

AN ADAPTIVE APPROACH FOR TEXTURE MODELLING

Michael T. Johnson and Mita D. Desai

Biomedical Image Processing Laboratory
The University of Texas at San Antonio
San Antonio, Texas 78429-0665

ABSTRACT

Markov Random Fields (MRFs) continue to be an important and useful representation for modelling textured images. Standard methods for MRF image modeling make use of the equivalent Gibbs distribution (GD) to express the joint probabilities of groups of neighboring pixels. In this paper, we investigate a new approach to the use of the GD in image modeling. Specifically, we develop an adaptive approach to the formation of clique potential functions for the distribution. Traditional tools, such as the Multi-Level Logistic (MLL) model, have been based on the use of a predetermined and identical set of potential functions. Here it is shown that by incorporating additional parameters into the model in order to control the shape of these functions, it is possible to arrive at a more complete parametrization of the image. A simple model based on this concept is described and implemented, and image simulations using the well-known Gibbs Sampler algorithm are constructed to demonstrate the usefulness of an adaptive set of potential functions.

1. BACKGROUND

Since the early 1980's, Markov Random Fields and the equivalent Gibbs Random Fields have found increasing application as image processing models for textured image data for image and texture recognition [1], [4], [6], segmentation [4], [5], and image restoration [3]. In the basic paradigm, an image lattice is modelled as a 2-D random field. If the local characteristics of regions within the image are consistent, the field can be modeled using the Markov property [2], with a group of pixels called the neighborhood region defining the region of influence of any pixel in the image. The usefulness of the MRF lies in its ability to equate the desired probability distribution of any pixel in the image with a simple conditional probability based on the pixels within its neighborhood region. The model is further simplified by the Clifford-Hammersley theorem, which equates the conditional probability of the MRF to the joint

probability described by the Gibbs distribution. The Gibbs distribution is given by:

$$P(X=x) = \frac{1}{Z} e^{-\frac{U(x)}{T}}$$

where T is a 'temperature' parameter [11] controlling the flatness of the distribution,

$$Z = \sum_{x \in X} e^{-\frac{U(x)}{T}}$$

is a normalizing constant (or partition function), and $U(x)$ is the energy function for the field, of the form

$$U(x) = \sum_{c \in C} V_c(x),$$

where c is any individual clique, C is the family of cliques realized under the defined neighborhood system, and the element $V_c(x)$ represents the potential function for each particular clique type. $V_c(x)$ produces an output dependent on the values of the members of the clique and the base function used.

It is this clique potential function which forms the building blocks of Gibbs distribution modeling. The function may be defined explicitly, or may be implied through equating the Gibbs distribution to some other probability model. The characteristics of the clique potential function depends on the characteristics of the particular Markov Random Field being modeled, and the flexibility of its definition allows for modeling of any Markovian image, if only the correct potential function can be determined for that image.

This research was supported by the
Texas Advanced Research Program.

2. GIBBSIAN MODELS

There are two distinctly different approaches for implementing the Gibbsian distribution with respect to an image lattice. The first approach is to demonstrate an equivalence between certain classes of GD and another commonly used distribution, such as the Gaussian or Poisson, and then proceed to use that distribution as a basis for images which meet the equivalence criteria. This method makes it possible to implement the Gibbs distribution in a mathematically rigorous fashion without the need for knowledge of the specific clique potential functions involved. The second approach is to develop clique potential functions directly, and to implement the Gibbs distribution based on those functions. In this type of implementation, a 'feature vector' holding a few descriptive parameters, such as a group of numbers representing 0, 90, 180, and 270 degree clustering coefficients, are fed into the pre-determined potential function equation to use in the GD calculation.

With either approach, there is an element of assumption involved in the development of GRF models. This is a result of the fact that, although the Hammersley-Clifford theorem establishes a one-to-one correspondence between MRFs and GRFs, it cannot and does not determine the exact form of the clique potential functions, which if calculated exactly for different texture classes, or even individual textures, would be different for every case. This can be seen by examining the purpose of the clique potential function within the GD, recognizing that the correct value to represent the output of the potential function would exactly represent the influence of one particular clique realization for the specific texture being examined. To assume that the form and characteristics of the basic clique potential function will remain constant for varying texture types places tight limits on the textures which can be reasonably represented.

It is instructive to examine the Multi-Level Logistic (MLL) model [4][7], which is a clustering-based GRF model that has been shown to have good results in image recognition and segmentation applications. The clique potential functions are defined according to the following formula:

$$V_c(x) = \begin{cases} -\beta & , \text{ if all } x \text{ in } c \text{ are equal} \\ \beta & , \text{ otherwise} \end{cases}$$

where β represents the specific parameter for that type of clique, as shown below in Figure 1.

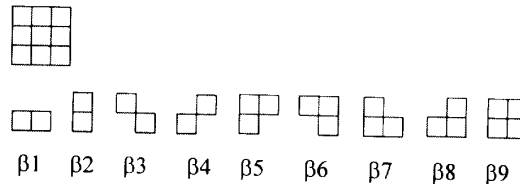


Figure 1. MLL Parameters

Single pixel clique parameters are specified strictly by the gray-level value of the pixel, so that

$$V_c(x) = \alpha_k, \text{ for } X=k.$$

In the MLL model, the α parameters control the marginal distribution of the pixel variables (clustering with respect to the center gray level value only), while all other parameters control the size and direction of the texture clustering.

3. ADAPTIVE CLIQUE POTENTIAL FUNCTIONS

We begin by analyzing the characteristics of the MLL model's clique potential functions, assuming for simplicity a 64x64 image with 8-level (3 bit) grey-level resolution. Since the basis for the potentials is the identity function, its value given a specific clique realization is based solely on whether or not all the pixels in the clique are identical. By definition, therefore, the model cannot recognize the difference between a clique composed of pixels of similar values, say 6 and 7, and a clique composed of pixels of nearly opposite values, say 0 and 7. Pixel similarities are completely ignored if there is no exact match.

A clique potential function which recognized non-identical similarities (for example, one which replaced the identity function with a monotonically decreasing linear or exponential potential function) should be more robust and powerful than the base MLL functions. Controlling the shape and characteristics of the potential functions themselves is as significant as controlling the amount and direction of clustering (i.e., the functions' amplitude). The model utilized herein is based on an exponential potential function, incorporating two additional parameters into the feature vector to control the shaping of the function curve.

For the purposes of determining the advantage of such an adaptive system, a relatively simple model is

used. The same basic feature vector as the MLL model is used, with two additional parameters as follows:

$$\theta = [\beta_1 \beta_2 \beta_3 \beta_4 A \tau]$$

where $\beta_1, \beta_2, \beta_3,$ and β_4 represent horizontal, vertical, 45°, and 135° clustering parameters, respectively, A represents function amplitude, and τ represents the decay constant of the exponential function. Single pixel cliques are assigned an α parameter of 1, and cliques with three or more pixels are not considered for computational simplicity. The potential function is given as:

$$V_c(x) = \beta \left(A - A e^{-\frac{|\Delta x_{ij}|}{\tau}} + 1 \right)$$

where $\Delta(x_{ij})$ is the difference between gray-level values of clique pixels, β represents the appropriate directional parameter, and A and τ are defined as stated above.

Figure 2 shows the difference between the potential functions of the standard MLL model and the new adaptive model defined here. Note that by setting $A=2$ and $\tau \ll 0.1$, the new model can be made to approximate the MLL model to within any degree of accuracy. In addition, as A and τ are made large, the functions become increasingly linear in nature.

4. RESULTS

Several texture simulations were conducted as a test of the new approach, using the Gibbs sampler algorithm [3]. The algorithm begins with a random image and changes pixels at random locations within the image, based on the GD probability function for that pixel's neighbors. The algorithm normally converges within 50 iterations, with 1 iteration equivalent to N^2 probability decisions over an $N \times N$ image.

Figure 3 illustrates some of the texture realizations corresponding to a single set of directional clustering parameters, in this case a +1.0 horizontal parameter with -1.0 vertical and diagonal parameters. The simulation at top uses the MLL model approach, while the other six use the new model with varying decay constants and amplitude parameters. The effect of a gradual potential function as opposed to the identity function is evident in a number of physical characteristics that can be observed: (1) increased contrast in images with larger decay constants; (2) increased smoothness in images having more nearly linear potential functions; and (3) increased image entropy corresponding to images with large decay constants and smaller amplitude parameters.

The concept of the adaptive potential function, used within the framework of the Gibbsian probability model, can also be expanded in scope to further enlarge the types of textures which may be modeled using the MRF approach. This may be accomplished by adding additional function-shaping parameters to control such potential function characteristics as maximum and minimum locations, function symmetry, symmetry between clique types, or even point-by-point function definitions if desired. Each of these alterations would increase the number of physical characteristics which are describable under the model.

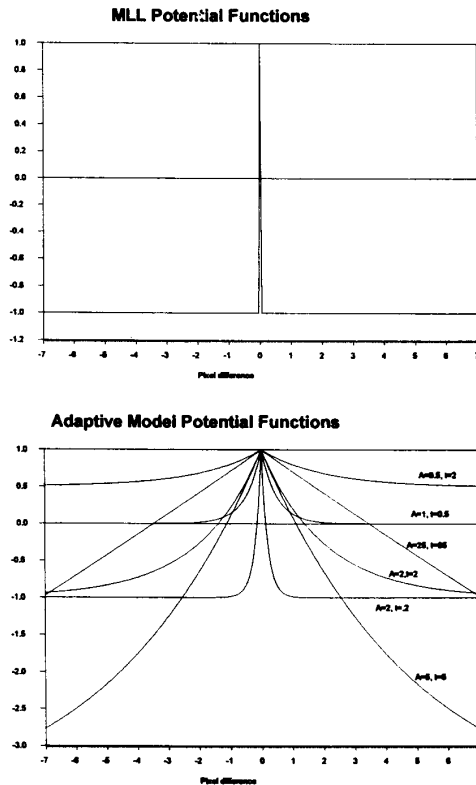


Figure 2. MLL vs. Adaptive potential functions

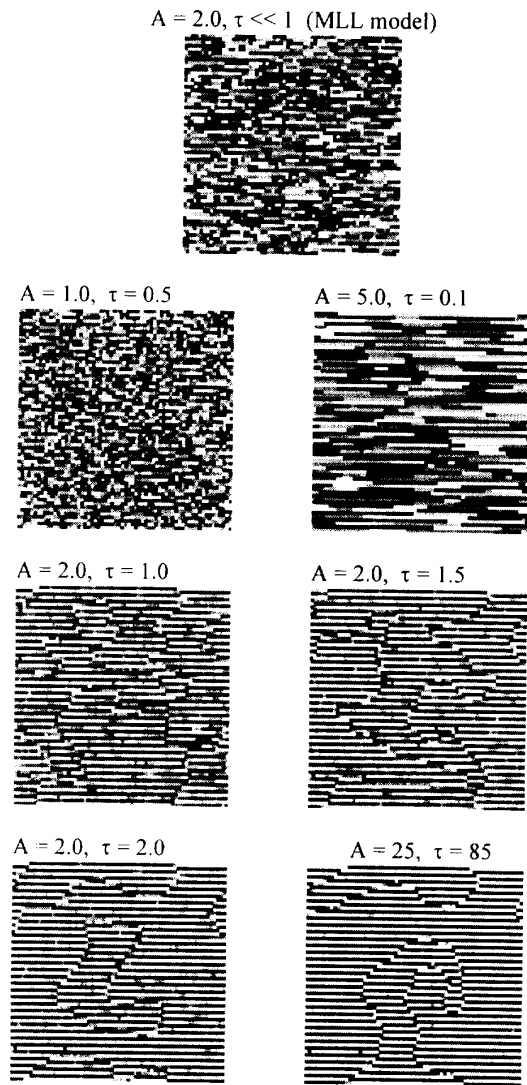


Figure 3. Simulated images with clustering parameters
 [+1.0 -1.0 -1.0 -1.0]

REFERENCES

- [1] G. R. Cross and A. K. Jain, "Markov Random Field texture models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI-5 No. 1, pp 25-39, 1983.
- [2] J. Besag, "Nearest-neighbor systems and the autologistic model for binary data," *J. Royal Statist. Soc., series B*, Vol. 34, pp75-83, 1972.
- [3] S. Geman and D. Geman, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI-6 No. 6, pp 721-741, 1984.
- [4] H. Derin and H. Elliott, "Modeling and Segmentation of noisy and textured images using Gibbs Random Fields," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9 No. 1, pp 39-55, 1987.
- [5] F. Cohen and D. Cooper, "Simple parallel hierarchical and relaxation algorithms for segmenting noncausal Markovian Random Fields," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9 No. 2, pp 195-219, 1987.
- [6] R. Chellappa and S. Chatterjee, "Classification of textures using Gaussian Markov Random Fields," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, Vol. ASSP-33 No. 4, pp 959-963, 1985.
- [7] S. Lakshmanan and H. Derin, "Simultaneous parameter estimation and segmentation of Gibbs Random Fields using simulated annealing," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 11 No. 8, pp 799-813, 1989.