BAYESIAN DECISION FEEDBACK FOR SEGMENTATION OF BINARY IMAGES

Srinivas R. Kadaba, Saul B. Gelfand and R. L. Kashyap
School of Electrical Engg., Purdue University, West Lafayette, IN 47907

ABSTRACT

We present real-time algorithms for the segmentation of binary images modeled by Markov Mesh Random Fields (MMRF's) and corrupted by independent noise. The goal is to find a recursive algorithm to compute the MAP estimate of each pixel of the scene using a fixed lookahead of D rows and D columns of the observations. The optimal algorithm for this is computationally expensive. Using both hard and soft (conditional) decision feedback, the complexity is reduced in a principled manner to allow a performance/complexity tradeoff. Simulation results demonstrate the viability of the algorithm and it's subjective relevance to the image segmentation problem.

I. INTRODUCTION

Consider the observation of an $L \times L$ lattice scene $\mathbf{X}$ in additive white noise $\mathbf{V}$ (known distribution function), the observations given by $\mathbf{Y} = \mathbf{X}+\mathbf{V}$. The goal is to compute MAP estimates of the pixels of $\mathbf{X}$. When we are interested in discrete features in the scene, the problem is one of segmentation. The segmentation problem is of significance in SAR imaging, medical imaging and military applications such as infrared imaging for detection of targets.

Previous approaches to this problem use one of two major stochastic models for the scene. One is via the use of Gibbs priors which entails the use of stochastic relaxation for the resulting optimization problem - this method was pioneered by Geman and Geman [1]. The use of relaxation renders this approach computationally intensive. The other is based on modeling the scene as discrete-valued with spatial interactions being specified in terms of transition probabilities of a Markov Mesh Random Field (MMRF). The posterior probability mass function is then computed for each pixel and the MAP estimate is computed by finding the maximum of a finite set. Well-known examples of this method are those of Derin *et al* [2] (fixed-interval smoothing) and Devijver [3] (recursive filtering and fixed-lag smoothing with lookahead of one row and one column). The methods in [2] involve multiple passes over the image thus requiring significant amounts of storage and computation while that in [3] requires large amounts of computation for large lags.

We propose an approach using a MMRF scene model and the concept of information states in recursive estimation [4] to compute fixed-lag recursive estimates of the scene pixels. The dimension of the information state is exponential in the lag and hence we propose the use of both hard and soft decision feedback (DF) [5] to allow large lags while maintaining the computational requirement at affordable levels. The result is a viable real-time segmentation algorithm called the Bayesian Conditional Decision Feedback (BCDF) algorithm that allows the user to choose the performance/complexity tradeoff.

Research supported by the IST Program of the BMDO monitored by the ONR under contract ONR N00014-91-J-4126 and by the NSF under contract NCR-9406073.

II. PROBLEM STATEMENT AND ASSUMPTIONS

All random fields considered are finite $L \times L$ lattices.

- The scene $\mathbf{X}$ (pixels take on values in a known set $\Omega = \{1, 2, \cdots, M\}$) obeys a third-order MMRF model [6] with known transition probabilities. The spatial causality of the MMRF model is necessary to allow a recursive structure for our algorithm. We recall

Definition 1: Let $\mathbf{X}_{(m,n)} = \{x_{(i,j)} : (i = 1, \ldots, m-1) \text{ or } (j = 1, \ldots, n-1)\}$. Then, a third order Markov Mesh Random Field is defined by the property

$$p(x_{(m,n)}/\mathbf{X}_{(m,n)}) = p(x_{(m,n)}/x_{(m,n-1)}, x_{(m-1,n-1)}, x_{(m-1,n)})$$

with appropriate boundary conditions.

- The noise field $\mathbf{V}$ is white and independent of $\mathbf{X}$ with $v_{(i,j)} \sim f(v) = \mathcal{N}(0, \sigma^2)$.
- The observation is $\mathbf{Y} = \mathbf{X} + \mathbf{V}$. We now refer to fig. 1(a) where the data set, the global and local state vectors and their partitions are shown for self-evident

0-7803-2431-5/95 $4.00 © 1995 IEEE

definitions. Our goal is: *Given the above observation model, find a recursive algorithm to compute,*

$$\hat{x}_{(m-D,n-D)} = \arg \max_{x_{(m-D,n-D)}} p(x_{(m-D,n-D)}/\mathbf{Y}_{(m,n)}), \quad D \geq 1 \quad (1)$$

i.e., the $D \times D$ (D row D column)-lag MAP estimate of the pixel $x_{(m,n)}$.

Proposition 1 specifies the information state $\mathbf{\Phi}_{(m,n)}$ in the optimal algorithm and propositions 2 and 3 specify the reduced dimension *pseudo-information states* in the suboptimal algorithms.

Proposition 1: The information state for estimating $\hat{x}_{(m-D,n-D)}$ is $\mathbf{\Phi}_{(m,n)} = p(\mathbf{S}^g_{(m,n)}/\mathbf{Y}_{(m,n)})$ and its dimension is $M^{D(D+1)+(m-D-1)}$.

Using the MMRF property, the independent noise model and Bayes' rule, a recursion can be written for $\mathbf{\Phi}_{(m,n)}$ as follows:

$$p(\mathbf{S}^g_{(m,n)}/\mathbf{Y}_{(m,n)}) \propto f(\mathbf{y}^g_{(m,n)}/\mathbf{c}^g_{(m,n)}) \cdots \sum_\alpha p(\mathbf{c}^g_{(m,n)}/\mathbf{S}^g_{(m,n-1)}) p(\mathbf{S}^g_{(m,n-1)}/\mathbf{Y}_{(m,n-1)}) \quad (2)$$

where $\alpha = \{\mathbf{c}^g_{(m-D-1,n)}, \mathbf{c}^-_{(m,n-D)}\}$. Thus the dimension increases as we progress across rows due to the presence of $\mathbf{c}^g_{(m-D-1,n)}$. Now, consider

Approximation A: At each site (m,n), all the past decisions $\hat{x}_{(1,1)}, \cdots, \hat{x}_{(1,L_1)}, \cdots, \hat{x}_{(m-D-1,1)}, \cdots, \hat{x}_{(m-D-1,L_1)} \cdots, \hat{x}_{(m-D,1)}, \cdots, \hat{x}_{(m-D,n-D-1)}$ are correct.

Let $\hat{\mathbf{T}}_{(m,n)} \triangleq \{\hat{x}_{(m-D-1,n)}, \cdots, \hat{x}_{(m-D-1,L_1)}, \hat{x}_{(m-D,1)}, \cdots, \hat{x}_{(m-D,n-D)}\}$.

Proposition 2: In view of approximation A, the *pseudo-information state* is $\mathbf{\Phi}^A_{(m,n)} = \left\{p\left(\mathbf{S}_{(m,n)}/\mathbf{Y}_{(m,n)}\right), \hat{\mathbf{T}}_{(m,n)}\right\}$ and its dimension is $M^{D(D+1)}$.

However, the dimension can still be large for large D. We now introduce another approximation based on partitioning the "local" state vector $\mathbf{S}_{(m,n)}$ as shown in fig. 1(b). Here $\mathbf{S}^+_{(m,n)}$ is a $J' \times J$ array of pixels where $0 \leq J' \leq (D+1), 0 \leq J \leq D$. (J', J) is called the *chip size.* Now for $x_{(i,j)} \in \mathbf{S}^-_{(m,n)}$, let $\tilde{x}_{(i,j)/(m,n)}$ denote the conditional MAP estimate

$$\tilde{x}_{(i,j)/(m,n)} = \arg \max_{x_{(i,j)}} p\left(x_{(i,j)}/\mathbf{S}^+_{(m,n)}, \mathbf{Y}_{(m,n)}\right)$$

Also, let $\tilde{\mathbf{S}}^-_{(m,n)} = \{\tilde{x}_{(i,j)/(m,n)} : x_{(i,j)} \in \mathbf{S}^-_{(m,n)}\}$.

Approximation B: At each site (m,n) the past conditional decisions $\tilde{\mathbf{S}}^-_{(m,n-1)}$ (conditioned on $\mathbf{S}^+_{(m,n-1)}$) are correct.

In other words, we assume $\mathbf{S}^-_{(m,n-1)}$ is conditionally deterministic given $\mathbf{S}^+_{(m,n-1)}$ and $\mathbf{Y}_{(m,n-1)}$ (*i.e.*, $\mathbf{S}^-_{(m,n-1)}$ is $\{\mathbf{S}^+_{(m,n-1)}, \mathbf{Y}_{(m,n-1)}\}$ measurable). Then,

Proposition 3: Assume approximations A and B hold. Then, the pseudo-information state is given by $\mathbf{\Phi}^{AB}_{(m,n)} = \{p(\mathbf{S}^+_{(m,n)}/\mathbf{Y}_{(m,n)}), \tilde{\mathbf{S}}^-_{(m,n)}, \hat{\mathbf{T}}_{(m,n)}\}$ and its dimension is $(D(D+1) - J'J)M^{D+1+J'(J-1)}$.

This pseudo-information state $\mathbf{\Phi}^{AB}_{(m,n)}$ forms the core of the BCDF algorithm. Note the successive reduction in the dimension of the information state. Following approximations A and B, the dimension is partially decoupled from the lag D. This is one of the principal features of the BCDF algorithm. The recursion for $\mathbf{\Phi}^{AB}_{(m,n)}$ proceeds as follows[1]. Similar to (2), compute (3) as given below:

$$p(\mathbf{S}^+_{(m,n-1)}, \mathbf{c}_{(m,n)}/\mathbf{Y}_{(m,n)}) \propto f(\mathbf{y}_{(m,n)}/\mathbf{c}_{(m,n)}) \cdot p(\mathbf{c}_{(m,n)}/\mathbf{S}^+_{(m,n-1)}) p(\mathbf{S}^+_{(m,n-1)}/\mathbf{Y}_{(m,n-1)}) \quad (3)$$

Then

$$p(\mathbf{S}^+_{(m,n)}/\mathbf{Y}_{(m,n)}) = \sum_{\mathbf{c}^+_J, \mathbf{c}^-} p(\mathbf{S}^+_{(m,n-1)}, \mathbf{c}_{(m,n)}/\mathbf{Y}_{(m,n)}) \quad (4)$$

Further,

$$p(\mathbf{S}^+_{(m,n)}, \mathbf{c}^-_{(m,n)}/\mathbf{Y}_{(m,n)}) = \sum_{\mathbf{c}^+_J} p(\mathbf{S}^+_{(m,n-1)}, \mathbf{c}_{(m,n)}/\mathbf{Y}_{(m,n)}) \quad (5)$$

The computation of $p(x_{(i,j)}/\mathbf{Y}_{(m,n)}, \mathbf{S}^+_{(m,n)}), x_{(i,j)} \in \mathbf{S}^-_{(m,n)}$ can be broken down into cases as

$$\begin{aligned} & p(x_{(i,j)}/\mathbf{Y}_{(m,n)}, \mathbf{S}^+_{(m,n)}) \\ & \propto \textstyle\sum_{\bar{\mathbf{c}}^+_J} p(\mathbf{S}^+_{(m,n-1)}, \mathbf{c}^+_{(m,n)}/\mathbf{Y}_{(m,n)}), \ x_{(i,j)} \in \mathbf{c}^+_J \\ & \propto \textstyle\sum_{\bar{\mathbf{c}}^-} p(\mathbf{S}^+_{(m,n)}, \mathbf{c}^-_{(m,n)}/\mathbf{Y}_{(m,n)}), \ x_{(i,j)} \in \mathbf{c}^- \\ & \propto \textstyle\sum_{\mathbf{c}^+_J} \Big(\mathcal{I}(\tilde{x}_{(i,j)/(m,n-1)} = x_{(i,j)}) \cdot p(\mathbf{S}^+_{(m,n-1)}, \mathbf{c}^+_{(m,n)}/\mathbf{Y}_{(m,n)}) \Big) \\ & \qquad x_{(i,j)} \in \mathbf{S}^-_{(m,n-1)} \setminus \mathbf{c}_{(m,n-D)} \end{aligned}$$

[1]For compactness and notational simplicity, we have denoted: $\mathbf{c}^+_J = \mathbf{c}^+_{(m,n-J)}$, $\mathbf{c}^- = \mathbf{c}^-_{(m,n)}$, $\bar{\mathbf{c}}^+_J = \mathbf{c}^+_{(m,n-J)} \setminus \{x_{(i,j)}\}$, $\mathbf{c}^- = \mathbf{c}^-_{(m,n)}$ and $\bar{\mathbf{c}}^- = \mathbf{c}^-_{(m,n)} \setminus \{x_{(i,j)}\}$.

The conditional estimates are obtained from

$$\tilde{x}_{(i,j)/(m,n)} = \arg \max_{x_{(i,j)}} p(x_{(i,j)}/\mathbf{Y}_{(m,n)}, \mathbf{S}^{+}_{(m,n)}) \quad (6)$$

Finally, compute

$$p(x_{(m-D,n-D)}/\mathbf{Y}_{(m,n)}) = \sum_{\gamma} \Big(\mathcal{I}(\tilde{x}_{(m-D,n-D)/(m,n-1)} = x_{(m-D,n-D)}) \; p(\mathbf{S}^{+}_{(m,n-1)}, \mathbf{c}^{+}_{(m,n)}/\mathbf{Y}_{(m,n)}) \Big) \quad (7)$$

where $\gamma = \left\{ \mathbf{c}^{+}_{J}, \mathbf{S}^{+}_{(m,n)} \right\}$. The desired fixed-lag estimate $\hat{x}_{(m-D,n-D)}$ is obtained using (7) in (1).

The algorithm is initialized by quantizing the first row and column observations and specifying $\mathbf{\Phi}^{AB}_{(D+2,1)}$. The recursion is restarted at each row. Termination is accomplished by tapering the lag to zero as the boundary is approached.

III. SIMULATIONS

The BCDF algorithm was applied to a number of synthetic binary images and real images to test its various aspects. The signal-to-noise ratio is defined as $SNR = \frac{|l_2 - l_1|}{\sigma}$. Fig. 2 shows one result where the algorithm is applied to the segmentation of a synthetic image. Fig. 3 shows the segmentation of a plane from a noisy image. For purposes of illustration, the original image was quantized into two levels and the transition probabilities were calculated using a relative frequency approach from this bilevel image. The results demonstrate that the algorithm achieves the desired objective, *i.e.*, labeling of the pixels to separate the desired feature from the background.

IV. ALGORITHM FEATURES

- The BCDF algorithm is pixel-wise recursive and is developed in a principled 1-step optimal Bayesian framework.
- The computational complexity is partially decoupled from the lag D and linked to the chip size. This gives an ability to maintain sufficient lag while controlling complexity. The chip size can be chosen by the user allowing a performance-complexity tradeoff.
- Boundary conditions can be addressed completely within the framework of the algorithm. No assumptions need be made about pixels that do not lie in the image frame.
- Transition probabilities can be estimated recursively and incorporated in the BCDF algorithm to allow an "adaptive" version. We have observed that the algorithm is reasonably robust to variations in the assumed transition probabilities at medium SNR's. Hence a simple relative frequency approach is sufficient for estimating the transition probabilities. However, sophisticated estimators can be used (at lower SNR's) at the expense of increased computation.
- A limitation of the algorithm is that the complexity increases rapidly for multilevel images. This motivates the investigation of a "reduced-state" approach.

V. CONCLUSIONS

In this paper, we investigated reduced complexity fixed-lag smoothing for the segmentation of binary images, in a effort to render real-time processing viable. Beginning with the optimal fixed-lag MAP pixel-by-pixel estimator for an MMRF model, we applied hard decision feedback to fix the size of the statistic and soft (conditional) decision feedback to further reduce complexity without reducing lag. This results in the Bayesian Conditional Decision Feedback (BCDF) algorithm where the exponential component of the complexity is partially decoupled from the lag and linked to a user-chosen parameter called the "chip size". The chip size can be chosen to yield the desired performance-complexity tradeoff. Simulations demonstrate the performance and viability of the algorithm.

VI. REFERENCES

[1] S. Geman and D. Geman, "Stochastic Relaxation, Gibbs Distributions and the Bayesian Restoration of Images", *IEEE. Trans. PAMI*, pp. 721–741, Nov. 1984.

[2] H. Derin *et al*, "Bayes smoothing algorithms for segmentation of binary images modeled by Markov random fields", *IEEE Trans. PAMI*, pp. 707–720, Nov. 1984.

[3] P. A. Devijver, "Segmentation of binary images using third-order Markov mesh image models", *in Proc. 8th Internat. Conf. Pattern Recognition*, Paris, 1986.

[4] P. R. Kumar and Pravin Varaiya, *Stochastic Systems: Estimation, Identification and Adaptive Control*, Prentice-Hall, Inc., NJ, 1986.

[5] G. K. Lee, S. B. Gelfand and M. P. Fitz, "Bayesian Decision Feedback Techniques for Deconvolution", *in Proc. Globecom*, 1994, Also in *IEEE J. Selected Areas in Communications*, to be published.

[6] K. Abend, T. J. Harley and L. N. Kanal, "Classification of Binary Random Patterns", *IEEE Trans. Info. Theory*, pp. 538–544, Oct. 1965.

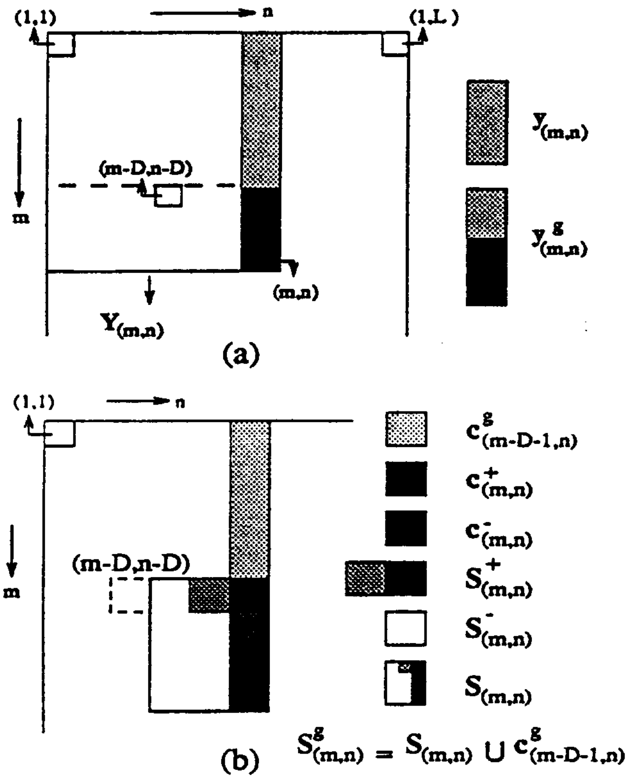

Figure 1: (a) data sets (b) global and local state vectors and partitions with $D = 3$ and (J', J)=(1,2)

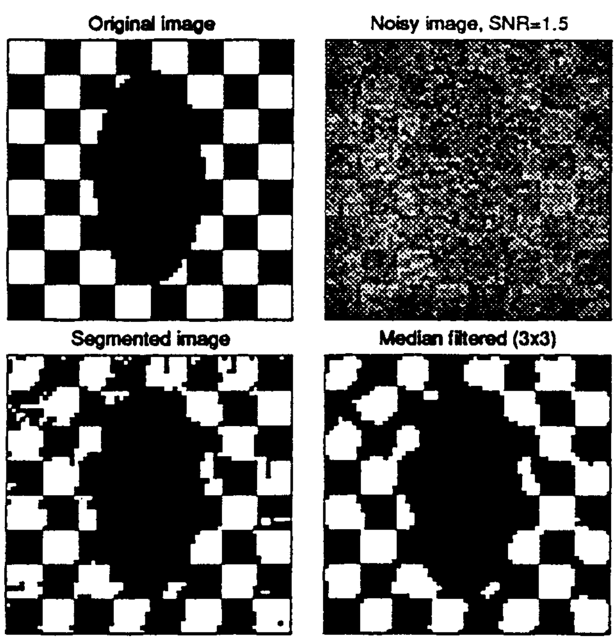

Figure 2: Segmentation of ellipse and checkerboard : transition probabilities known, $D = 3$, $(J', J) = (1, 2)$, SNR=1.5, Gaussian noise.

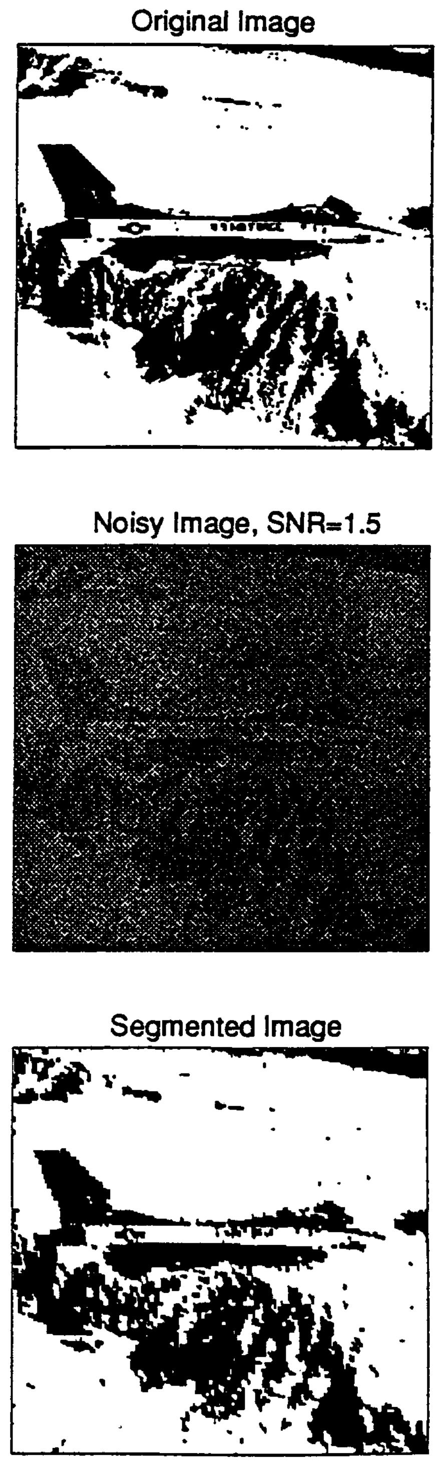

Figure 3: Segmentation of a Jet: transition probabilities known, $D = 3$, $(J', J) = (1, 2)$, SNR=1.5, Gaussian noise.