MONITORING RELAXATION ALGORITHMS USING LABELING EVALUATIONS

Shmuel Peleg*

Computer Vision Laboratory
Computer Science Center
University of Maryland
College Park, MD 20742

ABSTRACT

Relaxation algorithms employ an initial stochastic classification and a probabilistic model for deferring final decision. In such algorithms it is desirable to evaluate a classification based on an initial stochastic classification and a probabilistic model. Such evaluation can monitor the classification and ensure reasonable results. Specific classifications are evaluated, rather than evaluating the intermediate probabilistic classifications as was considered in previous work. Since relaxation algorithms are not guaranteed to converge to a reasonable solution, monitoring them is useful as a stopping criterion.

1. Introduction

Many classification tasks can be formulated as graph labeling. A node in the graph represents an object to be classified, a label represents a class, and constraints regarding the classes of pairs of objects are associated with the arcs. In contrast to search algorithms, which try different assignments of labels to nodes until an acceptable labeling is found, relaxation algorithms defer the choice of a particular label by assigning to the node a probability vector over all possible labels.

Relaxation algorithms [7,8,9,12] use a probabilistic model to update an initial stochastic labeling based on local neighborhoods, and to get an improved stochastic labeling. A probabilistic model can be, as in [9], the discrete joint distribution of all pairs of labels at neighboring pairs of nodes.

Several measures have been suggested for the evaluation of the intermediate stochastic labeling produced by relaxation [4,13]. However, when a stochastic labeling has an excellent evaluation, the specific labeling derived from it (by maximum selection, for example) can be useless. Since the final step in any relaxation process is almost always a selection of a specific labeling, evaluation of these labelings will be a better indication of the merit of the classification than the evaluation of the intermediate stochastic labeling.

*Present address: Dept. of Mathematics, Ben-Gurion University of the Negev, Beersheva, Israel.

Since two sources of knowledge are used in probabilistic classification, the inital stochastic labeling and the probabilistic model, there are also two corresponding parts for the evaluation: the probability of a specific labeling given the initial stochastic labeling, and the probability of a labeling given the probabilistic model.

In this paper a scene classification example is used. A color picture of a house is to be classified into five regions: brick, sky, grass, shadow, and brush. The initial stochastic classification was performed by clustering in color space [3]. When the maximal label is chosen at every pixel, a possible specific classification results. We also have a hand classification of the scene. After each iteration of relaxation, a specific labeling was constructed from the stochastic labeling, and compared to the hand segmentation. The percent of points different from the hand segmentation is shown in Table 1. Results are given for two relaxation schemes described in detail in [9]. Case (a) involves an updating based on all four neighbors of a node, and case (b) involves a product rule based on pairwise compatibilities.

iteration	a	b
0	4.60	4.60
1	4.07	3.88
2	3.92	3.93
3	3.86	3.98
4	3.85	4.04
10	3.73	4.37

Table 1. The percent of pixels whose classes are different from the hand classification, for the iterations of the following two relaxation schemes: (a) Updating based on four neighbors. (b) Product rule using pairwise relations.

2. Probabilities of labelings

Let $V = \{v_1,\ldots,v_n\}$ be a set of nodes, let $E^k \subseteq V^k$, $k \le n$, be a set of arcs (or relations), and let Λ be a set of labels. The random variable ℓ_i will designate the label associated with node v_i.

A stochastic labeling of the network assigns to each node v_i a probability vector $p_i: \Lambda \to [0,1]$. $p_i(\ell_i=\alpha)$ is the probability of assigning label α to node v_i. The stochastic labeling is initially constructed using a classifier, which probabilistically classifies the objects that correspond to the nodes using their individual features.

CH1499-3/80/0000-0054$00.75©1980 IEEE

A probabilistic model is a set of a priori probability distributions on E^k. For every $\langle v_1,\ldots,v_k\rangle \in E^k$, $P(\ell_1=\alpha_1,\ldots,\ell_k=\alpha_k)$ designates the a priori probability of nodes $v_1,\ldots,v_k$ being labeled $\alpha_1,\ldots,\alpha_k$ respectively.

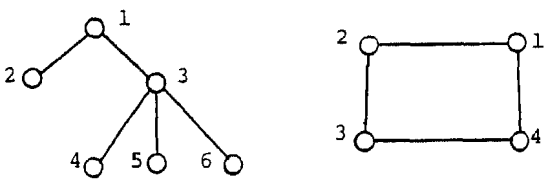
Every assignment of labels $\Omega = \alpha_1,\ldots,\alpha_n$ to $v_1,\ldots,v_n$ has two probabilities: the probability assigned to it by the stochastic labeling, $P_S(\Omega)$, and the probability assigned to it by the probabilistic model, $P_M(\Omega)$. The computation of P_S and P_M will now be discussed, and will follow the treatment in [10] .

2.1 The model probability P_M

To find P_M, we need the joint probability distribution of all the nodes, but we know only the distributions for some subsets of nodes. Extending probability distributions from subsets is a well known problem, and some methods for doing it are discussed in [1,2,5]. These methods can be applied to some graphs but are not practical for large and complicated graphs.

The most convenient graph for purposes of extension is a tree, under the assumption that the joint probabilities of every two nodes connected by an arc are given. For the dependency tree in Fig. 1(a), the maximum entropy extension to the pairwise probabilities for $\Omega = \alpha_1,\ldots,\alpha_6$ is

$$P(\ell_1=\alpha_1)\cdot P(\ell_2=\alpha_2|\ell_1=\alpha_1)\cdot P(\ell_3=\alpha_3|\ell_1=\alpha_1)$$
$$\cdot P(\ell_4=\alpha_4|\ell_3=\alpha_3)\cdot P(\ell_5=\alpha_5|\ell_3=\alpha_3)\cdot P(\ell_6=\alpha_6|\ell_3=\alpha_3).$$



a. Tree dependency b. Cyclic dependency

Figure 1. Graph dependencies.

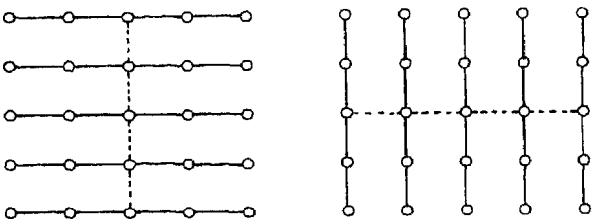
In general, it can be seen that for any tree G=(V,E), the extended probability for $\Omega = \alpha_1,\ldots,\alpha_n$ will be

$$P_M(\Omega) = \frac{\prod_{(v_i,v_j)\in E} P(\ell_i=\alpha_i,\ell_j=\alpha_j)}{\prod_{i=1}^{n} P(\ell_i=\alpha_i)^{N_i-1}} \qquad (1)$$

where N_i is the degree of node v_i. It can be shown that this extension is the maximum entropy extension having marginals which are identical to the given joint probabilities.

Expression (1) is correct only for tree (or forest) dependencies. For other dependencies, such as the one in Fig. 1(b), it is not valid. In such cases, the marginals of the distribution obtained when (1) is used will differ from the given pairwise probabilities. However, an approximation can be found by using a spanning tree of the given graph, as suggested in [2].

As the canonical spanning trees for rectangular arrays used for picture representation, the two "forests" in Fig. 2 are used. The horizontal spanning forest of Figure 2(a) takes into account only, but all, horizontal joint probabilities, and the vertical forest of Figure 2(b) uses vertical probabilities. To include both horizontal and vertical dependencies, the two probabilities can be combined by an average. In the rest of the paper the geometric mean is used for its ease of computation. Since the probability computation is done using logs, the arithmetic average of the logs is the log of the geometric mean.



a. Horizontal dependency. b. Vertical dependency

Figure 2. Canonical spanning forests for a rectangular array.

The log of the model probability P_M for labeling obtained by the relaxation iterations on the house picture is shown in Table 2. The model probability, as expected, increases with every iteration of relaxation, since additional model effects are introduced with every iteration.

iteration	a	b
0	-4.25	-4.25
1	-3.88	-3.68
2	-3.76	-3.52
3	-3.71	-3.48
4	-3.68	-3.43
10	-3.63	-3.38

Table 2. The log of P_M for the two schemes of Table 1 (multiplied by 10^{-3}).

2.2 The initial-labeling probability P_S

The probability P_S is based on the initial stochastic labeling of the graph. Since the initial classification is made for individual nodes, and no dependencies are given, there is no spanning tree. The only possible spanning forest is the collection of all individual nodes. Thus, the probability extension based on the initial labeling can only be the product of the individual label probabilities. Given the labeling $\Omega = \alpha_1,\ldots,\alpha_n$, the probability extension from the individual initial stochastic labeling $p_i^{(0)}(\ell_i=\alpha_i)$ is

$$P_S(\Omega) = \prod_{i=1}^{n} p_i^{(0)}(\ell_i=\alpha_i) \qquad (2)$$

Expression (2) is a special case of expression (1) of the previous section when E, the set of arcs, is empty.

The log of P_S for the picture relaxation example is shown in Table 3. As expected, P_S is maximal at the initial classification, since it was created directly from the initial classification. At every iteration P_S decreases, since repeated application of the probabilistic model at every

iteration weakens the effect of the initial labeling.

iteration	a	b
0	-3.66	-3.66
1	-5.11	-8.59
2	-6.20	-11.89
3	-6.95	-13.38
4	-7.64	-14.46
10	-8.93	-17.18

Table 3. The log of P_S for the two cases (multiplied by 10^{-2}).

3. Evaluation assuming independence

Assuming independence between the model and the initial labeling, the probability of a labeling Ω under both conditions is the product of the two probabilities. Thus, we get

$$P(\Omega) = P_S(\Omega)\cdot P_M(\Omega) \quad . \tag{3}$$

Among several labelings, the preferred one is the labeling having highest value of $P(\Omega)$.

Table 4 shows $\log(P(\Omega)) = \log(P_S(\Omega))+\log(P_M(\Omega))$ for the two picture relaxation cases. $P(\Omega)$ is maximal in case (b) after the first iteration, and this result agrees with the minimum difference (Table 1). In case (a), however, the optimal classification according to $P(\Omega)$ is after the second iteration, while in Table 1 the percent of different points is decreasing constantly. But of the total reduction of 0.87% of different points in all ten iterations, 0.68% is achieved in the first two iterations. Since usually no hand classification is available, the use of $P(\Omega)$ can suggest stopping the iterations when no improvement is obtained. Even when $P(\Omega)$ suggested stopping earlier than the best solution, this was not before most of the effect was accomplished.

iteration	a	b
0	-4.61	-4.61
1	-4.39	-4.54*
2	-4.38*	-4.71
3	-4.41	-4.82
4	-4.44	-4.88
10	-4.53	-5.10

Table 4. The log of $P(\Omega)$ for the two cases. The optimal solutions are marked with * (multiplied by 10^{-3}).

4. Evaluation using information theory

The classification problem can be viewed as information transmitted through a memoryless channel. A channel has an input alphabet, an output alphabet, and a probability matrix. The probability of any output symbol being emitted when α is entered is given in the probability matrix. In the classification problem, the source is the collection of objects with their "true" classification, the channel represents the information collection and transmission, and the receiver contains the local measures that can be used for classification. In image processing, for example, the source will be the scene with "house", "sky", and "bushes", the channel will represent the digitization and sampling of a picture of the scene, and the receiver will have the gray-level information.

When the received measurements are $a_1,\ldots,a_n$, a maximum likelihood labeling $\Omega = \alpha_1,\ldots,\alpha_n$ is that labeling which maximizes $\text{Prob}(\Omega|\, a_1,\ldots,a_n)$. This probability can be computed as follows: using Bayes' rule we have

$$\text{Prob}(\Omega|\, a_1,\ldots,a_n) = \frac{\text{Prob}(a_1,\ldots,a_n|\Omega)\cdot\text{Prob}(\Omega)}{\text{Prob}(a_1,\ldots,a_n)} \quad .$$

In a memoryless channel there is conditional independence,

$$\text{Prob}(a_1,\ldots,a_n|\Omega) = \prod_{i=1}^{n} \text{Prob}(a_i|\ell_i=\alpha_i).$$

Using the conditional independence and Bayes' rule we get

$$\text{Prob}(\Omega|\, a_1,\ldots,a_n) = \frac{\prod_{i=1}^{n}\text{Prob}(a_i)}{\text{Prob}(a_1,\ldots,a_n)} \cdot \prod_{i=1}^{n}\text{Prob}(\ell_i=\alpha_i|\, a_i)\cdot\frac{\text{Prob}(\Omega)}{\prod_{i=1}^{n}\text{Prob}(\ell_i=\alpha_i)}$$

Since the first factor in the above expression does not depend at all on Ω, and is constant for the given measurements $a_1,\ldots,a_n$, we can delete it for maximization purposes. $\text{Prob}(\Omega)$ is the a priori probability of Ω before any measures are taken, and is $P_M(\Omega)$ as described in Section 2.1. $\text{Prob}(\ell_i=\alpha_i|\, a_i)$ is the probability of ℓ_i being α_i based on the local measurement a_i, and it is the initial labeling $p^{(0)}(\ell_i=\alpha_i)$ given by the local detector. $\text{Prob}(\ell_i=\alpha_i)$ is the a priori probability of ℓ_i being α_i, and is the marginal of the joint distributions in the model. With these used, the above expression can be rewritten as

$$\text{Prob}(\Omega|\, a_1,\ldots,a_n) \tilde{=} I(\Omega) = \frac{P_S(\Omega)\cdot P_M(\Omega)}{\prod_{i=1}^{n}\text{Prob}(\ell_i=\alpha_i)} \quad . \tag{4}$$

We can see that the measure $I(\Omega)$ defined in (4) differs from the measure $P(\Omega)$ of (3) only by being divided by the product of the a priori probabilities of the labels in Ω. This division acts to increase the relative merit of labelings having labels with low a priori probability.

Table 5 displays $\log(I(\Omega))$ for the two picture relaxation cases. The optimum for case (b) is the same as when using $P(\Omega)$ [Section 3 and Table 4], and the optimum for case (a) is after the third iteration: slightly better than $P(\Omega)$ when compared with Table 1.

iteration	a	b
0	1.323	1.323
1	1.352	1.343*
2	1.357	1.330
3	1.358*	1.323
4	1.356	1.318
10	1.350	1.299

Table 5. The log of $I(\Omega)$ for the two cases. The optimal solutions are marked with * (multiplied by 10^{-4}).

5. Zero values

Since the measures $P(\Omega)$ and $I(\Omega)$ involve a product of joint probabilities, one pair of nodes with contradictory labels having a zero joint probability will cause the entire measure to be zero. In such a case, we need to minimize the number of contradictions.

The case of solving substitution ciphers by relaxation [12] is a good example. In a substitution cipher, every letter of the alphabet is replaced consistently by another letter. The key to the cipher is the permutation that maps every code letter into the original plaintext letter. In the relaxation, trigram probabilities derived from a large English text were used. When trigrams in the ciphered messages did not occur in the reference text used to compute the trigram probabilities, the evaluation measures will be zero even when the correct key is used.

Table 6 displays an example from [12]. For every iteration the suggested key is compared to the correct key, and the number of contradictions when using the key is also displayed. The number of contradictions is reduced with the iterations. Since relaxation failed to find the correct key, it is shown last for comparison. Its lowest number of contradictions suggests that using minimization algorithms with this measure could find the correct key.

a	b	c
0	18	337
1	14	301
2	9	143
3	7	48
4	6	42
5	4	12
6	3	10
7	2	9
8	2	9
9	2	9
10	1	8
11	1	8
*	0	1

Table 6. Iterations of relaxation for substitution ciphers. (a) iteration number; (b) number of incorrect entries; (c) number of contradictions. * is the correct key shown for comparison.

6. Concluding remarks

This paper has suggested a method of probabilistic evaluation of labelings derived by relaxation algorithms, based on the initial stochastic labeling and the probabilistic model. These measures can be used as termination criteria for relaxation, and using them prevents the relaxation from arriving at unreasonable results. These evaluations could also guide a search to find a labeling with a high measure, when relaxation cannot be applied or yields unsatisfactory results.

More examples of the reliability of the suggested measure are given in [11], from the domains of handwriting recognition and breaking substitution ciphers.

Acknowledgment

The support of the National Science Foundation under Grant MCS-76-23763 is gratefully acknowledged, as is the help of Kathryn Riley in preparing this paper.

References

1. D. T. Brown, A note on approximations to discrete probability distributions, Information and Control 2, 1959, 386-392.
2. C. K. Chow and C. N. Liu, Approximating discrete probability distributions and dependency trees, IEEE Trans. Information Theory IT-14, 1968, 462-467.
3. J. O. Eklundh, H. Yamamoto, and A. Rosenfeld, Relaxation methods in multispectral pixel classification, TR-662, Computer Science Center, University of Maryland, July 1978.
4. O. Faugeras and M. Berthod, Scene labeling: an optimization approach, Proceedings IEEE Conference on Pattern Recognition and Image Processing, Chicago, August 1979, 318-326.
5. P. M. Lewis, Approximating probability distributions to reduce storage requirements, Information and Control 2, 1959, 386-392.
6. S. Y. Lu and K. S. Fu, Stochastic tree grammar inference for texture synthesis and discrimination, Computer Graphics and Image Processing 9, 1979, 234-245.
7. A. Rosenfeld, Iterative methods in image analysis, Pattern Recognition 10, 1978, 181-187.
8. A. Rosenfeld, R. A. Hummel, and S. W. Zucker, Scene labeling by relaxation operations, IEEE Trans. Systems, Man, Cybernetics SMC-6, 1976, 420-433.
9. S. Peleg, A new probabilistic relaxation scheme, IEEE Trans. Pattern Analysis and Machine Intelligence PAMI-2, 1980, 362-369.
10. S. Peleg, Labeling evaluation in probabilistic networks, Information Sciences, in press.
11. S. Peleg, Monitoring relaxation algorithms using labeling evaluation, TR-842, Computer Science Center, University of Maryland, December 1979.
12. S. Peleg and A. Rosenfeld, Breaking substitution ciphers using a relaxation algorithm, Communications of the ACM 22, 1979, 598-605.
13. S. W. Zucker, Y. G. Leclerc, and J. L. Mohammed, Continuous relaxation and local maxima selection: conditions for equivalence, Report No. 78-15R, Computer Vision and Graphic Laboratory, McGill University, December 1978.