Correspondence

A New Probabilistic Relaxation Scheme

SHMUEL PELEG

Abstract—**Let a vector of probabilities be associated with every node of a graph. These probabilities define a random variable representing the possible labels of the node. Probabilities at neighboring nodes are used iteratively to update the probabilities at a given node based on statistical relations among node labels. The results are compared with previous work on probabilistic relaxation labeling, and examples are given from the image segmentation domain. References are also given to applications of the new scheme in text processing.**

Index Terms—**Image processing, line enhancement, relaxation, segmentation.**

I. Introduction

The problem of probabilistic graph labeling, or "probabilistic relaxation," is as follows.

Let $G = (V, E)$ be a graph with $V = \{v_1, \cdots, v_n\}$ the set of nodes and $E \subseteq V \times V$ the set of arcs, and let $\Lambda = \{\lambda_1, \cdots, \lambda_L\}$ be a set of labels. With every node $v_i \in V$ a random variable l_i is associated, specifying the label of node v_i. The true distribution of l_i is unknown, but based on some measurements, a probability distribution $P_i^{(0)}$: $\Lambda \to [0, 1]$ is estimated for every random variable l_i. In this paper iterative updating of these estimates is discussed, based on the estimates for the distributions at neighboring nodes, and on statistical relations among the random variables l_i at neighboring nodes. A similar concept is also used in probabilistic relaxation labeling [1], [4], [8]. Even though the probability updating rule and the coefficients used by existing relaxation schemes have no theoretical basis, relaxation has performed remarkably well in many domains. The updating rule developed in this paper is based on elementary probability theory and no guesswork was involved in the choice of rules or coefficients. As shown by examples, the performance of this method for image segmentation examples is at least as good as that of earlier relaxation schemes. The main advantage of this new method is that since the updating rule is analytically derived, all coefficients are defined, eliminating the need to guess them. Also, analytical studies of the properties of the process can hopefully be made to yield more general results than those obtained for relaxation [2], [10], [11]. Indeed, the new scheme has been generalized to allow more general neighborhoods of a node, and was applied successfully to handwriting recognition [6] and breaking substitution ciphers [5].

II. Probability Updating

Given a graph $G = (V, E)$, a set of labels Λ, and an estimated discrete probability distribution P_i: $\Lambda \to [0, 1]$ for each random variable l_i, new estimated probability distributions for the l_i's are to be computed.

Manuscript received December 29, 1978; revised December 5, 1979. This work was supported by the National Science Foundation under Grant MCS-76-23763.

The author is with the Computer Science Center, University of Maryland, College Park, MD 20742.

We can regard the probability vectors P_i as events, i.e., we can think of them as being chosen from a space of possible vectors. We shall now consider various prior and conditional probabilities involving these P_i events and the outcomes of the random variables l_i. In particular, we shall consider probabilities of the following form.

1) Prob $(l_i = \alpha \mid P_i)$; this is just the probability that node v_i has label α, given that the estimate for the probability distribution of labels is P_i. We denote this probability by $P_i(l_i = \alpha)$.

2) Prob $(P_i \mid l_i = \alpha)$; this is the probability that the estimate for the distribution for node v_i is P_i, given that the "true" observed label of v_i was α.

Evidently we have $P_i(l_i = \alpha) = \text{Prob}(P_i \mid l_i = \alpha) \cdot \text{Prob}(l_i = \alpha)/\text{Prob}(P_i)$. (The problem of actually calculating $\text{Prob}(P_i)$ will not be considered yet.)

Let v_i and v_j be two nodes such that $(v_i, v_j) \in E$. Then we can consider the joint events $(l_i = \alpha, l_j = \beta)$ and (P_i, P_j), and write

$$\text{Prob}(l_i = \alpha, l_j = \beta \mid P_i, P_j) \equiv P_{ij}(l_i = \alpha, l_j = \beta) = \frac{\text{Prob}(P_i, P_j \mid l_i = \alpha, l_j = \beta) \cdot \text{Prob}(l_i = \alpha, l_j = \beta)}{\text{Prob}(P_i, P_j)}. \quad (1)$$

We now assume that

$$\text{Prob}(P_i, P_j \mid l_i = \alpha, l_j = \beta) = \text{Prob}(P_i \mid l_i = \alpha)\, \text{Prob}(P_j \mid l_j = \beta).$$

This assumption has two parts. The first part is conditional independence:

$$\text{Prob}(P_i, P_j \mid l_i = \alpha, l_j = \beta) = \text{Prob}(P_i \mid l_i = \alpha, l_j = \beta) \cdot \text{Prob}(P_j \mid l_i = \alpha, l_j = \beta).$$

The second part is the direct dependence of P_i on l_i once l_i is determined:

$$\text{Prob}(P_i \mid l_i = \alpha, l_j = \beta) = \text{Prob}(P_i \mid l_i = \alpha).$$

Using the above assumption we get

$$\text{Prob}(P_i, P_j \mid l_i = \alpha, l_j = \beta) = \frac{\text{Prob}(l_i = \alpha \mid P_i)\, \text{Prob}(l_j = \beta \mid P_j)\, \text{Prob}(P_i)\, \text{Prob}(P_j)}{\text{Prob}(l_i = \alpha)\, \text{Prob}(l_j = \beta)} \equiv \frac{P_i(l_i = \alpha)\, P_j(l_j = \beta)\, \text{Prob}(P_i)\, \text{Prob}(P_j)}{\text{Prob}(l_i = \alpha)\, \text{Prob}(l_j = \beta)}.$$

The meaning of this assumption is that the probability estimate P_i is directly dependent only on l_i, and once l_i is given, the probability of the estimate being P_i is independent of P_j, $j \neq i$, and of l_j, $j \neq i$. While the conditional independence assumption is not always true, we had to make it since no results could be obtained otherwise. In the cases where this assumption is not true, we use it as the best approximation we have. Under this assumption, (1) becomes

$$P_{ij}(l_i = \alpha, l_j = \beta) = P_i(l_i = \alpha)\, P_j(l_j = \beta) \cdot \frac{\text{Prob}(l_i = \alpha, l_j = \beta)}{\text{Prob}(l_i = \alpha)\, \text{Prob}(l_j = \beta)} \cdot \frac{\text{Prob}(P_i)\, \text{Prob}(P_j)}{\text{Prob}(P_i, P_j)}. \quad (2)$$

0162-8828/80/0700-0362$00.75

From now on, we will denote

$$\frac{\text{Prob}(l_i = \alpha, l_j = \beta)}{\text{Prob}(l_i = \alpha)\,\text{Prob}(l_j = \beta)}$$

by $r_{ij}(\alpha, \beta)$. The quantities $r_{ij}(\alpha, \beta)$ are independent of the estimated distributions P_i. Thus, when we use (3a) in the sequel to reestimate the P_i's, the $r_{ij}(\alpha, \beta)$ remain constant during the entire process.

Now

$$\sum_{\beta \in \Lambda} P_{ij}(l_i = \alpha, l_j = \beta) = P_{ij}(l_i = \alpha)$$

and

$$\sum_{\lambda \in \Lambda} \sum_{\beta \in \Lambda} P_{ij}(l_i = \lambda, l_j = \beta) = 1.$$

We thus have

$$P_{ij}(l_i = \alpha) = \frac{\sum_{\beta \in \Lambda} P_{ij}(l_i = \alpha, l_j = \beta)}{\sum_{\lambda \in \Lambda} \sum_{\beta \in \Lambda} P_{ij}(l_i = \lambda, l_j = \beta)} = \frac{\sum_{\beta \in \Lambda} P_i(l_i = \alpha)\, P_j(l_j = \beta)\, r_{ij}(\alpha, \beta)}{\sum_{\lambda \in \Lambda} \sum_{\beta \in \Lambda} P_i(l_i = \lambda)\, P_j(l_j = \beta)\, r_{ij}(\lambda, \beta)} \tag{3}$$

since the factor $[\text{Prob}(P_i)\,\text{Prob}(P_j)]/\text{Prob}(P_i, P_j)$ cancels out.

Note that $P_{ij}(l_i = \alpha)$ still depends on v_j, since it is obtained by summing over the labels of node v_j and does not use other nodes. We can use (3) to compute a new distribution estimate P_i based on the previous distribution estimates P_i and P_j, by letting $P_i^{(n)}$ be our estimate of the distribution of l_i at the nth iteration, writing

$$P_{ij}^{(n+1)}(l_i = \alpha) = \frac{\sum_{\beta \in \Lambda} P_i^{(n)}(l_i = \alpha)\, P_j^{(n)}(l_j = \beta) \cdot r_{ij}(\alpha, \beta)}{\sum_{\lambda \in \Lambda} \sum_{\beta \in \Lambda} P_i^{(n)}(l_i = \lambda)\, P_j^{(n)}(l_j = \beta) \cdot r_{ij}(\lambda, \beta)} \tag{3a}$$

where $P_i^{(0)}$ is our initial estimate for the probability distribution of l_i.

By a method similar to that used to obtain (3a), we can derive an expression for the case where more than two nodes interact. Let $v_1, \cdots, v_n$ be nodes in V with their labels represented by random variables $l_1, \cdots, l_n$ having distribution estimates $P_1^{(k)}, \cdots, P_n^{(k)}$. An expression analogous to (3a) is

$$P_1^{(k+1)}(l_1 = \delta_1) = \frac{\sum_{\delta_2 \in \Lambda} \cdots \sum_{\delta_n \in \Lambda} \prod_{i=1}^{n} P_i^{(k)}(l_i = \delta_i) \cdot r_{1,\cdots,n}(\delta_1, \cdots, \delta_n)}{\sum_{\delta_1 \in \Lambda} \sum_{\delta_2 \in \Lambda} \cdots \sum_{\delta_n \in \Lambda} \prod_{i=1}^{n} P_i^{(k)}(l_i = \delta_i) \cdot r_{1,\cdots,n}(\delta_1, \cdots, \delta_n)} \tag{4}$$

where

$$r_{1,\cdots,n}(\lambda_1, \cdots, \lambda_n) = \frac{\text{Prob}(l_1 = \lambda_1, \cdots, l_n = \lambda_n)}{\prod_{i=1}^{n} \text{Prob}(l_i = \lambda_i)}.$$

When many nodes are considered, each having many possible labels, it is impractical to compute and store the above coefficients. For n nodes and L possible labels at each node we have L^n coefficients (r's). In such cases we leave the Bayesian domain, and use approximate methods to estimate a new distribution from all pairwise estimates. The simplest way to make such an estimate is to take an arithmetic average over all neighbors. Using the arithmetic average, the estimate $P_i^{(k+1)}(l_i = \alpha)$ is

$$P_i^{(k+1)}(l_i = \alpha) = \sum_{(i,j) \in E} C_j P_{ij}^{(k+1)}(l_i = \alpha) \tag{5}$$

where $P_{ij}^{(k+1)}(l_i = \alpha)$ is the estimate for the distribution of l_i based on $P_i^{(k)}$ and $P_j^{(k)}$ obtained using (3a), and the C_j's are weights such that $\Sigma_j C_j = 1$. In the examples given later, we use a simple average, with equal weights given to all the neighbors.

It is evident that computing the next estimates, $P_i^{(k+1)}(\lambda)$, by averaging the pairwise estimates $P_{ij}^{(k+1)}(\lambda)$ is not, in general, the optimal method. A reasonable function that combines all the pairwise estimates into one estimate should have the following properties.

1) $P_i^{(k+1)}(l_i = \lambda) = 0$ if there exists a neighbor v_j such that $P_{ij}^{(k+1)}(l_i = \lambda) = 0$. In this case no possible choice of a label at v_j will permit v_i to be labeled λ.

2) $P_i^{(k+1)}(l_i = \lambda) = 1$ if there exists a neighbor v_j such that $P_{ij}^{(k+1)}(l_i = \lambda) = 1$. This follows from 1) since the labeling at v_j does not allow any label different from λ at v_i.

A function that has the properties 1)–2) was developed by R. Kirby in [3]. This function is based on a sequential updating of the labeling at a node. The effect of one neighbor is computed, and with the new suggested estimates the effect of a second neighbor is computed. This process, as will be shown, does not depend on the order of the neighbors, and can be computed in one step. Define $Q_{ij}^{(n)}(\alpha)$ to be

$$Q_{ij}^{(n)}(\alpha) = \sum_{\beta \in \Lambda} P_j^{(n)}(\beta) \cdot r_{ij}(\alpha, \beta).$$

$Q_{ij}^{(n)}(\alpha)$ can be regarded as the strength of support that label α at v_i gets from its neighbor v_j. Using the above definition of $Q_{ij}^{(n)}(\alpha)$, expression (3a) becomes

$$P_{ij}^{(n+1)}(l_i = \alpha) = \frac{P_i^{(n)}(l_i = \alpha) \cdot Q_{ij}^{(n)}(\alpha)}{\sum_{\lambda} P_i^{(n)}(l_i = \lambda)\, Q_{ij}^{(n)}(\lambda)}. \tag{6}$$

Using $P_{ij}^{(n+1)}(l_i = \alpha)$ of (6) as a new estimate for $P_i(l_i = \alpha)$ (instead of $P_i^{(n)}(l_i = \alpha)$) and computing the effect of another neighbor v_k, we get

$$P_{ijk}^{(n+1)}(l_i = \alpha) = \frac{\dfrac{P_i^{(n)}(l_i = \alpha) \cdot Q_{ij}^{(n)}(\alpha)}{\sum_{\beta} P_i^{(n)}(l_i = \beta) \cdot Q_{ij}^{(n)}(\beta)} \cdot Q_{ik}^{(n)}(\alpha)}{\sum_{\lambda} \dfrac{P_i^{(n)}(l_i = \lambda) \cdot Q_{ij}^{(n)}(\lambda)}{\sum_{\beta} P_i^{(n)}(l_i = \beta) \cdot Q_{ij}(\beta)} \cdot Q_{ik}^{(n)}(\lambda)}$$

which reduces to

$$P_{ijk}^{(n+1)}(l_i = \alpha) = \frac{P_i^{(n)}(l_i = \alpha) \cdot Q_{ij}^{(n)}(\alpha) \cdot Q_{ik}^{(n)}(\alpha)}{\sum_{\lambda} P_i^{(n)}(l_i = \lambda)\, Q_{ij}^{(n)}(\lambda)\, Q_{ik}^{(n)}(\lambda)}$$

which is symmetric in j and k. For all neighbors of i, we have

$$P_i^{(n+1)}(l_i = \alpha) = \frac{P_i^{(n)}(l_i = \alpha) \prod_{(i,j) \in E} Q_{ij}^{(n)}(\alpha)}{\sum_{\lambda} P_i^{(n)}(l_i = \lambda) \prod_{(i,j) \in E} Q_{ij}^{(n)}(\lambda)}. \tag{7}$$

As was mentioned in [12], updating rules that have a multiplicative form like (7) converge faster than those having an additive form. This is also shown in examples later in this paper.

The updating rules are used in an iterative scheme as follows: every node v_i computes in parallel a new estimate for the distribution of l_i based on the old estimates of the distributions of l_i and its neighbors. Then, the old distribution estimates are replaced by the new ones, and another iteration begins. The iterations stop when the change in the estimates becomes very small.

III. The Coefficients

In the updating rules, we used the coefficients

$$r_{ij}(\alpha, \beta) = \frac{\text{Prob}(l_i = \alpha, l_j = \beta)}{\text{Prob}(l_i = \alpha) \cdot \text{Prob}(l_j = \beta)}.$$

These coefficients are independent of the estimated distributions $P_i^{(n)}$, can be computed ahead of time, and remain constant during the entire process. When we have a statistical model for the problem domain, the individual and joint probabilities can be computed using that model. Sometimes these probabilities can be computed by looking at many occurrences, and substituting the relative frequency for the probability. Consider, for example, graphs representing words, where the labels at the nodes are letters. By taking a large enough sample of English text we can find accurate estimates for the individual and joint probabilities of the letters.

In many cases we have neither a statistical model nor a big set that we can sample. In such cases we might try to use the initial distributions $P_i^{(0)}$ themselves as guesses at the real probabilities. This idea was used in [4], where relaxation coefficients were computed from the initial probabilities.

Let $S_1, \cdots, S_k$ be a division of the set of nodes V into subsets such that $\forall\, i \neq j S_i \cap S_j = \phi$ and $\cup_{i=1}^{k} S_i = V$, and also for every S_i the *a priori* distributions are identical for all $l \in S_i$. Then an average over the initial probability estimates $P_i^{(0)}$ can be an estimate for the individual probabilities. Thus, for the set S_k, we can use

$$\overset{*}{P}_k(l = \alpha) = \frac{1}{\|S_k\|} \sum_{v_i \in S_k} P_i^{(0)}(l_i = \alpha) \tag{8}$$

as an estimate of the *a priori* distribution for nodes in S_k.

To find the joint probabilities, we divide the set of arcs E into relations $A_1, \cdots, A_n$ such that $\forall_{i \neq j} A_i \cap A_j = \phi$ and $\cup_{i=1}^{n} A_i = E$, and also for every A_k the probabilities $\text{Prob}(l_i = \alpha, l_j = \beta)$ are identical for every $(v_i, v_j) \in A_k$. An estimate of the joint probability over the relation A_k is

$$\overset{*}{P}_k(l_i = \alpha, l_j = \beta) = \frac{1}{\|A_k\|} \sum_{(v_i, v_j) \in A_k} P_i^{(0)}(l_i = \alpha) \cdot P_j^{(0)}(l_j = \beta). \tag{9}$$

From (8) and (9) we can now compute the coefficients $r_{ij}(\alpha, \beta)$ for $v_i \in S_k$, $v_j \in S_l$, and $(v_i, v_j) \in A_m$:

$$r_{ij}(\alpha, \beta) = \frac{\overset{*}{P}_m(l_i = \alpha, l_j = \beta)}{\overset{*}{P}_k(l_i = \alpha) \cdot \overset{*}{P}_l(l_j = \beta)}. \tag{10}$$

In the sequel we demonstrate by an example the use of coefficients derived using (10).

IV. Fixed Points

The updating operator makes use of the estimates of the label distributions at neighboring nodes in computing the new label distribution estimates at a node. As the process iterates, further nodes begin to indirectly affect a node through their effects on its neighbors. It is expected that the iteration process will converge, since the further two nodes are from each other, the less they should affect each other. Examples shown later demonstrate that the changes from iteration to iteration are greatest at the first iteration, while after several iterations the changes are almost negligible. When there are N nodes, and L possible labels at each node, the updating operator is a continuous function from $[0, 1]^{N \times L}$ into itself. As was pointed out in [2] about the relaxation operator, it is well known that a continuous function from a compact, convex set into itself has at least one fixed point. The updating operator satisfies these conditions, and so it has a fixed point. Thus, if the process does converge, the limit is a fixed point of the updating operator.

The fixed points of the updating operators can be obtained as the solutions to a set of equations that may be derived from (5). Note in particular that the set of all nonambiguous labelings, where every node has a determined label, is a set of fixed points.

Lemma: If $P_i^{(n)}(l_i = \alpha) = 1$ and $P_i^{(n)}(l_i = \beta) = 0$ for all labels $\beta \neq \alpha$, and for every neighbor v_j of v_i there exists a label λ such that $P_j^{(n)}(l_j = \lambda) \neq 0$ and $r_{ij}(\alpha, \lambda) \neq 0$, then the distribution estimate $P_i^{(n+1)}$ will be identical to $P_i^{(n)}$.

Proof: From (3a) it follows that for every neighbor v_j, $P_{ij}^{(n+1)}$ will be identical to $P_i^{(n)}$, and since in (5) $P_i^{(n+1)}$ is the average of the $P_{ij}^{(n+1)}$, it also will be identical to $P_i^{(n)}$.

From the lemma it is clear that when the entire graph is labeled with probabilities zero and one only, and the conditions of the lemma hold for every pair of neighbors, the labeling does not change. When the probabilities are zero and one only, but the other conditions of the lemma are not satisfied, the updating operator is undefined, meaning that a label has probability one, but its joint probability with all possible neighbor labels is zero—a contradiction. (See [10] on the fixed points in relaxation processes.) These fixed points of the updating rule, which correspond to unambiguous labelings, are desired in most cases. The updating of probabilities based on neighboring probabilities is done in the hope that it will reduce ambiguity.

Another type of fixed point is the "no information" labeling, where at *all* nodes the label probabilities are identical to the *a priori* probabilities. Fortunately, in this case, a node must not change its probabilities only when *all* its neighbors have the "no information" labeling. Compared with the unambiguous labeling, where every node is stable when it has such a labeling, regardless of its neighbors, the "no information" labeling is very unstable, and is unlikely to be reached by the iteration process.

V. An Example: Pixel Classification

This section demonstrates the use of iterative probability updating by applying it to the problem of pixel classification.

Usually, classification cannot be done exactly, so every pixel is assigned a probability vector, representing its probability of being a member at each class.

In graph notation, every pixel is a node of the graph, the arcs connect every pixel to its immediate neighbors (4 or 8), and the node labels represent the possible classes. $P_i^{(0)}$ specifies the initial probability estimates assigned to the labels (classes) at node (pixel) v_i, based only on local measurements, such as might be given by a low-level classifier. These probability estimates are then updated. Let $\{v_1, \cdots, v_N\}$ be the N pixels in the picture, and let Λ be the set of possible labels for each pixel. We assume *a priori* uniformity over the picture, so that the *a priori* probabilities $\text{Prob}(l = \alpha)$ are the same for all pixels. The *a priori* joint probabilities $\text{Prob}(l_i = \alpha,\ l_j = \beta)$ also do not depend on the specific nodes, but only on the relation between them (e.g., right neighbor, etc.). As relations between pixels we consider neighborhood relations. Let

3	2	1
5	x	4
8	7	6

be the neighborhood of pixel x. The relation $R_k, k = 1, \cdots, 8$, includes all pairs (v_i, v_j) of pixels such that v_j is the same neighbor of v_i as k is of x.

Given these uniformity assumptions, it is simple to compute the coefficients r_{ij} for the process from the initial probability assignment, if we cannot determine them by any other method. Using (8), the *a priori* probabilities $\text{Prob}(l = \alpha)$ are estimated by

$$\overset{*}{P}(\alpha) = \frac{1}{N} \sum_{i=1}^{N} P_i^{(0)}(l_i = \alpha). \tag{11}$$

The estimates for the joint probabilities, according to (9), are

$$\overset{*}{P}_k(\alpha, \beta) = \frac{1}{\|R_k\|} \sum_{(a_i, a_j) \in R_k} P_i^{(0)}(l_i = \alpha)\, P_j^{(0)}(l_j = \beta), \quad k = 1, \ldots, 8 \tag{12}$$

where the sum is taken over all pairs of nodes having the same neighborhood relation R_k. The coefficients are then

$$\overset{*}{r}_k(\alpha, \beta) = \frac{\overset{*}{P}_k(\alpha, \beta)}{\overset{*}{P}(\alpha)\, \overset{*}{P}(\beta)}. \tag{13}$$

After finding the coefficients either by (13) or by some other method, the updating process can take place. For all pixels v_i, v_j such that $(v_i, v_j) \in R_k$ we compute

$$S_{i,k}^{(n+1)}(\alpha) = \sum_{\beta \in \Lambda} P_i^{(n)}(l_i = \alpha) \cdot P_j^{(n)}(l_j = \beta) \cdot r_k(\alpha, \beta)$$

and

$$q_{i,k}^{(n+1)}(\alpha) = \frac{S_{i,k}^{(n+1)}(\alpha)}{\sum_{\beta \in \Lambda} S_{i,k}^{(n+1)}(\beta)}, \qquad \alpha \in \Lambda.$$

Here $q_{i,k}^{(n+1)}$ is the new probability estimate for the labels at pixel v_i based upon the previous estimates at v_i and v_j, $(v_i, v_j) \in R_k$. Since every pixel has eight neighbors, using (5), the new $P_i^{(n+1)}$ will be the average of these eight estimates:

$$P_i^{(n+1)}(l_i = \alpha) = \frac{1}{8} \sum_{k=1}^{8} q_{i,k}^{(n+1)}(\alpha), \qquad \alpha \in \Lambda. \tag{14}$$

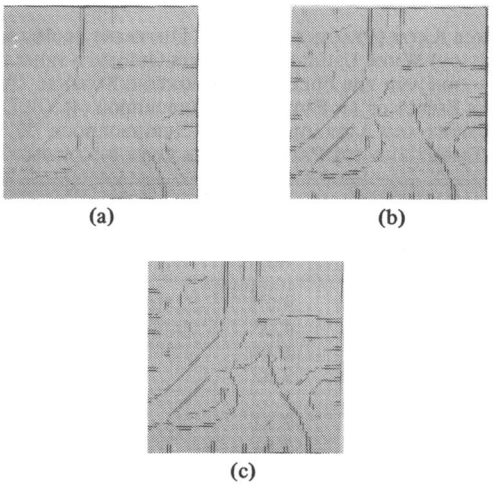


Fig. 1. The line enhancement process, shown by the initial labeling (a), and the 6th and 12th interations (b)-(c).

As a first example of pixel classification we use line detection and enhancement. Each pixel can be labeled as being on a line in one of eight directions, or not being on a line at all. The initial probabilities are estimated using eight line detectors in the eight directions (see [9] for the details).

The coefficients were obtained from the initial probability estimates using (13), and the iteration results are displayed in Fig. 1, where the initial labeling and the results after 6 and 12 iterations are shown. In these pictures, for each pixel the label representing the highest probability line direction is chosen, and a 3-point segment in that direction is displayed, with its darkness corresponding to its probability. The results in Fig. 1 are subjectively at least as good as those obtained previously using relaxation in [4] and [9], and the reduction in ambiguity is apparent.

The second example involves pixel classification on the basis of color. A picture of a house is to be classified into five regions: brick, sky, grass, shadow, and brush. The initial probabilistic classification is done by clustering in color space [1], [2]. The results of a hand classification, which we regard as the "true" classification, are also available for reference. The coefficients $r_{ij}(\alpha, \beta)$ were computed once from that "true" classification using (13). Another set of coefficients was computed in the same way but from the initial probabilistic classification. Table I shows the percent of pixels with classifications different from those in the hand segmentation, when coefficients derived from the hand segmentation were used. The three methods shown are (A) using a complete 4-neighbor updating (4), with no need to combine partial results; (B) averaging the pairwise effect over an 8-neighborhood using (5); and (C) using the product combination (7) of the pairwise effect over an 8-neighborhood. Table II shows the difference rate when coefficients derived from the initial probabilistic labeling are used. The methods shown are (A)-(C) as in Table I; and (D) using the old probabilistic relaxation rule with mutual information coefficients as described in [4]. The new probability updating schemes arrive at the minimal difference much faster than the old relaxation scheme, although the latter eventually reaches a slightly lower difference rate. Fig. 2 shows the

TABLE I

DIFFERENCE RATES (PERCENT OF POINTS DIFFERENT FROM THE HAND CLASSIFICATION) WHEN USING COEFFICIENTS OBTAINED FROM THE HAND SEGMENTATION AND THE FOLLOWING UPDATING SCHEMES. (A) USING 5-TUPLE EFFECT OF AN ENTIRE 4-NEIGHBORHOOD (4). (B) USING PAIRWISE EFFECT AND AVERAGING OVER 8-NEIGHBORHOOD (5). (C) USING PAIRWISE EFFECT AND THE PRODUCT RULE OVER 8-NEIGHBORHOOD (7).

	Error rates (percent)		
Iteration	(A)	(B)	(C)
0	4.60	4.60	4.60
1	4.05	4.05	3.85
2	3.84	4.00	3.87
3	3.82	3.82	3.93
4	3.82	3.82	3.99
5	3.81	3.82	4.02
6	3.81	3.85	4.04
7	3.82	3.84	4.10
8	3.82	3.82	4.11
9	3.79	3.85	4.13
10	3.81	3.89	4.15
11	3.82	3.89	4.16
12	3.80	3.92	4.19

TABLE II

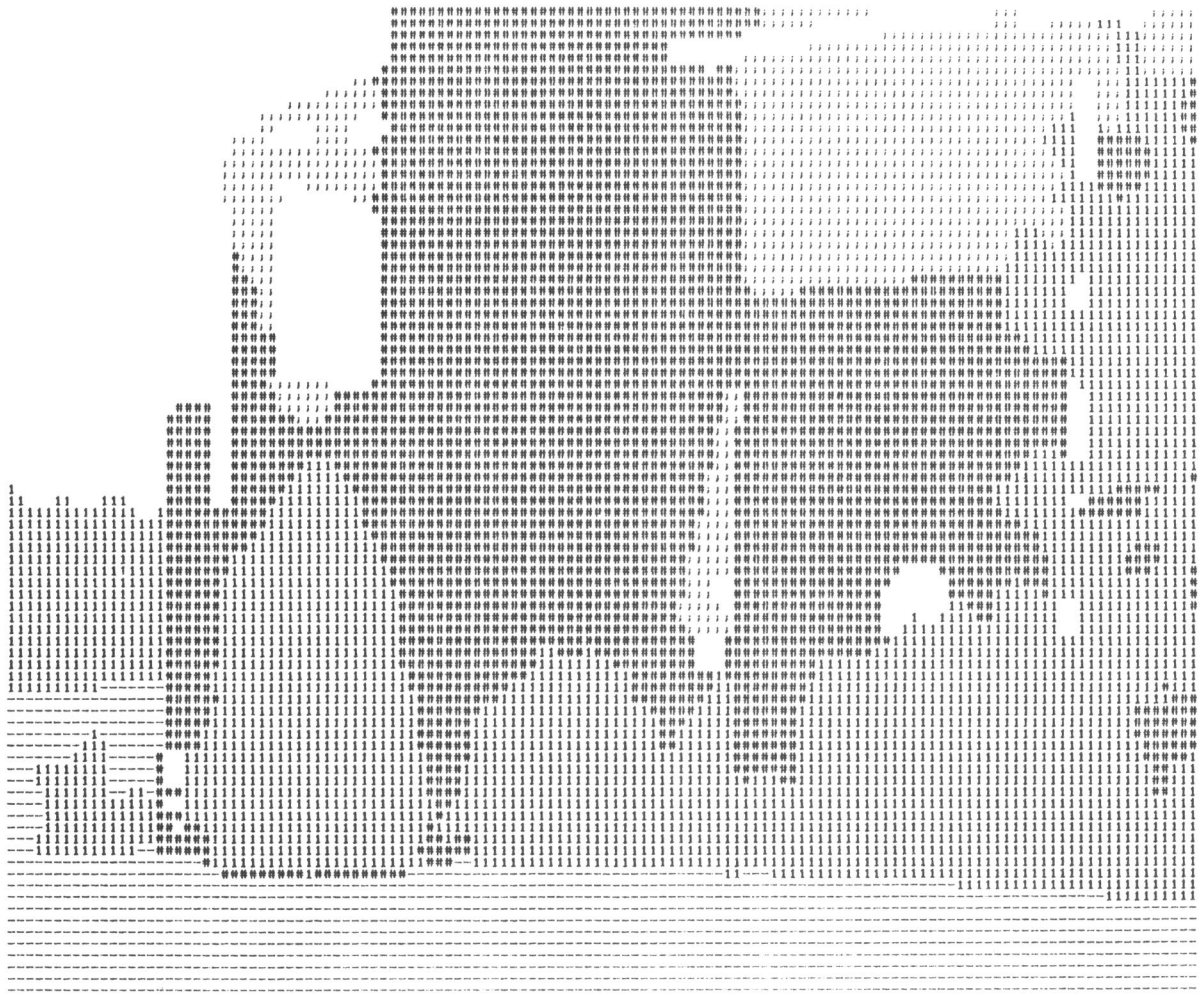
DIFFERENCE RATES WHEN USING COEFFICIENTS OBTAINED FROM THE INITIAL PROBABILISTIC SEGMENTATION AND THE FOLLOWING UPDATING SCHEMES. (A)–(C) SAME AS TABLE I. (D) CLASSICAL RELAXATION USING MUTUAL INFORMATION COEFFICIENTS.

	Error Rates (percent)			
Iteration	(A)	(B)	(C)	(D)
0	4.60	4.60	4.60	4.60
1	4.07	4.04	3.88	4.37
2	3.92	3.96	3.93	4.28
3	3.86	3.80	3.98	4.15
4	3.85	3.81	4.04	4.08
5	3.82	3.79	4.11	4.02
6	3.79	3.77	4.17	3.98
7	3.78	3.83	4.20	3.96
8	3.75	3.89	4.26	3.91
9	3.74	3.90	4.31	3.88
10	3.73	3.90	4.37	3.83
11	3.72	3.88	4.37	3.82
12	3.74	3.88	4.42	3.79[a]

[a]This method reaches an error rate of 3.63 on the 19th iteration.

(a)

Fig. 2. Segmentation of the house picture. Key–Brick: "#"; sky: (blank); grass: "–"; shadow: ";"; brush: "1". (a) Hand segmentation. (b) Initial probabilistic classification (4.6 percent difference). (c) After six iterations (3.77) percent difference).



(b)

Fig. 2. (*Continued*).

hand classification, the initial probabilistic classification (the maximum-probability class for each pixel is displayed), and iteration 6 of the probability updating using the average scheme and coefficients obtained from the initial probabilistic classification.

VI. Comparison with Previous Relaxation Scheme

The basic advantage of the new relaxation scheme is that it has been analytically derived. In addition to being more appealing theoretically, it is also practically improved by having more things well defined. The former relaxation scheme [8] had not defined the coefficients to be used. These coefficients sometimes required fine-tuning in order for the relaxation to produce good results, a process which was usually very lengthy. In the new relaxation, no tuning is needed.

Other theoretical problems with the former relaxation scheme which are not present in the new one are described in [7]. One problem discussed by Pavlidis in [13] involves independently labeled nodes. Another problem involves "no information" labelings. Section VI-A will briefly explain the theoretical faults in the former (nonlinear) relaxation. Section VI-B will do the same for the new relaxation scheme, this time proving that the above faults are not repeated.

A. Nonlinear Relaxation

As described in [8], the updating factor for the estimate $P_i^{(k)}(\lambda)$ (at the kth iteration) is

$$q_i^{(k)}(\lambda) = \frac{1}{n} \sum_j \sum_\alpha r_{ij}(\lambda, \alpha)\, P_j^{(k)}(\alpha) \tag{15}$$

where n is the number of objects, and the r_{ij} are compatibility coefficients in the range $[-1, 1]$. The new estimate of the probability of λ at a_i is

$$P_i^{(k+1)}(\lambda) = \frac{P_i(\lambda)\,[1 + q_i^{(k)}(\lambda)]}{\sum_\alpha P_i^{(k)}(\alpha)\,[1 + q_i^{(k)}(\alpha)]} \,. \tag{16}$$

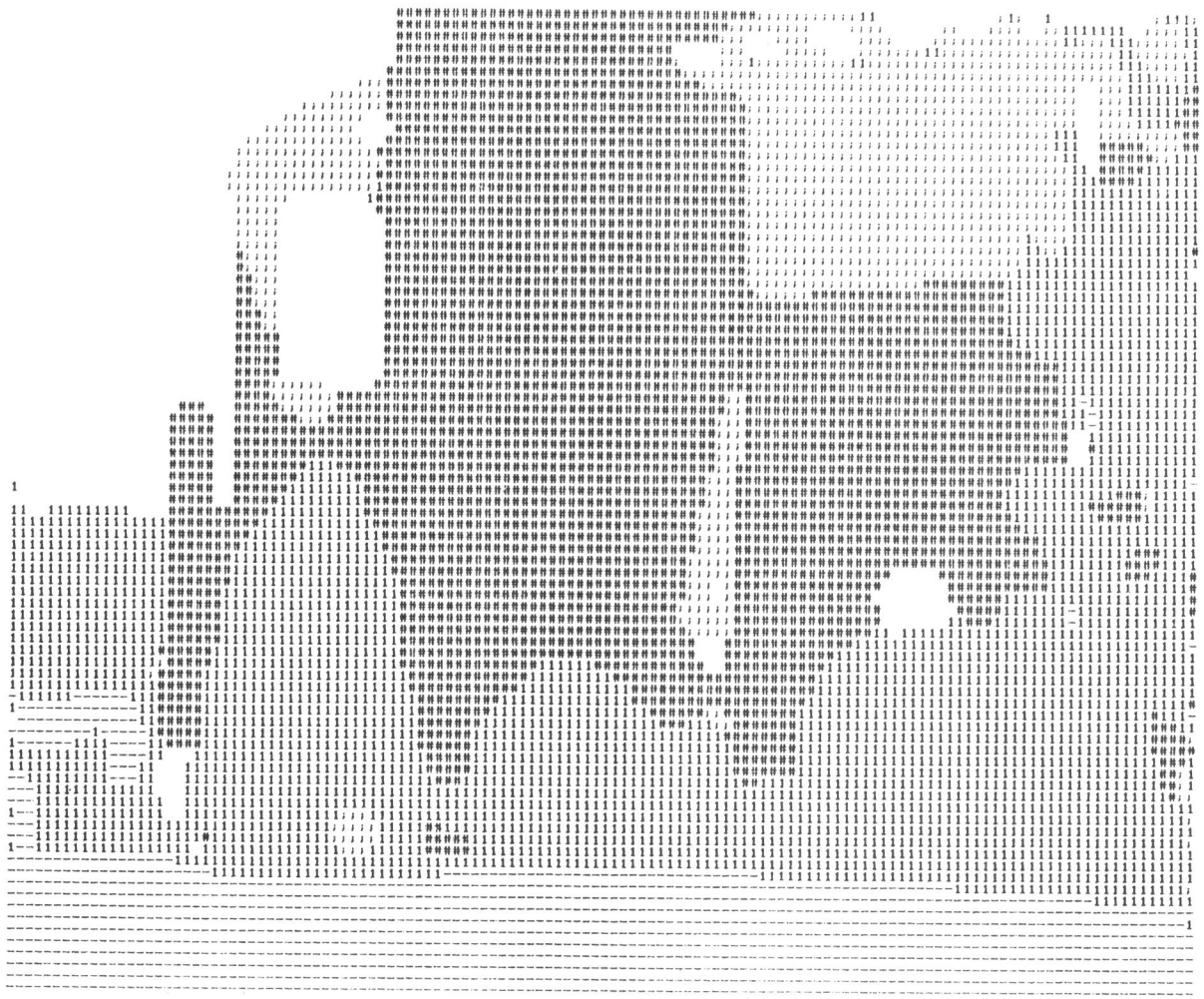
Thus, each $P_i^{(k)}(\lambda)$ is multiplied by $[1 + q_i^{(k)}(\lambda)]$, and the values are normalized at each object. The relaxation process is iterated until some termination criterion is met.

The following are some theoretical flaws of nonlinear relaxation.

1) Independent Labeling: Pavlidis [13] noted the following: if the labeling of every node is independent from that of its neighbors, then $r_{ij}(\lambda, \alpha) = 0$ for $i \neq j$, $r_{ii}(\lambda, \lambda) = 1$, and $r_{ii}(\lambda, \alpha) = -1$ when $\lambda \neq \alpha$. Under these conditions, (15) becomes

$$q_i^{(k)}(\lambda) = \frac{1}{n}\left[P_i^{(k)}(\lambda) - \sum_{\alpha \neq \lambda} P_i^{(k)}(\alpha)\right]$$

$$= \frac{1}{n}\,[2P_i^{(k)}(\lambda) - 1] \tag{17}$$



(c)

Fig. 2. (*Continued*).

since $\Sigma_\alpha P_i^{(k)}(\alpha) = 1$. Using (17) in (16), we obtain for the updating rule

$$P_i^{(k+1)}(\lambda) = \frac{P_i^{(k)}(\lambda) \cdot \left[1 + \frac{2}{n} P_i^{(k)}(\lambda) - \frac{1}{n}\right]}{1 - \frac{1}{n} + \frac{1}{n} \sum_\alpha [P_i^{(k)}(\alpha)]^2}. \tag{18}$$

This rule does not meet the natural expectation: the labeling should not change when the nodes are independent, and thus the information they provide is irrelevant. Pavlidis also proved that the updating rule (18) converges to a limit independent of $P^{(0)}$.[1]

2) "No Information" Labeling: A "no information" labeling is one that does not affect our existing knowledge. For every label λ and node v_i we have an *a priori* probability $\text{Prob}(l_i = \lambda)$ that node v_i will be labeled λ, without any relevant measurements about that node. A node v_i labeled by a labeling that satisfies

$$P_i(\lambda) = \text{Prob}(l_i = \lambda) \tag{19}$$

is said to be labeled with a "no information" labeling. It is desirable that when all neighbors of a node v_i are labeled with "no information" labelings, the updating rule should not change the labeling of v_i. But since the *a priori* probabilities are not incorporated in (16) in any way, this cannot happen.

[1]This does not occur when $r_{ii}(\lambda, \alpha) = 0$, but this requirement is not generally acceptable.

B. The New Relaxation Scheme

The new relaxation scheme described in Section II of this paper uses the following expression to compute new probability estimates:

$$P_{ij}^{(k+1)}(l_i = \alpha) = \frac{P_i^{(k)}(l_i = \alpha) \cdot \sum_{\beta \in \Lambda} P_j^{(k)}(l_j = \beta) \cdot r_{ij}(\alpha, \beta)}{\sum_{\lambda \in \Lambda} P_i^{(k)}(l_i = \lambda) \sum_{\beta \in \Lambda} P_j^{(k)}(l_j = \beta)\, r_{ij}(\lambda, \beta)} \tag{20}$$

where the coefficients r_{ij} are defined by

$$r_{ij}(\alpha, \beta) = \frac{\text{Prob}(l_i = \alpha, l_j = \beta)}{\text{Prob}(l_i = \alpha)\, \text{Prob}(l_j = \beta)}. \tag{21}$$

We will now show that faults found in the nonlinear relaxation are not found in the new relaxation.

1) Independent Labeling: In the probabilistic scheme, independence between nodes v_i and v_j is denoted by

$$\text{Prob}(l_i = \alpha \,|\, l_j = \beta) = \text{Prob}(l_i = \alpha) \tag{22}$$

for all labels α and β.

Assuming independence between all pairs of distinct nodes, we get from (21) and (22) that

$$r_{ij}(\alpha,\beta) = \frac{\text{Prob}(l_i=\alpha, l_j=\beta)}{\text{Prob}(l_i=\alpha)\,\text{Prob}(l_j=\beta)} = \frac{\text{Prob}(l_i=\alpha \,|\, l_j=\beta)}{\text{Prob}(l_i=\alpha)} = 1 \tag{23}$$

for all nodes v_i, v_j and all labels of α, β. Using (23) and (20) we get

$$P_{ij}^{(k+1)}(l_i=\alpha) = \frac{P_i^{(k)}(l_i=\alpha)\sum_{\beta\in\Lambda} P_j^{(k)}(l_j=\beta)}{\sum_{\lambda\in\Lambda} P_i^{(k)}(l_i=\lambda)\sum_{\beta\in\Lambda} P_j^{(k)}(l_j=\beta)} = P_i^{(k)}(l_i=\alpha) \tag{24}$$

since $\Sigma_{\beta\in\Lambda} P_j^{(k)}(l_j=\beta)=1$. The new estimate from each neighbor is exactly the old probability estimate, and any average of these will also be identical to the old estimate. Thus, the new relaxation scheme has the desired property that the estimates do not change when all nodes are independent.

2) "No Information" Labeling: As in Section VI-A2 a "no information" labeling at a node v_j occurs when

$$P_j^{(k)}(l_j=\alpha) = \text{Prob}(l_j=\alpha)$$

for all labels α. When a new probability estimate for a node v_i is computed from the labeling at v_j using the updating rule (20), we have

$$P_{ij}^{(k+1)}(l_i=\alpha) = \frac{P_i^{(k)}(l_i=\alpha)\cdot\sum_{\beta\in\Lambda}\text{Prob}(l_j=\beta)\,r_{ij}(\alpha,\beta)}{\sum_{\lambda\in\Lambda} P_i^{(k)}(l_i=\lambda)\sum_{\beta\in\Lambda}\text{Prob}(l_j=\beta)\,r_{ij}(\lambda,\beta)}. \tag{25}$$

Using the definition of r_{ij} from (21), we get

$$P_{ij}^{(k)}(l_i=\alpha) = \frac{P_i^{(k)}(l_i=\alpha)\cdot\sum_{\beta\in\Lambda}\text{Prob}(l_i=\alpha, l_j=\beta)/\text{Prob}(l_i=\alpha)}{\sum_{\lambda\in\Lambda} P_i^{(k)}(l_i=\lambda)\cdot\sum_{\beta\in\Lambda}\text{Prob}(l_i=\lambda, l_j=\beta)/\text{Prob}(l_i=\lambda)} = \frac{P_i^{(k)}(l_i=\alpha)\cdot\sum_{\beta\in\Lambda}\text{Prob}(l_j=\beta \,|\, l_i=\alpha)}{\sum_{\lambda\in\Lambda} P_i^{(k)}(l_i=\lambda)\sum_{\beta\in\Lambda}\text{Prob}(l_j=\beta \,|\, l_i=\lambda)} = P_i^{(k)}(l_i=\alpha) \tag{26}$$

since $\text{Prob}(l_i=\alpha,\ l_j=\beta)/\text{Prob}(l_i=\alpha) = \text{Prob}(l_j=\beta|l_i=\alpha)$, and $\Sigma_{\beta\in\Lambda}\text{Prob}(l_j=\beta|l_i=\alpha)=1$.

This shows that when a node has a "no information" labeling, it does not affect the new probability estimates based on it.

VII. Concluding Remarks

The new relaxation scheme has the appeal of being theoretically developed, and gives the desired results in the special cases described in the paper. Its performance, which is comparable to those of the "old" relaxation on the pixel classification problems treated in this paper, does not offset this appeal. A flexible definition of the neighborhood of a node enabled this approach to be successfully applied to handwriting recognition [6] and to breaking substitution ciphers [5].

Acknowledgment

The author gratefully acknowledges the help of K. Riley in preparing this paper.

References

[1] J. O. Eklundh, H. Yamamoto, and A. Rosenfeld, "Relaxation methods in multispectral pixel classification," Comput. Sci. Center, Univ. Maryland, Rep. TR-662, July 1978.
[2] J. O. Eklundh and A. Rosenfeld, "Convergence properties of relaxation," Comput. Sci. Center, Univ. Maryland, Rep. TR-701, Oct. 1978.
[3] R. Kirby, "A product rule relaxation method," *Comput. Graphics Image Processing*, to be published.
[4] S. Peleg and A. Rosenfeld, "Determining compatibility coefficients for curve enhancement relaxation processes," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-8, pp. 548–554, July 1978.
[5] —, "Breaking substitution ciphers using a relaxation algorithm," *Commun. Ass. Comput. Mach.*, vol. 22, pp. 598–605, Nov. 1979.
[6] S. Peleg, "Ambiguity reduction in handwriting with ambiguous segmentation and uncertain interpretation," *Comput. Graphics Image Processing*, vol. 10, pp. 235–245, July 1979.
[7] —, "A note on the advantages of a new probabilistic relaxation scheme," Comput. Sci. Center, Univ. Maryland, Rep. TR-739, Feb. 1979.
[8] A. Rosenfeld, R. A. Hummel, and S. W. Zucker, "Scene labeling by relaxation operations," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-6, pp. 420–433, June 1976.
[9] S. W. Zucker, R. A. Hummel, and A. Rosenfeld, "An application of relaxation labeling to line and curve enhancement," *IEEE Trans. Comput.*, vol. C-26, pp. 394–403, Apr. 1976.
[10] S. W. Zucker, E. V. Krishnamurthy, and R. Haar, "Relaxation processes for scene labeling: Convergence, speed, and stability," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-8, pp. 41–48, Jan. 1978.
[11] S. W. Zucker and J. L. Mohammed, "Analysis of probabilistic relaxation labeling processes," in *Proc. IEEE Conf. on Image Processing and Pattern Recognition*, Chicago, IL, 1978, pp. 307–312.
[12] S. W. Zucker, Y. G. Leclerc, and J. L. Mohammed, "Continuous relaxation and local maxima selection: Conditions for equivalence," Comput. Vision and Graphics Lab., McGill Univ., Rep. 78-157, Dec. 1978.
[13] T. Pavlidis, *Structural Pattern Recognition*. New York: Springer Verlag, 1977.

New Methods for Picture Reconstruction: Recursive and Causal Techniques

NELSON D. A. MASCARENHAS AND LUIZ F. V. FERNANDES

***Abstract*—The methods proposed by Kashyap and Mittal [1], [2] for the reconstruction of pictures from projections are reformulated under a regression model. These reinterpretations are based on least-squares and Bayesian formulations of a statistical linear model and lead to the derivation of recursive and causal algorithms which are more efficient. Experimental results with simulation of the filters are also presented.**

***Index Terms*—Causal filters, picture processing, picture reconstruction, recursive estimation, regression analysis, stochastic approximation.**

Manuscript received October 3, 1977; revised October 22, 1979. This work was supported by the Banco Nacional de Desenvolvimento Economico, Brazil, under Contract 211.

N. D. A. Mascarenhas was with the Instituto Tecnológico de Aeronáutica, Centro Técnico Aeroespacial, São José dos Campos, S.P., Brazil. He is now with the Instituto de Pesquisas Espaciais, Conselho Nacional de Desenvolvimento Científico e Tecnológico, São José dos Campos, S.P., Brazil.

L. F. V. Fernandes is with the Instituto Tecnológico de Aeronáutica, Centro Técnico Aeroespacial, São José dos Campos, S.P., Brazil.

0162-8828/80/0700-0369$00.75