A Parametric Texture Model based on
Joint Statistics of Complex Wavelet Coefficients

Javier Portilla and **Eero P. Simoncelli**

Center for Neural Science, and
Courant Institute of Mathematical Sciences
New York University
New York, NY 10003

Abstract

We present a universal statistical model for texture images in the context of an over-complete complex wavelet transform. The model is parameterized by a set of statistics computed on pairs of coefficients corresponding to basis functions at adjacent spatial locations, orientations, and scales. We develop an efficient algorithm for synthesizing random images subject to these constraints, by iteratively projecting onto the set of images satisfying each constraint, and we use this to test the perceptual validity of the model. In particular, we demonstrate the necessity of subgroups of the parameter set by showing examples of texture synthesis that fail when those parameters are removed from the set. We also demonstrate the power of our model by successfully synthesizing examples drawn from a diverse collection of artificial and natural textures.

Vision is the process of extracting information from the images that enter the eye. The set of all possible images is vast, and yet only a small fraction of these are likely to be encountered in a natural setting [38, 24, 18, 53]. Nevertheless, it has proven difficult to characterize this set of "natural" images, using either deterministic or statistical models. The class of images that we commonly call "visual texture" seems most amenable to statistical modeling. Loosely speaking, texture images are spatially homogeneous and consist of repeated elements, often subject to some randomization in their location, size, color, orientation, etc. Julesz pioneered the statistical characterization of textures by hypothesizing that the Nth-order joint empirical densities of image pixels (for some unspecified N), could be used to partition textures into classes that are preattentively indistinguishable to a human observer [35]. This work established the description of texture using homogeneous (stationary) random fields, the goal of determining a minimal set of statistical measurements for characterization, and the validation of texture models through human perceptual comparisons. Julesz *et al.* later proposed that pairwise ($N = 2$) statistics were sufficient [36], but then disproved this conjecture by producing example pairs of textures with identical statistics through second (and even third) order that were visually distinct [11, 37].

Since then, two important developments have enabled a new generation of more powerful statistical texture models. The first is the theory of Markov random fields, in which the

full model is characterized by statistical interactions within local neighborhoods. A number of authors have developed Markov texture models, along with tools for characterizing and sampling from such models [e.g. 29, 14, 27, 20]. The second is the use of oriented linear kernels at multiple spatial scales for image analysis and representation. The widespread use of such kernels as descriptions of early visual processing in mammals inspired a large number of models for texture classification and segmentation [e.g., 3, 59, 40]. The development of wavelet representations, which are based on such kernels, have revolutionized signal and image processing. A number of recent inspirational results in texture synthesis are based on multi-scale decompositions [12, 47, 45, 30, 48, 68, 19].

In this paper, we describe a universal parametric statistical model for visual texture, develop a novel method of sampling from this model, and re-examine the Julesz conjecture in the context of our model by comparing the appearance of original texture images with synthesized images that are considered equivalent under our model. We work within a fixed overcomplete multi-scale complex wavelet representation, and our Markov statistical descriptors are based on pairs of wavelet coefficients at adjacent spatial locations, orientations, and scales. In particular, we measure the expected product of the raw coefficient pairs (i.e., correlation), and the expected product of their *magnitudes*. For pairs of coefficients at adjacent scales, we also include the expected product of the fine scale coefficient with the phase-doubled coarse scale coefficient. Finally, we include a small number of marginal statistics of the image pixels and lowpass coefficients at different scales. We develop an efficient algorithm for synthesizing images subject to these constraints, which utilizes iterative projections onto sets. We demonstrate the necessity of each type of parameter by showing synthesis examples that fail when that subset of parameters is removed. Finally, we show a large set of examples of artificial and natural texture synthesis, demonstrating the power and flexibility of the model. Previous instantiations of this model have been described in [55, 54, 49].

1   A Framework for Statistical Texture Modeling

We seek a statistical description of visual texture that is consistent with human visual perception. A natural starting point is to define a texture as a real two-dimensional homogeneous random field (RF) $X(n, m)$ on a finite lattice $(n, m) \in L \subset \mathbf{Z}^2$. The basis for connecting such a statistical definition to perception is the hypothesis first stated by Bela Julesz [35] and reformulated by later authors [e.g. 64, 60, 68]: there exists a set of functions $\{\phi_k(X), k = 1 \ldots N_c\}$ such that samples drawn from any two RFs that are equal in expectation over this set are visually indistinguishable under some fixed comparison conditions. Mathematically,

$$\mathcal{E}\left(\phi_k(X)\right) = \mathcal{E}\left(\phi_k(Y)\right), \forall k \quad \Longrightarrow \quad \text{samples of X and Y are perceptually equivalent,} \quad (1)$$

where $\mathcal{E}(\cdot)$ indicates the expected value over the relevant RF. We refer to this set of functions as the *constraint functions*. The hypothesis establishes the importance of human perception as the ultimate criterion for texture equivalence, and postulates the existence of a universal set of statistical measurements that can capture this equivalence. The hypothesis also implies that this set of statistical measurements provides a parameterization of the space of visual textures. Julesz originally stated his conjecture in terms of Nth-order pixel statistics (assuming homogeneous RFs), but in later work he disproved it for cases $N = 2$ and $N = 3$ by constructing a set of counterexamples [11, 37]. Nevertheless, the general form of the hypothesis provides an appealing foundation for texture modeling.

2

Many variants of Julesz' conjecture may be formulated. For example, a more restricted version of the conjecture states that there exist such sets of functions for each individual texture, or for each subclass of texture. This form of the conjecture has been implicitly assumed in a number of texture models based on texture-specific adaptive decompositions [*e.g.*, 23, 68, 42]. A more ambitious version of the conjecture states that there exists a set of statistical measurements such that two textures are perceptually indistinguishable *if and only if* they are drawn from RFs matching those statistics. In this case, the set of statistical measurements are both sufficient and necessary to guarantee perceptual equivalence. This bidirectional connection between statistics and perception is desirable, as it ensures compactness as well as completeness (i.e., prevents over-parameterization) [12]. The ultimate goal is to achieve such a representation using, in addition, a set of visually meaningful parameters that capture independent textural features. If one were to fully succeed in producing such a set, they would also serve as a model for the early processes of human vision that are responsible for texture perception.

1.1 Testing the Julesz Conjecture

Despite the simplicity with which Julesz' conjecture is expressed, it contains a number of subtleties and ambiguities that make it difficult to test a specific model (i.e., set of constraints) experimentally. First, the definition of perceptual equivalence is a loose one, and depends critically on the conditions under which comparisons are made. Many authors refer to preattentive judgements, in which a human subject must make a rapid decision ("same" or "different") without careful inspection [*e.g.*, 40].

Another subtlety is that the perceptual side of the conjecture is stated in terms of individual images, but the mathematical side is stated in terms of the statistics of abstract probabilistic entities (RFs). In order to reconcile this inconsistency, one must assume that reasonable estimates of the statistics may be computed from single finite images. The relevant theoretical property is *ergodicity*: a spatially ergodic RF is one for which ensemble expectations (such as those in the Julesz conjecture) are equal to spatial averages over the lattice, in the limit as the lattice grows infinitely large. A number of authors have discussed both the necessity of and the complications arising from this property when interpreting the Julesz conjecture [*e.g.*, 26, 64, 60].

But ergodicity is an abstract property, defined only in the limit of an image of infinite size. In order to work with the Julesz conjecture experimentally, one must be able to obtain reasonable estimates of statistical parameters from *finite* images. Thus we define a stronger form of ergodicity:

> **Definition:** A homogeneous RF X has the property of *practical ergodicity* with respect to function $\phi : \mathbb{R}^{|L|} \to \mathbb{R}$, tolerance ϵ, and probability p, if and only if the spatial average of ϕ over a sample image $x(n, m)$ drawn from X is a good approximation to the expectation of ϕ with high probability:

$$\mathcal{P}_X \left(\left| \overline{\phi(x(n,m))} - \mathcal{E}\left(\phi(X)\right) \right| < \epsilon \right) \geq p. \tag{2}$$

Here, $\overline{\phi}$ indicates an average over all spatial translates of the image[1]

$$\overline{\phi(x(n,m))} \equiv \frac{1}{|L|} \sum_{(i,j) \in L} \phi\left(x(\lfloor n+i \rfloor_N, \lfloor m+j \rfloor_M)\right),$$

where (N, M) are the dimensions of the image pixel lattice L, and $\lfloor \cdot \rfloor_N$ indicates that the result is taken modulo N. We refer to these spatial averages as *estimates* or *measurements* of the statistical parameters. Clearly, for a given type of texture, there is a tradeoff between the size of the lattice, the complexity and spatial extent of ϕ, and the values of ϵ and p.

Several methodologies have been used for testing specific examples of the Julesz conjecture. Julesz and others constructed counterexamples by hand [11, 37, 26, 21, 64]. Specifically, they created pairs of textures with the same Nth-order pixel statistics that were strikingly different in appearance (see figure 13). Many authors have used classification tests (sometimes in the context of image segmentation) to evaluate texture models [e.g., 13, 3, 59, 6]. In most cases, this involves defining a distance measure between vectors of statistics computed on each of two texture images, and comparing this with the discriminability of those images by a human observer. This is usually done over some fixed test set of example textures, either artificial or photographic. But classification provides a fairly weak test of the bidirectional Julesz conjecture. Suppose one has a candidate set of constraint functions that is insufficient. That is, there exist pairs of texture images that produce identical statistical estimates of these functions, but look different. It is highly unlikely that such a pair (with exactly matching statistics) would happen to be in the test set, unless the examples in the set were artificially constructed in this way. Alternatively suppose one has a candidate set of parameters that is overconstrained. That is, there exist pairs of texture images that look the same, but have different statistics. Again, it is unlikely that such a pair would happen to be in the test set.

A more efficient testing methodology may be devised if one has an algorithm for sampling from a RF with statistics matching the estimated statistics of an example texture image. The usual approach is referred to as "synthesis-by-analysis" [e.g. 23, 26, 12, 10]. First, one estimates specific values c_k, corresponding to each of the constraint functions ϕ_k, from an example texture image. Then one draws samples from a random field X satisfying the statistical constraints:

$$\mathcal{E}\left(\phi_k(X)\right) = c_k, \qquad \forall k. \tag{3}$$

One then makes visual comparisons of the original example image with images sampled from this random field. Clearly, we cannot hope to fully demonstrate sufficiency of the set of statistics using this synthesis-by-analysis approach, as this would require exhaustive sampling from the RF associated with every possible texture. But if one assumes practical ergodicity of the generating RF for all visually relevant functions (with probability close to one, for visually chosen thresholds), then virtually all generated samples will be perceptually equivalent. Under this assumption, generating a single sample that is visually close to each example in a visually rich library of texture images provides compelling evidence of the power of the model.

A much stronger statement may be made regarding synthesis failures. A single generated sample that is visually different from the original texture indicates that the set of constraint functions is almost certainly insufficient. This is true under the assumption of practical ergodicity of the generating RF for the constraint functions of the model. In this paper, we demonstrate the importance of various subsets of our proposed parameter set by showing

---

[1]We assume periodic boundary handling for simplicity.

that the removal of each subset results in a failure to synthesize at least one of the example textures in our library.

In summary, the synthesis-by-analysis methodology for testing a model of visual texture requires the following ingredients:

- A candidate set of constraint functions, $\{\phi_k\}$. This is the topic of section 2.

- A library of example textures. We have assembled a well-known set of images by Brodatz [8], the VisTex database [61], a set of artificial computer-generated texture patterns, and a set of our own digitized photographs. For purposes of this paper, we utilize only grayscale images.

- A method of estimating statistical parameters. In this paper, we assume practical ergodicity (as defined above), and compute estimates of the parameters by spatially averaging over single images.

- An algorithm for generating samples of a RF satisfying the statistical constraints. Computational efficiency is an important consideration: An inefficient sampling algorithm can impose practical limitations on the choice of the constraint functions, and constitutes an obstacle for visually testing and modifying sets of constraint functions since this process typically requires subjective visual comparison of many synthesized textures. This is discussed in the following subsections.

- A method of measuring the perceptual similarity of two texture images. In this paper, we use only informal visual comparisons.

1.2 Random Fields from Statistical Constraints

Given a set of constraint functions, $\{\phi_k\}$, and their corresponding estimated values for a particular texture, $\{c_k\}$, we need to define and sample from a RF that satisfies equation (3). A mathematically attractive choice is the density with maximum entropy that satisfies the set of constraints [33]. Zhu *et al.* [68] have developed an elegant framework for texture modeling based on maximum entropy and have used it successfully for synthesis. The maximum entropy density is optimal in the sense that it does not introduce any constraints on the RF beyond those of equation (3). The form of the maximum entropy density may be derived by solving the constrained optimization problem using Lagrange multipliers [34]:

$$\mathcal{P}\left(\vec{x}\right) \propto \prod_k e^{-\lambda_k \phi_k(\vec{x})}, \tag{4}$$

where $\vec{x} \in \mathbb{R}^{|L|}$ corresponds to a (vectorized) image, and the λ_k are the Lagrange multipliers. The values of the multipliers must be chosen such that the density satisfies the constraints given in equation (3). But the multipliers are generally a complicated function of the constraint values, c_k, and solving typically requires time-consuming iterative numerical approximation schemes. Furthermore, sampling from this density is non-trivial, and typically requires computationally demanding algorithms such as the Gibbs sampler [27, 68], although recent work by Zhu *et al.* on Monte Carlo Markov Chain methods has reduced these costs significantly [69].

Our texture model is based on an alternative to the maximum entropy formulation that has recently been formalized by Zhu *et al.* [69]. In particular, our synthesis algorithm [55, 54, 49]

operates by sampling from the ensemble of images that yield the same *estimated* constraint values:

$$\mathcal{T}_{\vec{\phi},\vec{c}} = \left\{ \vec{x} : \overline{\phi_k(\vec{x})} = c_k, \forall k \right\}. \tag{5}$$

If we assume $\mathcal{T}_{\vec{\phi},\vec{c}}$ is compact (easily achieved, for example, by including the image variance as one of the constraints), then the maximum entropy distribution over this set is uniform. Zhu et al. [69] have termed this set the *Julesz Ensemble*, and have shown that the uniform distribution over this set is equivalent to the maximal entropy distribution of equation (4) in the limit as the size of the lattice grows to infinity.

1.3 Sampling via Projection

Given a set of constraint functions, ϕ_k, and their corresponding values, c_k, the sampling problem becomes one of selecting an image at random from the associated Julesz ensemble defined in equation (5). Consider a deterministic function that maps an initial image $\vec{x}_0$ onto an element of $\mathcal{T}_{\vec{\phi},\vec{c}}$:

$$p_{\vec{\phi},\vec{c}} : \mathrm{I\!R}^{|L|} \longrightarrow \mathcal{T}_{\vec{\phi},\vec{c}}.$$

If $\vec{x}_0$ is a sample of a RF X_0 defined over the same lattice L, then the function $p_{\vec{\phi},\vec{c}}$ induces a RF on $\mathcal{T}_{\vec{\phi},\vec{c}}$:

$$X_t = p_{\vec{\phi},\vec{c}}(X_0).$$

Assuming X_0 is a homogeneous RF, and that $p_{\vec{\phi},\vec{c}}$ is a translation-invariant function (easy conditions to fulfill), the resulting X_t will also be a homogeneous RF.

By this construction, texture synthesis is reduced to drawing independent samples from X_0, and then applying the deterministic function $p_{\vec{\phi},\vec{c}}$. Note that X_t is guaranteed to be practically ergodic with respect to the set $\{\phi_k\}$ for any $\epsilon > 0$ and any p. Thus, a single sample drawn from X_t that is visually different from the original example serves to demonstrate that the set of constraint functions is insufficient.

In general, the entropy of X_t will have a complicated dependence on both the form of the projection function $p_{\vec{\phi},\vec{c}}$ and the distribution of X_0. Solving for a choice of X_0 that maximizes the entropy of X_t is just as difficult as solving for the maximum entropy density defined in the previous section. Instead, we choose a high-entropy distribution for X_0: Gaussian white noise of the same mean and variance as that of the original image, $\vec{x}_0$. In practice, this choice seems to produce an X_t with fairly high entropy. By deciding not to insist on a uniform (and thus, maximal entropy) density over the Julesz ensemble, we obtain a considerable gain in efficiency of the algorithm and flexibility in the model, as we will show throughout this paper.

1.4 Projection onto Constraint Surfaces

The number and complexity of the constraint functions $\{\phi_k\}$ in a realistic model of a texture make it very difficult to construct a single projection function $p_{\vec{\phi},\vec{c}}$. Thus, we consider an iterative solution, in which the constraints are imposed sequentially, rather than simultaneously. Specifically, we seek a set of functions

$$p_k : \mathrm{I\!R}^{|L|} \longrightarrow \mathcal{T}_k,$$

where $\mathcal{T}_k$ is the set of images satisfying constraint k,

$$\mathcal{T}_k = \left\{ \vec{x} : \overline{\phi_k(\vec{x})} = c_k \right\},$$

such that by iteratively and repeatedly applying these functions, we arrive at an image in $\mathcal{T}_{\vec{\phi},\vec{c}}$. This can be expressed as follows:

$$\vec{x}^{(n)} = p_{\lfloor n \rfloor_{N_c}} \left(\vec{x}^{(n-1)} \right), \tag{6}$$

with $\vec{x}^{(0)} = \vec{x_0}$ an initial image. Assuming this sequence of operations converges, the resulting image will be a member of the Julesz ensemble:

$$p_{\vec{\phi},\vec{c}}(\vec{x_0}) \equiv \lim_{n \to \infty} \vec{x}^{(n)} \in \mathcal{T}_{\vec{\phi},\vec{c}}. \tag{7}$$

Since we do not, in general, know how the projection operations associated with each constraint function will interact, it is preferable to choose each function p_k to modify the image as little as possible. If the functions ϕ_k are smooth, the adjustment that minimizes the Euclidean change in the image is an orthogonal projection of the image onto the manifold of all the images satisfying that particular constraint. This method has been extensively used in the form of projecting onto convex sets (POCS) [66]. In the restricted case of two constraints whose solution sets are both convex and known to intersect, the procedure is guaranteed to converge to a solution. In the case of more than two sets, even if all of them are convex, convergence is not guaranteed [66]. Interesting results have been reported in applying alternating orthogonal projections onto non-convex sets [65]. The texture synthesis algorithm developed by Heeger and Bergen [30] is based on an iterative sequence of histogram-matching operations, each of which is an orthogonal projection onto a non-convex set. In this paper, we will use a relatively large number of constraints, and only few of these are convex. Nevertheless, the algorithm has not failed to get close to convergence for the many hundreds of examples on which it has been tested.

1.5 Gradient Projection

By splitting our projection operation into a sequence of smaller ones, we have greatly simplified our mathematical problem. However, in many practical cases, even solving a single orthogonal projection may be difficult. Since our method already involves two nested loops (going repeatedly through the set of constraints), we need to find efficient, single-step adjustments for each constraint or group of constraints. A simple possibility corresponds to moving in the direction of the gradient of $\phi_k(\vec{x})$:

$$\vec{x}\,' = \vec{x} + \lambda_k \vec{\nabla} \phi_k(\vec{x}), \tag{8}$$

where λ_k is chosen such that

$$\phi_k(\vec{x}\,') = c_k. \tag{9}$$

The rationale of the gradient projection is the same as for the orthogonal case: we want to adjust $\phi_k(\vec{x})$ while changing $\vec{x}$ as little as possible. It is interesting to note that the gradient vector $\vec{\nabla}\phi_k(\vec{x})$, when evaluated at images $\vec{x}$ that lie on the surface corresponding to the constraint, $\mathcal{T}_k$, is orthogonal to that surface. This means that in the limit as $\phi_k(\vec{x}) \to c_k$,

orthogonal and gradient projections are equivalent. It also implies that if we orthogonally project our gradient-projected vector $\vec{x}\,'$ back onto the original constraint set, we obtain the original vector $\vec{x}$. In this sense, the gradient projection and the orthogonal projection are inverse of each other.

The calculation of $\vec{\nabla}\phi_k(\vec{x})$ is usually straightforward, and all that remains in implementing the projection is to solve for the constant λ_k that satisfies equation (9). In general, however, there may not be a single solution. When there are multiple solutions for λ_k, we simply choose the one with the smallest magnitude (this corresponds to making the minimal change in the image). When there is no solution, we solve for the λ_k that comes closest to satisfying the constraint of equation (9).

We can extend the previous idea from the adjustment of a single constraint to the adjustment of a subset of related constraints $\left\{\vec{\phi_{\mathcal{S}}}, \vec{c_{\mathcal{S}}}\right\}$, where $\mathcal{S} \subset \{k = 1 \ldots N_c\}$. In this case, we seek values of $\lambda_k, \forall k \in \mathcal{S}$ such that the projected vector

$$\vec{x}\,' = \vec{x} + \sum_{k \in \mathcal{S}} \lambda_k \vec{\nabla}\phi_k(\vec{x}) \tag{10}$$

satisfies the constraints:

$$\vec{\phi}_{\mathcal{S}}(\vec{x}\,') = \vec{c}_{\mathcal{S}}. \tag{11}$$

2 Texture Model

How should one go about choosing a set of constraint functions? Since texture comparisons are to be done by human observers, one source of inspiration is our knowledge of the earliest stages of human visual processing. If the set of constraint functions can be chosen to emulate the transformations of early vision, then textures that are equivalent according to the constraint functions will be equivalent at this stage of human visual processing. As is common in signal processing, we proceed by first decomposing the signal using a linear basis. The constraint functions are then defined on the coefficients of this basis.

2.1 Local Linear Basis

There is a long history of modeling the response properties of neurons in primary visual cortex, as well as the performance of human observers in psychophysical tasks, using localized oriented bandpass linear filters [e.g., 28]. These decompositions are also widely used for tasks in computer vision. In addition, recent studies of properties of natural images indicate that such decompositions can make accessible higher-order statistical regularities [e.g., 24, 63, 16, 67, 55]. Especially relevant are recent results on choosing bases to optimize information-theoretic criterion, which suggest that a basis of localized oriented operators at multiple scales is optimal for image representation [43, 2]. Many authors have used sets of multi-scale bandpass filters for texture synthesis (see introduction).

We wish to choose a fixed multi-scale linear decomposition whose basis functions are spatially localized, oriented, and roughly one octave in bandwidth. In addition, our sequential projection algorithm requires that we be able to invert this linear transformation. In this regard, a bank of Gabor filters at suitable orientations and scales [e.g., 17, 47, 48] would be inconvenient. An orthonormal wavelet representation [e.g., 15, 41] suffers from a lack of
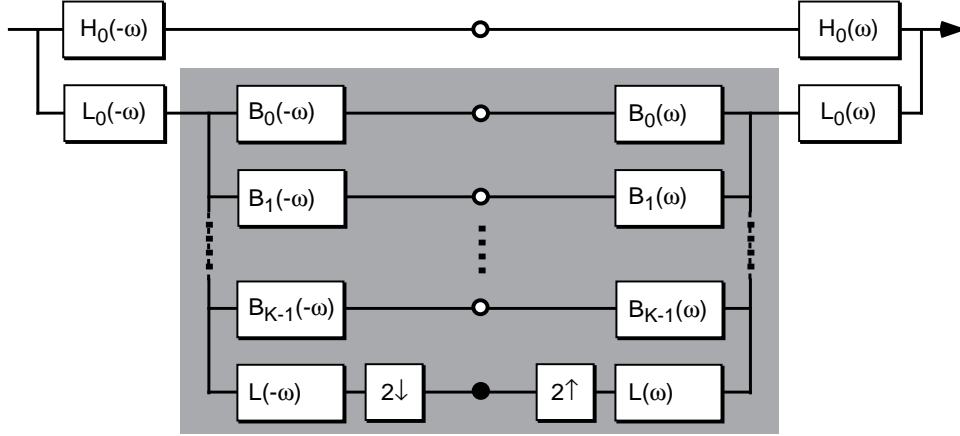
Figure 1. System diagram for the steerable pyramid [56]. The input image is initially split into high- and lowpass bands. The lowpass band is then further split into a lower-frequency band and a set of oriented subbands. The recursive construction of a pyramid is achieved by inserting a copy of the diagram contents indicated by the shaded region at the location of the solid circle (i.e., the lowpass branch).

translation-invariance, which is likely to cause artifacts in an application like texture synthesis [57]. Thus, we chose to use a "steerable pyramid" [57, 56], since this transform has nice reconstruction properties (specifically, it is a tight frame), in addition to properties of translation-invariance and rotation-invariance. The basis functions may be interpreted as directional derivatives of some lowpass kernel.

For this paper, we have extended the original steerable pyramid representation to include complex "analytic" filters. That is, the real and imaginary parts correspond to a pair of even- and odd-symmetric filters (analogous to a Hilbert Transform pair in one dimension). This allows us to utilize measures of local phase and energy in some of our texture descriptors. Such measures have proved important throughout computer vision [e.g., 39, 44], and are again motivated by what is known of biological vision.

Similar to conventional orthogonal wavelet decompositions, the steerable pyramid is implemented by recursively splitting an image into a set of oriented subbands and a lowpass residual band. The system diagram for the transform is shown in figure 1. The filters used in this transformation are polar-separable in the Fourier domain, where they may be written as:

$$
L(r, \theta) = \begin{cases} 2\cos\left(\frac{\pi}{2} \log_2\left(\frac{4r}{\pi}\right)\right), & \frac{\pi}{4} < r < \frac{\pi}{2} \\ 2, & r \leq \frac{\pi}{4} \\ 0, & r \geq \frac{\pi}{2} \end{cases}
$$

$$
B_k(r, \theta) = H(r)G_k(\theta), \quad k \in [0, K-1],
$$

with radial and angular parts

$$
H(r) = \begin{cases} \cos\left(\frac{\pi}{2} \log_2\left(\frac{2r}{\pi}\right)\right), & \frac{\pi}{4} < r < \frac{\pi}{2} \\ 1, & r \geq \frac{\pi}{2} \\ 0, & r \leq \frac{\pi}{4} \end{cases}
$$

$$
G_k(\theta) = \begin{cases} \alpha_K \left[\cos(\theta - \frac{\pi k}{K})\right]^{K-1}, & |\theta - \frac{\pi k}{K}| < \frac{\pi}{2} \\ 0, & \text{otherwise,} \end{cases}
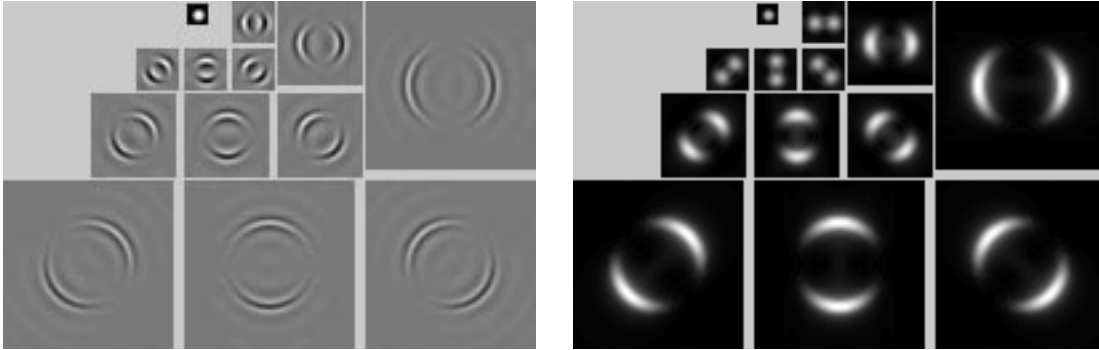$$

9

Figure 2. A 3-scale, 4-orientation complex steerable pyramid representation of a disk image. Left: real parts of oriented bandpass images at each scale and the final lowpass image. Right: magnitude (complex modulus) of the same subbands. Note that the highpass residual band is not shown.

where r, θ are polar frequency coordinates, and

$$\alpha_K = 2^{K-1} \frac{(K-1)!}{\sqrt{K[2(K-1)]!}}.$$

The lowpass band is subsampled by a factor of two along both axes (and thus requires an amplitude of two). Unlike conventional orthogonal wavelet decompositions, the subsampling does not produce aliasing artifacts, as the support of the lowpass filter $L(r, \theta)$ obeys the Nyquist sampling criterion. The recursive procedure is initialized by splitting the input image into lowpass and highpass portions, using the following filters:

$$L_0(r, \theta) = L(\frac{r}{2}, \theta)/2$$
$$H_0(r, \theta) = H(\frac{r}{2}, \theta).$$

For all examples in this paper, we have used $K = 4$ orientation bands, and $N = 4$ pyramid levels (scales), for a total of 18 subbands (16 oriented, plus highpass and lowpass residuals). Figure 2 shows an example image decomposition. The basis set forms a tight frame, and thus the transformation may be inverted by convolving each complex subband with its associated complex-conjugated filter and adding the results. Alternatively, one may reconstruct from the non-oriented residual bands and either the real or imaginary portions of the oriented bands.

2.2 Statistical Constraints

Assuming one has decomposed an image using a set of linear filters, the constraint functions may be defined on the coefficients of this decomposition. Previous authors have used autocorrelation [e.g., 13, 6, 52, 5, 48], nonlinear scalar functions [1], and marginal histograms [23, 30, 68] of subbands. But recent nonparametric models of joint coefficient densities have produced the most visually impressive synthesis results. Popat and Picard [46] used a mixture model to capture regions of high probability in the joint density of small sets of coefficients at adjacent spatial positions and scales. Debonet and Viola [19] showed impressive

results by directly resampling from the scale-to-scale joint histograms on small neighborhoods of coefficients. These nonparametric texture models, along with our own studies of joint statistical properties in the context of other applications [55, 9], motivated us to consider the use of joint statistical constraints.

In general, we have used the following heuristic strategy for incrementally augmenting an initial set of constraint functions:

1. Initially choose a set of basic parameters and synthesize texture samples using a large library of examples.

2. Gather examples of synthesis failures, and classify them according to the visual features that distinguish them from their associated original texture examples. Choose the group of failures that produced the poorest results.

3. Choose a new statistical constraint that captures the visual feature most noticeably missing from the failure group. Incorporate this constraint into the synthesis algorithm.

4. Verify that the new constraint achieves the desired effect of capturing that feature by re-synthesizing the textures in the failure group.

5. Verify that the original constraints are still necessary. For each of the original constraints, find a texture example for which synthesis fails when that constraint is removed from the set.

This strategy resembles the greedy entropy-minimization approach proposed by Zhu *et al.* [68], but differs in that: 1) the constraint set is not adapted for a single texture, but to a reference set of textures; and 2) the procedure is driven by perceptual criteria rather than information-theoretic criteria. Our constraint set was developed using this strategy, beginning with a set containing only correlation measures. Below, we describe each of the resulting constraints, providing some motivation for its inclusion, and demonstrating its necessity by showing two examples of synthesis failures when that constraint is removed from the set.

Marginal Statistics. The statistics of gray-level texture pixels express the relative amount of each intensity in the texture. Many texture models make direct use of the pixel histogram to characterize this distribution (e.g. [26, 30, 68, 48]). A more complete description of the gray-level distribution, which includes some basic spatial information, consists of characterizing the marginal statistics of the blurred versions of the image at different scales. In our current model we have found it sufficient to measure three normalized sample moments (variance, skewness, and kurtosis) together with the range (minimum and maximum intensities) for the pixel statistics, and the skewness and kurtosis of the lowpass images computed at each level of the recusive pyramid decomposition illustrated in figure 1 (their variance is included in the autocorrelation, described next). Figure 3 demonstrates the necessity of these pixel-domain statistics for proper synthesis of two different example textures.

Coefficient Correlation. The coefficients of our wavelet decomposition are typically correlated for two reasons. First, the representation is highly overcomplete, so the coefficients lie within a linear subspace. More importantly, covariances of subband coefficients can arise from spectral peaks (i.e., periodicity) or ridges (i.e., globally oriented structure) in a texture [25].
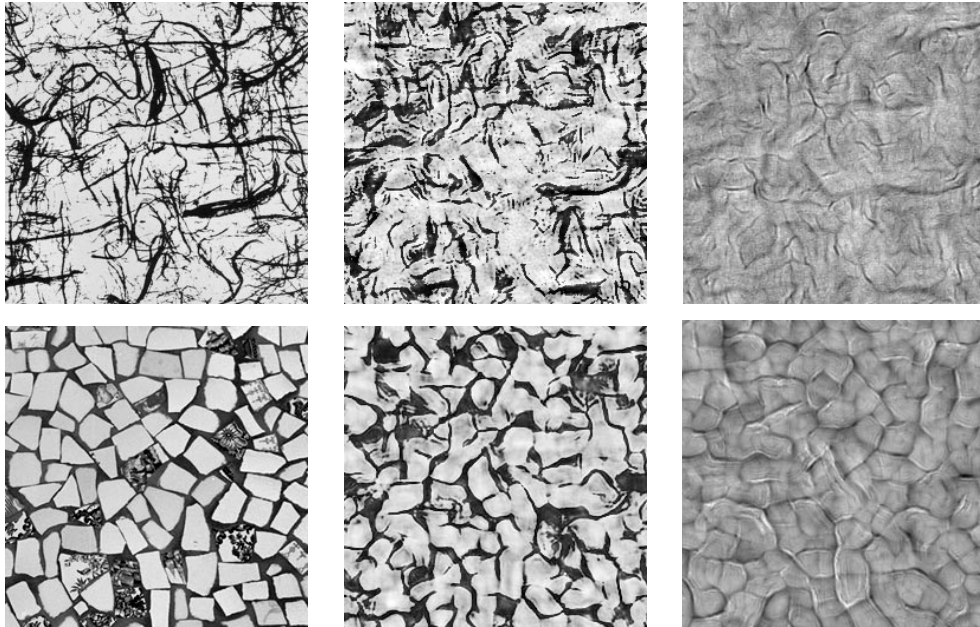
Figure 3. Necessity of marginal constraints. Left column: original texture images. Middle: Images synthesized using full constraint set. Right: Images synthesized using all but the marginal constraints.

In order to represent such spectral features, we should include the local autocorrelation of each subband as a texture descriptor. However, due to the overcompleteness of our linear representation, the spatial correlation of the subband responses are highly redundant, and thus unsuitable for a compact model. A more efficient set of parameters which describe the same type of features is obtained using the local autocorrelation of the lowpass images computed at each level of the recursive pyramid decomposition. This set of parameters provide high spectral resolution in the low frequencies and low spectral resolution in high frequencies, which is a natural solution for a scale-invariant modeling of images (see [58]). It has been known for some time that correlation measurements alone are not sufficient to capture the structure of many natural textures [37, 51, 23]. But in the context of our model, they are still necessary to represent periodic structures and long-range correlations. This is illustrated in figure 4.

Magnitude Correlation. Each of the parameters described thus far have been used by previous authors for texture synthesis. But these parameters are not sufficient (at least in the context of our choice of basis) to represent many textures. In recent work, we have studied the joint statistics of wavelet coefficient amplitudes for natural images [55, 9, 62], and have found that these are quite regular. In particular, we have examined the conditional histograms of pairs of coefficient amplitudes at adjacent spatial locations, orientations, and scales. We find that the variance of the conditional density scales with the square of the conditioning coefficient, even when the raw coefficients are uncorrelated. There is an intuitive explanation for this: the "features" of real images give rise to large coefficients in local spatial
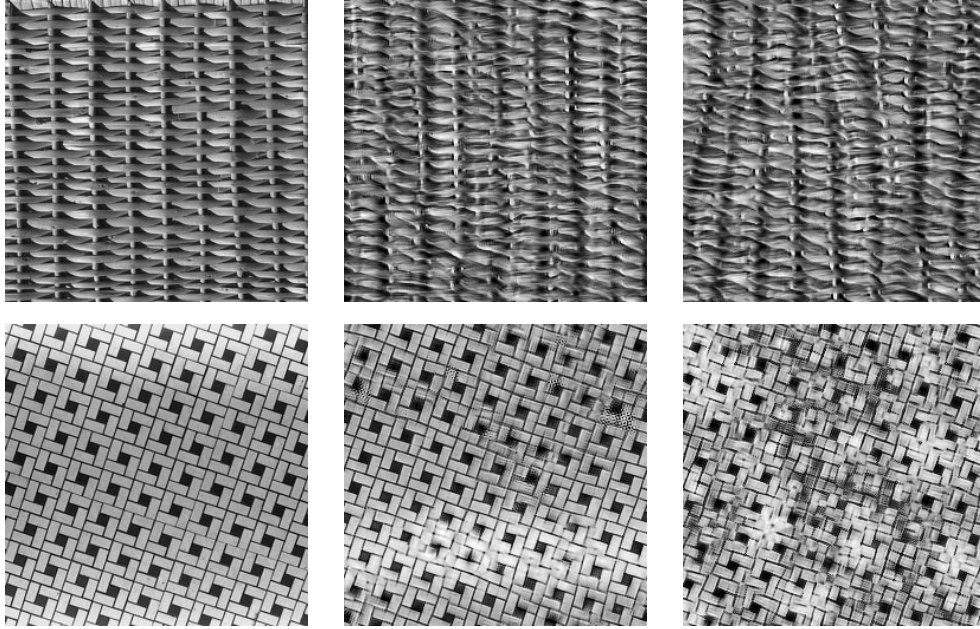
Figure 4. Necessity of raw autocorrelation constraints. Left column: original texture images. Middle: Images synthesized using full constraint set. Right: Images synthesized using all but the autocorrelation constraints.

neighborhoods, as well as at adjacent scales and orientations. The use of joint statistics of rectified subband coefficients also appears in the human vision literature in the form of "second-order" texture analyzers and models [see 28, 4]. The idea is to decompose the image with a linear basis, rectify or square the coefficients, and then apply a second linear transform. These measures may be used to represent or compare textures, or to segment an image into homogeneous simpler ("first-order") texture regions.

Figure 5 shows the steerable pyramid coefficient magnitudes of two texture images. One can see that the magnitudes (and the relationship between them) capture important structural information about each of these textures. In particular, large magnitudes appear at the same locations in the "squares" texture, and large magnitudes of the two diagonal orientations are anti-correlated in the herringbone texture. Such correlations are often present despite the fact that the raw coefficients may be uncorrelated. This can occur because variations in phase across the image lead to cancellation. Furthermore, the magnitude correlation is not purely due to the linear basis we have chosen: Different images reveal different degrees of correlation. As a simple summary of this dependence, we compute the correlation of the complex magnitude of pairs of coefficients at adjacent positions, orientations and scales. Note that computing the cross-scale statistics requires that the coarse subband be upsampled and interpolated to match the dimensions of the fine subband. Figure 6 demonstrates the necessity of the magnitude correlation constraints: high contrast locations in the image are no longer organized along lines or edges.
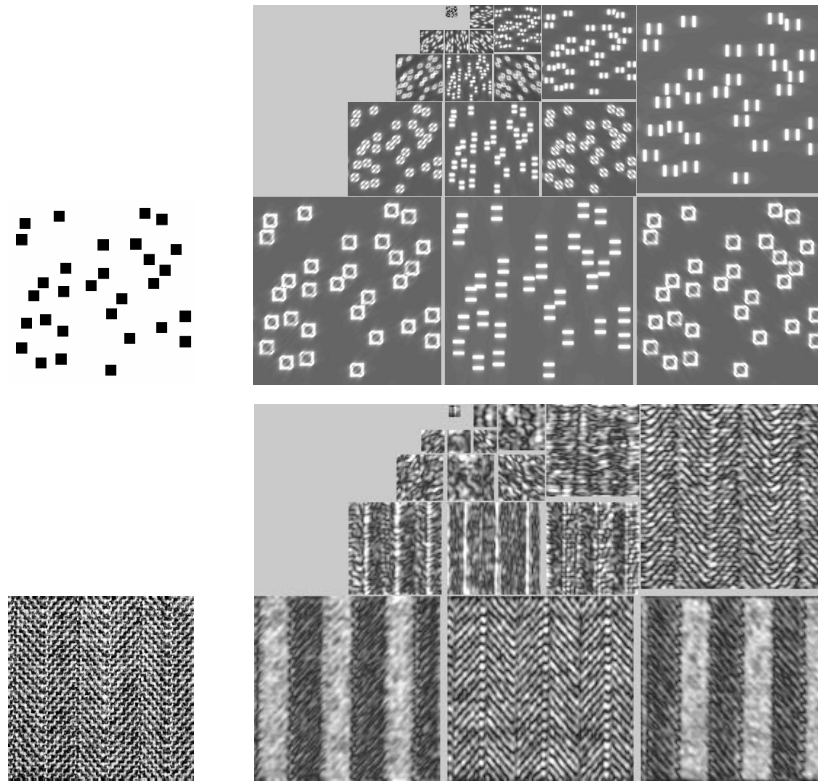
Figure 5. Normalized magnitude responses of the steerable pyramid subbands for two example textures images (shown at left).
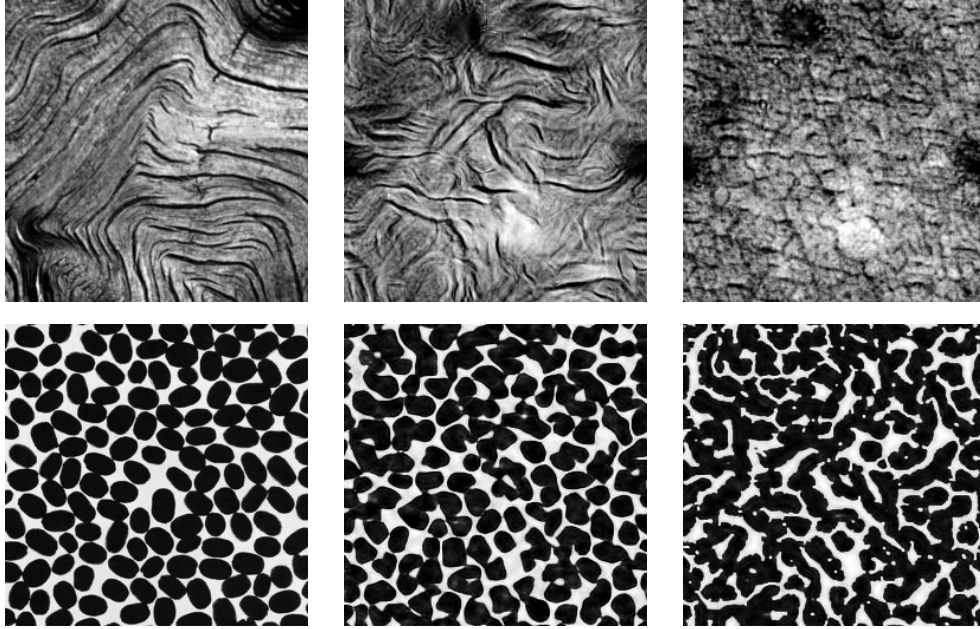
Figure 6. Necessity of magnitude correlation constraints. Left column: original texture images. Middle: Images synthesized using full constraint set. Right: Images synthesized using all but the magnitude auto- and cross- correlation constraints.

Cross-scale Phase Statistics In earlier instantiations of our model [54], we found that many examples of synthesis failure were due to inability to represent the phase of the responses to local features, such as edges and lines. For example, a white line on a dark background will give rise to a zero phase response in the coefficients of all scales along that line, whereas a dark line on a light background will produce a π response. Similarly, an edge will produce either $+\frac{\pi}{2}$ or $-\frac{\pi}{2}$ phase responses along the edge, depending on the polarity of the edge. As an intermediate case, a feature with a sawtooth profile will create phase responses that shift with the spatial frequency of the subband. Such gradients of intensity occur, for example, when convex objects are illuminated diffusely, and are responsible for much of the three dimensional appearance of these textures.

In order to capture the local phase behavior, we have developed a novel statistical measurement that captures the relative phase of coefficients of bands at adjacent scales. In general, the local phase varies linearly with distance from features, but the rate at which it changes for fine-scale coefficients is twice that of those at the coarser scale. To compensate for this, we double the complex phase of the coarse-scale coefficients, and compute the cross-correlation of these modified coefficients with the fine-scale coefficients. Mathematically,

$$\phi(f, c) = \frac{c^2 \cdot f^*}{|c|},$$

where f is a fine-scale coefficient, c is a coarse-scale coefficient at the same location. Defining $\hat{c} = c^2/|c|$, we can express the expectation of that product as:

$$\mathcal{E}\left(\hat{c}f^*\right) \quad = \quad \mathcal{E}\left(f_r\hat{c}_r + f_i\hat{c}_i\right) + i\mathcal{E}\left(f_r\hat{c}_i - f_i\hat{c}_r\right)$$

15

input

real/imag   mag/phase   real/imag   mag/phase
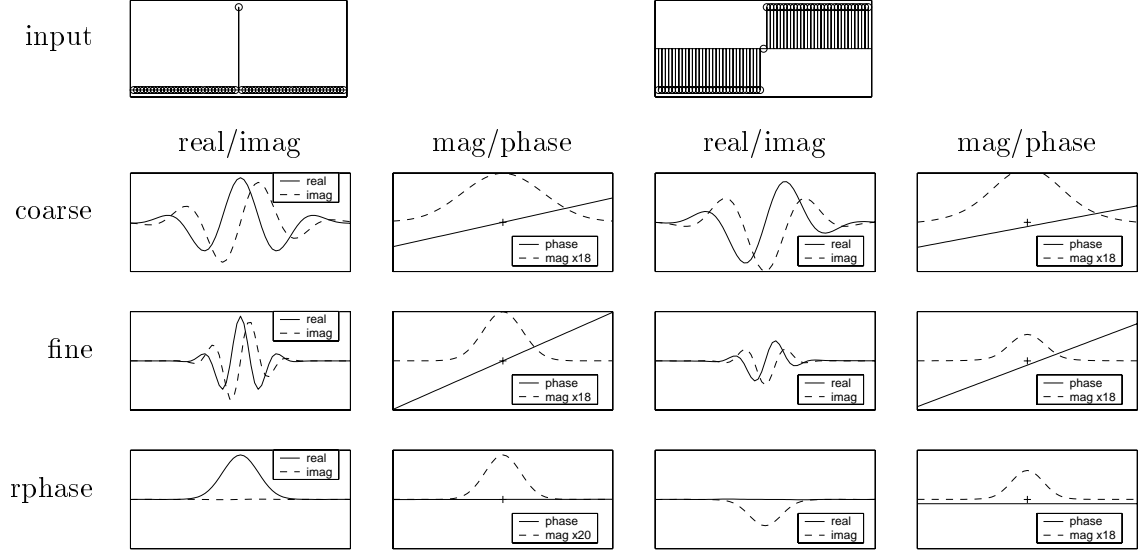
coarse

fine

rphase

Figure 7. Illustration of the "relative phase" statistic for two one-dimensional signals, shown in the top row. Left two columns: Responses to an impulse signal. Right two columns: Responses to a step edge signal. The left plots in each pair of columns show real/imaginary parts, and the right plots show the corresponding magnitude/phase.

$$\approx \quad 2\mathcal{E}\left(f_r\hat{c}_r\right) + 2i\mathcal{E}\left(f_r\hat{c}_i\right),$$

where the approximation holds because $(\hat{c}_r, \hat{c}_i)$ is a quadrature-phase pair (approximately), as is (f_r, f_i). Thus, relative phase is captured by the two real expectations $\mathcal{E}\left(f_r\hat{c}_r\right)$ and $\mathcal{E}\left(f_r\hat{c}_i\right)$. This statistic is illustrated in figure 7 for two one-dimensional signals: An impulse function, and a step edge. Note the difference in the real/imaginary parts of the relative phase statistic for the two signals. Figure 8 demonstrates the importance of this phase statistic in representing textures with strong illumination effects. In particular, when it is removed, the synthesized images appear much less three dimensional and lose the detailed structure of shadows.

2.3   Summary of Statistical Constraints

As a summary of the statistical model proposed, we enumerate the set of statistical descriptors, review the features they capture, and give their associated number of parameters.

- Marginal statistics: skewness and kurtosis of the partially reconstructed lowpass images at each scale ($2(N+1)$ parameters), variance of the highpass band (1 parameter), and mean, variance, skew, kurtosis, minimum and maximum values of the image pixels (6 parameters),

- Raw coefficient correlation: Central samples of the autocorrelation of the partially reconstructed lowpass images, including the lowpass band ($(N+1) \cdot \frac{M^2+1}{2}$ parameters) These characterize the salient spatial frequencies and the regularity (linear predictability) of the texture, as represented by periodic or globally oriented structures.

- Coefficient magnitude statistics: Central samples of the autocorrelation of magnitude of each subband ($N \cdot K \cdot \frac{M^2+1}{2}$ parameters), cross-correlation of each subband magnitudes
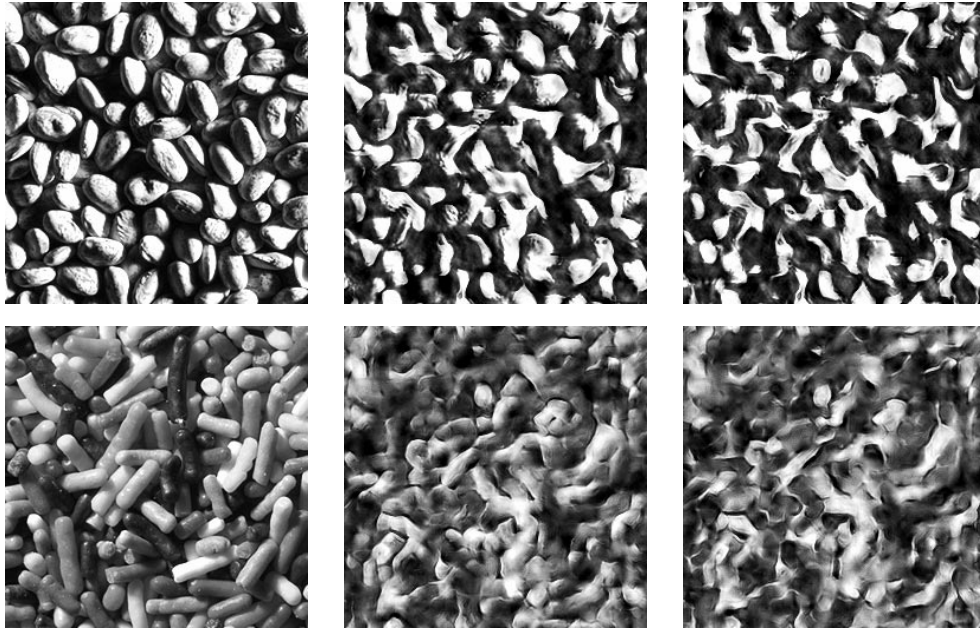
16

Figure 8. Necessity of cross-scale phase constraints. Left column: original texture images. Middle: Images synthesized using full constraint set. Right: Images synthesized using all but the cross-scale phase constraints.

with those of other orientations at the same scale ($N \cdot \frac{K(K-1)}{2}$ parameters), and cross-correlation of subband magnitudes with all orientations at a coarser scale ($K^2(N-1)$ parameters). These represent structures in images (e.g., edges, bars, corners), and "second order" textures.

- Cross-scale phase statistics: cross-correlation of the real part of coefficients with both the real and imaginary part of the phase-doubled coefficients at all orientations at the next coarser scale ($2K^2(N-1)$ parameters). These distinguish edges from lines, and help in representing gradients due to shading and lighting effects.

For our texture examples, we have made choices of $N{=}4$, $K{=}4$ and $M{=}7$, resulting in a total of 710 parameters. We emphasize that this is a *universal* (non-adapted) parameter set. High quality synthesis results may be achieved for many of the individual textures in our set using a much-reduced subset of these parameters. But, as we have shown in figures 3–8, removal of any one group of parameters leads to synthesis failures for some textures.

3   Implementation and Results

3.1   Sequential Projection, Convergence, and Efficiency

Figure 9 shows a block diagram of our synthesis-by-analysis algorithm. The process is initialized with an image containing samples of Gaussian white noise. The image is decomposed into a complex steerable pyramid. A recursive coarse-to-fine procedure imposes the statistical constraints on the lowpass and bandpass subbands, while simultaneously reconstructing
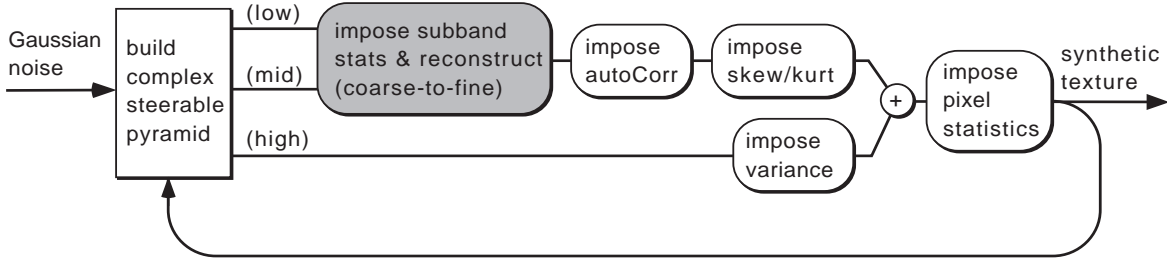
Figure 9: Top level block diagram of recursive texture synthesis algorithm.

a lowpass image. A detailed diagram of this coarse-to-fine procedure is shown in figure 10. The autocorrelation of the reconstructed lowpass image is then adjusted, along with the skew and kurtosis, and the result is added to the variance-adjusted highpass band to obtain the synthesized texture image. The marginal statistics are imposed on the pixels of this image, and the entire process is repeated. For accelerating the convergence, at the end of the loop, we amplify the change in the image from one iteration to the next by a factor of 1.8. Details of each of the adjustment operations are given in the Appendix.

The algorithm we have described is fairly simple, but we cannot guarantee convergence. The projection operations are not exactly orthogonal, and the constraint surfaces are not all convex. Nevertheless, we find that convergence is achieved after about 50 iterations, for the hundreds of textures we have synthesized. In addition, once convergence has been achieved (to within some tolerance), the synthetic textures are quite stable, oscillating only slightly in their parameters. Figure 11 shows the evolution of a synthetic texture, illustrating the rapid visual convergence of the algorithm. Our current implementation (in MatLab) requires roughly 20 minutes to synthesize a 256×256 texture (in 50 iterations) on a 500 Mhz Pentium workstation.

3.2 Synthesis Results

A visual texture model must be tested visually. In this section, we show that our model is capable of representing an impressive variety of textures.[2] Figure 12 shows a set of synthesis results on artificial texture images. The algorithm handles periodic examples quite well, producing output with the same periodicities and structures. Note that the absolute phase of the synthesized images is random, due to the translation-invariant nature of the algorithm, and the fact that we are treating boundaries periodically in all computations. Also shown are results on a few images composed of repeated patterns placed at random non-overlapping positions within the image. Surprisingly, our statistical model is capable of representing these types of texture, and the algorithm does a reasonable job of re-creating these images, although there are significantly more artifacts than in the periodic examples.

Figure 13 shows two pairs of counterexamples that have been used to refute the Julesz conjecture. The leftmost pair were originally created by Julesz [37]: they have identical third-order pixel statistics, but are easily discriminated by human observers. Our model succeeds, in that it can reproduce the visual appearance of either of these textures. In particular, we have seen that the strongest statistical difference arises in the magnitude correlation statistics.

---

[2]Additional examples and source code are available on the internet at http://www.cns.nyu.edu/~lcv/texture/.
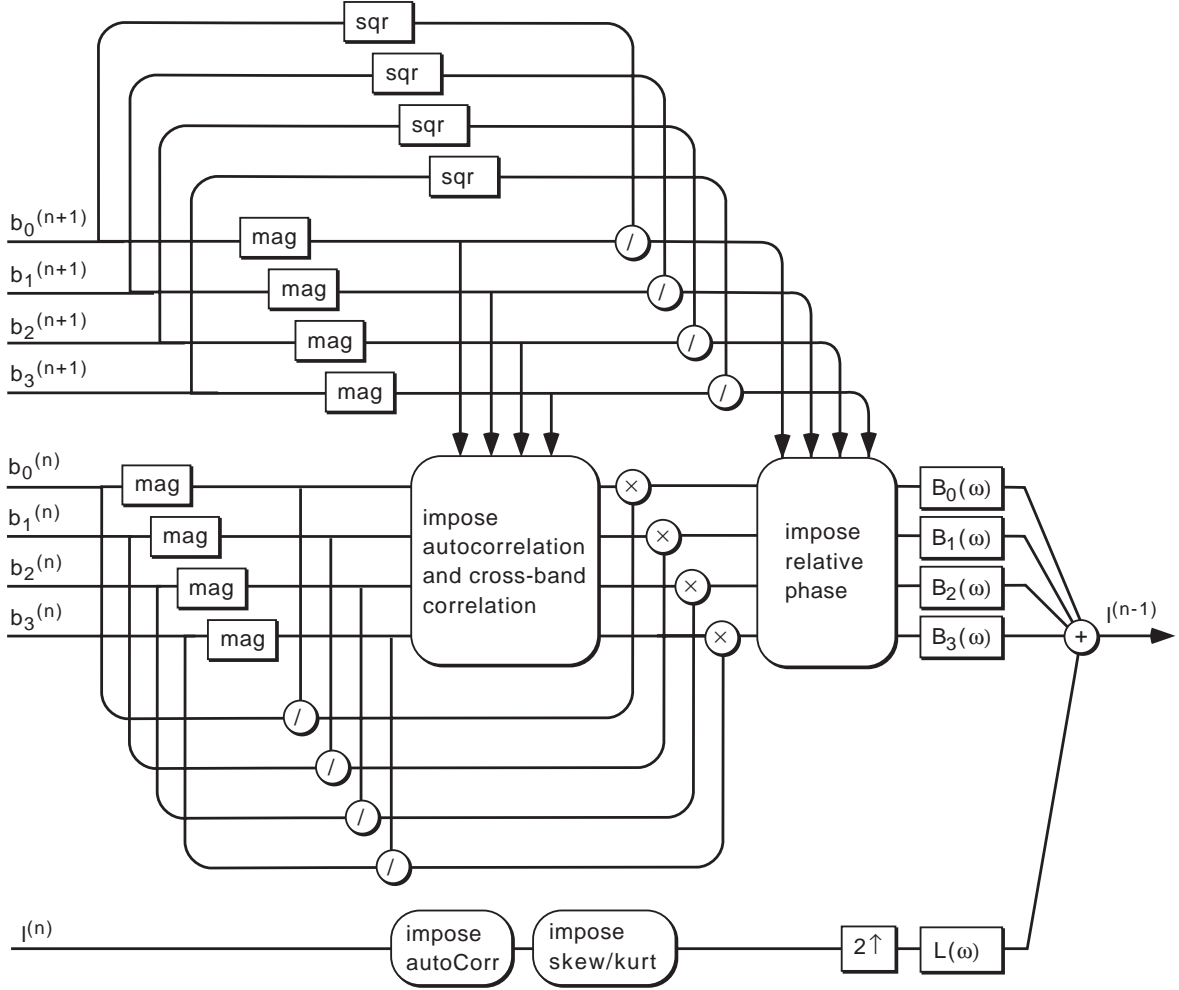
Figure 10. Block diagram describing the coarse-to-fine adjustment of subband statistics and reconstruction of intermediate scale lowpass image (gray box of figure 9).
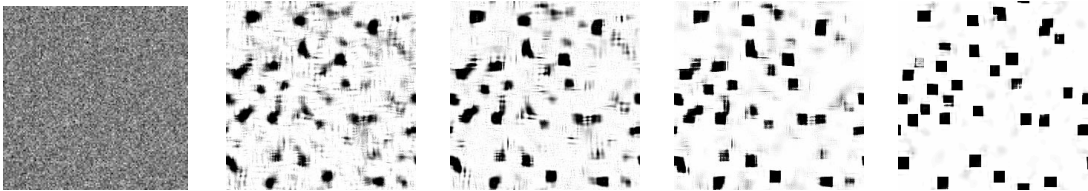

Figure 11. Example texture synthesis progression, for 0, 1, 2, 4 and 64 iterations. Original image shown in figure 5.
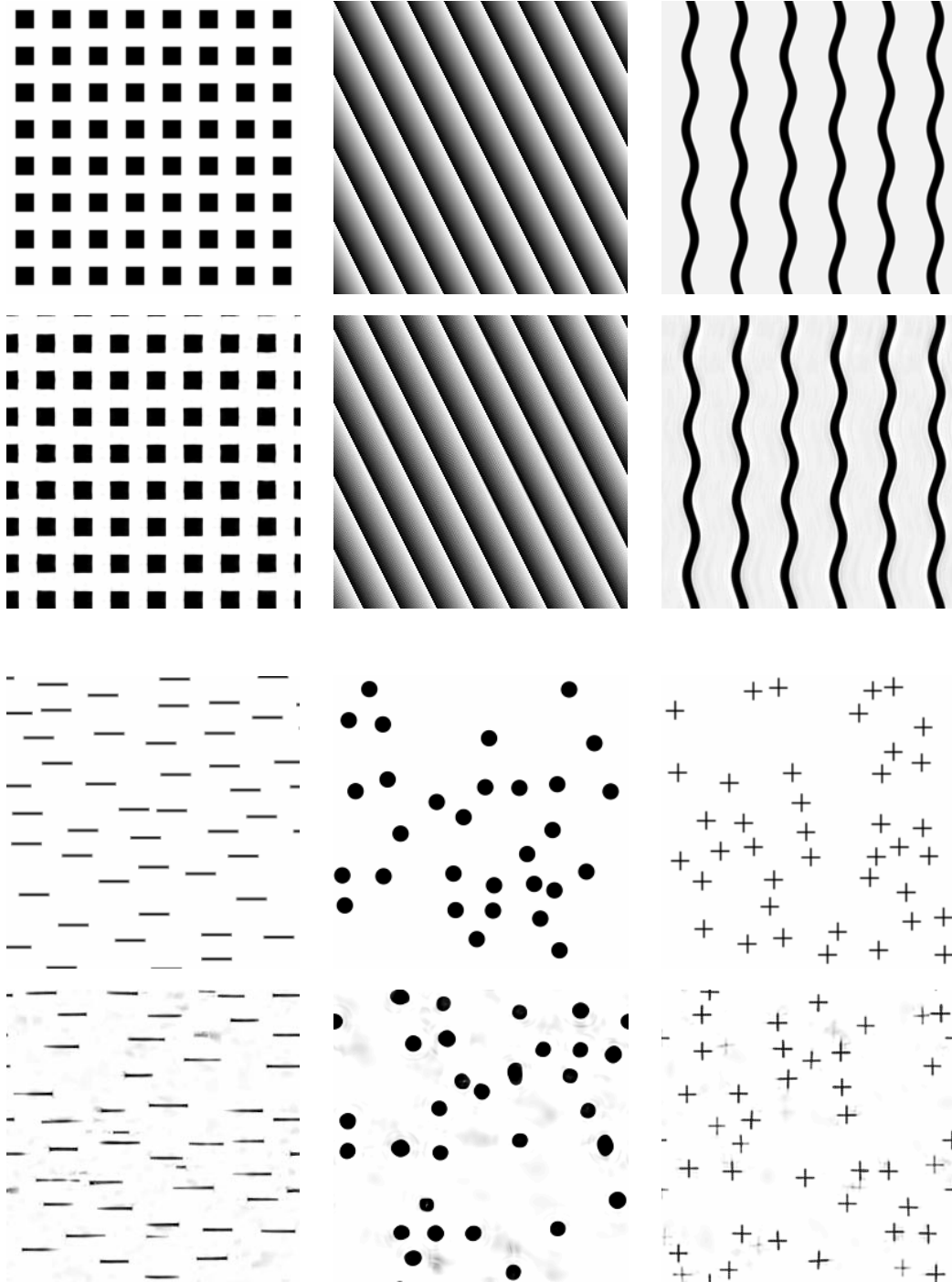
Figure 12. Synthesis results on artificial textures. For each pair of textures, the upper image is the original texture, and the lower image is the synthesized texture.
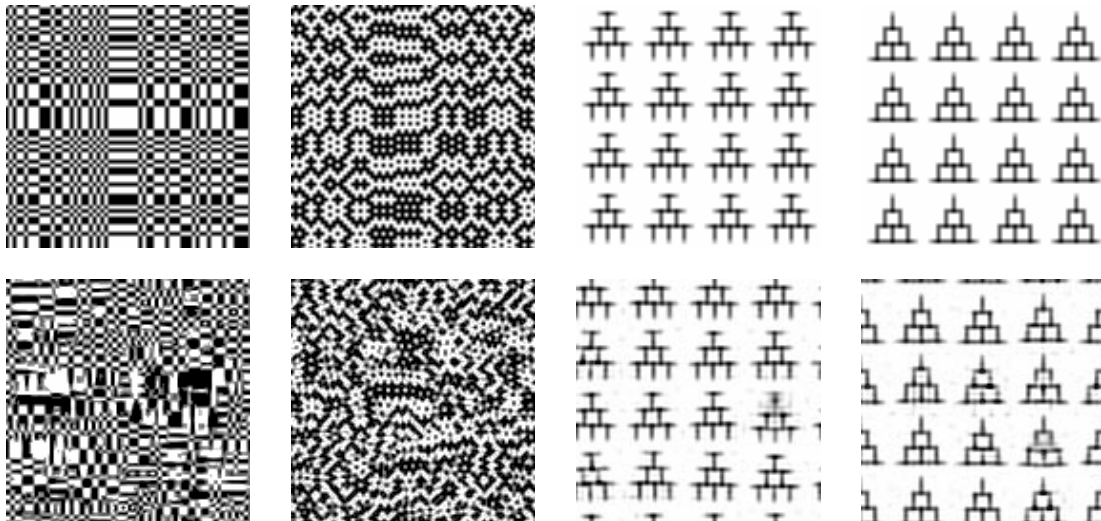
Figure 13. Synthesis of classic counterexamples to the Julesz conjecture [37, 64] (see text). Top row: original artificial textures. Bottom row: Synthesized textures.

The rightmost pair were constructed by Yellott [64], to have identical *sample* autocorrelation. Again, our model does not confuse these, and can reproduce the visual appearance of either one.

Figure 14 shows synthesis results photographic textures that are pseudo-periodic, such as a brick wall and various types of woven fabric. Figure 15 shows synthesis results for a set of photographic textures that are aperiodic, such as the animal fur or wood grain. Figure 16 shows several examples of textures with complex structures. Although the synthesis quality is not as good as in previous examples, we find the ability of our model to capture salient visual features of these textures quite remarkable. Especially notable are those examples (in all three figures) for which shading produces a strong impression of three-dimensionality.

Finally, it is instructive to apply the algorithm to images that are structured and highly inhomogeneous. Figure 17 shows a set of synthesis results on an artificial image of a "bull's eye", and photographic images of a crowd of people and a face. All three produce quite compelling results that capture the local structure of the original images, albeit in a globally disorganized fashion.

As explained in section 1.1, synthesis failures indicate insufficiency of the parameters. In keeping with the methodology prescribed in 2.2, we have created a set of artificial textures that have some of the attributes that our model fails to reproduce. Figure 18 shows synthesis examples for these examples. The first of these contains black bars of all orientations on a white background. Although a texture of single-orientation bars is reproduced fairly well (see figure 12), the mixture of bar orientations in this example leads to the synthesis of curved line segments. In general, the model is unable to distinguish straight from curved contours, except when the contours are all of the same orientation. The same type of artifact may be seen in the synthesis of pine shoots (figure 16). The second example contains bars of different grayscale value. The synthesized image does not capture the abruptness of bar endings nearly as well as the black-and-white example in figure 12. This is because the pixel marginal statistics
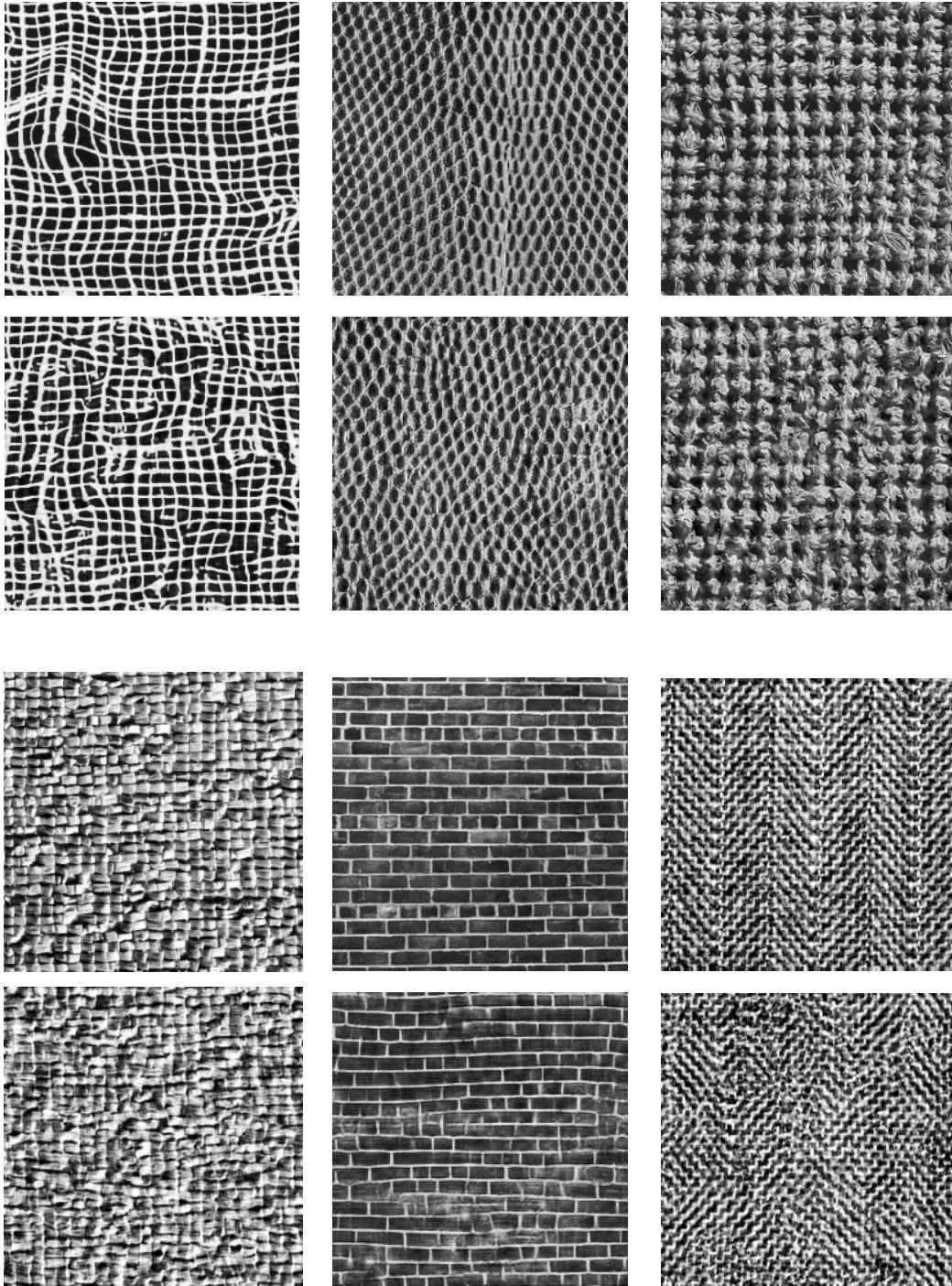
Figure 14. Synthesis results on photographic pseudo-periodic textures. See caption of figure 12.
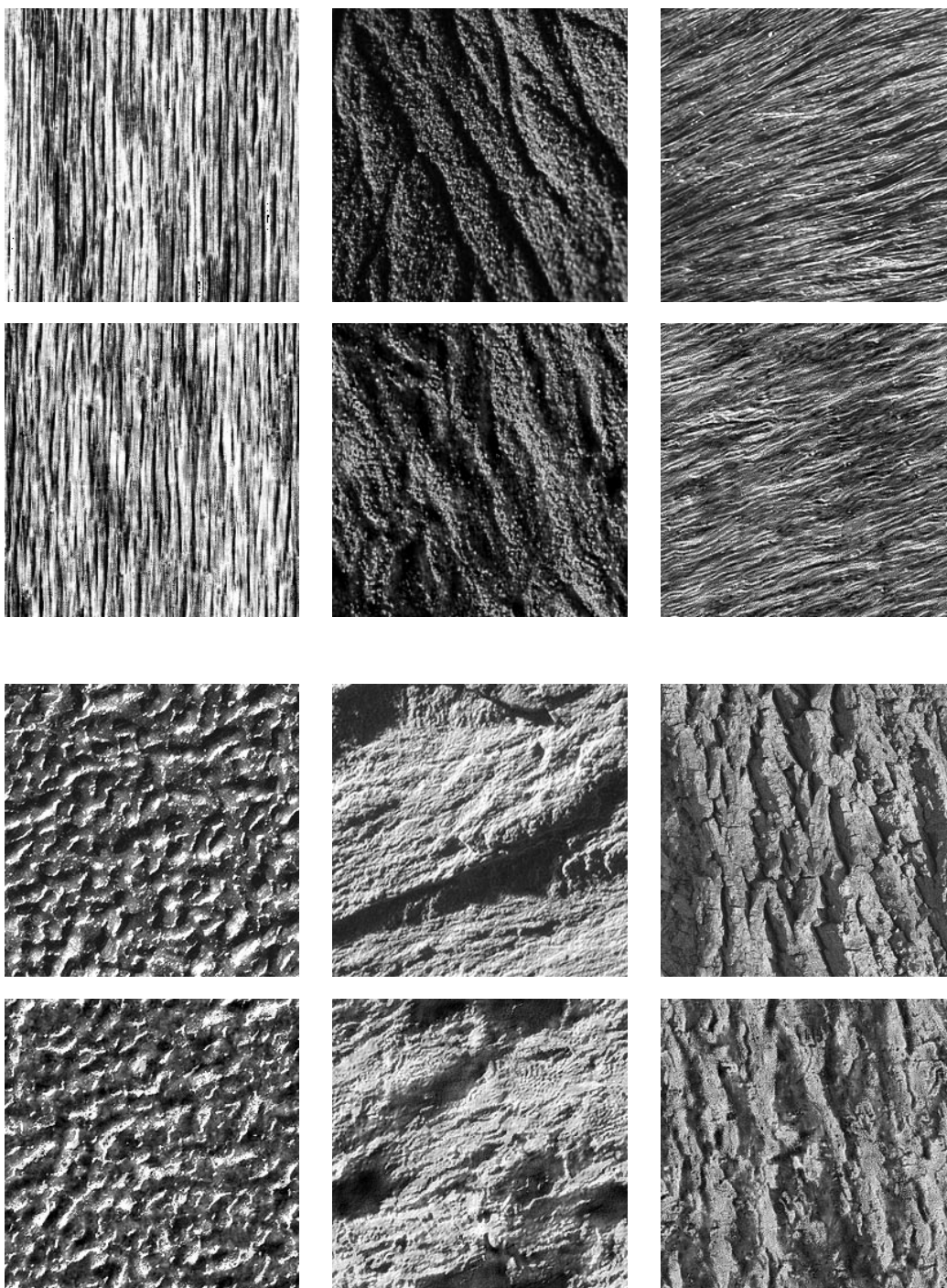
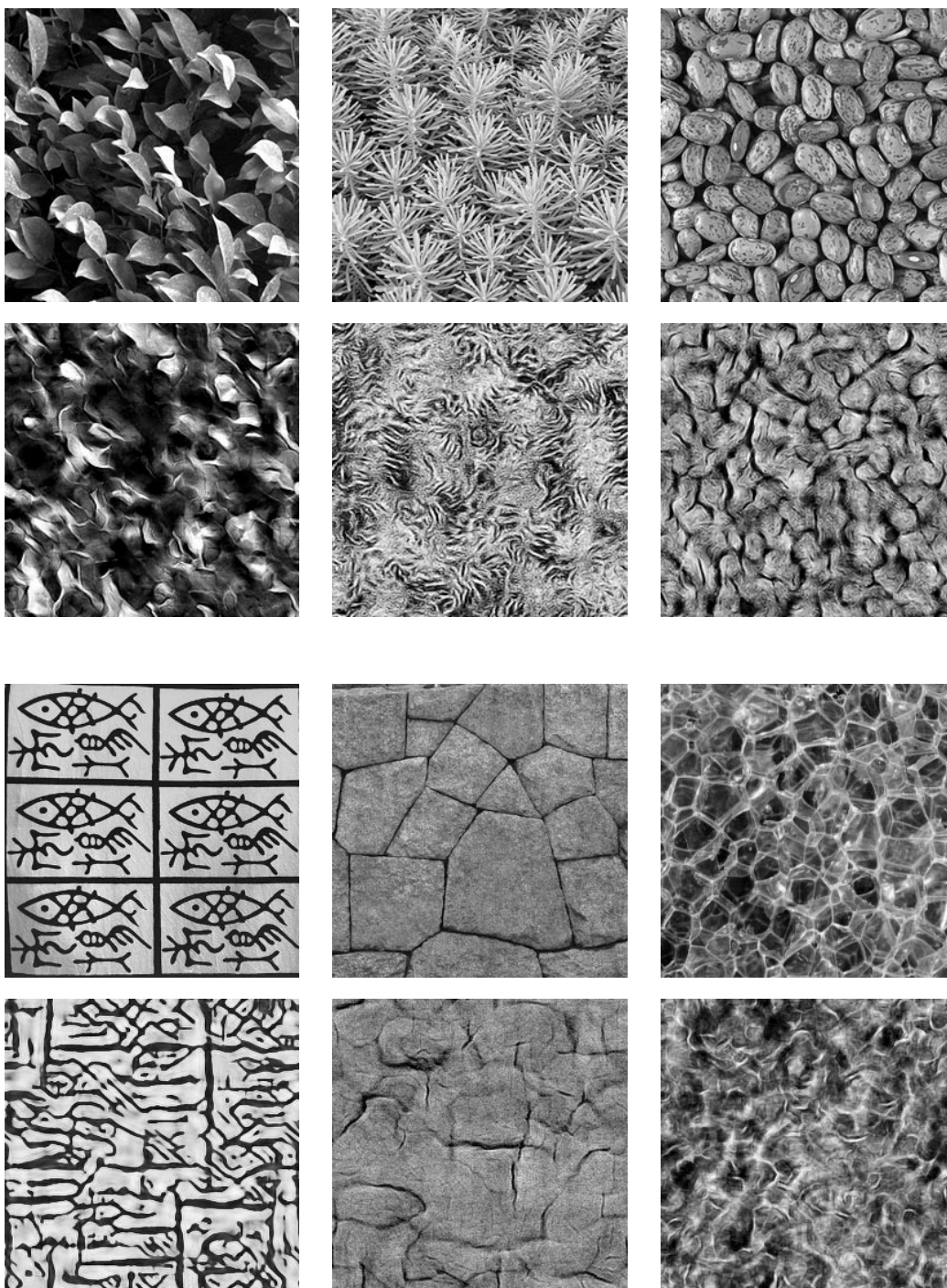**Figure 15:** Synthesis results on photographic aperiodic textures. See caption of figure 12.

.

Figure 16. Synthesis results on complex structured photographic textures. See caption of figure 12.

Figure 17. Synthesis results on inhomogeneous photographic images not usually considered to be "texture".

provide an important constraint in the black-and-white example, but are not as informative in the grayscale example. The third example contains ellipses of all orientations. Although the synthesized result contains black lines of appropriate thickness and curvature, most of them do not form closed contours. A similar artifact is present in synthesis of beans (figure 16). This and the previous problem are due in part to the lack of representation of "end points" in the model. Finally, the fourth example contains polygonal patches of constant gray value. Within each subband of the steerable pyramid, the local phase at polygon edges is equally likely to be positive as negative, and thus averages to zero. The local phase descriptor is unable to distinguish lines and edges in this case, and the resulting synthetic image has some transitions that look like bright or dark lines, rather than step edges. An analogous problem occurs with textures containing lines of different polarities. Again, the pixel marginal statistics are not beneficial in this example.

3.3 Extensions

As a demonstration of the flexibility of our approach, we can modify the algorithm to handle applications of constrained texture synthesis. In particular, consider the problem of extending a texture image beyond its spatial boundaries (spatial extrapolation). We want to synthesize an image in which the central pixels contain a copy of the original image, and the surrounding pixels are synthesized based on the statistical measurements of the original image. The set of all images with the same central subset of pixels is convex, and the projection onto such a convex set is easily inserted into the iterative loop of the synthesis algorithm. Specifically, we need only re-set the central pixels to the desired values on each iteration of the synthesis
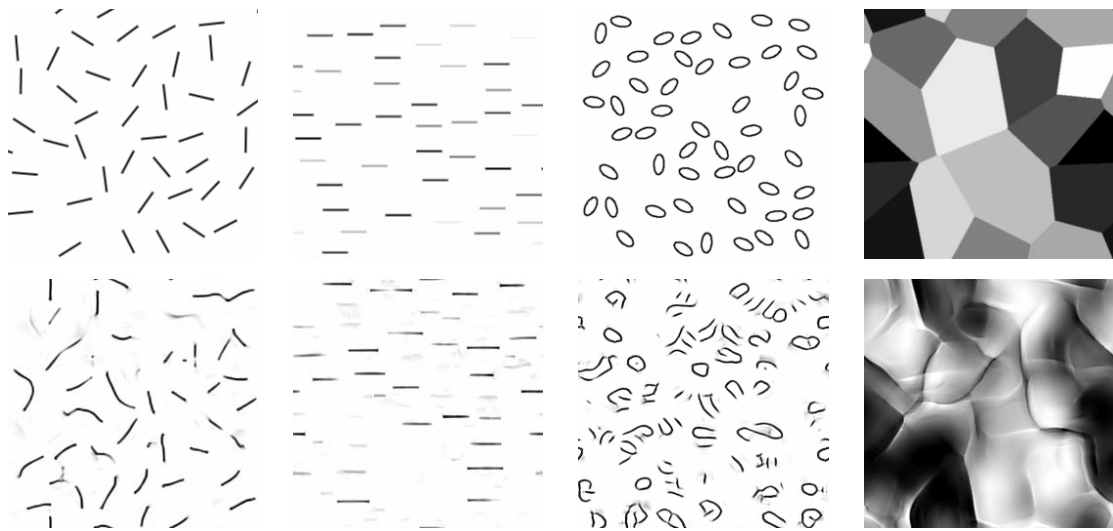
Figure 18. Artificial textures illustrating failure to synthesize certain texture attributes. See text.

loop. In practice, this substitution is done by multiplying the desired pixels by a smooth mask (a raised cosine) and adding this to the current synthesized image multiplied by the complement of this mask. The smooth masks prevents artifacts at the boundary between original and synthesized pixels, whereas convergence to the desired pixels within the mask support region is achieved almost perfectly. This technique is applicable to the restoration of pictures which have been destroyed in some subregion ("filling holes") [e.g., 32], although the estimation of parameters from the defective image is not straightforward. Figure 19 shows a set of examples that have been spatially extrapolated using this method. Observe that the border between real and synthetic data is barely noticeable. An additional potential benefit is that the synthetic images are seamlessly periodic (due to circular boundary-handling within our algorithm), and thus may be used to tile a larger image.

Finally, we consider the problem of creating a texture that lies visually "in between" two other textures. The parameter space consisting of spatial averages of local functions has a type of convexity property in the limit as the image lattice grows in size.[3] Figure 3.3 shows three images synthesized from parameters that are an average of the parameters for two example textures. In all three cases, the algorithm converges to an interesting-looking image that appears to be a patchwise mixtures of the two initial textures, rather than a new homogeneous texture that lies perceptually between them. Thus, in our model, the subset of parameters corresponding to textures (homogeneous RFs) is not convex!

---

[3]Linear combinations of the statistical measurements of two different textures correspond to the statistics of an image divided into two regions containing the two textures. Subband coefficients near the boundary will be corrupted, but in the limit as the image size goes to infinity, their contribution to the statistics goes to zero.
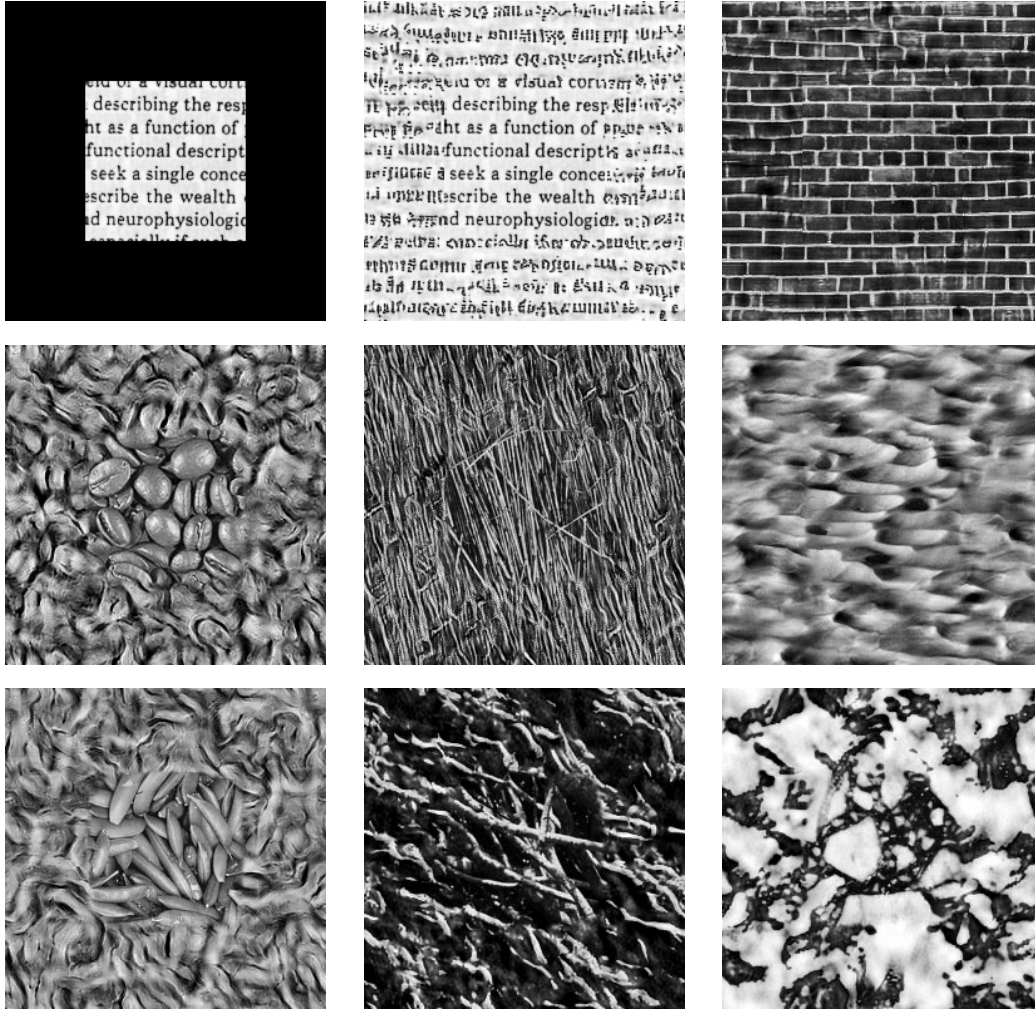
Figure 19. Spatial extrapolation of texture images. Upper left corner: example of an initial image to be extrapolated. Center shows an example texture image, surrounded by a black region indicating the pixels to be synthesized. Remaining images: extrapolated examples (central region of constrained pixels is the same size and shape in all examples).
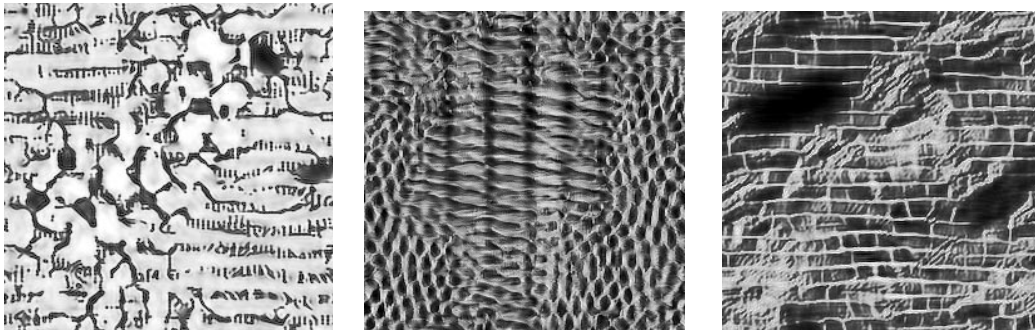
Figure 20. Examples of "mixture" textures. Left: text (figure 19) tile mosaic (figure 3); Middle: lizard skin (figure 14) and woven cane (figure 4); Right: plaster (figure 15) and brick (figure 14).

4 Discussion

We have described a universal parametric model for visual texture, based on a novel set of pairwise joint statistical constraints on the coefficients of a multi-scale image representation. We have described a framework for testing the perceptual validity of this model in the context of the Julesz conjecture, and developed a novel algorithm for synthesizing model textures using sequential projections onto the constraint surfaces. We have demonstrated the necessity of subsets of our constraints by showing examples of textures for which synthesis quality is substantially decreased when that subset is removed from the model. And we have shown the power and flexibility of the model by synthesizing a wide variety of artificial and natural textures, and by applying it to the constrained synthesis of boundary extrapolation. We find it especially remarkable that the model can reproduce textures with complex structure and lighting effects.

The most fundamental issue in our model is the choice of statistical constraints. We have motivated these through observations of structural and statistical properties of images, and "reverse-engineering" of early human visual processing, and we have refined and augmented these measurements by observing failures to synthesize particular types of texture. Nevertheless, there is room for improvement. First, the failures of figure 18 suggest that our constraint set is still insufficient. A more difficult problem is that our particular choices are not unique, and we cannot be sure that an alternative set of measurements would not produce similar (or better) synthesis results. Finally, the fact that some regions of the parameter space do not correspond to homogeneous RFs (see the mixture textures in figure 3.3) indicates that the current parameterization should be rearranged to give rise to a more useful topological distribution of the subset of parameters that correspond to textures. Convexity of this subset, in particular, would bring us closer to defining a perceptual distance metric for texture, and would allow us to interpolate between texture examples.

Our synthesis algorithm is closest to (and is, in fact, a generalization of) that of Heeger and Bergen [30]. It has the drawback that we cannot prove convergence. In our experience with hundreds of texture examples, however, the algorithm has never failed to converge to an image that nearly satisfies the statistical constraints. A deeper problem is that our model (in particular, the implicit probability density on the space of images) is not fully determined by

the set of constraints, but is also determined by the sampling algorithm.

It is worth comparing our model with several recent successful texture models. By far the most successful approaches in terms of visual appearance have been the non-parametric sampling techniques (notably, [45, 19, 22]). Despite the beautiful synthesis results, however, these techniques do not provide a compact representation of texture (the representation *is* the example texture image). Additionally, they are not easily extensible to problems that require one to infer a model from corrupted or partial information.

Amongst the parametric approaches, high quality results have been obtained using marginal histograms of filter responses as statistical constraints [30, 68]. In the case of our particular basis set, we find that marginal statistics are insufficient to represent many of the more complex textures. This is not to say that the marginals of a larger set of filters would not be sufficient. In fact, Zhu *et al.* have shown, using a variant of the Fourier projection-slice theorem used for tomographic reconstruction, that large numbers of marginals are sufficient to uniquely constrain a high-dimensional probability density [68]. But increasing the number of marginals, in addition to making the representation less compact, will also slow down the convergence of most sampling algorithms. It remains to be seen whether simple models of the joint statistics, such as that introduced in this paper, can provide a more efficient representation.

Another interesting concept used in a number of recent models is to adapt the basis to the statistical properties of each individual texture example [7, 25, 48, 70, 42]. The model by Zhu *et al.*, in particular, produces high-quality synthesis examples using a surprisingly small set of filters. The drawback with this approach is the additional computational expense (often substantial) in the filter-selection process.

We envision a number of extensions of our texture modeling approach. The results can be made much more visually compelling by introducing color[4]. Preliminary results indicate that the alternating projections approach can be applied to other problems such as denoising [50]. Since the representation of a texture image is quite compact, the synthesis technique might be used in conjunction with a compression system in order to "fabricate" detail rather than encode it exactly. Finally, we believe our synthesis approach might be extended for use in constrained synthesis applications such as repairing defects (*e.g.*, scratches or holes), "super-resolution" zooming, and "painting" texture onto an image [*e.g.*, 31].

A   Adjustment of Constraints

In this appendix, we give the mathematical details of the method by which each group of statistical constraints is imposed. Throughout, we use the symbol t to denote the statistics estimated from the original sample image $\vec{x}_t$.

A.1   Adjusting Marginal Statistics

We adjust the mean, variance, skew, kurtosis and range using the following procedure: 1) subtract the sample mean, 2) adjust the sample skewness, 3) adjust the sample kurtosis, 4) adjust the sample variance and mean, and 5) adjust the range. We denote the central sample

---

[4]Examples are available on the internet at `http://www.cns.nyu.edu/~lcv/texture/`.

moments as follows:

$$\mu_n(\vec{x}) = \begin{cases} \overline{x(i,j)}, & n = 1 \\ \overline{(x(i,j) - \mu_1(\vec{x}))^n}, & n > 1. \end{cases} \tag{12}$$

For readability, we typically drop the dependence on $\vec{x}$. The coefficients of skewness and kurtosis are defined as

$$\eta(\vec{x}) = \frac{\mu_3(\vec{x})}{(\mu_2(\vec{x}))^{1.5}} \qquad \kappa(\vec{x}) = \frac{\mu_4(\vec{x})}{(\mu_2(\vec{x}))^2}. \tag{13}$$

In the following paragraphs, we describe the adjustment of each parameter.

Mean and Variance. These adjustments are accomplished in the usual way:

$$\vec{x}\,' = (\vec{x} - \mu_1)\sqrt{\frac{\mu_2^t}{\mu_2}} + \mu_1^t,$$

which corresponds to two gradient (and, in this case, orthogonal) projections.

Range. The range of $\vec{x}$ is an exception in our set of constraints, because it can not be written as the expectation of a function applied to the image. We include it nevertheless, because it helps to improve the visual appearance of the results, as well as the convergence of the algorithm. The adjustment is made by clipping those pixel values that are outside the desired range. This procedure is an orthogonal projection of the image onto a hypercube of dimension $|L|$, which is a convex set.

Skewness. In order to adjust the skewness, we assume $\mu_1(\vec{x}) = 0$, and compute the components of $\vec{\nabla}\eta(\vec{x})$:

$$\frac{\partial \eta}{\partial x(i,j)} = \frac{3}{|L|\mu_2^{3/2}}\left(x(i,j)^2 - \mu_2^{1/2}\eta x(i,j) - \mu_2\right). \tag{14}$$

Dropping the scale factor and vectorizing, we define a vector in the direction of the gradient:

$$\vec{g} \equiv \vec{x} \odot \vec{x} - \mu_2^{1/2}\eta\,\vec{x} - \mu_2, \tag{15}$$

where $\odot$ indicates element-wise multiplication. Following equation (8), we set the sample skewness of the projected image equal to the desired skewness:

$$\eta(\vec{x} + \lambda\vec{g}) = \frac{\mu_3(\vec{x} + \lambda\vec{g})}{\mu_2^{3/2}(\vec{x} + \lambda\vec{g})} = \eta_t. \tag{16}$$

Using the fact that $\mu_1(\vec{x} + \lambda\vec{g}) = 0$ when $\mu_1(\vec{x}) = 0$, and substituting equation (15) into (16), we obtain:

$$\eta_t = \frac{\sum_{n=0}^3 p_n \lambda^n}{\left[\sum_{n=0}^2 q_n \lambda^n\right]^{3/2}}, \tag{17}$$

30

where

$$
\begin{aligned}
p_3 &= \mu_6 - 3\mu_2^{1/2}\eta\mu_5 + 3\mu_2(\eta^2 - 1)\mu_4 + \mu_2^3(2 + 3\eta^2 - \eta^4) \\
p_2 &= 3(\mu_5 - 2\mu_2^{1/2}\eta\mu_4 + \mu_2^{5/2}\eta^3) \\
p_1 &= 3(\mu_4 - \mu_2^2(1 + \eta^2)) \\
p_0 &= \eta\mu_2^{3/2} \\
q_2 &= \mu_4 - (1 + \eta^2)\mu_2^2 \\
q_1 &= 0 \\
q_0 &= \mu_2.
\end{aligned}
\tag{18}
$$

Squaring both sides of equation (17), yields the algebraic equation:

$$
\sum_{n=0}^{6} a_n \lambda^n = 0,
\tag{19}
$$

where

$$
\begin{aligned}
a_6 &= p_3^2 - \eta_t^2 q_2^3, \\
a_5 &= 2p_3 p_2 \\
a_4 &= p_2^2 + 2p_3 p_1 - 3\eta_t^2 q_2^2 q_0 \\
a_3 &= 2(p_3 p_0 + p_2 p_1) \\
a_2 &= p_1^2 + 2p_2 p_0 - 3\eta_t^2 q_2 q_0^2 \\
a_1 &= 2p_1 p_0 \\
a_0 &= p_0^2 - \eta_t^2 q_0^3.
\end{aligned}
\tag{20}
$$

From the six solutions of equation (19) we choose the minimum amplitude solution satisfying equation (16) in the interval around $\lambda = 0$ where $\eta(\lambda)$ has positive slope. When there is no solution within this interval, one of its extreme values is chosen.

Kurtosis. This adjustment is analogous to the skewness case. Assuming $\mu_1(\vec{x}) = 0$, we define a vector in the direction of the gradient:

$$
\vec{g} \equiv \vec{x} \odot \vec{x} \odot \vec{x} - \alpha\vec{x} - \mu_3,
$$

where $\alpha = \mu_4/\mu_2$. Applying equation (8) and the definition (13) we obtain

$$
\kappa_t = \frac{\sum_{n=0}^{4} p_n \lambda^n}{\left[\sum_{n=0}^{2} q_n \lambda^n\right]^2},
\tag{21}
$$

where

$$
\begin{aligned}
p_4 =\ & \mu_{12} - 4\alpha\mu_{10} - 4\mu_3\mu_9 + 6\alpha^2\mu_8 + 12\alpha\mu_3\mu_7 + 6\mu_3^2\mu_6 - 4\alpha^3\mu_6 - 12\alpha^2\mu_3\mu_5 \\
& + \alpha^4\mu_4 - 12\alpha\mu_3^2\mu_4 + 4\alpha^3\mu_3^2 + 6\alpha^2\mu_3^2\mu_2 - 3\mu_3^4 \\
p_3 =\ & 4\left(\mu_{10} - 3\alpha\mu_8 - 3\mu_3\mu_7 + 3\alpha^2\mu_6 + 6\alpha\mu_3\mu_5 + 3\mu_3^2\mu_4 - \alpha^3\mu_4 - 3\alpha^2\mu_3^2 - 3\mu_4\mu_3^2\right) \\
p_2 =\ & 6\left(\mu_8 - 2\alpha\mu_6 - 2\mu_3\mu_5 + \alpha^2\mu_4 + (\mu_2 + 2\alpha)\mu_3^2\right) \\
p_1 =\ & 4\left(\mu_6 - \alpha^2\mu_2 - \mu_3^2\right) \\
p_0 =\ & \mu_4 \\
q_2 =\ & p_1/4 \\
q_1 =\ & 0 \\
q_0 =\ & \mu_2.
\end{aligned}
\tag{22}
$$

We now solve the fourth degree algebraic equation given in (21), and follow the same criterion as with the skewness for choosing a solution.

A.2   Adjusting Subband Auto-correlation

Assuming L is a rectangular grid of $N \times M$ pixels, we will use the following estimator for the (circular) autocorrelation of an image $\vec{x}$:

$$A(n, m) = \overline{x(i, j) x(\mid i + n \mid_N, \mid j + m \mid_M)}, \tag{23}$$

where $\mid a + b \mid_N$ indicates the sum of a and b modulo N. The partial derivatives of this statistic with respect to the pixel at location (i, j) are given by

$$\frac{\partial A(n, m)}{\partial x(i, j)} = \frac{1}{|L|} \left[x(\mid i + n \mid_N, \mid j + m \mid_M) + x(\mid i - n \mid_N, \mid j - m \mid_M) \right]. \tag{24}$$

Following equation (10) we arrive at an expression for the projected image:

$$x\,'(i, j) = x(i, j) + \frac{1}{|L|} \sum_{(n,m) \in \mathcal{N}} \lambda_{n,m} \left[x(\mid i + n \mid_N, \mid j + m \mid_M) + x(\mid i - n \mid_N, \mid j - m \mid_M) \right], \tag{25}$$

where $\mathcal{N}$ is the neighborhood composed by all the (one-sided, non-redundant) samples (n, m) of the autocorrelation we want to adjust (for example, if we choose to fix the central 7×7 samples of the autocorrelation, then $|\mathcal{N}| = (7^2 + 1)/2 = 25$), and $\{\lambda_{n,m}\}$ is a set of unknown real constants. We can write the previous expression as

$$x\,'(i, j) = x(i, j) \otimes h_{\vec{\lambda}}(i, j), \tag{26}$$

where $\otimes$ indicates circular convolution and $h_{\vec{\lambda}}(i, j)$ is an even-symmetric kernel with spatial support equal to a symmetrized version of $\mathcal{N}$. Thus, the autocorrelation is adjusted by applying a zero-phase moving average filter to the image.[5]

We can solve for the (non-redundant) samples of the filter $h_{\vec{\lambda}}$ by setting the autocorrelation samples of the updated image to be the desired ones:

$$A\,'(n, m) = \sum_{(k,l) \in \mathcal{N}} \sum_{(p,q) \in \mathcal{N}} h_{\vec{\lambda}}(k, l) h_{\vec{\lambda}}(p, q) \tag{27}$$
$$[A_t(n + p + k, m + q + l) + A_t(n - p + k, m - q + l) +$$
$$A_t(n + p - k, m + q - l) + A_t(n - p - k, m - q - l)], \qquad \forall (n, m) \in \mathcal{N}.$$

This system of $|\mathcal{N}|$ quadratic equations and $|\mathcal{N}|$ unknowns might have up to $2^{|\mathcal{N}|}$ valid solutions. As we are interested in a filter that changes the image as little as possible, a natural initial guess for the solution is the identity filter, $\delta(i, j)$ (the Kronecker delta). However, a better initial guess can be computed by solving for the even symmetric kernel $A_h(n, m)$ with $2|\mathcal{N}| - 1$ non-zero samples that satisfies

$$A_t(n, m) = A(n, m) * A_h(n, m), \tag{28}$$

for $(n, m) \in \mathcal{N}$, where $*$ indicates convolution.

Arranging the samples of the current autocorrelation $A(n, m)$ that influence the convolution result in a circulant matrix and A_t and A_h in column vectors, the previous expression

---

[5]Accordingly, if we wanted to fix the autocorrelation using an orthogonal projection, we would use the inverse operation: a zero-phase auto-regressive (all pole) filter localized in $\mathcal{N}$. But the calculation of such a filter is more difficult than in the moving average case.

can be written as a system of $|\mathcal{N}|$ linear equations and $|\mathcal{N}|$ unknowns. When the resulting A_h is positive definite (i.e., its DFT, which is real, is non negative), this kernel can be interpreted as $A_h(n,m) = h_{\vec{\chi}}(n,m) * h_{\vec{\chi}}(-n,-m)$, and we can solve for a $h_{\vec{\chi}}$ satisfying the constraints by computing the square root of its Fourier transform and inverse transforming. In general, this Fourier spectrum will not be positive, and we will have to use the square root of its absolute value:

$$h_{\vec{\chi}}(i,j) \simeq DFT^{-1}\left\{\sqrt{|DFT\left\{A_h(i,j)\right\}|}\right\}. \tag{29}$$

Despite the use of this somewhat crude approximation, we obtain accuracy typically greater 70 dB SNR after only a few iterations of the synthesis loop.

A.3 Adjusting Subband Cross-Correlations

At the coarsest scale of the pyramid decomposition, we need to adjust the cross-correlation of coefficient magnitudes at all K orientations. Given the set of subbands $\{x_k(i,j), k = 1\ldots K\}$, the elements of the $K \times K$ correlation matrix $\mathbf{C}$ are computed as:

$$C_{n,m} = \overline{x_n(i,j)x_m(i,j)}. \tag{30}$$

The partial derivatives of these matrix elements with respect to a coefficient of the kth subband are:

$$\frac{\partial C_{n,m}}{\partial x_k(i,j)} = \begin{cases} \frac{1}{|L|}x_n(i,j), & m = k \\ \frac{1}{|L|}x_m(i,j), & n = k \\ 0, & \text{otherwise.} \end{cases} \tag{31}$$

Applying equation (8) for obtaining the gradient, vectorizing, and dropping the scale factor yields:

$$\vec{x}_n{}' = \vec{x}_n + \sum_{m=1}^{K} \lambda_{n,m}\vec{x}_m, \tag{32}$$

for $n = 1\ldots K$, where $\lambda_{n,m} = \lambda_{m,n}$. That is, the updated subbands are a symmetric linear combination of the original subbands. Arranging our K vectors $\vec{x}_k$ as the rows of a matrix $\mathbf{X}$, we can express the previous equation as:

$$\mathbf{X}' = \mathbf{MX}, \tag{33}$$

where $\mathbf{M}$ is a symmetric $K \times K$ matrix containing the $\lambda_{n,m}$. The cross-correlation of the original subbands may be expressed in matrix form: $\mathbf{C} = \frac{1}{|L|}\mathbf{XX}^T$.

We must now solve for $\mathbf{M}$ that achieves the desired cross-correlation. This may be expressed as:

$$\frac{1}{|L|}\mathbf{X}'(\mathbf{X}')^T = \frac{1}{|L|}\mathbf{MX}(\mathbf{MX})^T = \mathbf{MCM}^T = \mathbf{C_t}, \tag{34}$$

corresponding to a system of $K(K+1)/2$ quadratic equations and the same number of unknowns (distinct elements of $\mathbf{M}$). As in the case of the autocorrelation, we can obtain a good approximation for a solution of this system using the following direct technique. First, we compute the eigenvector factorization of both the desired and the current correlation matrices:

$$\mathbf{C} = \mathbf{VDV}^T, \qquad \mathbf{C_t} = \mathbf{V_t D_t V_t}^T, \tag{35}$$

where the $\mathbf{D}$ matrices are diagonal with strictly positive entries, and the $\mathbf{V}$ matrices are orthogonal. By substituting these eigenvector expansions into equation (34), it is easy to verify that the complete set of solutions for $\mathbf{M}$ are of the form:

$$\mathbf{M} = \mathbf{V_t}\mathbf{D_t}^{1/2}\mathbf{O}\mathbf{D}^{-1/2}\mathbf{V}^T, \tag{36}$$

where $\mathbf{O}$ is any orthonormal matrix. A solution that is nearly optimal (specifically, it becomes symmetric as $\mathbf{C} \to \mathbf{C_t}$) is:

$$\mathbf{O} = \mathbf{V_t}^T\mathbf{V}. \tag{37}$$

In contrast with the auto-correlation adjustment, this solution provides an exact adjustment of the cross-correlation.

A.4 Adjusting Cross-Correlation with Fixed Subbands

Our sequential algorithm proceeds from coarse to fine scales. At all scales but the coarsest, we need not adjust only the cross-correlation of coefficients across the K orientations, but also the cross-correlation with the previously adjusted subbands at the coarser scale. This is done for both the coefficient magnitudes, and also for the phase-doubled complex coefficients. We define the cross-correlation matrix $\mathbf{B}$ of the subbands to be adjusted with a set of M_s subbands $\{y_k(i,j)\}$ defined on the same lattice L:

$$B_{n,m} = \overline{x_n(i,j)y_m(i,j)}. \tag{38}$$

Differences in the sampling rates between the two subbands are eliminated by upsampling and interpolating the coarse-scale bands.

The gradient contains the following components:

$$\frac{\partial B_{n,m}}{\partial x_k(i,j)} = \begin{cases} \frac{1}{|L|}y_m(i,j), & n = k \\ 0, & \text{otherwise.} \end{cases} \tag{39}$$

Combining these components with those computed in the previous section, vectorizing, and discarding scale factors, we obtain an expression for the updated coefficients:

$$\vec{x}_k{}' = \vec{x}_k + \sum_{n=1}^{K} \lambda_{k,n}\vec{x}_n + \sum_{m=1}^{M_s} \mu_{k,m}\vec{y}_m, \tag{40}$$

for $k = 1\ldots K$. As before, $\lambda_{k,n} = \lambda_{n,k}$, but note that $\mu_{k,m}$ need not exhibit symmetry. Analogous to the development of the previous section, we write this equation in a matrix form:

$$\mathbf{X}' = \mathbf{M}\mathbf{X} + \mathbf{K}\mathbf{Y}, \tag{41}$$

where $\mathbf{M}$ is an unknown $K \times K$ symmetric matrix, and $\mathbf{K}$ is an unknown $M_s \times K$ matrix. We must now solve for $\mathbf{M}$ and $\mathbf{K}$ that achieve the desired cross-correlations:

$$\begin{aligned}
\frac{1}{|L|}\mathbf{X}'(\mathbf{X}')^T &= \frac{1}{|L|}\left[\mathbf{M}\mathbf{X}\mathbf{X}^T\mathbf{M}^T + \mathbf{M}\mathbf{X}\mathbf{Y}^T\mathbf{K}^T + \mathbf{K}\mathbf{Y}\mathbf{X}^T\mathbf{M}^T + \mathbf{K}\mathbf{Y}\mathbf{Y}^T\mathbf{K}^T\right] \\
&= \mathbf{M}\mathbf{C}\mathbf{M}^T + \mathbf{M}\mathbf{B}\mathbf{K}^T + \mathbf{K}\mathbf{B}^T\mathbf{M}^T + \mathbf{K}\mathbf{E}\mathbf{K}^T \\
&= \mathbf{C_t}, \tag{42}
\end{aligned}$$

and

$$\frac{1}{|L|}\mathbf{X}\,'\mathbf{Y}^T = \frac{1}{|L|}\left[\mathbf{MXY}^T + \mathbf{KYY}^T\right]$$
$$= \mathbf{MB} + \mathbf{KE}$$
$$= \mathbf{B_t}, \tag{43}$$

where $\mathbf{E} = \mathbf{YY}^T$. Solving equation (43) for $\mathbf{K}$ gives:

$$\mathbf{K} = (\mathbf{B}_t - \mathbf{MB})\mathbf{E}^{-1}. \tag{44}$$

Substituting this into equation (42) gives a symmetric quadratic constraint on $\mathbf{M}$:

$$\mathbf{M}\left[\mathbf{C} - \mathbf{BE}^{-1}\mathbf{B}^T\right]\mathbf{M}^T = \mathbf{C}_t - \mathbf{B}_t\mathbf{E}^{-1}\mathbf{B}_t^T. \tag{45}$$

This constraint is in the same form as equation (34), and thus the solution for $\mathbf{M}$ may be found using the same eigenvalue technique. The solution for $\mathbf{K}$ is then obtained by substituting $\mathbf{M}$ into equation (44).

References

[1] C H Anderson and W D Langer. Statistical models of image texture. Technical report, Washington U. Medical School, 1997. Available at ftp://shifter.wustl.edu/pub/.

[2] A J Bell and T J Sejnowski. The 'independent components' of natural scenes are edge filters. *Vision Research*, 37(23):3327–3338, 1997.

[3] J R Bergen and E H Adelson. Visual texture segmentation based on energy measures. *J. Opt. Soc. Am. A*, 3:99, 1986.

[4] J R Bergen and M S Landy. Computational modeling of visual texture segregation. In M S Landy and J A Movshon, editors, *Computational Models of Visual Processing*, pages 253–271. MIT Press, Cambridge, MA, 1991.

[5] C A Bouman and M Shapiro. A multiscale random field model for Bayesian image segmentation. *IEEE Trans. Image Proc.*, 3(2), 1994.

[6] A C Bovik, M Clark, and W S Geisler. Multichannel texture analysis using localized spatial filters. *IEEE Pat. Anal. Mach. Intell.*, 12(1):55–73, 1990.

[7] A C Bovik, M Clark, and W S Geisler. Localized measurements of emergent image frequencies by Gabor wavelets. *IEEE Pat. Anal. Mach. Intell.*, 38:691–712, 1992.

[8] P Brodatz. *Textures: A Photographic Album for Artists and Designers*. Dover, New York, 1966.

[9] R W Buccigrossi and E P Simoncelli. Image compression via joint statistical characterization in the wavelet domain. *IEEE Trans Image Proc*, 8(12):1688–1701, December 1999.

[10] J A Cadzow, D M Wilkes, R A Peters II, and X Li. Image texture synthesis–by–analysis using moving–average models. *IEEE Trans. Aerospace and Elec. Sys.*, 29(4):1110–1122, 1993.

[11] T M Caelli and B Julesz. Experiments in the visual perception of texture. *Biol. Cybernetics*, 28:167–175, 1978.

[12] D Cano and T H Minh. Texture synthesis using hierarchical linear transforms. *Signal Processing*, 15:131–148, 1988.

[13] P C Chen and T Pavlidis. Segmentation by texture using correlation. *IEEE Pat. Anal. Mach. Intell.*, 5(1):64–69, 1983.

[14] G Cross and A Jain. Markov random field texture models. *IEEE Trans PAMI*, 5:25–39, 1983.

[15] I Daubechies. Orthonormal bases of compactly supported wavelets. *Comm. on Pure and Appl. Math.*, 41:909–996, 1988.

[16] J G Daugman. Complete discrete 2–D Gabor transforms by neural networks for image analysis and compression. *IEEE Trans. Acoust. Speech Signal Proc.*, 36(7):1169–1179, 1988.

[17] J G Daugman and D M Kammen. Pure orientation filtering: A scale-invariant image-processing tool for perception research and data compression. *Behavior Research Methods, Instruments, & Computers*, 18(6):559–564, 1986.

[18] John G. Daugman. Entropy reduction and decorrelation in visual coding by oriented neural receptive fields. *IEEE Trans. Biomedical Engineering*, 36(1):107–114, 1989.

[19] J De Bonet and P Viola. A non-parametric multi-scale statistical model for natural images. In *Adv in Neural Info Processing*, volume 9. MIT Press, December 1997.

[20] H Derin and H Elliott. Modeling and segmentation of noisy and textured images using Gibbs random fields. *IEEE Pat. Anal. Mach. Intell.*, 9(1):39–55, 1987.

[21] P Diaconis and D Freedman. On the statistics of vision: the julesz conjecture. *J. Math. Psychol*, 24:112–118, 1981.

[22] A A Efros and T K Leung. Texture synthesis by non-parameteric sampling. In *Proc. Int'l Conference on Computer Vision*, Corfu, 1999.

[23] O D Faugeras and W K Pratt. Decorrelation methods of texture feature extraction. *IEEE Pat. Anal. Mach. Intell.*, 2(4):323–332, 1980.

[24] D J Field. Relations between the statistics of natural images and the response properties of cortical cells. *J. Opt. Soc. Am. A*, 4(12):2379–2394, 1987.

[25] J M Francos, A Z Meiri, and B Porat. A unified texture model based on a 2-D Wold-like decomposition. *IEEE Trans. Signal Proc.*, 41(8):2665–2678, 1993.

[26] A Gagalowicz. A new method for texture fields synthesis: Some applications to the study of human vision. *IEEE Pat. Anal. Mach. Intell.*, 3(5):520–533, 1981.

[27] S Geman and D Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Pat. Anal. Mach. Intell.*, 6:721–741, 1984.

[28] N Graham. *Visual pattern analyzers*. Oxford University Press, New York, 1989.

[29] M Hassner and J Sklansky. The use of Markov random fields as models of texture. *Comp. Graphics Image Proc.*, 12:357–370, 1980.

[30] D Heeger and J Bergen. Pyramid-based texture analysis/synthesis. In *Proc. ACM SIGGRAPH*, August 1995.

[31] Aaron Hertzmann. Painterly rendering with curved brush strokes of multiple sizes. In *Proc. ACM SigGraph*, pages 453–460, 1998.

[32] A N Hirani and T Totsuka. Combining frequency and spatial domain information for fast inter-active image noise removal. In *ACM SIGGRAPH*, pages 269–276, 1996.

[33] E T Jaynes. Information theory and statistical mechanics. *Phys. Rev.*, 106:620–630, 1957.

[34] E T Jaynes. Where do we stand on maximum entropy? In R. D. Levine and M. Tribus, editors, *The Maximal Entropy Formalism*. MIT Press, Cambridge, MA, 1978.

[35] B Julesz. Visual pattern discrimination. *IRE Trans Info Theory*, IT-8, 1962.

[36] B Julesz, E N Gilbert, L A Shepp, and H L Frisch. Inability of humans to discriminate between visual textures that agree in second-order statistics – revisited. *Perception*, 2:391–405, 1973.

[37] B Julesz, E N Gilbert, and J D Victor. Visual discrimination of textures with identical third-order statistics. *Biol. Cybernetics*, 31:137–140, 1978.

[38] D Kersten. Predictability and redundancy of natural images. *J. Opt. Soc. Am. A*, 4(12):2395–2400, 1987.

[39] H Knutsson and G H Granlund. Texture analysis using two-dimensional quadrature filters. In *Workshop on Computer Architecture for Pattern Analysis and Image Database Management*, pages 206–213. IEEE Computer Society, 1983.

[40] J Malik and P Perona. Preattentive texture discrimination with early vision mechanisms. *J. Opt. Soc. Am. A*, 7:923–932, May 1990.

[41] S G Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Pat. Anal. Mach. Intell.*, 11:674–693, July 1989.

[42] R Manduchi and J Portilla. Independent component analysis of textures. In *Proc. Int'l Conference on Computer Vision*, Corfu, 1999.

[43] B A Olshausen and D J Field. Natural image statistics and efficient coding. *Network: Computation in Neural Systems*, 7:333–339, 1996.

[44] P Perona and J Malik. Detecting and localizing edges composed of steps, peaks and roofs. In *Proc. 3rd Intl. Conf. Computer Vision*, Osaka, Japan, 1990.

[45] K Popat and R W Picard. Novel cluster-based probability model for texture synthesis, classification, and compression. In *Proc SPIE Vis Comm*, Cambridge, MA, 1993.

[46] K Popat and R W Picard. Cluster-based probability model and its application to image and texture processing. *IEEE Trans Im Proc*, 6(2):268–284, 1997.

[47] M Porat and Y Y Zeevi. Localized texture processing in vision: Analysis and synthesis in Gaborian space. *IEEE Trans Biomedical Eng.*, 36(1):115–129, 1989.

[48] J Portilla, R Navarro, O Nestares, and A Tabernero. Texture synthesis-by-analysis based on a multiscale early-vision model. *Optical Engineering*, 35(8):2403–2417, 1996.

[49] J Portilla and E Simoncelli. Texture modeling and synthesis using joint statistics of complex wavelet coefficients. In *IEEE Workshop on Statistical and Computational Theories of Vision*, Fort Collins, CO, 1999. Available at http://www.cis.ohio-state.edu/~szhu/SCTV99.html.

[50] J Portilla and E Simoncelli. Image denoising via adjustment of wavelet coefficient magnitude correlation. In *Seventh IEEE Int'l Conf on Image Proc*, Vancouver, September 10-13 2000. IEEE Computer Society.

[51] W K Pratt, O D Faugeras, and A Gagolowicz. Visual discrimination of stochastic texture fields. *IEEE Trans. on Systems Man and Cybernetics*, 8:796–804, Nov 1978.

[52] T R Reed and H Wechsler. Segmentation of textured images and Gestalt organization using spatial/spatial-frequency representations. *IEEE Pat. Anal. Mach. Intell.*, 12(1):1–12, 1990.

[53] D L Ruderman and W Bialek. Statistics of natural images: Scaling in the woods. *Phys. Rev. Letters*, 73(6):814–817, 1994.

[54] E Simoncelli and J Portilla. Texture characterization via joint statistics of wavelet coefficient magnitudes. In *Fifth IEEE Int'l Conf on Image Proc*, volume I, Chicago, October 4-7 1998. IEEE Computer Society.

[55] E P Simoncelli. Statistical models for images: Compression, restoration and synthesis. In *31st Asilomar Conf on Signals, Systems and Computers*, pages 673–678, Pacific Grove, CA, November 1997. IEEE Computer Society. Available from http://www.cns.nyu.edu/~eero/publications.html.

[56] E P Simoncelli and W T Freeman. The steerable pyramid: A flexible architecture for multi-scale derivative computation. In *Second Int'l Conf on Image Proc*, volume III, pages 444–447, Washington, DC, October 1995. IEEE Sig Proc Society.

[57] E P Simoncelli, W T Freeman, E H Adelson, and D J Heeger. Shiftable multi-scale transforms. *IEEE Trans Information Theory*, 38(2):587–607, March 1992. Special Issue on Wavelets.

[58] A Tabernero, J Portilla, and R Navarro. Duality of log-polar image representations in the space and the spatial-frequency domains. *IEEE Trans. on Signal Processing*, 47(9):2469–79, September 1999.

[59] M R Turner. Texture discrimination by Gabor functions. *Biol. Cybern.*, 55:71–82, 1986.

[60] J D Victor. Images, statistics and textures: Implications of triple correlation uniqueness for texture statistics and the Julesz conjecture: Comment. *J. Opt. Soc. Am. A*, 11(5):1680–1684, May 1994.

[61] VisTex: an online collection of visual textures. MIT Media Laboratory, 1995. Available from http://www-white.media.mit.edu/vismod/imagery/VisionTexture/vistex.html.

[62] M J Wainwright and E P Simoncelli. Scale mixtures of Gaussians and the statistics of natural images. In S. A. Solla, T. K. Leen, and K.-R. Müller, editors, *Adv. Neural Information Processing Systems*, volume 12, pages 855–861, Cambridge, MA, May 2000. MIT Press. Presented at Neural Information Processing Systems, Dec 1999.

[63] A. B. Watson. Efficiency of a model human image code. *J. Opt. Soc. Am. A*, 12:2401–2417, 1987.

[64] J I Yellott. Images, statistics and textures: Implications of triple correlation uniqueness for texture statistics and the Julesz conjecture. *J. Opt. Soc. Am. A*, 10(5):777–793, May 1993.

[65] D C Youla. Generalized image restoration by the method of alternating orthogonal projections. *IEEE Trans. Circuits and Systems*, 25:694–702, 1978.

[66] D C Youla and H Webb. Image restoration by the method of convex projections. *IEEE Trans. Med. Imaging*, 1:81–101, 1982.

[67] C Zetzsche, B Wegmann, and E Barth. Nonlinear aspects of primary vision: Entropy reduction beyond decorrelation. In *Int'l Symposium, Society for Information Display*, volume XXIV, pages 933–936, 1993.

[68] S Zhu, Y Wu, and D Mumford. Filters, random fields and maximum entropy (FRAME) – towards the unified theory for texture modeling. In *IEEE Conf Computer Vision and Pattern Recognition*, June 1996.

[69] S C Zhu, X Liu, and Y N Wu. Exploring the Julesz ensemble by efficient Markov chain monte carlo. In *IEEE Workshop on Statistical and Computational Theories of Vision*, Fort Collins, CO, 1999. Available at http://www.cis.ohio-state.edu/~szhu/SCTV99.html.

[70] S C Zhu, Y N Wu, and D Mumford. Minimax entropy principle and its application to texture modeling. In *Neural Computation*, volume 9, pages 1627–1660, 1997.