On the use of the 1D Boolean model for the description of binary textures

M Petrou, M Arrigo and J A Vons

Dept. of Electronic and Electrical Engineering,
University of Surrey, Guildford GU2 5XH, United Kingdom.

e-mail: ees1mp@ee.surrey.ac.uk

Abstract

In this paper we investigate the use of the Poisson model in the creation and the description of binary textures, in conjunction with two forms of image scanning, namely raster and Hilbert scan. We apply our model to some binary random textures for which we estimate the Poisson parameters using Maximum Likelihood Estimation and find that the values we obtain are sufficiently distinct and sufficiently constant over the same texture, to allow the use of this model in texture identification.

1 Introduction

One of the major topics in image processing is texture identification. The aim of the work described in this paper is to investigate the properties of the Boolean model for creating and describing textures. The model has been investigated intensively from the theoretical point of view, but whether it could be used for real texture images in practice, has not been established yet.

The simplest form of the Boolean model is the 1D version of it. As texture is a spatial property, it is obvious that the most appropriate model for its description would be the 2D Boolean model, where a Poisson point process is used to place randomly some primitive shapes described by parameters with prespecified probability density functions. This approach, however, presents a chicken and egg situation where one has to identify the texels, ie the primitive models, before the modelling takes place. Thus, the approach does not seem easy to use. The 1D model on the other hand is much more straightforward in its implementation and use. To bypass the problem of dimensionality, and introduce some spatial property to the model, we use it in conjunction with Hilbert scanning of the image.

The basic principles of the Boolean model are explained in section 2. In section 3 we shall briefly discuss the Hilbert scanning of a 2D array. In section 4 we shall use the Boolean model in conjunction with both Hilbert and conventional raster scanning to create some sample textures and thus explore the meaning of the two model parameters in terms of texture appearance. In section 5 we shall describe how the model was used to estimate the parameters of 10 different binary textures and discuss how useful it is to discriminate these textures. We shall present our conclusions in section 6.

BMVC 1996 doi:10.5244/C.10.41

2 The one-dimensional Boolean model

The 2D Boolean random set model can be used for describing and creating texture images. For this purpose, a texture is assumed to consist of a large number of certain primitive shapes. The Boolean model consists of two independent statistical processes, a shape process and a Poisson point process. The outcomes of the shape process determine the shapes of the primitives, and the outcomes of the Poisson point process determine where these shapes appear. In a typical realization of a Boolean model, shapes tend to overlap each other.

In the one-dimensional case, the shapes of the Boolean model are simply line segments. For each segment the left end-point is considered to be the origin of the shape. From this origin the shape is said to *emanate* to the right. Thus, in the 1D Boolean model, a shape is completely identified by the length of the line segment and the position of its origin.

The locations of the origins of the shapes in a Boolean model are the outcomes of a Poisson point process with probability p. This probability is called the *marking probability* and is one of the parameters of the Boolean model. The other parameters describe the form of the shapes. In the one dimensional case, this is just the length of the line segments, which are distributed according to some distribution function $C(k)$, $k = 0, 1, \ldots$. $C(k)$ is the probability that a line segment has length less than or equal to k:

$$C(k) = P(K \leq k), \qquad k = 0, 1, \ldots \tag{1}$$

Because k is discrete, $C(k) = c_0 + c_1 + c_2 + \ldots + c_k$, where $c_i = P(K = i)$ is the probability that a line segment has length i, and $C(0) \equiv 0$ [2].

The marking probability and the segment length distribution together form the one-dimensional Boolean model. They can be combined to one expression for the probability that a given line segment of length less than or equal to k emanates from a given point. This probability is given by the distribution function $F(k)$ [3]:

$$F(k) = 1 - p + pC(k), \qquad \text{for } k = 0, 1, \ldots \tag{2}$$

If $F(k) = 0$ then no line segment emanates from the concerning point. If $F(k) = i \neq 0$ then that point is the origin of a line segment with length i. Usually the segment length distribution is a function of a parameter, say θ. For a given segment length distribution, the only parameters of the one dimensional Boolean model are p and θ.

The Boolean model can be used either to create a texture image, or to describe an existing texture.

3 Hilbert scanning of an image

If we create a 1D string of numbers, we can wrap it up to form an image, by reading its values in a prespecified way. The most straightforward way is to use it in a raster form where successive segments of a certain length are placed one below another (horizontal raster) or one next to the other (vertical raster) to form the image. In this case, however, the model applies only along the direction of wrapping, and it does not quantify the relationship between the neighbouring pixels in the orthogonal to the raster direction. It is well known, however, that, in the continuous case, a 1D curve can fill up a 2D space if the curve has fractal dimension that approaches 2. In the discrete case of a lattice, we can use the Hilbert scanning, which in effect keeps a curve for as long as possible in the vicinity of a pixel, before it allows it to jump out of that neighbourhood [4, 5]. Such a scanning of an image enhances the neighbourhood structure of the 1D curve. An example of such scanning is shown in figure 1. In this paper, we shall use all three types of scanning of an image: raster horizontal, raster vertical and Hilbert, in order either to wrap a created string to form an image, or to scan a given image to form a string from which we shall estimate the parameters of the image.

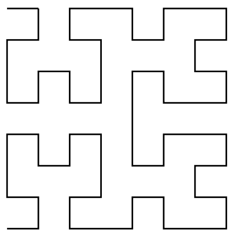


Figure 1: Hilbert scanning

4 Creating Boolean images

We used the model described in section 2 to create strings of length 16,384 pixels for various parameter values. For all these strings, the black segment lengths were assumed to be Poisson distributed:

$$C(k) = e^{-\theta} \sum_{m=0}^{k} \frac{\theta^m}{m!} \qquad k = 0, 1, 2... \tag{3}$$

Here, θ is the parameter of the black segment length distribution, and denotes the mean segment length. We then wrapped these strings either in raster format or by Hilbert scanning to form 2D images 128×128 pixels in size.

Figure 2 shows the images created using the vertical raster as well as the Hilbert scanning formats. The images obtained by horizontal raster format are not shown as they are simply rotated versions of those shown at the top of figure 2.

Both parameters result in a darker image when they are increased. However, increasing only the mean segment length results in a coarser image. As the marking probability and the mean segment length increase, more line segments overlap, leaving less uncovered pixels. Eventually, all line segments overlap resulting in a completely black image.

From the distinct appearance of these two sets of images, we can see that the parameter values by themselves are not enough to characterise a texture. So the wrapping sequence is as important as the parameter values used. However, what is important to note is that the raster format together with the parameter values

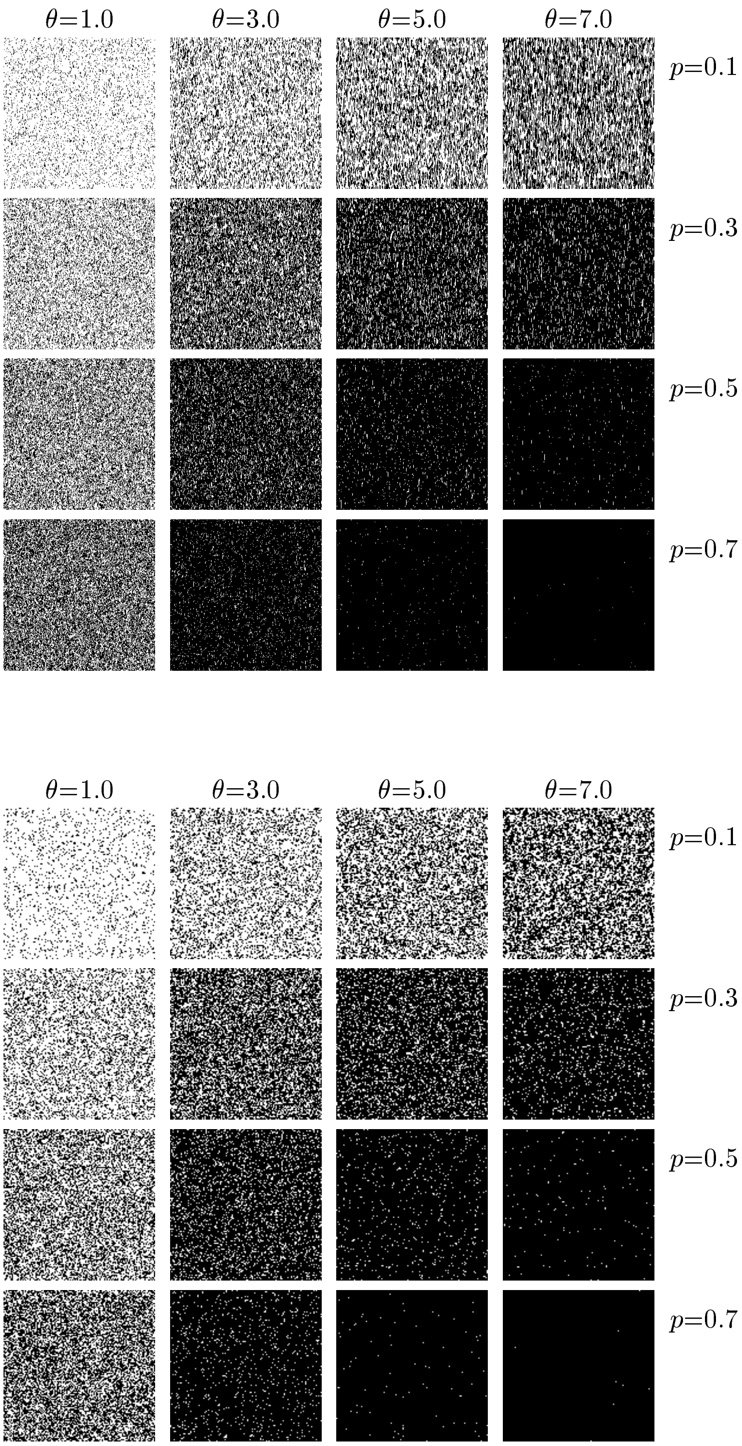


Figure 2: Textures (128×128) created by wrapping a 1D Boolean string using vertical raster (top) and Hilbert (bottom) scanning respectively.

are enough to create quite a range of binary textures, from strongly directional to completely homogeneous. Clearly, one could also create directionality along the diagonal orientation by raster scanning the lattice along that direction.

5 Characterising textures by the Boolean parameters and the wrapping format

One question often asked about parametric models of texture, is whether the model parameters characterise the texture uniquely. To answer that, after having created a texture image with known parameters, we try to estimate these parameters. We performed hundreds of experiments for this purpose, using the maximum likelihood approach described in [2] and [3].

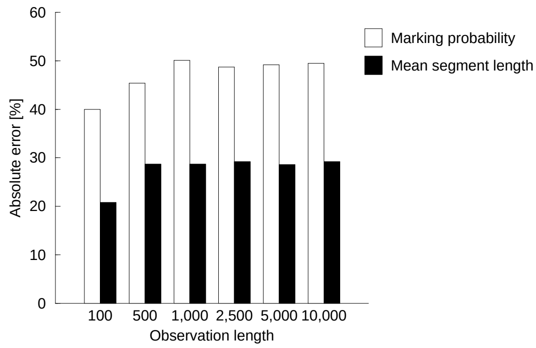


Figure 3: Relative errors of the maximum likelihood estimates of the 1D Boolean image created with $p = 0.3$ and $\theta = 3.0$ with different observation lengths.

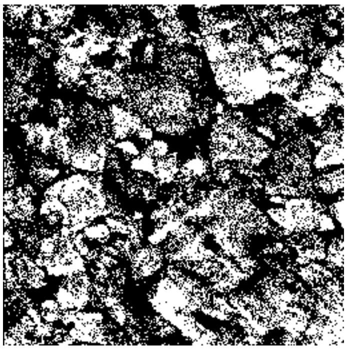
Figure 4 shows some typical results of the accuracy with which the values of the Boolean model could be defined by the above method, for various sizes of the sequence used. At first sight these results seem rather disappointing as the parameters recovered are typically 50% or 30% wrong. It seems that unless we have a good parameter estimation method, we cannot use these parameter values. The uncertainty clearly arises from the fact that there are more than one options by which a certain sequence can arise since when the sequence is created, if a pixel is already black due to a large segment placed earlier, its value does not change when a closer neighbouring pixel becomes the origin of a new black segment. It is this irreversible non-linear step in the creation of these images that prevents us from recovering the parameters of the process exactly. The Boolean image with $p = 0.5$ and $\theta = 8.0$ has so few white pixels, that only one observation contains enough black and white run-lengths to base a reliable estimation on.

However, the exact recovery of the parameters of the process, is probably the wrong question to be asking. Images, even binary textures, are not expected to adhere really to the Boolean model. The Boolean model, just like any other model, should simply be considered as yet another reference configuration which can be used to describe the data in some way. So, in a highly dimensional hyper-space where an image can be represented by a point, the use of a model is nothing else than the creation of some landmark points which can be used to tessellate the space in classes. As long as this tesselation corresponds to the tesselation obtained by using real images, the model is adequate for texture classification. Thus, we should view the process of estimating the Boolean model parameters of a binary texture as yet another way of producing features of this texture and the questions that we should be asking are the following: "How stable these features are for different realizations of the same texture?" and "Are they useful texture classifiers, by differing significantly more between the realizations of *different* textures than they differ between the realizations of the *same* texture?"

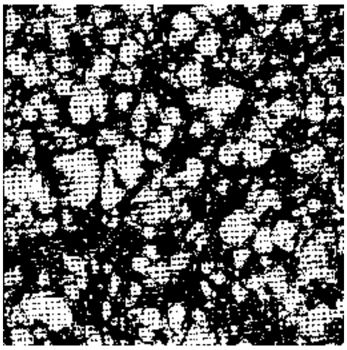
To answer the above two questions we chose 9 textures from [1] and we digitised them. The textures were chosen to be binary, stochastic or semi-stochastic and as representative as possible. There were among them some pairs of similar textures, as well as homogeneous and directional textures. Each image was digitised to a 256×256 size, and it was scanned in three different ways: Raster horizontal, raster vertical and Hilbert. Each time the Boolean parameters of the string created were estimated using the maximum likelihood estimation. Then each image was divided in four quarters and the parameters for each sub-image were estimated using again the three different scans. This was done so that we had more than one realizations of each texture. The mean and the standard deviation of each set of five parameters estimated were computed, and used as an indication of the expected parameter value and its stability over different realizations of the same texture.

The images with the calculated parameter values are presented in figures 4 to 6.

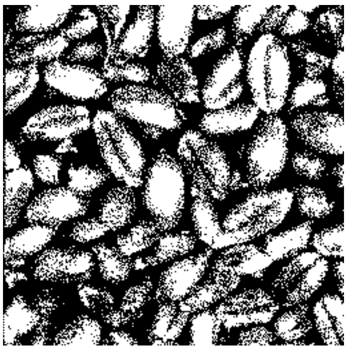
From the results presented in figures 4 to 6, we can see that the parameters calculated are quite stable over the realizations of the same texture and their difference from one texture to the next, even for similar textures is outside the range of their variation over the same texture. It seems therefore, from these results that the Boolean model in conjunction with the three modes of image scanning can be used for the classification of binary textures, to the extent that the textures used in our experiments are representative.



	Hilbert		Vert.		Horiz.	
	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$
Image	0.5204	1.442	0.5228	1.531	0.5219	1.504
SubImag 1	0.5949	1.493	0.5767	1.598	0.5729	1.557
SubImag 2	0.4951	1.427	0.5126	1.535	0.4926	1.547
SubImag 3	0.5093	1.479	0.5173	1.538	0.5270	1.509
SubImag 4	0.5195	1.392	0.5131	1.471	0.5224	1.438
Mean	0.5278	1.4466	0.5285	1.5346	0.5274	1.5110
σ	0.0347	0.0363	0.0244	0.0402	0.0258	0.0419

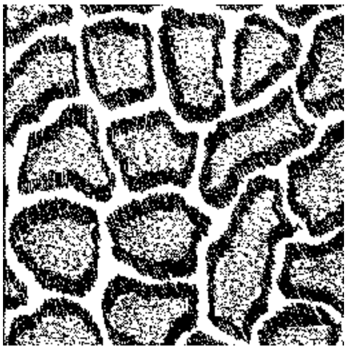


	Hilbert		Vert.		Horiz.	
	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$
Image	0.4185	1.609	0.4317	1.749	0.4016	1.880
SubImag 1	0.4310	1.623	0.4247	1.764	0.3969	1.830
SubImag 2	0.3855	1.597	0.4523	1.819	0.4081	1.972
SubImag 3	0.3917	1.583	0.4303	1.745	0.4056	1.892
SubImag 4	0.4509	1.635	0.4472	1.775	0.4021	1.927
Mean	0.4157	1.6094	0.4372	1.7704	0.4029	1.9002
σ	0.0245	0.0184	0.0106	0.0266	0.0138	0.0475

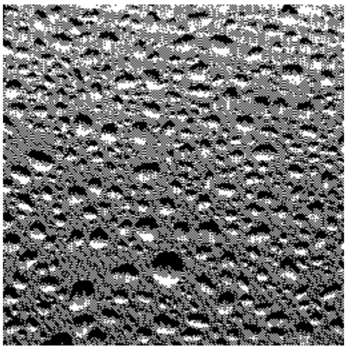


	Hilbert		Vert.		Horiz.	
	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$
Image	0.4190	1.379	0.4320	1.494	0.4380	1.485
SubImag 1	0.4251	1.363	0.4174	1.517	0.4374	1.499
SubImag 2	0.3996	1.403	0.4113	1.502	0.4793	1.494
SubImag 3	0.4071	1.386	0.4660	1.498	0.4115	1.521
SubImag 4	0.4208	1.354	0.4505	1.471	0.4619	1.463
Mean	0.4163	1.3770	0.4354	1.4964	0.4456	1.4924
σ	0.0122	0.0172	0.0204	0.0149	0.0232	0.0189

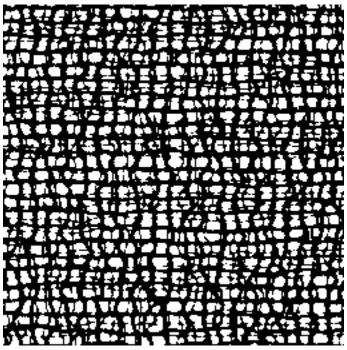
Figure 4: Some binary textures and their corresponding Boolean parameters estimated from the full image and its four quadrants separately.



	Hilbert		Vert.		Horiz.	
	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$
Image	0.3872	1.496	0.3841	1.499	0.3938	1.529
SubImag 1	0.3647	1.457	0.3565	1.461	0.3749	1.516
SubImag 2	0.3756	1.501	0.3774	1.520	0.4184	1.570
SubImag 3	0.4121	1.537	0.3950	1.572	0.3786	1.545
SubImag 4	0.4040	1.506	0.4005	1.555	0.4080	1.562
Mean	0.3887	1.4994	0.3827	1.5214	0.3947	1.5444
σ	0.0175	0.0256	0.0154	0.0396	0.0167	0.0200

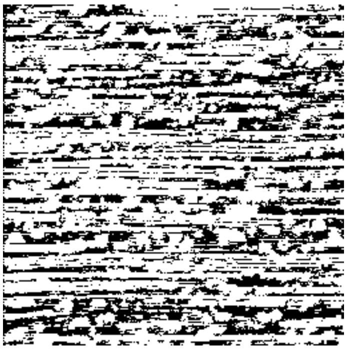


	Hilbert		Vert.		Horiz.	
	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$
Image	0.8105	1.158	0.8306	1.162	0.7831	1.184
SubImag 1	0.7574	1.116	0.7835	1.117	0.7272	1.147
SubImag 2	0.7626	1.117	0.7832	1.120	0.8322	1.225
SubImag 3	0.8535	1.196	0.8802	1.202	0.7283	1.144
SubImag 4	0.8705	1.194	0.8917	1.206	0.8476	1.222
Mean	0.8109	1.1562	0.8338	1.1614	0.7837	1.1844
σ	0.0460	0.0351	0.0461	0.0383	0.0504	0.0349

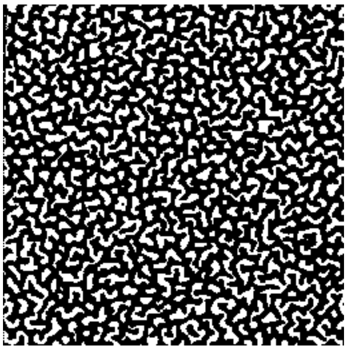


	Hilbert		Vert.		Horiz.	
	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$
Image	0.3775	1.930	0.3885	2.043	0.3844	2.108
SubImag 1	0.3840	1.906	0.4038	1.995	0.3321	2.211
SubImag 2	0.3989	1.904	0.3769	2.021	0.3713	1.974
SubImag 3	0.3623	1.885	0.3636	2.076	0.3459	2.247
SubImag 4	0.3823	1.906	0.3954	1.991	0.3516	2.055
Mean	0.3810	1.9062	0.3856	2.0252	0.3570	2.119
σ	0.0118	0.0143	0.0141	0.0316	0.0198	0.1000

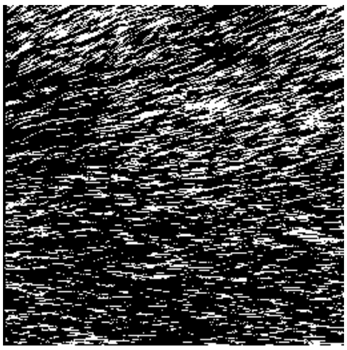
Figure 5: Some binary textures and their corresponding Boolean parameters estimated from the full image and its four quadrants separately.



	Hilbert		Vert.		Horiz.	
	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$
Image	0.2577	1.549	0.3365	1.453	0.1793	1.738
SubImag 1	0.2354	1.505	0.3080	1.441	0.1631	1.737
SubImag 2	0.2723	1.534	0.3563	1.442	0.1718	1.734
SubImag 3	0.2519	1.545	0.3288	1.493	0.1898	1.708
SubImag 4	0.2650	1.562	0.3524	1.464	0.1907	1.757
Mean	0.2565	1.5390	0.3364	1.4586	0.1789	1.7348
σ	0.0126	0.0192	0.0174	0.0191	0.0106	0.0157



	Hilbert		Vert.		Horiz.	
	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$
Image	0.3777	1.951	0.3813	2.786	0.3805	2.804
SubImag 1	0.3693	1.993	0.3805	2.771	0.3763	2.794
SubImag 2	0.3750	1.986	0.3847	2.846	0.3777	2.769
SubImag 3	0.3742	1.940	0.3766	2.758	0.3842	2.852
SubImag 4	0.3694	1.925	0.3853	2.806	0.3846	2.833
Mean	0.3731	1.9590	0.3824	2.7928	0.3807	2.8104
σ	0.0033	0.0263	0.0041	0.0306	0.0033	0.0292



	Hilbert		Vert.		Horiz.	
	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$	$\hat{p}$	$\hat{\theta}$
Image	0.6445	1.775	0.7989	1.685	0.4571	2.433
SubImag 1	0.6011	1.807	0.8217	1.694	0.5047	2.231
SubImag 2	0.6023	1.610	0.7530	1.535	0.4164	2.508
SubImag 3	0.6635	1.766	0.8501	1.811	0.4813	2.315
SubImag 4	0.6819	1.751	0.8503	1.761	0.4322	2.581
Mean	0.6407	1.7418	0.8148	1.6972	0.4583	2.4136
σ	0.0341	0.0684	0.0364	0.0933	0.0320	0.1267

Figure 6: Some binary textures and their corresponding Boolean parameters estimated from the full image and its four quadrants separately.

6 Discussion and conclusions

We have explored here the use of 1D Boolean model to describe binary textures. We have combined it with the Hilbert scanning of images to introduce some 2D properties to it. We have shown that according to the scanning format, the same model can produce very different looking textures. Thus, we advocate the use of several scanning formats to calculate the parameters of this model that can be used as texture descriptors.

Imposing a 1D model on a 2D image to describe a 2D property like texture, may seem ad hoc and wrong. However, although texture as perceived by the human eye is a spatial property, the image creating the perceived impression does not cease to be a signal and the issue is whether we can discriminate between these signals that could give rise to different textural perceptions when they are sequentially read into a lattice in certain ways. Thus, the models we use to describe textures are only seed points used in the hyper-space where we have as many axes as we have pixels in an image, and each image is nothing more than a point. When a new model is used, the relevant question to be asked is whether the versions of the model obtained by various combinations of its parameters impose a Delaunay tesselation of the hyper-space of textures so that in each cell only one texture is found. We have attempted to partially answer this question by considering 5 different realizations of the same texture, 9 different representative textures and calculated their parameters. Our preliminary results presented here, show that the intraclass variation of the parameters estimated is smaller than the interclass variations. This encourages us to investigate this approach further. In particular, we plan to investigate the stability of the parameters calculated when the realizations of each texture are restricted to sub-images of 20×20 or even 10×10 pixels in order to check the feasibility of using these parameters for texture segmentation as opposed to just for texture classification.

References

[1] Artfile Patterns, Phaidon, Oxford, 1990, ISBN 0 7148 2670 7.

[2] E. R. Dougherty and J. C. Handley, 1995. "Recursive maximum-likelihood estimation in the one-dimensional discrete Boolean random set model", *Journal of Signal Processing*, vol. 43, no. 1.

[3] J. C. Handley, E. R. Dougherty, 1995. "Binary texture estimation using linear samples", in Machine Vision Applications in Industrial Inspection III, SPIE Vol 2423.

[4] S Kamata, P Arnulfo and E Kawaguchi, 1991. "A method of computing Hilbert curves in two and three dimensional spaces", Trans. IEICE, vol. J74-D-II, No 9, pp 1217–1226.

[5] S Kamata, R O Eason and E Kawaguchi, 1993. "An efficient Hilbert scanning algorithm and its application to data compression", SCIA'93, Vol II, pp 1333–1339.