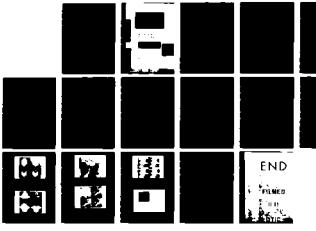
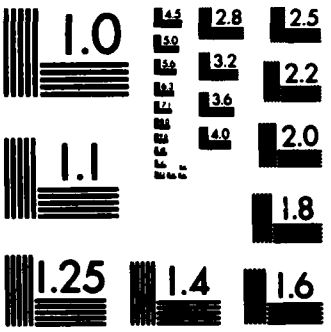
AD-A133 406 APPLICATION OF THE GIBBS DISTRIBUTION TO IMAGE SEGMENTATION(U) MASSACHUSETTS UNIV AMHERST DEPT OF ELECTRICAL AND COMPUTER EN.. H ELLIOTT ET AL. AUG 83 1/1

UNCLASSIFIED UMASS-ECE-AUG83-2 N00014-83-K-0059 F/G 12/1 NL





Communications and Control Systems

(12)

Application of the Gibbs Distribution to Image Segmentation

by

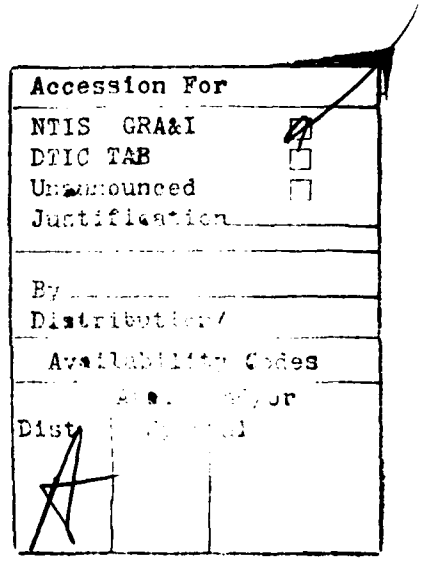
H. Elliott[1], H. Derin[1], R. Cristi[1], D. Geman[2]

[1]Department of Electrical and Computer Engineering
[2]Department of Mathematics and Statistics
University of Massachusetts
Amherst, MA 01003

Univ. of Mass. Tech. Rep. #UMASS-ECE-AUG83-2
University of Massachusetts
August 1983

ABSTRACT

This paper describes a new statistical approach to image segmentation. Making use of Gibbs distribution models of Markov random fields a dynamic programming based segmentation algorithm is developed. A number of examples are presented which give an indication of the potential of this approach.



I. INTRODUCTION

This report presents a new statistical approach to the image segmentation problem. By modelling image data as a Markov random field characterized by a Gibbs distribution, a dynamic programming algorithm is developed. The primary contribution of the paper is this new near optimal method for processing scenes described by the non-causal Gibbs model.

Image segmentation, the process of grouping image data into regions with similar features is a component process in image understanding systems and also serves as a tool for image enhancement. As such it has received considerable attention in the literature. Many techniques work well on noise free images with slow spatial variation in intensity. However, when the data is noisy or textured, these algorithms become less reliable. In this case it can be advantageous to statistically model the noise and any texture which is random in nature. Furthermore, one must also take advantage of two-dimensional spatial ergoticity to average the effects of noise. If a region is spatially ergodic then a pixel and its neighbors will have similar statistical properties. In its simpler forms, the Gibbs model can be used to exploit this type of spatial continuity, and this is its primary role in the segmentation algorithm.

Use of the Gibbs distribution dates back to the work of Ising [1] in 1925 who modelled molecular interaction in ferromagnetic materials, and it has received considerable attention in both the statistical mechanics and statistics literature [2]. However, only recently have attempts been made to apply it to problems in image processing. In [4], the autobinomial form of the Gibbs distribution was used to model texture. The algorithm in [5] segments textured images hierarchically operating on successively smaller blocks and uses Gibbs distributions to model texture. To our knowledge, the work in [6] represents the first application of the Gibbs model to Image Segmentation. The algorithm in [6] is highly parallel in nature with the flavor of a 'relaxation' algorithm and requires a number of iterative passes on the image data. The algorithm we propose processes the data in a raster scan fashion, and only requires a single scan of the data. It will be important to further study the trade-offs in the various algorithms.

The report is organized as follows. Section II defines the segmentation problem in a statistical framework, introduces some notation, and presents some background on Markov random fields. Section III then presents the dynamic programming algorithm in detail for the case of segmenting images consisting of uniform intensity regions in high levels of additive white Gaussian noise. Section IV presents results of applying the algorithms to some experimentally generated images consistant with this model as well as some synthetic aperture radar images which are clearly inconsistent with the assumed model. These results clearly demonstrate the

applicability of the technique to realistic data as well as the robustness of the algorithm with respect to modelling assumptions. In Section V, some comments and concluding remarks are given, and extensions to this work which are in progress are briefly outlined.

II. PROBLEM FORMULATION AND MATHEMATICAL BACKGROUND

Preliminary Definitions

Let a class of scenes be characterized by a discrete finite random field $X = [X_{ij}]$ of size $(N_1 X N_2)$, and let a realization of this field or a specific scene be represented by the matrix $x = [x_{ij}]$. It will be assumed that each pixel (i, j) can belong to one of M distinct region types and that $X_{ij} = m$ if pixel (i, j) is a member of region m, $m \in [1, 2, \ldots, M]$. Associated with a specific scene is a set of K, $(N_1 X N_2)$ observation matrices $\bar{y} = \{y^k\}_1^K$, $y^k = [y_{ij}^k]$. For simplicity of exposition, we will assume K = 1 and simply define y^1 to be $y = [y_{ij}]$. However, it should be pointed out that the algorithms presented below extend trivially to the case of multiple observations such as with Landsat data. Since regions can be textured or contain observation noise the range space of y_{ij} is larger than that of X_{ij}. Thus y will be assumed to be a realization of a real valued random field $Y = [Y_{ij}]$. The general model which can we will employ is

$$Y_{ij} = F_{ij}(X_{ij}) + W_{ij} \tag{1}$$

The field W_{ij} is a random noise field, and the mapping F_{ij} can be used to characterize texture models. In the case which will be described in most detail, a region will be characterized by constant intensity so that

$$F_{ij}(X_{ij}) = r_m \text{ if } X_{ij} = m \tag{2}$$

Furthermore, N_{ij} will be assumed to be a white Gaussian field with zero mean and variance σ^2, i.e.

$$W_{ij} \sim \mathcal{N}(o, \sigma^2) \tag{3}$$

MAP Segmentation

The segmentation problem can now be simply stated as follows. Given the observation matrix $y = [y_{ij}]$ find an estimate $\tilde{x} = [\tilde{x}_{ij}]$ of the scene realization $x = [x_{ij}]$. The algorithm presented below attempts to maximize the posterior probability or likelihood of $\tilde{x}$ given y. In particular if P(.) is an appropriate probability measure, then one would like to find the estimate $\tilde{x}$ which maximizes $P(X = \tilde{x} \mid Y = y)$. Using Bayes rule

$$P(X=\tilde{x} \mid Y=y) = \frac{P(Y=y \mid X=\tilde{x})\ P(X=\tilde{x})}{P(Y=y)} \tag{4}$$

Since P(Y=y) is independent of the estimate $\tilde{x}$, we can equivalently maximize

$$P(X=\tilde{x},\ Y=y) = P(Y=y \mid X = \tilde{x})\ P\ (X = \tilde{x}) \tag{5}$$

or

$$\ln P\ (X=\tilde{x},\ Y=y) = \ln P(Y=y \mid X=\tilde{x}) + \ln P(X=\tilde{x}) \tag{6}$$

The dynamic programming algorithm presented in the next section is an approximation to one which guarantees finding the $\tilde{x}$ which maximizes (6). It should be pointed out that the difficulty in maximizing (6) is that it is a joint log-likelihood for all the image data. It does not simply describe the likelihood of a single pixel. In particular the maximizing $\tilde{x}$, is one of $M^{N_1 N_2}$ possibilities.

Markov Random Fields and the Gibbs Distribution

Obviously, any processing algorithm for maximization of (6) will depend critically on the form of P(X=x) and P(Y=y | X=x). In this subsection, these measures are defined in the case where X is a Markov random field characterized by a Gibbs distribution, and Y is given by (1) - (3).

To begin it will be helpful to introduce some additional notation. Let L de defined as a $N_1 X N_2$ lattice characterizing all pixel locations in the scene, i.e.

$$L = \{(i,j): 1 \leq i \leq N_1, \quad 1 \leq j \leq N_2\} \tag{7}$$

Next let η_{ij} define a set of neighbor sites for the pixel (i, j) but excluding (i, j) itself. Two simple neighborhoods are depicted in figure 1. These are:

$$\eta_{ij}^1 = \{(1,m): 0 < (i-1)^2 + (j-m)^2 \leq 1\} \tag{8}$$

$$\eta_{ij}^2 = \{(1, m): 0 < (i-1)^2 + (j-m)^2 \leq 2\} \tag{9}$$

Finally, define η_L to be the collection of all neighborhoods in L, or a neighborhood system. Then η_L^1 and η_L^2 characterize the collection of all neighborhoods η_{ij}^1 and η_{ij}^2, i.e.

$$\eta_L^1 = \{\eta_{ij}^1 : (i,j) \in L\} \tag{10}$$

$$\eta_L^2 = \{\eta_{ij}^2 : (i,j) \in L\} \tag{11}$$

Given this notation, a Markov random field can then be defined as

<u>Definition 1:</u>

A random field on a lattice L is a markov random field with respect to a neighborhood system η_L if and only if

$$P(X_{ij} = x_{ij} \mid X_{lm} = x_{lm}, (l, m) \in L, (l,m) \neq (i,j)$$

$$= P(X_{ij} = x_{ij} \mid X_{lm} = x_{lm}, (l,m) \in \eta_{ij}) \tag{12}$$

If in addition to (12) $P(X = x) > 0$ for all realizations x, then X can be characterized by a Gibbs distribution defined on the neighborhood system η_L[2], [3]. In order to define the Gibbs distribution it is necessary to introduce the notion of the cliques of a neighborhood system. Simply, a

clique is any set of pixel locations for which any two are neighbors of each other. Figure 2 show the types of cliques found for the η_L^1 and η_L^2 neighborhood systems. Let $C(\eta_L)$ be the collection of cliques for the neighborhood system η_L. The Gibbs distribution can now be defined as follows

Definition 2:

The Gibbs probability distribution has the form

$$P(X = x) = \frac{1}{Z} e^{U(x)} \tag{13}$$

$$U(x) = \sum_{c \in C(\eta_L)} V_c(x) \tag{14}$$

$\triangleq$ energy function

$V_c(x) \triangleq$ potential associated with clique C

$$Z = \sum_x e^{U(x)} \tag{15}$$

$\triangleq$ partition function

As can be seen from the definition Z is simply a normalizing constant so that the sum of the probabilities of all realizations, x, add to one. Thus the key functions in determining the properties of the distribution are the potential functions $V_c(x)$. The only limitation on $V_c(x)$ is that it only depend on the values of the pixels in clique c. We will consider homogeneous fields where the form of $V_c(x)$ is fixed by the structure of the clique c and not its location in the lattice. For the segmentation algorithm discussed in the next section, the potential functions were chosen to exploit spatial continuity, and for simplicity of calculation. We have modelled the region clustering as being characterized by a Markov random field over the neighborhood system η_L^2. The most general form we use for the potential functions $V_c(x)$ are given in Table 1.

No.	Clique Structure		Potential
1	•	[(i,j)]	$V_c(x) = \alpha_m$ if $x_{ij} = m$
2	••	[(i,j),(i, j+1)]	$V_c(x) = \beta_1$ if $x_{ij} = x_{i\,j+1}$; $-\beta_1$ otherwise
3	• •	[(i,j),(i-1,j)]	$V_c(x) = \beta_2$ if $x_{ij} = x_{i+1,\,j}$; $-\beta_2$ otherwise
4	• ••	[(i,j),(i-1,j+1)]	$V_c(x) = \beta_3$ if $x_{ij} = x_{i-1,j+1}$; $-\beta_3$ otherwise
5	•• •	[(i,j),(i+1, j+1)]	$V_c(x) = \beta_4$ if $x_{ij} = x_{i+1,\,j+1}$; $-\beta_4$ otherwise
6	•• ••	[(i,j), (i+1, j), (i, j+1)]	$V_c(x) = \gamma_1$ if all x_{ij} in c are equal; $-\gamma_1$ otherwise
7	• ••	[(i,j),(i-1,j), (i,j+1)]	$V_c(x) = \gamma_2$ if all x_{ij} in c are equal; $-\gamma_2$ otherwise
8	•• •	[(i,j),(i,j+1),(i+1,j+1)]	$V_c(x) = \gamma_3$ if all x_{ij} in c are equal; $-\gamma_3$ otherwise
9	• ••	[(i,j),(i,j+1),(i-1,j+1)]	$V_c(x) = \gamma_4$ if all x_{ij} in c are equal; $-\gamma_4$ otherwise
10	•• ••	[(i,j),(i-1,j),(i,j+1), (i,j+1)]	$V_c(x) = \sigma_1$ if all x_{ij} in c are equal; $-\sigma_1$ otherwise

TABLE 1
Potential Functions for Segmentation Algorithm

The parameters α_i, β_i, γ_i, σ_i need to be estimated for certain classes of scenes. Although this is a difficult problem, there are methods in the literature for estimating these parameters, see [3] - [5]. However, we are not using the Gibbs distribution to model textures or detailed shape. We just want to model the fact that regions are clusters of pixels, i.e. spatial continuity within regions. For the examples in section IV, we use only cliques 1-5, and in many cases just cliques 2 and 3, e.g. $\gamma_i=\sigma_i=0$. The nonzero parameters were chosen by trial and error. Although for a given signal to noise rati the algorithm was relatively insensitive to the choice of these values, as the signal to noise ratio changed we found it necessary to modify the values. We are presently studying this phenomena more carefully as well as determining new schemes for estimating these parameters for the context in which they are being used.

Finally, since a particular realization of the Gibbs field, x, assigns each pixel to one of the M region types, using the Gaussian noise model an appropriate form for $P(Y=y \mid X=x)$ is

$$P(Y=y \mid X=x) = \prod_{m=1}^{M} \prod_{(i,j)\in S_m} \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left(\frac{1}{2\sigma^2}\,(y_{ij}-r_m)^2\right) \qquad (16)$$

$$S_m = \{\, (i,j):\ X_{ij} = m \,\} \qquad (17)$$

III. NEAR OPTIMA MAP SEGMENTATION

In this section, an optimal algorithm is posed for processing images consisting of a few rows. The complete near-optimal algorithm is then obtained by applying this optimal processor on overlapping strips of the larger $N_1 X N_2$ image. This algorithm will be near optimal when correlation between the random variables in X_{ij} drops rapidly as their vertical distance increases. This assumption appears reasonable in the sense that it can often be shown for one dimensional Markov chains. However, calculation of correlations in a Markov random field is extremely difficult even for the simplest case (Ising Model [2]), and is an unresolved problem in the statistics literature.

First consider the problem of segmenting a DXN_2 image, $D << N_1$. Let $\ell(.)$ denote a log-likelihood function. Using the Gibbs likelihood and the conditional data likelihoods derived in the last subsection of section II, we can write the following expression for the joint log-likelihood of the image data y and a realization x of the Gibbs field X:

$$\ell(Y=y,\ X=x) = \ell(Y=y \mid X=x) + \ell(X=x) \tag{18}$$

$$\ell(X=x) = -\ln Z + \sum_{c \varepsilon C_L(\eta^2)} V_c(x) \tag{19}$$

$$\ell(Y=y \mid X=x) = \frac{-DN_2}{2} \ln(2\pi\sigma^2) - \sum_{m=1}^{M} \sum_{(i,j)\varepsilon S_m} \frac{1}{2\sigma^2} (y_{ij}-r_m)^2 \tag{20}$$

Observe that we can calculate (18) recursively as follows

$$\ell_{N_1} = \ell(Y=y,\ X=x)$$

$$\ell_o = \frac{-DN_1}{2} \ln(2\pi\sigma^2) - \ln Z$$

$$\ell_k = \ell_{k-1} + \sum_{c \varepsilon C^{k-1,\,k}} V_c(x) - \sum_{m=1}^{M} \sum_{(i,j)\varepsilon S_m^k} \frac{1}{2\sigma^2} (y_{ij}-r_m)^2$$

$C^{k-1,\,k} = \{c \in C_{\backslash}{}^{2}$: c contains only pixels in column k or only pixels in columns k - 1 and k $\}$

$$S_m^k = \{ (i,j): x_{ij} = m,\ j = k\} \quad 1 \leq i \leq W$$

This recursion in conjunction with the principle of optimality allows formulation of a forward dynamic programming algorithm [7] for finding $\tilde{x}$. The state space associated with the dynamic programming algorithm has dimension M^D since there are M^D possible segmentations of each column of the DXN_2 scene. This implies that the algorithm would have N_2 iterations with on the order of M^{2D} calculations during each iteration. Thus this algorithm may only be computationally tractable for small values of M and D, e.g. $2 \leq M, D \leq 4$. Although using the technique described below, full size images can be processed with reasonable computation speeds, we do only recommend the algorithm be implemented for segmenting images into at most M=4 region types. We are presently working on a method which allows this algorithm to be applied iteratively to segment images for which $M > 4$. More will be said regarding this in Section V. Since this is a standard dynamic programming application, details will not be given. However, two important remarks are in order.

Remark 1

Observe that the value of $_o$ is independent of any segmentation $\tilde{x}$, and hence the algorithm can be initialized by setting $_o = 0$. In particular, there is no need to undertake the difficult task of calculating the partition function Z.

Remark 2

In order to calculate potentials for all cliques in column one and row one of the (DXN_2) image, it is necessary to assume a segmentation for a ficticious column zero and row zero. This corresponds to a boundary condition for the Markov random field X. An appropriate choice for these boundary conditions will be discussed below.

In order to use the dynamic programming algorithm described above which is capable of optimally processing a DXN_2 strip of an N_1XN_2 image $N_1 >> D$, we will assume the random variables X_{ij} and $X_{i+D,\,j}$ for all (i,j) to have negligible correlation or covariance. Thus, the segmentation for row

i should be negligibly impacted by the segmentation of row i+D. In view of this, the complete segmentation procedure is described by the following algorithm.

Segmentation Algorithm

Step 0) Choose a value for D, $2 \leq D \leq 4$,

Step 1) Set I = 1

Step 2) Apply Dynamic Programming Algorithm to rows I through I+D-1

Step 3) Store the segmentation for row I

Step 4) Set I = I+1

Step 5) If $I \leq N_1 - D + 1$, go to 2

Step 6) D = D-1

Step 7) If $D \geq 1$, go to 2

Step 8) Stop

To summarize, the dynamic programming algorithm is applied to overlapping image strips of width D, but only the segmentation of the first row of that strip is used. For example, the processing of rows 1 through D yields a segmentation for row 1, and the processing of rows 2 through D+1 yields a segmentation of row 2. Under the correlation assumption above, this algorithm is near optimal since the data in row I + D will have little impact on the segmentation of row I.

As pointed out in Remark 2 above, the strip dynamic programming algorithm requires a fixed segmentation or boundary condition for row I-1 and column 0. For I = 1, (i.e. row 0) and column zero, we arbitrarily assume all pixels to be backround pixels, i.e. $\tilde{x}_{i0} = \tilde{x}_{0i} = M_b$ where $M_b \in [1, 2, \ldots, M]$ and M_b is the assumed background intensity. Although this is somewhat arbitrary, if the correlation assumption holds both vertically and horizontally, it will have negligible impact on the segmentation of the scene for $i > D$ and $j > D$. For $I > 1$ we use the fixed segmentation of the previous row to initialize the strip processor.

IV. EXAMPLES

In this section, some examples are presented which are representative of the performance of the algorithm and which highlight some of its properties. To begin, conside: figures 3 and 4. They show the results of applying the algoirthm to (64 X 64) test images consisting of an object either an ellipse or a diamond, on a background. Thus, M=2. For these examples, the images were corrupted by additive white zero mean Gaussian noise fields with variances such that the signal to noise ratio S/N=2 where we define

$$S/N = \frac{|r_1 - r_2|}{\sigma}$$

The values of r_1, r_2, and σ were assumed known and the algorithm was applied with D=4 and all Gibbs parameters set to zero except the β_i. For the ellipse of Figure 3 $\beta_i = 0.2$ while for the diamond of Figure 4 $\beta_i = 0.15$ i=1,2,3,4. The top row of each figure from left to right show the original and noise corrupted images, while the bottom row from left to right show the Gibbs algorithm segmentation, and the result of filtering the segmented image through a (3 X 3) median filter. As can be seen by comparison of the two figures, the algorithm performs well at this signal to noise ratio but as the magnitude of the Gibbs parameters decreases, the algorithm becomes subject to more single pixel errors. This is consistent with expectation since in this case the Gibbs distribution is weighted less relative to the data terms in the likelihood function. Thus there is less emphasis on spatial continuity in the likelihood.

Figures 5 - 7 show the result of applying the algorithm to an ellipse in noise such that S/N = 1 using D=2,3,4. Although performance is best for D=4 there is surprisingly little degredation in performance as D decreases. In all three figures β_i=0.175, i=1,2,3,4 and all other Gibbs parameters were zero.

Figures 8 and 9 show the results of applying the algorithm to a diamond in noise such that S/N=1. In both figures D=4, however by comparison of figures 8 and 9, one can see the improvement in performance which can be obtained by assuming some additional prior knowledge of shape. For Figure 8 β_i=0.2 i=1,2,3,4 and all other Gibbs parameters were zero. For Figure 9, realizing that the object had diagonal edges, the diagonal cliques were emphasized relative to the horizontal and verticle cliques, i.e. $\beta_1=\beta_2$=0.15 and $\beta_3=\beta_4$=0.25 while all other parameters were zero. For

the latter case, the shape of the diamond was more obvious in the segmentation.

For Figures 10 - 17, he algorithm was applied to two test images, each containing M=4 region types. For Figures 10 - 15 D=2 while for Figures 16 and 17 D=3. We usel D=2 to keep computation times down, however as can be seen from Figures 12 - 15, while performance at S/N=1.5 is reasonable, performance at S/N=1 is not. For M=4, we define the signal to noise ration as

$$S/N = \min_{i \neq j} \frac{|r_i - r_j|}{\sigma}$$

Increasing to D=3 did considerably improve performance at S/N=1, however CPU time for this case on a VAX 11/780 was 60 minutes as compared to 4 minutes for M=4 and D=2 and 7 minutes for M=2 and D=4. This is our motivation for going to the hierarchical scheme for handling multi-region images ($M \geq 3$) which will be briefly outlined in Section IV below. We also point out that we are rewriting our code to make use of look-up tables and anticipate a factor of 10 or better improvement in computation speed.

Finally, for Figures 19 - 21, the algorithm was applied to (64 X 64) sections of the synthetic aperture radar image shown in Figure 18. Figure 19 shows the result of applying algorithm to a small bay, while in Figure 20, there is a river with some bridges, and in Figure 21 there is a boat. In all cases, we assumed M=2, and obtained r_1, r_2 and σ by applying the method of moments under the assumption that the $(64)^2$ pixels in the scene where samples of a random variable characterized by a distribution which was a mixture of two Gaussian distributions. Each picture shows the original scene and three segmentations corresponding to different Gibbs parameters. For Figure 19, proceeding clockwise around the picture we had α_1=-.05 (white), α_2=.05 (black) and all β_i=0.125, $\alpha_1=\alpha_2=0$ and all β_i=0.125, $\alpha_1=\alpha_2$ and all β_i=0.15. For Figure 20, we had, α_1=.05 (white), α_2=-.05 (black), and all β_i=0.1, α_1=.05, α_2=-.05, and all β_i=0.125, $\alpha_1=\alpha_2=0$ and all β_i=0. For Figure 21 we had, $\alpha_1=\alpha_2=0$ and all β_i=0.2, $\alpha_1=\alpha_2=0$ and all β_i=0.35, α_1=0.1, α_2=-0.1 and all β_i=0.35.

V. CONCLUDING REMARKS

This report has presented a new approach to segmentation of noisy images. It uses the Gibbs distribution to model spatial continuity or clustering properties of regions. The algorithm is recursive in nature, requires a single pass on the data, and works well at low signal to noise ratios.

Presently, two extensions to this algorithm are being developed. The first allows computationally efficient segmentation of images with more than two region types. If there are M region types, the two region version of the algorithm is applied M-1 times. Thus computation time grows linearly with the number of regions. To see how this can be done consider the case where M=4 and for convenience, the regions have been labelled so that $r_1 < r_2 < r_3 < r_4$. The two region algorithm can first be applied using region means of r_2 and r_3. Since $r_1 < r_2$ and $r_4 > r_3$ the result will be a segmentation grouping regions 1 and 2 together, and regions 3 and 4 together. Next the two region algorithm is applied to the image using region means of r_1 and r_2. However, only pixels classified as being in the region with mean r_2 during the first pass are classified in the second pass. Those classified as being in the region with mean r_3 during the first pass are ignored. This yields a three class segmentation into regions 1, 2 and the combined regions 3 and 4. To separate regions 3 and 4 a third pass is made using region means r_3 and r_4 but ignoring those pixels assigned to regions 1 and 2 during the first pass. We feel that this hierarchical algorithm will have little effect on the near optimality of the overall approach, however we have no examples to show at this time.

The second extension is to textured images. In this case, instead of modelling regions as being of constant intensity but imbedded in uncorrelated noise, we assume textured regions are realizations of a second Gibbs distribution. In this case, we are developing algorithms similar to the one given above, but with the Gaussian (quadratic) data term in the log-likelihood replaced by terms associated with the second Gibbs distribution characterizing texture. In this context, we are also developing new methods for Gibbs parameter estimation in texture images. The main difference between our work on texture modelling and that in [4], [5] is that we are experimenting with different types of potential functions $V_c(x)$.

In conclusion, we feel that the applications of Gibbs models to problems in image processing and image understanding are just beginning to

emerge. It is potentially a very powerful tool but there are still many problems to be resolved.

REFERENCES

[1] E. Ising, Zeitschrift hysik, Vol. 31, p.253, 1925.

[2] R. Kinderman, J.L. Snell, Contempory Mathematics: Markov Fields and their Applications, Vol. 1, American Mathematical Society Providence, R.I., 1980.

[3] J. Besag, 'Spatial Interaction and the Statistical Analysis of Lattice Systems,' J. Royal Statistical Society, Series B, Vol. 36, pp 192 - 236, 1974.

[4] G.R. Cross and A.K. Jain, 'Markov Random Field Texture Models', IEEE Trans on PAMI, Vol. PAMI-5, No. 1, pp. 25-39, 1983.

[5] F. Cohen, Parallel Adaptive Heirarchical Segmentation of Textured Images Using Noncausal Markovian Fields, Ph.D Dissertation, Brown University, August 1983.

[6] D. Geman and S. Geman, 'Stochastic Relaxation, Gibbs Distributions, and the Bauesian Restoration of Images,' August 1983 (In preparation).

[7] R. Bellman, Dynamic Programming, Princeton University Press, Princeton N.J., 1957.

[8] K. Abend, T.J. Harley, and L.N. Kanal, 'Classification of Binary Random Patterns,' IEEE Trans. Inform. Theo., Vol. IT-11, pp 538 - 544, October 1965.

[9] L.N. Kanal, 'Markov Mesh Models,' Image Modeling, Academic Press, 1980.

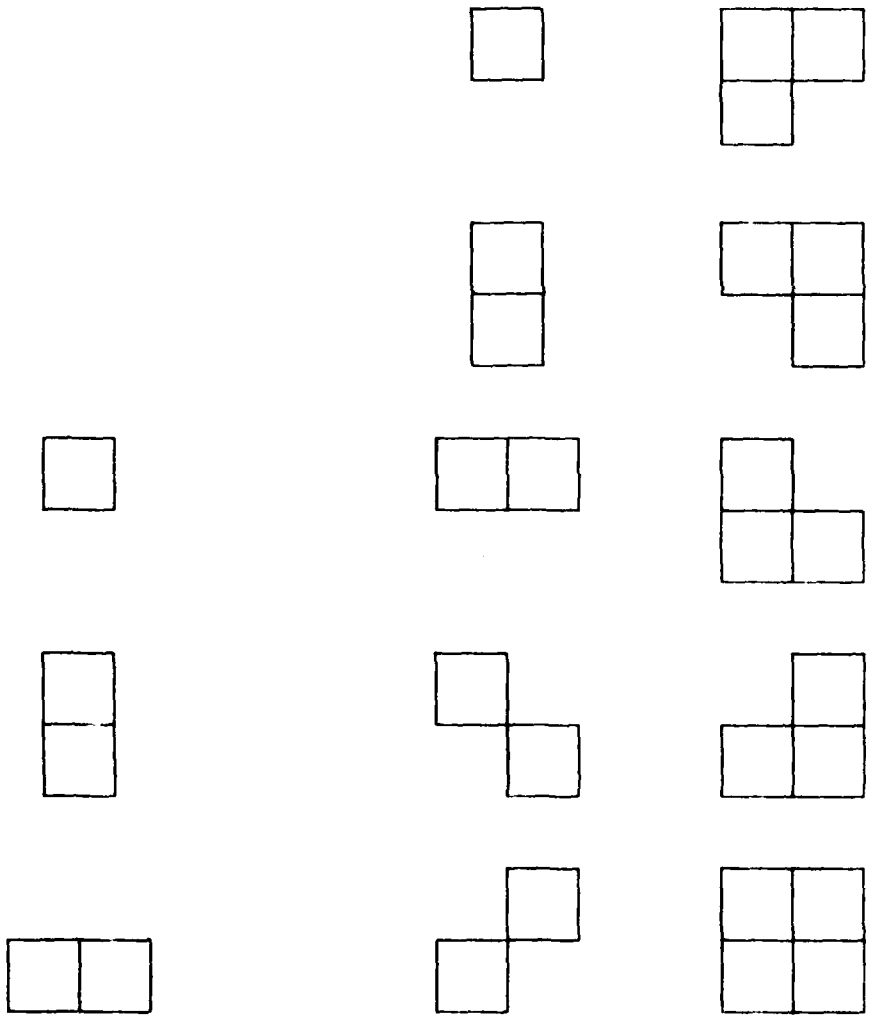
[10] D.K. Pickard, 'A Curious Binary Lattice Process,' J. Appl. Prob., Vol. 14, pp. 717 - 731, 1977.

[11] D.B. Cooper, H. Elliott, F. Cohen, L. Reiss and P. Symosek, 'Stochatic Boundary Estimation and Object Recognition,' Computer Graphics and Image Processing, Vol. 10, pp. 326 - 355, April 1980.

[12] F.R. Hansen and H. Elliott, 'Image Segmentation Using Simple Markov Random Field Models,' Computer Graphics and Image Processing, Vol. 20, pp 101 - 132, 1982.

(a) (b)

Figure 1 - Neighborhoods η_{ij}^1 (a) and η_{ij}^2 (b)



(a) (b)

Figure 2 - Cliques for neighborhood systems η_L^1 (a) and η_L^2 (b)

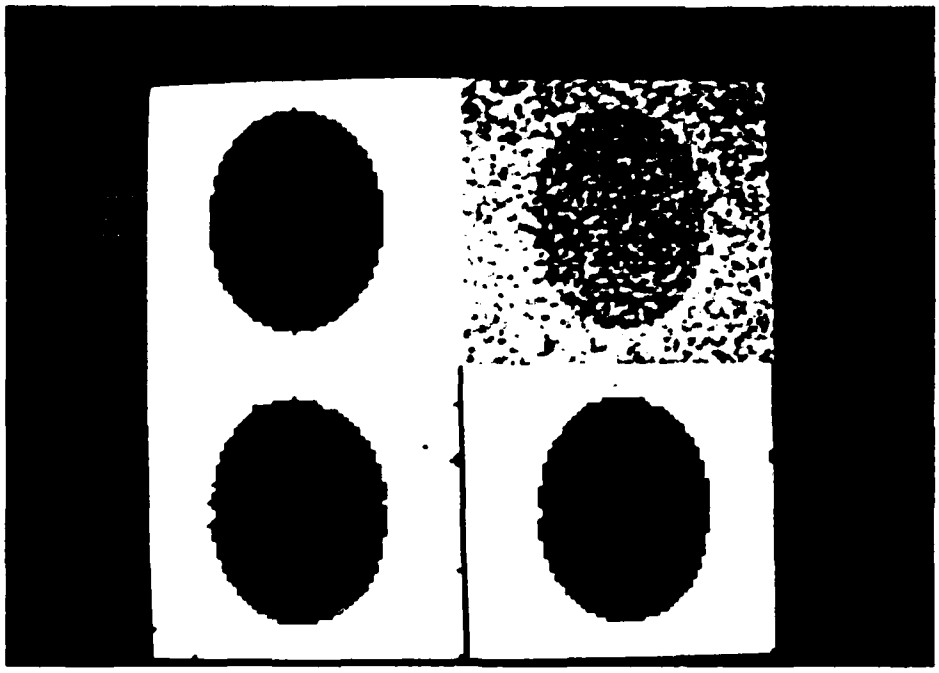


Figure 3 - Segmentation of an ellipse with S/N=2

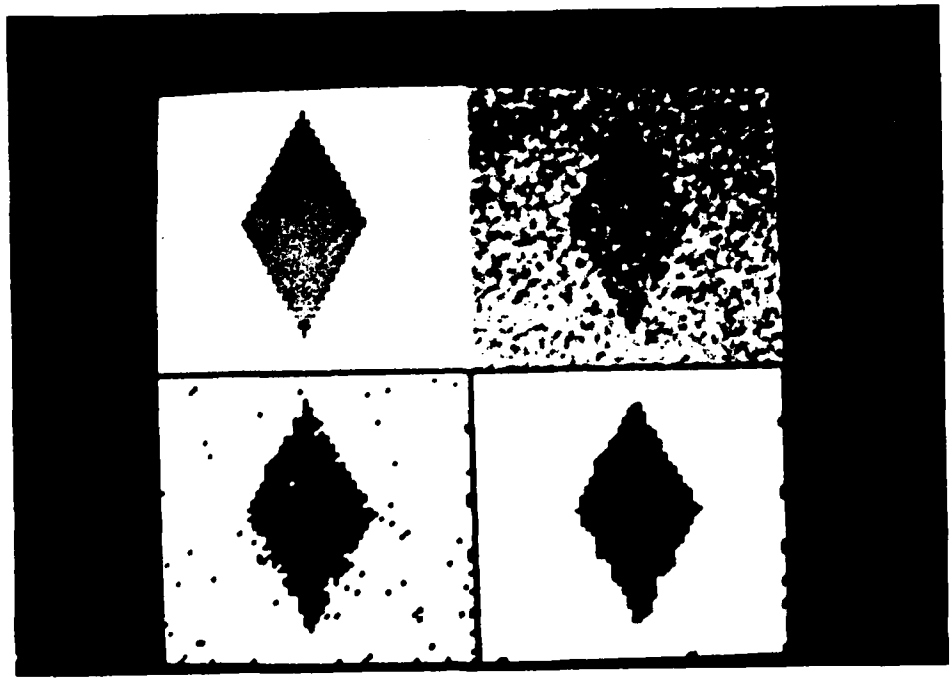
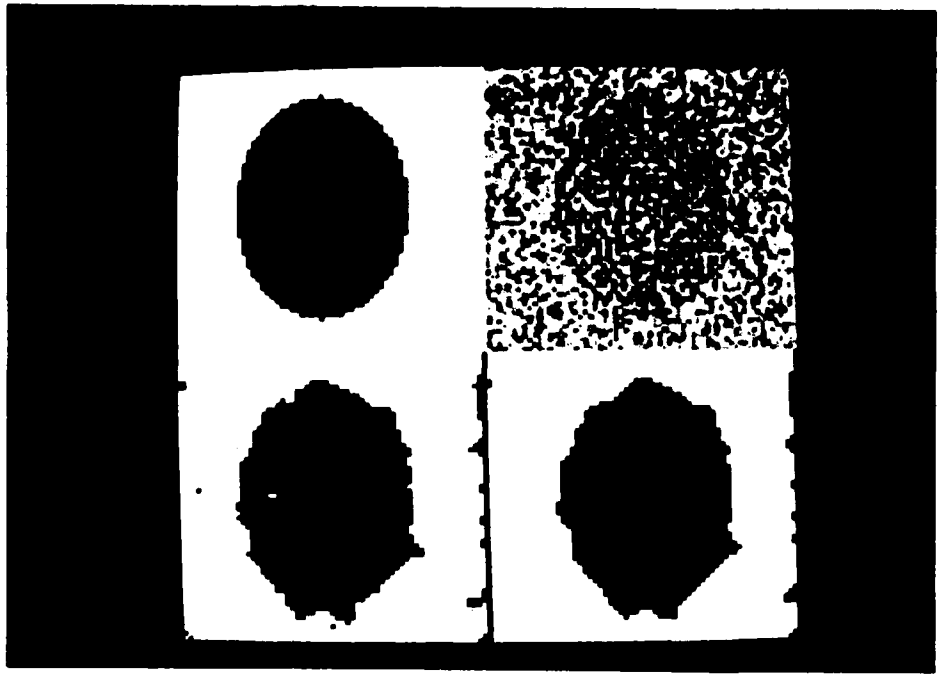


Figure 4 - Segmentation of a diamond with S/N=2

Figure 5 - Segmentation of an ellipse with S/N=1 and D=2



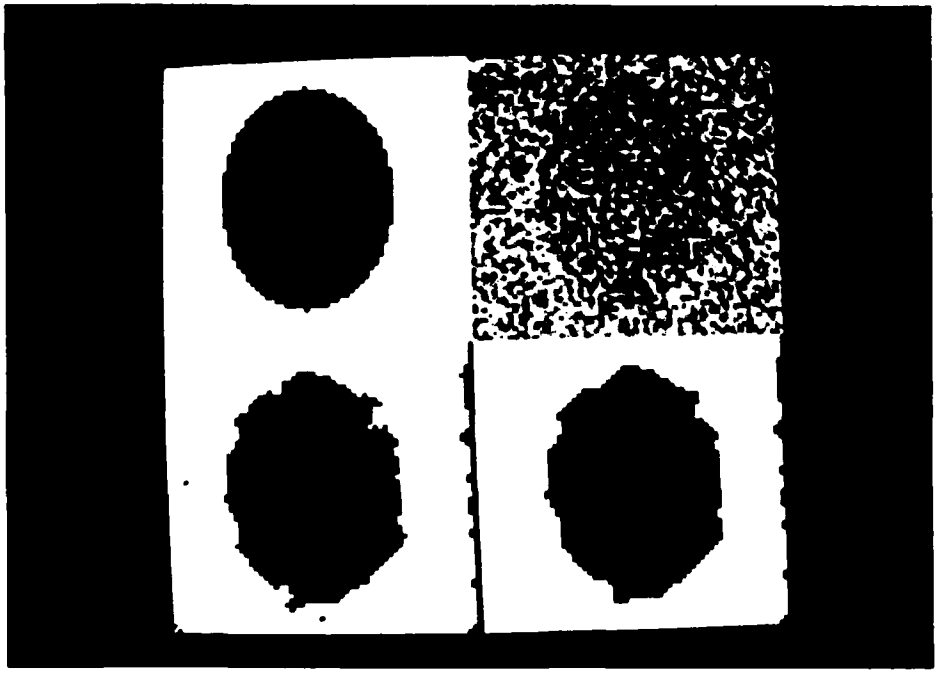


Figure 6 - Segmentation of an ellipse with S/N=1 and D=3

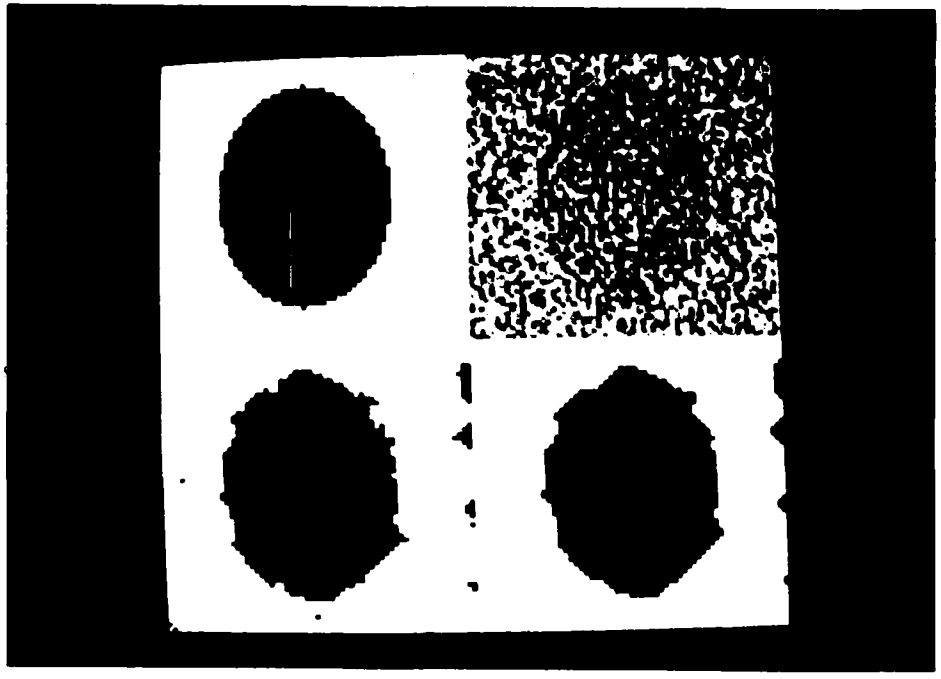


Figure 7 - Segmentation of an ellipse with S/N=1 and D=4

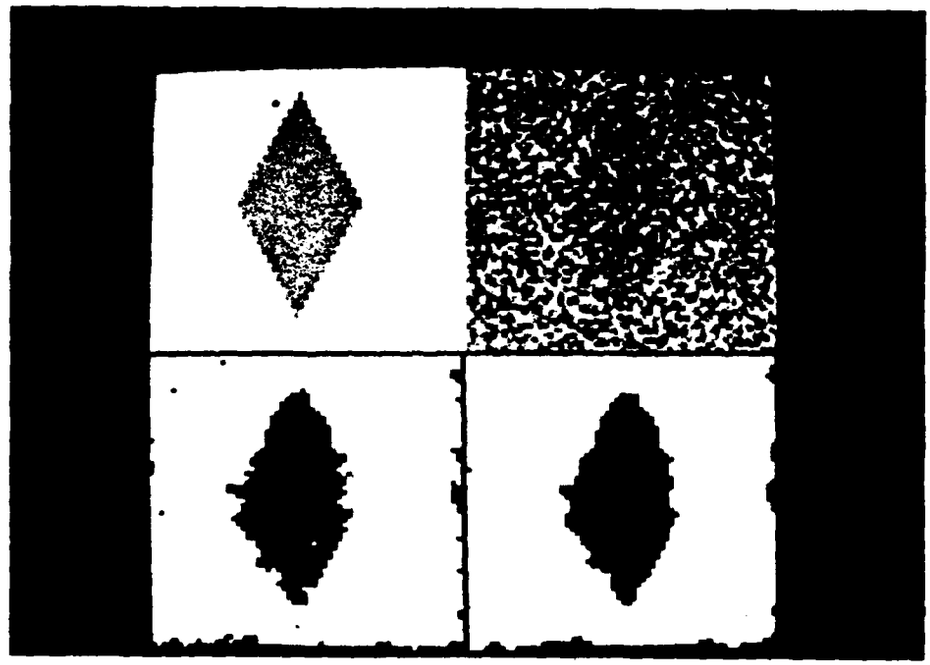


Figure 8 - Segmentation of a diamond with S/N=1 and all β_i equal

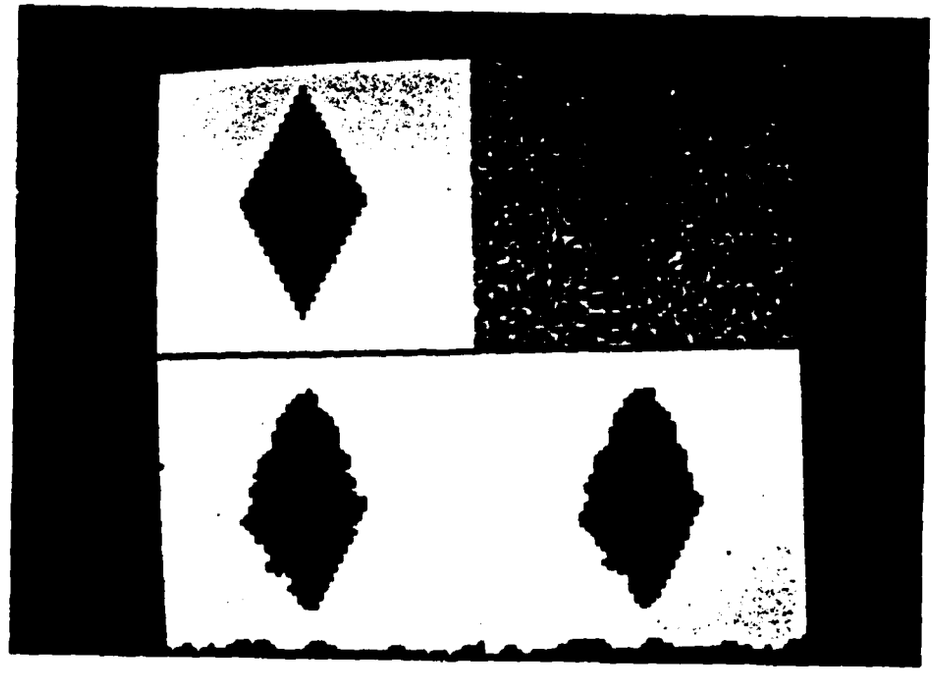


Figure 9 - Segmentation of a diamond with S/N=1 and β_i for diagonal cliques emphasized

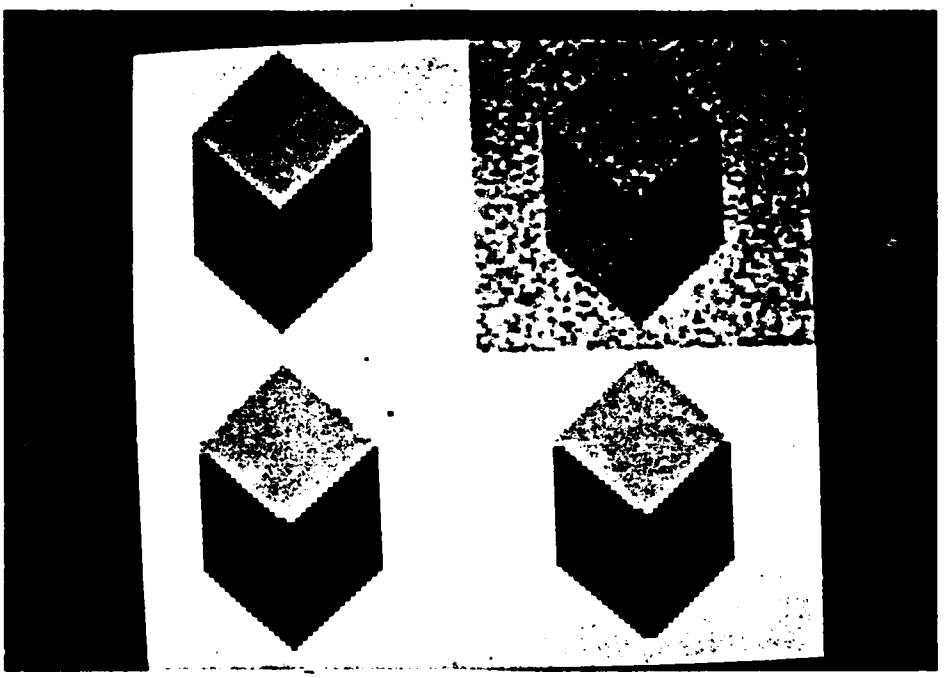


Figure 10 - Segmentation of a four region diamond test image with S/N=2 and D=2

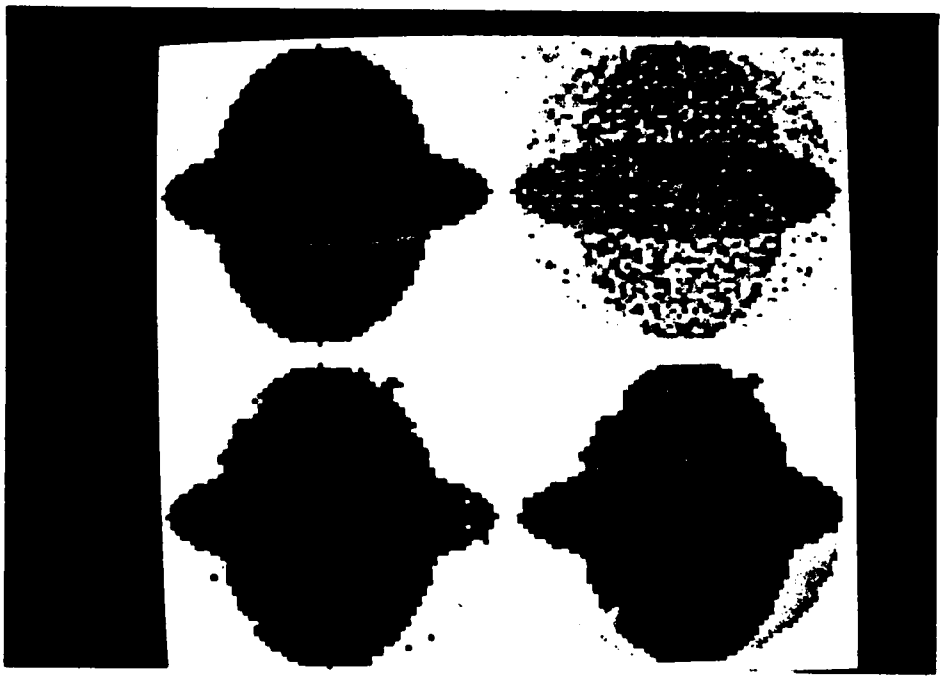


Figure 11 - Segmentation of a four region elliptical test image with S/N=2 and D=2

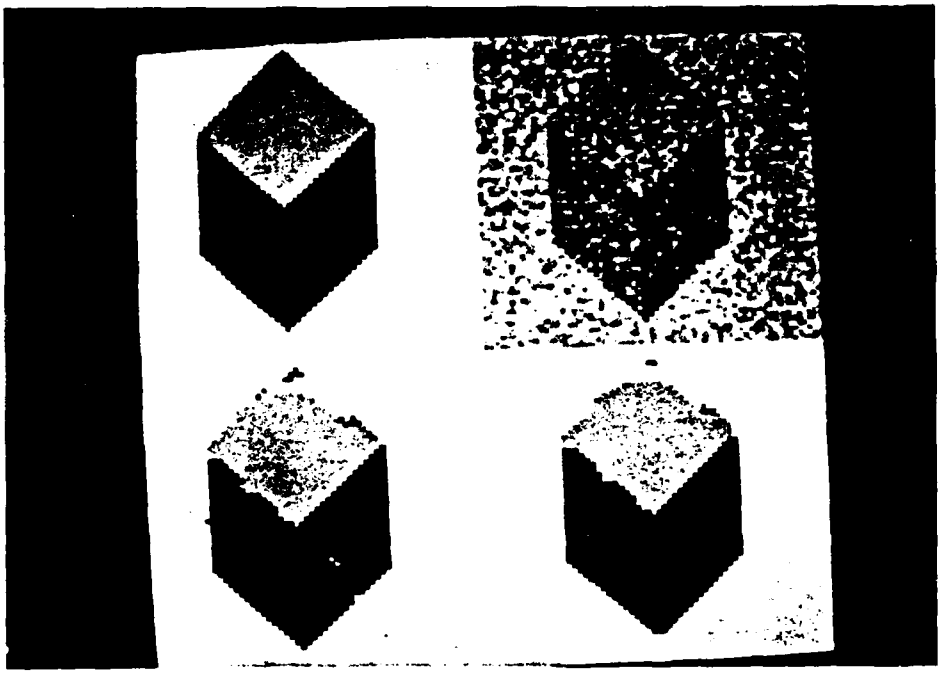


Figure 12 - Segmentation of a four region diamond test image with S/N=1.5 and D=2

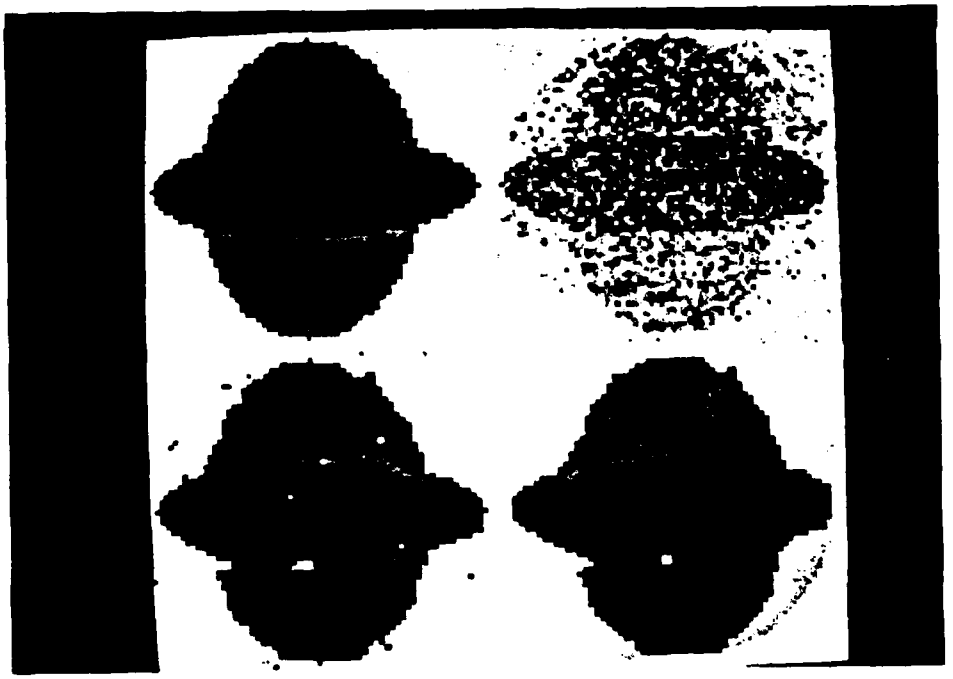


Figure 13 - Segmentation of a four region elliptical test image with S/N=1.5 and D=2

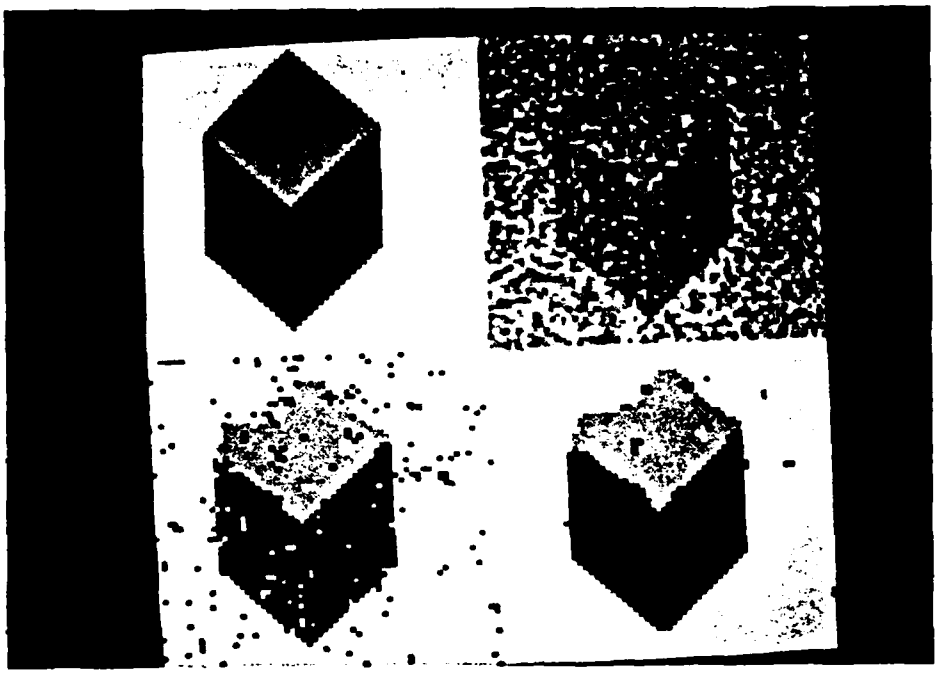


Figure 14 - Segmentation of a four region diamond test image with S/N=1 and D=2

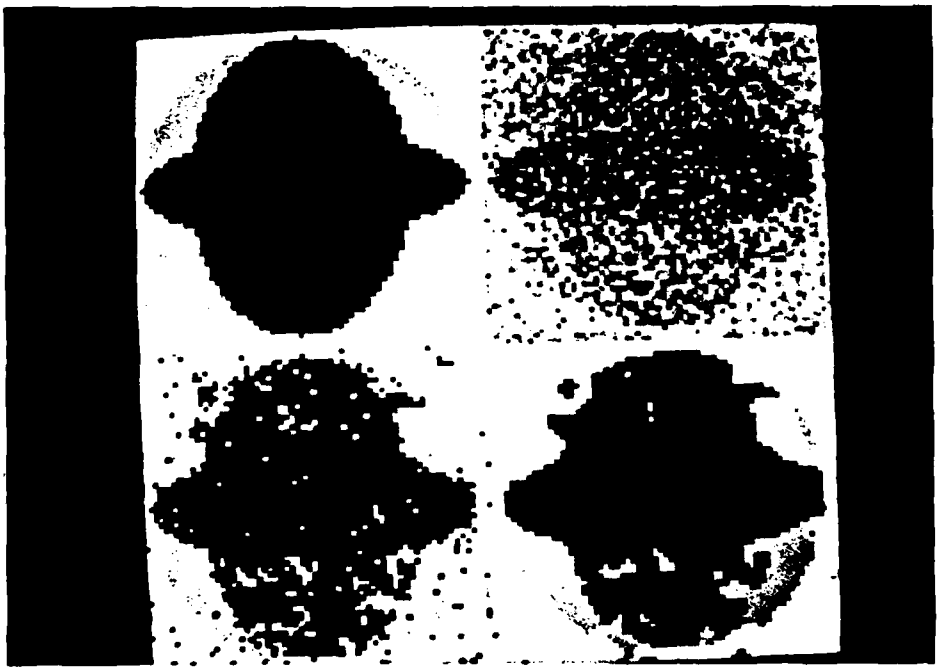


Figure 15 - Segmentation of a four region elliptical test image with S/N=1 and D=2

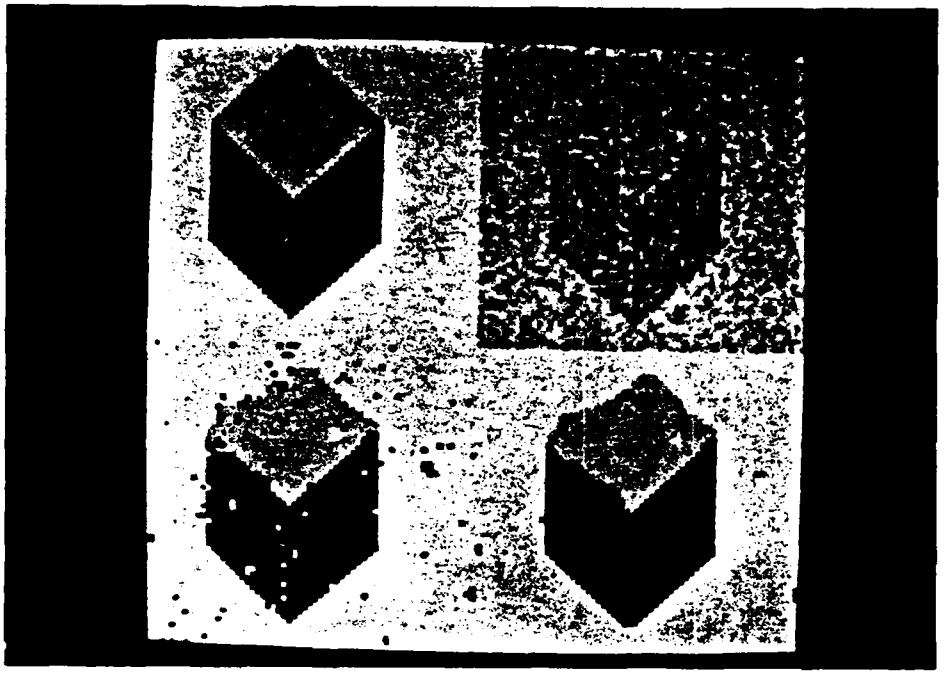


Figure 16 - Segmentation of a four region diamond test image with S/N=1 and D=3

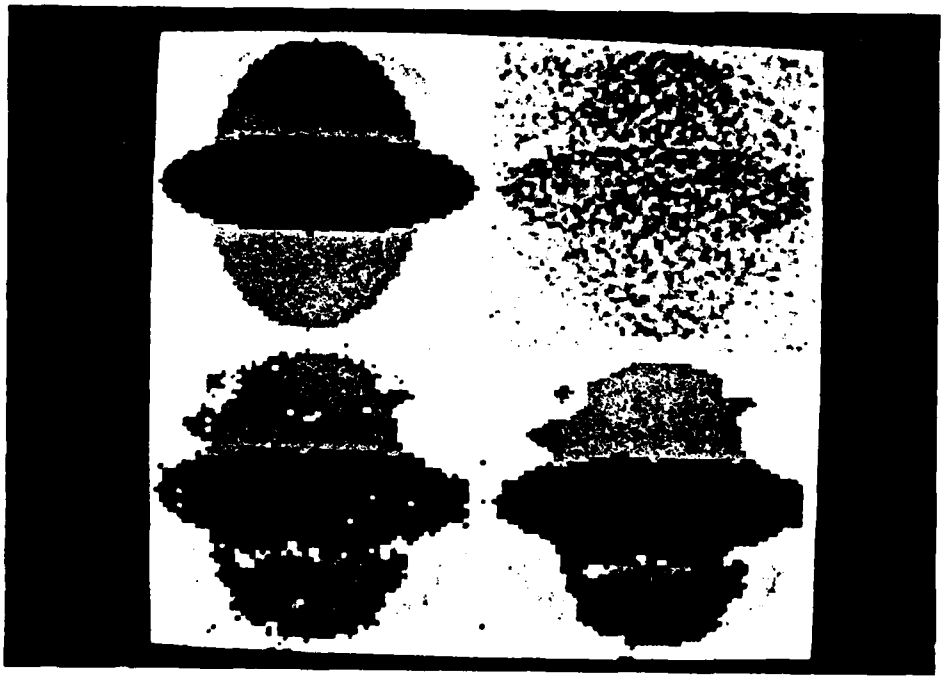


Figure 17 - Segmentation of a four region elliptical test image with S/N=1 and D=3

Figure 18 - (512 X 512) SAR Image

Figure 19 - Three segmentations of a (64 x 64)
Section of SAR image containing a small bay

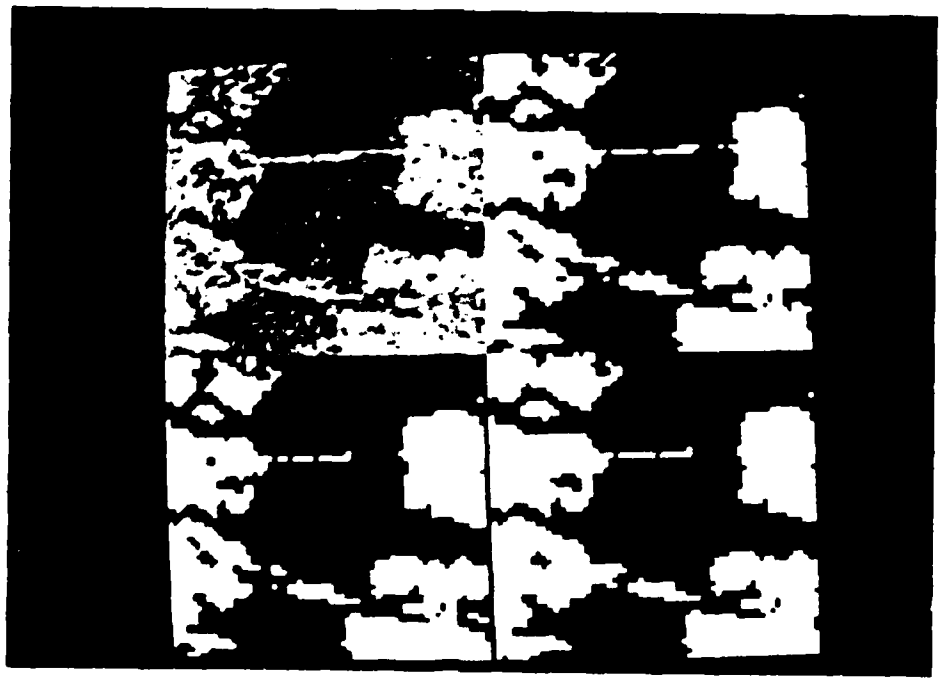


Figure 20 - Three segmentations of a (64 x 64) section of SAR image containing river with bridges

Figure 21 - Three segmentations of a (64 x 64) section of SAR image containing a boat

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER UMASS-ECE-Aug83-2	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle)		5. TYPE OF REPORT & PERIOD COVERED Intermediary, 1/82-8/83
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) H. Elliott, H. Derin, R. Cristi, D. Geman		8. CONTRACT OR GRANT NUMBER(s) N00014-83-K-0059
9. PERFORMING ORGANIZATION NAME AND ADDRESS Dept. of Electrical and Computer Engineering University of Massachusetts Amherst, MA 01003		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Statistics and Probability Program Office of Naval Research, Code 436 Arlington, VA 22217		12. REPORT DATE August 1983 13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Markov fields, Gibbs distribution, image segmentation, dynamic programming

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This paper describes a new statistical approach to image segmentation. Making use of Gibbs distribution models of Markov random fields, a dynamic programming algorithm is developed. A number of examples are presented which give an indication of the potential of this approach, and the algorithm is applied to SEASAT SAR imagery.

DD FORM 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-LF-014-6601

unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

END

FILMED

11-83

DTIC