation with varying neighbourhoods and its mapping to array processors," *Computer Vision, Graphics, and Image Processing*, vol. 40, pp. 54–78, 1987.
- [17] Richard O. Duda and Peter E. Hart, *Pattern Classification and Scene Analysis*, John Wiley, Stanford Research Institute, Menlo Park, California, 1973.
- [18] Stephen E. Fienberg, *The Analysis of Cross-Classified Categorical Data*, vol. Second Edition, MIT Press, 1981.
- [19] Andre Gagalowicz, Song De Ma, and C. Tournier-Lasserre, "Third order model for non homogeneous natural textures," in *Proceedings 8th International Conference on Pattern Recognition (ICPR)*, Paris, France, 1986, vol. 1, pp. 409–411.
- [20] D. Geman, "Random fields and inverse problems in imaging," in *Lecture Notes in Mathematics*, vol. 1427, pp. 113–193. Springer-Verlag, 1991.
- [21] Donald Geman, Stuart Geman, Christine Graffigne, and Ping Dong, "Boundary detection by constrained optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 609–628, 1990.
- [22] Stuart Geman and Donald Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 721–741, 1984.
- [23] S. Geman and C. Graffigne, "Markov random field image models and their applications to computer vision," *Proceedings of the International Congress of Mathematicians*, pp. 1496–1517, 1986.
- [24] Basilis Gidas, "A renormalization group approach to image processing problems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 2, pp. 164–180, 1989.
- [25] G. R. Grimmett, "A theorem about random fields," *Bulletin of the London Mathematical Society*, vol. 5, pp. 81–84, 1973.

- [26] Michal Haindl, "Texture synthesis," *CWI Quarterly*, vol. 4, pp. 305–331, 1991.
- [27] J. M. Hammersley and P. Clifford, "Markov fields on finite graphs and lattices," unpublished, 1971.
- [28] David J. Heeger and James R. Bergen, "Pyramid-based texture analysis/synthesis," in *Proceedings ICIP-95: 1995 International Conference on Image Processing*, Washington, D.C., 1995, pp. 648–651.
- [29] R. Hu and M. M. Fahmy, "Texture segmentation based on a hierarchical Markov random field model," *Signal processing*, vol. 26, no. 3, pp. 285–305, 1992.
- [30] Bela Julesz, "Visual pattern discrimination," *IRE transactions on Information Theory*, vol. 8, pp. 84–92, 1962.
- [31] R. L. Kashyap, "Characterization and estimation of two-dimensional ARMA models," *IEEE Transactions on Information Theory*, vol. 30, pp. 736–745, 1984.
- [32] Zoltan Kato, Josiane Zerubia, and Marc Berthod, "Unsupervised parallel image classification using a hierarchical Markovian model," in *1995 IEEE 5th International Conference on Computer Vision*, Bob Werner, Ed., Los Alamitos, 1995, pp. 169–174, IEEE Computer Society Press.
- [33] Lawrence L. Lapin, *Probability and Statistics for Modern Engineering*, PWS-KENT, Boston, 1990.
- [34] Mark R. Luetzgen, William C. Karl, Alan S. Willsky, and Robert R. Tenny, "Multiscale representation of Markov random fields," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3377–3396, 1993.
- [35] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equations of state calculations by fast computing machines," *Journal of Chemical Physics*, vol. 21, pp. 1087–1091, 1953.
- [36] John Moussouris, "Gibbs and Markov random systems with constraints," *Journal of Statistical Physics*, vol. 10, no. 1, pp. 11–33, 1974.
- [37] Rafael Navarro and Javier Portilla, "Robust method for texture synthesis-by-analysis based on a multiscale Gabor scheme," in *SPIE Electronic Imaging Symposium, Human Vision and Electronic Imaging '96*, Bernice Rogowitz and Jan Allebach, Eds., San Jose, California, 1996, vol. 2657, pp. 86–97.
- [38] Rupert Paget and Dennis Longstaff, "Extracting the cliques from a neighbourhood system," *IEE Proceedings Vision, Image and Signal Processing*, vol. 144, no. 3, pp. 168–170, June 1997, <http://www.vision.ee.ethz.ch/~rpaget>.
- [39] Kris Popat and Rosalind W. Picard, "Novel cluster-based probability model for texture synthesis, classification, and compression," in *Proceedings SPIE visual Communications and Image Processing*, Boston, 1993.
- [40] P. P. Raghu, R. Poongodi, and B. Yegnanarayana, "Unsupervised texture classification using vector quantization and deterministic relaxation neural network," *IEEE Transactions on Image Processing*, vol. 6, no. 10, pp. 1376–1387, 1997.
- [41] G. C. Rota, "On the foundations of combinatorial theory," *Zeitschrift Fur Wahrscheinlichkeitstheorie Und Verwandte Gebiete*, vol. 2, pp. 340–368, 1964.
- [42] Lynne Seymour, *Parameter estimation and model selection in image analysis using Gibbs-Markov random fields*, Ph.D. thesis, The University of North Carolina, Chapel Hill, 1993.
- [43] B. W. Silverman, *Density estimation for statistics and data analysis*, Chapman and Hall, London, 1986.
- [44] Demetri Terzopoulos, "Image analysis using multigrid relaxation methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 2, pp. 129–139, 1986.
- [45] Song Chun Zhu, Yingnian Wu, and David Mumford, "FRAME: filters, random fields, rnd minimax entropy towards a unified theory for texture modeling," *Proceedings 1996 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 686–693, 1996.

Algorithm 1: *Nonparametric multiscale MRF texture synthesis***Input:**

$y \leftarrow$ source texture image
 $N_y \times M_y \leftarrow$ size of source image y
 $N_x \times M_x \leftarrow$ size of synthetic image x
 $o \leftarrow$ order of the neighbourhood system

Begin

1. Define number of grid levels M as, $M \leq 1 + \log_2 (\min \{N_x, M_x, N_y, M_y\})$.
 2. Define image x as being on a set of sites S as given by (27).
 3. Define the multigrid representation of image x as the set of subset of sites $S^l \subseteq S$ for $0 \leq l < M$ as given by (28).
 4. Similarly, define image y as being on a set of sites, S_y (27), with a multigrid representation as the set of subset of sites $S_y^l \subseteq S$ for $0 \leq l < M$ as given by (28).
 5. Initialise pixel temperatures $t_s = 1, \forall s \in S$.
 6. **For** $l = M - 1$ to 0 **do**
 - 6.1. Define neighbourhood \mathcal{N}_s^l w.r.t. order o as given by (29).
 - 6.2. **While** $t_s \neq 0, \forall s \in S^l$ **do**
 - 6.2.1. choose a set of sites $S_{i.i.d.} = \{s \in S, \mathcal{N}_s^l \cap S_{i.i.d.} = \emptyset, t_s > 0\}$.
 - 6.2.2. **For all** $s \in S_{i.i.d.}$ **in parallel do**
 - 6.2.2.1. Estimate the LCPDF for $x_s = \lambda, \forall \lambda \in \Lambda$ via (12), or (25) for the strong LCPDF. In both cases $(\mathbf{z} - \mathbf{Z}_p)$ is defined by (30).
 - 6.2.2.2. Choose new x_s by sampling its LCPDF, as in *Gibbs Sampler*, or choose the mode as in *ICM*.
 - 6.2.2.3. Update t_s via (31).
 - 6.2.3. **done**
 - 6.3. **done**
 7. **done**
- end**

Fig. 3. Parallel implementation of our nonparametric multiscale MRF texture synthesis algorithm

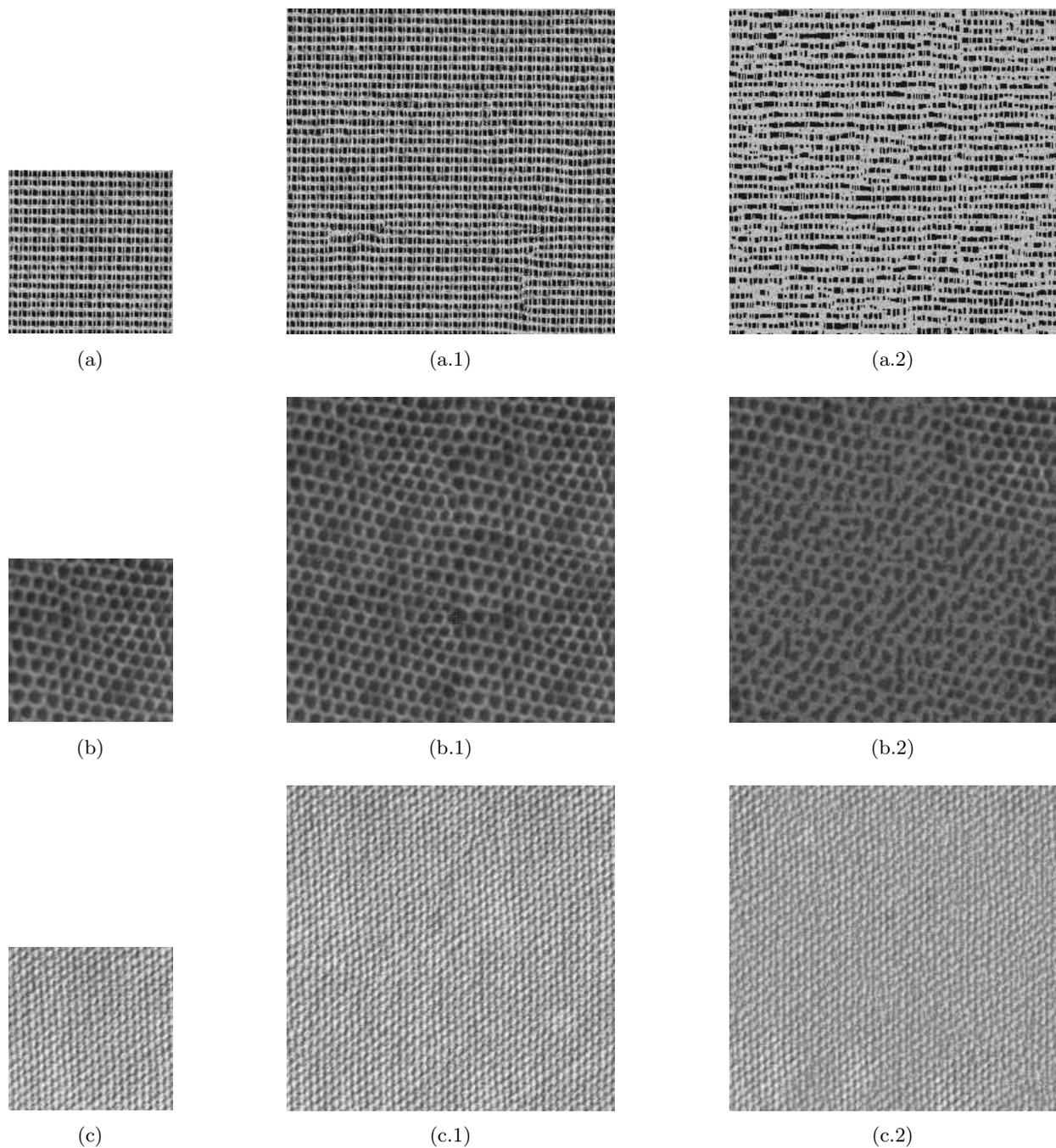


Fig. 4. Brodatz textures: (a) D21 - French canvas; (b) D22 - Reptile skin; (c) D77 - Cotton canvas; (?1) synthesised textures - MRF model, neighbourhood order $o = 18$; (?2) synthesised textures - strong MRF model, neighbourhood order $o = 8$ and limited to 3rd order cliques.

Algorithm 2: *Nonparametric multiscale MRF texture segmentation/recognition***Input:**

$x \leftarrow$ input image
 $y \leftarrow$ source texture image
 $N_x \times M_x \leftarrow$ size of input image x
 $N_y \times M_y \leftarrow$ size of source image y
 $o \leftarrow$ order of the neighbourhood system
 $W_s \leftarrow$ classification window

begin

1. Define number of grid levels M as, $M \leq 1 + \log_2 (\min \{N_x, M_x, N_y, M_y\})$.
 2. Define probability map x^L and image x as being on a set of sites S as given by (27).
 3. Define the multigrid representation of the image, x , as the set of images, $x^l = \{x_s^l = (x_{2i,2j}^{l-1} + x_{2i+1,2j}^{l-1} + x_{2i,2j+1}^{l-1} + x_{2i+1,2j+1}^{l-1})/4, s = (i, j) \in S^l\}$, where $S^l = \{s = (i, j) : 0 \leq i < N_x/2^l, 0 \leq j < M_x/2^l\}$, for $0 \leq l < M$.
 4. Similarly, define the multigrid representation of the image, y , as the set of images, y^l on the sites S_y^l for grid levels $0 \leq l < M$.
 5. Define neighbourhood \mathcal{N} w.r.t. order o via (3).
 6. Initialise probability map to $x^L = \{x_s^L = 1, s \in S\}$.
 7. **For** $l = M - 1$ to 0 **do**
 - 7.1. Obtain the set of LCPDF samples $\{P^l(y_q^l, y_t^l, t \in \mathcal{N}_q), \mathcal{N}_q \subset S_y^l\}$ from the source image y^l via (34).
 - 7.2. **For all** $s \in S^l$ **in parallel do**
 - 7.2.1. Calculate $P^l(x_s^l, x_r^l, r \in \mathcal{N}_s)$ via (33).
 - 7.2.2. Calculate the Kruskal-Wallis Probability $P_{W_s}^l$ from (36) via (35).
 - 7.2.3. Correct edge and boundary effects by equating $P_{W_s}^l = \max_{r \in W_s} P_{W_r}^l$, $W_r \subset S^l$.
 - 7.2.4. Label all $x_r^L = x_r^L \cdot P_{W_s}^l$, for which $r = (p, q) \in S$, $s = (p/2^l, q/2^l)$.
 - 7.3. **done**
 8. **done**
- end**

Fig. 5. Parallel implementation of our nonparametric multiscale MRF texture segmentation/recognition algorithm

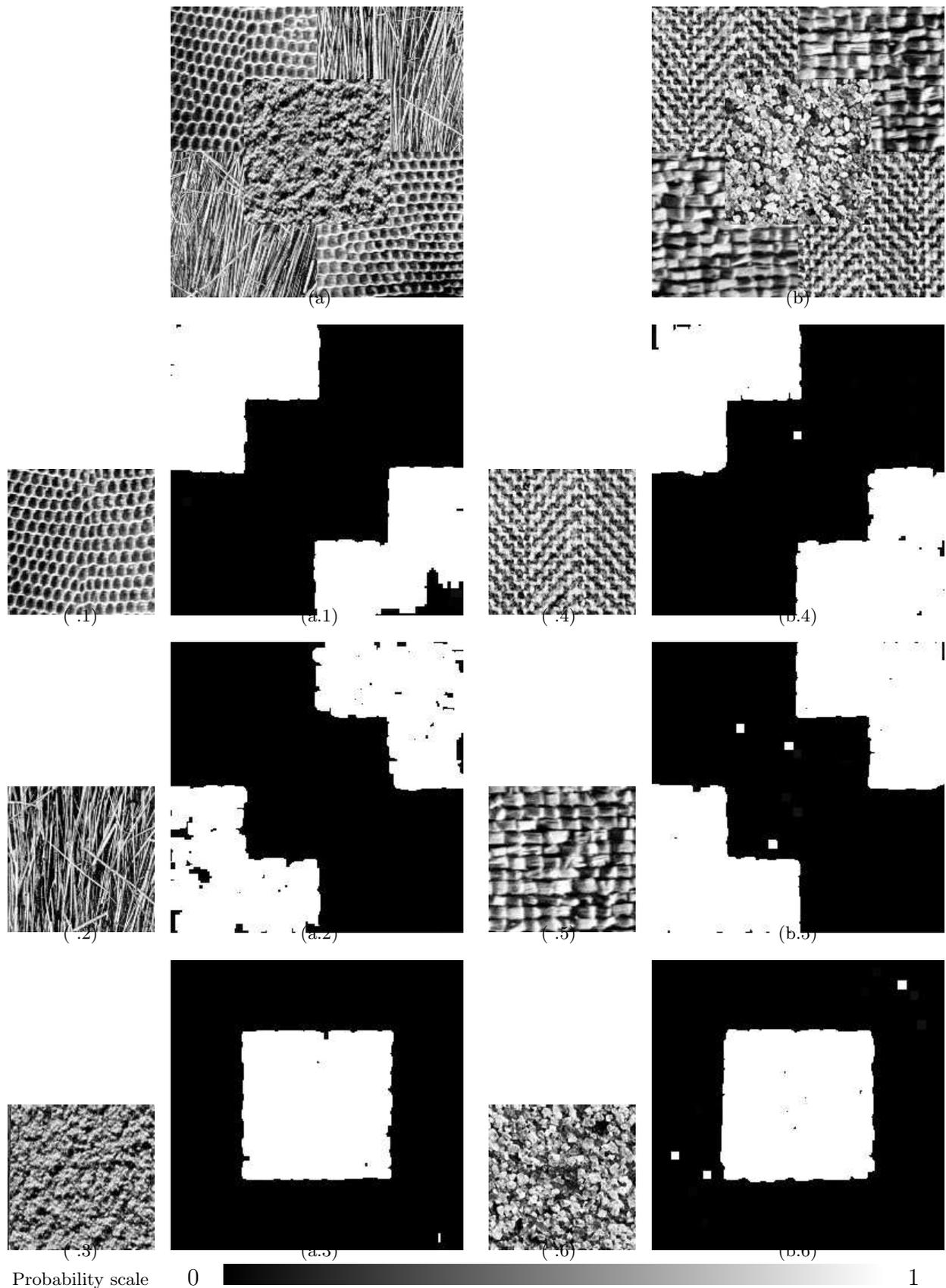


Fig. 6. Probability maps of Brodatz texture mosaics (a) and (b): (.1) D3 - Reptile skin; (.2) D15 - Straw; (.3) D57 - Handmade paper; (a.1) (a.2) (a.3) the respective probability maps of (a); (.4) D17 - Herringbone weave; (.5) D84 - Raffia; (.3) D29 - Beach sand; (b.4) (b.5) (b.6) the respective probability maps of (b).

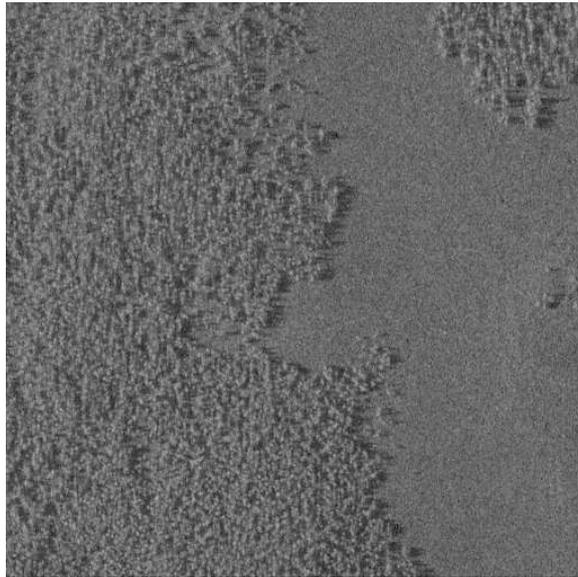


Fig. 7. Airborne SAR image of Cultana.

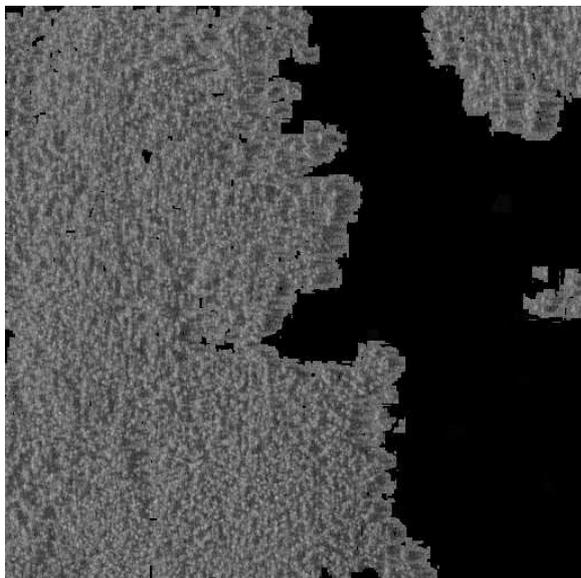


Fig. 8. Probability map of the trees superimposed on to Cultana image.

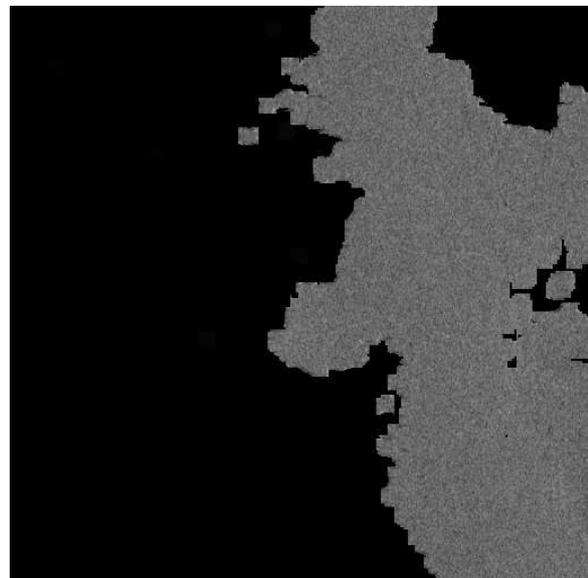


Fig. 9. Probability map of the grass superimposed on to Cultana image.