Texture synthesis-by-analysis method based on a multiscale early-vision model

Javier Portilla

Rafael Navarro

Oscar Nestares

Instituto de Optica (CSIC)

Serrano 121

28006 Madrid, Spain

E-mail: javier@pixar.optica.csic.es

Antonio Taberno

Universidad Politécnica de Madrid

Facultad de Informática

Boadilla del Monte

28660 Madrid, Spain

E-mail: ant@asterix.fi.upm.es

Abstract. A new texture synthesis-by-analysis method, applying a visually based approach that has some important advantages over more traditional texture modeling and synthesis techniques is introduced. The basis of the method is to encode the textural information by sampling both the power spectrum and the histogram of homogeneously textured images. The spectrum is sampled in a log-polar grid using a pyramid Gabor scheme. The input image is split into a set of 16 Gabor channels (using four spatial frequency levels and four orientations), plus a low-pass residual (LPR). The energy and equivalent bandwidths of each channel, as well as the LPR power spectrum and the histogram, are measured and the latter two are compressed. The synthesis process consists of generating 16 Gabor filtered independent noise signals with spectral centers equal to those of the Gabor filters, whose energy and equivalent bandwidths are calculated to reproduce the measured values. These bandpass signals are mixed into a single image, whose LPR power spectrum and histogram are modified to match the original features. Despite the coarse sampling scheme used, very good results have been achieved with nonstructured textures as well as with some quasi-periodic textures. Besides being applicable to a wide range of textures, the method is robust (stable, fully automatic, linear, and with a fixed code length) and compact (it uses only 69 parameters). © 1996 Society of Photo-Optical Instrumentation Engineers.

Subject terms: texture synthesis; Gabor channels; multiscale image representations.

Paper 02105 received Oct. 1, 1995; revised manuscript received Feb. 13, 1996; accepted for publication Feb. 13, 1996.

1 Introduction

Texture synthesis is necessary for the generation of realistic images from a collection of compressed data. Synthesis-by-analysis (S-A) methods are used to extract a set of significant features from a homogeneously textured image, which is subsequently used to produce an image with the same visual appearance (see Fig. 1).

Traditionally, two main approaches have been taken to study visual texture: the structural model and the statistical one.¹ Although real textures are neither completely structured nor entirely stochastic, purely statistical approaches, such as the one used in this work, are suitable for modeling many real textures.

Most people would agree that an ideal texture synthesizer should fulfill the following requirements:

1. Wide field of application. The synthesizer should be applicable to a wide range of different input textures. Furthermore, the synthesis quality should not be affected either by changes in the spatial scale, within a reasonable range, or in the orientation of the input image (isotropy of the model).
2. Compactness. Data compression (i.e., using a small set of parameters to describe texture) is one of the main purposes of the S-A methods.

3. Low computational cost. Practical applications require efficient algorithms.
4. Full automatism. Some S-A methods demand human supervision at some of their stages, and consequently they could not compete, in a practical environment, with fully automatic ones.
5. Robustness. Low sensibility to quantization noise in the parameters space, stability, input-independence of the processing, etc., are important and desirable features.

Even confining the study to nonstructured, gray level textures, it turns out that no current S-A method complies with all these demands. As shown in Ref. 2, only those models based on the autocorrelation function of the texture are able to provide a practical synthesis method that can be applied to a wide range of input texture classes. This finding agrees with the fact that first and second order statistics are the most important features to describe purely stochastic textures for the human observer.³ Gibbs samplers, Markov chains, fractal models, etc., although interesting from a theoretical point of view and for some applications, have proven to be unsuitable for generic texture S-A methods. Unfortunately, even the autocorrelation-based methods have important drawbacks. Next we summarize the main

Fig. 1 Block diagram of the texture S-A method and its goal: to achieve the same visual appearance in both the original and the synthetic textures.

problems of the two most accepted models of this class: the autocorrelation and histogram (AC-H) model and the autoregressive (AR) model.

Usually in the AC-H model, gradient algorithms are used to simultaneously impose the autocorrelation function and the histogram (or a related set of first order statistic parameters) (Refs. 4 and 5) to a random 2-D signal. These algorithms are highly time consuming when applied to realistic images (e.g., 256×256 pixels and 256 gray levels). In both models, a selection of the significant values of the parameter set must be carried out to keep the processing time within a reasonable bound, as well as to achieve some data compression. This step is difficult to automatize reliably, and makes the amount of processing to depend on the input texture. Moreover, textures that can be well modeled with few parameters using an AC-H model require many more parameters when modeled as AR signals, and vice versa. Another problem is that synthetic textures generated applying gradient methods using the AC-H model present a noisy aspect (see Ref. 2 for instance), due to the high frequencies contained in the noise seed take a long time to converge. Some drawbacks of the AR models are (1) they require an initial set of pixels (in causal models); (2) they do not directly control the first order statistics; (3) the texture visual features are very sensitive to the AR filter order and to small deviations in its values; and (4) the stability of the AR filters is not guaranteed.

The basic idea of the method proposed here is to use *a priori* information about the human visual system (HVS) to develop a compact and visually efficient sampling of the autocorrelation function, through a multiscale Gabor representation.⁶ This kind of visual model has shown to be very useful in texture analysis (see, for instance, Refs. 7 to 9). However, up to now, much less work has been done in applying visual models to texture synthesis,¹⁰ and as far as we know, our method is the first one proposed in this context based on a fully automatic S-A scheme.¹¹ The proposed method is robust, it uses noniterative simple operations as linear filtering and histogram matching, and,

Fig. 2 Frequency representation of the multiscale Gabor scheme used. The filled circles represent the bandwidths of the Gabor filters at half-peak height. There are four frequency levels distributed by octaves and four orientations (vertical, horizontal and the two diagonals). A low-pass residual channel, in the center, is needed to cover the very low frequencies.

although it is far from an accurate parametric model, it provides good results over a wide range of input textures.

This paper is organized as follows. In Section 2 we review the mentioned multiscale Gabor image representation scheme, and then we introduce our S-A method, based on that scheme. Sections 3 and 4 describe the feature extraction and synthesis stages, respectively. Results are presented and discussed in Section 5. Finally, Section 6 summarizes the distinctive features of the texture S-A method. In addition, an Appendix is included that explain how the synthetic channel bandwidths and amplitudes are calculated from the set of the input texture's extracted features.

2 Model

2.1 Multiscale Gabor Scheme

In a previous work,⁶ a multiscale model was proposed that schematically imitates the visual coding in the early stages of the HVS, by simply applying a set of 4×4 (four frequency levels, four orientations) Gabor filters to the digital images. This particular scheme is suitable for a fast pyramid implementation in either the spatial or the spatial-frequency domains.^{6,11} It has been successfully applied to a variety of tasks involving local multiscale processing, such as spatially variant image restoration¹² and fusion¹³ as well as to texture analysis⁹ (segmentation and classification).

In contrast with the mentioned tasks, the coding and synthesis of homogeneous single-texture images can be done by purely global operations, using a set of global descriptors. In this case, we do not take direct advantage of the optimal joint resolution of Gabor filters, but of the visually adapted power spectrum sampling of the scheme used (see Figs. 2 and 3), as well as of the good properties of Gaussian functions.

Fig. 3 Radial slice of the Gabor multiscale scheme in the frequency domain. The overlapping between adjacent filters has been enhanced by shading.

Only the real, even part of zero-phase Gabor functions has been used (working with real signals, we can obtain the imaginary part of a Gabor filtered image as the Hilbert transform of its real part). The pyramidal structure and the symmetries, zero-crossing samples and separability of the scheme filters facilitates their very efficient implementation.¹⁴

A 2-D Gabor function centered at the spatial origin can be expressed as

$$g(x, y)_{f, \theta, \alpha, \gamma, \psi} = \exp\{-\pi a^2[(x \cos \theta + y \sin \theta)^2 + \gamma^2(y \cos \theta - x \sin \theta)^2]\} \times \exp[i2\pi f(x \cos \theta + y \sin \theta) + i\psi]. \quad (1)$$

In our scheme, each Gabor filter has $\gamma=1$ (circular symmetry) and $\psi=0$. For each different resolution level p , the radial frequency f is

$$f_p = 0.25 \cdot 2^{p-4} \text{ cycles/pixel for } p=1..4 \quad (2)$$

and for each orientation q , the angles θ are

$$\theta_q = (q-1) \frac{\pi}{4} \text{ rad for } q=1..4. \quad (3)$$

The factor a depends on the resolution level and it is proportional to the radial frequency:

$$a_p = k f_p \text{ for } p=1..4 \quad (4)$$

with the constant k being

$$k = \frac{1}{3} \left(\frac{\pi}{\ln 2} \right)^{1/2}. \quad (5)$$

In this way, the radial and angular bandwidths are constant on a log-polar scale:

$$B_r = \log_2 \left[\frac{f + a(\ln 2/\pi)^{1/2}}{f - a(\ln 2/\pi)^{1/2}} \right] = 1 \text{ octave}, \quad (6)$$

Table 1 Number of parameters used for each one of the four sets of extracted features.

Gabor Channel Energies	Gabor Channel Equivalent BWs	Averaged LPR frequency moduli	Compressed histogram
16	32	5	16

$$B_a = 2 \tan^{-1} \left[\frac{\gamma a (\ln 2/\pi)^{1/2}}{f} \right] = 36.87 \text{ deg.} \quad (7)$$

2.2 General Description of the Model

This model is focused on nonstructured, gray level textures. Texture is modeled as a 2-D random field, which enables a purely statistical treatment. Its power spectral density (or, equivalently, its autocorrelation function) and its probability density function are estimated by the power spectrum and the histogram of the textured image, respectively. Trying to simultaneously impose first and second order statistics to a random signal is highly expensive, and it requires the use of iterative algorithms that minimize a global error function.⁴ Instead of that, a much faster sequential approach has been followed here, consisting of first imposing the second order and then the first order statistics. Although nonexact, this method provides good visual results (see Ref. 15, for instance).

The set of 16 Gabor filters of Fig. 2 samples the Fourier domain in a log-polar scheme, using four directions (horizontal, vertical and the two diagonals) and four radial frequencies (distributed by octaves). The result of applying one of these filters to the input image is called a Gabor channel. The very low frequencies of the image are covered by an additional low-pass residual (LPR) channel. The energy and equivalent bandwidths (BW) along the u and v frequency axes of the Gabor channels [(1+2)×16 parameters], plus the compressed version of the modulus of the LPR frequencies (5 parameters), provide a rough approach, though very visually efficient, to the power spectral density of the texture. Finally, the 256 gray level histogram is measured and compressed to just 16 values. Thus, we use only 48+5+16=69 parameters to characterize the input texture (see Table 1).

Demodulation has been applied to the Gabor channels, thus enabling a reduction in their number of samples by a factor of 4 (by subsampling in a factor of 2 in each dimension). However, the resulting channels become complex, so that the real compression ratio is 4/2=2. Such a compression increases the efficiency of the subsequent processes.⁸ This method is symmetrically applied in the synthesis stage: the half-size, complex low-pass synthetic channels, become full size, real bandpass synthetic channels, through their expansion and modulation.

The basic idea of the synthesis algorithm is to generate synthetic channels similar to the Gabor channels of the original texture, and to construct the synthetic texture by merging them. The synthesis algorithm starts with the generation of a set of 16 independent complex white noise 2-D discrete signals. Each noise signal is convolved with a Gaussian filter. The two BWs (along the two frequency axes) and the amplitude of each one of these filters are calculated to obtain the same energy and equivalent BW

Fig. 4 General scheme of the texture S-A method.

values in the Gabor channels of the synthetic texture as those measured in the original texture's Gabor channels. (It is important to note the difference existing between the synthetic channels, as independently generated bandpass noise signals that try to imitate the Gabor channels of the original texture, and the Gabor channels of the synthetic texture, which are the result of applying the multiscale Gabor filtering scheme to the synthetic texture.) Then, the filtered noise signals (1) are spatially expanded by a factor of 2 in each dimension, (2) each one is modulated to its corresponding central frequency of the multiscale Gabor scheme, and (3) they are mixed together into a single image. The moduli of the LPR frequencies of this image are substituted by the decompressed version of the previously averaged original values. Finally, the original histogram (again, its decompressed version) is imposed to the resulting image. The entire S-A method is summarized in the block diagram of Fig. 4.

3 Feature Extraction

In this section we give a detailed description of the parameter extraction process. Four sets of parameters are extracted from the input image: (1) the energy and (2) the equivalent BWs of its Gabor channels; (3) the modulus of its LPR frequencies; and (4) the histogram. The two last sets are compressed.

In the notation used, a subindex 0 means that the referred signal comes from demodulating a bandpass signal (at the feature extraction stage), or is a low-pass signal that has not yet been modulated (at the synthesis stage).

3.1 Gabor Filtering and Demodulation

The Gabor channels are obtained by convolving the input image with the set of 4×4 Gabor masks. Using Eqs. (1) to (4), the even Gabor filter of frequency level p and orientation q can be expressed as

$$g_{pq}(x, y) = 2a_p^2 \operatorname{Re}[g(x, y)_{f=f_p, \theta=\theta_q, a=a_p, \gamma=1, \psi=0}], \quad (8)$$

having been scaled so that its spectral maximum has unity response. Then the even Gabor channels are

$$t_{pq}(x, y) = t(x, y) * g_{pq}(x, y), \quad (9)$$

where $t(x, y)$ is the textured input image, p and q vary from 1 to 4, and the symbol $*$ means convolution. (Note that using a multiresolution scheme only one filter is actually needed for each orientation. This is applied to shrunk versions, by factors 1, 2, 4 and 8, of the input image. For simplicity, the multiresolution approach has not been reflected in the mathematical expressions.) The filters have been designed in the spatial domain, applying the efficient techniques described in Ref. 14. The chosen size has been 13×13 .

It is easy to prove that the demodulated complex channels can be obtained by

$$t_{0_{pq}}(x, y) = 2\{t_{pq}(x, y) \exp[-i2\pi f_p(x \cos \theta_q + y \sin \theta_q)]\} * h_{LP}(x, y), \quad (10)$$

where $h_{LP}(x, y)$ is an ideal low-pass filter with cutoff frequency $f_c = 0.25$. Such a low-pass filter is applied to remove the high frequency terms before subsampling the demodulated channels by a factor of 2 in each dimension. As the filtering, the demodulation process can be done in a very efficient way thanks to the pyramidal scheme used, which enables using only one pair of quadrature grids (a complex exponential) for the four different frequencies of each orientation (this applies also to the modulation at the synthesis stage).

3.2 Energy of the Gabor Channels

The energy of the Gabor channels provides information about the main directions and levels of detail of the texture. For this task, a logarithmic distribution of the filters in the spatial frequency domain is visually optimal (see, for instance, Ref. 16). This set of 4×4 parameters gives a very compact first approximation to the power spectral density of the texture. The mean square value of each complex demodulated channels is calculated by

$$e_{pq} = \frac{1}{N_p^2} \sum_{x=0}^{N_p-1} \sum_{y=0}^{N_p-1} |t_{0_{pq}}(x, y)|^2, \quad (11)$$

where N_p is the number of samples of the demodulated channel in each dimension (dependent on the resolution level p). We term $\mathbf{e}_o^a$ the vector constituted by the energy values of the 16 Gabor channels (indices a and o mean that the vector has been obtained by analyzing the original texture image). Table 2 shows the root-mean-square (rms) values, normalized by each corresponding mean, for three different textures. As can be seen, such information is very descriptive of the main orientations and spatial scales of textures. Note that the angular values are referring to the channel orientations in the frequency domain.

3.3 Equivalent Bandwidths of the Gabor Channels

The information provided by the energy of the Gabor channels is highly significant, but it is still far from being visually complete. A Gabor channel that comes from filtering

Table 2 Percentage ratio between the rms values of the Gabor channels and the mean gray-level value, for three Brodatz' textures.

	Water				Sand				Wood			
	0 deg	45 deg	90 deg	135 deg	0 deg	45 deg	90 deg	135 deg	0 deg	45 deg	90 deg	135 deg
<i>f1</i>	3.5	3.9	2.9	2.2	5.3	6.3	4.6	6.0	8.3	3.2	6.7	3.1
<i>f2</i>	5.3	9.9	6.4	5.8	9.3	8.5	9.0	11.3	14.8	3.6	2.3	3.9
<i>f3</i>	2.9	6.9	23.3	6.3	12.1	11.1	12.1	14.0	24.5	3.3	2.4	3.6
<i>f4</i>	3.3	4.2	15.6	3.8	10.8	9.2	10.6	11.1	21.1	3.3	2.8	3.3

an image having a single frequency is visually very different from another with the same energy coming from a white noise image. We have experienced that the degree of spectral spreading of the power spectrum at different spectral locations is an essential feature to visually characterize the texture. The presence in the power spectrum of peaks and straight radial lines is reflected in the image by the existence of regular spatial intervals and straight lines along principal directions, features which are very significant to the HVS.

Different approaches are found in the literature to mathematically describe these features. Porat and Zeevi,¹⁰ also using a Gabor expansion, measured the variance of the local spatial frequencies (in radial frequency and orientation). Francos et al.¹⁷ used a 2-D Wold-like decomposition to separate, in the frequency domain, the deterministic from the random texture components. Basically, what they did was detecting frequency peaks and radial straight lines in the original image spectrum and reproducing them (modulus and phase) in the synthetic image spectrum, together with an AR modeled random component.

Our approach consists of measuring the equivalent BWs (as defined later) of each demodulated Gabor channel along the u and v frequency axes. (We have adopted this solution in the current implementation for simplicity, although a measurement along the radial and angular directions would provide a higher degree of isotropy to the S-A method. This improvement is included in further implementations.) Using a small set of parameters ($2 \times 16 = 32$), the peaks and straight radial lines of the original image power spectrum can be detected and reproduced at the synthesis stage, modeling the Gabor channels as bandpass signals with a power spectrum of elliptical Gaussian shape of adjustable BWs. Although this method is not exact (due to the limited spectral resolution of our sampling scheme, e.g., several frequency peaks inside a Gabor channel are not resolved), it provides, in most cases, a high degree of visual resemblance between the original and the synthetic textured images.

As a measure of the equivalent BW of a 1-D signal, we have computed the area of its normalized power spectrum (in a way similar to Ref. 18). To apply this to a demodulated Gabor channels, we need to convert its 2-D power spectrum into a pair of 1-D normalized power spectra. This is achieved by integrating (adding, in the discrete form) their power spectrum along the two frequency axes, and dividing the result by their respective maxima:

$$P_{0_{pq}}(u, v) = |T_{0_{pq}}(u, v)|^2 = |\text{DFT}[t_{0_{pq}}(x, y)]|^2; \quad (12)$$

$$P_{u0_{pq}}(u) = \sum_v P_{0_{pq}}(u, v); \quad \bar{P}_{u0_{pq}}(u) = \frac{P_{u0_{pq}}(u)}{\max(P_{u0_{pq}}(u))}; \quad (13)$$

$$P_{v0_{pq}}(v) = \sum_u P_{0_{pq}}(u, v); \quad \bar{P}_{v0_{pq}}(v) = \frac{P_{v0_{pq}}(v)}{\max(P_{v0_{pq}}(v))}; \quad (14)$$

where DFT is a discrete Fourier transform. Their areas are calculated by integrating once again:

$$S_{u_{pq}} = \sum_u \bar{P}_{u0_{pq}}(u) \quad (15)$$

$$S_{v_{pq}} = \sum_v \bar{P}_{v0_{pq}}(v). \quad (16)$$

It is easy to prove that, disregarding the small effects of the discretization in the calculus and assuming that the spectral values at the highest frequency along both axes are almost zero (i.e., a low degree of spectral aliasing), a 2-D Gaussian signal with the following mathematical expression, or any other one with equal power spectrum:

$$g_{eq_{pq}}(x, y) = C \exp[-2\pi(u_{pq}^2 x^2 + v_{pq}^2 y^2)] \quad (17)$$

(where C is an arbitrary constant), would yield the same $S_{u_{pq}}$ and $S_{v_{pq}}$ values as the original (p, q) Gabor channel. This signal has half-peak BWs in the frequency axes directions

$$B_{u_{pq}} = \left(\frac{2 \ln 2}{\pi} \right)^{1/2} S_{u_{pq}} \quad (18)$$

and

$$B_{v_{pq}} = \left(\frac{2 \ln 2}{\pi} \right)^{1/2} S_{v_{pq}}, \quad (19)$$

which can be seen as the equivalent BWs of the (p, q) Gabor channel. These values (1) parametrize the spectral

Fig. 5 Lowest level resolution Gabor channels and the squared support considered for the LPR channel.

shape of the input texture Gabor channels and (2) are used to calculate the BWs of the synthesis filters, at the synthesis stage.

3.4 DFT Modulus in the LPR Channel

The values of the autocorrelation function at very low frequencies, although often not considered as texture descriptors, are nevertheless necessary for a complete visual description of many real textured images. Therefore, we have parametrized the Fourier modulus of the LPR channel. Searching a trade-off between a low overlapping with the lowest resolution Gabor channels (so as to have the least influence on their statistical features at the synthesis stage) and an adequate covering of the very low frequencies, a squared spectral region containing the frequencies lower than $1/64$ cycles/pixel in each dimension has been chosen for the LPR channel. Naming the DFT of the input image:

$$T(u, v) = \text{DFT}[t(x, y)], \quad (20)$$

the LPR spectral moduli chosen are

$$T_r(u, v) = |T(u, v)| \quad \text{for } |u|, |v| < \frac{1}{64} \text{ cycles/pixel}. \quad (21)$$

Figure 5 represents the four lowest resolution level even Gabor channels (at their half-peak height) and the squared support of the LPR frequencies considered.

In real signals, only about a half of the LPR discrete frequencies have independent values. For the image size used in the current implementation ($N=256$), there are $[(N/32-1)^2+1]/2=25$ independent values. This number has been considered too high, regarding the relatively small visual importance of this set of parameters. Thus, it has been reduced to just five parameters, by taking the average value within each considered area (see Fig. 6). One of these values correspond to the dc component, another two values are the averages along the two axes (excluding the dc) and finally, two more values are computed for the oblique fre-

Fig. 6 Five LPR regions of averaged frequency samples for $N=256$. The averages of the spectrum modulus over the five depicted areas are used as the descriptors of the LPR channel.

quencies' regions. This nonuniform spectral sampling has been done considering the different visual importance of each set of frequencies.

3.5 Histogram

Even an exact adjustment of the synthetic image power spectrum to that of the original image would not yield a high visual resemblance between the two images, since their first order statistics will be very different in general. First order statistics are fundamental to characterize visually the textures,³ and so the histogram of the input image (or a set of first order statistics parameters, as range, mean and variance) should be considered in a S-A method.

Typical gray level digitized images use 1 byte/pixel, resulting in a histogram of 256 values. This number can be drastically reduced by a proper coding, without important losses in its visually significant information. We have empirically tested that the fine detail of the histogram is usually not visually relevant, and that 16 values are more than enough to represent the key features of the first order statistics of most real textures (this is in agreement with Ref. 19).

The compression method used consists of low-pass filtering the histogram, followed by subsampling. Special care must be taken to preserve the extreme values of the histogram. These are often very significant, because they accumulate the gray level values that are out of the dynamic range of the machine used to capture the image. For preserving the extreme values of the histogram, as well as for avoiding edge artifacts when filtering, an opposite specular reflection has been applied to the histogram edges (see Fig. 7). It is easy to prove that the extreme values of a 1-D signal that has been lengthened using this technique remain unchanged when an even-symmetric mask, with unity response to the zero frequency and number of samples less than twice the edge width considered, is used to filter it.

Fig. 7 Graphical explanation of the edge lengthening (opposite specular reflection) applied to preserve extreme values and to avoid edge artifacts when low-pass filtering the histograms.

3.6 Number of Parameters versus Image Size

The number of parameters used in the feature extraction are shown in Table 1. In addition to the small number of parameters (69) comparing with other methods (to properly compare we have to regard also the size and number of gray levels of the images used), another interesting feature of this S-A method is the independence of this number from the spatial dimensions of the input image. The use of image sizes other than $N=256$ affects the computational cost of the process, but does not change the number of parameters to be extracted, as the number of Gabor filters used remains the same. Its only effect on the quality of the synthesis is a reduction, in the case of enlarging the images, of the LPR channel's spectral resolution (due to the spectral modulus being averaged in larger areas), which is not visually important for homogeneously textured images. This effect is like seeing a textured image through a growing window from a fixed distance: as the window grows, lower frequencies appear, but they are less and less perceivable, and can be characterized with less parameters, whereas the other perceived features do not change. Only if we go farther from the texture (i.e., if we shrink the image to a smaller visual angle), the very low frequencies become higher and more visually significant, but, in return, we lose the high frequency content (more important in normal textures). This would correspond to applying the synthesizer to a reduced version of the image. In both cases we would use the same number of parameters, but the first solution generally yields better visual results.

4 Synthesis Procedure

The synthesis process is carried out through seven sequential stages (see Fig. 4): (1) noise generation; (2) Gaussian filtering of the noise signals; (3) weighting of the filtered noise signals; (4) modulation of the weighted filtered noise signals, which become into the synthetic channels; (5) merging the synthetic channels into a single image; (6) equalization of the LPR frequencies; and (7) adjustment of the histogram.

The synthetic textured image can be expressed as

$$\tilde{r}(x,y) = \varphi \left[\sum_{q=1}^4 \sum_{p=1}^4 s_{pq}(x,y) + r(x,y) \right], \quad (22)$$

where $s_{pq}(x,y)$ is the synthetic channel of frequency p and orientation q , $r(x,y)$ represents the synthetic LPR channel,

and φ is the nonlinear, monotonously increasing function that corrects the histogram of the synthetic texture.

4.1 Noise Generation

Sixteen independent signals of complex white Gaussian noise are generated, one for each Gabor channel. Their DFT moduli have been forced to be constant at all frequencies to avoid random effects in their power spectrum overlapping with the Gabor channels and also to simplify the subsequent mathematical expressions. These signals are complex because each one is the seed of a synthetic low-pass channel, which, similarly to the complex demodulated Gabor channels, become complex when placed around the zero frequency. Their mean is zero and their variance σ^2 is a value adapted to the dynamic range of the synthesizer.

4.2 Gaussian Filtering: Adjustment of the Channel Equivalent BWs

The computer generated noise signals are convolved with 2-D elliptical Gaussian masks (separable), which provide them an elliptical Gaussian spectral shape. The synthetic channel of resolution level p and orientation q , before being modulated, can be expressed as

$$s_{0_{pq}}(x,y) = k_{pq} n_{pq}(x,y) * g_{s_{pq}}(x,y), \quad (23)$$

where $n_{pq}(x,y)$ is a white noise 2-D discrete signal of variance σ^2 , k_{pq} is the factor that controls the energy of the (p,q) synthetic channel, and

$$g_{s_{pq}}(x,y) = b_{u_{pq}} b_{v_{pq}} \exp \left[-\pi (b_{u_{pq}}^2 x^2 + b_{v_{pq}}^2 y^2) \right] \quad (24)$$

is an elliptical Gaussian filter with their axes parallel to the u and v frequency axes, and having unity response to the zero frequency. The factors $b_{u_{pq}}$ and $b_{v_{pq}}$ control the respective BWs of the (p,q) synthetic channel. These BWs must be adjusted in such a way that, when the Gabor filtering scheme is applied to the synthetic texture, the resulting equivalent BWs of its Gabor channels have equal values to those measured in the input image. However, due to the overlapping between the channels, the exact calculation of these BWs is difficult. A first rough approximation consist of disregarding the channels' overlapping. Using this approximation, the equivalent BWs of the synthetic channels are given by the equivalent BWs of their corresponding Gabor channels, plus a constant that compensates for the BW narrowing effect caused by the Gabor filters used at the analysis stage (this calculation is done in Appendix Section 7.1). The error introduced by this approximation, although not mathematically negligible, does not seem to affect significantly the visual quality of the results, as shown in Section 5.2.

Instead of permitting a continuous range to the BW values, a small set of discrete BW values were chosen for both the parametrization and the synthesis processes. As explained in the Appendix (Section 7.2), the calculation of the synthetic channels' energies is much more reliable if we measure the spectral spreading of a discrete set of synthesis filters on to the Gabor channels than if we apply a theoretical expression [Eq. (52)] of this spreading, which is valid only for perfectly Gaussian filters. The results have shown

Fig. 8 Pyramidal mixing of the synthetic channels, including the equalization of the LPR frequencies.

that five quantization levels (proportionally adapted to each resolution level, thanks to the multiresolution scheme) are enough to provide a good characterization, in visual terms, of the Gabor channels' spectral shapes. These values have been chosen by applying 1-D optimal quantization (using the k -means algorithm) to 364 equivalent BW values obtained from 12 different textures. Due to the limited spatial support used for the filters (13×13), the narrowest BW synthesis filters are not really Gaussian and present some ringing in their power spectrum [see Figs. 11(e) and 14(e) in Sec. 5.2]. However, these artifacts have little visual repercussion.

4.3 Weighting Factors: Adjustment of the Channel Energy

The factor k_{pq} of Eq. (23) controls the energy of the associated synthetic channel. To achieve the same mean squared values in the Gabor channels of the synthetic texture as those measured in the original one, it is necessary to mathematically model the energy contribution of each synthetic channel to each Gabor channel. This can be done by

1. calculating or measuring the proportion of energy that each synthetic channels spread over each one of the Gabor channels (only the contribution to neighbor Gabor channels will be significant)
2. expressing the total energy measured in each Gabor channel as a sum of energy contributions from all the synthetic channels. Each energy contribution can, in turn, be expressed as the product of the energy of the synthetic channel by a weighting factor (the spectral overlapping coefficient; these weighting factors are arranged in the overlapping matrix $\mathbf{R}$)
3. obtaining the synthetic channel energies as the product of the measured energy vector by the inverse of the overlapping matrix
4. adjusting the energy of the synthetic channels to the calculated values (this is done by scaling their samples by the k_{pq} factors).

This linear calculus is possible since the synthetic channels are statistically independent, and so, the energy of their sum equals the sum of their energies. The energy adjusting process is explained in detail in the Appendix, Sec. 7.2.

4.4 Modulation

In a symmetric way as done at the feature extraction stage, at the synthesis stage the low-pass filtered noise signals are

expanded by a factor of 2 in each spatial dimension before being modulated. Each filtered and weighted complex noise signal is modulated by the central frequency corresponding to its indices (p, q) , and its real part (multiplied by 2 to have unity response in its frequency maxima) is extracted, thus obtaining the synthetic channel:

$$s_{pq}(x, y) = 2 \operatorname{Re}\{s_{0_{pq}}(x, y) \times \exp[i2\pi f_p(\cos \theta_q x + \sin \theta_q y)]\}. \quad (25)$$

4.5 Merging the Synthetic Channels

The mixing of the synthetic channels simply consists of their summation into a single image:

$$\tilde{r}_1(x, y) = \sum_{p=1}^4 \sum_{q=1}^4 s_{pq}(x, y). \quad (26)$$

Since we have applied a multiscale representation, this addition is efficiently done in a pyramidal way (see Fig. 8), adding up the four synthetic channels of the lowest resolution level and spatially stretching the result by a factor of 2, which in turn is added to the four synthetic channels of the next resolution level, and so on, until we add the highest frequency channels.

4.6 Equalization of the LPR Channel

The LPR-equalized version of the synthetic texture can be expressed as a sum of the unequalized version $\tilde{r}_1(x, y)$ [see Eq. (26)] plus a synthetic low-pass residual $r(x, y)$:

$$\tilde{r}_2(x, y) = \tilde{r}_1(x, y) + r(x, y). \quad (27)$$

This equalization is done in the frequency domain. First, the five average values of the LPR frequency moduli obtained at the feature extraction stage are decompressed, by merely replicating them in their respective spectral areas. The resulting square of spectral moduli is imposed on the lowest frequencies of the synthetic mix obtained before, keeping the phase unchanged:

$$\tilde{T}_1(u, v) = \operatorname{DFT}[\tilde{r}_1(x, y)] = |\tilde{T}_1(u, v)| \exp[j\psi_{\tilde{T}_1}(u, v)], \quad (28)$$

$$\tilde{T}_2(u, v) = M(u, v) \exp[j\psi_{\tilde{T}_1}(u, v)], \quad (29)$$

where, terming $T'_r(u,v)$ the reconstructed version (the result of having been compressed and decompressed) of the original $T_r(u,v)$ [see Eq. (21)]:

$$M(u,v) = \begin{cases} T'_r(u,v), & \text{for } |u|, |v| < \frac{1}{64} \\ |\tilde{T}_1(u,v)|, & \text{for } |u|, |v| \geq \frac{1}{64} \end{cases} \quad (30)$$

Thus, the LPR-equalized version of the synthetic texture is

$$\tilde{r}_2(x,y) = \text{DFT}^{-1}[\tilde{T}_2(u,v)]. \quad (31)$$

Therefore, using Eq. (27), the spectrum of $r(x,y)$ is

$$R(u,v) = \text{DFT}[r(x,y)] = \begin{cases} [T'_r(u,v) - |\tilde{T}_1(u,v)|] \exp[j\psi_{\tilde{T}_1}(u,v)], & \text{for } |u|, |v| < \frac{1}{64} \\ 0, & \text{for } |u|, |v| \geq \frac{1}{64}. \end{cases} \quad (32)$$

To efficiently carry out this operation, it is advantageous to mix the synthetic channels of the lowest resolution level and to modify the central part of its spectrum before going on with the pyramidal mixing (see Fig. 8). In such a way, two $N/8 \times N/8$ size fast Fourier transforms (FFTs) are performed, instead of the two $N \times N$ FFTs that would be necessary if this operation were done on the entire image.

4.7 Histogram Matching

First, the compressed original histogram is decompressed to its former size by expanding and low-pass filtering it (it must be adjusted so that its sum yields exactly N^2). Then, the histogram matching method (see Ref. 20, for instance) is used to modify the Gaussian-like histogram of the equalized mix of synthetic channels $\tilde{r}_2(x,y)$, reshaping it into the decompressed version of the original one. The adjustment can be expressed as the application of a nonlinear function φ , obtained from the actual and the desired histograms:

$$\tilde{r}(x,y) = \varphi[\tilde{r}_2(x,y)]. \quad (33)$$

This last step of the synthesis process strongly improves the resemblance of the synthetic texture to the original one. Analytically, however, it introduces a scale factor (typically ranging from 1 to 2) in the mean-squared value of the synthetic texture's Gabor channels. Nevertheless, the almost perfect invariance of this factor for all the channels (empirically verified) makes this change unimportant in visual terms. (We have not found a mathematically rigorous reason for this invariance. However, if the local frequency modulus responses, in a joint space-spatial frequency representation as the Gabor one, are uncorrelated with the local gray level, which it seems to be true for the homogeneous stochastic textures, it could be concluded that all the frequencies will be equally affected, in statistical terms, by the gray level-dependent gain caused by the application of a nonlinear function.) Note that the global energy of the synthetic texture is not affected by this scaling factor, as it is forced by the histogram adjustment to be equal as the original energy value. The scaling factor can be explained by regarding the larger amount of spectral energy around

the principal frequencies of the scheme that the synthetic texture has with respect to the original, as a consequence of the synthetic channels being located around these central frequencies.

4.8 Synthesis Equation

In summary, the synthetic texture can be mathematically expressed as a function of several parameters, which have been calculated by processing the set of parameters extracted from the original texture. Combining Eqs. (22) to (27) and (33) we obtain

$$\begin{aligned} \tilde{r}(x,y) = & \left[2 \sum_{p=1}^4 \sum_{q=1}^4 b_{u_{pq}} b_{v_{pq}} k_{pq} \text{Re}(\{n_{pq}(x,y) \right. \\ & * \exp[-\pi(b_{u_{pq}}^2 x^2 + b_{v_{pq}}^2 y^2)]\} \\ & \left. \times \exp[j2\pi f_p(\cos \theta_q x + \sin \theta_q y)]\right) + r(x,y) \right]. \end{aligned} \quad (34)$$

5 Results

A good control over the mathematical parameters of the synthetic texture is necessary to be able to know whether the visual differences of the synthetic texture with respect to the original are due to the limitations of the model or to a lack of accuracy of the synthesis algorithm. In this section, we present and discuss the visual results obtained, after having measured the degree of accuracy of the synthesis algorithm.

5.1 Numerical Results

Ten textures from the Brodatz album²¹ were used to test the synthesis algorithm. Here we report the result of an objective comparison of the sets of parameters obtained from the 10 pairs of original-synthetic textures. This has been done by using the mean squared difference between the original and the synthetic textures' features, averaged over the 10 textures used, and expressed as a SNR in decibels.

Histogram. A high level of accuracy has been achieved (36 dB).

LPR channel modulus. The histogram modification changes the mean value (zero frequency) of the LPR-equalized sum of filtered noise signals and introduces a scaling factor for the rest of the LPR frequencies. This factor ranges, for different textures, from 1 to 2 having a mean value of around 1.5. Compensating for this factor in each pair of original-synthetic texture sets of parameters, the SNR raises to 28 dB.

Energy of the channels. The effect just described appears in this set of parameters and similar scaling factors are obtained for each texture pair. After compensating for this effect, the degree of concordance is 32 dB. Thus, in both this and the former cases, the proportion of the pairs of synthetic-original texture parameters is preserved, but not the absolute values. The cause of these discrepancies was explained in Section 4.7.

Equivalent BWs of the channels. In the current implementation, the equivalent BWs of the channels have had a special treatment, as explained in Section 4.2. In this case, the measured data are the absolute value of the differences between the quantization levels of the original and the synthetic texture's equivalent bandwidths. Using five levels, the mean value obtained for this difference is 0.62. Although relatively high, this error is mainly concentrated at the low energy channels (for which the neglected overlapping of adjacent channels is usually more important), therefore having little visual significance.

As shown next, the visual consequences of the synthesis algorithm imperfections are very small, and much less relevant than those caused by the limitations of the texture model itself.

5.2 Visual Results

We present some significant cases of different kinds of textures to evenly illustrate, from a visual point of view, the main features of this S-A method. Our results have been obtained using six Brodatz's textures: D38 (water), D29 (sand), D68 (wood), D12 (bark), D15 (straw) and D17 (cloth). They have different degrees of structure and order, features that strongly influence on the quality of the results. None of them have been selected, nor even processed (rotated, scaled, etc.), to favor the synthesis performance.

Five images are displayed for each synthesis example (Figs. 9 to 14). In each figure part (a) is the original textured image; part (b) is the synthesis result; part (c) is the synthetic image built using the same spectral phase as part (b) and the power spectrum and histogram of part (a) (the power spectrum has been imposed first and then the histogram); and parts (d) and (e) are, respectively, the power spectra (on a logarithmic gray scale) of the original and the synthetic textured images. Comparison between parts (a), (b) and (c) is useful to evaluate how good the visual coding of the power spectrum and the histogram is, as well as to evaluate the ability of the statistical method used to model the different textures; as shown, a bad synthesis result is normally due to the inadequacy of the statistical measures for describing the visual features of some textures, and it is seldom caused by the losses of the coding of these statistical data. Comparison between parts (d) and (e) reveals how the original spectrum has been coded.

Water texture. The water texture (Fig. 9) has a very low degree of structure, so it can be well modeled as filtered noise. The main difference between the original [Fig. 9(a)] and the synthetic texture [Fig. 9(b)] is that the smooth, slightly tilted waves in the original appear more strongly marked and totally horizontal in the synthetic texture. This is due to the spectral shift that the S-A method causes toward the central 4×4 frequencies of the scheme (a higher number of channels would reduce this effect). This effect can be directly appreciated by comparing Figs. 9(d) and 9(e). On the other hand, from the high visual resemblance existing between Figs. 9(a) and 9(c), we conclude that the spectral phase of this water image is highly random.

Sand texture. This texture (Fig. 10) is more structured, although its basic structural elements (the grains) are small enough to be visually irrelevant when the texture is glo-

Fig. 9 Results of the water texture synthesis: (a) original texture, (b) synthetic texture, (c) texture composed with the Fourier modulus of (a) and the Fourier phase of (b), (d) power spectrum of (a) and (c), and (e) power spectrum of (b).

bally perceived. Obviously, there are no grains in the synthetic texture, but the global impression that it causes is very similar to that of the original one. In contrast with the former case, no significant improvement is achieved now by using the original autocorrelation and histogram, as can be seen by comparing Figs. 10(a) and 10(b) with Fig. 10(c). This fact is very remarkable, since the number of parameters used in Fig. 10(b) is 69, while in Fig. 10(c) it is 33,025 (a compression factor of the statistical data around 500, and of the original image around 1000).

Wood texture. This wood texture (Fig. 11) is very directional and has a moderate degree of structure. Although the structural information is not captured, its power spectrum can be well modeled with our coding scheme, and in this case this feature is very visually relevant. Consequently, the

Fig. 10 Results of the sand texture synthesis: (a) original texture, (b) synthetic texture, (c) texture composed with the Fourier modulus of (a) and the Fourier phase of (b), (d) power spectrum of (a) and (c), and (e) power spectrum of (b).

Fig. 11 Results of the wood texture synthesis: (a) original texture, (b) synthetic texture, (c) texture composed with the Fourier modulus of (a) and the Fourier phase of (b), (d) power spectrum of (a) and (c), and (e) power spectrum of (b).

result is quite good, and again, there are no relevant visual differences between images of Figs. 11(b) and 11(c).

Bark texture. This texture (Fig. 12) has both relevant structural and nonstructural features. The reproduction of the nonstructural features of the original still provides a somewhat similar appearance to the synthetic texture, although now the differences are apparent. As it could be expected, no significant improvement is achieved by imposing the uncoded statistics [compare Figs. 12(a), 12(b), and 12(c)].

Straw texture. This is, by far, the worst case (Fig. 13). The straw is very structured and, besides, it is not very regular. In addition, in the current implementation straight spectral radial lines in oblique directions can not be reproduced. However, the most important reason for this poor

Fig. 12 Results of the bark texture synthesis: (a) original texture, (b) synthetic texture, (c) texture composed with the Fourier modulus of (a) and the Fourier phase of (b), (d) power spectrum of (a) and (c), and (e) power spectrum of (b).

Fig. 13 Results of the straw texture synthesis: (a) original texture, (b) synthetic texture, (c) texture composed with the Fourier modulus of (a) and the Fourier phase of (b), (d) power spectrum of (a) and (c), and (e) power spectrum of (b).

result is that a purely statistical approach is not suitable for characterizing this kind of texture [compare the images in Figs. 13(a) and 13(c)].

Cloth texture. This is a very structured and ordered texture (Fig. 14). Although its structural features (its vertical bands with alternating diagonal directions) can be neither captured nor reproduced, its power spectrum can be. Its high degree of frequency concentration is very significant visually. Thus the synthesis result is quite good. Again, the resemblance existing between the images in Figs. 14(b) and 14(c) is remarkable, which demonstrates the efficiency of the visual coding performed.

From these and other results we conclude that our visual-statistical approach is able to provide a good visual description for those homogeneous textures with little

Fig. 14 Results of the cloth texture synthesis: (a) original texture, (b) synthetic texture, (c) texture composed with the Fourier modulus of (a) and the Fourier phase of (b), (d) power spectrum of (a) and (c), and (e) power spectrum of (b).

structure, or with structural elements that are not very prominent visually (e.g., because their small size). Good results are obtained with textures having very different degrees of order (for instance, the sand and cloth textures).

6 Summary and Discussion

The multiscale scheme used in the proposed S-A method allows for an efficient coding and synthesis of the relevant features of stochastic textures, both in computational and compactness terms. The main difference between this and other, commonly used, methods is that, instead of describing the power spectral density of the texture using the autocorrelation function (or the best fitted AR filter), which are suitable only for some kinds of stochastic textures, it uses a multiscale, visually based spectral sampling scheme. This scheme always provides reasonably good results for stochastic, low structured textures, no matter their degree of spectral entropy, and it achieves it using a small and fixed number of parameters (69).

A certain amount of error exists in the spectral contents location of the synthesis results that is caused by the shift of the frequency contents of the original texture to the central frequencies of the Gabor channels. This inaccuracy, however, is in most cases not important in visual terms. For example, the spectral peaks in Figs. 14(d) and 14(e) (cloth texture) are neither the same number nor exactly placed at the same frequencies, but they cause an almost equal appearance of the images in Figs. 14(c) and 14(b) [the main difference with the original, Fig. 14(a), is due to the loss of the phase information]. In other cases (e.g., the water texture) this effect has a stronger visual impact, but it is always within tolerable bounds (those of the spectral coverage of each Gabor channel).

In return for this lack of accuracy, the proposed method has many important advantages. One of them is the possibility of adequately modeling textures with an either dispersed or concentrated power spectral density, permitting the synthesis of stochastic textures with very different degrees of order. Such a wide range of application would be impossible using autocorrelation-based methods with a small and fixed number of parameters.

Besides completeness and compactness, the robustness of this S-A method is one of its main advantages. First, it is fully automatic: no adaptive calculus of the size and shape of the neighborhood field is required, which is a serious problem in both the AC and the AR models. Second, the Gaussian filtering applied at the synthesis stage is inherently stable, in contrast with the AR synthesis methods. In addition, the resulting texture has a very low visual sensitivity to noise in their parameters. This has been proved by using only five quantization levels for the equivalent bandwidths in the current implementation. A similar coarse quantization could have been applied to the channels energy. In contrast, AR models suffer from a high sensitivity to noise in their parameters.

Finally, both analysis and synthesis stages are based on linear filtering, which can be done very efficiently. Therefore, there is no need for nonlinear iterative algorithms, which are not always reliable nor efficient. The main computational cost corresponds to the filtering, and to the DFTs necessary for calculating the equivalent BWs of the Gabor channels, which are moderately low due to the multiscale

scheme used. Other calculations, as the inversion of the 16×16 overlapping matrix or the histogram adjustment, involve a much lower computational load. The very computationally expensive measurement of the spectral overlapping coefficients, being independent of the particular input texture, is carried out only once at the calibration stage.

Our general conclusion is that the proposed texture S-A method can in many cases be a real alternative to currently accepted methods that do not take advantage of the HVS characteristics. Our model specially fits those common cases where the pursued aim is the visual resemblance between the original and the reconstructed textured image, and for which the generality and robustness of the method are essential. Other approaches that do not have a visual base may be more suitable for those industrial or scientific texture processing tasks that require less generality but more accuracy.

7 Appendix

7.1 Calculation of the Synthesis Filter Bandwidths

The (p, q) Gabor channel of the synthetic texture can be expressed as [analogous to Eq. (9)]:

$$\tilde{T}_{pq}(x, y) = \tilde{T}(x, y) * g_{pq}(x, y). \quad (35)$$

Neglecting the effect of adjacent synthetic channels over this Gabor channel and using a linear approximation of φ in its application interval:

$$\varphi(\lambda) \cong c\lambda + d, \quad (36)$$

the Gabor channel (p, q) of the synthetic texture can be approximated by

$$\tilde{T}_{pq}(x, y) \cong c s_{pq}(x, y) * g_{pq}(x, y). \quad (37)$$

(With usual histograms of textured images, the mean of the synthetic texture and the energy of its Gabor channels correspond very accurately to those that we would get applying an affine (scale and shift) function to the synthetic texture before being adjusted its histogram. As regards the equivalent BWs of the Gabor channels, the histogram matching does not seem to influence them significantly. See numerical results in Section 5.1.)

Demodulating (i.e., moving the channel's spectrum to the zero frequency), we can write

$$\tilde{T}_{0_{pq}}(x, y) \cong c s_{0_{pq}}(x, y) * g_{0_p}(x, y), \quad (38)$$

where $\tilde{T}_{0_{pq}}(x, y)$ is the corresponding demodulated Gabor channel of the synthetic texture and

$$g_{0_p}(x, y) = a_p^2 \exp[-\pi a_p^2(x^2 + y^2)] \quad (39)$$

results from removing the modulating term of Eq. (7). Substituting Eqs. (23), (24), and (39) in Eq. (38), we obtain

$$\begin{aligned} \tilde{T}_{0_{pq}}(x, y) \cong & c k_{pq} n_{pq}(x, y) * \exp\{-\pi[(a_p^2 + b_{u_{pq}}^2)x^2 \\ & + (a_p^2 + b_{v_{pq}}^2)y^2]\}. \end{aligned} \quad (40)$$

Comparing this expression with Eq. (17), and taking into account that the noise has a flat power spectrum, we obtain

$$b_{u_{pq}} = (2S_{u_{pq}}^2 - a_p^2)^{1/2} \quad (41)$$

$$b_{v_{pq}} = (2S_{v_{pq}}^2 - a_p^2)^{1/2} \quad (42)$$

which, substituted in Eq. (24) yield the corresponding synthesis filters. Therefore, for each synthetic channel we use a Gaussian filter calculated by substituting the measured factors $S_{u_{pq}}$ and $S_{v_{pq}}$ [Eqs. (15) and (16)] in the Eq. (41) and (42), and subsequently substituting the resulting factors in Eq. (24).

7.2 Calculation of the Synthetic Channel Amplitudes

Using Eqs. (22), (35) and (36) we can approximate the (p, q) Gabor channel of the synthetic texture by

$$\begin{aligned} \tilde{r}_{pq}(x, y) &\equiv c \left[\sum_{i=1}^4 \sum_{j=1}^4 s_{ij}(x, y) + r(x, y) \right] * g_{pq}(x, y) \\ &\equiv c \sum_{i=1}^4 \sum_{j=1}^4 s_{ij}(x, y) * g_{pq}(x, y), \end{aligned} \quad (43)$$

where the spectral overlapping of the (p, q) Gabor channel with the LPR channel (small even for the lowest frequency channels, as shown in Fig. 5) has been neglected. Due to the statistical independence of the synthetic channels, we can obtain the mean squared value of each Gabor channel by simply adding the mean square values of the summed terms in the preceding equation:

$$\langle |\tilde{r}_{pq}(x, y)|^2 \rangle \equiv c^2 \sum_{i=1}^4 \sum_{j=1}^4 \langle |s_{ij}(x, y) * g_{pq}(x, y)|^2 \rangle. \quad (44)$$

We can write this expression in the frequency domain (through Parseval's theorem) as

$$\langle |\tilde{T}_{pq}(u, v)|^2 \rangle \equiv c^2 \sum_{i=1}^4 \sum_{j=1}^4 \langle |S_{ij}(u, v) G_{pq}(u, v)|^2 \rangle, \quad (45)$$

which can be transformed into

$$\langle |\tilde{T}_{pq}(u, v)|^2 \rangle \equiv c^2 \sum_{i=1}^4 \sum_{j=1}^4 \mu_{pqij} \langle |S_{ij}(u, v)|^2 \rangle, \quad (46)$$

where

$$\mu_{pqij} = \frac{\langle |S_{ij}(u, v)|^2 |G_{pq}(u, v)|^2 \rangle}{\langle |S_{ij}(u, v)|^2 \rangle} \quad (47)$$

is the spectral overlapping coefficient (SOC) associated with the Gabor channel (p, q) and the synthetic channel (i, j) . Each (p, q, i, j) SOC depends on the two BWs of the (i, j) synthetic channel and its spectral location relative to the (p, q) Gabor channel. It can be calculated by integrating the product of the two 2-D Gaussian power spectra involved in each coefficient. These are

$$|S_{ij}(u, v)|^2 \propto \exp \left[-2\pi \left(\frac{u^2}{b_{u_{ij}}^2} + \frac{v^2}{b_{v_{ij}}^2} \right) \right] * [\delta(u - u_{s_{ij}}, v - v_{s_{ij}}) + \delta(u + u_{s_{ij}}, v + v_{s_{ij}})] \quad (48)$$

and

$$|G_{pq}(u, v)|^2 = \exp \left[-\frac{2\pi}{a_l^2} (u^2 + v^2) \right] * [\delta(u - u_{a_{pq}}, v - v_{a_{pq}}) + \delta(u + u_{a_{pq}}, v + v_{a_{pq}})], \quad (49)$$

where $u_{pq} = f_p \cos \theta_q$, $v_{pq} = f_p \sin \theta_q$, and the subindices s and a refer to the synthetic and the Gabor channels, respectively. Neglecting the small discretization effects, we can write Eq. (47) as

$$\mu_{pqij} = \frac{\iint_{u,v} |S_{ij}(u, v)|^2 |G_{pq}(u, v)|^2 du dv}{\iint_{u,v} |S_{ij}(u, v)|^2 du dv} \quad (50)$$

which, using Eqs. (48) and (49), and operating, yields

$$\begin{aligned} \mu_{pqij} &= \frac{a_p^2}{[(a_p^2 + b_{u_{ij}}^2)(a_p^2 + b_{v_{ij}}^2)]^{1/2}} \\ &\times \exp \left\{ -2\pi \left[\frac{(u_{a_{pq}} - u_{s_{ij}})^2}{a_p^2 + b_{u_{ij}}^2} + \frac{(v_{a_{pq}} - v_{s_{ij}})^2}{a_p^2 + b_{v_{ij}}^2} \right] \right\}. \end{aligned} \quad (51)$$

Instead of using this theoretical expression, if the implemented filters are not exactly Gaussian-shaped, as in our implementation, it is more reliable to empirically obtain these coefficients (to calibrate the synthesizer) for some discrete set of synthesis BWs. The calibration consists of measuring the SOC of each possible pair of a Gabor channel and a synthetic channel by synthesizing monochannel textures, applying to them the Gabor filtering scheme, and using Eq. (47).

To simplify the mathematical expressions, it is convenient to use a single number to index both the frequency level and the orientation of the channels (e.g., $n = 4q + p, m = 4j + i$). Doing this, we can define

$$e_n^a = \langle |\tilde{T}_{p(n)q(n)}(u, v)|^2 \rangle, \quad (52)$$

$$e_m^s = \langle |S_{i(m)j(m)}(u, v)|^2 \rangle, \quad (53)$$

and

$$\rho_{nm} = \mu_{p(n)q(n)i(m)j(m)}, \quad (54)$$

which enables us to write Eq. (46) in the simpler form:

$$e_n^a = c^2 \sum_{m=1}^{16} \rho_{nm} e_m^s. \quad (55)$$

This expression can be put in matrix form as

$$\mathbf{e}_s^a := c^2 \mathbf{R} \mathbf{e}_o^s. \quad (56)$$

We must impose $\mathbf{e}_s^a = \mathbf{e}_o^a$ to obtain the same energy values in the Gabor channels of the synthetic texture as those measured in the original. Substituting in Eq. (56) and solving, it finally yields

$$\mathbf{e}_s^s = \frac{1}{c^2} \mathbf{R}^{-1} \mathbf{e}_o^a. \quad (57)$$

Thus, from the set of parameters of the Gabor channels energy measured in the original image ($\mathbf{e}_o^a$) and the calculus or the measurement of the SOC's (arranged in the overlapping matrix $\mathbf{R}$), we obtain the energy that the synthetic channels must have. In practice, the calculation of c is not necessary, because the result of the histogram matching is not affected by factor scaling, and it can be ignored (i.e., taken as unity). Once we know the mean square value of each synthetic channel, the calculation of the k_{pq} factors, which governs the amplitude of the synthetic channels [see Eq. (23)], is straightforward:

$$e_{n(p,q)}^s = k_{pq}^2 \sigma^2 \int \int |\bar{S}_{pq}(u,v)|^2 du dv = \frac{1}{2} k_{pq}^2 \sigma^2 b_{u_{pq}} b_{v_{pq}}, \quad (58)$$

from which it yields

$$k_{pq} = \frac{1}{\sigma} \left(\frac{2e_{n(p,q)}^s}{b_{u_{pq}} b_{v_{pq}}} \right)^{1/2}. \quad (59)$$

(This equation is valid only if the synthesis filters are really Gaussian and frequency aliasing is negligible; if not, the synthetic channel energy can be adjusted by measuring the actual mean square value for an arbitrary value of k_{pq} and multiplying the synthetic channel by the square root of the desired and actual mean square values' ratio.)

Acknowledgments

We are thankful to M. A. Losada, M. Rynders and S. Marcos for useful comments and style suggestions on the final wording and figure preparation of this paper. This work has been supported by the CICYT, Spain, under grant TIC/94-0849.

References

1. R. M. Haralick, "Statistical and structural approaches to texture," *Proc. IEEE* **67**(5), 786–804 (1979).
2. H. Ivernen and T. Lonnestad, "An evaluation of stochastic models for analysis and synthesis of gray-scale texture," *Pattern Recog. Lett.* **15**, 575–585 (1994).
3. B. Julesz, "Visual pattern discrimination," *IRE Trans. Inform. Theory* **8**(1), 84–92 (1962).
4. A. Gagalowicz and S. D. Ma, "Sequential synthesis of natural textures," *Comput. Graph. Image Process.* **30**, 289–315 (1985).
5. S. D. Ma and A. Gagalowicz, "A parallel method for natural texture synthesis," in *Proc. 7th Int. Conf. on Pattern Recognition*, Vol. 1, pp. 90–92, IEEE (1984).
6. R. Navarro and A. Tabernero, "Gaussian wavelet transform: two alternative fast implementations for images," *Multidimen. Syst. Signal Process.* **2**, 421–436 (1991).
7. M. R. Turner, "Texture discrimination by Gabor functions," *Biol. Cybernet.* **55**, 71–82 (1986).
8. A. C. Bovik, M. Clark and W. S. Geisler, "Multichannel texture analysis using localized spatial filters," *IEEE Trans. Pattern Anal. Mach. Intell.* **12**, 55–73 (1990).
9. A. Tabernero and R. Navarro, "Texture Analysis by Gabor cells: a fast pyramid implementation," in *Optics in Medicine, Biology and Environment Research*, G. von Bally and S. Khanna, Eds., Vol. 1, pp. 272–274, Elsevier, Amsterdam (1993).
10. M. Porat and Y. Y. Zeevi, "Localized texture processing in vision: analysis and synthesis in the Gaborian space," *IEEE Trans. on Biomed. Eng.* **36**, 115–129 (1989).
11. R. Navarro, O. Nestares, J. Portilla and A. Tabernero, "Several experiments on texture analysis, coding and synthesis by Gabor wavelets," *Pub. Inst. Opt. Daza de Valdés* **52**, 1–30 (1995).
12. G. Cristobal and R. Navarro, "Space and frequency variant image enhancement based on a Gabor representation," *Pattern Recog. Lett.* **15**, 273–277 (1994).
13. J. Santamaria and M. T. Gómez, "Visible-IR image fusion based on Gabor wavelets composition," *Proc. Eur. Opt. Soc.* **3**, 97–98 (1993).
14. O. Nestares, R. Navarro, J. Portilla and A. Tabernero, "Efficient spatial-domain implementation of a multiscale image representation based on Gabor functions," submitted for publication.
15. J. A. Cadzow, D. M. Wilkes, R. A. Peters, and X. Li, "Image texture synthesis-by-analysis using moving-average models," *IEEE Trans. Aerospace Electron. Syst.* **29**(4), 1110–1122 (1993).
16. A. Rosenfeld, *Multiresolution Image Processing and Analysis*, Springer-Verlag, New York/Berlin (1984).
17. J. M. Francos, A. Z. Meiri and B. Porat, "A unified texture model based on a 2-D Wold-like decomposition," *IEEE Trans. Signal Process.* **41**(8), 2665–2678 (1993).
18. B. MacLennan, "Gabor representation of spatiotemporal visual images," Technical Report CS-91-144, Computer Science Department, University of Tennessee (1991).
19. D. L. Ruderman and W. Bialek, "Statistics of natural images: scaling in the woods," *Phys. Rev. Lett.* **73**, 814–817 (1994).
20. R. C. González and P. Wintz, "Direct histogram specification," in *Digital Image Processing*, 2nd. ed., pp. 152–157, Addison-Wesley, Reading, MA (1987).
21. P. Brodatz, *Textures: A Photographic Album for Artists and Designers*, Dover, New York (1966).

Javier Portilla graduated from Escuela Técnica Superior de Ingenieros de Telecomunicación, Universidad Politécnica de Madrid, Spain, in 1994. He has worked on image processing at the Image and Vision group of the Instituto de Optica, Consejo Superior de Investigaciones Científicas, since 1992, where he is currently working toward his PhD degree under a fellowship. His research interests include multiscale image and sequences processing, texture modeling and synthesis, and applied mathematics.

Rafael Navarro received the MS and PhD degrees in physics from the University of Zaragoza, Spain, in 1979 and 1984, respectively. From 1985 to 1986 he was an optical and image processing engineer at the Instituto de Astrofísica de Canarias. He joined the Instituto de Optica of the Consejo Superior de Investigaciones Científicas in 1987, where he is currently a senior scientific researcher. Since 1988 he has headed the Imaging & Vision group and since 1994 he has been associate director of the Instituto de Optica. He is interested in human vision, optics and image processing.

Oscar Nestares graduated from Escuela Técnica Superior de Ingenieros de Telecomunicación, Universidad Politécnica de Madrid, Spain, in 1994. He has worked on image processing in the Image and Vision group of the Instituto de Optica, Consejo Superior de Investigaciones Científicas, since 1992, where he is working toward his PhD degree under a fellowship. His research interests are human and computer vision, computer models of hu-

man visual system, image sequences processing, and optical flow extraction.

Antonio Tabernero graduated in physics in 1988 from Universidad Complutense de Madrid. He was a research assistant at the Instituto de Optica in Madrid from 1988 to 1992 when he received a PhD degree from the Universidad Complutense de Madrid with a thesis on Gabor functions and their applications in the modeling of the human visual system. During 1993 he was a visiting postdoctoral student at the vision group in the Human Interface Re-

search Branch at the National Aeronautics and Space Administration Ames Research Center. He is currently a member of the faculty at the Computer School of the Universidad Politécnica de Madrid. His research interests are computational vision, image processing, joint representations, and fractal compression.