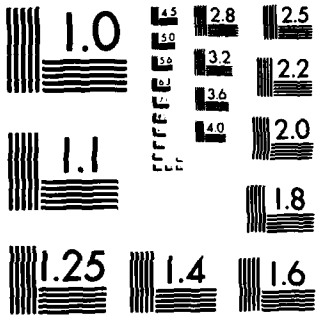
AD-A151 503 BAYESIAN ESTIMATION OF ONE DIMENSIONAL DISCRETE MARKOV RANDOM FIELDS..(U) MASSACHUSETTS INST OF TECH CAMBRIDGE LAB FOR INFORMATION AND D.. J L MARROQUIN DEC 84 1/1

UNCLASSIFIED LIDS-P-1423 ARO-20980.9-MA DAAG29-84-K-0005 F/G 12/1 NL

END



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD-A151 503

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: ARO 20980.9-MA

2. GOVT ACCESSION NO.: N/A

3. RECIPIENT'S CATALOG NUMBER: N/A

TITLE (and Subtitle): Bayesian Estimation of One Dimensional Discrete Markov Random Fields

5. TYPE OF REPORT & PERIOD COVERED: Technical Report

6. PERFORMING ORG. REPORT NUMBER: LIDS-P-1423

AUTHOR(s): J.L. Marroquin

8. CONTRACT OR GRANT NUMBER(s): Army Research Office ARO-DAAG29-84-K-0005

PERFORMING ORGANIZATION NAME AND ADDRESS:
M.I.T.
Laboratory for Information and Decision Systems
Cambridge, MA 02139

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

CONTROLLING OFFICE NAME AND ADDRESS:
U. S. Army Research Office
Post Office Box 12211
Research Triangle Park, NC 27709

12. REPORT DATE: December 1984

13. NUMBER OF PAGES: 34 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.

DTIC ELECTE MAR 15 1985 S A

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

NA

18. SUPPLEMENTARY NOTES

The view, opinions, and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

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

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

We present two deterministic algorithms for the maximum a posteriori estimation of a one dimensional, binary Markov random field from noisy observations. Extensions to other related problems, such as one dimensional signal matching, and estimation of continuous valued Markov random fields are also discussed. Finally, we present an experimental comparison of the performance of our optimal algorithms with a stochastic approximation scheme (simulated annealing).

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

DTIC FILE COPY

85 03 06

ARO 20980.9-MA

December, 1984

LIDS-R-1423

Accession For
NTIS GRA&I
DTIC TAB
Unannounced
Justification
By
Distribution/
Availability Codes
Dist | Avail and/or Special
A-1

BAYESIAN ESTIMATION OF ONE DIMENSIONAL DISCRETE MARKOV RANDOM FIELDS.

J. L. Marroquin

ABSTRACT: We present two deterministic algorithms for the maximum a posteriori estimation of a one dimensional, binary Markov random field from noisy observations. Extensions to other related problems, such as one dimensional signal matching, and estimation of continuous valued Markov random fields are also dicussed. Finally, we present an experimental comparison of the performance of our optimal algorithms with a stochastic approximation scheme (simulated annealing).

This report describes research done within the Laboratory for Information and Decision Systems and the Artificial Intelligence Laboratory at the Massachusetts Institute of Technology. Support for the A.I. Laboratory's research is provided in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research Contract N00014-80-C-0505. The author was supported by the Army Research Office under contract ARO-DAAG29-84-K-0005.

Laboratory for Information and Decision Systems.
Massachusetts Institute of Technology.
Cambridge, Mass. 02139.

1. Introduction.

Recently,there has been a lot of interest about the use of Markov random field (MRF) models and Bayesian estimation techniques for signal processing tasks, such as image restoration [G1,G2]; surface reconstruction [M1], and image segmentation [E1,M1]. The use of these techniques is very attractive, since it provides a unified framework for the formulation of a variety of problems, and it permits the incorporation of a priori knowledge about the nature of the solutions that one expects to find in a simple and elegant way.

One serious problem associated with their use, however, is that it involves the global minimization of a non-convex "energy" function of many variables, and although stochastic approximation methods, such as "Simulated Annealing" [K1], have been found effective for finding a solution, their computational efficiency leaves much to be desired.

This situation provides the motivation for trying to exploit the structure of each particular problem to find more efficient (possibly deterministic) methods to find the optimal estimates. In this paper we study the simplest problem of this class: finding the maximum a posteriori (MAP) estimate of a one dimensional binary Markov random field, and show that it is indeed possible to find efficient deterministic algorithms for its solution.

1.1. Formulation of the Problem.

Consider a one dimensional lattice with N nodes, and suppose that at each node j there is a cell whose state can be modelled as a random variable F_j, which can take only two values: $F_j \in \{k_0, k_1\}$. Suppose also that the conditional probabilities for the collection F satisfy:

$$\Pr(F_j = f_j \mid F_i = f_i, \quad i \neq j) = \Pr(F_j = f_j \mid F_i = f_i, \quad i \in [1, N]; \quad |i - j| = 1)$$

that is, F is a first order MRF. In this case, it can be shown [P1,K2] that the joint probability density of the configuration F is given by the Gibbs distribution:

$$P(F = f) = \frac{1}{Z} \exp[-\frac{1}{\alpha} \sum_{i=1}^{N-1} V(f_i, f_{i+1})] \tag{1}$$

where Z is a normalizing constant, α is a parameter and the functions V are the potentials of the system. In particular, we will consider the potentials:

$$V(f_i, f_{i+1}) = \begin{cases} -1, & \text{if } f_i = f_{i+1} \\ 1, & \text{if } f_i \neq f_{i+1} \end{cases}$$

In this case, F corresponds to the one dimensional Ising model of ferromagnetic phenomena for a finite lattice with free boundaries, and α can be interpreted as the natural temperature of the system.

Suppose now that we have some noisy observations of a particular realization f of the field F. Our problem is to find the "best" estimate for f given these observations and our prior knowledge about the properties of f.

We will use the following model for the observation process g:

$$g_i = H(f_i, n_i)$$

where n_i is a white noise process (so that n_i is independent of n_j for all $i \neq j$) independent of F, and H is a deterministic function invertible with respect to n_i, so that we can write the conditional distribution of g in the form:

$$P(g \mid f) = \frac{1}{Z_g} \exp[-\sum_{i=1}^{N} \Phi_{f_i}(g_i)]$$

where Z_g is a constant independent of f, and Φ_{k_0}, Φ_{k_1} are deterministic functions.

Two familiar instances of this model are: the binary symmetric channel with error rate ϵ, in which case

$$\Phi_k(g_i) = \ln[\frac{(g_i - k)^2}{4}\epsilon + \left(1 - \frac{(g_i - k)^2}{4}\right)(1 - \epsilon)],$$

and the case of additive or multiplicative white noise (not necessarily Gaussian). (For additive white Gaussian noise,

$$\Phi_k(g_i) = \frac{1}{2\sigma^2}(g_i - k)^2)$$

Using Bayes rule, we find that the posterior distribution is:

$$P(f \mid g) = \frac{1}{P(g) \cdot Z \cdot Z_g} \exp[-\frac{1}{\alpha} \sum_i V(f_i, f_{i+1}) - \sum_i \Phi_{f_i}(g_i)]$$

which is also a Gibbs measure. Since $P(g)$ and Z_g are constants for a given set of observations, the Bayesian (MAP) estimate for f is found by minimizing the "energy function":

$$U(f) = \sum_{i=1}^{N-1} V(f_i, f_{i+1}) + \alpha \sum_i \Phi_{f_i}(g_i) \tag{2}$$

In the particular case of additive white Gaussian noise, the equivalent problem (for $f_i \in \{-1, 1\}$) is to minimize:

$$U(f) = \sum_i (f_i - f_{i+1})^2 + \frac{\gamma}{2} \sum_i (f_i - g_i)^2 \tag{3}$$

or equivalently, minimize

$$U(f) = -\sum_i f_i f_{i+1} - \frac{\gamma}{2} \sum_i f_i g_i$$

where $\gamma = \frac{\alpha}{2\sigma^2}$ corresponds to the signal to noise ratio. In physical terms, this minimization can be interpreted as the problem of finding the ground state [1] of an Ising ferromagnet subject to a spatially varying external magnetic field (whose magnitude is proportional to g), a system which is of current interest in physics.

What makes this problem particularly hard is that the value of each f_j is constrained to be in the non-convex set $\{k_0, k_1\}$. If we relax this condition, in the case of additive Gaussian noise, (3) becomes a convex function (since it is a positive definite quadratic form), and its (unique) minimum can be found efficiently, for example, by a gradient descent method. Alternatively, we may construct a linear dynamic system with the same (exponential) covariance function as the process f, and use a Kalman filter to find the MMSE estimate. However, it is not clear how to use these relaxed solutions to find the correct (binary) optimal estimate. Instead, we will now present two algorithms for minimizing (3) directly, which have the additional advantage of being able to handle other (non Gaussian) noise models.

2. Dynamic Programming Formulation.

In this section we present an algorithm for finding the global minimum of (2), which, based on dynamic programming principles, reduces the problem to a sequence of one dimensional optimizations.

As we will see, this algorithm generates, as a byproduct, a family of solutions which can be considered as descriptions of the field f at different scales, so that the coarse descriptions, which are computed very fast, are progressively refined until the optimal (finest scale) configuration is found.

This approach is based on the following idea:

A configuration f is completely characterized by the value of f_1, and the set $\mathcal{L}_n$ defined by:

$$\mathcal{L}_n = \{L \ : \ f_L \neq f_{L+1}\} \quad ; \quad |\mathcal{L}_n| = n.$$

We will call the n elements of $\mathcal{L}_n$ the "boundaries" of the configuration f. Since these boundaries correspond to odd bonds between neighboring cells, we can define an equivalent energy function as:

$$U(f) = n + \frac{\alpha}{2}\hat{U}(f)$$

$$\text{with } \hat{U}(f) = \sum_i \Phi_{f_i}(g_i), \quad f_i \in \{k_0, k_1\} \tag{4}$$

For a fixed n, U depends only on the value of f_1, and on the position of the n boundaries, that is, on $n+1$ variables. To make this dependence more explicit, let us define the functions

$$G(L) = \sum_{j=1}^{L} (\Phi_{k_0}(g_j) - \Phi_{k_1}(g_j)) \tag{5}$$

Let U_0 and U_1 denote the energy functions corresponding to the configurations with $f_1 = k_1$ and k_0, respectively, for a given set of boundaries

$$\mathcal{L}_n = \{L_1, \ldots L_n\}, \quad L_1 < \ldots < L_n \tag{6}$$

We have that, for n even,

$$U_0(n, \mathcal{L}_n) = n + \frac{\alpha}{2}[\sum_{j=1}^{L_1} \Phi_{k_0}(g_j) + \sum_{L_1+1}^{L_2} \Phi_{k_1}(g_j) + \ldots + \sum_{L_n+1}^{N} \Phi_{k_0}(g_j)] =$$

$$= n + \frac{\alpha}{2}[G(L_1) - G(L_2) + \ldots - G(L_n) + \sum_{j=1}^{N} \Phi_{k_0}(g_j)]$$

$$U_1(n, L_n) = n + \frac{\alpha}{2}[\sum_{j=1}^{L_1} \Phi_{k_1}(g_j) + \sum_{L_1+1}^{L_2} \Phi_{k_0}(g_j) + \ldots + \sum_{L_n+1}^{N} \Phi_{k_1}(g_j)] =$$

$$n + \frac{\alpha}{2}[-G(L_1) + \ldots + G(L_n) - G(N) + \sum_{j=1}^{N} \Phi_{k_0}(g_j)] \tag{7}$$

and for n odd,

$$U_0(n, L_n) = n + \frac{\alpha}{2}[G(L_1) - G(L_2) + \ldots + G(L_n) - G(N) + \sum_{j=1}^{N} \Phi_{k_0}(g_j)]$$

$$U_1(n, L_n) = n + \frac{\alpha}{2}[-G(L_1) + \ldots - G(L_n) + \sum_{j=1}^{N} \Phi_{k_0}(g_j)] \tag{8}$$

(Note that $\sum_j \Phi_{k_0}(g_j)$ does not depend on f).

Let $S_n^{(0)}, S_n^{(1)}$ be the sets of boundaries that minimize U_0 and U_1, respectively. Then, the optimal energy for a given n is:

$$U_n^* = min[U_0(n, S_n^{(0)}), U(n, S_n^{(1)})] \tag{9}$$

We will define S_n to be the corresponding optimal set of boundaries.

The determination of $S_n^{(k)}$ is an n-dimensional optimization problem. However, as we will show below, it is possible to decompose it into a sequence of one dimensional optimizations using a dynamic programming formulation. With this approach we also get, as a bonus, the solutions $S_1^{(k)}, \ldots, S_{n-1}^{(k)}$, $k \in \{0, 1\}$, and the corresponding optimal energies. If we set $n = N$, the solution to the original problem (3), $U^*(n^*, S_{n^*})$ can then be found by a one dimensional search. This strategy, however, can be dramatically improved by the use of the following facts:

(i) We can reduce substantially the search space for the location of the optimal boundaries $L_j \in S_{n^*}$.

(ii) The sequences $\{U_1^*, U_3^*, \ldots\}$ and $\{U_2^*, U_4^*, \ldots\}$ are unimodal. This, together with the fact that the dynamic programming algorithm uses S_{j-1} to compute S_j provides us with an efficient stopping criterion for the computation of the sequence $\{S_1, \ldots, S_{n^*}\}$.

(iii) The expected value of n^* is usually small.

We will now describe the algorithm, and analyze each one of these facts.

2.1. Search Space for the Optimal Boundaries.

Let

$$P_M = \{M_1, M_2, \ldots\} =$$

$$= \{j: \quad G(j-1) \leq G(j) \geq G(j+1), \text{ with } G(j-1) \neq G(j+1)\} \tag{10}$$

$$\mathcal{P}_m = \{m_1, m_2, \ldots\} =$$

$$= \{j: \quad G(j-1) \geq G(j) \leq G(j+1), \text{ with } G(j-1) \neq G(j+1)\} \tag{11}$$

(Conventionally we include $j = 1$ in $\mathcal{P}_M$, if $0 < G(1) \geq G(2)$, and include it in $\mathcal{P}_m$ if $0 > G(1) \leq G(2)$). We define the set $\mathcal{P}$ as

$$\mathcal{P} = \mathcal{P}_M \bigcup \mathcal{P}_m = \{P_1, \ldots, P_r\}$$

(Note that $\mathcal{P}$ corresponds to the set of places where the sequence $\{\Phi_{k_0}(g_j) - \Phi_{k_1}(g_j)\}$ changes sign).

In what follows, we will call the elements of $\mathcal{P}_M$, $\mathcal{P}_m$ and $\mathcal{P}$, the "maxima", "minima", and "critical points" of G, respectively.

Let $S_{n^\bullet +}$ ($S_{n^\bullet -}$) denote the subsets of $S_{n^\bullet}$ formed by those boundaries L_j whose corresponding term $G(L_j)$ has positive (negative) coefficient in $U^*_{n^\bullet}$, i.e., if

$$S_{n^\bullet} = S^{(k)}_{n^\bullet} = \{L_1, \ldots L_{n^\bullet}\},$$

then,

$$S_{n^\bullet +} = \{L_{1+k}, L_{3+k}, \ldots\}$$

$$S_{n^\bullet -} = S_{n^\bullet} - S_{n^\bullet +} \tag{12}$$

With these definitions, we have:

Theorem 1: $S_{n^\bullet +} \subseteq \mathcal{P}_m$ and $S_{n^\bullet -} \subseteq \mathcal{P}_M$.

To see why this is true, let f^{ML} denote the maximum likelihood estimate for f [2] obtained by:

$$f^{ML}_j = \begin{cases} k_1, & \text{if } \Phi_{k_1}(g_j) > \Phi_{k_0}(g_j) \\ 0, & \text{otherwise} \end{cases}$$

and let f^* be the optimal estimate. Suppose that for some j we have, say, $L_j \in S_{n^\bullet +} - \mathcal{P}_m$. Suppose $L_j \in (P_k, P_{k+1})$, for some $P_k, P_{k+1} \in \mathcal{P}$. Clearly, either $P_k \in \mathcal{P}_m$ or $P_{k+1} \in \mathcal{P}_m$. Suppose, for definiteness that $P_k \in \mathcal{P}_m$.

If $P_k \notin S_{n^\bullet}$, the configuration $\{L_1, \ldots L_{j-1}, P_k, L_{j+1}, \ldots L_{n^\bullet}\}$ has lower energy than $S_{n^\bullet}$ (we decrease $\hat{U}$ without altering n), which is a contradiction. If $P_k \in S_{n^\bullet}$, then either

$$f^*((P_k, L_j)) \neq f^{ML}((P_k, L_j))$$

$$\text{or } f^*((L_j, P_{k+1})) \neq f^{ML}((L_j, P_{k+1}))$$

and so, we get a lower energy configuration by deleting L_j and either P_k or P_{k+1} (we decrease simultaneously n and $\hat{U}$). A similar argument can be used if $L_j \in [1, P_1)$ or $L_j \in (P_r, N]$.∎

This result means that we can use $\mathcal{P}$ to constrain the search space for the boundaries of each subproblem (i.e., for each fixed n), which now becomes:

For $n < |\mathcal{P}|$ fixed, find $S_n = \{L_1, \ldots L_n\}$ with

$$S_{n+} \subseteq \mathcal{P}_m \text{ and } S_{n-} \subseteq \mathcal{P}_M \tag{13}$$

such that $U(n, S_n) < U(n, L_n)$ for all $L_n \subseteq \mathcal{P}$.

Note that theorem 1 guarantees that the constrained and unconstrained solutions will coincide only for $n = n^*$, so that for $n \neq n^*$, S_n may, in general, be suboptimal.

2.2. Dynamic Programming (DP) Algorithm.

From equations (7) and (8), it is clear that, for any fixed n, the determination of the optimal (constrained) configurations $S_n^{(0)}, S_n^{(1)}$ is equivalent to the solution of the optimization problems:

For $S_n^{(0)}$:

$$\text{Minimize} \quad [G(L_1) - G(L_2) + \ldots]$$

with $L_1, L_3, \ldots \in \mathcal{P}_m$, and $L_2, L_4, \ldots \in \mathcal{P}_M$.

For $S_n^{(1)}$:

$$\text{Maximize} \quad [G(L_1) - G(L_2) + \ldots]$$

with $L_1, L_3, \ldots \in \mathcal{P}_M$, and $L_2, L_4, \ldots \in \mathcal{P}_m$.

Let us consider the maximization problems. Assume, for definiteness that the first critical point of G is a maximum, i.e., $M_1 < m_1$, and define the sequences:

$$D_1(k) = \sup_{i \geq k} G(M_i)$$

$$L_1(k) = \{\min L \ : \ G(M_L) = D_1(k)\}, \quad k = 1 \ldots |\mathcal{P}_M| \tag{14}$$

Clearly, $M_{L_1(1)}$ is the optimal location of the boundary for $n = 1$ (i.e., $S^{(1)} = \{M_{L_1(1)}\}$), and from $D_1(1)$ we can easily compute the corresponding energy. We now define, for $j \geq 1$:

$$D_{2j}(k) = \sup_{i \geq k} \{D_{2j-1}(i+1) - G(m_i)\}$$

$$D_{2j+1}(k) = \sup_{i \geq k}\{D_{2j}(i) - G(m_i)\}$$

and

$$L_{2j}(k) = \{\min L \quad : \quad D_{2j}(k) = D_{2j-1}(L+1) - G(M_L)\}$$

$$L_{2j+1}(k) = \{\min L \quad : \quad D_{2j+1}(k) = D_{2j}(L) + G(M_L)\} \tag{15}$$

One can check that, for n odd,

$$S_n^{(1)} = \{M_{L_n(1)}, m_{L_{n-1}(L_n(1))}, \ldots, M_{L_1(L_2(\ldots(L_n(1))\ldots)}\} \tag{16}$$

and the optimal energy is:

$$U_1(n) = n + \frac{\alpha}{2}[-D_n(1) + \sum_j \Phi_{k_0}(g_j)] \tag{17}$$

For n even, we define:

$$D_1^{'}(k) = \sup_{i \geq k}\{-G(m_i)\}$$

$$L_1^{'}(k) = \{\min L \quad : \quad D_1^{'}(k) = -G(m_L)\}$$

$$D_{2j}^{'}(k) = \sup_{i \geq k}\{D_{2j-1}^{'}(i) + G(M_i)\}$$

$$L_{2j}^{'}(k) = \{\min L \quad : \quad D_{2j}^{'}(k) = D_{2j-1}^{'}(L) + G(M_L)\}$$

$$D_{2j+1}^{'} \sup_{i \geq k}\{D_{2j}^{'}(i+1) - G(m_i)\}$$

$$L_{2j+1}^{'}(k) = \{\min L \quad : \quad D_{2j+1}^{'}(k) = D_{2j}^{'} - G(m_L)\} \tag{18}$$

and get:

$$S_n^{(1)} = \{M_{L_n^{'}(1)}, \ldots m_{L_1^{'}(L_n^{'}(\ldots L_n^{'}(1))\ldots)}\} \tag{19}$$

$$U_1(n) = n + \frac{\alpha}{2}[-D_n^{'}(1) - G(N) + \sum_j \Phi_{k_0}(g_j)] \tag{20}$$

For the minimization problems, that is, for the computation of $S_n^{(0)}$, assuming again that $M_1 < m_1$, we have, for n even:

$$d_1(k) = \inf_{i \geq k}\{-G(m_i)\}$$

$$l_1(k) = \{\min l \quad : \quad d_1(k) = -G(m_l)\}$$

and for $j \geq 1$,

$$d_{2j}(k) = \inf_{i \geq k}\{d_{2j-1}(i) + G(M_i)\}$$

$$l_{2j}(k) = \{\min l \quad : \quad d_{2j}(k) = d_{2j-1}(l) + G(M_l)\}$$

$$d_{2j+1}(k) = \inf_{i \geq k} \{d_{2j}(i+1) - G(m_i)\}$$

$$l_{2j+1}(k) = \{\min l \quad : \quad d_{2j+1}(k) = d_{2j}(l+1) - G(m_l)\} \tag{21}$$

The solutions are:

$$S_n^{(0)} = \{M_{l_n(1)}, \ldots, m_{l_1(l_2(\ldots(l_n(1))\ldots)}\}$$

$$U_0(n) = n + \frac{\alpha}{2}[d_n(1) + \sum_j \Phi_{k_0}(g_j)] \tag{22}$$

For n odd:

$$d_1'(k) = \inf_{i \geq k} \{G(M_i)\}$$

$$d_{2j}'(k) = \inf_{i \geq k} \{d_{2j-1}'(i+1) - G(m_i)\}$$

$$d_{2j+1}'(k) = \inf_{i \geq k} \{d_{2j}'(i) + G(M_i)\} \tag{23}$$

with the corresponding definitions for $l_j'(k)$. The solutions are:

$$S_n^{(0)} = \{M_{l_n'(i)}, \ldots, M_{l_1',(\ldots(l_n',(1))\ldots)}\}$$

$$U_0(n) = n + \frac{\alpha}{2}[d_n'(1) - G(N) + \sum_j \Phi_{k_0}(g_j)] \tag{24}$$

The case for which $m_1 < M_1$ is treated in a similar way.

The recursions (15), (18), (21) and (23), together with equations (9) and (10), allow us to compute the sequences $\{S_1, S_2, \ldots\}$ and $\{U_1^*, U_2^*, \ldots\}$ using only one dimensional optimizations. We now turn to the problem of determining the optimal value n^* for the number of boundaries.

2.3. Stopping Criterion.

In this section we prove the following:

Theorem 2. Suppose that every (constrained) optimal configuration in the sequence $\{S_1, S_2, \ldots\}$ is unique (i.e., for every n, if $S_n' \neq S_n$, and $S_n' \subseteq P$, then $U(n, S_n') > U_n^*$) and that for some n, $U_{n+2}^* > U_n^*$. Then, $U_{n+2k}^* > U_n^*$, for all $k \geq 1$.

This result will provide us with an efficient stopping criterion for the dynamic programming recursions described in the previous section; since the first local minima for the subsequences $\{U_1^*, U_3^*, \ldots\}$ and $\{U_2^*, U_4^*, \ldots\}$ are the global ones, we can terminate the computations once we have found them.

To prove the theorem, we will need the following lemmas:

Lemma 1. Let $S_k = \{L_1, \ldots, L_k\}$ and $S_{k+2} = \{L'_1, \ldots L'_{k+2}\}$ be the optimal boundaries (with corresponding configurations f_k and f_{k+2}) for $n = k$ and $n = k+2$, respectively. Suppose that $k + 2 < |P|$. Then, $S_k \subseteq S_{k+2}$ (i.e., S_{k+2} is a refinement of S_k), provided S_k is unique.

Proof:

We will assume that for some j, $L_j \in S_k - S_{k+2}$, and arrive at a contradiction. We consider three cases:

Case 1: Suppose that for some i,

$$[L'_i, L'_{i+1}] \bigcap S_k = \emptyset$$

In this case, we claim that we can find some index p such that

$$[L'_p, L'_{p+1}] \bigcap S_k = \emptyset$$

and

$$f_{k+2}((L'_p, L'_{p+1}]) \neq f_k((L'_p, L'_{p+1}])$$

Suppose that this is not the case. Then, L'_i, L'_{i+1} are the only elements of S_{k+2} in some interval (L_j, L_{j+1}) (or in one of the extreme intervals $[1, L_1), (L_k, N]$) and

$$f_{k+2}((L'_i, L'_{i+1}]) \neq f_k((L'_i, L_{i+1}])$$

Suppose

$$[L'_i, L_{i+1}] \subseteq (L_j, L_{j+1})$$

By condition (13), we have that $L_j \neq L'_{i-1}$ (otherwise, L_j would be a local maximum and minimum of G at the same time). But then, since S_k is optimal, we can find a configuration with $k+2$ boundaries whose energy is lower than that of S_{k+2}, by moving L'_i to L_j (or L'_{i+1} to L_{j+1}), which contradicts the optimality of S_{k+2}. A similar argument holds if

$$[L'_i, L'_{i+1}] \subseteq [1, L_1) \text{ or } (L_k, N]$$

This proves our claim.

So, suppose that

$$[L'_p, L'_{p+1}] \bigcap S_k = \emptyset$$

and

$$f_{k+2}((L'_p, L'_{p+1}]) \neq f_k((L'_p, L'_{p+1}]).$$

Form

$$S'_k = \{L'_1, \ldots, L'_{p-1}, L'_{p+2}, \ldots, L'_{k+2}\}$$

and let $f_k^{'}$ be the corresponding configuration, chosen in such a way that $f_k^{'}(1) = f_k(1)$ (and therefore, $f_k^{'}([L_p^{'}, L_{p+1}^{'}]) = f_k([L_p^{'}, L_{p+1}^{'}])$).

Let $\Delta\hat{U}$ be the change in $\hat{U}$ (see eq. (4)) associated with setting:

$$f([L_p^{'}, L_{p+1}^{'}]) = f_{k+2}([L_p^{'}, L_{p+1}^{'}]).$$

We have that

$$\hat{U}(S_{k+2}) = \hat{U}(S_k^{'}) + \Delta\hat{U}.$$

Now, we put:

$$S_{k+2}^{'} = \{L_1, \ldots, L_j, L_p^{'}, L_{p+1}^{'}, \ldots, L_k\}.$$

Since S_k is optimal, we have that:

$$\hat{U}(S_{k+2}) = \hat{U}(S_k^{'}) + \Delta\hat{U} > \hat{U}(S_k) + \Delta\hat{U} = \hat{U}(S_{k+2}^{'}),$$

which contradicts the optimality of S_{k+2}.

Case 2:

$$([1, L_1^{'}] \bigcup [L_{k+2}^{'}, N]) \bigcap S_k = \emptyset$$

Suppose that $L_1^{'} \in [1, L_1^{'})$. We must have

$$f_{k+2}([1, L_1^{'})) \neq f_k([1, L_1^{'}))$$

Otherwise, if $L_1 = L_2^{'}$, condition (13) generates a contradiction; if $L_1 > L_2^{'}$, we are in case 1, and if $L_1 < L_2^{'}$, S_{k+2} is not optimal, since we get a lower energy configuration by moving $L_1^{'}$ to L_1.

So,

$$f_{k+2}([1, L_1^{'}]) \neq f_k([1, L_1^{'}]).$$

By a similar argument, we get that

$$f_{k+2}([L_{k+2}^{'}, N]) \neq f_k([L_{k+2}^{'}, N]).$$

Now, proceeding as in case 1, we form:

$$S_k^{'} = \{L_2^{'}, \ldots, L_{k+1}^{'}\}$$

and let $f_k^{'}$ be the corresponding configuration, chosen in such a way that $f_k^{'}(1) = f_k(1)$

Let $\Delta\hat{U}$ be the change in $\hat{U}$ associated with setting:

$$f([L_1^{'}, L_2^{'}]) = f_{k+2}([L_1^{'}, L_2^{'}]) \quad \text{and}$$

$$f([L'_{k+1}, L'_{k+2}]) = f_{k+2}([L'_{k+1}, L'_{k+2}])$$

so that

$$\hat{U}(S_{k+2}) = \hat{U}(S'_k) + \Delta\hat{U}.$$

Now, we form:

$$S'_{k+2} = \{L'_1, L_1, \ldots, L_k, L'_{k+2}\},$$

Since S_k is optimal, we have that:

$$\hat{U}(S_{k+2}) = \hat{U}(S'_k) + \Delta\hat{U} > \hat{U}(S_k) + \Delta\hat{U} = \hat{U}(S'_{k+2}),$$

which again contradicts the optimality of S_{k+2}.

Case 3:

$$\text{For all } i, \quad [L'_i, L'_{i+1}] \bigcap S_k \neq \emptyset,$$

$$\text{and} \quad ([1, L'_1] \bigcup [L'_{k+2}, N]) \bigcap S_k \neq \emptyset \qquad (*)$$

To make (*) hold, we must be able to place k boundaries in $k+3$ (ovelapping) closed intervals, without omitting any interval. Moreover, since condition (13) must hold, we cannot put $L_j = L'_i$ and $L_{j+1} = L'_{i+2}$ for any i, j. But this is impossible; so, our proof is finished. ∎

Lemma 2. Let $\Delta\hat{U}_k = \hat{U}(S_k) - \hat{U}(S_{k+2})$. Then, $\Delta\hat{U}_k \leq \Delta\hat{U}_{k-2}$, for all $k \in [3, |P|-2]$.

Proof:

Consider the optimal configurations S_k, S_{k+2}, S_{k+4}, and suppose that $\Delta\hat{U}_{k+2} > \Delta\hat{U}_k$. Using lemma 1, let

$$S_k = \{L_1, \ldots, L_k\};$$

$$S_{k+2} = \{L_1, \ldots, L'_1, L'_2, \ldots, L'_k\}.$$

By condition (13) and lemma 1, there are only two valid forms for S_{k+4}. We consider each case separately:

Case 1: S_{k+4} is of the form:

$$S_{k+4} = \{L_1, \ldots, L_p, L'_1, L'_2, L_{p+1} \ldots, L''_1, L''_2, \ldots\}$$

(i.e., the refinements corresponding to S_{k+2} and S_{k+4} are disjoint).

Then, for

$$S'_{k+2} = \{L1, \ldots, L_p, L_{p+1}, \ldots, L''_1, L''_2, \ldots\},$$

we have

$$\hat{U}(S'_{k+2}) = \hat{U}(S_k) - \Delta\hat{U}_{k+2} < \hat{U}(S_k) - \Delta\hat{U}_k = \hat{U}(S_{k+2}),$$

which is a contradiction.

Case 2: S_{k+4} is of the form:

$$S_{k+4} = \{L_1, \ldots, L_j, L_1^{'}, L_1^{''}, L_2^{''}, L_2^{'}, \ldots\}$$

(i.e., S_{k+4} is a subrefinement of the refinement introduced by S_{k+2}).
Let

$$a = -\hat{U}(\{L_1, \ldots, L_j, L_1^{'}, L_1^{''}, L_{j+1}, \ldots\} + \hat{U}(S_k)$$
$$b = \hat{U}(\{L_1, \ldots, L_j, L_1^{''}, L_2^{''}, L_{j+1}, \ldots\} - \hat{U}(S_k)$$
$$c = -\hat{U}(\{L_1, \ldots, L_j, L_2^{''}, L_2^{'}, L_{j+1}, \ldots\} + \hat{U}(S_k)$$

We have that,

$$\Delta\hat{U}_k = a + c - b$$
$$\Delta\hat{U}_{k+2} = b.$$

By assumption,

$$b > a + c - b$$

and therefore,

$$\Delta\hat{U}_k = a + c - b < \frac{a+c}{2} \leq \max(a, c).$$

Now, let $S_{k+2}^{'}$ be formed from S_k by the refinement:

$$\begin{cases} L_1^{'}, L_1^{''}, & \text{if } a = \max(a, c) \\ L_2^{''}, L_2^{'}, & \text{if } c = \max(a, c) \end{cases}$$

Then,

$$\hat{U}(S_{k+2}^{'}) = \hat{U}(S_k) - \max(a, c) < \hat{U}(S_k) - \Delta\hat{U}_k = \hat{U}(S_{k+2}),$$

which is a contradiction.∎

Now we prove theorem 2:
Suppose $U_{k+2}^{*} > U_k^{*}$. Then,

$$k + 2 + \frac{\alpha}{2}\hat{U}(S_{k+2}) > k + \frac{\alpha}{2}\hat{U}(S_k)$$

now, by lemma 2 we have:

$$U_{k+4}^{*} = k + 4 + \frac{\alpha}{2}\hat{U}(S_{k+4}) = k + 4 + \frac{\alpha}{2}(\hat{U}(S_k) - \Delta\hat{U}_{k+2}) >$$

$$> k+2+\frac{\alpha}{2}(\hat{U}(S_k)-\Delta\hat{U}_{k+2}) > k+2+\frac{\alpha}{2}(\hat{U}(S_k)-\Delta\hat{U}_k) =$$

$$= k+2+\frac{\alpha}{2}\hat{U}(S_{k+2}) = U^*_{k+2} \quad \blacksquare$$

2.4. Expected Value of n^*.

First, we compute the (prior) probability density function $p(n)$ for the number n of odd bonds in the original field f.

Let $N_b = N-1$ be the total number of bonds. We can rewrite equation (1) as:

$$P(\omega = f) = \frac{1}{Z}e^{\frac{1}{\alpha}(N_b-2n)} \tag{25}$$

The total number of configurations compatible with a given n is $2C_n^{N_b}$, and so,

$$p(n) = \frac{2C_n^{N_b}\exp[\frac{1}{\alpha}(N_b-2n)]}{\sum_{k=0}^{N_b} C_k^{N_b}\exp[\frac{1}{\alpha}(N_b-2k)]} =$$

$$= C_n^{N_b}\left(\frac{e^{1/\alpha}}{e^{1/\alpha}+e^{-1/\alpha}}\right)^{N_b-n}\left(\frac{e^{-1/\alpha}}{e^{1/\alpha}+e^{-1/\alpha}}\right)^{n} \tag{26}$$

which is a binomial distribution. Therefore,

$$E[n] = N_b\left(\frac{e^{-1/\alpha}}{e^{1/\alpha}+e^{-1/\alpha}}\right)$$

$$\mathrm{Var}[n] = N_b\left(\frac{1}{e^{1/\alpha}+e^{-1/\alpha}}\right) \tag{27}$$

We note that as $\alpha \uparrow \infty$, $E[n] \uparrow N_b/2$, and as $\alpha \downarrow 0$, $E[n] \downarrow 0$ (and $\mathrm{var}[n] \downarrow 0$) exponentially fast. This means that if the natural temperature of the system is not too high, we can expect that n^*, the MAP estimate for n, to be relatively small.

2.5. Relation to Multiscale Filtering.

An interesting characteristic of the DP formulation is that the solutions to each of the subproblems (which in fact correspond to a minimization of $\hat{U}$ (eq. (4)) are independent of the value of the parameter α. The role of this parameter is to determine the number of regions (n^*) that will be present in the optimal configuration. In this sense, it can be regarded as a "scale" parameter that controls the aggregation of the subregions into larger units, and the algorithm can be used to produce multiscale descriptions (in the style of the "fingerprints" treated in [W1,Y1])

of the input signals (several heuristic solutions to this problem have been proposed; see for example [B3,P2,P3]).

If we interpret the algorithm in this way, it becomes natural to ask whether a family of linear operators can do the same job in a much cheaper way. Let us formulate this question in more precise form (in what follows, we will consider a "continuous time" problem obtained from the original one as a limit when $N \uparrow \infty$ (provided that the observations are different from 0 only in a finite interval), since it simplifies the notation. It should be clear that the same arguments apply to the discrete case).

Consider a family of filters $\{F_L\}$ with the following properties:

(i) Each $F_L(x)$ is a symmetric and non-negative function of x.

(ii) For each L, $F_L(x)$ is a decreasing function of $|x|$, and $F_L(x) \downarrow 0$ as $|x| \uparrow \infty$ fast enough, so that F_L can be approximated by a function with finite support.

(iii) All the filters are normalized:

$$\int_{-\infty}^{\infty} F_l(x)dx = 1, \quad \text{for all } L.$$

(iv) The filters become sharper as $L \downarrow 0$:

$$\int_0^b F_{L_2}(x)dx < \int_0^b F_{L_1}(x)dx$$

implies that $L_2 > L_1$

Particular examples of acceptable families are:

(i) The family of rectangular boxes B_L:

$$B_L(x) = \begin{cases} \frac{1}{2L}, & \text{if } |x| \leq L \\ 0, & \text{otherwise} \end{cases}$$

(ii) The family of Gaussian Kernels:

$$G_L(x) = \frac{1}{\sqrt{2\pi}L} \exp[-\frac{x}{2L^2}]$$

Suppose we convolve the function $g(x) - \frac{1}{2}$ ($g(x)$ is a continuous time approximation to the observations) with a set of filters from the family $\{F_L\}$. If we start with L large enough, the function

$$h_L = (g - \frac{1}{2}) \ * \ F_L$$

will be practically constant, and therefore, it will have no zeroes. As we decrease L, zero crossings of h_L will begin to appear. To each of these zero crossings, we will associate a boundary, and form the configurations $\hat{S}_1, \hat{S}_2, \ldots$ with $1, 2, \ldots$ boundaries respectively, that correspond to the first, first two, etc. zero crossings of h_L (we are ignoring, at this point, the question of the precise localization of these boundaries. With additional contraints on the family $\{F_L\}$, it is possible, in principle, to localize them by decreasing L in a continuous fashion, and then tracing the position of each zero crossing to the finest ($L = 0$) level; see [Y1]. For the moment, let us assume that we can identify the zero crossings of $g - \frac{1}{2}$ that correspond to those of h_L, for all L).

The question that we ask is the following:

If $S_1, S_2, \ldots$ are the optimal boundary configurations produced by the DP algorithm,is it true that

$$S_k = \hat{S}_k$$

for all k?

As we now show, this is not the case.

Consider the signal $g(x)$ defined by:

$$g(x) = 1 \quad ,$$

$$\text{for } x \in [l_1, l_1 + 2a] \bigcup [l_2, l_2 + 2b] \bigcup [l_2 + 4b, l_2 + 6b] \bigcup$$
$$\bigcup [l_2 + 8b, l_2 + 10b] \bigcup [l_2 + 12b, l_2 + 14b] \bigcup [l_2 + 16b, l_2 + 18b] \quad ,$$

and $g(x) = 0$, otherwise. Here, l_1, l_2, a and b are some positive numbers chosen in such a way that, if L_0 is the starting L, we take $l_2 - l_1 - a >> L_0$, so that, by property (ii), there is no interaction between $[l_1, l_1 + a]$ and $[l_2, l_2 + 18b]$ (see figure 1).

Suppose that the zero crossings corresponding to $[l_1, l_1 + a]$ appear first (as a single double zero) at $L = L_1$, and those corresponding to $[l_2, l_2 + 18b]$ at $L = L_2$. Then,

$$\int_0^a F_{L_1}(x)dx = \int_a^\infty F_{L_1}(x)dx \tag{28}$$

$$\int_0^b F_{L_2}(x)dx + \int_{3b}^{5b} F_{L_2}(x)dx + \int_{7b}^{9b} F_{L_2}(x)dx =$$
$$= \int_b^{3b} F_{L_2}(x)dx + \int_{5b}^{7b} F_{L_2}(x)dx + \int_{9b}^{\infty} F_{L_2}(x)dx \tag{29}$$

Now, for $a > b$, we have:

$$\hat{U}(\{l_1, l_2\}) = 10b$$
$$\hat{U}(\{l_3, l_4\}) = 8b + 2a > \hat{U}(\{l_1, l_2\})$$

and therefore, $S_2 = \{l_1, l_2\}$.

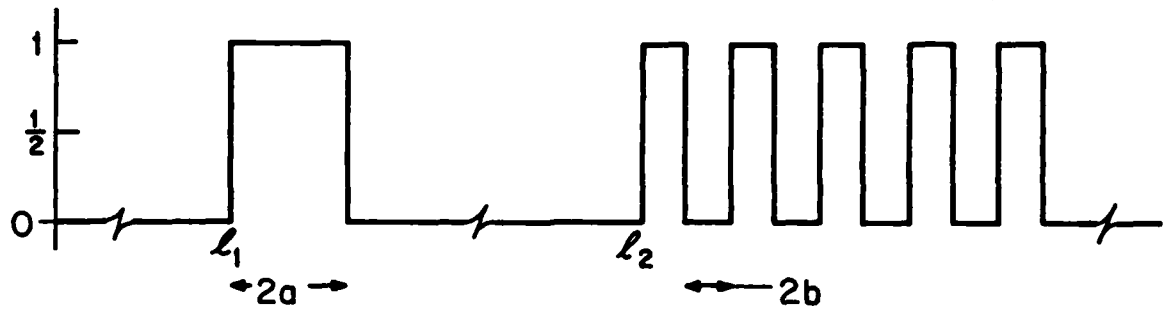


Figure 1. (See text).

We claim that we can find some a, b with $a > b$ such that

$$\int_0^a F_{L_2}(x)dx < \int_a^\infty F_{L_2}(x)dx$$

If this is true, we find, using (28) and conditions (iii) and (iv), that it implies that $L_2 > L_1$, and therefore, $\hat{S}_2 = \{l_3, l_4\}$.

We now prove our claim:

Let $a = b + \frac{\epsilon}{2}$, where we choose ϵ so that

$$\int_b^{b+\epsilon/2} F_{L_2}(x)dx = \int_{3b}^{5b} F_{L_2}(x)dx \tag{30}$$

(property (ii) guarantees that we can find such ϵ). From (29),

$$\int_b^\infty F_{L_2}(x)dx = \int_0^b F_{L_2}(x)dx + 2\int_{3b}^{5b} F_{L_2}(x)dx + 2\int_{7b}^{9b} F_{L_2}(x)dx$$

and from (30),

$$\int_a^\infty F_{L_2}(x)dx = \int_{b+\epsilon/2}^\infty F_{L_2}(x)dx = \int_b^\infty F_{L_2}(x)dx - \int_b^{b+\epsilon/2} F_{L_2}(x)dx =$$

$$= \int_0^b F_{L_2}(x)dx + \int_b^{b+\epsilon/2} F_{L_2}(x)dx + 2\int_{7b}^{9b} F_{L_2}(x)dx = \int_0^{b+\epsilon/2} F_{L_2}(x)dx + 2\int_{7b}^{9b} F_{L_2}(x)dx >$$

$$> \int_{7b}^{9b} F_{L_2}(x)dx = \int_0^a F_{L_2}(x)dx \quad \blacksquare$$

This result does not mean, of course, that families of linear filters cannot be used for producing useful multiscale descriptions of signals; it only means that these descriptions cannot, in general, be considered as MAP estimates of MRF models.

It is possible, however, to design non-linear methods that are guaranteed to find optimal estimates, and that are computationally much more efficient (although less flexible) than the DP algorithm. We will present one such method in section 3.

2.6. Extensions.

In this section we present two related problems which can, in principle, be solved using the DP approach, although, as we will see, in a less efficient way.

2.6.1. Continuous Valued Markov Random Fields.

Let us consider the problem of estimating a piecewise constant signal corrupted by additive white Gaussian noise. We model the signal $\{f_i\}$ as a MRF with potential

$$V(f_i, f_{i+1}) = \begin{cases} 1, & \text{if } f_i = f_{i+1} \\ -1, & \text{otherwise} \end{cases} \tag{31}$$

and global states distributed according to (1).

The observations are given by:

$$g_i = f_i + n_i$$

where n is a white Gaussian process. The Bayesian (MAP) estimate for f is again found by minimizing eq.(4):

$$U(f) = n + \frac{\alpha}{2}\hat{U}$$

$$\hat{U} = \sum_{i+1}^{N}(f_i - g_i)^2$$

where n is the number of places where $f_i \neq f_{i+1}$, and $\alpha = \frac{\alpha}{2\sigma^2}$. Note that in this case, f_i is not restricted to $\{0, 1\}$, but can take any real value.

Proceeding as we did in section 2, we consider the sequence of subproblems obtained by putting $n = 0, 1, 2, \ldots$.

For any fixed n, $\hat{U}$ will depend only on the n integer variables that correspond to the location of the boundaries between regions of constant f, since given these

boundaries $\mathcal{L} = \{L_1, \ldots L_n\}$, the optimal estimate for f on any interval $(L_i, L_{i+1}]$ (we put $L_0 = 1$ and $L_{n+1} = N$) is:

$$f((L_i, L_{i+1}]) = \frac{1}{L_{i+1} - L_i} \sum_{j=L_i+1}^{L_{i+1}} g_j.$$

If we define $G_{k,l}$ (for $k < l$) as:

$$G_{k,l} = (1 - 2(l-k))\left(\frac{1}{l-k} \sum_{i=k}^{l} g_i\right)^2 \tag{32}$$

We get that:

$$\hat{U}(\mathcal{L}_n) = \sum_{i=1}^{N} g_i^2 + \sum_{j=1}^{n+1} G_{L_{j-1}, L_j} \tag{33}$$

(note that $\sum g_i^2$ is a constant for a given set of observations). Using dynamic programming principles, we can now write the recursions:

$$F_0(k) = G_{k,N}$$

$$F_{j+1}(k) = \inf_{i>k} \{G_{k,i} + F_j(i)\}$$

$$L_{j+1}(k) = \{L \quad : \quad G_{k,L} + F_j(L) = F_{j+1}(k)\} \tag{34}$$

The optimal solution, for each given n is:

$$S_n = \{L_n(1), L_{n-1}(L_n(1)), \ldots, L_1(L_2(\ldots(L_n(1))\ldots)\}$$

and the corresponding energy,

$$U(n, S_n) = n + \frac{\alpha}{2}[\sum_{i=1}^{N} g_i^2 + F_n(1)] \tag{35}$$

The solution to our problem will be S_{n^*}, where:

$$U(n^*, S_{n^*}) = \inf_n \{U(n, S_n)\} \tag{36}$$

Unfortunately, in this case we cannot guarantee the unimodality of any subsequence of $\{\hat{U}(S_n)\}$ (although we believe that the sequence will be unimodal in many cases) and so, (36) has to be computed, in principle, by an exhaustive one dimensional search. Another unpleasantness is that, unlike the binary case, the search space for the variables L_i cannot be reduced in any obvious way.

2.6.2. One Dimensional Signal Matching.

Another problem for which this formulation may be useful is the following:

Let f be a one dimensional MRF with potential given by (31), that can take values on the set

$$Q = \{-m, -m+1, \ldots, -1, 0, 1, \ldots, m\}$$

for some positive integer m. Suppose we observe two binary sequences g_R, g_L which are formed from f according with the following stochastic model:

$$g_R(i) = A_\rho(i)$$

$$g_L(i) = \begin{cases} g_R(i+f_i), & \text{with prob. } 1-\epsilon, \text{ if } \phi_f(i) = 0 \\ B_\rho(i), & \text{with prob. } \epsilon, \text{ if } \phi_f(i) = 0 \\ B_\rho(i), & \text{with prob. } 1, \text{ if } \phi_f(i) = 1 \end{cases} \tag{37}$$

Here, A_ρ and B_ρ are independent Bernoulli processes (also independent from f) with density ρ (so, $A_\rho(i), B_\rho(i) \in \{0,1\}$); $\epsilon \in (0,1)$ is the error rate, and ϕ_f is an "occlusion indicator" whose value depends deterministically on f in the following way:

$$\phi_f(i) = \begin{cases} 1, & \text{if } f_{i-k} \geq f_i + k, \text{for some integer } k \in [0,m] \\ 0, & \text{otherwise} \end{cases} \tag{38}$$

A well known instance of this problem is the matching of a row of a random dot stereogram with density ρ [J1], when the components of the stereo pair are corrupted by noise. In this case, the use of a MRF model for the disparity f corresponds to a quantification of the assumption of the existence of "dense solutions" ([J1]; see also [M3]), and the use of the occlusion indicator corresponds to the "ordering constraint" [B2].

To formulate the estimation problem, we will consider the sequence g_L as "observations", while g_R will play the role of a set of parameters. Thus, we have (assuming, for simplicity that $\rho = \frac{1}{2}$):

$$P(g_L(i) = k \mid f, g_R) = P_{g|f}(k) =$$

$$= \begin{cases} 1-\epsilon, & \text{if } \phi_f(i) = 0 \text{ and } g_R(i+f_i) = k \\ \epsilon, & \text{if } \phi_f(i) = 0 \text{ and } g_R(i+f_i) \neq k \\ \frac{1}{2}, & \text{if } \phi_f(i) = 1 \end{cases}$$

Putting:

$$\Psi_r(i) = -\ln(1-\epsilon)\delta(g_L(i) - g_R(i+r)) -$$

$$-\ln\epsilon(1 - \delta(g_L(i) - g_R(i+r))$$

where

$$\delta(x) = \begin{cases} 1, & \text{if } x = 0 \\ 0, & \text{otherwise} \end{cases}$$

we get that the MAP estimate for f is obtained by minimizing

$$U(f) = n - \frac{\beta}{2} \sum_i \ln P_{g|f}(g_L(i)) =$$

$$= n + \frac{\beta}{2} \sum_i [\ln 2\phi_f(i) + (1 - \phi_f(i))\Psi_{f_i}(i)]. \tag{39}$$

The use of the DP algorithm for minimizing (39) is complicated by the fact that, given the boundaries $\mathcal{L}_n$, the optimal estimate for f in the interval $(L_i, L_{i+1}]$ depends on the estimate on $(L_{i-1}, L_i]$, since this last choice determines the extent of the occluded region.

However, if we assume that the size of the regions of constant disparity is relatively large compared with the size of the occluded areas (as it normally happens in most practical cases), we can estimate f given $\mathcal{L}_n$ using the formula:

$$f((L_i, L_{i+1}]) = \hat{f}(L_i, L_{i+1}) =$$

$$= \{k\colon \sum_{i=L_i+1}^{L_{i+1}} \delta(g_L(i) - g_R(i+k)) > \sum_{i=L_i+1}^{L_{i+1}} \delta(g_L(i) - g_R(i+l)), \text{ for all } l \in Q\} \tag{40}$$

Defining:

$$G_{k,l} = \sum_{i=k+1}^{l} \Psi_{\hat{f}(k,l)}(i) \tag{41}$$

we can write the dynamic programming recursions:

$$F_0(k) = G_{k,N}$$

$$L_0(k) = N$$

$$F_{j+1}(k) = \inf_{i>k} \{G_{k,i} + F_j(k+\Delta) + \Delta \ln 2\}$$

$$L_{j+1} = \{L\colon G_{k,L} + F_j(L+\Delta) + \Delta \ln 2 = F_{j+1}(k)\}$$

$$\text{with } \Delta = \min(0, \quad \hat{f}(k,i) - \hat{f}(i, L_j(i)) \tag{42}$$

The optimal location of the boundaries, for any given n is:

$$S_n = \{L_n(1), L_{n-1}(L_n(1)), \ldots, L_1(L_2(\ldots(L_n(1))\ldots)\}$$

The optimal configuration is computed using (40), and the corresponding energy, using (38) and (39).

3. An Alternative Algorithm.

In this section we present an algorithm that finds the MAP estimate for a binary MRF in time $O(N)$, and using storage which is also $O(N)$, and is, therefore, practically as efficient as it can be.

We consider again a first order MRF F on a one dimensional lattice of length N, but we now assume that $F_i \in \{-1, 1\}$ (there is no loss of generality in this assumption, since any binary process can be brought into this form by a memoryless linear transformation).

Using the model for the observations described in section 1, and reasoning as we did in section 2, we find that the MAP estimation problem is equivalent to the minimization of

$$U(f) = n + \sum_i \Psi_{f_i}(g_i), \qquad f_i \in \{-1, 1\} \tag{43}$$

where n is, as before, the total number of odd bonds in the configuration f, and

$$\Psi_j = \frac{\alpha}{2}\Phi_j$$

We now present a method for performing this minimization, and a proof of its optimal performance.

Description of the Algorithm.

The idea in which this method is based is the following:

We start scanning the sequence $\{g_i\}$, say, from the left, with some initial estimate k for f_1, and set $l_0 = 1$. Whenever we process a new observation g_j, we ask if we can lower the energy by putting a boundary in the best possible location l within the interval $[l_0, j]$. If this is the case, we put the boundary at l, that is:

$$\text{set } f_i = k, \qquad \text{for } i \in [l_0, l]$$

$$\text{set } k = -k,$$

$$\text{and set } l_0 = l + 1.$$

Otherwise, we just set $f_j = f_{j-1}$, and continue to process the next observation.

When we reach g_N, we take f_N as the initial estimate and run the same process backwards (in fact, we can make this backward run as soon as we get the second boundary) to get the final solution.

Formally, the algorithm is as follows:

Definition of Variables.

i: Current position.

l_0: Pointer to the beginning of the current region.

l: Current optimal location of the boundary in the interval $[l_0, i]$.

k: Current estimate for $f([l_0, l])$.

U_p: Energy increment associated with the assignment $f([l_0, i]) = k$.

U_m: Energy increment associated with the assignment $f([l_0, i]) = -k$.

U_b: Energy increment associated with the assignment $f([l_0, l]) = k; f((l, i]) = -k$.

si: Best local (maximum likelihood) estimate for f_i.

$sim1$: Best local (maximum likelihood) estimate for f_{i-1}.

U_{pl}: Energy increment associated with the assignment $f([l_0, l]) = k$.

U_{ml}: Energy increment associated with the assignment $f([l_0, l]) = -k$.

U_{temp}: Temporary storage register.

M: A very large positive number.

K_0: Switch indicating the method for estimating f_1.

Algorithm $A1(K_0)$:

1: Initialization.

Set $l_0 = l = 1; U_p = U_m = U_{ml} = 0; U_b = 1; U_{pl} = M$.

Set $k = 1$, if $K_0 = 0$ and $\Psi_{+1}(g_1) < \Psi_{-1}(g_1)$;
-1, if $K_0 = 0$ and $\Psi_{+1}(g_1) \geq \Psi_{-1}(g_1)$;
K_0, if $K_0 \neq 0$.

Set $sim1 = k$

2: Main Loop: For i from 1 to N do:

Begin

Set $si = 1$, if $\Psi_{+1}(g_i) < \Psi_{-1}(g_i)$;
-1, otherwise.

2.1: See if the optimal boundary location needs to be updated:

If ($si \neq k$ and $si \neq sim1$ and $U_p - U_{pl} - U_m + U_{ml} < 0$) do :

Update boundary location:

Set :

$$l = i - 1$$
$$U_{pl} = U_p$$
$$U_{ml} = U_m$$
$$U_b = U_p + 1$$

2.2: Update energy increments:

Set :

$$U_p = U_p + \Psi_{+1}(g_i)$$
$$U_m = U_m + \Psi_{-1}(g_i)$$
$$U_b = U_b + \Psi_{+1}(g_i)$$

2.3: See if a new boundary has to be introduced:

If $(U_b + 1 < U_p)$ do :

Introduce a new boundary:

For j from l_0 to l do : Set $f_j = k$

Set :

$$
\begin{aligned}
k &= -k \\
l_0 &= l + 1 \\
U_{temp} &= U_p - U_{pl} \\
U_p &= U_m - U_{ml} \\
U_m &= U_{temp} \\
U_{pl} &= M \\
U_b &= U_m + 1
\end{aligned}
$$

2.4: Set $sim1 = si$

End

3: See if the last boundary has to be introduced:

If $(U_b < U_p)$ do :

3.1: For j from l_0 to l set $f_j = k$.

3.2: Set $l_0 = l + 1$.

3.3: Set $k = -k$.

4: Fill the last region:

For j from l_0 to N set $f_j = k$.

End.

3.2. Optimality of Algorithm $A1$.

The optimality of this algorithm follows from the following propositions:

Proposition 1: Let $S^* = \{l_1, \ldots l_n\}$ be the optimal boundary configuration, and suppose that l_k, for $k < n$ was detected by $A1$. Then, l_{k+1} will be the next boundary

detected by $A1$.

Proof:

Suppose l_k was detected by $A1$, and let L be the next boundary detected. We will assume that $L \neq l_{k+1}$ and arrive at a contradiction. We will consider three cases:

Case 1: Suppose $A1$ detects L at $j < l_{k-1}$.

Then, we must have that

$$U_p(j) > U_p(L) + U_m(j) - U_m(L) + 2$$

and therefore,

$$U(\{l_1, \ldots, l_k, L, j, l_{k+1}, \ldots\}) < U(S^*)$$

which is a contradiction.

Case 2: Suppose $A1$ detects L at $j \in (l_{k+1}, l_{k+2}]$.

This means that at j we had that L was the optimal location for the boundary. In particular,

$$U_p(l_{k+1}) + U_m(j) - U_m(l_{K+1}) > U_p(L) + U_m(j) - U_m(L)$$

which implies that

$$U_p(L) + U_m(l_{k+2}) - U_m(L) < U_p(l_{k+1}) + U_m(l_{k+2}) - U_m(l_{k+1})$$

and therefore,

$$U(\{l_1, \ldots, l_k, L, l_{k+2}, \ldots\} < U(S^*)$$

which is a contradiction.

Case 3: Suppose that $A1$ has not detected any new boundary at $j = l_{k+2} + 1$.

Then, we must have:

$$U_p(l_{k+2} + 1) < U_b(l_{k+2} + 1) + 1$$

which means that

$$U(\{l_1, \ldots l_k, l_{k+3}, \ldots\} < U(S^*)$$

which is again a contradiction. ∎

Proposition 2: If $A1$ runs from left to right starting at a point l_0, and generates the boundaries $\{l_1, l_2, \ldots\}$, then, $l_j \in S^*$ (the set of boundaries of the optimal configuration) for $j \geq 2$.

Proof:

Let f^*, f_{A1} be the optimal configuration, and the one generated by $A1$, respectively.

Let

$$L_0 = \sup\{j \in S^* \quad : \quad j < l_1\}$$

$$L = \inf\{j \in S^* \quad : \quad j > l_1\}$$

If $L_0 = l_0$, we apply proposition 1 and finish the proof; so, let us assume that $L_0 \neq l_0$, and that l_1 was detected at i. We have two cases:

Case 1: $L_0 > l_0$. We claim that in this case, $l_1 \in S^*$, and therefore, by proposition 1, $l_j \in S^*$ for $j \geq 1$. To prove this claim, we consider two subcases:

Case 1-a: $f^*((l_0, L_0)) \neq f_{A1}((l_0, L_0))$.

In this case, we have:

$$2 + U_m(i) - U_m(l_1) + U_p(l_1) < U_p(i)$$

and therefore,

$$2 + U_m(i) - U_m(l_1) + U_p(l_1) - U_p(L_0) < U_p(i) - U_p(L_0)$$

which implies that $l_1 \in S^*$.

Case 1-b: $f^*((l_0, L_0)) = f_{A1}((l_0, L_0))$.

Suppose $l_1 \notin S^*$. We have that, at location i,

$$U_p(l_1) + U_m(i) - U_m(l_1) + 2 < U_p(L_0) + U_m(i) - U_m(L_0) + 2$$

since otherwise, L_0 would have been a better location for the boundary. However, this implies that

$$U_p(l_1) + U_m(L) - U_m(l_1) < U_p(L_0) + U_m(L) - U_m(l_1)$$

which means that we can improve S^* by moving L_0 to l_1, which is a contradiction.

Case 2: $L_0 < l_0$.

Again, we consider two subcases:

Case 2-a: $f^*((L_0, l_0)) = f_{A1}((L_0, l_0))$.

Let U_+, U_- be the energy increments with respect to L_0:

$$U_+(i) = \sum_{j=L_0}^{i} \Psi_{+k}(g_j)$$

$$U_-(i) = \sum_{j=L_0}^{i} \Psi_{-k}(g_j)$$

Note that

$$U_p(i) = U_+(i) - U_+(l_0) \qquad \text{and}$$

$$U_m(i) = U_-(i) - U_-(l_0)$$

Since l_1 was detected at i, we have:

$$2 + U_m(i) - U_m(l_1) + U_p(l_1) < U_p(i)$$

and therefore,

$$2 + U_-(i) - U_-(l_1) + U_+(l_1) < U_+(i)$$

which means that $l_1 \in S^*$.

Case 2-b: $f^*((L_0, l_0)) \neq f_{A1}((L_0, l_0))$.

Using the same definitions for U_+, U_-, we have that, by the optimality of S^*, for some $j > L$,

$$U_-(j) - U_-(L) + U_+(L) + 2 < U_+(j)$$

and therefore,

$$U_-(j) - U_-(L) + U_+(L) - U_+(l_1) + 2 < U_+(L) - U_+(l_1)$$

which means that if $A1$ detects l_1, it must detect L too, unless it detected l_2 first, but in this case we have that, for some $p < j$,

$$U_-(p) - U_-(l_2) + U_+(l_2) - U_+(l_1) + 2 < U_+(p) - U_+(l_1)$$

which implies that $l_2 \in S^*$. This completes the proof. ∎

It should be clear that these results can be easily extended to the case where $A1$ runs backwards (from right to left). With this extension, we get the following complete optimal procedure:

Algorithm $A2$:

1: Run $A1$ from left to right. Detect $\{l_1, \ldots, l_n\}$.

2: Run $A1$ backwards (starting from l_2. Get either

$$\{l_2, \ldots, l_n\} \quad \text{or} \quad \{l_1', l_2, \ldots, l_n\}$$

In either case, this is the optimal solution.

The only thing that remains to be proved is that the determination of the optimal location for a boundary is in fact performed by step 2.1 of $A1$. We have the following:

Proposition 3: Suppose that $A1$ detected a boundary at (or started from) l_0. Then, the optimal location l of the next boundary has to be updated only at places where

$si = -k$ and $sim1 = k$. Suppose i is one such place. The optimal location will be:

$$l = \begin{cases} i-1, & \text{if } U_p(i-1) - U_m(i-1) < U_{pl} - U_{ml} \\ l, & \text{(the current value) otherwise} \end{cases}$$

Proof:

First, we note that a necessary and sufficient condition for l to be the optimal location of the boundary at the point i is that, for $j \in [l_0, i-1]$:

$$U_p(l) + U_m(i) - U_m(l) \leq U_p(j) + U_m(i) - U_m(j)$$

or equivalently,

$$U_p(l) - U_m(l) \leq U_p(j) - U_m(j)$$

Suppose l was the optimal location at $i-1$, and we process observation i. We consider several cases:

Case 1: $sim1 = -k$

In this case, we show that l remains the optimal location:

By construction, we have that:

$$U_p(i-1) = U_p(i-2) + \Psi_{+k}(g_{i-1})$$

$$U_m(i-1) = U_m(i-2) + \Psi_{-k}(g_{i-1})$$

Since $sim1 = -k$ we have that,

$$\Psi_{+k}(g_{i-1}) - \Psi_{-k}(g_{i-1}) > 0$$

and therefore,

$$U_p(i-1) - U_m(i-1) = U_p(i-2) - U_m(i-2) + \Psi_{+k}(g_{i-1}) - \Psi_{-k}(g_{i-1}) >$$

$$> U_p(i-2) - U_m(i-2) \geq U_p(l) - U_m(l)$$

so that l remains the optimal location.

Case 2: $sim1 = k$

In this case we have that

$$U_p(i-1) - U_m(i-1) < U_p(i-2) - U_m(i-2)$$

This means that the minimal value for $U_p(i) - U_m(i)$ on a block for which $si = k$ will be obtained at the extremal point where $si = -k$ and $sim1 = k$, and since, by theorem 1, this is the only point where a boundary might be placed, it is sufficient to update the optimal location only at these points. So, suppose $sim1 = k$ and $si = -k$.

If

$$U_{pl} - U_{ml} < U_p(i-1) - U_m(i-1),$$

then,

$$U_{pl} - U_{ml} < U_p(j) - U_m(j) \text{ for } j \in [l_0, i-1]$$

because l was the optimal location outside the last block where $si = k$. By the same token, it is clear that if

$$U_{pl} - U_{ml} \geq U_p(i-1) - U_m(i-1),$$

the new optimal location will be $i-1$.∎

3.3. Experimental Evaluation of Simulated Annealing.

Once we have an efficient method for getting the optimal estimate for the process f, it is interesting to use it to evaluate the performance of other algorithms, such as simulated annealing (see [K1] for a description of the method, and [G1,M1] for examples of its application to MRF estimation problems).

In order to make this comparison, we implemented both algorithms in a computer, and performed some numerical experiments. The field f was generated using Metropolis algorithm [M2,G1,G2], and the observations g by adding to f an independent white Gaussian random process of given power spectral density σ^2. For the annealing schedule, we used the formula (see [G1]):

$$T = \frac{k \cdot \beta \ln 2}{\ln(j+1)} \tag{44}$$

where β is the natural temperature of the field f; T is the annealing temperature, and j is the iteration number. By a trial and error procedure we found that the optimal value for the constant k was 0.5.

Figure 2 shows the results of a typical experiment for a lattice of 50 points (we used $\beta = 2$ and $\sigma = 0.8$ for the parameters of the processes). The top row shows the original field f (black squares mean $f_i = 1$, and white ones, $f_i = 0$); the second row, the maximum likelihood estimate obtained by

$$f_{ML}(i) = \begin{cases} 1, & \text{if } g_i - \frac{1}{2} \geq 0 \\ 0, & \text{otherwise} \end{cases}$$

The third row is the optimal estimate obtained by $A2$, and the last row, the result obtained by simulated annealing after 50 global iterations. The corresponding values for the energy (equation (2)) were:

$$U(f) = 52.083$$

$$U(f_{ML}) = 43.38$$

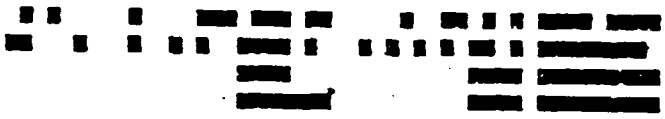


Figure 2. (see text).

$$U(f_{A2}) = 29.34$$

$$U(f_{SA}) = 29.5$$

It is interesting to note that, for this problem, the convergence of the simulated annealing algorithm is extremely fast. We can get a near optimal (and in many cases optimal) solution in less than 10 iterations, with the appropriate setting of the constant k in equation (44).

4. Discussion.

We have presented two deterministic algorithms for finding the optimal (MAP) estimate of a binary, one dimensional MRF from noisy observations.

$A2$, the algorithm presented in section 3, is without doubt the most efficient, and its complexity ($O(N)$) is certainly optimal for this problem. However, it is very difficult to extend it, so that it can be applied to other related problems (such as those presented in section 2.6) which, in principle, can be solved with the dynamic programming approach that we presented in section 2. In these cases, however, the absence of results of the type of theorems 1 and 2 (which provided us with a substantial reduction of the search space, and with an efficient stopping criterion in the binary case) make the application of the DP algorithm computationally more expensive, although still perfectly feasible. It would be interesting to implement these algorithms and use them (as we did in section 3.3 for the binary case) as benchmarks for the evaluation of algorithms whose optimal performance is still

uncertain, such as the proposed extensions to simulated annealing for handling continuous-valued variables [G2].

An important open question is whether efficient deterministic algorithms can be designed for the estimation of two dimensional MRF's. A direct extension of the techniques we have presented here is not posible; the main difficulty in the two dimensional case is that the geometry of the boundaries between uniform regions (which in the one dimensional case are simply points), causes a combinatorial explosion of the number of possible configurations compatible with a given total boundary length. Our results, however, show that in principle it is possible to exploit the structure of the energy function of a particular class of estimation problems to design efficient algorithms for its global minimization.

Acknowledgements: I want to thank Prof. Sanjoy K. Mitter for his encouragement and enthusiasm, and for many valuable ideas which he contributed to this work.

Notes:

[1] In the language of statistical mechanics, the "state" of a system at a given temperature is a probability measure (the Gibbs measure) defined on the phase space of the system (in our case, $\{-1,1\}^N$). Since at zero temperature the Gibbs measure becomes a delta function at the global minimum of the corresponding energy function (assuming it is unique), the global minimizer f^* completely specifies the ground state.

[2] Since we are using a white noise model, the maximum likelihood estimate for f is obtained by the independent maximization of each term of the likelihood function:

$$\ln[Z_g P(g \mid f)] = -\sum_j \Phi_{f_j}(g_j), \quad f_j \in \{k_0, k_1\}.$$

References.

[B1] Besag J. "Spatial Interaction and the Statistical Analysis of Lattice Systems". J. Royal Stat. Soc. B 34 75-83 (1972).

[B2] Baker H.H. and Binford T.O. "Depth from Edge and Intensity Based Stereo" Seventh International Joint Conference on Artificial Intelligence, August 1981, 631-636.

[B3] Blumenthal A.F., Davis L.S. and Rosenfeld A. "Detecting Natural Plateaus in One-Dimensional Patterns. IEEE Trans. on Computers p. 178-179. Feb., 1974.

[C1] Cross G.C. and Jain A.K. "Markov Random Field Texture Models". IEEE Trans. PAMI 5 25–39 (1983).

[E1] Elliot H., Derin R., Christi R. and Geman D. "Application of the Gibbs Distribution to Image Segmentation". Univ. of Massachusetts Technical report (1983).

[G1] Geman S. and Geman D. "Stochastic Relaxation, Gibbs Distribution, and the Bayesian Restoration of Images". Unpublished manuscript. Brown University (1983).

[G2] Grenander U. "Analyzing Images". Unpublished Manuscript. Brown University (1984).

[G3] Gidas B. "Non–Stationary Markov Chains and the Convergence of the Annealing Algorithm". Unpublished Manuscript. Rudgers University (1984).

[J1] Julesz B. "Binocular Depth Perception of Computer Generated Patterns" Bell Sys Tech J 39 (1960) 1125-1162.

[K1] Kirkpatrick S. Gelatt C.D. and Vecchi M.P. "Optimization by Simulated Annealing". Science 220 (1983) 671–680.

[K2] Kindermann R. and Snell J.L. "Markov Random Fields and their Applications". Vol 1. Amer. Math. Soc. (1980).

[K3] Kashyap R.L. and Chellappa R. "Estimation and Choice of Neighbors in Spatial Interaction Models of Images". IEEE Trans. on Info. Theory 29 60-72 (1983).

[M1] Marroquin J. "Surface reconstruction Preserving Discontinuities". A.I. Memo 792 and LIDS-P-1402. MIT (1984).

[M2] Metropolis N. et. al. "Equation of State Calculations by Fast Computing Machines". J. Phys. Chem. 21 6 (1953) 1087.

[M3] Marr D. and Poggio T. "A Theory of Human Stereo Vision". Proc. Roy. Soc. Lond. B 204 (1979) 301–328.

[P1] Preston C.J. "Gibbs States on Countable Sets". Camb. Univ. Press (1974).

[P2] Pavlidis T. "Waveform Segmentation Through Functional Approximation" IEEE Trans. on Computers 22, 689-697 (1983).

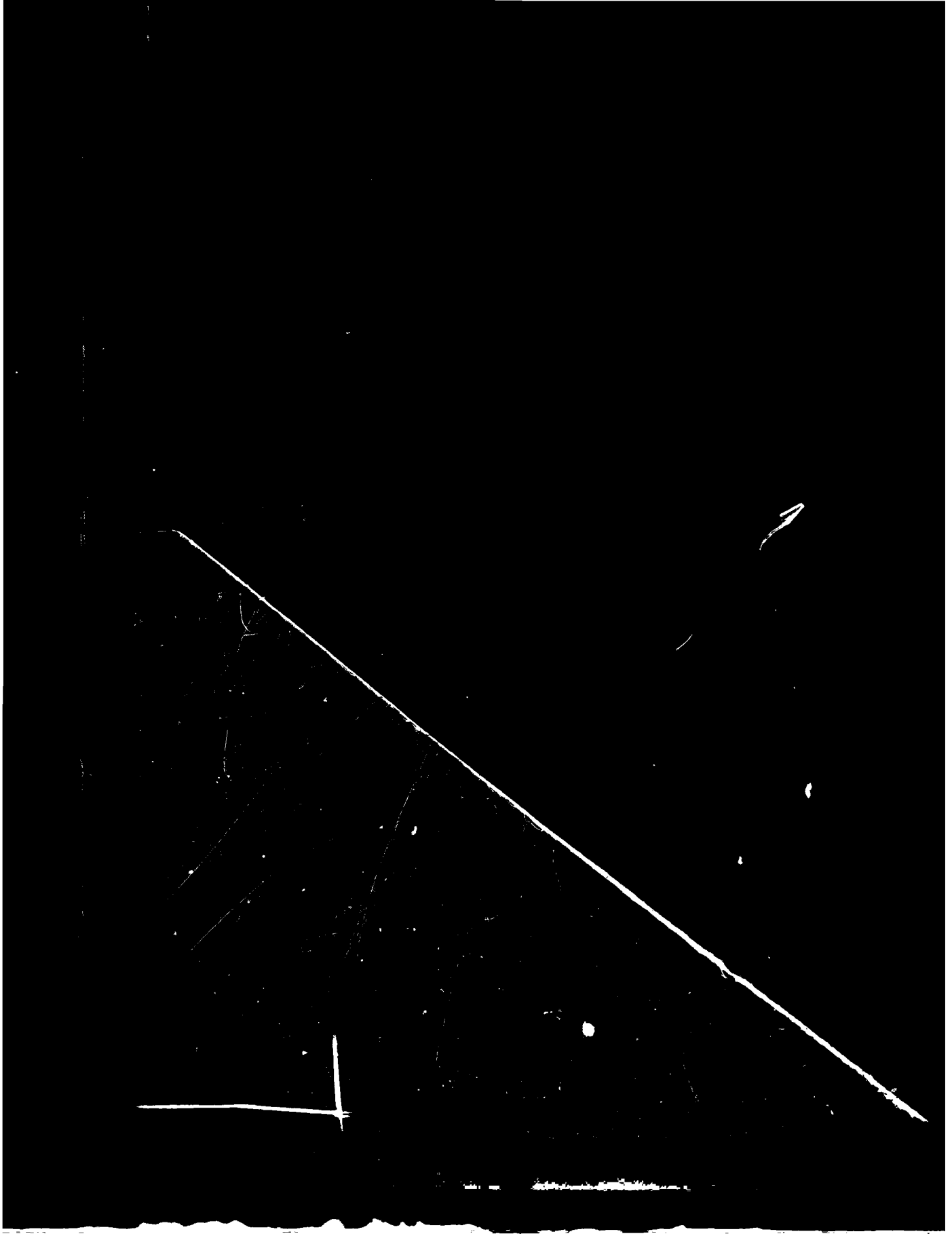
[P3] Prazdny K. "Waveform Segmentation and Description using Edge Preserving Smoothing" Computer Vision, Graphics and Image Processing 23, 327-333 (1982).

[W1] Witkin A. "Scale Space Filtering", Proc. IJCAI, Karlsruhe (1983) 1019-1021.

[Y1] Yuille A. and Poggio T, "Fingerprints Theorems for Zero Crossings" A.I. Memo 730. MIT (1983).

END

FILMED

4-85

DTIC