. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1999 (in press).

Exploring Texture Ensembles by Efficient Markov Chain Monte Carlo

– A conclusion to the Julesz quest?

Song Chun Zhu[1], Xiu Wen Liu[1], Ying Nian Wu[2]

Abstract

This article presents a mathematical definition of texture – the *Julesz ensemble* $\Omega(\mathbf{h})$, which is the maximum set of images that share identical statistics $\mathbf{h}$ defined on the infinite lattice Z^2. Then texture modeling is posed as an inverse problem: given a set of images sampled from an unknown Julesz ensemble $\Omega(\mathbf{h}_*)$, we search for the sufficient and necessary statistics $\mathbf{h}_*$ which define the ensemble. An Julesz ensemble $\Omega(\mathbf{h})$ has an associated probability distribution $q(\mathbf{I};\mathbf{h})$ which is uniform over the images in the ensemble and has zero probability outside. In a companion paper[30], $q(\mathbf{I};\mathbf{h})$ is proven to be the *limit distribution* of the Gibbs model (FRAME model or Markov random field) derived by the minimax entropy principle[33] as the image lattice $\Lambda \to Z^2$. This conclusion has established the intrinsic link between the scientific definition of texture on Z^2 and the mathematical models of texture on finite lattices, and it generates two interesting impacts in computer vision. 1). The engineering practice of synthesizing textures by matching statistics has been put on a firm mathematical foundation. 2). We are released from the burden of learning the expensive FRAME model in feature pursuit, model selection and texture synthesis. In this paper, an algorithm is proposed for sampling Julesz ensembles by efficient Markov chain Monte Carlo, which generates random texture images by moving along filter coefficients and thus extends the traditional single site Gibbs sampler. This paper also compares four popular statistical measures in the literature, namely, moments, rectified functions, marginal histograms and joint histograms of linear filter responses in terms of their descriptive abilities. Our experiments suggest that a small number of bins in marginal histograms are sufficient for capturing a variety of texture patterns, and the full joint histograms appear to be an over-fit. We illustrate our theory and algorithm by successfully synthesizing a number of natural textures.

1. Department of Computer and Information Sciences, The Ohio State University, Columbus, OH 43210.

2. Department of Statistics, University of California, Los Angeles, CA 90095.

1 Introduction and motivations

In his seminar paper in 1962[19], Julesz initiated researches on texture by asking the following fundamental question:

> *What features and statistics are characteristic of a texture pattern, so that texture pairs that share the same features and statistics cannot be told apart by pre-attentive human visual perception?*

Julesz's quest has been the scientific theme in texture modeling and perception in the last three decades, and his question raised two major challenges. The first is in psychology and neurobiology: what *features and statistics* are the *basic elements* in human texture perception? The second lies in mathematics and statistics: given a set of consistent statistics, how do we generate random texture images with the identical statistics? In mathematical language, we should be able to explore the *ensemble of texture images* that have exactly the same statistics. The first question cannot be answered without solving the second one. As Julesz conjectured[21], the ideal theory of texture should be somewhat similar to the theory of trichromacy, which states that any visible color is a linear combination of three basic colors: red, green, and blue. In texture theory, this corresponds to the search for (i) the basic texture statistics and (ii) a method for mixing the exact amount of statistics in a given recipe.

The search for features and statistics went a long way beyond the Julesz 2-gon statistics conjecture. Examples include co-occurrence matrices, run-length statistics[29], sizes and orientations of various textons[28], cliques in Markov random fields[8], as well as dozens of other measures. All these features have rather limited expressive powers. A quantum jump occurred in the 1980s when Gabor filters[11], filter pyramids, and wavelets transforms[10] were introduced to image representation. Another advance occurred in the 1990s when simple statistics, e.g. first and second order moments, were replaced by histograms (either marginal or joint)[18, 32, 9], which contain all the higher order moments.

Research on mathematical methods for rendering texture pairs with identical statistics has also been extensive in the literature. The earliest work was task specific. For example, the "4-disk method" by (Caelli et al., 1978) was designed for rendering random texture pairs sharing 2-gon statistics[5]. In computer vision, many systematic methods have been developed. For examples: 1). Gagalowicz and Ma[12] used steepest descent minimization of the sum of squared errors of the intensity histograms and the two-pixel auto-correlation

matrices , 2). Heeger and Bergen[18] used pyramid collapsing for matching marginal histograms of filter responses. 3). Anderson and Langer[2] used steepest descent method minimizing the match errors of rectified functions. 4). De Bonet and Viola[9] matched full joint histogram using Markov trees. 5). Portilla and Simoncelli[27] matched various correlations through repeated projections onto statistics constraint surfaces.

Despite the recent successes of these methods in texture synthesis and their computational convenience, the following three fundamental questions remain unanswered.

1. What is the mathematical definition of texture adopted in the above methods? Is it technically sound to synthesize textures by minimizing statistics errors, without explicit statistical modeling?

2. How are the above texture synthesis methods related to the rigorous mathematical models of textures, for example, Markov random field models (Cross and Jain, 1983) [8] and minimax entropy models (Zhu, Wu and Mumford, 1997)[33]?

3. In practice, the above methods for synthesizing textures do not guarantee a close match of statistics, nor do they intend to **sample** the ensemble of images with identical statistics. Is there a general way of designing algorithms for efficient statistics matching?

This paper answers problem 1 and 3, and briefly review an answer to question 2. An detailed study of question 2 is referred to a companion paper[30].

First, we defining the Julesz ensemble. Given a set of statistics $\mathbf{h}$ extracted from a set of observed images of a texture pattern, such as histograms of filter responses, the Julesz ensemble is defined as the set of all images that share the same statistics as the observed. An Julesz ensemble, denoted by $\Omega(\mathbf{h})$, has an associated probability distribution $q(\mathbf{I};\mathbf{h})$ which is uniform over the images in the ensemble and has zero probability outside. The Julesz ensemble leads to a mathematical definition of texture on the infinite lattice Z^2. In a companion paper[30], Wu and Zhu have proven that the Julesz ensemble (or the uniform distribution $q(\mathbf{I};\mathbf{h})$ is equivalent to the Gibbs ensemble, and the latter is the *limit* of the minimax entropy (FRAME) model $p(\mathbf{I};\boldsymbol{\beta})$ as the image lattice goes to infinity. The ensemble equivalence reveals two significant facts in texture modeling and synthesis.

1. On large image lattices, we can draw images from the Julesz ensemble $\Omega(\mathbf{h})$ without learning the expensive FRAME model. These sampled images are typical of the

corresponding Gibbs model $p(\mathbf{I};\boldsymbol{\beta})$, and can be used effectively for texture synthesis, feature pursuit, and model selection[33].

2. For a large enough (say 256×256 pixels) image sampled from the Julesz ensemble, any local image patch of the image given its environment follows the Gibbs distribution (or FRAME) derived by the minimax entropy principle. Thus, FRAME is an inevitable model for textures on small lattices.

Second, this paper proposes an effective algorithm for sampling the Julesz ensemble by Markov chain Monte Carlo (MCMC). In the traditional single-site Gibbs sampler[13, 33], the MCMC transition matrix is designed in the following way. At each step, a pixel is chosen at random or in a fixed scan order, and the intensity value of the pixel is updated according to its conditional distribution given the intensities at neighboring pixels. Since the filters used in the texture models are often very large (e.g., 32×32 pixels), flipping one pixel at a time has very little effects on the filter responses. As a result, the single site Gibbs sampler is very inefficient, and the formation and change of local texture features may take a long time.

We approach the efficiency problem by designing conditional moves of filter coefficients. For a linear filter, the window function [1] spans one dimension in the image space. Thus for a set of K filters, we have K axes at each pixel, and these axes do not have to be orthogonal to each other,. We propose random moves along these axes, and thus the proposed moves update large patches of an image so that local features can be formed and changed effectively.

Third, we compare four popular statistical measures in the literature: moments, rectified functions, marginal histograms and joint histograms of Gabor filter responses. Our experiments show that moments and rectified functions are not sufficient for texture modeling. We shall also pursue the minimum set of statistics that can describe texture patterns, and our experiments demonstrate that a small number of bins in the marginal histograms are sufficient in capturing a variety texture patterns, so the full joint histogram appears to be an over-fit. We demonstrate our theory and algorithm by synthesizing many natural textures.

[1] This is called impulse response in Engineering, and it is called the receptive field in Neurosciences.

The paper is organized as follows. We start with a discussion of features and statistics in Section 2 which is followed by a mathematical study of texture modeling in Section 3. Section 4 shows a group of experiments on texture synthesis using the Gibbs sampler. Section 5 describes the feature pursuit experiments in selecting the histogram bins and compares a variety of statistics. Section 6 presents a generalized Gibbs sampler for efficient MCMC sampling. Finally, we conclude the paper with a discussion in Section 7.

2 Image features and statistics

In a first step to pursue the "trichromacy" theory for texture, we review some important image features and statistics that have been used in texture modeling.

Let $\mathbf{I}$ be an image defined on a finite lattice $\Lambda \subset \mathrm{Z}^2$, then for each pixel $v = (x, y) \in \Lambda$, the intensity value at v is denoted by $\mathbf{I}(v) \in S$, with S being a finite interval on the real line or a finite set of quantized grey levels. We denote by $\Omega_\Lambda = S^{|\Lambda|}$ the space of all images on Λ.

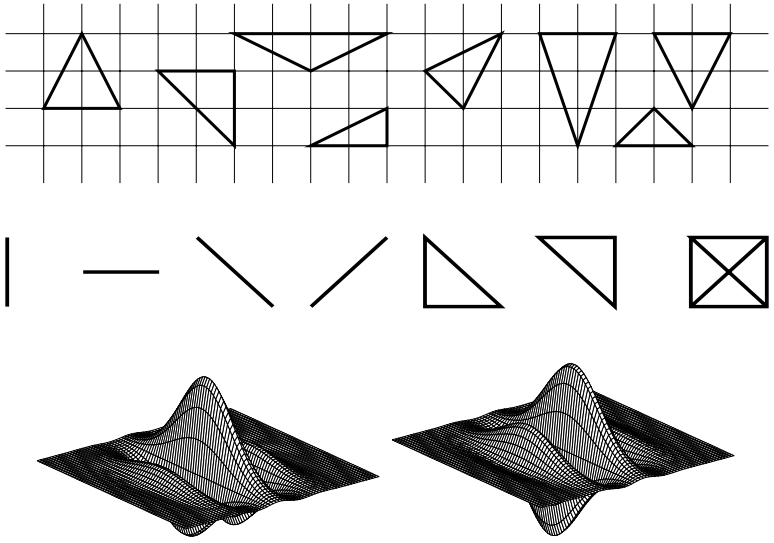


Figure 1 Choices of image features. Top row: various 3-polygons on the image lattice. Middle row: cliques in Markov random fields. Bottom row: Gabor filters with cosine (left) and sine (right) components.

In modeling homogeneous texture images, we start with exploring a finite set of statistics of some local image features. Figure 1 shows the three major categories of image

features studied in the literature.

The first category includes the k-gons proposed by Julesz. A k-gon is a polygon of k vertices indexed by $\alpha = (u_1, u_2, ..., u_k)$, where $\mu_i = (\Delta x_i, \Delta y_i)$ is the displacement of the i-th vertex if we put the center of the k-gon at the origin. Δx_i and Δy_i must be integers. If we move this k-gon on the lattice, under some boundary conditions we collect a set of k-tuples

$$\{ \ (\mathbf{I}(v+u_1), \mathbf{I}(v+u_2), ..., \mathbf{I}(v+u_k)) \ : \ v \in \Lambda \ \}.$$

The k-gon statistic is the k-dimensional joint intensity histogram of these k-tuples, and it is also called the *co-occurence matrix*. To be more specific, if we quantize the intensity value into $1, 2, ..., g$, then the k-gon statistic of $\mathbf{I}$ is expressed as

$$\mathbf{h}^{(\alpha)}(b_1, b_2, ..., b_k; \mathbf{I}) = \frac{1}{|\Lambda|} \sum_{v \in \Lambda} \delta(b_1 - \mathbf{I}(v+u_1)) \cdots \delta(b_k - \mathbf{I}(v+u_k)), \quad \forall (b_1, b_2, ..., b_k) \in \{1, 2, ..., g\}^k.$$

$\delta()$ is the Dirac delta function with unit mass at zero and zero everywhere else. We assume that boundary conditions are properly handled (e.g., periodic boundary condition). Figure 1 displays a set of triangles in the top row. One co-occurence matrix is computed for each type of polygon. The k-gon statistics suffer from the curse of dimensionality for even a small k. For instance, for k as small as 4, and $g = 10$, the dimensionality of a 4-gon statistic is comparable to the size of the image. Summarizing the image into k-gon statistics can hardly achieve any data reduction.

The second type of features are the cliques in Markov random fields (MRF), as shown in the middle row of figure 1. Given a neighborhood system on the lattice Λ, a clique is a set of pixels that are neighbors of each other, so a clique is a special type of k-gon. Let $\alpha = (u_1, u_2, ..., u_{k_\alpha})$ be the index for different types of cliques under a neighborhood system. According to the Hammersley-Clifford theorem[3], a Markov random field model has the Gibbs form

$$p(\mathbf{I}) = \frac{1}{Z} \exp\{- \sum_{\alpha} \sum_{v \in \Lambda} U_\alpha(\mathbf{I}(v+u_1), \mathbf{I}(v+u_2), ..., \mathbf{I}(v+u_{k_\alpha}))\}$$

where Z is the normalization constant or the partition function and U_α are potential functions of k_α variables. The above Gibbs distribution can be derived from the maximum entropy principle under the constraints that $p(\mathbf{I})$ reproduces, on average, the co-occurence matrices $\mathbf{h}^{(\alpha)}(b_1, ..., b_{k_\alpha}; \mathbf{I}), \forall \alpha$. *Therefore, the Gibbs model integrates all the co-occurence matrices for the cliques into a single probability distribution.* Unfortunately this general

MRF model also suffers from the curse of dimensionality even for small cliques. The existing MRF texture models are much simplified so as to reduce the dimensionality of potential functions, such as in auto-binomial models[8], Gaussian MRF models[6] and ϕ-models[14].

The co-occurrence matrices (or joint intensity histograms) on the polygons and cliques have been proven inefficient in describing real world images and irrelevant to biologic vision systems. In the late 1980s, it was realized that real world imagery is better represented on the spatial/frequency bases, such as Gabor filters[11], wavelet transforms[10], and filter pyramids. These filters are often called image *features*. Given a set of filters $\{F^{(\alpha)}, \alpha = 1, 2, ..., K\}$, a sub-band image $\mathbf{I}^{(\alpha)} = F^{(\alpha)} * \mathbf{I}$ is computed for each filter $F^{(\alpha)}$.

Thus the third method in texture analysis extracts statistics on the sub-band images or pyramid instead of the intensity image. From a dimension reduction perspective, the filters characterize local texture features, as a result, very simple statistics of the sub-band images can capture information that otherwise would require k-gon or clique statistics of very high dimensions.

While Gabor filters are well grounded in biologic vision[7], very little is known about how visual cortices pool statistics across images. Figure 2 displays four popular choices of statistics in the literature.

1. Moments of a single filter response, e.g. mean and variance of $\mathbf{I}^{(\alpha)}$ in figure 2a,
$$\mathbf{h}^{(\alpha,1)}(\mathbf{I}) = \frac{1}{|\Lambda|}\sum_{v\in\Lambda}\mathbf{I}^{(\alpha)}(v),$$
$$\mathbf{h}^{(\alpha,2)}(\mathbf{I}) = \frac{1}{|\Lambda|}\sum_{v\in\Lambda}(\mathbf{I}^{(\alpha)}(v) - \mathbf{h}^{(\alpha,1)})^2.$$

2. Rectified functions that resemble the responses of "on/off" cells[2]:
$$\mathbf{h}^{(\alpha,+)}(\mathbf{I}) = \frac{1}{|\Lambda|}\sum_{v\in\Lambda}R^{+}(\mathbf{I}^{(\alpha)}(v)),$$
$$\mathbf{h}^{(\alpha,-)}(\mathbf{I}) = \frac{1}{|\Lambda|}\sum_{v\in\Lambda}R^{-}(\mathbf{I}^{(\alpha)}(v)),$$
where the functions $R^{+}(), R^{-}()$ are shown in figure 2b.

3. One bin of the empirical histogram of $\mathbf{I}^{(\alpha)}$,
$$\mathbf{h}^{(\alpha)}(b;\mathbf{I}) = \frac{1}{|\Lambda|}\sum_{v\in\Lambda}\delta(b_i - \mathbf{I}^{(\alpha)}), \quad \forall\alpha.$$

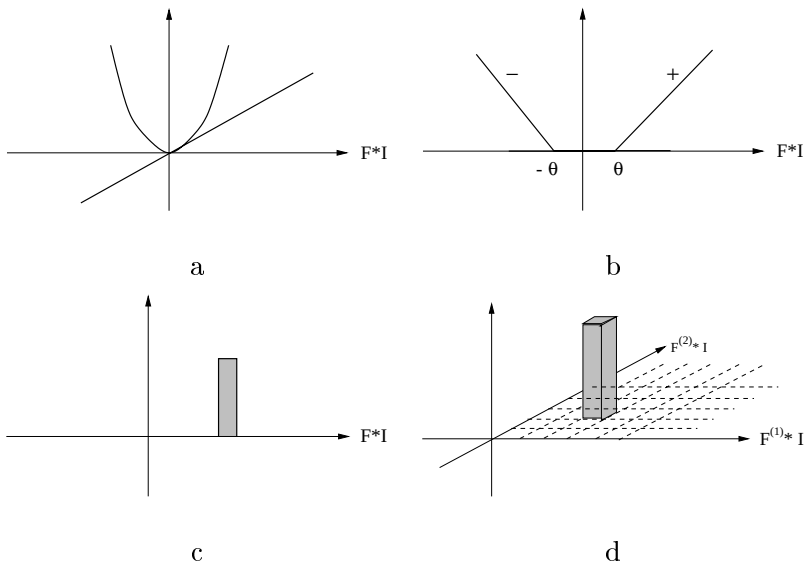


Figure 2 Four choices of statistics in the feature space: a). Moments. b). Rectified functions for the "on/off" cells. c). A bin of the marginal histogram using rectangle window or Gaussian window, d). a bin of joint histogram. This figure is generalized from (Anderson and Langer, 1996).

---

where $\delta()$ is a window function shown in figure 2c.

4. One bin of the full joint histogram,

$$\mathbf{h}(b_1, b_2, ..., b_k; \mathbf{I}) = \frac{1}{|\Lambda|} \sum_{v \in \Lambda} \delta(b_1 - \mathbf{I}^{(1)}(v)) \cdots \delta(b_k - \mathbf{I}^{(k)}(v)), \tag{1}$$

where $(b_1, b_2, ..., b_k)$ is the index for one bin in a k-dimensional histogram in figure 2d.

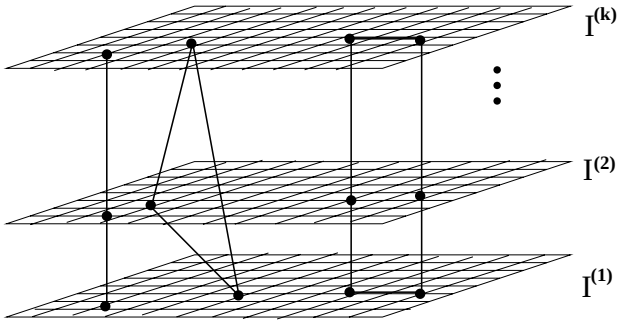


Figure 3 The general forms of statistics are co-occurence matrices for polygons in the image pyramid.

Perhaps the most general statistics are the co-occurence matrices (histograms) for polygons whose vertices are on the image pyramid across multiple layers, as displayed in figure (3). For a k-gon in the image pyramid, a co-occurrence matrix (or joint histogram) can be computed,

$$\mathbf{h}^{(\alpha,k)}(b_1, b_2, ..., b_k; \mathbf{I}) = \frac{1}{|\Lambda|} \sum_{v \in \Lambda} \delta(b_1 - \mathbf{I}^{(l_1)}(v + u_1)) \cdots \delta(b_k - \mathbf{I}^{(l_k)}(v + u_k)). \qquad (2)$$

In the above definition, $\alpha = ((l_1, u_1), (l_2, u_2), ..., (l_k, u_k))$ is the index for the polygons in the pyramid, where l_i, u_i are respectively the indexes for the sub-band and the displacement of the i-th vertex of the polygon. It is easy to see all the traditional co-occurence matrices and marginal and joint histograms are specials case of the statistics in equation (2). For example, it reduces to the traditional co-occurence matrix when the pyramid contains only the intensity image $\mathbf{I}$; it reduces to the full joint histogram[9] in equation (1) when the polygon α is a straight line crossing the pyramid; it captures spatial correlations[27], such as parallalism if the polygon has two straight lines crossing the image as illustrated by the rightmost polygon in figure 3.

Obviously, joint statistics and correlations are useful in aligning image features crossing the spatial and frequency domains, such as edges and some elaborate details of texture elements. The question is how far one should pursue these high order statistics. The complexity of the statistics are limited by both the computational complexity and the statistical efficiency in estimating the model with finite data.

In the literature, Heeger and Bergen made the first attempt to generate textures using marginal histograms[18], and Zhu, Wu, and Mumford studied a new class of Markov random field model that can reproduce the marginal statistics[33].[2] Recently De Bonet and Viola[9], and Simoncelli and Portilla[27] have argued that joint statistics and correlations are needed for synthesizing some texture patterns. The argument is mainly based on the fact that matching the marginal statistics for the filters of the steerable pyramid used by Heeger and Bergen[18] cannot reproduce some texture patterns. This argument is questionable because computationally, the Heeger and Bergen algorithm does not guarantee a close match of statistics, nor is it intended to *sample* from a texture ensemble. In theory, it is provable that marginal statistics are sufficient for reconstructing the full probability distribution of a texture[33]. Of course, this conclusion does not necessarily prevent us

[2] In the Zhu, Wu and Mumford framework, it is straight forward to derive Markov random field models that matches other statistics discussed in this section.

from using joint statistics or the correlations between filter responses. Then there are two key questions that need to be studied: 1). what is the minimum set of statistics for defining a texture pattern? 2). how can we sample textures unbiasly from the set of texture images that share identical statistics? That is, we need to make sure that we are not sampling from a special subset.

3 Julesz ensemble and MCMC sampling of texture

In a second step to answer the Julesz's quest, in this section, we first propose a mathematical definition of texture–the Julesz ensemble, and then we study an algorithm for sampling images from the Julesz ensemble.

3.1 The Julesz ensemble - a mathematical definition of texture

Given a set of K statistics $\mathbf{h} = \{\mathbf{h}^{(\alpha)} \ : \ \alpha = 1, 2, ..., K\}$ which have been normalized with respect to the size of the lattice $|\Lambda|$, an image $\mathbf{I}$ is mapped into a point $\mathbf{h}(\mathbf{I}) = (\mathbf{h}^{(1)}(\mathbf{I}), ..., \mathbf{h}^{(K)}(\mathbf{I}))$ in the *space of statistics*. Let

$$\Omega_\Lambda(\mathbf{h}_o) = \{\mathbf{I} : \ \mathbf{h}(\mathbf{I}) = \mathbf{h}_o\}$$

be the set of images sharing the same statistics $\mathbf{h}_o$. Then the image space Ω_Λ is partitioned into equivalence classes

$$\Omega_\Lambda = \cup_{\mathbf{h}} \Omega_\Lambda(\mathbf{h}).$$

Due to intensity quantization in finite lattices, in practice one needs to relax the constraint on statistics and to define the image set as

$$\Omega_\Lambda(\mathcal{H}) = \{\mathbf{I} : \ \mathbf{h}(\mathbf{I}) \in \mathcal{H}\},$$

where $\mathcal{H}$ is an open set around $\mathbf{h}_o$.

$\Omega_\Lambda(\mathcal{H})$ specifies a uniform distribution on the image space Ω_Λ.

$$q(\mathbf{I}; \mathcal{H}) = \begin{cases} \frac{1}{|\Omega_\Lambda(\mathcal{H})|} & \text{for } \mathbf{I} \in \Omega_\Lambda(\mathcal{H}). \\ 0 & \text{otherwise.} \end{cases}$$

$|\Omega_\Lambda(\mathcal{H})|$ is the volume of the set.

Definition *Given a set of normalized statistics* $\mathbf{h} = \{\mathbf{h}^{(\alpha)} : \alpha = 1, 2, ..., K\}$, *a Julesz ensemble* $\Omega(\mathbf{h})$ *is the limit of* $\Omega_\Lambda(\mathcal{H})$ *as* $\Lambda \to \mathrm{Z}^2$ *and* $\mathcal{H} \to \{\mathbf{h}\}$ *under some boundary conditions.*

A Julesz ensemble $\Omega(\mathbf{h})$ is a mathematical idealization of $\Omega_\Lambda(\mathcal{H})$ for an infinite lattice. As $\Lambda \to \mathrm{Z}^2$, it makes sense to let the normalized statistics $\mathcal{H} \to \{\mathbf{h}\}$. We assume $\Lambda \to \mathrm{Z}^2$ in the sense of van Hove[15], i.e., the ratio between the size of the boundary condition and the size of Λ goes to 0, $\frac{|\partial\Lambda|}{|\Lambda|} \to 0$. In engineering practice, we often conside a lattice big enough if $\frac{|\partial\Lambda|}{|\Lambda|}$ is very small, e.g. $\frac{1}{15}$. Thus with a slight abuse of notation, we consider a large enough image as an infinite image in the rest of the paper.

A Julesz ensemble $\Omega(\mathbf{h})$ defines a texture pattern on Z^2, and it maps textures into the space of feature statistics $\mathbf{h}$. By analogy to color, as an electro-magnetic wave with wavelength $\lambda \in [400, 700]$nm defines an unique visible color, a statistic value $\mathbf{h}$ defines a texture pattern! [3] We shall study the relation of the Julesz ensemble to the mathematical models of texture in the next section.

A mathematical definition of texture could be different from a texture category in human texture perception. The latter has very coarse precision on the statistics $\mathbf{h}$ and is often influenced by experiences. For example, Julesz proposed that texture pairs which are pre-attentively not segmentable belong to the same category. Recently many groups have reported that texture pairs which are not pre-attentively segmentable by naive subjects become segmentable after practice[22]. This phenomenon is similar to color perception.

With the mathematical definition of texture, texture modeling is posed as an inverse problem. Suppose we are given a set of images $\Omega_{\text{obs}} = \{\mathbf{I}_{\text{obs},1}, \mathbf{I}_{\text{obs},2}, ..., \mathbf{I}_{\text{obs},M}\}$ as observed training image, which are sampled from an unknown Julesz ensemble $\Omega(\mathbf{h}_*)$. The objective of texture modeling is to search for the set of *sufficient* and *necessary* statistics $\mathbf{h}_*$.

We first choose a set of K statistics from a dictionary B discussed in section 2, we compute the normalized statistics over the observed images $\mathbf{h}_{\text{obs}} = (\mathbf{h}^{(1)}_{\text{obs}}, ..., \mathbf{h}^{(K)}_{\text{obs}})$, with

$$\mathbf{h}^{(\alpha)}_{\text{obs}} = \frac{1}{M}\sum_{i=1}^{M} \mathbf{h}^{(\alpha)}(\mathbf{I}_{\text{obs},i}) \quad \alpha = 1, 2, ..., K. \tag{3}$$

Then we define an ensemble of texture images using $\mathbf{h}_{\text{obs}}$.

$$\Omega_{K,\epsilon} = \{\mathbf{I} : D(\mathbf{h}^{(\alpha)}(\mathbf{I}), \mathbf{h}^{(\alpha)}_{\text{obs}}) \le \epsilon, \quad \forall\alpha\}, \tag{4}$$

[3] We name this ensemble after Julesz to remember his pioneering work on texture. This does not necessarily mean that Julesz defined texture pattern with this mathematical formulation.

where D is some distance, such as L_1 distance for histograms. Then $\Omega_{K,\epsilon} \to \Omega_K$ is a mathematical idealization as $\Lambda \to \mathrm{Z}^2, \epsilon \to 0$. The ensemble Ω_K implies a uniform probability distribution $q(\mathbf{I}; \mathbf{h})$ over Ω_K, whose entropy is $\log |\Omega_K|$.

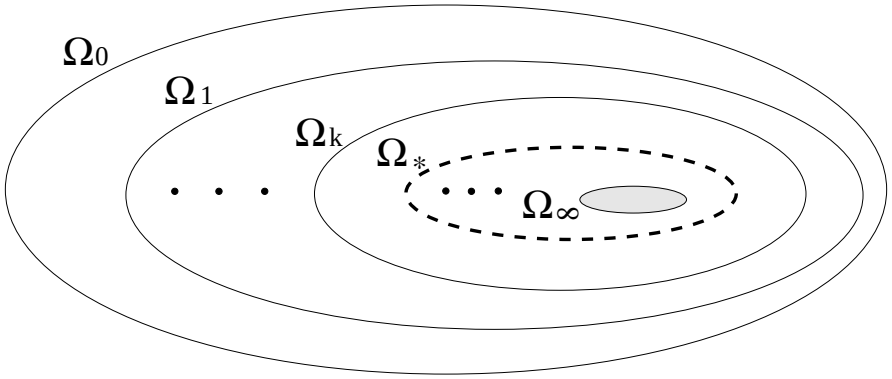


Figure 4 The volume (or entropy) of Julesz ensemble decreases monotonically with more statistical constraints added.

In search for the underlying Julesz ensemble Ω_*, one can adopt a pursuit strategy used by (Zhu, Wu, and Mumford, 1997)[33]. When $k = 0$, we have $\Omega_0 = \Omega_\Lambda$. Suppose at step k, a statistics $\mathbf{h}$ is chosen, then at step $k+1$ a statistic $\mathbf{h}^{(k+1)}$ is added to have $\mathbf{h}_+ = (\mathbf{h}, \mathbf{h}^{(k+1)})$. $\mathbf{h}^{(k+1)}$ selected for the largest entropy decrease among all statistics in the dictionary B,

$$\mathbf{h}^{(k+1)} = \arg\max_{\beta \in B} \text{entropy}(q(\mathbf{I}; \mathbf{h})) - \text{entropy}(q(\mathbf{I}; \mathbf{h}_+) = \log |\Omega_k| - \log |\Omega_{k+1}|. \quad (5)$$

The decrease of entropy is called the *information gain* of $\mathbf{h}^{(k+1)}$.

As shown in figure 4, with more and more statistics added, the entropy or volume of the Julesz ensemble decreases monotonically

$$\Omega_\Lambda = \Omega_0 \supseteq \Omega_1 \supseteq \cdots \supseteq \Omega_k \supseteq \cdots$$

Obviously introducing too many statistics will lead to an "over-fit". In the limit of $k \to \infty$, Ω_∞ only includes the observed images in Ω_{obs} and their translated versions.

With the observed finite images, the choice of statistics $\mathbf{h}$ and the Julesz ensemble $\Omega(\mathbf{h})$ is an issue of model complexity that has been extensively studied in the statistics literature. In the minimax entropy model[33, 32], an AIC criterion[1] is adopted for model selection. The intuitive idea of AIC is simple. With finite images, we should measure the

fluctuation of the new statistics $\mathbf{h}^{(k+1)}$ over the training images in Ω_{obs}. Thus when a new statistic is added, it brings information as well as estimation error. The feature pursuit process should stop when the estimation error brought by $\mathbf{h}^{(k+1)}$ is over its information gain.

3.2 The Gibbs ensemble and ensemble equivalence

To make this paper self-contained, this section briefly discusses the Gibbs ensemble and the equivalence between the Julesz ensemble and Gibbs ensemble. A detailed study is referred to a companion paper[30].

Given a set of observed images Ω_{obs} and the statistics $\mathbf{h}_{\text{obs}}$, the other theory for texture analysis pursues probabilistic texture models, in particular the Gibbs distributions or MRF models.

One general class of MRF model is the minimax entropy model FRAME studied by Zhu, Wu, and Mumford in 1997[32, 33]. The FRAME model derived from the maximum entropy principle has the Gibbs form

$$p(\mathbf{I};\boldsymbol{\beta}) = \frac{1}{Z(\boldsymbol{\beta})}\exp\{-\sum_{\alpha=1}^{K} < \beta^{(\alpha)}, \mathbf{h}^{(\alpha)}(\mathbf{I}) >\} = \frac{1}{Z(\boldsymbol{\beta})}\exp\{< \boldsymbol{\beta}, \mathbf{h}(\mathbf{I}) >\}. \tag{6}$$

The parameters $\boldsymbol{\beta} = (\beta^{(1)}, \beta^{(2)}, ..., \beta^{(K)})$ are Lagrange multipliers. The values of $\boldsymbol{\beta}$ are determined so that $p(\mathbf{I};\boldsymbol{\beta})$ reproduces the observed statistics,

$$E_{p(\mathbf{I};\boldsymbol{\beta})}[\mathbf{h}^{(\alpha)}(\mathbf{I})] = \mathbf{h}^{(\alpha)}_{\text{obs}} \quad \alpha = 1, 2, ..., K. \tag{7}$$

The selection of statistics is guided by a minimum entropy principle.

As the image lattice becomes large enough so that large sampling effects take place, and the fluctuations of the normalized statistics diminish at the speed of $\frac{1}{|\Lambda|}$. Thus as $\Lambda \to \mathrm{Z}^2$, the FRAME model converges to a *limiting random field* which is a uniform distribution over a *typical set* of $p(\mathbf{I};\boldsymbol{\beta})$ which we called the *Gibbs ensemble* in the absence of phase transition. [4]

[4] In the computation of a feature statistic $\mathbf{h}(\mathbf{I})$, we need to define boundary conditions so that the filter responses in Λ are well defined. In case of phase transition, the limit of a Gibbs distribution is not unique, and it depends on the boundary conditions. However, the equivalence between Julesz ensemble and Gibbs ensemble holds even with phase transition. The study of phase transition is beyond the scope of this paper.

In a companion paper[30], we have proven that the Gibbs ensemble given by $p(\mathbf{I};\boldsymbol{\beta})$ is equivalent to the Julesz ensemble specified by $q(\mathbf{I};\mathbf{h}_{\text{obs}})$. The relationship between $\boldsymbol{\beta}$ and $\mathbf{h}_{\text{obs}}$ is expressed in equation (7). Intuitively, $q(\mathbf{I};\mathbf{h}_obs)$ is defined by a "hard" constraint, while the Gibbs model $p(\mathbf{I};\boldsymbol{\beta})$ is defined by a "soft" constraint. Both use the observed statistics $\mathbf{h}_{\text{obs}}$, and the typical set of $p(\mathbf{I};\boldsymbol{\beta})$ becomes the Julesz ensemble as the lattice Λ gets big enough.

The ensemble equivalence reveals two significant facts in texture modeling.

1. Given a set of statistics $\mathbf{h}$, we can synthesize typical texture images by sampling the Julesz ensemble $\Omega(\mathbf{h})$ without learning the expensive FRAME models[33]. The latter needs to learn the parameters $\boldsymbol{\beta}$ in the Gibbs distribution, which is time consuming. Thus feature pursuit, model selection, and texture synthesis can be done effectively with the Julesz ensemble.

2. For any images sampled from the Julesz ensemble, a local image patch of the image given its environment follows the Gibbs distribution (or FRAME) derived by the minimax entropy principle. Therefore, the Gibbs model $p(\mathbf{I};\boldsymbol{\beta})$ provides a parametric form for the conditional distribution of $q(\mathbf{I};\mathbf{h})$ on small image patches. $p(\mathbf{I};\boldsymbol{\beta})$ should be used for tasks, such as texture classification and segmentation.

The learning of Julesz ensembles also uses the minimax entropy principle. First, the definition of $\Omega(\mathbf{h})$ as the *maximum* set of images sharing statistics $\mathbf{h}$ is equivalent to a maximum entropy principle. Second, the pursuit of statistics in equation (5) uses a minimum entropy principle. Therefore a grand unifying picture emerges for texture modeling under the minimax entropy theory.

3.3 Sampling the Julesz ensemble

Sampling the Julesz ensemble is by no means a trivial task! As $|\Omega_K|/|\Omega_\Lambda|$ is exponentially small, the Julesz ensemble has almost zero volume in the image space. Thus reject-sampling methods are inappropriate, and we resort to Markov chain Monte Carlo methods.

First, we define a function

$$G(\mathbf{I}) = \begin{cases} 0, & \text{if } D(\mathbf{h}^{(\alpha)}(\mathbf{I}), \mathbf{h}^{(\alpha)}_{\text{obs}}) \leq \epsilon, \quad \forall \alpha. \\ \sum_{\alpha=1}^{K} D(\mathbf{h}^{(\alpha)}(\mathbf{I}), \mathbf{h}^{(\alpha)}_{\text{obs}}), & \text{otherwise.} \end{cases}$$

Then the distribution

$$q(\mathbf{I}; \mathbf{h}, T) = \frac{1}{Z(T)} \exp\{-G(\mathbf{I})/T\} \tag{8}$$

goes to a Julesz ensemble Ω_K, as the temperature T goes to 0. The $q(\mathbf{I}; \mathbf{h}, T)$ can be sampled by the Gibbs sampler or other MCMC algorithms.

Algorithm I: Sampling the Julesz ensemble

Given texture images $\{\mathbf{I}_{\text{obs},i},\ i = 1, 2, ..., M\}$.
Given K statistics (filters) $\{F^{(1)}, F^{(2)}, ..., F^{(K)}\}$.
Compute $\mathbf{h}_{\text{obs}} = \{\mathbf{h}^{(\alpha)}_{\text{obs}}, \quad \alpha = 1, ..., K\}$.
Initialize a synthesized image $\mathbf{I}$ (e.g. white noise).
$T \leftarrow T_0$.
Repeat
  Randomly pick a location $v \in \Lambda$,
    For $\mathbf{I}(v) \in S$ Do
      Calculate $q(\mathbf{I}(v) \mid \mathbf{I}(-v); \mathbf{h}, T)$.
    Randomly draw one pixel from $q(\mathbf{I}(v) \mid \mathbf{I}(-v); \mathbf{h}, T)$.
  Reduce T after each sweep.
  Record samples when $D(\mathbf{h}^{(\alpha)}(\mathbf{I}), \mathbf{h}^{(\alpha)}_{\text{obs}}) \leq \epsilon$ for $\alpha = 1, 2, ..., K$.
Until enough samples are collected.

In the above algorithm $q(\mathbf{I}(v) \mid \mathbf{I}(-\vec{v}); \mathbf{h}, T)$ is the conditional probability of the pixel value $\mathbf{I}(v)$ with intensities for the rest of the lattice fixed. A sweep is to flip $|\Lambda|$ pixels in a random visiting scheme.

Due to the equivalence between the Julesz ensemble and the Gibbs ensemble[30], the sampled images from $q(\mathbf{I}; \mathbf{h})$ and those from $p(\mathbf{I}; \boldsymbol{\beta})$ share the same statistics in that they produce not only the same statistics in $\mathbf{h}$, but also statistics extracted by any other filters, linear or nonlinear. It is worth emphasizing one key concept which has been misunderstood in some computer vision work: the Julesz ensemble is the set of "typical" images for the Gibbs model $p(\mathbf{I}; \boldsymbol{\beta})$, not the "most probable" images that minimize the Gibbs potential (or energy) in $p(\mathbf{I}; \boldsymbol{\beta})$.

One can use algorithm I for selecting statistics $\mathbf{h}$, as it was done in [33]. That is, one can pursue new statistics by decreasing the entropy as measured in equation (5). An in-depth discussion is referred to [30].

4 Experiment: sampling the Julesz ensemble

In our first set of experiments, we select all of the 56 filters (Gabor filters at various scales and orientations and small Laplacian of Gaussian filters) discussed in [33]. The largest filter window size is 19×19 pixels. We choose $\mathbf{h}$ to be the marginal histograms of these filters and sample the Julesz ensemble using algorithm I. For each texture pattern, it is often true that only a small subset of filters are necessary, but we use a common filter set in this section. We shall discuss statistics pursuit issues in the next section. It is almost impractical to learn a FRAME model integrating all these 56 filters in our previous work[33]; the computation is much easier using the simple model $q(\mathbf{I}; \mathbf{h})$.

We run the algorithm over a broad set of texture images collected from various sources. The results are displayed in figures 5, 6, 7, 8, and 9. The left columns show the observed textures, and the right columns display the synthesized images whose sizes are 256×256 pixels. For these textures, the marginal statistics are matched closely (less than 1% error for each histogram) after about 50 to 100 sweeps, starting with a temperature $T_o = 3$. Since the synthesized images are finite, the matching error ϵ cannot be infinitely small. In general, we set $\epsilon \propto \frac{1}{|\Lambda|}$.

These experiments demonstrate that Gabor filters and marginal histograms are sufficient in capturing a wide variety of homogeneous texture patterns. For example, the cloth pattern in the middle row of figure 6 has very regular structures, which are reproduced fairly well in the synthesized texture image. Also the crosses in figure 8 is synthesized without the special match filter used in [33]. This demonstrates that Gabor filters at various scales align up without using the joint histograms explicitly. The alignment or high order statistics are accounted for through the interactions of the filters.

Figures 8 shows a periodic checker board pattern and a cheetah skin pattern. The synthesized checker board is not strictly periodic, and we believe this is caused by the fast annealing process, i.e. the T in algorithm I decreases too fast, so that the long range effects did not propagate across the image before local patterns form. The synthesized cheetah skin pattern is homogeneous whereas the observed pattern is not.

Our experiments reveal two problems.

The first problem is demonstrated in the two failed examples in figure 9. The observed texture patterns have large structures whose periods are longer than the biggest Gabor filter windows in our filter set. As a result, these periodic patterns are scrambled in the

two synthesized images, while the basic texture features are well preserved.

The second problem is with the effectiveness of the Gibbs sampler. If we scale up the checker board image so that each square of the check board is 15×15 pixels in size, then we have to choose filters with large window sizes. It becomes infeasible to match the marginal statistics closely using the Gibbs sampler in algorithm I, since flipping one pixel at a time is inefficient for such large patterns. This suggests that we should search for more efficient sampling methods that can update large image patches. We believe this problem should occur for other statistics matching methods, such as steepest descent[12, 2]. The inefficiency of the Gibbs sampler is also reflected in its slow mixing rate. After the first image is synthesized, it takes a long time for the algorithm to generate an image which is distinct from the first image. That is, the Markov chain moves very slowly in the Julesz ensemble.

5 Searching for sufficient and efficient statistics

This section compares various sets of statistics discussed in section 2.

First, the full joint histogram defined in equation (1) appears to be an over-fit for most of the natural texture patterns. Thus the MCMC algorithm matching the joint statistics generates texture images from a *subset* of the *true* Julesz ensemble. Our argument is based on two observations. I). Our experiments partially shown in section (4) demonstrate the sufficiency of marginal statistics. II). Suppose that one uses a modest number of filters, e.g. 10 filters, and suppose that each filter response is quantized into 10 bins, then the joint histogram has 10^{10} bins, whereas one often has only a 128×128 texture image as training data. There are far too few pixels to estimate the full joint statistics reliably.

Our conclusion about the sufficiency of marginal statistics should not be overstated. We believe that this conclusion is only valid for general texture appearance. For example, one may construct a texture pattern with hundreds of Asian faces as texture elements, where the detailed alignment of eyes, mouths, and noses are semantically important. Image features extracted by deformable face templates will become prominent. Indeed the statistics extracted by template can be considered as the general statistics extracted by some polygons across the image pyramid in equation (2).

In general, we argue that it is still possible that joint statistics crossing a small number of (say $2 \sim 3$) filters at some bins (see figure 2.d), i.e., not the full joint histogram or full co-

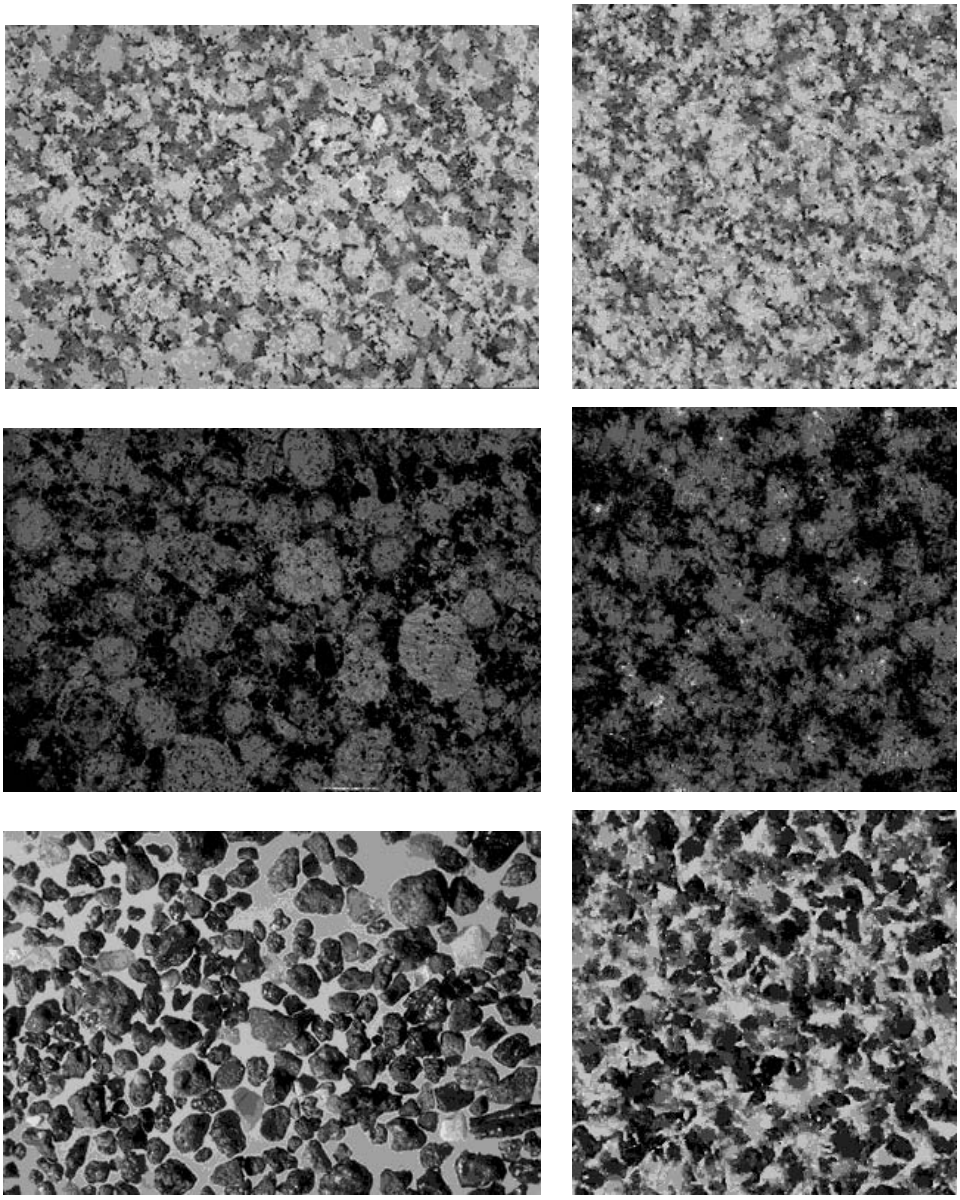


Figure 5 Left column: the observed texture images, right column: the synthesized texture images that share the exact histograms with the observed for 56 filters.

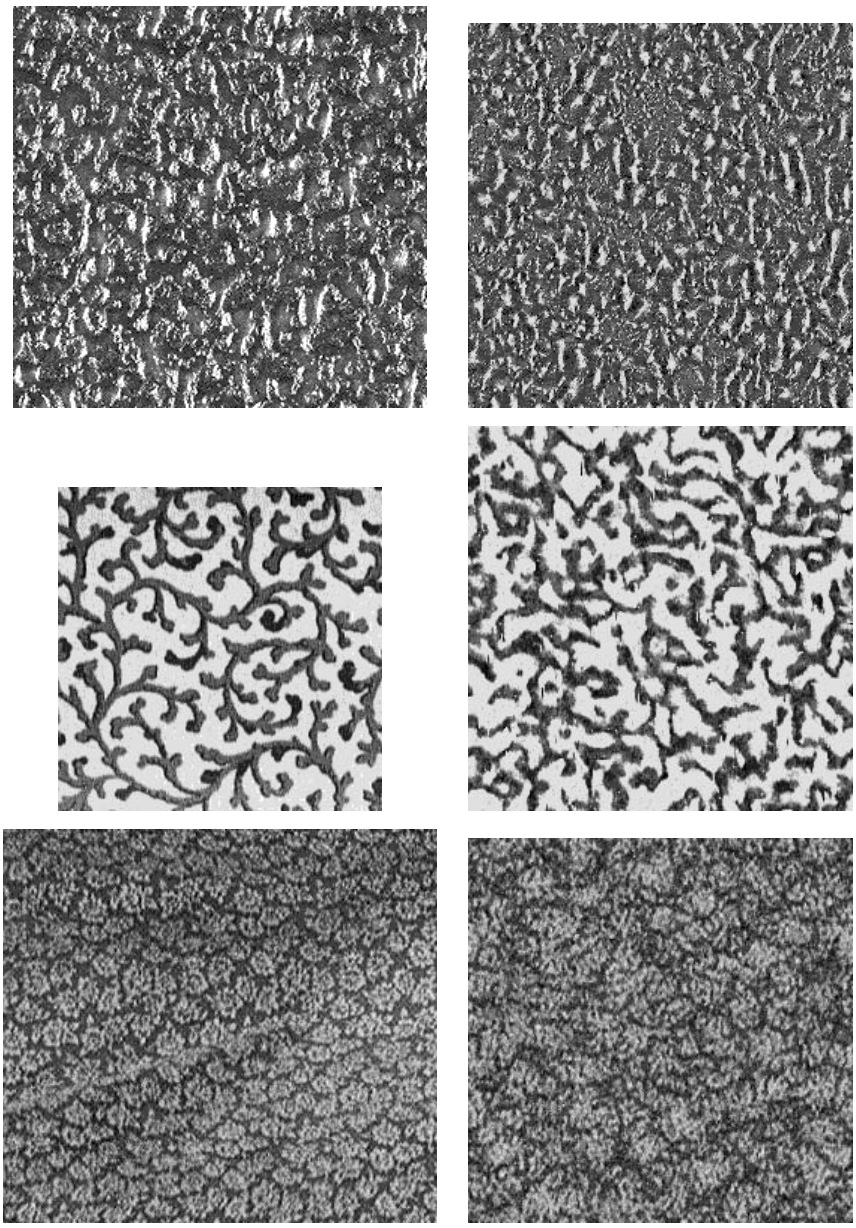


Figure 6 Left column: the observed texture images, right column: the synthesized texture images that share the exact histograms with the observed for 56 filters.

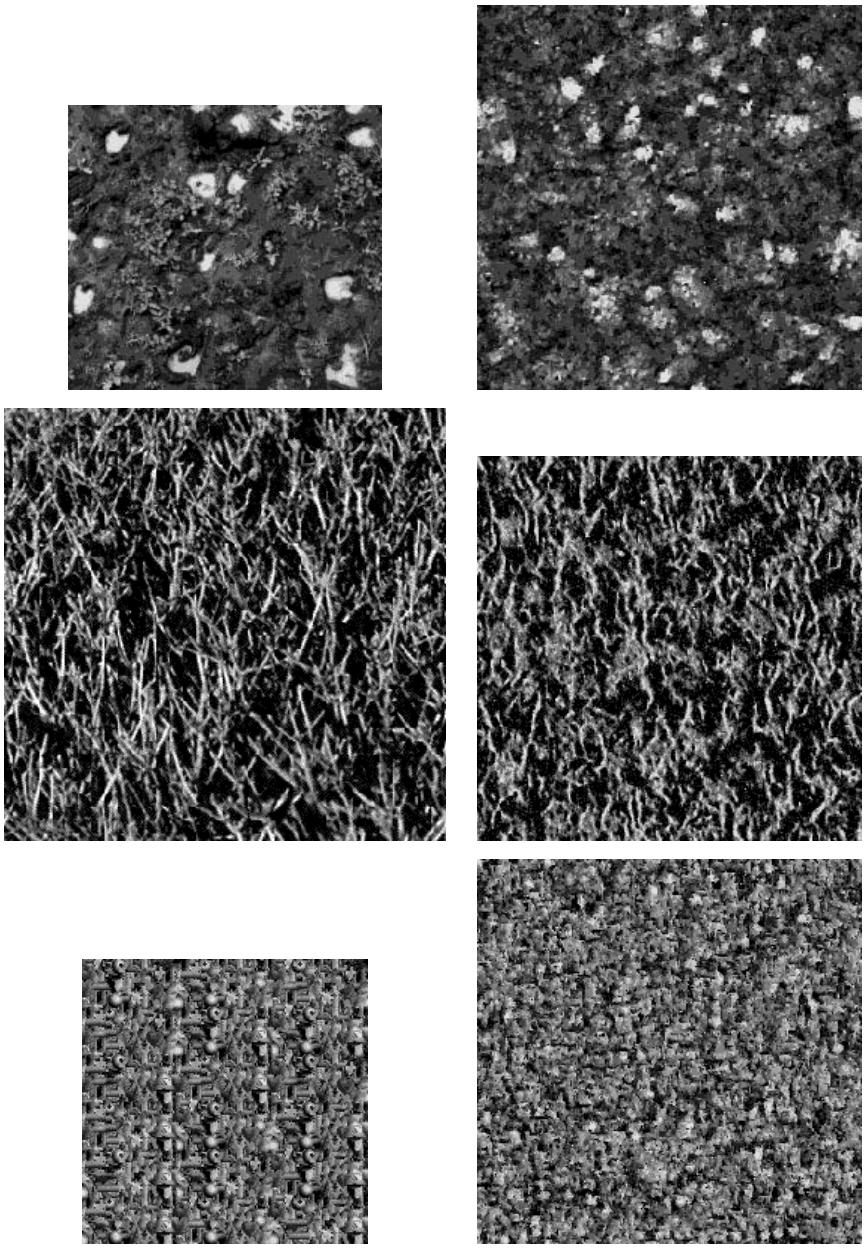


Figure 7 Left column: the observed texture images, right column: the synthesized texture images that share the exact histograms with the observed for 56 filters.

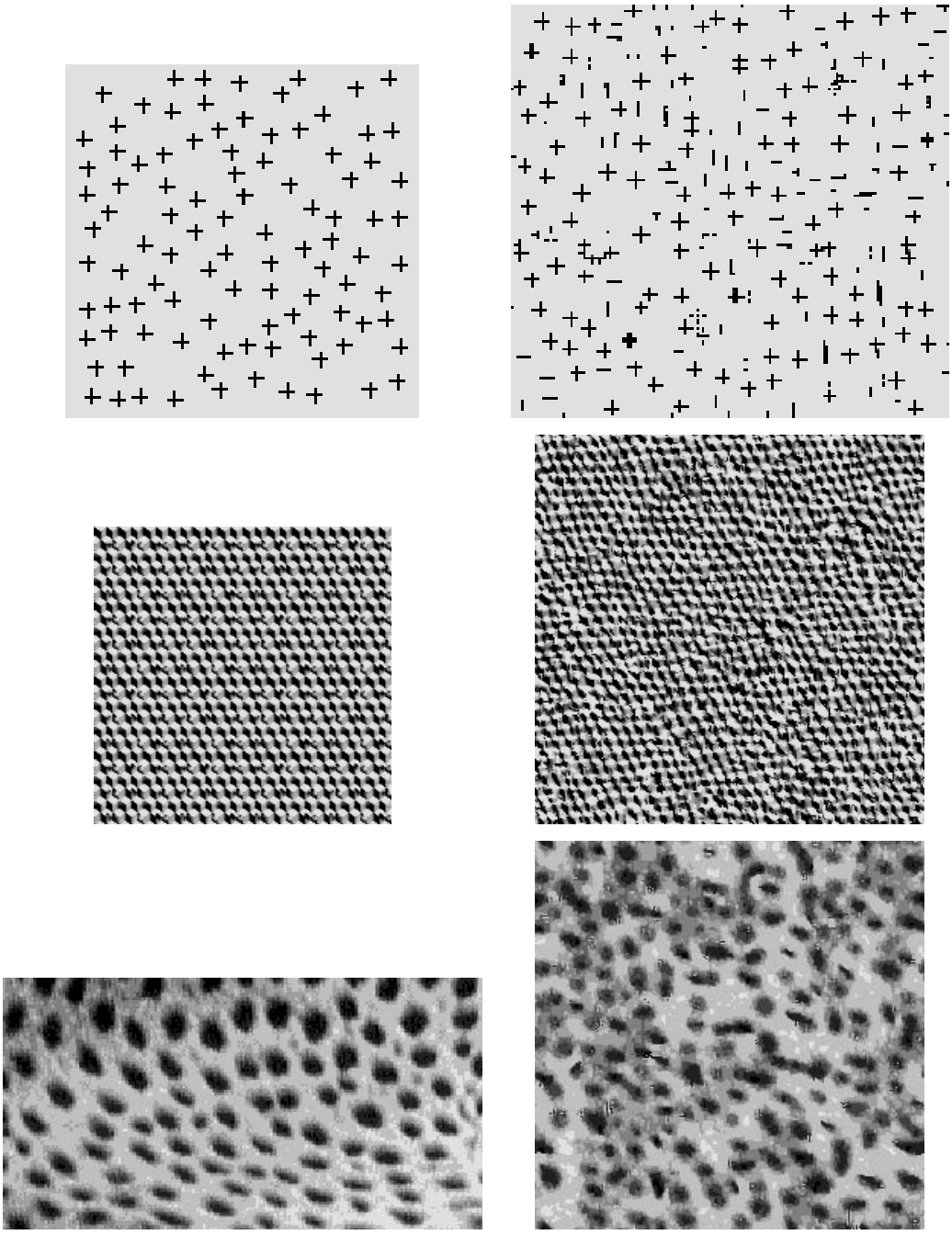


Figure 8 Left column: the observed texture images, right column: the synthesized texture images that share the exact histograms with the observed for 56 filters.

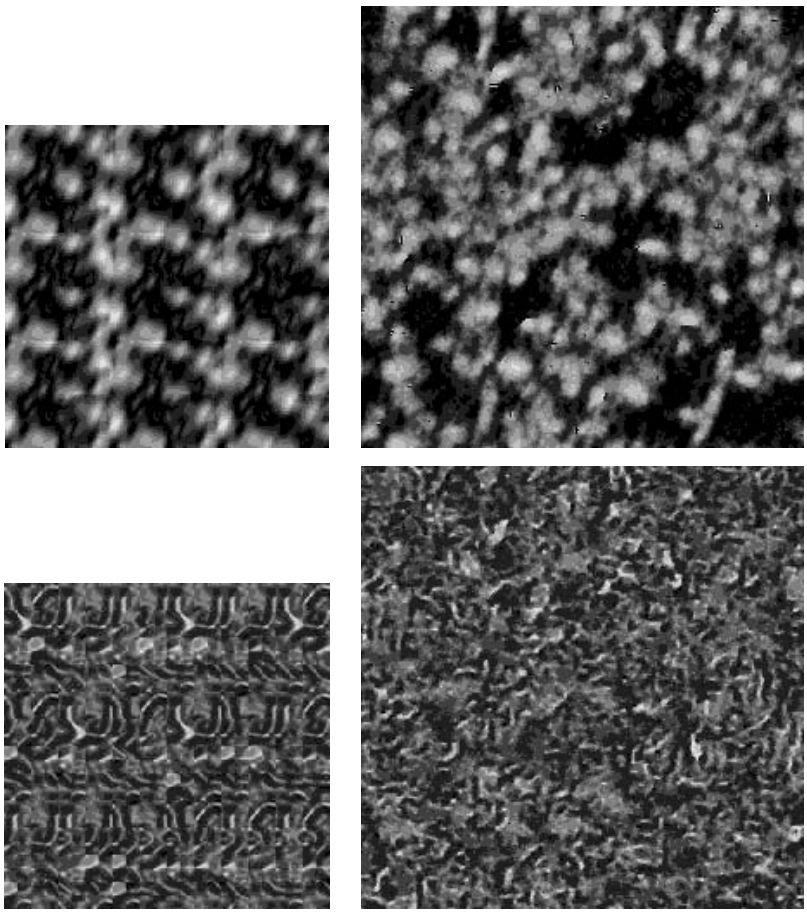
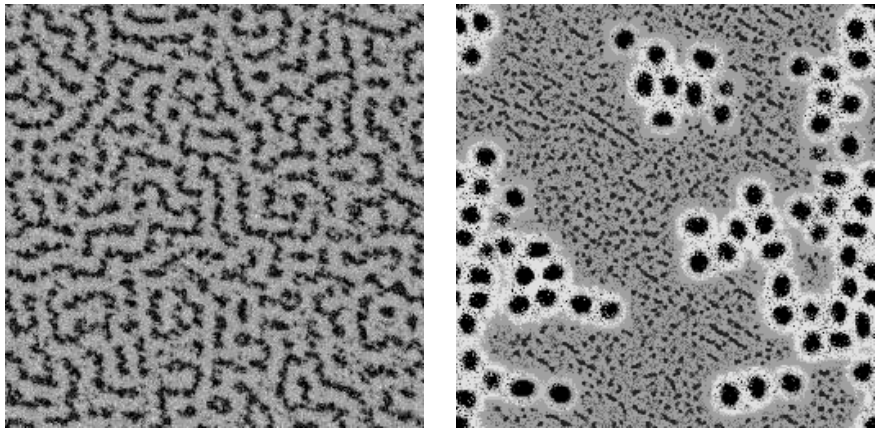


Figure 9 Left column: the observed texture images, right column: the synthesized texture images that share the exact histograms with the observed for 56 filters. The large periodic patterns are missing in the synthesized images due to the lack of large filters.

occurrence matrices, are important for some image features, especially for texton patterns with deliberate details. Unfortunately, combinations of such statistics grow exponentially in the statistics pursuit procedure, we leave this topic for future research.

Second, we have computed the mean and variance for each sub-band images $\mathbf{h}^{(\alpha,i)}, \alpha = 1, 2..., K, 1 = 1, 2$. As one may expect, the image matching the $2K$ statistics have noticeable differences from the observed ones. To save space, we don't show the synthesized images.

Third, we have also calculated the two rectified functions $\mathbf{h}^{(\alpha,+)}, \mathbf{h}^{(\alpha,-)}\alpha = 1, 2..., K$. Although the rectified functions emphasize the tails of the histograms, which often correspond to texture features with large filter responses, the random images from the Julesz ensemble have noticeable differences from the observed.



a b

Figure 10 Two sampled texture pattern from the Julesz ensemble matching the rectified functions of 56 filters. a). $\theta = 0.2$, b). $\theta = 0.8$.

Figure 10 displays an experiment of texture synthesis using the rectified functions. The observed texture is shown in Figure 11a, and we match $\mathbf{h}^{(\alpha,+)}, \mathbf{h}^{(\alpha,-)}$ for all 56 filters used before. We varied the parameter θ in R^-, R^+, so that $100*\theta$ percent of the pixels lies between $[-\theta, \theta]$. We display random examples from the Julesz ensembles with $\theta = 0.2, 0.8$ respectively.

Fourth, we proceeded to study statistics which are simpler than marginal histograms. In particular, we are interested in knowing how many histogram bins are necessary for

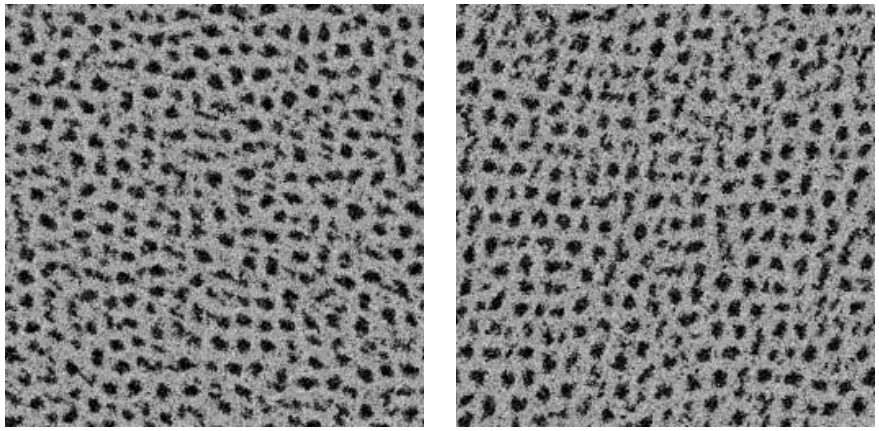


Figure 11 An example of texture synthesis by a number of bins. Left: the observed image. Right: the sampled image using 34 bins

texture synthesis and how the bins are distributed. We adopted the filter pursuit method developed by Zhu, Wu, and Mumford [33]. There are 56 filters total, and each histogram has 11 bins except for the intensity filter that has 8 bins. The algorithm selects one bin from the 605 bins each time according to an information criterion[5]. Suppose m bins have been selected, and $\mathbf{I}$ is a sample of the Julesz ensemble matching the k bins, then the information in the i-th bin of the histogram $\mathbf{h}^{(\alpha)}(b; \mathbf{I})$ is measured by the matching error of this bin in a quadratic form,

$$d(\mathbf{h}^{(\alpha)}(b)) = \frac{(\mathbf{h}^{(\alpha)}(b; \mathbf{I}) - \mathbf{h}^{(\alpha)}_{\text{obs}}(b))^2}{2\sigma^2_{\perp}}. \tag{9}$$

In the above equation, $\sigma^2_{\perp}$ is the variance of $\mathbf{h}^{(\alpha)}(b)$ de-correlated with the previously selected statistics (see the appendix of [33]). Intuitively the bigger the difference is, the more information the bin carries about the texture pattern. The $d(\mathbf{h}^{(\alpha)}(b))$ is a second order Taylor approximation to the entropy decrease $\Delta\text{entropy}(\mathbf{h}^{(\alpha)}(b))$ (see equation (5)) in the FRAME models in Taylor expansion. Because the entropy rate of the Julesz ensemble is the same as that of the FRAME model, $d(\mathbf{h}^{(\alpha)}(b))$ also measures the entropy decrease in the Julesz ensemble. Details about the approximation and the computation of $\sigma^2_{\perp}$, are in [33]. Since $d(\mathbf{h}^{(\alpha)}(b))$ is always positive, this means that as more statistics are used, the model becomes more accurate. This is not true when we only have finite obser-

[5]There is one redundant bin for each histogram.

vations, because we cannot estimate the observed histograms and variances exactly. Thus the information gain $d(\mathbf{h}^{(\alpha)}(b))$ is balanced by a model complexity term that accounts for statistical fluctuations, such as the AIC criterion discussed in [33].

Because the second order approximation is only good locally, in the first few pursuit steps, we used the L_1 distance $d(\mathbf{h}^{(\alpha)}(b)) = ||\mathbf{h}^{(\alpha)}(b) - \mathbf{h}^{(\alpha)}_{\text{obs}}(b)||_1$, and then use the quadratic distance after the sum of the matching errors for all 605 bins is below a certain threshold in terms of L_1 distance.

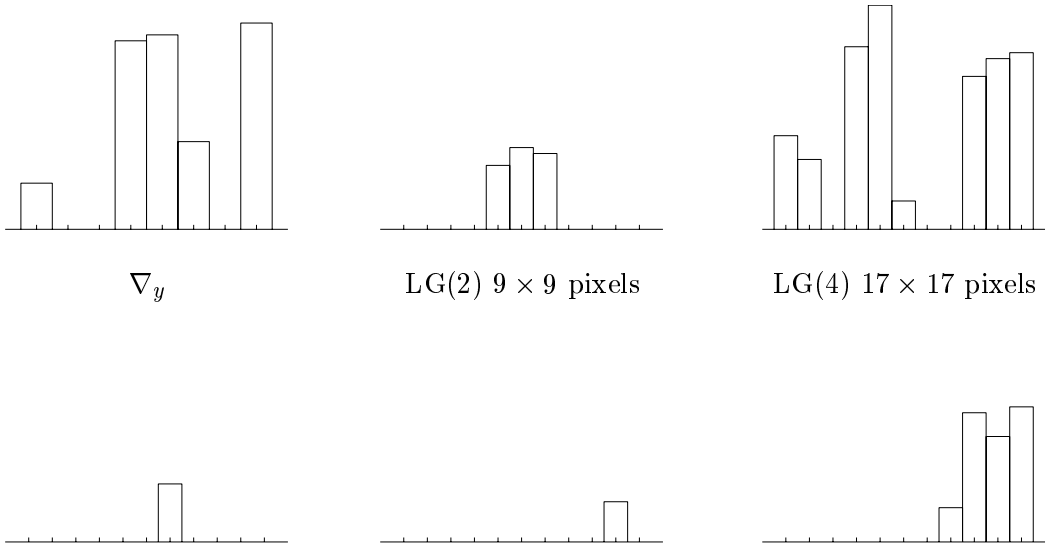


Figure 12 The selected bins for 6 filter histograms, and the height of each bin reflects the information gain.

Figure 11 displays one example of bin selection. The left image is the observed image, and it was originally synthesized by matching the marginal histograms of 8 filters to a real texture pattern. We use this synthesized image as the observation, since it provides a ground truth in statistics selection. The right image is a sample from the Julesz ensemble using 34 bins. The selected bins for six filters are shown in figure 12. The height of each bin reflects the information gains, the higher bins are more important. The other two filters are the intensity filter (all 8 bins are chosen) and the ∇_x filter whose bin selection is very similar to ∇_y.

Our bin pursuit experiments reveal two interesting facts. First, only a small number of histogram bins are necessary for many textures. Second, the selected bins are roughly divided into two categories. One includes the bins near the center of histograms of some small filters, such as the gradient filters and Laplacian of Gaussian filters. These central bins enforce smooth appearances. The other bins are near the two tails of some large filters, which create image patterns. Such observations seems to confirm the design of Gibbs reaction-diffusion equations[34], where the central bins stand for diffusion effects and the tail bins are for reaction effects.

6 Designing efficient MCMC samplers.

This section studies methods for efficient Markov chain Monte Carlo sampling.

6.1 A generalized Gibbs sampler

Improving the efficiency of MCMC has been an important theme in statistics, and many strategies have been proposed in the literature[16, 17]. However, there are no good strategies that are universally applicable, nor is there currently a satisfactory theory that can guide the search for better MCMC algorithms. In fact, the design of good MCMC moves often comes from heuristics of the problem domain.

Let $\mathbf{I}(\cdot, t) \in \Omega_\Lambda$ be an 2D image visited by the Markov chain at time t. For any pixel $v \in \Lambda$, $\mathbf{I}(v, t)$ denotes the intensity value of the pixel at time t. We observe that the Gibbs sampler in algorithm I adopts the following proposed moves in the image space,

$$\mathbf{I}(\cdot, t) \longrightarrow \mathbf{I}(\cdot, t+1) = \mathbf{I}(\cdot, t) + (\Delta - \mathbf{I}(v, t))\boldsymbol{\delta}_v, \quad \forall\, \Delta \in S.$$

The pixel $v = (x, y) \in \Lambda$ is visited with a uniform distribution. $\boldsymbol{\delta}_v \in \Omega_\Lambda$ is an image matrix on Λ which has intensity one at pixel v, and zero everywhere else. It is a Dirac delta function translated to v and represented by a matrix. This algorithm is often called *single-site Gibbs*, as it moves along the dimensions of single pixel intensity.

As $\boldsymbol{\delta}_v$ is an axis in Ω_Λ, it is natural to generalize single-site Gibbs by moving along other axes,

$$\mathbf{I}(\cdot, t) \longrightarrow \mathbf{I}(\cdot, t+1) = \mathbf{I}(\cdot, t) + [(\Delta^{(\alpha)} - \mathbf{I}^{(\alpha)}(v, t))/c^{(\alpha)}]W_v^{(\alpha)}, \quad \forall \Delta^{(\alpha)}. \tag{10}$$

In equation (10), $\mathbf{I}^{(\alpha)}(v, t) = F^{(\alpha)} * \mathbf{I}(v, t)$ is the response of filter $F^{(\alpha)}$ at pixel v, and $\Delta^{(\alpha)}$ is the coefficient. $W_v^{(\alpha)}$ is a window function centered at pixel $v \in \Lambda$, and for convenience

we choose $W^{(\alpha)}$ to be the window of filter $F^{(\alpha)}$ used for extracting texture features. $W_v^{(\alpha)}$ is a 2D matrix by translating the center of the window $W^{(\alpha)}$ to v and setting pixels outside the window to zero. $c^{(\alpha)}$ is the norm of the window function $W^{(\alpha)}$.

In equation (10), it is easy to check that

$$\mathbf{I}^{(\alpha)}(v, t+1) = F^{(\alpha)} * \mathbf{I}(v, t+1) = \Delta^{(\alpha)} - \mathbf{I}^{(\alpha)}(v, t).$$

In the new proposed moves, as long as the intensity filter $\delta()$ is chosen as one of the window function, then one of the basic move is to flip the intensity at a single pixel. Thus the designed Markov chains will be ergodic and aperiodic and have a unique invariant distribution on Ω_Λ. We call such a generalized Gibbs sampler the *window Gibbs sampler*.

In the window Gibbs sampler, one sweep visits each pixel $v \in \Lambda$ once, and at each pixel, a window $W^{(\alpha)}$ is selected from the set of K filters according to a probability $p(\alpha)$. For example, we compute

$$p_1(\alpha) = \frac{||\mathbf{h}^{(\alpha)}(\mathbf{I}) - \mathbf{h}^{(\alpha)}_{\text{obs}}||_1}{\sum_{\beta=1}^{K} ||\mathbf{h}^{(\beta)}(\mathbf{I}) - \mathbf{h}^{(\beta)}_{\text{obs}}||_1}, \quad \alpha = 1, 2, ..., K. \tag{11}$$

$||\cdot||_1$ denotes $L-1$ distance. The filters with large matching errors have more chances to update. Thus the designed Markov chain is inhomogeneous, as the Markov transition probability changes over time. Another choice for $p(\alpha)$ can be

$$p_2(\alpha) = \frac{||\log \mathbf{h}^{(\alpha)}(\mathbf{I}) - \log \mathbf{h}^{(\alpha)}_{\text{obs}}||_1}{\sum_{\beta=1}^{K} ||\log \mathbf{h}^{(\beta)}(\mathbf{I}) - \log \mathbf{h}^{(\beta)}_{\text{obs}}||_1}, \tag{12}$$

which emphasizes matching the tails of the histogram $\mathbf{h}^{(\alpha)}(\mathbf{I})$.

Once α is selected, the value of $\Delta^{(\alpha)}$ in equation (10) is sampled from the conditional distribution

$$\Delta^{(\alpha)} \sim p_3(\Delta^{(\alpha)}) = \frac{q(\mathbf{I}(\cdot, t) + [\Delta^{(\alpha)} - \mathbf{I}^{(\alpha)}(v, t)]W_v^{(\alpha)}/c;\ \mathbf{h})}{\sum_{\Delta} q(\mathbf{I}(\cdot, t) + [\Delta - \mathbf{I}^{(\alpha)}(v, t)]W_v^{(\alpha)}/c;\ \mathbf{h})}.$$

In the denominator the summation is over Δ being the center of each bin in the marginal histogram $\mathbf{h}^{(\alpha)}_{\text{obs}}$.

It is interesting to see that the computational complexity for estimating $p_3(\Delta^{(\alpha)})$ is almost the same as computing $p(\mathbf{I}(v)|\mathbf{I}_{-v})$ in algorithm I. In the appendix, we briefly discuss the implementation details for computing the conditional distributions $p_3(\Delta^{(\alpha)})$.

We briefly represent some results in the following section, before we discuss other ways of designing samplers.

6.2 Experiment on the window Gibbs sampler

This section compares the performance of the window Gibbs sampler against the single-site Gibbs sampler.

Figure 13a displays a cheetah skin texture. This image has been used in figure 8, but the image size in this experiment is as four times large as the one used before. The marginal histograms of eight filters are chosen as the statistics $\mathbf{h}$. Both the single-site Gibbs and the window Gibbs algorithms are simulated with two initial conditions: $\mathbf{I}_u$, a uniform noise image and $\mathbf{I}_c$, a constant white image. We monitor the total matching error at each sweep of the Markov chain,

$$E = \sum_{\alpha=1}^{K} ||\mathbf{h}^{(\alpha)}(\mathbf{I}_{\text{syn}}) - \mathbf{h}^{(\alpha)}_{\text{obs}}||_1.$$

Figure 13 displays the results for the first 100 sweeps. Figure 13b,d are the results of the single site Gibbs starting with $\mathbf{I}_u$ and $\mathbf{I}_c$ respectively. Figure 13c,e are the results of the window Gibbs starting with $\mathbf{I}_u$ and $\mathbf{I}_c$ respectively. The change of E is plotted in figure 14 against the number of sweeps. The dash-dotted curve is for the single site Gibbs starting with $\mathbf{I}_u$, and the dotted curve is that for the single site Gibbs starting with $\mathbf{I}_c$. In both cases, the matching errors remain very high. The two dashed curves are for the window Gibbs starting with $\mathbf{I}_c$ and $\mathbf{I}_u$ respectively. For the latter two curves, the errors drop under 0.08. That means less than 1% error for each histogram on average. The one starting with $\mathbf{I}_u$ drops faster than the one from $\mathbf{I}_c$.

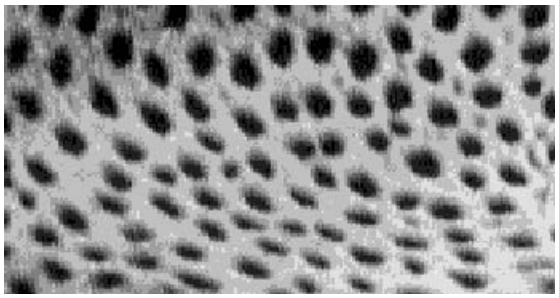
We now use another metric to test the effectiveness of the window Gibbs. We measure the Euclidean distance that the Markov chain travels during one sweep,

$$D(t) = \sqrt{\frac{1}{|\Lambda|}\sum_{v\in\Lambda}(\mathbf{I}(v,t) - \mathbf{I}(v,t-|\Lambda|))^2}.$$

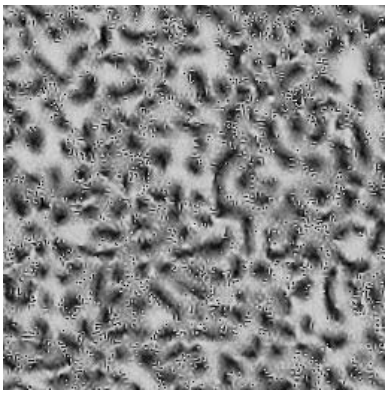
Figure 15 shows $D(t)$ for the single-site Gibbs (dash-dotted and dotted curves) and the window Gibbs (solid and dashed curves) starting with $\mathbf{I}_u, \mathbf{I}_c$ respectively.

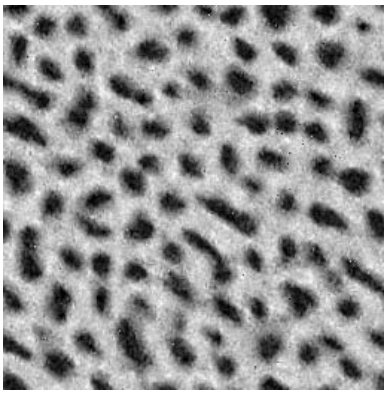
In summary, the window Gibbs algorithm outperforms the single-site Gibbs in at least two aspects. 1). The window Gibbs can match statistics faster, particularly where statistics of long range features are involved. 2). The window Gibbs moves faster after statistics matching. Thus it can render texture images of different appearances.

A rigorous theory for why the window Gibbs is more effective has yet to be found. We only have some intuitive ideas. We believe that the window functions provide better

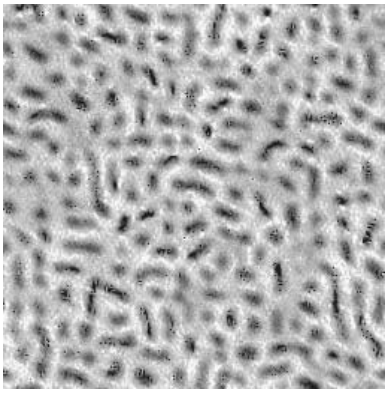


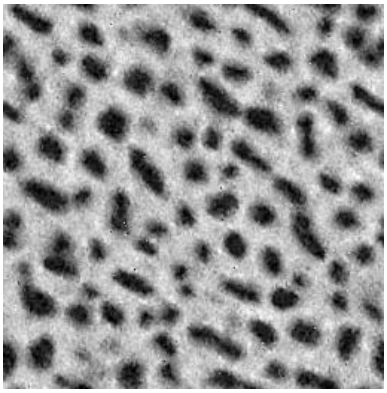
a





b c





d e

Figure 13 a) is the observed cheetah skin texture pattern, the image size of this image is as four times large as the one in figure 8. b) and d) are two sampled images using single site Gibbs sampler starting with a uniform noise image (b) and a constant image (d) respectively. c) and e) are the sampled images using the window Gibbs sampler starting with the same images as in b) and d) respectively.

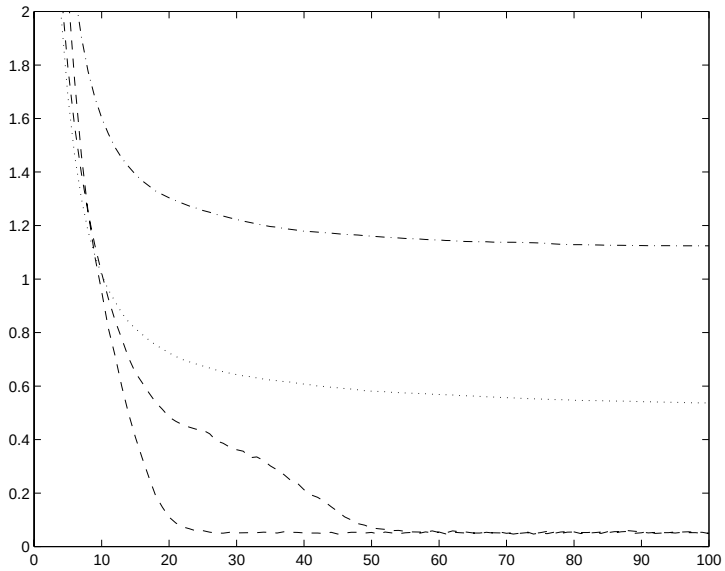


Figure 14 The statistics matching error in L_1 distance summed over 8 filters. The horizontal axis is the number of sweeps in MCMC (see text for explanation).

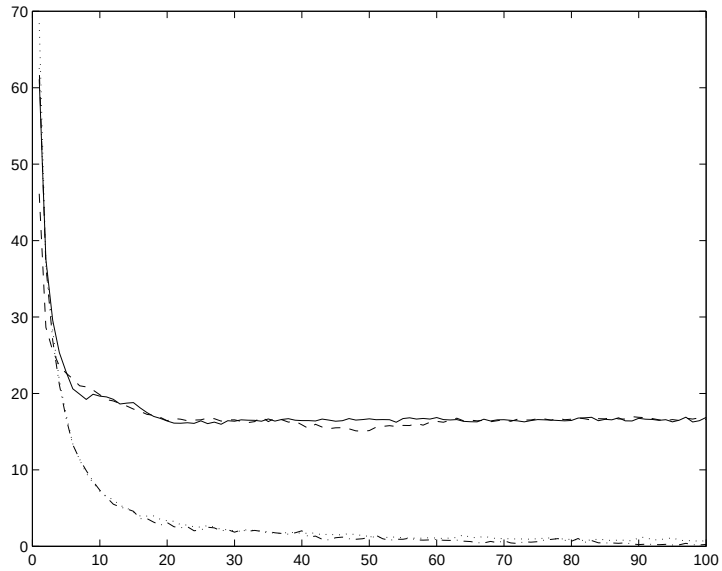


Figure 15 The averaged intensity differences by MCMC in one sweep (see text for explanation).

directions than the coordinate directions employed by the single-site Gibbs for sampling $q(\mathbf{I};\mathbf{h})$ in at least two aspects. One is that it is easier to move towards a mode of $q(\mathbf{I};\mathbf{h})$ along these directions because they lead to substantial changes in local filter responses, so the local spatial features can be formed very quickly. The other aspect is that it is easier to escape from a local mode along these directions, which is important especially in the low temperature situation. Of course, it is likely that these two aspects may require the use of different window functions.

6.3 Discussion on other MCMC methods

Equation (10) provides a way to design new families of Markov chains. We briefly discuss a few issues that are worth exploring in further research.

1. What is the optimal design for the directions of the Gibbs sampler? It was proven that these directions may not have to be linear, and that they can be any transformation groups[24].

2. In the window Gibbs, we adopted a simple heuristic, i.e. the probability $p(\alpha)$ to choose the directions of moves non-uniformly. In general, one can choose other non-uniform probabilities, and statistics other than those used in the Julesz ensemble to drive the Markov chain. For example, one may select the joint statistics (histograms) as the proposal probability $q(\Delta^{(1)}, \Delta^{(2)}, ..., \Delta^{(K)})$ to move the coefficients jointly. Such moves may be capable of creating texture elements (or textons) at a location v because of the alignment of many filters.

7 Conclusion

In this paper, each Julesz ensemble on Z^2 is defined as a texture pattern, and texture modeling is posed as an inverse problem. The remaining difficulty in texture modeling is to search for efficient statistics that characterize texture patterns.

Although section 2 provides a finite set of features and statistics, selecting the most efficient set of statistics within this dictionary is tedious. In particular, if we consider the general joint histogram bins in equation (10), we face an exponential number of combinations. Furthermore, as the filter set is often over complete, many different combinations of statistics can produce similar results.

Many texture patterns contain semantic structures presented in a hierarchical organization, as discussed in Marr's primal sketch[25]. For example, figure 16 shows three texture patterns: the hair of a woman, the twig of tree, and zebra stripes. In these texture images, the basic texture elements and their spatial relationships are clearly perceivable. Indeed, the perception of these elements plays important role for precise texture segmentation. We believe that modeling textures by filters and histograms is only a first order approximation, and further study has to be done to account for the hierarchic organization of the texture elements. This implies that we should look for meaningful features and statistics in a geometric hierarchy outside the current dictionary.

Figure 16 Texture patterns with flows.

The significance of studying image ensembles is far beyond texture modeling and synthesis. Object shapes[35] can also be studied using the concept of ensembles, and so are other image patterns, such as clutter[34]. For example, Zhu and Mumford[34] studied two ensembles: tree clutter and building images, and thus they can seperate the two patterns using Bayes inference. Furthermore, the typical images in a given applications should be studied in order to design efficient algorithms and to analyse the performance of algorithms. Recently Yuille and Coughlan[31] have applied the concept of ensembles to derive the fundamental bounds of road detection.

Acknowledgments

This work was supported by an grant DAAD19-99-1-0012 from the Army Research Office and an NSF grant. We thank the anonymous reviewers for helpful comments that greatly improve the presentation of the paper.

Appendix: computing the conditional probability.

This appendix specifies the implementation details for estimating the conditional probability $p(\Delta^{(\alpha)})$ in the window Gibbs sampler.

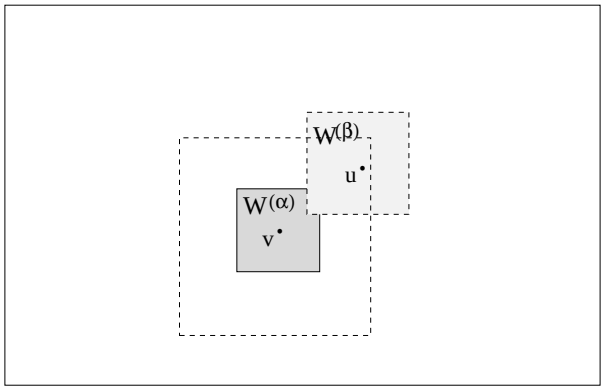


Figure 17 The computation for flipping a window function at v. The response of filter $F^{(\beta)}$ at u needs to be updated if the two windows $W^{(\alpha)}$ at v and $W^{(\beta)}$ at u overlaps. The big dashed rectangle covers the pixels u whose filter responses should be updated.

Suppose that a pixel $v \in \Lambda$ and a filter α are selected at a certain step of the window Gibbs algorithm. The algorithm proposes a move

$$\mathbf{I}(\cdot, t) \longrightarrow \mathbf{I}(\cdot, t+1) = \mathbf{I}(\cdot, t) + [(\Delta^{(\alpha)} - \mathbf{I}^{(\alpha)}(v, t))/c^{(\alpha)}]W_v^{(\alpha)}, \quad \Delta^{(\alpha)} \in S. \tag{13}$$

To compute the probability $p(\Delta^{(\alpha)})$, we need to update the histograms $\mathbf{h}^{(\beta)}$ for the subband image $\mathbf{I}^{(\beta)}(\cdot, t), \beta = 1, 2, .., K$ with $\mathbf{I}(\cdot, t)$ being the status of the Markov chain at time t.

As shown in figure 17, the filter response $\mathbf{I}^{(\beta)}(u, t+1)$ at point $u \in \Lambda$ needs to be updated if the window $W_u^{(\beta)}$ overlaps with $W_v^{(\alpha)}$.

The new response of filter $F^{(\beta)}$ at u is

$$\mathbf{I}^{(\beta)}(u, t+1) = \mathbf{I}^{(\beta)}(u, t) + [(\Delta^{(\alpha)} - \mathbf{I}^{(\alpha)}(v, t))][W_v^{(\alpha)} * W_u^{(\beta)}/c^{(\alpha)}].$$

Therefore, we need to compute the norm $W_v^{(\alpha)} * W_u^{(\beta)}/c^{(\alpha)}$ and save it in a table. Thus updating $\mathbf{I}^{(\beta)}(u)$ has only $O(1)$ complexity.

References

[1] H. Akaike, "On entropy maximization principle", In *Applications of Statistics*, ed. Krishnaiah, P.R., p27-42, Amsterdam: North-Holland, 1977.

[2] C. H. Anderson and W. D. Langer, "Statistical models of image texture", *Unpublished preprint*, Washington University, St. Louis, MO, 1996. (Send email to cha@shifter.wustl.edu to get a copy).

[3] J. Besag, "Spatial interaction and the statistical analysis of lattice systems", *Journal of the Royal Statistical Society, Series B,* vol 36, pp192-236, 1973.

[4] J. A. Bucklew, *Large deviation techniques in decision, simulation, and estimation*, John Wiley and Sons, New York, 1990.

[5] T. Caelli, B. Julesz, and E. Gilbert, "On perceptual analyzers underlying visual texture discrimination: Part II," *Biological Cybernetics*, 29(4), p. 201-214 1978.

[6] R. Chellapa and A. Jain (eds) *Markov Random Fields*, 1993.

[7] C. Chubb and M. S. Landy, "Orthogonal distribution analysis: a new approach to the study of texture perception." *Comp. Models of Visual Proc.* Landy. etc (ed.) MIT press 1991.

[8] G. R. Cross and A. K. Jain, "Markov random field texture models." *IEEE, PAMI*, **5**, 25-39. 1983.

[9] J. S. De Bonet and P. Viola, "A non-parametric multi-scale statistical model for natural images", *Advances in Nueral Information Processing* 10, 1997.

[10] I. Daubechies, *Ten lectures on wavelets.* Society for Industry and Applied Math, Philadephia, PA 1992.

[11] J. Daugman, "Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Journal of Optical Soc. Amer.* Vol.2, No.7, 1985.

[12] A. Gagalowicz and S. D. Ma, "Model driven synthesis of natural textures for 3D scenes", *Computers and Graphics.* 10:161-170, 1986.

[13] S. Geman and D. Geman. "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images". IEEE Trans. on PAMI 9(7), pp 721-741. 1984.

[14] S. Geman and C. Graffigne, "Markov random field image models and their applications to computer vision." *Proc. Int. Congress of Math.*, Berkeley, CA, 1986.

[15] H. O. Georgii, *Gibbs Measures and Phase Transition.* de Gruyter, New York, 1988.

[16] W. R. Gilks and R. O. Roberts, "Strategies for improving MCMC", chapter 6 in W. R. Gilks et al (eds) *Markov Chain Monte Carlo in Practice*, Chapman & Hall, 1997.

[17] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, *Markov Chain Monte Carlo in Practice*, Chapman & Hall, 1997.

[18] D. J. Heeger and J. R. Bergen, "Pyramid-based texture analysis/synthesis." *SIGGRAPHS*, 1995.

[19] B. Julesz, "Visual pattern discrimination", *IRE Transactions on Information Theory*, IT-8:84-92, 1962.

[20] B. Julesz, "Toward an axiomatic theory of preattentive vision", pp. 585-612 in *Dynamic aspects of neocortical function*, G. Edelman et al. (Eds.), Wiley, New York 1984.

[21] B. Julesz, *Dialogues on Perception*, 1995.

[22] Karni, A and Sagi, D, " Where practice makes perfect in texture discrimination –evidence for primary visual cortex plasticity", *Proc. Nat. Acad. Sci. US,* vol. 88, 4966-4970, 1991.

[23] S. Kullback, and R. A. Leibler, "On information and sufficiency", *Annual Math. Stat.* vol.22, pp79-86, 1951.

[24] J. S. Liu, C. Sabatti, and Y. N. Wu, "Haar measure and Gibbs sampler", preprint of the department of statistics, Univ. of Michigan, 1999.

[25] D. Marr, *Vision*, W.H. Freeman and Company, New York, 1982.

[26] K. Popat and R. W. Picard, "Cluster based probability model and its application to image and texture processing", *IEEE Trans. Imformation Processing*, 6(2), 1997.

[27] J. Portilla and E. P. Simoncelli, "Texture representation and synthesis using correlation of complex wavelet coefficient magnitudes", *Prof. IEEE workshop on Statistical and Computational Theories of Vision*, Fort Collins, CO, June, 1999.

[28] R. Vistnes, "Texture models and image measures for texture discrimination", *Int'l Journal of Computer Vision*, 3, 313-336, 1989.

[29] J. Weszka, C. R. Dyer, and A. Rosenfeld, "A comparative study of texture measures for terrain classification" *IEEE Trans. Sys. Man, Cybern.* Vol. SMC-6, April, 1976.

[30] Y. N. Wu, S. C. Zhu and X. W. Liu, "The equivalence of Julesz and Gibbs ensembles", *Proc. Int'l Conf. on Computer Vision*, Corfu, Greece, September, 1999.

[31] A. L. Yuille and J. M. Coughlan, "Visual search: fundamental bounds, order parameters, and phase transition", *Proc. of IEEE Workshop on Statistical and Computational Theories of Vision*, June 1999 (with CVPR99).

[32] S. C. Zhu, Y. N. Wu, and D. Mumford. "FRAME: Filters, Random field And Maximum Entropy: — Towards a Unified Theory for Texture Modeling", *Int'l Journal of Computer Vision*, 27(2) 1-20, March/April, 1998.

[33] S. C. Zhu, Y. N. Wu, and D. Mumford. "Minimax entropy principle and its application to texture modeling". Neural Computation, Vol. 9, no 8, Nov. 1997.

[34] S. C. Zhu and D. Mumford. "Prior learning and Gibbs reaction-diffusion." *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol.19, no.11, Nov. 1997.

[35] S. C. Zhu, "Embedding Gestalt laws in Markov random fields", *IEEE Trans. on PAMI*, (in press), 1999.