A Continuous Relaxation Labeling Algorithm for Markov Random Fields

LIONEL PELKOWITZ

Abstract—**A probabilistic relaxation algorithm is described for labeling the vertices of a Markov random field (MRF) defined on a finite graph. The proposed algorithm has two features which make it attractive. First, the multilinear structure of the relaxation operator allows simple, necessary, and sufficient convergence conditions to be derived. The second advantage is local optimality. Given a class of MRF's indexed by a parameter c, such that when $c = 0$ the vertices are independent, it is shown that the estimates of the *a posteriori* probabilities generated by the algorithm differ from the true values by terms that are at least second order in c.**

I. Introduction

The term relaxation labeling (RL) refers to a class of algorithms for assigning a state or label to each vertex in a graph, by iterating a transformation until a fixed point is reached. The transformation must be local in the sense that the output at a given vertex depends only on the input at that vertex and its neighbors. RL algorithms are classified as being either discrete or continuous. In discrete relaxation (DR), we begin by assigning an initial label or set of labels to each vertex in the graph. At each iteration of the relaxation operator, these labels are modified until a stable configuration is reached. In continuous relaxation, each vertex is assigned a weight vector containing one component for each possible label. The weights are constrained to be nonnegative and to sum to unity. The relaxation transformation is iterated until the weights converge. Then each vertex is assigned the state corresponding to the largest component of the weight vector. When the weights are interpreted as probabilities, then CR is also referred to as probabilistic relaxation (PR). In general, the weights may be interpreted as simple measures of confidence or evidence, rather than actual probabilities [5].

One of the main characteristics of the traditional approach to PR [1]–[5], [20] is that the iterated transformation depends only on the *a priori* compatibility of the state of a given vertex with the states of its neighbors and does not depend on the observations. The dependence of the final probabilities on the observed data is achieved by making the initial probabilities data-dependent. One of the problems with this approach is that it leads to transformations whose convergence properties are difficult to analyze. This is because an acceptable transformation cannot have a unique fixed point, for if it did, the solution would be independent of the data. Instead, we require multiple fixed points, and in certain cases, local convergence results can be derived which state that each fixed point is surrounded by some domain of attraction [5]. However, it does not seem possible to prove that for any set of initial probabilities, the algorithm will converge to a fixed point that is dependent on the observations in some meaningful way.

A considerable amount of research has been devoted to algorithms for segmenting images, or more generally, graphs modeled by Markov random fields (MRF) [6], [7]. Most of these algorithms attempt to find the global configuration which has the maximum *a posteriori* probability (MAP) using a deterministic [10]–[12] or stochastic [9] implementation of discrete relaxation. Derin *et al.* [8] present an MRF segmentation algorithm which estimates the *a posteriori* probability density for each pixel. Their technique processes the image in strips, using the two-dimensional extension of a one-dimensional Bayes smoothing algorithm, and is not a relaxation method. An alternative approach is to "collapse" the probability vectors after each iteration, by assigning to each pixel the state with the highest estimated probability and using this state assignment to compute the probability estimates in the next iteration. This approach forms the basis of the ICM algorithm, [21]–[23].

This paper presents a PR algorithm for labeling a MRF defined on an arbitrary finite graph. Using Bayes' rule, a simple linear relationship is derived between the conditional univariate marginal probability density of a given vertex and the conditional multivariate joint density of its neighbors. The relaxation transformation is obtained by replacing the unknown joint conditional density with its maximum entropy estimate, given the univariate, marginal, conditional density of each of the neighbors, this being simply the product of the marginals. The result is a multilinear operator which, in contrast to the traditional approach, requires no normalization to ensure that the probability estimates lie between 0 and 1. Furthermore, the coefficients of this operator depend both on the observations and the local characteristics of the MRF. We are therefore dealing with a random operator, i.e., an operator-valued random element.

The proposed algorithm will be referred to as maximum entropy relaxation (MER), because the maximum entropy estimate of the joint density is used to construct the relaxation operator. It is hoped that this choice of terminology will not result in MER being confused with the unrelated technique known as maximum entropy image restoration [13].

MER has two features that make it attractive. First, the multilinear structure of the relaxation operator allows one to derive simple, necessary, and sufficient conditions for it to be a contraction mapping and hence to converge to a unique fixed point. Since the operator is data-dependent, the fixed point will be a function of the observations. However, it will be completely independent of the initial probabilities. The second advantage is local optimality. If we consider a class of MRF's indexed by a parameter c, such that when $c = 0$ the vertices are independent, then it is shown that the estimates of the *a posteriori* probability generated by the relaxation algorithm differ from the true values by terms that are at least second order in c. Hence the method is locally optimum at $c = 0$.

Section II of this paper summarizes the statistical model and mathematical notation that are used in subsequent sections. In section III the relaxation algorithm is derived and two simple examples are discussed. Section IV deals with the local optimality property. Necessary and sufficient conditions for convergence will be treated in a forthcoming paper [15].

II. Notation and Basic Assumptions

Let $G = (V, E)$ be an N point graph, where $V = \{v(n),\ n = 1, 2, \cdots, N\}$ is the set of vertices and E is the set of edges. We shall assume that there are no edges joining a given point to itself. $B(n)$ will denote the set of all neighbors of $v(n)$, i.e., the set of all vertices $v(m)$ for which there exist an edge joining $v(n)$ and $v(m)$. $M(n)$ will denote the number of vertices in $B(n)$.

Associated with each vertex in G is a state, or label, chosen from a finite, K element set S. Without loss of generality, we can take S to be the set of integers from 1 to K. $x(n)$ will denote the state of $v(n)$ and X will denote the vector in S^N whose nth component is $x(n)$. In addition, we define $X(n)$ to be the $N-1$ dimensional vector, corresponding to X, with the nth component removed, i.e., $X(n) = [x(1), x(2), \cdots, x(n-1), x(n+1), \cdots, x(N)]$ and $Z[n]$ to be the $M(n)$ dimensional vector, whose components are the set of $x(n)$, such that $v(n)$ belongs to $B(n)$, arranged in some specific order. We will use the notation

Manuscript received July 29, 1988; revised January 22, 1989 and October 27, 1989. This work was supported by the Defense Research Establishment in Valcartier, PQ.

The author was with Computing Devices Company, Ottawa, Canada. He is currently with Imago Manufacturing Ltd., 300-1750 Courtwood Crescent, Ottawa, ON, Canada.

IEEE Log Number 8933538.

0018-9472/90/0500-0709$01.00 ©1990 IEEE

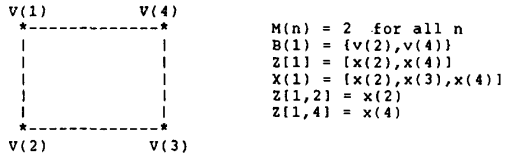

Fig. 1. Illustration of notation for $N = 4$.

$Z[n,m]$ to denote the component of $Z[n]$ corresponding to the vertex $v(m)$ in $B(n)$. (See Fig. 1.) $X|B(n)$ will denote the neighborhood configuration $Z[n]$ obtained by taking the restriction of the global configuration X to $B(n)$.

The statistics of the state assignments are governed by a probability measure P on the global state space S^N, under which X is a Markov random field (MRF) on G, i.e.,

$$P\{x(n)=k|X(n)\}=P\{x(n)=k|Z[n]\}. \tag{1}$$

The conditional probabilities on the right side of (1) are known as the local characteristics of the MRF. We will also assume that for every configuration X

$$P\{X\}>0. \tag{2}$$

MRF's that satisfy (2) are referred to as Gibbs random fields. The state of each vertex is assumed to be not directly observable. Instead, we are given a set of noisy observations of the form

$$y(n)=h(x(n),u(n)). \tag{3}$$

The $u(n)$ represent the observation noise and are arbitrary random elements, which are assumed to be jointly independent and independent of all the $x(n)$. The $y(n)$ take values in some measurable observation space H. Usually, H will either be identical to the state space S or will be some subset of R^M. For example, if S corresponds to K distinct voltage levels that are corrupted by additive white Gaussian noise, then $H=R^1$. We shall use the notation Y and $Y(n)$ to denote the vectors $(y(1),y(2),\cdots,y(N))$ and $(y(1),\cdots,y(n-1),y(n+1),\cdots,y(N))$, respectively.

III. Derivation of the Multilinear Transformation

Our goal is to estimate the conditional density $P\{x(n)=j|Y\}$, using a relaxation algorithm, in which the local processor at each vertex $v(n)$ has available only the observation $y(n)$ and the current estimates of $p\{x(m)=k|Y\}$ for $k=1,2,\cdots,K$ and $v(m)$ contained in $B(n)$. The first step in the derivation of the algorithm is to derive an expression for $P\{x(n)=j|Y\}$ in terms of the conditional neighborhood probabilities. The required expression is given in the following lemma.

Lemma 1:

$$P\{x(n)=j)|Y\}=\sum_{Z[n]}A(n,j;Z[n])P\{Z[n]|Y\} \tag{4}$$

where

$$A(n,j;Z[n])=\frac{P\{y(n)|x(n)=j\}P\{x(n)=j|Z[n]\}}{\sum_k P\{y(n)|x(n)=k\}P\{x(n)=k|Z[n]\}} \tag{5}$$

and $P\{Z[n]|Y\}$ denotes the joint conditional density of the vector $Z[n]$.

The proof of this lemma is a straightforward application of Bayes' rule, the MRF property (1), and the independence assumptions on the observation noise. It is given for reference in the Appendix. Note that (5) implies that each of the coefficients $A(n,j;Z[n])$ lies in the interval $[0,1]$, and that

$$\sum_{j=1}^{K}A(n,j;Z[n])=1. \tag{6}$$

We will sometimes use the notation $A(n,j;X)$, where X is a global configuration of the entire graph G, to denote the value of $A(n,j;Z[n])$ when $Z[n]=X|B(n)$. We can then rewrite (4) as

$$P\{x(n)=j)|Y\}=\sum_X A(n,j;X)P\{X|Y\}. \tag{7}$$

If the local characteristics of the MRF and the conditional probabilities $P\{y(n)|x(n)\}$ are all known, then $A(n,j;Z[n])$ can be easily computed as a function of $y(n)$ for each n, j, and $Z[n]$. Furthermore, it follows from (2) that the denominator of (5) can only be zero if $P\{y(n)|x(n)=k\}$ is equal to zero for all k, or equivalently, if $P\{y(n)\}=0$. Hence each $A(n,j;Z[n])$ is defined, with probability one.

The problem with using (4) to compute the conditional probabilities $P\{x(n)=j)|Y\}$ is that the conditional neighborhood probabilities $P\{Z[n]|Y\}$ on the right side of (4) are unknown. Now, by assumption, the local processor at $v(n)$ has access only to the observation $y(n)$, from which it can compute $A(n,j;Z[n])$, and to the estimates of $P\{x(m)=k|Y\}$, for all k, generated by its neighbors in $B(n)$. A reasonable strategy would therefore be to compute some estimate of the multivariate density $P\{Z[n]|Y\}$, given the univariate densities $P\{x(m)=k|Y\}$, and to substitute this estimate in (4). This substitution would then define a transformation on the set of conditional densities which could be iterated until a fixed point is reached.

The preceding discussion provides the motivation for the second step in the algorithm that consists of replacing $P\{Z[n]|Y\}$ in (4) by the maximum entropy (ME) estimate of $P\{Z[n]|Y\}$ given the marginals $P\{x(m)|Y\}$, for $v(m)$ contained in $B(n)$. However, it is easily shown that the ME estimate is just the product of these marginals (see [17, p. 917]). If we let $p(n,j;t)$ denote the estimate of $P\{x(n)=j|Y\}$ at the tth iteration, then the relaxation transformation obtained using the maximum entropy estimate can be written as

$$\begin{aligned}p(n,j;t+1)&=\sum_{Z[n]}A(n,j;Z[n])P_{ME}\{Z[n];t\}\\&=\sum_X A(n,j;X)P_{ME}\{X;t\}\\&\qquad n=1,2,\cdots,N\\&\qquad j=1,2,\cdots,K\end{aligned} \tag{8}$$

where

$$P_{ME}\{Z[n];t\}=\prod_{v(m)\in B(n)}p(m,Z[n,m];t) \tag{9}$$

and

$$P_{ME}\{X;t\}=\prod_{v(n)\in G}p(n,x(n);t). \tag{10}$$

We will denote this transformation by T and, as noted in the introduction, the overall algorithm will be referred to as maximum entropy relaxation (MER).

Next let $D(K)$ denote the set of K element probability vectors in R^K and let $D(K,N)$ denote the N-fold Cartesian product of $D(K)$ with itself. We will use p to denote a probability vector in $D(K,N)$. It is apparent from (8) that T is a multilinear transformation on $D(K,N)$. Furthermore (6) implies that T maps $D(K,N)$ into itself. Since the maximum entropy estimates defined by (9) and (10) depend only on t implicitly, through the values of $p(n;t)$, we will sometimes write them as $P_{ME}\{Z[n];p\}$ and $P_{ME}\{X;p\}$, respectively.

The relaxation algorithm is implemented by assigning arbitrary values to the initial probabilities $p(n;0)$ and iterating T until the probabilities converge to some fixed point $p(n,j)$. Then $v(n)$ is assigned the label $j'(n)$, which maximizes $p(n,j)$. In [15], necessary and sufficient conditions are given which insure that the algorithm converges to a unique fixed point.

For fixed n and j, the set of coefficients $\{A(n,j;Z[n])\}$, as $Z[n]$ varies over all possible configurations of $B(n)$, specify a tensor—i.e., a scalar valued multilinear function of $KM(n)$ variables—that is computed using (8). Furthermore, each coefficient $A(n,j;Z[n])$ is a function of the single observation $y(n)$. We shall therefore refer to these multilinear functions as the observation tensors and to $A(n,j;Z[n])$ as the observation tensor coefficients. Since the relaxation operator T is defined in terms of these tensors, it too is data-dependent. As pointed out in the introduction, this dependence allows T to have a unique fixed point, for each set of observations Y, that is independent of the initial values assigned to the probabilities. In this respect, MER differs from the traditional approach to PR [1]–[5] in which the operator is fixed and the initial probabilities are assigned values based on the observations. It is interesting to note that in [4], Peleg and Rosenfeld estimate the compatibility coefficients used in the relaxation transformation from the initial probabilities, which are themselves estimated from the observations. This results in a transformation that is effectively data-dependent, but whose structure is very different from the multilinear transformation derived in this paper.

Example 1: Suppose that G is a triangle and that each vertex is a neighbor of the other two. Suppose also that the state space consists of the two-element set $S=\{-1,+1\}$ and that the MRF statistics are specified by the potential function [6], [7]

$$U(X)=b[x(1)x(2)+x(2)x(3)+x(3)x(1)] \tag{11}$$

where b is an arbitrary constant, which may be positive or negative. (Note that to be consistent with the notation adopted in section II, we should have taken S to be $\{1,2\}$ and then replaced each x in (11) by $2x-3$.) The joint density of X is given by

$$P\{X\}=C\exp[-U(X)] \tag{12}$$

where C is a normalizing constant. Next let

$$a=\exp[-b] \tag{13}$$

and let $N+$ denote the number of vertices whose state is $+1$. It then follows from (11)–(13) that the joint density of X is given by

$$\begin{aligned} P\{X\}&=Ca^3 \quad \text{if } N+=0 \text{ or } 3\\ &=C/a \quad \text{if } N+=1 \text{ or } 2 \end{aligned} \tag{14}$$

and that

$$C=\frac{a}{2a^4+6}. \tag{15}$$

Furthermore, if (ijk) is any permutation of (123), then

$$\begin{aligned} P\{x(k)|x(i),x(j)\}&=\frac{a^4}{i+a^4} \quad \text{if } x(i)=x(j)=x(k)\\ &=\frac{1}{1+a^4} \quad \text{if } x(i)=x(j)\neq x(k)\\ &=\frac{1}{2} \quad \text{if } x(i)\neq x(j). \end{aligned} \tag{16}$$

Note the $x(n)$ are positively correlated if $b<0$, and negatively correlated if $b>0$. If $b=0$, then $a=1$ and (16) implies that the $x(n)$ are independent.

Next, consider the observations. We will assume that each $y(n)$ is generated by observing $x(n)$ through a binary symmetric channel with probability of error w, and then define the likelihood ratio

$$\begin{aligned} L(y(n))&=\frac{P\{y(n)|x(n)=+1\}}{P\{y(n)|x(n)=-1\}}\\ &=\frac{1-w}{w} \quad \text{if } y(n)=+1\\ &=\frac{w}{1-w} \quad \text{if } y(n)=-1. \end{aligned} \tag{17}$$

Using (5), (16), and (17), the observation tensors can easily be determined. The resulting relaxation transformation is given by

$$\begin{aligned} p(1,1;t+1)=&\frac{L(y(1))}{a^4+L(y(1))}p(2,-1;t)p(3,-1;t)\\ &+\frac{L(y(1))}{1+L(y(1))}\{p(2,-1;t)p(3,1;t)\\ &+p(2,1;t)p(3,-1;t)\}\\ &+\frac{L(y(1))a^4}{1+L(y(1))a^4}p(2,1;t)p(3,1;t). \end{aligned} \tag{18}$$

Since the MRF and the observation statistics are invariant with respect to a permutation of the vertices, the update equations for $p(2,j;t)$ and $p(3,j;t)$ are obtained from (18) by performing the appropriate permutation. Furthermore, since there are only two states, we have that

$$p(n,1;t)+p(n,-1;t)=1. \tag{19}$$

We can therefore eliminate the variables $p(n,-1;t)$ from the relaxation transformation, and hence obtain a set of three update equations for the three variables $p(n,1;t)$, $n=1,2,3$. Also, using Bayes' rule and (14), the exact value of the conditional density $P\{X|Y\}$ and the marginals $P\{x(n)|Y\}$ can be computed for any set of observations Y and compared with the estimates obtained using the relaxation algorithm. This is most easily done for the case when all the observations are equal. In this case, the symmetry forces all three probability estimates to be equal, provided that the initial estimates $p(n,1;0)$, $n=1,2,3$ are set to the same value. The fixed point of T can then be obtained by solving a quadratic equation in one unknown. Fig. 2 contains plots of both the true marginal conditional density $P\{x(n)=1|y(1)=y(2)=y(3)=1\}$ and the estimate obtained from the relaxation algorithm, for different values of a and w.

The main point to note is that estimated probabilities using MER are tangent to the true probabilities in the neighborhood of $b=0$. As we move away from $b=0$, the estimates may, but do not necessarily, diverge from the exact probabilities. In this example the iteration does not converge for sufficiently large values of b, even though the relaxation equation can be shown to have a unique solution in $[0,1]$. Lack of convergence typically occurs when the evidence provided by the observations contradicts the expectations resulting from the MRF characteristics. Specifically, in the context of the present example, if b is large, then (11) and (12) imply that it is unlikely for all three vertices to be in the same state. However, the assumed equality of the three observed values suggests the opposite conclusion.

The main difficulty in implementing the transformation defined by (8) is that the number of terms in the summation is, in general, equal to the number of distinct neighborhood configurations $Z[n]$, i.e., $K^{M(n)}$; and unless K and $M(n)$ are both small, this number will be extremely large. One solution to this problem is to specify the local characteristics of the MRF so that most $Z[n]$ are "don't care" configurations, for which $P\{x(n)=j|Z[n]\}=P\{x(n)=j\}$ for all j. At a given vertex $v(n)$,

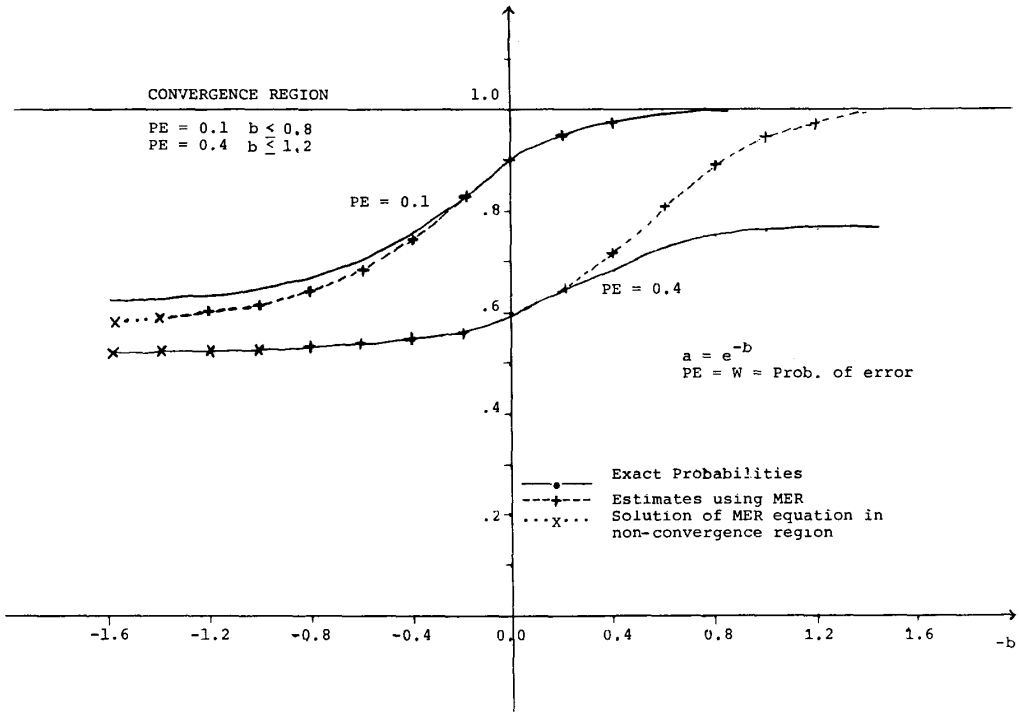

Fig. 2. Exact versus estimated conditional probabilities.

we will let $D\{n\}$ denote the set of "don't care" configurations and $C\{n\}$ will denote its complement. We then have

$$A(n,j;Z[n]) = \frac{P\{y(n)|x(n)=j\}P\{x(n)=j\}}{\sum_k P\{y(n)|x(n)=k\}P\{x(n)=k\}} \triangleq D(n,j) \quad (20)$$

for all $Z[n]$ in $D\{n\}$. The equation for T can then be rewritten as

$$p(n,j;t+1) = \sum_{Z[n]\in C\{n\}} A(n,j;Z[n])P_{ME}\{Z[n];t\} + D(n,j) \sum_{Z[n]\in D\{n\}} P_{ME}\{Z[n];t\}. \quad (21)$$

Rearranging terms, we get

$$p(n,j;t+1) = D(n,j) + \sum_{Z[n]\in C\{n\}} (A(n,j;Z[n]) - D(n,j))P_{ME}\{Z[n];t\}. \quad (22)$$

If the number of configurations in $C\{n\}$ is not too large, then T can be efficiently implemented using (22).

Example 2: Suppose that the relaxation algorithm is to be used for detecting edges in digitized images. In this case the vertices of G can correspond to nonoverlapping three-by-three blocks of pixels in the image. The observations $y(n)$ will correspond to the output of some edge detection operator and may be a vector (e.g., the gradient) or a scalar (e.g., the magnitude of the gradient). In most applications, $B(n)$ consists of either the four or eight nearest neighbors of $v(n)$. The simplest possible state space consists of two elements, i.e., $S = \{\text{nonedge, edge}\}$. Improved performance can be achieved by using a nine-element state space consisting of the nonedge state and eight additional states corresponding to eight possible edge orientations. However, choosing $K = 9$ and $M(n) = 8$ results in $9^8 = 43\,046\,721$ possible configurations for each $Z[n]$. The computation may be simplified, as suggested in the preceding, by specifying the MRF characteristics so that the vast majority of these, which occur very infrequently, are "don't care" configurations. The simplest way of doing this is to include in each $C\{n\}$ only nine configurations of $Z[n]$, corresponding to a straight edge passing through $v(n)$, in one of eight possible directions and one configuration with no edge points. Alternatively, we could define $C\{n\}$ to consists of all configurations of $Z[n]$ containing either no edges or exactly two edges, both pointed at the center pixel, with matching bright and dark sides. It is easily verified that this class has 57 members. The algorithm may be simplified even further by exploiting the symmetry properties of the MRF. This topic will be treated in a forthcoming paper [16].

We conclude this section with the caveat that the transition probabilities in a MRF cannot be assigned arbitrarily [6]. This difficulty was avoided in Example 1 by defining the MRF directly in terms of the Gibbs distribution.

IV. Local Optimality of Maximum Entropy Relaxation

In this section we consider a family of MRF's indexed by a real parameter c. We suppose that the local characteristics $P\{x(n)=j|Z[n];c\}$ are differentiable functions of c such that when $c=0$ the different vertices become independent, i.e.,

$$P\{x(n)=j|Z[n];0\} = P\{x(n)=j;0\} = \text{unconditional prior probability that } x(n)=j, \text{ for } c=0. \quad (23)$$

The goal of this section is to show that the estimates of the conditional probabilities $P\{x(n)=j|Y\}$ generated by MER, differ from the true values of these probabilities only by terms

which are second-order and higher in c. In this sense the MER technique is locally optimum at $c = 0$. We begin by rewriting (4) to indicate the dependence on c:

$$P\{x(n) = j|Y;c\} = \sum_{Z[n]} A(n,j;Z[n];c)P\{Z[n]|Y;c\}. \quad (24)$$

Using (5) and (23), it follows that at $c = 0$

$$A(n,j;Z[n];0) = \frac{P\{y(n)|x(n) = j\}P\{x(n) = j;0\}}{\sum_k P\{y(n)|x(n) = k\}P\{x(n) = k;0\}}$$

$$\triangleq A(n,j;0) \quad \text{(independent of } Z[n]). \quad (25)$$

By setting $c = 0$ in (24) and using the fact that $P\{Z[n]|Y;c\}$ is a PDF that must sum to one, we get that

$$P\{x(n) = j|Y;0\} = A(n,j;0). \quad (26)$$

Note also that at $c = 0$ the vertices are by assumption independent, and in addition, for all values of c the observation noise random elements $u(n)$ are independent of each other and of the $x(n)$. It therefore follows that for $c = 0$, the different vertices are conditionally (i.e., given Y) independent. Hence

$$P\{Z[n]|Y;0\} = \prod_{m \in B(n)} P\{x(m) = Z[n,m];0\}$$

$$= \prod_{m \in B(n)} A(m,Z[n,m];0). \quad (27)$$

We next consider the first-order term. If we differentiate each side of (24) with respect to c at $c = 0$ and use $'$ to denote differentiation with respect to c, we get

$$P'\{x(n) = j|Y;0\} = \sum_{Z[n]} A'(n,j;Z[n];0)P\{Z[n]|Y;0\}$$

$$+ A(n,j;0)\frac{d}{dc}\left\{\sum_{Z[n]} P\{Z[n]|Y;c\}\right\}\Bigg|_{c=0} \quad (28)$$

where (25) was used to factor $A(n,j;0)$ out of the summation over $Z[n]$. Now the second term on the RHS of (28) is identically zero, since the sum over $Z[n]$ is identically one. Combining (27) and (28) then gives

$$P'\{x(n) = j|Y;0\}$$

$$= \sum_{Z[n]} A'(n,j;Z[n];0) \prod_{m \in B(n)} A(m;Z[n,m];0). \quad (29)$$

Next suppose that $P\{x(n) = j|Y;c\}$ is estimated using a relaxation algorithm of the form

$$p(n,j;c;t+1) = \sum_{Z[n]} A(n,j;Z[n];c)\hat{P}\{Z[n];c;t\} \quad (30)$$

where $\hat{P}\{Z[n];c;t\}$ is some estimate of $P\{Z[n]|Y;c\}$ constructed using only the values of $p(m,k;t)$, for m in $B(n)$ and k in S, and whose functional form is independent of all the statistical parameters characterizing the MRF and the observation noise, including c. Then, at $c = 0$ we have

$$p(n,j;0;t+1) = \sum_{Z[n]} A(n,j;0)\hat{P}\{Z[n];0;t\}$$

$$= A(n,j;0) \quad \text{for all } t. \quad (31)$$

Thus, when $c = 0$, the relaxation algorithm converges in a single iteration to the true value of $P\{x(n) = j|Y;c\}$. Next, differentiate each side of (30) with respect to c, at $c = 0$, to get

$$p'(n,j;0;t+1) = \sum_{Z[n]} A'(n,j;Z[n];0)\hat{P}\{Z[n];0;t\}$$

$$= \sum_{Z[n]} A'(n,j;Z[n];0)\hat{P}\{Z[n];0;1\}$$

$$\text{(for } t > 0). \quad (32)$$

Note that, as in (28), the second term in the derivative involving $\hat{P}'\{Z[n];0;t\}$ is identically zero. If $\hat{P}\{Z[n];c;t\}$ is constructed using the maximum entropy estimate, then

$$\hat{P}\{Z[n];0;t\} = \prod_{m \in B(n)} p(m,Z[n,m];0;t)$$

$$= \prod_{m \in B(n)} A(m,Z[n,m];0) \quad \text{for } t > 0. \quad (33)$$

Thus, when $c = 0$ and the maximum entropy estimate of $P\{Z[n]|Y;c\}$ is used, then $p'(n,j;0;t)$ converges in two iterations to the true value of $P'\{x(n) = j|Y;0\}$. Now consider again the general case, where the relaxation process is implemented using any arbitrary estimate $\hat{P}\{Z[n];c;t\}$ constructed from the values of $p(m,k;c;t)$ and which does not depend explicitly on any of the system parameters which specify the MRF local characteristics or the observation noise statistics. It follows from (32) that, using this process, the values of $p'(n,j;0;t)$ will converge in two iterations. Furthermore, to converge to the true value for all possible system parameters, the relaxation process must be implemented using the maximum entropy estimate.

Example 1: Consider the MRF used in example 1 of section III. If the quantity b that appears in the expression for the potential given in (11) is varied, then we obtain a family of MRF's that satisfies the hypotheses of this section, with b playing the role of the parameter c. An examination of Fig. 2 verifies that for both low (0.1) and high (0.4) probability of error, at $b = 0$, the value and the derivative, with respect to b, of $P\{x(n) = 1|Y;b\}$ agree with estimates provided by MER.

V. Conclusion

A probabilistic relaxation algorithm which applies to Markov random fields over any finite graph has been described. The algorithm makes use of the maximum entropy estimate of the joint density of the vertices in each local neighborhood and is therefore termed maximum entropy relaxation (MER). The technique was proven to be locally optimum as the dependence between vertices approaches zero.

Two important topics relating to MER were briefly mentioned, but not developed. The first of these is the study of convergence criteria. As noted in the introduction, the multilinear structure of the relaxation transformation allows one to derive simple, necessary, and sufficient conditions for this transformation to be a contraction mapping. The contraction mapping theorem [14] then insures that the relaxation process converges to a unique fixed point and also provides a bound at each iteration, on the distance of the current state vector from this fixed point. This bound allows the iterations to be truncated prior to convergence. A detailed treatment of the convergence criteria is given in [15].

The second topic, mentioned at the end of section III, relates to simplifying the relaxation algorithm by exploiting the symmetry properties of the observation tensors. This is done in [16] by applying the maximum entropy principle in a less trivial way than it was used in this paper. The resulting relaxation transformation is a multinomial, rather than multilinear, operator, and it can be used with larger neighborhoods than are computationally feasible for the basic method presented in this paper.

We conclude this paper with a suggestion for a generalization of MER that may result in improved performance at the expense of increased computational load. The generalization is obtained by noting that the maximum entropy estimate of a joint density, given the marginals, is a special case of the minimum discrimination information (MDI) method of estimating a joint density suggested by Good [17] and further developed by

Ireland, Ku, and Kullback [18], [19]. In this method the estimate $\hat{P}\{Z[n]\}$ of the unknown conditional density $P\{Z[n]|Y\}$ is obtained by minimizing the discrimination information between $\hat{P}$ and a reference density P_0, subject to the constraints on the marginals. The discrimination information is also referred to as the Kullback–Liebler distance. If P_0 is taken to be the uniform density, then the result is just the maximum entropy estimate of (9), i.e., the product of the marginals. In the proposed generalization of MER, which we shall call minimum discrimination–information relaxation (MDR), P_0 is taken to be the unconditional joint density of $Z[n]$, i.e., $P\{Z[n]\}$, which is assumed to be known.

The problem with this approach is that there is no formula to compute the MDI estimate from the marginals, except for the special case where the components of $Z[n]$ are independent under P_0. It can, however, be computed using the iterative procedure presented in [18] and [19].

For the family of MRF's indexed by a parameter c, discussed in Section IV, it can be shown that the local optimality property proved for MER is also valid for MDR. Moreover, since $P\{Z[n]\}$ depends explicitly on c, it is hoped that using it as the reference density will result in improved performance for large values of c.

Appendix A
Proof of Lemma 1

We begin by using the theorem of total probabilities to write

$$P\{x(n)|Y\} = \sum_{X(n)} P\{x(n), X(n)|Y\} = \sum_{X(n)} P\{x(n)|X(n), Y\} P\{X(n)|Y\}. \quad (34)$$

Using Bayes' rule, we then get

$$P\{x(n)|X(n), Y\} = P\{x(n)|y(n), X(n), Y(n)\} = \frac{P\{y(n)|x(n), X(n), Y(n)\} P\{x(n)|X(n), Y(n)\}}{P\{y(n)|X(n), Y(n)\}}. \quad (35)$$

Since the observation noise $u(n)$ is by assumption independent of $u(m)$ for $m \neq n$ and also independent of all the $x(m)$, it follows from (3) that

$$P\{y(n)|x(n), X(n), Y(n)\} = P\{y(n)|x(n)\}. \quad (36)$$

Next,

$$P\{x(n)|X(n), Y(n)\} = \frac{P\{x(n), X(n), Y(n)\}}{P\{X(n), Y(n)\}} = \frac{P\{Y(n)|x(n), X(n)\} P\{x(n), X(n)\}}{P\{Y(n)|X(n)\} P\{X(n)\}}. \quad (37)$$

But again, by the independence assumptions on the observation noise, it follows from (3) that

$$P\{Y(n)|x(n), X(n)\} = P\{Y(n)|X(n)\}. \quad (38)$$

Combining (37) and (38), we get

$$P\{x(n)|X(n), Y(n)\} = P\{x(n)|X(n)\}. \quad (39)$$

The denominator of (35) can now be written as

$$\begin{aligned} P\{y(n)|X(n), Y(n)\} &= \sum_{x(n)} P\{y(n), x(n)|X(n), Y(n)\} \\ &= \sum_{x(n)} P\{y(n)|x(n), X(n), Y(n)\} \cdot P\{x(n)|X(n), Y(n)\} \\ &= \sum_{x(n)} P\{y(n)|x(n)\} P\{x(n)|X(n)\} \end{aligned} \quad (40)$$

using (36) and (39). Combining (34)–(36), (39), and (40) results in

$$P\{x(n)=j|Y\} = \sum_{X(n)} \frac{P\{y(n)|x(n)=j\} P\{x(n)=j|X(n)\}}{\sum_k P\{y(n)|x(n)=k\} P\{x(n)=k|X(n)\}} \cdot P\{X(n)|Y\}. \quad (41)$$

Note that (41) was derived using only the independence assumptions on the observation noise, without the Markov random field property. If we now include this property, as specified by (1), then (41) becomes

$$P\{x(n)=j|Y\} = \sum_{X(n)} A(n, j; Z[n]) p\{X(n)|Y\} \quad (42)$$

where $A(n, j; Z[n])$ is given by (5). Finally, if we let $U[n]$ denote the components of $X(n)$ that are not part of $Z[n]$, (42) becomes

$$\begin{aligned} P\{x(n)=j|Y\} &= \sum_{Z[n]} \sum_{U[n]} A(n, j; Z[n]) P\{X(n)|Y\} \\ &= \sum_{Z[n]} A(n, j; Z[n]) \left(\sum_{U[n]} P\{Z[n], U[n]|Y\} \right) \\ &= \sum_{Z[n]} A(n, j; Z[n]) P\{Z[n]|Y\}. \end{aligned} \quad (43)$$

Acknowledgment

The author wishes to thank one of the anonymous reviewers for pointing out the work of Kullback on estimating joint densities.

References

[1] A. Rosenfeld, R. A. Hummel, and S. W. Zucker, "Scene labelling by relaxation operations," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-6, pp. 420–453, June 1976.
[2] S. Peleg, "A new probabilistic relaxation scheme," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. PAMI-2, pp. 362–369, July 1980.
[3] S. W. Zucker, E. V. Krishnamurthy, and R. L. Haar, "Relaxation processes for scene labelling: Convergence, speed and stability," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-8, pp. 41–48, Jan. 1978.
[4] S. Peleg and A. Rosenfeld, "Determining compatibility coefficients for curve enhancement relaxation processes," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-8, pp. 548–555, July 1978.
[5] R. A. Hummel and S. W. Zucker, "On the foundations of relaxation labelling processes," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-5, pp. 267–287, May 1983.
[6] J. E. Besag, "Spatial interaction and the statistical analysis of lattice systems," *J. Roy. Statist. Soc. Ser. B 36*, 192–236, 1974.
[7] R. Kinderman and J. L. Snell, *Markov Random Fields and Their Applications*. Providence, RI: Amer. Math. Soc., 1980.
[8] H. Derin, H. Elliot, R. Cristi, and D. Geman, "Bayes smoothing algorithms for segmentation of images modelled by Markov Random Fields," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, pp. 707–720, Nov. 1984.
[9] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, pp. 721–741, Nov. 1983.
[10] R. Cristi and M. Shridhar, "A parallel algorithm for image segmentation based on the Gibbs field model," in *Proc ISCAS 85*, pp. 127–130.
[11] H. Derin and C. S. Won, "A parallel image segmentation algorithm using relaxation with varying neighborhoods and its mapping to array processors," *Comput. Vision Graph. Image Process.*, vol. 40, pp. 54–78, 1987.

[12] F. S. Cohen and D. B. Cooper, "Simple parallel hierarchical and relaxation algorithms for segmenting noncausal Markovian random fields," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, pp. 195–219, Mar. 1987.
[13] H. J. Trussel, "The relationship between image restoration by the maximum *a posteriori* probability method and a maximum entropy method," *IEEE Trans. Acoust. Speech Signal Process.*, vol. ASSP-28, no. 1, pp. 114–117, Feb. 1980.
[14] M. Vidyasagar, *Nonlinear Systems Analysis*. Englewood Cliffs, NJ: Prentice Hall, 1978.
[15] L. Pelkowitz, "Convergence criteria for maximum entropy relaxation labeling," submitted for publication.
[16] ____, "On the use of symmetry in maximum entropy relaxation labeling," submitted for publication.
[17] I. J. Good, "Maximum entropy for hypothesis formulation, especially for multidimensional contingency tables," *Ann. Math. Stat.*, vol. 34, pp. 911–934.
[18] H. Ku and S. Kullback, "Approximating discrete probability distributions," *IEEE Trans. Inform. Theory*, vol. IT-15, no. 4, pp. 444–447, July 1969.
[19] C. T. Ireland and S. Kullback, "Contingency tables with given marginals," *Biometrika*, vol. 55, pp. 179–188, Mar. 1968.
[20] R. M. Haralick, "Decision making in context," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. PAMI-5, no. 4, pp. 417–428, July 1983.
[21] J. Besag, "On the statistical analysis of dirty pictures," *J. Roy. Statist. Soc., Series B*, vol. 48, no. 3, pp. 259–302, 1986.
[22] J. Kittler and J. Foglein, "Contextual classification of multispectral pixel data," *Image Vision Comput.*, vol. 2, no. 1, pp. 13–19, Feb. 1984.
[23] J. Kittler and J. Foglein, "Contextual decision rules for objects in lattice configurations," in *Proc. 7th Int. Conf. Pattern Recognition*, Montreal, PQ, 1984, pp. 270–272.

A Class of New KNN Methods for Low Sample Problems

GUTURU PARTHASARATHY, MEMBER, IEEE, AND B. N. CHATTERJI

Abstract—**The K-nearest neighbor (KNN) estimates proposed by Loftsgaarden and Quesenbery [1] give unbiased and consistent estimates of $p(X)$ when K, the number of nearest neighbors considered, and N, the total number of observations available, tend to infinity such that $K/N \to 0$. Hence excellent results may be obtained in large sample problems by using the KNN method for either density estimation or classification. A class of new KNN estimates is proposed as weighted averages of K KNN estimates, and it is shown that in small sample problems they give closer estimates to the true probability density than the traditional KNN estimates. Further, on the basis of some experimental results, we demonstrate that the KNN rules based on these estimates are suitable for small sample classification problems.**

I. Introduction

Loftsgaarden and Quesenbery [1] proposed a very useful and simple method for nonparametric estimation of the probability density function $p(X)$ of a random variable X from N observations of X. This method is known as the KNN method. The density estimate according to this method is given by

$$\hat{p}_N(X) = \frac{K-1}{N} \cdot \frac{1}{V(K,N,X)} \tag{1}$$

where $V(K,N,X)$ is the smallest hypervolume enclosing all the points at least as near to X as the Kth nearest neighbor of X. On application of this method to the classification problem, we get the KNN rule which classifies an observation with unknown classification by assigning it to the class most heavily represented among its K nearest neighbors. When K and $N \to \infty$ such that $K/N \to 0$, the KNN estimates can be shown to be unbiased and consistent estimates of $p(X)$. Hence, in large sample problems, the KNN estimates or the KNN majority rules for classification give excellent results. But, for small sample problems, a KNN classification rule with the facility to weight the evidence of samples nearer to the unknown observation more heavily is intuitively more appealing. Using this idea, Dudani [2] developed a KNN rule called the distance-weighted KNN rule and tried to establish the merit of his rule on the basis of some experimental results. Later, however, some researchers [3], [4] pointed out that Dudani made an unfair comparison of his rule to a traditional KNN rule in which all ties are reckoned as errors. Furthermore they attempted to establish by means of some experimental results that the distance-weighted KNN rule does not offer any advantage over the traditional KNN majority rules with facility to resolve ties judiciously. Yet, since the sample size to dimensionality ratios for the problems they have chosen are large, their conclusions may not be valid for small sample problems. In this paper, we show that KNN estimates formed as weighted averages of K KNN estimates give, under some assumptions, closer estimates to the true probability density than the usual KNN estimates. Again, by applying these estimates to the classification problem, we arrive at some distance-weighted KNN rules that have more of a theoretical basis than the one proposed by Dudani. The relatively superior performance of these KNN rules (*vis-a-vis*, the traditional ones) in small sample problems is demonstrated by means of some experimental results.

II. Average KNN Method for Density Estimation and Classification

Let us consider the KNN estimate of $p(X)$ given by the (1). By substituting in this equation the values 1, 2, $\cdots$ etc. for K, we get the 1NN, 2NN, etc. estimates. The equation thus formed for the 1NN estimate has a term involving the volume enclosing the first neighbor and hence has the information about the distance of the first neighbor. Similarly, the other estimates have the information about the distances of different neighbors. So, for incorporating this information on a statistical basis, a new estimate of $p(X)$ may be formulated as an average of the different nearest-neighbor estimates. When Euclidean metric is employed for measuring distances, it is given by

$$\begin{aligned}\hat{p}(X) &= \frac{1}{K} \cdot \sum_{i=1}^{K} \frac{i-1}{N} \cdot \frac{1}{V(i,N,X)} \\ &= \frac{1}{K} \cdot \sum_{i=1}^{K} \frac{i-1}{N} \cdot \frac{1}{\{2\pi^{n/2} r_i^n\}/\{n \cdot \Gamma(n/2)\}} \\ &= \frac{C}{K} \cdot \sum_{i=1}^{K} \frac{i-1}{N} \cdot \frac{1}{r_i^n} \quad \text{(say)}\end{aligned} \tag{2}$$

In this equation, r_i is the radial distance of the ith nearest neighbor from the unknown observation, $\Gamma(\cdot)$ is the gamma function, n is the dimensionality of the sample space, and C is a constant given by

$$C = \frac{n \cdot \Gamma(n/2)}{2\pi^{n/2}}. \tag{3}$$

Now let $\hat{p}_{N1}(X)$, $\hat{p}_{N2}(X), \cdots$, and let $\hat{p}_{NK}(X)$ be the 1NN, 2NN, $\cdots$ KNN estimates of $p(X)$ from independent sets of

Manuscript received December 22, 1988; revised July 1, 1989 and November 6, 1989.
The authors are with the Department of Electronics and Electrical Communication Engineering, Indian Institute of Technology, Kharagpur-721 302, India. A preliminary version of this paper was presented at the International Conference on Computers, Systems, and Signal Processing, Bangalore, India, Dec. 9–12, 1984 (sponsored by IEEE Computer Society, IEEE Circuits and Systems Society and IISc).
IEEE Log Number 8933544.

0018-9472/90/0500-0715$01.00 ©1990 IEEE