REFERENCES

- [1] R. Bracho and A.C. Sanderson "Segmentation of images based on intensity gradient informations," in *Proc. IEEE Computer Soc. Conf. Computer Vision and Pattern Recognition*, San Francisco, CA, June 19-23, 1985, pp. 341-347.
- [2] A.D. Brink, "Gray-level thresholding of images using a correlation criterion," *Pattern Recog., Lett.*, vol. 9, no. 5, pp. 335-341, June 1989.
- [3] K. Castleman and R. Wall, "Automatic systems for chromosome identification," in *Nobel Symp. 23—Chromosome Identification*. New York: Academic, 1973.
- [4] C.K. Chow and T. Kaneko, "Automatic boundary detection of the left ventricle from cineangiograms," *Comput. Biomed. Res.*, vol. 5, pp. 388-410, 1972.
- [5] R. Haralick, "Image segmentation survey," in *Fundamentals of Computer Vision*, O.D. Faugeras, Ed. London: Cambridge University Press, 1983.
- [6] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst. Man, Cybern.*, vol. 9, no. 1, pp. 377-393, 1979.
- [7] A. Perez and R.C. Gonzalez, "An iterative thresholding algorithm for image processing," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, no. 6, Nov. 1987.
- [8] W.K. Pratt, *Digital Image Processing*. New York: Wiley, 1978.
- [9] J. M.S. Prewitt, "Object enhancement and extraction," in *Picture Processing and Psychopictorics*, Lipkin and Rosenfeld, Eds. New York: Academic, 1970.
- [10] T.W. Ridler, S. Calvard, "Picture thresholding using an iterative selection method," *IEEE Trans. Syst. Man, Cybern.*, vol. SMC-8, no. 8, pp. 629-632, Aug. 1978.
- [11] T.G. Stockham, "Image processing in the context of a visual model," *Proc. IEEE*, vol. 60, pp. 828-842, 1972.
- [12] F.M. Wahl, K.Y. Wong, and R.G. Casey, "Block segmentation and text extraction in mixed text/image documents," *Comput. Graphics, Image Processing*, vol. 20, pp. 375-390, 1982.
- [13] R.J. Wall, "The gray level histogram for threshold boundary determination in image processing with applications to the scene segmentation problem in human chromosome analysis," Ph.D. dissertation, Univ. California, Los Angeles, 1974.
- [14] R. Wilson and M. Spann, *Image Segmentation and Uncertainty*. New York: Wiley, 1988.

Generating Connected Textured Fractal Patterns Using Markov Random Fields

Levent Onural

Abstract—A procedure that yields textured and connected binary fractals is presented. The texture is imposed by modeling the fractal as a Markov random field at every resolution level. The model size and the parameters specify the texture. The generation starts at a coarser level and continues at finer levels. Connectivity, which is a global property, is maintained by restricting the flow of the sample generating Markov chain within a limited subset of all possible outcomes of the Markov random field. Sample patterns are shown.

Index Terms—Fractal generation, fractals, Markov random fields, pattern modeling, textures.

Manuscript received June 26, 1987; revised February 14, 1991. Recommended for acceptance by O. Faugeras.

The author is with the Department of Electrical and Electronics Engineering, Bilkent University, Ankara, Turkey.
IEEE Log Number 9100179.

I. INTRODUCTION

It has been observed that many natural objects have some sort of inherent fractal structure. In addition to the unusual beauty of various fractal shapes and the fascinating mathematics of nonlinear recursions, this empirical observation is one of the main reasons of growing intellectual and scientific interest in fractals.

The generation of fractal patterns to simulate natural objects is currently an active field. Some of the generation methods in the literature are based on geometric and statistical observations. No attempt is made to understand the physical basis of the formation of the natural object. The goal in this kind of approach is to come up with an algorithm and, thus, artificially generate patterns that structurally and statistically match the natural object [1],[2]. The scope of this correspondence falls into this group, too. Generation methods of another kind simulate natural fractal objects based on the underlying physics. Actually, the physics governing the formation of the original object is simulated, and mostly, it is shown that the process yields a fractal [3]-[11]. Although the approaches in the second group give more insight to the assumed physical formation of the natural object, and in a sense checks its validity, a complete physical description is not available in many cases. The first type of generation methods totally bypass the physics, therefore, they are not limited by the associated inadequacies. A pure statistical approach is more general and more desirable, especially for design (generation) problems, because it is possible to come up with easily controllable, well structured and flexible models to yield the desired outputs for a wide variety of applications.

Another group of existing studies are reports on analysis of some natural objects to check whether they exhibit some sort of fractal property or not [12]-[20]. Almost invariably, a fractal dimension measurement is performed using some image processing techniques. The measurement techniques vary from the direct application of the fundamental fractal dimension definition to sophisticated indirect methods. The fractal nature of an object is established by the existence of a fractal dimension. The ultimate goal in many of these studies is to characterize irregularly shaped, geometrically strange and complicated, but natural or naturally quite feasible objects. Such a characterization is vital in identification and classification, as well as, in simulation and generation. Although it has been clearly known right from the very beginning that the fractal dimension alone is far from characterizing a fractal object, there is a tendency to improperly state fractal dimension measurements as fractal characterization techniques and propose a classification based solely on the fractal dimension as an adequate one. It is obvious that an adequate classification requires many additional parameters and the fractal dimension can only be a starting step toward such a classification. Many authors have proposed additional parameters to achieve certain fractal descriptions [1],[12],[14].

This study is intended to propose a statistical model with a number of parameters to achieve a complete characterization, and to give an algorithm to generate sample fractals. For this end, the binary valued Markov random field model is employed. The model parameters are the basis for the classification. Since these parameters yield a corresponding texture, the fractals modeled by MRF's as in this correspondence may be properly called *textured fractals*. This correspondence deals solely with connected fractals. Connected patterns form a subset of all possible possible outcomes of a MRF. An algorithm to generate samples from the desired subset is presented with its justification.

The description of the fractal concept as related to the scope of

this correspondence, the model, and the algorithm to generate sample outcomes are given in the next sections.

II. FRACTALS WITH TEXTURE

A two-dimensional discrete binary fractal pattern possesses the following properties:

- 1) It consists of pixels on a two-dimensional regular grid. For convenience, a rectangular grid may be assumed.
- 2) If it is completely covered by squares of sizes ρ by ρ pixels, $\rho = r^k$, r is an integer constant greater than 1, and $k = 1, 2, \dots$, the number of squares needed to cover is given by

$$N(\rho) = c \left(\frac{1}{\rho} \right)^D. \quad (1)$$

(The $\rho = r^k$ condition may be removed for a wider definition.) The real number D is a constant which may take values in the open interval $(0, 2)$, and it is called the *fractal dimension*. Thus, a fractal is a pattern whose hierarchical (pyramid) representation satisfies the relationship (1). Each covering (each level of the pyramid representation) is called a resolution level. Sometimes, the strict relationship given by the above equation is relaxed to hold for the mean value of the number $N(\rho)$ which is a random variable for this case, preferably with a small variance. In this correspondence, this approach is not adopted since each sample fractal is required to have the same number of pixels.

Although the simple properties outlined above are enough to have a 2-D discrete binary fractal, many other properties may be added to impose some specific features. Here in this correspondence, we will elaborate on the fractals that are connected in the four-nearest neighbor sense. This condition enforces the D to take values in the interval $[1, 2)$.

One well-known method of generating fractals which are as defined above may be called *carving*. Simply, a large square (at least equal to the overall size of the fractal) is carved out at different resolution levels according to a prescribed procedure (see [1]). The nature of the carving process defines a texture but the relationship is not well established in the literature. A carving process may have two parts: deterministic and random. For example, the square may be partitioned into r^2 smaller squares by dividing each side by r (the deterministic part), and then, p out of these r^2 small squares may be deleted randomly without violating the connectivity (the random part). The divide-and-delete (carving) procedure is repeated at a finer scale for each of the remaining $r^2 - p$ squares. This goes on until the finest scale (1 pixel = 1 square) is reached, resulting in a fractal with $D = \log(r^2 - p) / \log r$. The fractal has a texture as a direct consequence of the prescribed process. If one wishes to have more control over the texture, the carving process can be made more complex, probably with the incorporation of some more parameters. For instance, instead of deleting p squares as in the example above, one may choose to delete a random number of squares, where the randomness is specified by a distribution (see [1] for an example). However, the relationship between the model parameters and the texture remains rather heuristic and the variety of possible textures may be limited.

The application of the MRF model solves this problem by establishing a link between the model parameters and the texture. Note that *any* random field can be represented as a MRF if the order (the neighborhood size) is chosen large enough. Any random texture can be prescribed and implemented using a MRF model. Thus the model is a general and simple one. The only limitations are the practical ones which may become severe if the order becomes too large.

v	u	w'
t		t'
w	u'	v'

Fig. 1. Second-order neighborhood in 2-D space.

A. Markov Random Fields

The fundamentals of MRF's can be found in the literature (see, for example, [25],[28],[29],[36],[37],[38]). Here in this paragraph a brief description is included to clarify the fractal model being discussed.

A *random field* is defined over a lattice system. Here in this correspondence, the examples are over a two-dimensional regularly spaced sites. A random variable is associated with each site. In a *Markov random field*, the value of a site random variable is independent of the nonneighbor site values if the values of the neighborhood site random variables are given. The set of neighborhood sites corresponding to each site should be specified. A *second-order neighborhood* is given in Fig. 1. The probabilities of the values of the center sites, given their neighborhood values, must have a consistent form. The form adopted in this correspondence, for the second order neighborhood of Fig. 1 and for binary random variables, can be written as

$$P(X(i, j) = x \mid X(i', j') = x_{i'j'}, (i', j') \in \eta_{ij}) = \frac{\exp xT}{1 + \exp T}, \quad x, x_{i'j'} \in \{0, 1\}, \quad \forall (i, j) \quad (2a)$$

where

$$T = \alpha + \beta_1(t + t') + \beta_2(u + u') + \gamma_1(v + v') + \gamma_2(w + w') + \xi_1(tu + u'w + w't') + \xi_2(tv + v'u' + ut') + \xi_3(tw + w'u + u't') + \xi_4(t'u + uv + v't') + \eta(tuv + t'u'v' + tu'w + t'u'w') \quad (2b)$$

$$t, t', u, u', v, v', w, w' \in \{0, 1\}.$$

If the field is specified to be statistically symmetrical (i.e., *isotropic*), then $\beta_1 = \beta_2 = \beta$, $\gamma_1 = \gamma_2 = \gamma$, and $\xi_1 = \xi_2 = \xi_3 = \xi_4 = \xi$. A graphical representation of the *cliques* as defined by the terms in the parentheses in (2b) can be found in Hassner and Sklansky [29], Geman and Geman [30], or in Derin and Elliott [31]. The texture is fully specified if the neighborhood, and the associated parameters of the model are given. For the second-order statistically symmetrical MRF, there are only five parameters to be specified: α , β , γ , ξ and η .

A *textured fractal* is modeled as a Markov random field at its every resolution level. However, since our fractals have to be connected, we have to restrict our interest to a subset of the outcomes of the MRF. Monte Carlo procedure to generate sample outcomes of a specified MRF are given in the literature. However, these algorithms do not restrict the samples to a subset of all possible outcomes. To generate fractals, we need to restrict the samples to get only globally connected patterns. An algorithm for this purpose is presented in the next section.

III. GENERATING TEXTURED FRACTALS

To generate fractals with a prescribed texture, we start with a large black square and partition it into r^2 smaller squares. The large

Fig. 2. The generation of fractals with texture using the Markov random field model. (a) The initial pattern: a large black square. (b) The result of the first hierarchical (coarsest) level. First, 7 of the 16 smaller squares of (a) are deleted; the resultant pattern is the initial state of the Markov chain; the chain evolves to steady-state to yield an outcome, as shown in (b), of the specified Markov random field model. The fractal dimension is $\log 9 / \log 4 = 1.585$. The second order neighborhood, with symmetrical parameters [see (3)] $\beta = 2.168$, $\gamma = 0.906$, $\xi = -0.562$, $\eta = 0.406$ is chosen. (c) The procedure for the transition from (a) to (b) is also used at a finer level (second hierarchical level) for the transition from (b) to (c). (d) The same procedure may be continued indefinitely at finer and finer scales. The third level is shown.

square is surrounded by white sites up to the border of the finite region of interest. An example is given in Fig. 2, where $r = 4$. The white surroundings are always kept white throughout the generation process, and all changes take place within the initial large black square. The "always white" surroundings are among the *exterior sites* [25]. The exterior sites are not included in the lists of step 1 below. To have a fractal with fractal dimension $D = \log(r^2 - p) / \log r$, p of these r^2 squares must be deleted. At the beginning, this r by r pattern (each smaller square is taken as a unit, and this unit may be referred as a *site*) is carved out to get a thinner pattern by arbitrarily deleting p of the r^2 object (black) sites without paying attention to the texture at all, but without violating the connectivity. The texture is specified by selecting the neighborhood size and the associated parameters of the MRF model. Once this specification is complete, what is needed is a Markov chain which will generate the desired samples at the steady-state. Then the above pattern, which is connected and has exactly the required number of black and white sites, is taken as the initial state of the Markov chain. The chain used for the figures of this correspondence is an interchange algorithm. Two sites are picked up at each attempt, and some tests are conducted. As a consequence, either the site values are left as they are, or they are interchanged. This keeps the numbers of black and white sites at the desired values throughout the evolution of the chain. The complete interchange algorithm is as follows:

- 1) A list of the coordinates of all black sites is made. Note that the number of black sites N_b is known and fixed throughout the evolution of the chain. Similarly, a list of all interior white sites which has at least one black neighbor is also made, and the number of such white border sites N_{wb} is counted. N_{wb_m} is a constant upper bound to this number and fixed at the beginning of the chain.
- 2) At this step, a ratio $p = N_{wb} / N_{wb_m}$ is computed. The next state of the chain is chosen to be the same as the current state with probability $1 - p$. Therefore, a random number s that is uniformly distributed in $[0, 1]$ is generated, and if $s > p$, the algorithm jumps to item 5; otherwise, it continues with item 3.
- 3) This step is a feasibility check. A black site from the first list and a white site from the second list are chosen randomly. Connectivity and covering checks are performed. If interchanging the site values yields a connected pattern, and if the covering property as required by the fractal nature is not violated by the interchange, then this interchange is feasible. Otherwise, the next state is the same as the current state, so the algorithm goes to item 5. The covering property violation will be explained later.
- 4) At this step, a ratio R is computed. Let $X(m, n) = x_{mn}$ for all (m, n) be the present state. Similarly, let $X(m, n) = y_{mn}$ be the pattern which would be obtained if the proposed interchange

is performed. This ratio is given as (see [28] or [25] for a general definition; this is a special case of the definition given in these references):

$$R = \frac{P(X(i, j) = 1 \mid X(i', j') = y_{i'j'}, (i', j') \in \eta_{ij}) P(X(k, l) = 0 \mid X(k', l') = x_{k'l'}, (k', l') \in \eta_{kl})}{P(X(i, j) = 0 \mid X(i', j') = y_{i'j'}, (i', j') \in \eta_{ij}) P(X(k, l) = 1 \mid X(k', l') = x_{k'l'}, (k', l') \in \eta_{kl})} \quad (3a)$$

(i, j) and (k, l) are the coordinates of the original white (white=0) and black (black=1) pixels that are being tested for the interchange, respectively. This ratio is a measure of the relative likelihoods of the proposed pattern, which is obtained by interchanging the pixels, and the current pattern. All four conditional probabilities in this expression can be computed easily using (2). It can be shown that this ratio does not depend on α . If $R > 1$ the pixels being tested are interchanged to get the next pattern. If $R < 1$, then the pixels are interchanged with probability R . Note that even if there is an interchange, the new pattern is still connected and does not violate the covering requirement because of the tests done in item 3. Note that the parameter α is not needed for the algorithm.

- 5) This step is the stopping criteria. If the steady-state has been reached, the sample is ready. Otherwise, steps 1–5 are repeated. Note that, if there was no interchange at the last attempt, step 1 need not be repeated.

Once the steady state is reached, the final pattern, which is a connected pattern with the prescribed texture, is taken as the coarsest level covering of the fractal (see Fig. 2(b)). Let N_b denote the number of black sites of this pattern. The generation of the next finer level may now start from the already generated coarse pattern. The algorithm for this level is exactly the same as the previous one: each square of the coarse pattern is partitioned into r by r smaller squares. The white sites (peripheries) of the coarser pattern always stay white from now on (i.e., become the exterior) and all changes are within the black interior. This assures a fractal pattern, together with the covering violation check mentioned in step 3 of the algorithm: The covering property is violated if all of the finer squares under a coarse black square become white. The coarser level pattern is again thinned as dictated by the fractal dimension, as before $N_b p$ of the $N_b r^2$ finer squares (sites) are removed; the number of black sites is updated as $N_b \leftarrow N_b(r^2 - p)$; after a proper update of the constant N_{wbm} , the same interchange algorithm is employed to yield the final pattern of this finer level at the steady state. Note that the number of total sites, as well as the number of black (object) sites, exponentially grow as the levels get finer. However, their rates are different: the total number is multiplied by r^2 at each level, whereas the number of black sites is multiplied by r^D . In fact, this is the feature which makes an object a fractal. The resolution levels may be continued indefinitely. (See Figs. 2(c) and (d), for example.)

A. About the Algorithm

The algorithm given above deserves some justification. The fundamental approach is the same as in many other algorithms in the literature. (See, for example [28].) A symmetric, aperiodic, and irreducible Markov chain is chosen with the one-step transition matrix P^* . The states of the Markov chain are the connected patterns with the desired number of black sites. Then this Markov chain is converted to another one with the steady-state probabilities $\{q(j)\}$, and with the one-step transition matrix P . P and P^* are related as

$$p(i, j) = \begin{cases} p^*(i, j) \frac{q(j)}{q(i)} & \text{if } q(i) > q(j) \\ p^*(i, j) & \text{if } q(i) \leq q(j) \end{cases}$$

The diagonal entries of the matrix P are taken to have the summations of its entries over a column equal to one.

For the algorithm given in this correspondence, the $q(j)/q(i)$ ratios are computed in step 4 easily as R . The Markov chain with the matrix P^* corresponds to the case where step 4 is omitted, i.e., replaced by "always do the proposed change." The nondiagonal entries of the matrix P^* are either a constant equal to $k = 1/(N_{wbm} N_b)$ as a consequence of the step two of the algorithm, or 0. The 0 entries correspond to transitions to those feasible patterns which cannot be reached from the present state in a single step. It is straightforward to show that the nonzero entries are equal to a constant because the probability of picking up a pair at any state is $1/(N_{wb} N_b)$, and from step 2, the interchange probability $p^*(i, j)$ (when step 4 is altered as explained above) becomes $\left(\frac{N_{wb}}{N_{wbm}}\right) \left(\frac{1}{N_{wb} N_b}\right) = \frac{1}{N_{wbm} N_b}$. Since N_{wbm} and N_b are fixed throughout the chain, the nondiagonal entries are always constant, and this assures a symmetric P^* matrix. Keeping the lists of step 1 and making the selections from these lists increases the rate of convergence to the steady state significantly. The chain is a single chain because every connected pattern can be eventually reached from any other connected pattern by the proposed interchange procedure; there are no periodic states. In addition, there are finite number of states; therefore, the steady-state probabilities exist, and they are equal to the desired MRF distribution (see the theorem given in [28]).

The parameter α is not needed in any interchange algorithm or, in general, in any algorithm that controls the number of pixels with a given value. This is expected because this parameter controls the relative numbers of black and white pixels; the algorithm explicitly controls the same feature and thus overrides α .

IV. SAMPLE PATTERNS

As discussed earlier, the parameters of the model determine the texture. For example, for the second order symmetric model of (2), there are five parameters to be specified. As noted in the step 4 of the algorithm, one of these parameters, α , is irrelevant in the interchange algorithms. Thus there are four parameters in addition to the fractal dimension to control the texture. The model parameters other than the fractal dimension will be called the *texture parameters* from now on. A relationship between these parameters and the implied texture can be given if the forms of the cliques they are related are considered. The parameter β controls the tendency to form vertical and horizontal bars: as β gets larger, the tendency toward clustering in the form of vertical and horizontal bars in the pattern increases. Negative values of β correspond to avoidance from such clustering. Similarly, γ controls the diagonal bars. ξ is associated with L shaped groupings of three pixels in any orientation. η can be interpreted as a measure of the frequency of groupings of four pixels in the form of a 2 by 2 square. Note that, all of these patterns might occur at the same location, i.e., a pixel may be an element of a bar and an L, etc., at the same time. Generated fractal patterns with the same fractal dimension but different texture parameters are shown in Fig. 3. For example, the parameters of Fig. 3(a) may be interpreted as follows: there is a strong tendency for horizontal and vertical bars (relatively higher β), whereas diagonal bars and solid clustering are mostly avoided (relatively lower γ and η). As it is seen, the interpretation is

$$\beta = 0.6 \quad \gamma = -0.6 \quad \xi = 0.1 \quad \eta = -0.1$$

(a)

$$\beta = -2.0 \quad \gamma = 2.0 \quad \xi = 1.0 \quad \eta = -3.0$$

(b)

$$\beta = 2.0 \quad \gamma = 1.0 \quad \xi = -1.5 \quad \eta = -2.5$$

(c)

Fig. 3. Three fractals with the same fractal dimension, $\log 9 / \log 4 = 1.585$, but different textures. The texture parameters are given with the figures.

(a)

(b)

(c)

(d)

Fig. 4. Four fractals with the same texture parameters as given in Fig. 2, but different fractal dimensions. The fractal dimensions are 1.2925, 1.5000, 1.5850 and 1.6610, for (a), (b), (c), and (d), respectively.

consistent with the pattern of Fig. 3(a). Fig. 3(b) is in favor of a checkerboard type of texture due to high γ and low β . Other textures may be interpreted similarly. Some patterns with the same four texture parameters but different fractal dimensions are given in Fig. 4.

It is also possible to have different texture parameters, and thus different textures, at various resolution levels. In this case, (1) still holds, and so the object is still a fractal. But the self-similarities are

Fig. 5. A fractal with a resolution-varying texture. The coarsest level parameters are: $\beta = 0.6$, $\gamma = -0.6$, $\xi = 0.1$, and $\eta = -0.1$. The next finer level has $\beta = 0.3$, $\gamma = -0.3$, $\xi = 0.1$, $\eta = 0.1$, and the third level parameters are $\beta = -2$, $\gamma = 2$, $\xi = 1$, and $\eta = -3$.

weaker. An example to such an object is shown in Fig. 5.

V. CONCLUSION

An algorithm which yields textured connected fractal patterns is presented. The generated fractals are modeled as connected outcomes of Markov random fields at every resolution level. The texture is controlled by the parameters of the MRF model being employed. The algorithm is general: it can be easily extended to higher dimensions, and larger window sizes can be used for more complex textures.

Note that there is no need to have the same texture at every resolution level. The covering property as given by (1) is enough to have a fractal. It is possible to have a wider variety of fractals by having different textures at every resolution level. This can be achieved by having different model parameters while generating different resolution levels. If desired, one may even choose different window sizes. This flexibility can be successfully used to model many natural patterns as textured fractals, and simulated patterns can be generated. For this purpose, the model parameters have to be estimated from the existing actual pattern.

Finally, it should be noted that the D of (1) may not be a constant, but a function of ρ . In this case, it is definitely possible to cover a much wider set of patterns. However, the generated patterns may not be called fractals, unless the fractal definition implied by (1) is modified.

ACKNOWLEDGMENT

The author thanks M. İzzet Gürelli for the computer program that provided the figures in this correspondence.

REFERENCES

- [1] J.E. Martin and K.D. Keefer, "Regular random fractals and the n -parameter model," *J. Phys. A*, vol. 18, pp. L625-L631, 1985.
- [2] D.C. Hong, S. Havlin, and H.E. Stanley, "Family of growth fractals with continuously tunable chemical dimension," *J. Phys. A*, vol. 18, pp. L1103-L1107, 1985.
- [3] P. Meakin, "Diffusion-controlled cluster formation in 2-6 dimensional space," *Phys. Rev. A*, vol. 27, no. 3, pp. 1495-1507, Mar. 1983.
- [4] —, "Diffusion-limited aggregation in three dimensions: results from a new cluster-cluster aggregation model," *J. Colloid Interface Sci.*, vol. 102, no. 2, pp. 491-504, Dec. 1984.
- [5] —, "The structure of two-dimensional Witten-Sander aggregates," *J. Phys. A*, vol. 18, pp. L661-L666, 1985.
- [6] P. Meakin and Z.B. Djordjević, "Cluster-cluster aggregation in two-monomer systems," *J. Phys. A*, vol. 19, pp. 2137-2153, 1986.
- [7] T.A. Witten and M.E. Cates, "Tenuous structures from disorderly growth processes," *Sci.*, vol. 232, pp. 1607-1612, June 27, 1986.
- [8] T.A. Witten and L.M. Sander, "Diffusion-limited aggregation," *Phys. Rev. B*, vol. 27, no. 9, pp. 5686-5697, May 1983.
- [9] T.A. Witten, Jr. and L.M. Sander, "Diffusion-limited aggregation, a kinetic critical phenomenon," *Phys. Rev. Lett.*, vol. 47, no. 19, pp. 1400-1403, Nov. 1981.
- [10] M. Kolb, "Reversible diffusion-limited cluster aggregation," *J. Phys. A*, vol. 19, pp. L263-L268, 1986.
- [11] R. Jullien, "A new model of cluster aggregation," *J. Phys. A*, vol. 19, pp. 2129-2136, 1986.
- [12] S.R. Forrest and T.A. Witten, Jr., "Long-range correlations in smoke-particle aggregates," *J. Phys. A*, vol. 12, no. 5, pp. L109-L117, 1979.
- [13] M.D. Normand and M. Peleg, "Determination of the fractal dimension of a particle silhouette using image-processing techniques," *Powder Technol.*, vol. 45, pp. 271-275, 1986.
- [14] R.M. Malzbender, R.C. Mockler, and W.J. O'Sullivan, "Topological and geometrical properties of DLA clusters," *J. Phys. A*, vol. 18, pp. L1143-L1147, 1985.
- [15] A. Margolina, "Three fractal dimension of cluster perimeters generated by a kinetic walk," *J. Phys. A*, vol. 18, pp. L651-L656, 1985.
- [16] R. Orbach, "Dynamics of fractal networks," *Sci.*, vol. 231, pp. 814-819, 21 Feb. 1986.
- [17] P. Pfeifer and D. Avnir, "Chemistry in noninteger dimensions between two and three. 1. Fractal theory of heterogeneous surfaces," *J. Chem. Phys.*, vol. 79, no. 7, pp. 3558-3565, Oct. 1983.
- [18] M. Lewis and D.C. Rees, "Fractal surfaces of proteins," *Sci.*, vol. 230, pp. 1163-1165, Dec. 1985.
- [19] B.H. Kaye, "Fractal dimension and signature waveform characterization of fine particle shape," *Amer. Lab.*, pp. 55-63, Apr. 1985.
- [20] —, "Specification of the ruggedness and/or texture of a fine particle profile by its fractal dimension," *Powder Technol.*, vol. 21, pp. 1-16, 1978.
- [21] N.N. Clark, "Three techniques for implementing digital fractal analysis of particle shape," *Powder Technol.*, vol. 46, pp. 45-52, 1986.
- [22] A.P. Pentland, "Fractal based description of natural scenes," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-6, no. 6, pp. 661-674, Nov. 1984.
- [23] B.B. Mandelbrot, *Fractals: Form, Chance and Dimension*. San Francisco, CA: Freeman, 1977.
- [24] —, *The Fractal Geometry of Nature*. San Francisco, CA: Freeman, 1982.
- [25] J. Besag, "Spatial interactions and the statistical analysis of lattice systems," *J. Roy. Stat. Soc. B*, vol. 36, pp. 192-236, 1974.
- [26] J. Moussouris, "Gibbs and Markov random systems with constraints," *J. Stat. Phys.*, vol. 10, no. 1, pp. 11-33, 1974.
- [27] H. Derin, H. Elliott, R. Cristi, and D. Geman, "Bayes smoothing algorithms for segmentation of binary images modeled by Markov random fields," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-6, no. 6, pp. 707-720, Nov. 1984.
- [28] G.R. Cross and A.K. Jain, "Markov random field texture models," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-5, no. 1, pp. 25-39, Jan. 1983.
- [29] M. Hassner and J. Sklansky, "The use of Markov random fields as models of texture," *Comput. Graphics Image Processing*, vol. 12, pp. 357-370, 1980.
- [30] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-6, no. 6, pp. 721-741, Nov. 1984.
- [31] H. Derin and H. Elliott, "Modeling and segmentation of noisy and textured images using Gibbs random fields," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-9, no. 1, pp. 39-55, Jan. 1987.
- [32] B.D. Ripley, "Statistics, images, and pattern recognition," *Canadian J. Stat.*, vol. 14, no. 2, pp. 83-111, 1986.
- [33] F.S. Cohen and D.B. Cooper, "Simple parallel hierarchical and relaxation algorithms for segmenting noncausal Markovian random fields," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-9, no. 2, pp. 195-219, Mar. 1987.
- [34] J.M. Hammersley and D.C. Handscomb, *Monte Carlo Methods*. London: Methuen, 1964.
- [35] P.A. Flinn, "Monte Carlo calculation of phase separation in a two-dimensional Ising system," *J. Stat. Phys.*, vol. 10, no. 1, pp. 89-97, 1974.

- [36] V. Isham, "An introduction to spatial point processes and Markov random fields," *Int. Stat. Rev.*, vol. 49, pp. 21–43, 1981.
- [37] R. Kinderman and J. L. Snell, *Markov Random Fields and Their Applications*. Providence, RI: American Mathematical Society, 1980.
- [38] Y. A. Rosanov, *Markov Random Fields*. New York: Springer-Verlag, 1982.

A Simulation-Based Estimator for Hidden Markov Random Fields

Ari Veijanen

Abstract—A new estimator is introduced for estimating the parameters of a Markov random field X from inaccurate observations. The marginal distributions of $(X_{i,j}, X_{i+u,j+v})(u, v = -1, 0, 1)$ are first estimated from an image. Then, random fields X^* are simulated with the probability of $X_{i,j}^* = a$ and $X_{i+u,j+v}^* = b$ nearly equal to the estimate of $P\{X_{i,j} = a, X_{i+u,j+v} = b\}$. A simulation method similar to the Gibbs sampler is used. The parameters of the Markov random field model are estimated from the X^* 's with the pseudolikelihood method.

Index Terms—Estimation, image analysis, Markov random field, pseudolikelihood, simulation.

I. INTRODUCTION

WE study estimation problems in image analysis based on a random field model. The true scene $X = \{X_{i,j}\}$ on a lattice $L = \{(i, j) : i = 1, 2, \dots, n; j = 1, 2, \dots, m\}$ is often modeled as a Markov random field [4]–[8], [11], [23]. The observations $Y_{i,j}$ are assumed to follow some known distribution given the $X_{i,j}$'s. In the case when X is observed, there exist several methods for estimating the parameter θ of the random field model [3], [7], [19]. Estimation from inaccurate observations is more difficult. One solution has been to estimate the parameter from classifications. Estimation and classification alternate in many iterative methods [4], [7], [9], [15], [20]. We first obtain classifications $\hat{X}_{i,j}$ using an initial estimate of θ , and then a new estimate is computed from the $\hat{X}_{i,j}$'s to be used in the next classification step. This is repeated until the final classification $\hat{X}$ is obtained. If there are few classification errors, then $\hat{X}$ resembles X and the estimates from $\hat{X}$ and X are similar as well. In general, estimates based on classifications are biased, especially if the image is too strongly smoothed. Estimation methods without simultaneous classification are therefore perhaps preferable [1], [2], [6], [13], [18], [25].

As a simple example of a Markov random field, consider Pickard random field [12]. It is easy to estimate its parameters from an image, since certain marginal distributions of a Pickard random field [13] are tractable. For example, the marginal distribution of $(X_{i,j}, X_{i,j+1})$ is known.

Even in the general case it is possible to estimate the probabilities

$$P\{X_{i,j} = a, X_{i+u,j+v} = b\} (u, v = -1, 0, 1) \quad (1.1)$$

Manuscript received February 25, 1990; revised February 4, 1991. This work was supported by the Finnish Academy and the Emil Aaltonen Foundation.

The author is with the Department of Statistics, University of Helsinki, Helsinki, Finland.

IEEE Log Number 9100180.

from an image. However, it is not known how they depend on θ . The estimates of (1.1) are thus not useful as such. In our method, random fields X^* are simulated given the estimates of (1.1). The probability of $X_{i,j}^* = a$ and $X_{i+u,j+v}^* = b$ is close to the estimate of the probability $P\{X_{i,j} = a, X_{i+u,j+v} = b\}$. In other words, we try to construct random fields X^* with the same local characteristics as the true scene X . The X^* 's should resemble X so much that pseudolikelihood estimates [3], [4] and X^* are close to estimates from X . This approach is called SPLEMD, simulated pseudolikelihoods given estimated marginal distributions. Other estimation methods based on simulations have been presented in [5], [17], [21], [22], and [25].

It has been suggested that the proportions of classes in the restoration should be close to the estimated proportions (Green in the discussion of [4]). This idea is generalized using the estimates of probabilities (1.1) as constraints for the corresponding proportional frequencies in the restored image.

II. THE MODEL

Consider first a Markov (Gibbs) random field $X = \{X_{i,j}\}$ on a lattice $L = \{(i, j) : i = 1, 2, \dots, n; j = 1, 2, \dots, m\}$. The random variables $X_{i,j}$ are assumed to have τ values $(1, 2, \dots, \tau)$. For a model with pairwise interactions, the probability mass function is

$$P\{X = x; \theta\} = Z_\theta^{-1} \exp\{U\theta(x)\} \quad (2.1)$$

where Z_θ is the normalizing constant

$$\begin{aligned} U\theta(x) &= \sum_{a=1}^{\tau} n_a(x) \alpha(a) + \sum_{k=1}^4 \sum_{a=1}^{\tau} \sum_{b=1}^{\tau} n_{a,b}^k(x) \beta^k(a, b), \\ n_a(x) &= \text{card}\{(i, j) \in L : x_{i,j} = a\}, \\ n_{a,b}^1(x) &= \text{card}\{((i, j), (i, j+1)) \in L \times L : x_{i,j} \\ &\quad = a, x_{i,j+1} = b\}, \\ n_{a,b}^2(x) &= \text{card}\{((i, j), (i+1, j)) \in L \times L : x_{i,j} \\ &\quad = a, x_{i+1,j} = b\}, \\ n_{a,b}^3(x) &= \text{card}\{((i, j), (i+1, j+1)) \in L \times L : x_{i,j} \\ &\quad = a, x_{i+1,j+1} = b\} \end{aligned} \quad (2.2)$$

and

$$\begin{aligned} n_{a,b}^4(x) &= \text{card}\{((i, j), (i+1, j-1)) \in L \times L : x_{i,j} \\ &\quad = a, x_{i+1,j-1} = b\}. \end{aligned}$$

The vector of frequencies $n_{a,b}^k(X)$ is denoted by $\mathbf{n}(X)$. Note that the $n_a(X)$'s can be derived from $\mathbf{n}(X)$. The parameters $\alpha(a)$ describe the frequency of each class, and the parameters $\beta^k(a, b)$ describe their dependencies. For example, if $\beta^1(a, b)$ increases while the other parameters are held constant, then the probability $P\{X_{i,j} = a, X_{i,j+1} = b\}$ increases. We use constraints

$$\sum_{a=1}^{\tau} \alpha(a) = 0, \quad \sum_{b=1}^{\tau} \beta^k(a, b) = 0 \quad \text{and} \quad \beta^k(a, b) = \beta^k(b, a) \quad (2.3)$$

($a, b = 1, 2, \dots, \tau; k = 1, 2, 3, 4$). A positive value of $\beta^k(a, b)$ is interpreted so that the interactions of class a with class b are stronger than the mean of interactions of class a with other classes. Under the conditions (2.3), the parameters $\alpha(a)$ and $\beta^k(a, b)$ for $a < \tau$ and $b \leq a$ determine the values of all the other parameters $\alpha(a)$ and $\beta^k(a, b)$.