Proceedings of the SPIE AeroSense Conference on Automatic Target Recognition, Orlando, FL, April, 2000.

Mathematical Modeling of Clutter: Descriptive vs. Generative Models

Song-Chun Zhu and Cheng-En Guo

The Ohio State University, Columbus, OH 43210.

ABSTRACT

In this article, we present two mathematical paradigms for clutter modeling. Both paradigms pose clutter modeling as a statistical inference problem, and pursue probabilistic models for characterizing observed training images. The two paradigms differ in the forms (or families) of models that they choose and in their philosophical assumptions on real world clutter patterns. The first paradigm studies *descriptive* models, such as Markov random field (MRF) models and the minimax entropy models(Zhu, Wu, and Mumford 1997).[14] In this modeling paradigm, image features are first extracted from images, and and statistics of these features are calculated. The latter define an *image ensemble*–called the Julesz ensemble which is an equivalence class where all images share the same feature statistics. For any large images from this ensemble, a local patch given its boundary condition is then Gibbs (or MRF) models,[11] We shall review the recent conclusions about ensemble equivalence studied in (Wu, Zhu and Liu, 1999).[11] The second paradigm studies generative model, such as the random collage model (Lee and Mumford, 1999).[6] In contrast to a descriptive model, a generative model introduces hidden variables which are assumed to be the underlying causes producing the observed image. For example, trees and rock for clutter. The learning process makes inference about the hidden variables. We shall discuss a texton model for clutter and effective Markov chain Monte Carlo methods for stochastic inference. We shall also reveal the deep relationship between the two modeling paradigm.

Keywords: Clutter Modeling, Gibbs models, minimax entropy, generative models, Markov chain Monte Carlo.

1. INTRODUCTION

In recent years, there has been increasing interest in modeling clutter for ATR as well as computer vision applications. This trend is well reflected in three aspects. The first is the study of statistics of natural images and generic image models[5].[6] The second is the development of learning theory for estimating realistic models of images, such as the Julesz ensembles[13] and minimax entropy learning.[14] The third is the study of performance bounds for ATR and other vision tasks, where clutter models, as null hypotheses, are crucial for accurate detection and classification.[9,4,12,10] In this paper, we overview recent progress on the second aspects: learning realistic clutter models from training images. We use examples of optical images such as those in figure 1, assuming the same theory and algorithms are applicable to other sensor imagery such as FLIR and SAR images.

Figure 1. Stochastic patterns in nature.

One basic observation in clutter modeling is that real world imagery consists of patterns generated by multiple stochastic processes, such as the placement of objects, their texture patterns and shapes. Thus visual patterns

S. C. Zhu and C. E. Guo are with the Department of Computer and Information Science, The Ohio State University, Columbus, OH 43210. The work is supported by an ARO grant DAAD19-99-1-0012 and an NSF grant IIS-98-77-127. We thank Y. N. Wu for stimulating discussion on generative models. Send correspondence to {szhu, cguo}@cis.ohio-state.edu.

we observed in a given application domain are samples from an underlying probabilistic distribution. The latter characterize the *ensemble* of images in a certain environment. For example, battle field scenes in a desert are very different from those in a tropical area. So we pose clutter modeling as a statistical inference problem, and the task of quantifying clutter is fulfilled by estimating a probabilistic distribution $p(\mathbf{I})$ for clutter pattern $\mathbf{I}$. $p(\mathbf{I})$ is often estimated from a set of training images so that it minimizes the Kullback-Leibler divergence $D(f||p)$ where $f(\mathbf{I})$ is assumed to be the unknown "true" distribution.

All models for clutter are divided into two general families: descriptive and generative.

1). **Descriptive models**. Descriptive models are often called *exponential* models in statistics, *Gibbs or maximum entropy* models in statistical physics, and *declarative* models in computer science. This family of models for clutter include Markov random field models and the minimax entropy models.[14] In a descriptive model, an image $\mathbf{I}$ is characterized by a set of features which are extracted from $\mathbf{I}$ by deterministic transforms. Then statistics of these features are pooled across the image to form the so-called the *sufficient statistics*. The image model assigns equal probability to all images in an equivalence class for where images share the same feature statistics. Each equivalence class is called an *Julesz ensemble* in (Zhu, Liu and Wu 1999). A Julesz ensemble corresponds to an uniform distribution on a 2D plane. It was shown in (Wu, Zhu and Liu, 1999) that for a large image from the Julesz ensemble, any local patch given its boundary condition is subject to a Gibbs distribution (or MRF model).[11] The descriptive models become more and more flexible and general when the number of extracted features increases.

2). **Generative models**. Generative model are often called *mixture* models in statistics and they are called *causal* or *procedure* models in computer science. In contrast to a descriptive model, a generative model introduces hidden variables which are assumed to be the underlying causes producing the observed image. For example, it introduces explicitly concepts for clutter, such as trees and rock. The value of the hidden variables are subject to a posterior distribution and should be inferred during the learning process. The hidden variables are closely related to but very different from those features in a descriptive model. Examples of generative models for clutter include the random collage model[6] and texton model that we are going to present in this paper.

This paper is organized as follows. In section (2), we present a general mathematical framework for clutter modeling. Then we study the descriptive models in sections (3), where we review the Julesz and Gibbs ensembles. Then in section (4) we study multi-layer generative models. We present recent experiments on clutter modeling in both cases. We conclude the paper by a discussion in section (5).

2. A MATHEMATICAL FRAMEWORK OF CLUTTER MODELING

In this section, we study the basic assumptions and theories for clutter modeling in the framework of statistical inference, or the so-called unsupervised learning in the machine learning community.

Given a set of M observed images* $S = \{\mathbf{I}_1^{\rm obs}, \mathbf{I}_2^{\rm obs}, ..., \mathbf{I}_M^{\rm obs}\}$. We assume that these images are samples from an image ensemble in a given application domain. The latter is governed by an unknown probability distribution $f(\mathbf{I})$ – dubbed the "God's model". The objective of clutter modeling is to estimate $f(\mathbf{I})$ by a probability model $p(\mathbf{I})$ from the training data S so that $p(\mathbf{I})$ approaches $f(\mathbf{I})$. One conventional measure for the distance of two distributions is the so-called *Kullback-Leibler divergence* or *cross entropy*,[2]

$$\begin{aligned} KL(f(\mathbf{I})||p(\mathbf{I})) &= \int f(\mathbf{I}) \log \frac{f(\mathbf{I})}{p(\mathbf{I})} d\mathbf{I}, \\ &= E_f[\log f(\mathbf{I})] - E_f[\log p(\mathbf{I})]. \end{aligned}$$

Therefore, we seek an optimal probability $p^*(\mathbf{I})$ which has minimum distance from f,

$$p^* = \arg\min_{p \in \Omega_p} KL(f(\mathbf{I})||p(\mathbf{I})) = \arg\max_{p \in \Omega_p} E_f[\log p(\mathbf{I})] \approx \arg\max_{p \in \Omega_p} \sum_{i=1}^{M} \log p(\mathbf{I}_i), \tag{1}$$

where Ω_p is the family of distributions where p^* is searched for. In the above deduction, we notice that $E_f[\log f(\mathbf{I})] = -\text{entropy}(f(\mathbf{I}))$ is a fixed constant independent of $p(\mathbf{I})$. We replace the expectation $E_f[\log p(\mathbf{I})]$ by the sample mean $\frac{1}{M}\sum_{i=1}^{M} \log p(\mathbf{I}_i)$. Equation (1) is often called the maximum likelihood estimate (MLE) in statistics.

*They can be other visual patterns, such as range images, object shapes and so on.

One general procedure is to search for $p(\mathbf{I})$ in a sequence of nested probability families,

$$\Omega_0 \subset \Omega_1 \subset \cdots \subset \Omega_k \rightarrow \Omega_f \ni f.$$

k indexes the dimensionality of the space or flexibility of the model within the space, for example, the number of free parameters in a model. As k increases, the probability family should be general enough to contain the true distribution $f(\mathbf{I})$.

Now a key question is: in what family (or space) Ω_p should we search for $p(\mathbf{I})$? There are two families of Ω_p based on two philosophic assumptions, and both are general enough for approximating any distributions $f(\mathbf{I})$.†

Choice I: descriptive models.

The general philosophy in constructing descriptive models is that pixels intensities $\mathbf{I}$ are the *only* variables that we want to deal with. That is, we accept the statistical fluctuations as given (or as described) and we assume that there is no underlying causes for generating $\mathbf{I}$. We characterize $\mathbf{I}$ by a set of K features, and compute the statistics for these features across the image. The statistics are denoted by $\phi_j(\mathbf{I}), j = 1, 2, ..., K$.

Then we obtain Gibbs distributions in exponential forms derived from the *maximum entropy principle* subject to the constraints that the model should reproduce the same statistics as observed.

$$p(\mathbf{I}; \boldsymbol{\beta}) = \frac{1}{Z(\boldsymbol{\beta})} \exp\{-\sum_{j=1}^{K} \beta_j \phi_j(\mathbf{I})\}.$$

$\boldsymbol{\beta} = (\beta_1, ..., \beta_K)$ are the parameters for the model. In a limit as the image lattice goes to infinity, the statistical fluctuations diminishes,‡ and the Gibbs model $p(\mathbf{I}; \boldsymbol{\beta})$ is focused on an equivalence class which is called the Julesz ensemble.[11] If we choose a sufficient set of features and statistics, then the Julesz ensemble should be identical to the ensemble of images in our task domain.

The descriptive learning proceeds in two phases guided by the minimax entropy principle.[14]

Phase I: we augment the dimension of the space by increasing the number of feature statistics and generate a sequence of exponential families,

$$\Omega_1^d \subset \Omega_2^d \subset \cdots \Omega_K^d \rightarrow \Omega_f.$$

Phase II: within a family Ω_k^d, we estimate the optimal parameters,

$$\boldsymbol{\beta}^* = \arg \max_{p \in \Omega_K^d} \sum_{i=1}^{M} \log p(\mathbf{I}_i^{\text{obs}}; \boldsymbol{\beta}).$$

Choice II: generative models.

The general philosophy of generative models is that one assumes that observed images are only part of the random variables and there are hidden random variables influencing the appearance of image $\mathbf{I}$. We denote by W the hidden variables which represent contents and structures of the "world" captured in the image $\mathbf{I}$. In this way, we assume the existence of a joint probability distribution $f(\mathbf{I}, W)$. For example, in figure 1, the images are observed as pixels, but trees and twigs are not directly observable. These hidden variables are supposed to generate the observed data $\mathbf{I}$, and they are inferred from the observed data.

Let $W = (w_1, w_2, ..., w_k)$ be the vector-valued hidden variable, by MLE in equation (1), we estimate a model by MLE in equation (1).

$$p(\mathbf{I}; \Theta) = \int \cdot \int p(\mathbf{I}, w_1, w_2, ..., w_k; \boldsymbol{\beta}) \prod_{i=1}^{k} dw_i.$$

The learning procedure for generative models consists of two phases.

†To our knowledge, all probability models are categorized into these two choices. There seem no other alternatives.

‡This is only true in the absence of phase transitions. We have not observed phase transition phenomena in natural texture patterns so far.

Phase I: incrementally add hidden variables and corresponding parameters β to augment the joint model Thus we have a sequence of mixture families

$$\Omega_1^g \subset \Omega_2^g \subset \cdots \subset \Omega_K^g \rightarrow \Omega_f \ni f.$$

Phase II: within Ω_k^g, we estimate the parameters Θ by MLE, which include making inference about W.

In the rest of this paper, we shall further discuss the following three questions.

1. *What are the relationships between generative models and descriptive models?* We show that a descriptive model is always needed for characterizing the hidden variables and is a precursor of generative model.

2. *What are the relationship between image features in a descriptive model and hidden variables in a generative model?* We show that good image features in a descriptive model have high mutual information[2] with the hidden variables in a generative model. We argue that these feature extraction techniques is an effective computational means for making inference about the hidden variables in a generative model. This is discussed in a recent stochastic computing framework called data driven Markov chain Monte Carlo.[15]

3. *What are effective algorithms for learning the descriptive models and generative models?* We shall briefly introduce the data driven Markov chain Monte Carlo algorithms for inference and stochastic gradient for optimization.

3. FAMILY ONE: DESCRIPTIVE MODELS

In this section, we review the recent progress in clutter modeling by descriptive models. We shall discuss the Julesz ensemble, the FRAME model and equivalence of the two concepts on large lattice. then we briefly discuss the computational issues. Detailed studies of these issues are referred to a series of papers,[14],[11] ,[13].[16]

3.1. Julesz image ensembles

We denote by $\mathbf{I}$ an image defined on a lattice Λ, and $\Omega_\Lambda = G^{|\Lambda|}$ the space of all possible images with G being the grey level of pixel intensities.

Motivated by receptive fields of neurons in primary visual cortex and wavelet coding theory, we choose a set of Gabor filters at various scales and orientations to extract spatial features of clutter. Given K Gabor filters $\{F^{(1)}, ..., F^{(K)}\}$, we convolve the filters with the image $\mathbf{I}$ to obtain K subband images

$$\{\mathbf{I}^{(1)}, ..., \mathbf{I}^{(K)}\}, \quad \text{with} \quad \mathbf{I}^{(i)} = F^{(i)} * \mathbf{I},\ i = 1, 2, ..., K$$

Then we compute the image statistics averaged over the entire image under the assumption that the image is homogeneous. We choose the normalized histograms of these K subband images,

$$\mathbf{h}(\mathbf{I}) = (h^{(1)}(\mathbf{I}), ..., h^{(K)}(\mathbf{I})),$$

where $h^{(i)}$ is the normalized intensity histogram of $\mathbf{I}^{(i)}$, for $i = 1, 2, ..., K$.

Given statistics $\mathbf{h}(\mathbf{I})$, one can partition the image space Ω_Λ into equivalence classes where all images have identical statistics,

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

With a slight abuse of notation here, we denote by $\mathbf{h}(\mathbf{I})$ the extracted statistics from $\mathbf{I}$, and $\mathbf{h}$ a value for the statistics. For finite Λ, the exact constraint $\mathbf{h}(\mathbf{I}) = \mathbf{h}$ may not be satisfied, so we relax this constraint, and replace $\Omega_\Lambda(\mathbf{h})$ by

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

with $\mathcal{H}$ being a small set around $\mathbf{h}$. Then we can define the uniform counting measure or the uniform probability distribution on $\Omega_\Lambda(\mathcal{H})$ as

$$q(\mathbf{I}; \mathcal{H}) = \begin{cases} 1/|\Omega_\Lambda(\mathcal{H})|, & \text{if } \mathbf{I} \in \Omega_\Lambda(\mathcal{H}), \\ 0, & \text{otherwise,} \end{cases} \tag{2}$$

where $|\Omega_\Lambda(\mathcal{H})|$ is the volume of or the number of images in $\Omega_\Lambda(\mathcal{H})$. The Julesz ensemble (Zhu, Liu and Wu, 2000)[13] is then defined below.

Definition *Given a set of feature statistics* $\mathbf{h}(\mathbf{I}) = (h^{(1)}(\mathbf{I}), ..., h^{(K)}(\mathbf{I}))$*, a Julesz ensemble of type* $\mathbf{h}$ *is a limit of* $q(\mathbf{I}; \mathcal{H})$ *as* $\Lambda \to Z^2$ *and* $\mathcal{H} \to \mathbf{h}$ *with some boundary condition.*§

For homogeneous clutter patterns, as $\Lambda \to Z^2$ the statistical fluctuations diminish, a Julesz ensemble is identified with an unique clutter pattern. It is always helpful to imagine the Julesz ensemble of type $\mathbf{h}$ as the image set $\Omega_\Lambda(\mathbf{h})$ on a large Λ. In practice, we often ignore the minor complication that the constraint $\mathbf{h}(\mathbf{I}) = \mathbf{h}$ may not be exactly satisfied, and shall simply take $\mathcal{H}$ to be $\mathbf{h}$.

3.2. Gibbs distributions and FRAME models

While a Julesz ensemble identifies a homogeneous clutter pattern uniquely by type $\mathbf{h}$ on Z^2, on a finite small lattice the statistics of different Julesz ensembles overlap due to statistical fluctuations, and image boundary condition plays an important role. Therefore, on finite lattice, clutter is best represented by a conditional probability distribution. Very often one calls the conditional distribution a *model* of clutter. In target recognition tasks, the conditional distribution provides the local null hypothesis whereas the Julesz ensemble defined globally is not useful.

Among the studied descriptive models, Markov random field models, or equivalently the Gibbs distributions, are the most popular and elegant ones. Recently, (Zhu, Wu, and Mumford, 1997)[14] proposed a class of MRF models called FRAME (Filter, Random field, And Maximum Entropy). The basic idea is as follows.

Given statistics $\mathbf{h}(\mathbf{I})$ used in the texture definition, we compute a model $p(\mathbf{I})$ so that it has the expected statistics $\mathbf{h}$, i.e.,

$$E_p[\mathbf{h}(\mathbf{I})] = \mathbf{h}.$$

This is a "soft" constraint because it only requires that the statistics $\mathbf{h}(\mathbf{I})$ should be matched on average in comparison with the hard constraint $\mathbf{h}(\mathbf{I}) = \mathbf{h}$ in the Julesz ensemble.

Then a maximum entropy distribution, called FRAME, is selected among all distributions that satisfy the constraint. The distribution has the following exponential form

$$p(\mathbf{I}; \boldsymbol{\beta}) = \frac{1}{Z_\Lambda(\boldsymbol{\beta})} \exp\{- < \boldsymbol{\beta}, \mathbf{H}(\mathbf{I}) >\}, \tag{3}$$

where $\boldsymbol{\beta}$ is the parameters of the model and $Z_\Lambda(\boldsymbol{\beta})$ is the normalizing constant. Parameters $\boldsymbol{\beta}$ are solved from the constraint $E_{p(\mathbf{I};\boldsymbol{\beta})}[\mathbf{h}(\mathbf{I})] = \mathbf{h}$. $p(\mathbf{I}; \boldsymbol{\beta})$ in equation (3) is a general form which generalizes existing MRF texture models. A detailed account of the FRAME models and the minimax entropy principle in selecting statistics $\mathbf{h}(\mathbf{I})$ is referred to an early paper (Zhu, Wu, and Mumford, 1997).[14] Unlike the Julesz ensembles, the FRAME models assign probabilities to all the images defined on Λ. Although the FRAME models are less straightforward than the Julesz ensembles, they are much more analytically tractable due to the Markov property.

3.3. Equivalence between Julesz ensembles and FRAME models

In most recent work by (Wu, Zhu and Liu, 1999),[11] the equivalence between the Julesz ensemble and the FRAME model has been established using the fundamental principle of equivalence of ensembles in statistical mechanics. Due to space limitation, we only review the significant implications of the equivalence.

Given a set of feature statistics $\mathbf{h}$, one can construct a Julesz ensemble $\Omega_\Lambda(\mathcal{H})$ and a FRAME model $p(\mathbf{I}; \boldsymbol{\beta})$ on lattice Λ, and the former is associated with a uniform distribution $q(\mathbf{I}; \mathcal{H})$. Then we have the following conclusions.

- As $\Lambda \to Z^2$, a FRAME model concentrates its probability mass uniformly on a corresponding Julesz ensemble $\Omega(\mathbf{h})$. Therefore, the Julesz ensemble characterizes the global statistical property of the FRAME model;
- For a large image randomly sampled from a Julesz ensemble, any local patch of the image given its environment follows the conditional distribution specified by a corresponding FRAME model. Therefore, the FRAME model describes the local statistical property of the Julesz ensemble, and is an *inevitable descriptive model* on finite (small) lattice.

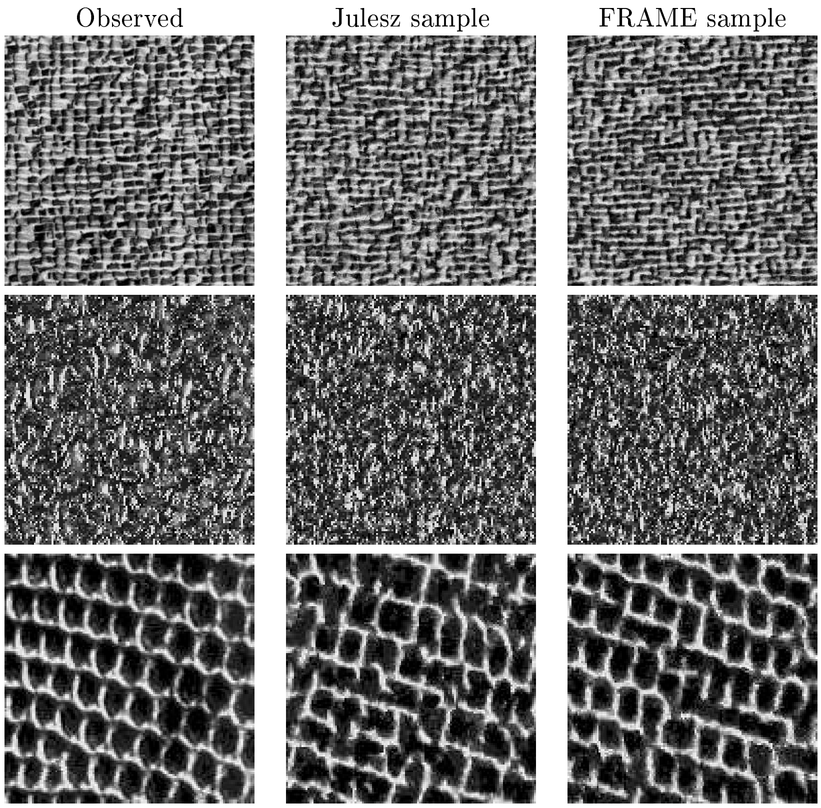


Figure 2. For each row, the left image is observed as the training image, the middle image is a typical sample from the Julesz ensemble, and the right image is a typical sample from the corresponding FRAME model. An example from (Wu, Zhu and Liu, 2000).

The ensemble equivalence principle is deeply rooted in statistical mechanics dated back to Gibbs 1902. One fundamental property shared by the Julesz ensemble and the Gibbs (FRAME) model is that both are descriptive. In modeling clutter/texture patterns, both methods use statistical properties to constrain the set of images. The Julesz ensemble impose the constraints for images on large lattice, whereas the Gibbs (FRAME) models impose a soft constraints on the probability distribution. The learning process for both methods is guided by a minimax entropy principle, at each step a new feature is added to reduce the entropy of the Gibbs model (or the volume of the Julesz ensemble). For a fixed set of feature statistics $\mathbf{h}$, $p(\mathbf{I}; \boldsymbol{\beta})$ is constructed by maximum entropy (or a uniform distribution in the Julesz ensemble).

The learning procedure for descriptive models of clutter/texture consists of two phases.

Phase I: Pursuing information bearing feature statistics $\mathbf{h}$ from a dictionary by sampling from the Julesz ensemble using Markov chain Monte Carlo methods.[13]

II: Learning the parameters $\boldsymbol{\beta}$ by MLE using various likelihood maximization techniques, such as maximum pseudo-likelihood, maximum partial likelihood, and maximum satellite likelihood.[16] Some algorithms, such as the maximum satellite likelihood can approximately estimate $\boldsymbol{\beta}$ with the speed of seconds, which is two orders of magnitude faster than the original algorithm in (Zhu, Wu, and Mumford, 1997).[14]

We demonstrate some experiments in figure 2. We refer to (Wu, Zhu and Liu, 2000)[11] for more results. For each row, the left image is observed as the training image. We extract histograms of filtered images using a set of Gabor filters. The middle image is a typical sample from the Julesz ensemble defined by the extracted histograms. Then we also learn a FRAME model, and the right image is a typical sample from the FRAME model.

[§]We assume $\Lambda \to Z^2$ in the sense of van Hove, i.e., the ratio between the boundary and the size of Λ goes to 0.

4. FAMILY TWO: GENERATIVE MODELS

In this section, we study the second family – generative models for clutter/texture patterns. We start with studying some simple and popular generative image models including principal component analysis (PCA), sparse coding. Then we proceed to discuss more sophisticated texton models and the relationship between descriptive and generative models.

4.1. Generative models: basic ideas and simple examples.

To see the differences between a generative model and a descriptive model, we compare two equations,

$$\mathbf{h}(\mathbf{I};\Theta) = 0, \tag{4}$$

and

$$\mathbf{I} = g(W;\Theta). \tag{5}$$

Equation (4) is a descriptive model which constrains $\mathbf{I}$ by its feature $\mathbf{h}(\mathbf{I})$, and the $\mathbf{h}(\mathbf{I})$ is a deterministic function or *transform* of image $\mathbf{I}$. For example, in the Julesz ensemble, $\mathbf{h}(\mathbf{I})$ is a set of histogram of filtered images computed from $\mathbf{I}$. In contrast, equation (5) is a generative model, and $\mathbf{I}$ is produced by *hidden variables or causes* W. In a generative model, the randomness of $\mathbf{I}$ comes from the randomness of the underlying causes W. In a descriptive model, the randomness in $\mathbf{I}$ results from maximum entropy within the set of images satisfying the constraints, i.e. a uniform distribution in unconstrained dimensions.

There are two popular generative image models in the literature.

The first is principal component analysis (PCA). In PCA an image $\mathbf{I}$ is generated by a linear combination of some orthogonal vectors $\mathbf{V}_i, i = 1, 2, ..., m$ plus Gaussian noise $\mathbf{n}$. $\mathbf{n}$ accounts for the rest components.

$$\mathbf{I} = \sum_{i=1}^{m} \alpha_i \mathbf{V}_i + \mathbf{n}, \tag{6}$$

where $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_m)$ with α_i being the magnitude of the i-th component. In PCA $W = (\boldsymbol{\alpha}, \mathbf{n})$ is the hidden variables or causes, and $\Theta = (\mathbf{V}_1, \mathbf{V}_2, ..., \mathbf{V}_m)$ is the parameters. in PCA, all hidden variables in W are assumed to be iid Gaussian distributions.

The second is image coding using over-complete bases[1].[8] The image coding model is a linear superposition of some wavelet window functions,

$$\mathbf{I} = \sum_{i=1}^{M} a_i \Psi_i + \mathbf{n}. \tag{7}$$

The parameters $\Theta = \{\Psi_1, ..., \Psi_M\}$ form an over-complete dictionary and $W = (a_1, a_2, ..., a_M, \mathbf{n})$ are the hidden causes. $\mathbf{n}$ is set to zero for lossless coding. Since Θ is over-complete, there are many possible W for generating $\mathbf{I}$. Thus W is inferred from $\mathbf{I}$. In the literature, $a_i, i = 1, 2, ..., M$ are assumed to be iid Laplacian distributed for sparse coding $a_i \sim \exp^{-|a_i|/\delta}$[8] and $\mathbf{n}$ are iid Gaussian distributed.

In a generative model, the distribution of $\mathbf{I}$ is a projection of the joint probability density $p(\mathbf{I}, W; \Theta)$

$$p(\mathbf{I};\Theta) = \int p(\mathbf{I}|W;\Theta)p(W;\Theta)dW. \tag{8}$$

In PCA, since both $p(\mathbf{I}|W;\Theta)$ and $p(W;\Theta)$ are Gaussian distributed, $p(\mathbf{I};\Theta)$ is also a Gaussian distribution. Thus Θ can be analytically solved. They are the eigenvectors of the covariance matrix of observed images, and this is straight forward MLE following equation (1).

Both of the above image models are limited in at least three aspects.

1. The assumption of iid distribution for hidden variables W are naive. In real clutter patterns, they are highly correlated between each other. For example, along a long edge, the wavelet coefficiencies are highly correlated.
2. The assumption of linear superposition is not realistic because real clutter patterns include sharp boundaries caused by occlusions between objects.
3. There lacks the concept of objects and the iid Laplacian distribution is far from enough in enforcing the sparse coding principle.[8]

4.2. A generative texton model

In this section, we propose a generative model for homogeneous clutter patterns as a step towards more realistic clutter models. Figure 3 shows three examples of clutter images. In each image, we observe distinct elements which repeats across the image and form spatial structures. For example, the circular blobs, cracks and tree trunk segments illustrated by rectangles. These elements are superimposed on background patterns which has fine scale stochastic features. In some images, there may be more than 2 levels, see the left image in figure 1.

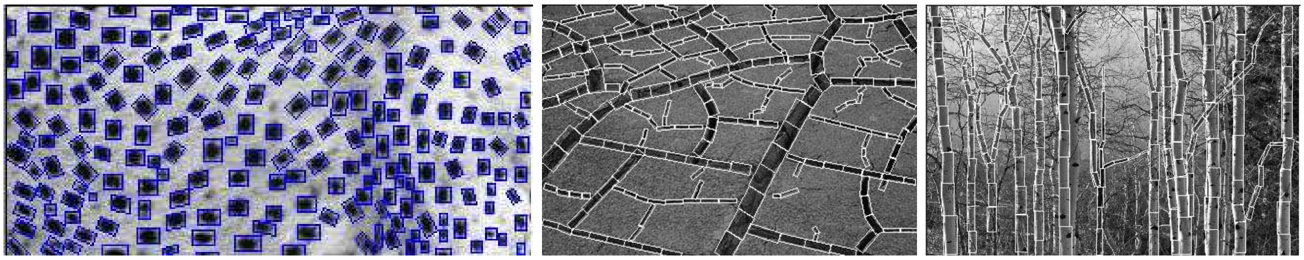


Figure 3. Clutter patterns with basic elements: texels or textons.

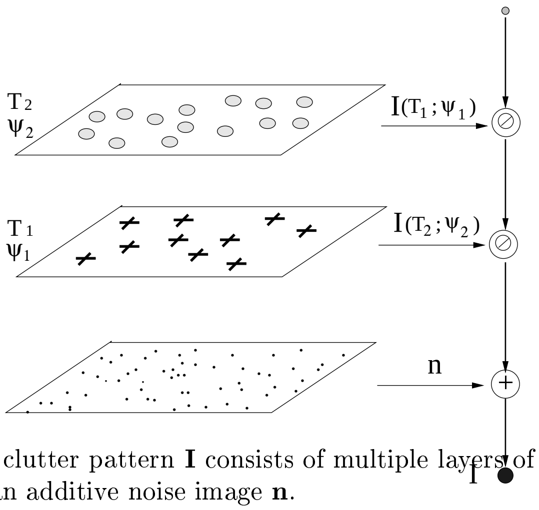


Figure 4. A generative model for clutter pattern $\mathbf{I}$ consists of multiple layers of texton map $\mathbf{I}(\mathbf{T}_i;\Psi_i), i = 1, 2..., k$ superimposed with occlusion plus an additive noise image $\mathbf{n}$.

We propose a multi-layer generative model illustrated in Figure 4. An image $\mathbf{I}$ is rendered by k (e.g. $k = 2$ or 3) layers of stochastic processes which are superimposed on each other with occlusion. Each layer has a set of texture elements (called textel or texton) forming a *texton map* $\mathbf{T}_i$. $\mathbf{T}_i$ includes random variables for the number of textons n_i and the locations x_{ij}, y_{ij}, orientations, and scales τ_{ij}, s_{ij} for the j-th token.

$$\mathbf{T}_i = (n_i, \{(x_{ij}, y_{ij}, \tau_{ij}, s_{ij}, A_{ij}) \ : \ j = 1, 2, ..., n_i.\}) \quad i = 1, 2, ..., k. \tag{9}$$

Each texton is generated from an unique mini-template pattern Ψ_i through similarity transform as well as intensity variations, such as lighting. We use A_{ij} to account for other aspects, such as variables for small deformation and lighting variations which can be characterized by PCA coefficiencies.

At each layer, the texton map $\mathbf{T}_i$ and the mini-template Ψ_i¶ decide an image $\mathbf{I}_i = \mathbf{I}(\mathbf{T}_i;\Psi_i)$. In image $\mathbf{I}_i$, pixels not covered by the textons are transparent. Then image $\mathbf{I}$ is generated by the following model

$$\mathbf{I} = \mathbf{I}(\mathbf{T}_1;\Psi_1) \ \oslash \ \mathbf{I}(\mathbf{T}_2;\Psi_2) \ \oslash \cdots \oslash \ \mathbf{I}(\mathbf{T}_2;\Psi_2) \ + \ \mathbf{n}. \tag{10}$$

The symbol $\oslash$ denotes occlusion, i.e. $\mathbf{I}_1 \oslash \mathbf{I}_2$ means $\mathbf{I}_1$ occludes $\mathbf{I}_2$.

¶Donoho et al calls them ridgelets, curvelets, wedgelets in a similar spirit, see[1] and reference therein.

In this generative model, the hidden variables are

$$W = (k, \{(\mathbf{T}_i, d_i) : i = 1, 2, ..., k\},\ \mathbf{n}),$$

where d_i indexes the order (or relative depth) of the i-th layer. W is assumed to be the underlying causes and structures. For example, in figure 3, the cheetah skin pattern (left) is generated by two layers: the top level is the blobs illustrated by the rectangles, and the bottom level is a smooth white background which can be generated by additive models as splines. Each knot of the spline is a texton with a Gaussian window function Ψ. Similarly we can generate the middle and right images with two layers using bar-shaped cracks or tree trunks at various scales and orientations.

The likelihood model $p(\mathbf{I}|W;\Theta)$ is simply Gaussian distributed following the distribution of $\mathbf{n}$. In the next section we discuss a descriptive model for W.

4.3. A descriptive model of texton map

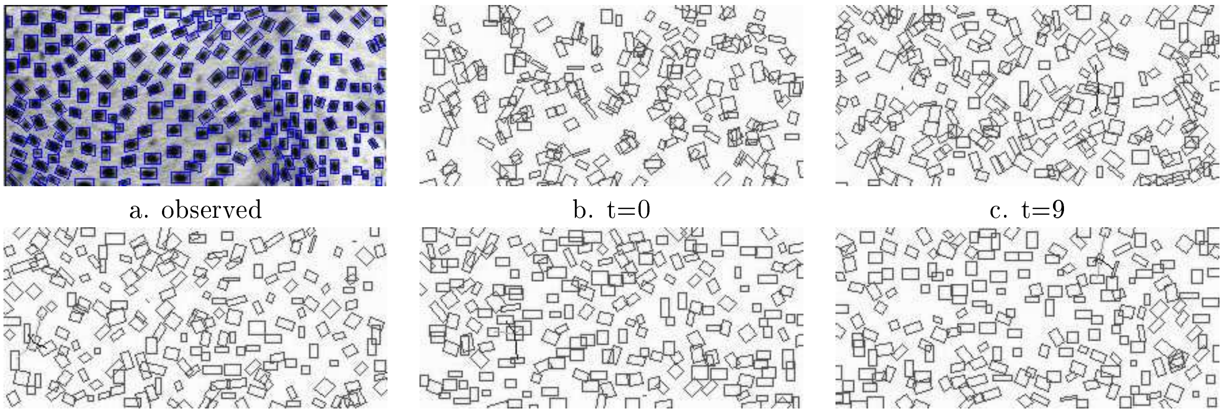
As an approximation, we assume the texton layers are independent of each other, i.e. $\mathbf{T}_i$ and $\mathbf{T}_j$ are independent if $i \neq j$. Within each layer we need a model that accounts for the regularities and structures in the texton map. Since our generative model in equation (10) does not have hidden variables "behind" $\mathbf{T}_i$, the model of $\mathbf{T}_i$ must be descriptive. Inevitably we choose the Gibbs models. In the future, we may introduce deeper hidden variables, such as object shapes and Gestalt groupings to extend the current model.

Suppose we are given a texton map $\mathbf{T}$, we learn a model $p(\mathbf{T};\boldsymbol{\beta})$ following the learning procedure for descriptive models. we first measure a set of features which characterize important spatial relationship between each elements. The features we chose measures the relative distance and orientations between two adjacent textons in a neighborhood. We then calculate the histogram of the features $H_j(\mathbf{T})$. A Gibbs (maximum entropy) model is then obtained,

$$p(\mathbf{T};\boldsymbol{\beta}) = \frac{1}{Z(\boldsymbol{\beta})} \exp\{-\beta_0 n - \sum_{j=1}^{K} < \beta_j, H_j(\mathbf{T}) >\}. \tag{11}$$

In $p(\mathbf{T};\boldsymbol{\beta})$, we $H_j(\mathbf{T})$ is the unnormalized histogram of certain feature, and all $\beta_j, j = 1, 2, ..., K$ are vector valued Lagrange multipliers. Each β_j is shifted so that it has zero sum. λ_0 is a parameter controlling the texton density.

We choose not to unfold the details of the descriptive model, instead we show some experiments in figures 5 and 6 for how we learn and sample $p(\mathbf{T};\boldsymbol{\beta})$.



a. observed b. t=0 c. t=9

d. t=69 e. t=141 f. t=234

Figure 5. The observed image is shown in a with texture elements illustrated by the rectangular windows. b)-f) are typical images during the learning process of a Gibbs model $p(\mathbf{T};\boldsymbol{\beta})$.

Suppose we are given a texton map in a supervised way, see in figure 5.a, then the rest figures shows typical images sampled during the learning process when we update $\boldsymbol{\beta}$. At step $t = 234$, the synthesized texton map has

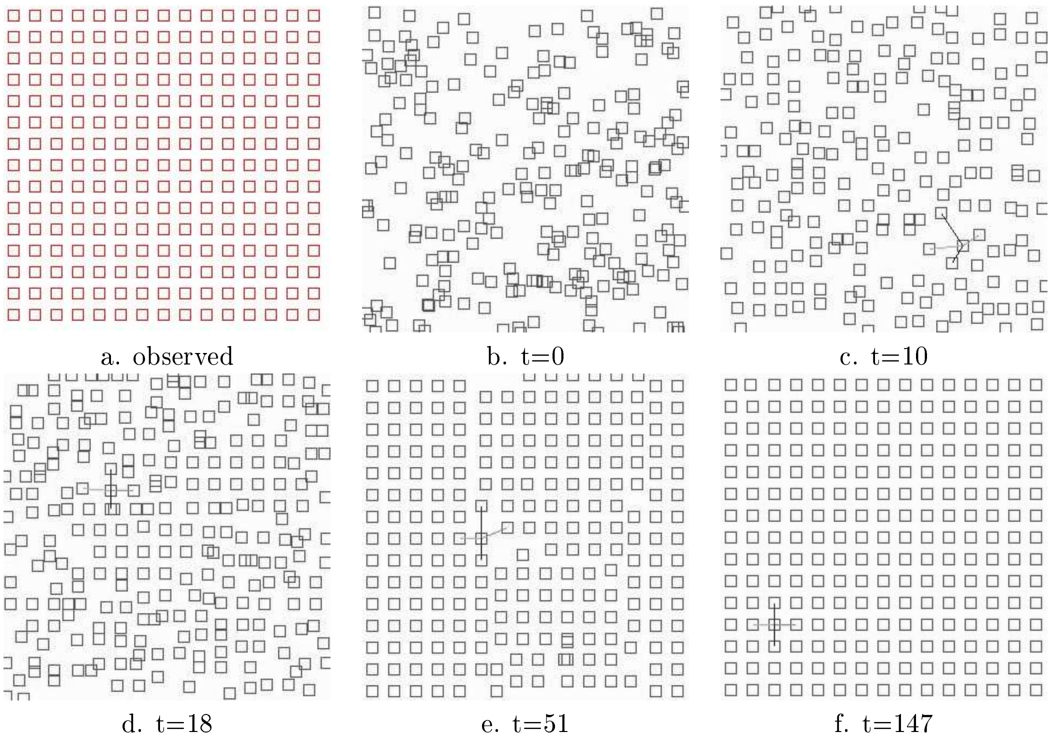


Figure 6. The observed image is shown in a) which has periodical properties. b)-f) are typical images during the learning process of a Gibbs model $p(\mathbf{T};\boldsymbol{\beta})$.

statistics close to the observed with $< 5\%$ error in histograms. Figure 6 shows a regular pattern. It is much harder to learn $p(\mathbf{T};\boldsymbol{\beta})$ for this pattern as it is extremely "cold", i.e. the Julesz ensemble contained only a few images. Special annealing strategy is used to sample this pattern In each picture, we show the 4 neighbors for one texton.

4.4. Data clustering and stochastic inference

In practice, the hidden variables $\mathbf{T}_i, i = 1, 2..., k$ are not observable, and they are inferred in the learning process. Thus the inference of $\mathbf{T}$ and the learning of Θ proceed in two iterative steps.

Let S be the observed images set, without loss of generality, we assume only one image is observed as training data. We learn the generative model for $p(\mathbf{I};\Theta)$ following the MLE procedure in equation (1).

$$\Theta^* = \arg\max_{\Theta\in\Omega^g_K} \log p(\mathbf{I}^{\text{obs}};\Theta); \quad p(\mathbf{I};\Theta) = \int p(\mathbf{I}|\mathbf{T};\Psi)p(\mathbf{T};\boldsymbol{\beta})d\mathbf{T}; \quad p(\mathbf{T};\boldsymbol{\beta}) = \prod_{i=1}^{k} p(\mathbf{T}_i;\boldsymbol{\beta}_i).$$

Our goal is to learn the parameters Θ, which include both the texton templates and the Lagrange multipliers for the Gibbs models.

$$\Theta = (\Psi, \boldsymbol{\beta}), \quad \Psi = (\Psi_1, ..., \Psi_k), \quad \text{and} \quad \boldsymbol{\beta} = (\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, ..., \boldsymbol{\beta}_k).$$

Ψ_i and $\boldsymbol{\beta}_i$ are respectively the texton template (wavelet) and parameters for one layer of the texton model in $p(\mathbf{T}_i;\boldsymbol{\beta}_i)$.

To maximizing the log-likelihood, we take the derivative with respect to Θ, and set it to zero.

$$\frac{\partial \log p(\mathbf{I}^{\text{obs}};\Theta)}{\partial\Theta} = \int \frac{\partial \log p(\mathbf{I}^{\text{obs}},\mathbf{T};\Theta)}{\partial\Theta} p(\mathbf{T}|\mathbf{I}^{\text{obs}};\Theta)d\mathbf{T} = 0.$$

We adopt a stochastic gradient method, which solves the above equations in two iterative steps.

- Step I. We estimate the left hand side using Monte Carlo integration. Given current Θ, we sample typical $\mathbf{T}^{\mathrm{syn}}$ from $p(\mathbf{T}|\mathbf{I};\Theta)$. This is a perceptual inference process.

- Step II. Then with $\mathbf{I}^{\mathrm{obs}}$ and $\mathbf{T}^{\mathrm{syn}}$, we learn Θ by gradient assent. Since we know

$$\log p(\mathbf{I}^{\mathrm{obs}}, \mathbf{T}^{\mathrm{syn}}; \Theta) = \log p(\mathbf{I}^{\mathrm{obs}}|\mathbf{T}^{\mathrm{syn}}; \Psi) + \log p(\mathbf{T}_1^{\mathrm{syn}}; \boldsymbol{\beta}_1) + \cdots + p(\mathbf{T}_k^{\mathrm{syn}}; \boldsymbol{\beta}_k).$$

  As the likelihood and parameters are separable, the second step is divided into two parts:

  a). Computing Ψ by maximizing $\log p(\mathbf{I}^{\mathrm{obs}}|\mathbf{T}^{\mathrm{syn}}; \Psi)$, and

  b). Computing $\boldsymbol{\beta}_i$ by maximizing $p(\mathbf{T}_i^{\mathrm{syn}}; \boldsymbol{\beta}_i)$ – a learning process in descriptive families. We have shown successful examples in figures 5 and 6 for this part.

The main computational complexity comes from sampling $p(\mathbf{T}|\mathbf{I}^{\mathrm{obs}};\Theta)$. Also we should start with good initial values for $\mathbf{T}$ and Ψ. The computational complexity in making inference about $\mathbf{T}$ is perhaps the direct reason for why people are forced to use the over-simplified PCA and image coding models in practice.

Fortunately the computational problem in our texton model is still solvable, thank to two recent technique breakthroughs.

The first technique is data clustering[7] and transformed component analysis (TCA).[3] Figure 7 displays the results of data clustering for three typical images. Given an image $\mathbf{I}$, we first apply a set of Gabor filters at various scales and orientations, and thus we obtained a vector-valued feature at each pixel $F(x,y) = (\mathbf{I}^{(1)}(x,y), \cdots \mathbf{I}^{(K)}(x,y))$. Then we do a data clustering in the feature spaces using K-mean clustering algorithm (we also used EM without much difference). We take $K = 49$ clusters, then we do a pseudo-inverse for the 49 centers to obtain the image patches shown in the second row. As we can see that the basic elements, such as bars at various orientations, blobs, and crosses are easily detected. These image patches can be merged through some transformation to form the initial solution for Ψ. We shall justify the data clustering and TCA from a learning theory.

The second technique is a stochastic inference technique called data driven Markov chain Monte Carlo[15] (see this proceedings). Through the data clustering (or bottom-up) process, we should also know roughly where the textons are and at what scale and orientations. That is, at each point (x,y) (or subsample the image), we compute the probability using the bottom-up technique,

$$p(s(x,y), \tau(x,y), A(x,y)|F), \quad F(x,y) = (\mathbf{I}^{(1)}(x,y), \cdots \mathbf{I}^{(K)}(x,y)), \quad \forall (x,y) \in \Lambda.$$

Therefore for each distinct textons Ψ_i, we have a so-called saliency-map for such probabilities.

Then we use these saliency map as *importance proposal probabilities* to drive the Markov chain Monte Carlo process. In this way, the saliency maps guide the MCMC search to tiny subspaces to achieve effective solutions. We shall report the final result in upcoming papers.

5. DISCUSSION

We present two learning paradigms for clutter modeling: descriptive versus generative learning. We observed the two families of models are deeply related. I). In representation, the descriptive model is a precursor for generative models. II). In computation, feature extraction is an effective way to drive the inference process for computing the hidden variables.

REFERENCES

1. S. Chen, D. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit", *SIAM Journal on Scientific Computing*, 20(1):33-61, 1999.
2. T. Cover and Thomas, *Elements of Information Theory*, 1994.
3. B. Frey and N. Jojic, "Transformed component analysis: joint estimation of spatial transforms and image components", *Proc. of Int'l Conf on Computer Vision*, Corfu, Greece, 1999.
4. U. Grenander, M.I. Miller, and A. Srivastava, "Hilbert-Schmidt lower bounds for estimators on matrix on Lie groups, preprint of Pattern Theory Group, Brown University, 1997.

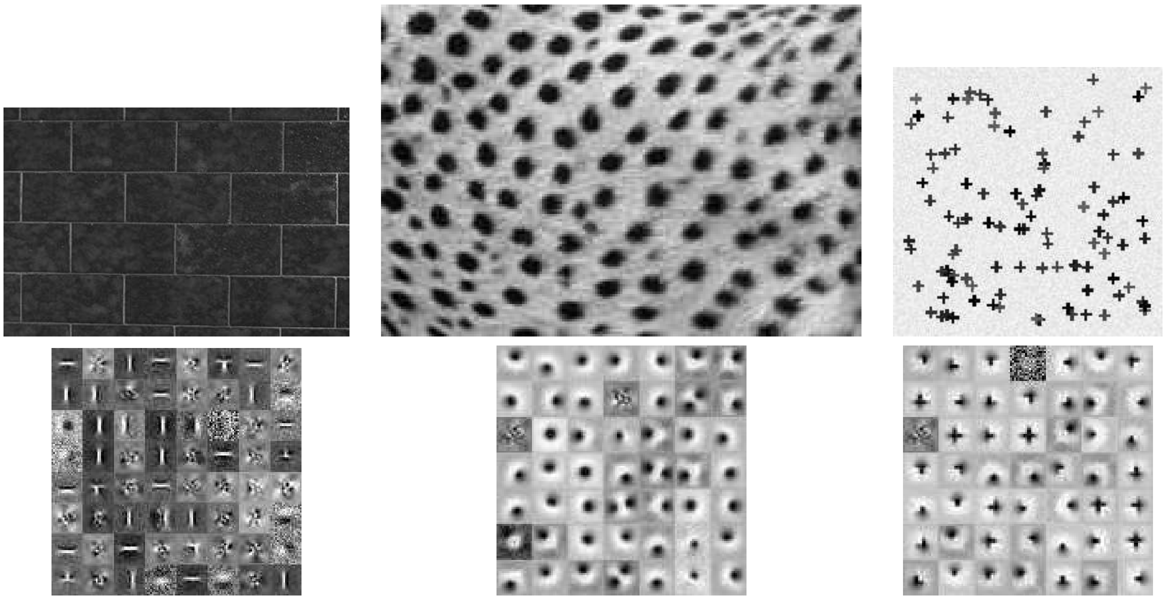


Figure 7. Three texture images and their mosaic images with 49 cluster centers.

5. J. Huang and D. B. Mumford, "Statistics of Natural Images and Models", *Proc. of Int'l conf. on Computer Vision and Pattern Recognition*, Fort Collins, CO, June, 1999.
6. A. Lee and D. B. Mumford, "An occlusion model generating scale-invariant images", *Proc. of IEEE workshop on Statistical and Computational theories of Vision*, June 1999.
7. T. Leung and J. Malik, "Recognizing surface using three-dimensional textons", *Proc. of Int'l Conf. on Computer VIsion*, Corfu, Greece, 1999.
8. B. A. Olshausen and D. J. Field, "Natural image statistics and efficient coding", *Proc. of workshop on Information Theory and the Brain*, Setpember, 1995.
9. J.A. Ratches, C.P. Walters, R.G. Buser and B.D. Guenther, "Aided and automatic target recognition based upon sensory inputs from image forming systems", *IEEE Trans. PAMI*, vol.19, No.9, Sept.1997.
10. J. Sullivan, A. Blake, M. Isard, and J. MacCormick, "Object localization by Bayesian correlation", *proc. of Int'l Conf. on Computer Vision*, Corfu, Greece, September, 1999.
11. 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.
12. A. L. Yuille and J. M. Coughlan, "Fundamental limits of Bayesian inference: order parameters and phase transitions for road tracking", *IEEE trans. on PAMI*, vol.22, no2, 2000.
13. S. C. Zhu, X. W. Liu and Y. N. Wu, "Exploring Texture Ensembles by Efficient Markov Chain Monte Carlo", *IEEE trans. on Pattern Analysis and Machine Intelligence*, (in press, 2000).
14. 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.
15. S. C. Zhu, Z. W. Tu and R. Zhang, "Stochastic inference by data driven Markov chain Monte Carlo", *SPIE conf. on ATR* (this proceedings), April, 2000.
16. S.C. Zhu and X. W. Liu, "Learning in Gibbs fields: how fast and how accurate can it be?", *Proc. of Int'l conf. on Computer Vision and Pattern recognition*, June, 2000.