REFERENCES

[1] M. L. Tsetlin, "On the behavior of finite automata in random media," *Automat. Remote Contr.*, vol. 22, pp. 1345–1354, 1961.
[2] V. I. Krinskii, "An asymptotically optimal automaton with exponential convergence," *Biophysics*, vol. 9, pp. 484–487, 1964.
[3] V. I. Varshavskii and I. P. Vorontsova, "On the behavior of stochastic automata with variable structure," *Automat. Remote Contr.*, vol. 24, pp. 327–333, 1963.
[4] K. S. Fu, "Learning control systems—Review and outlook," *IEEE Trans. Automat. Contr.*, vol. AC-15, pp. 210–221, Apr. 1970.
[5] K. S. Narendra and M. A. L. Thathachar, "Learning automata—A survey," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-4, pp. 323–334, 1974.
[6] K. S. Narendra, A. E. Wright, and L. G. Mason, "Application of learning automata to telephone traffic routing," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-7, no. 11, pp. 785–792, 1977.
[7] P. R. Srikantakumar and K. S. Narendra, "A learning model for routing in telephone networks," *SIAM J. Control Opt.*, vol. 20, no. 1, pp. 34–57, 1982.
[8] K. S. Narendra and P. Mars, "The use of learning algorithms in telephone traffic routing—A methodology," *Automatica*, vol. 19, no. 5, pp. 495–502, 1983.
[9] M. S. Chrystall, K. S. Narendra, and P. Mars, "Learning automata adaptive routing for datagram and virtual call packet switched networks," (submitted for publication to *IEEE Trans. Commun.*).
[10] V. Yu Krylov and M. L. Tsetlin "Games between automata," *Automat. Remote Contr.*, vol. 24, pp. 889–900, 1963.
[11] B. Chandrasekaran and D. W. C. Shen, "Stochastic automata games," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-5, pp. 145–149, Apr. 1969.
[12] R. Viswanathan and K. S. Narendra, "Competitive and cooperative games of variable-structure stochastic automata," *J. Cybern.*, vol. 3, no. 1, pp. 1–23, 1973.
[13] S. Lakshmivarahan and K. S. Narendra, "Learning algorithms for two person zero-sum stochastic games with incomplete information," *Math. Oper. Res.*, vol. 6, no. 4, 1981.
[14] ——, "Learning algorithms for two person zero-sum stochastic games with incomplete information: A unified approach," *SIAM J. Contr. Opt.*, vol. 20, no. 4, pp. 541–552, 1982.
[15] M. A. L. Thathachar and K. R. Ramakrishnan, "On line optimization with a team of learning automata," paper presented at *IFAC Symposium on Theory and Applications of Digital Control*, New Delhi, India, Jan. 1982.
[16] S. Lakshmivarahan, *Learning Algorithms: Theory and Applications*. New York: Springer-Verlag, 1981.
[17] M. F. Norman, *Markov Processes and Learning Models*. New York: Academic, 1972.
[18] K. R. Ramakrishnan, "Hierarchical systems and co-operative games of learning automata," Ph.D. dissertation, Dept. Eng., Indian Institute of Science, Bangalore, India, 1982.
[19] S. Lakshmivarahan and M. A. L. Thathachar, "Absolutely expedient learning algorithms for stochastic automata," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, pp. 281–286, 1973.
[20] K. S. Narendra and R. M. Wheeler, "An N-player sequential stochastic game with identical payoffs," Center for Systems Science Technical Rep. 8209, Yale University, New Haven, CT, Aug. 1982.

Classification and Tracking Using Local Optimization

SHMUEL PELEG AND ALLON NATHAN

***Abstract*—When a heuristic function is available to evaluate classifications, a special search procedure is applied to find a classification optimizing this function. A specific application to image segmentation is presented, including several examples. The major difference between this approach and previous optimization attempts is the use of deterministic rather than probabilistic classifications. The approach is also applied to object tracking in image sequences.**

Manuscript received November 23, 1981; revised April 29, 1983. This work was supported by the Hebrew University's Central Research Foundation.

The authors are with the Department of Computer Sciences, The Hebrew University of Jerusalem, 91904 Jerusalem, Israel.

I. INTRODUCTION

The classification problem can be described as follows: let $V = \{1, \cdots, n\}$ be a set of objects, and let Ω be a set of classes (labels). A classification is an assignment of each object to a unique class. Given a set of measurements on the objects, a classification is sought that will be consistent with the measurements while conforming to some general constraints.

The probabilistic relaxation approach [1], [2] assigns to every object a probability vector over all possible classes. Using an expression designed to improve compatibility among objects, the probability vectors are updated based on the neighboring vectors. The vectors are updated iteratively until satisfactory results are obtained. To get a classification from the probability vectors, every object is assigned to its most probable class. Relaxation methods have been used for various problems, and in some of these they present the only known solution [6]. However, they are not guaranteed to improve classification, and in many cases, after improved classification during the first few iterations, classification deteriorates as the iterations continue. An evaluation method has been suggested [3] to monitor the relaxation and to determine the best classification from all those suggested by the relaxation.

Probabilistic labeling also has an optimization approach [4], [5] which tries to find a probabilistic labeling that maximizes some measure. However, a statistical knowledge of the process is needed, and the relation between an optimal probabilistic labeling and an optimal deterministic classification is not clear.

This correspondence describes a method for finding a classification that optimizes some heuristic merit function. The heuristic function is defined over the deterministic classifications and thus can be developed and understood better than functions over probabilistic classifications. The method has later been successfully applied to the substitution cipher problem [11] with very fast convergence. The next section will describe the method in general, and the following sections will explain its application to image segmentation and object tracking.

II. OPTIMAL DISCRETE CLASSIFICATION

Let $S: V \rightarrow \Omega$ be a classification where every object $i \in V$ is assigned to a class $\lambda \in \Omega$. A heuristic function H assigns a merit to every classification, and a classification that optimizes H is sought.

Given a classification S, we denote by $S[i|\lambda]$ the classification in which object i is assigned the label λ, but which is otherwise identical to S. Given a global merit function $H(S)$, we seek a local merit function h such that given a classification S, $h(S[i|\lambda]) > h(S[i|\mu])$ probably implies $H(S[i|\lambda]) > H(S[i|\mu])$. The function h should be such that it can be computed on a local neighborhood of an object i, and its computation is faster than that of H. Sometimes, a strict implication relation can be achieved between h and H, but since many decisions are made based on that relation, a probable implication can be used since, even if the relation holds in most (but not all) cases, the value of H will still be improved.

The atomic step in the optimization algorithm involves examining one object, and using the local function h to determine the best possible labeling for that object given the current configuration. Namely, the value of $h(S[i|\lambda])$ is computed for *all* labels λ, and a label λ_m is chosen such that $h(S[i|\lambda_m]) \geq h(S[i|\mu])$ for all labels μ. The labeling of that object is changed (if necessary) to λ_m.

To get global improvement of the merit function H, the local atomic step is repeated over all nodes several times. If a parallel machine is available, all nodes can choose their best label and switch to it simultaneously. In such scheme, when all nodes are updated in parallel, a label that seems optimal in the current context might not be optimal in the updated context. Even with

0018-9472/83/1100-1158$01.00

this effect, however, experiments showed that the overall effect still optimizes the global merit H.

When serial updating is performed, several orderings can be used. The simplest is to examine all nodes in some order, and iterate the process several times. Alternatively, an ordering function can be used to focus the effort of the optimization on interesting objects and to avoid computation involving objects whose classification may not be of great importance. The function h itself can serve as an ordering function to process at each step the object that has maximum improvement potential. Initially, h should be applied to all objects, and every object i should be assigned a potential: the improvement in h if the object i changes its label to its optimal label. At each step, only the node with maximum potential will be actually updated. After updating, the node gets a potential of zero (since it already has the optimal label), and all affected neighbors update their potentials.

III. Image Segmentation

In this section the optimal classification method is applied to image segmentation. Martelli and Montanari [7] and Weszka and Rosenfeld [8] discussed the role of "roughness" and discrepancy from the original image in smoothing and thresholding. Narayanan *et al.* [9] applied roughness and discrepancy cost measures in gray level smoothing and segmentation using a steepest descent method. The steepest descent method proved to be computationally expensive. Using the discrete optimization with almost the same cost function provides a fast good-quality segmentation. Another element that is introduced into the cost function is the fit of the segmentation to the gray level edges.

Let $F(x, y)$ be the given image. We look for a labeling $L(x, y)$ that minimizes the cost

$$H = \sum_{x,y} R(x,y) + \alpha \sum_{x,y} D(x,y) + \beta \sum_{x,y} G(x,y) = \sum_{x,y} h(x,y)$$

where R is a roughness measure, D is a discrepancy measure, and G is an edge-fit measure. The global measure H is broken into a sum of local measures, $h(x, y) = R(x, y) + \alpha D(x, y) + \beta G(x, y)$. Minimizing h at a pixel will cause a reduction in the value of H.

The roughness measure can be chosen from among many existing measures for gray level images, notably the digital Laplacian and the digital gradient magnitude. However, since we would like to find the roughness of a labeled image, these were not used. Instead, the global roughness measure for a labeled image $L(x, y)$ was defined as the number of neighboring pixel pairs (in the four-neighbor sense) that have different labels. If we define $(a \neq b)$ to be one if $L(a) \neq L(b)$ and zero otherwise, then the roughness of $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$ is $(a \neq b) + (c \neq d) + (a \neq c) + (b \neq d)$. When a pixel changes its label, the effect on the global measure is only through its relations to the labels of its four immediate neighbors. So, in the configuration

$$\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}, \qquad R(e) = (e \neq b) + (e \neq d) + (e \neq h) + (e \neq f).$$

The global roughness is half the sum of all local pixel roughness. Alternatively, the roughness measure can be defined over an eight-neighborhood, or over any other neighborhood desired.

The discrepancy measure $D(x, y)$ needs some assumptions on the gray level distribution of the objects. If the gray level is assumed to be constant within objects, then all pixels having the same label should also have (after segmentation) the same gray level. Thus a gray level g_λ is associated with every label λ, and g_λ will be the average of the original gray levels of all pixels labeled λ:

$$g_\lambda = \frac{1}{\|L(x,y) = \lambda\|} \sum_{L(x,y)=\lambda} F(x,y)$$

where $F(x, y)$ is the gray level in location (x, y) of the original picture. The local discrepancy, assuming the foregoing model, will be

$$D(x,y) = \left(F(x,y) - g_{L(x,y)}\right)^2.$$

This discrepancy function was used in the experiments reported in the next section.

Haralick uses in his facet model [10] the assumption that gray levels vary linearly within segments. In that case, the best fitting plane P_λ: $I^2 \rightarrow I$ can be found by linear regression for all points labeled λ to minimize the distance

$$\sum_{L(x,y)=\lambda} \left(F(x,y) - P_\lambda(x,y)\right)^2.$$

Once this best fitting plane has been found, the local discrepancy function is defined by

$$D(x,y) = \left(F(x,y) - P_{L(x,y)}(x,y)\right)^2.$$

The third element used in the cost function is the match between segment borders and gray level edges in the original

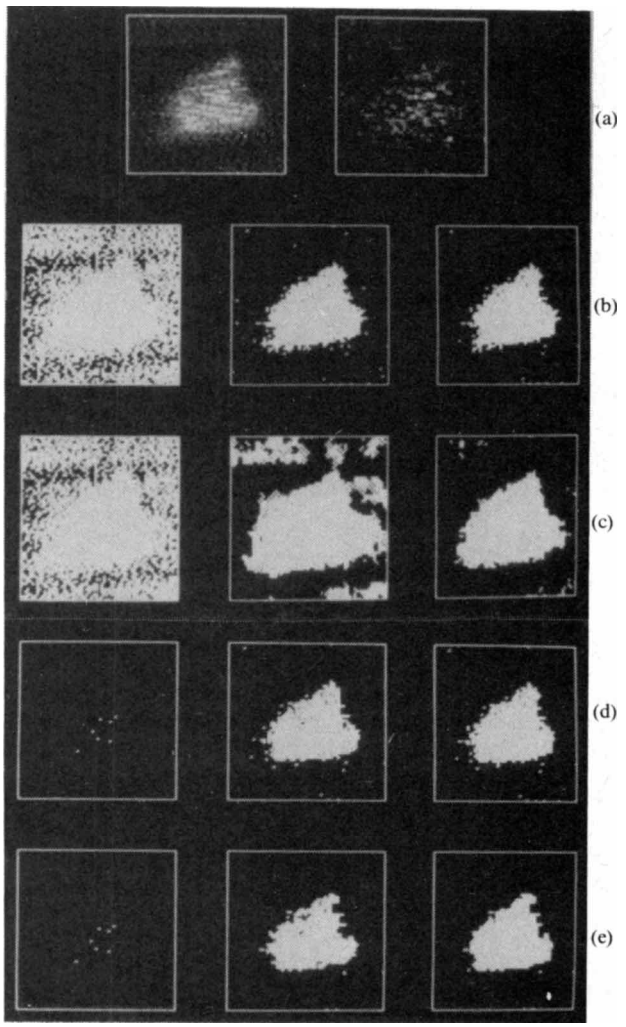


Fig. 1. Segmentation of infrared picture of tank. (a) Original picture (left) and gradient picture. (b)–(e) Initial (left), intermediate (center), and final (right) segmentations for different initial segmentations and weights (see text).

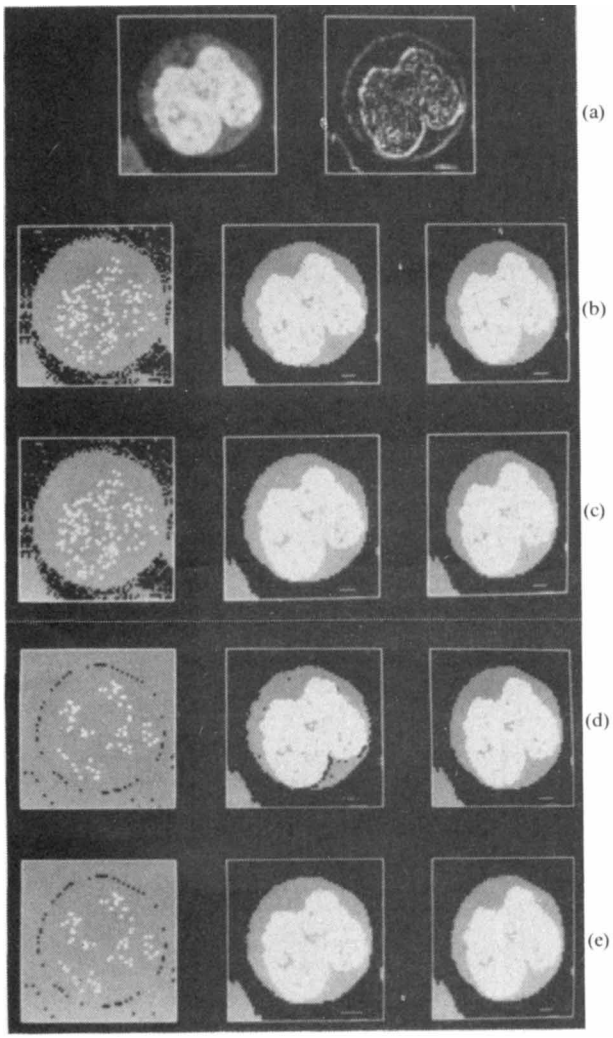


Fig. 2. Same as Fig. 1 but for segmentation of white blood cell.

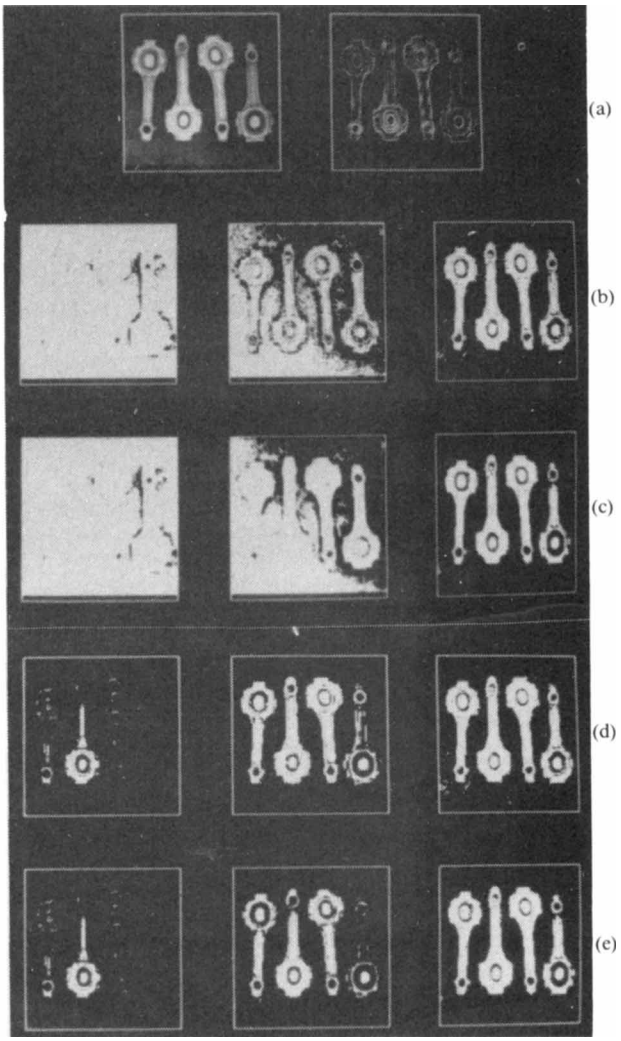


Fig. 3. Same as Fig. 1 but for segmentation of automobile parts.

picture. Let $\nabla(x, y)$ be the digital gradient magnitude value for location (x, y) in the original gray level picture. If segment borders fit gray level edges, they should occur in locations having high gradient. We define an edge-matching cost component as

$$G(x, y) = \begin{cases} -\nabla(x, y), & \text{if } L(x, y) \neq L(x', y') \\ & \text{for some neighbor } (x', y'). \\ \nabla(x, y), & \text{if } L(x, y) = L(x', y') \\ & \text{for all neighbors } (x', y'). \end{cases}$$

This component will decrease the total cost when segment borders also have high gradient value and increase the cost when interior pixels have high gradient value. In our experiments, the Robert's operator was used $\left(\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}^2 + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}^2\right)$, but if necessary, the direction of the gradient can also be compared with the class boundary direction to get an even better figure of merit.

IV. Segmentation Examples

The local measure $h(x, y|\lambda)$ for the label λ at the pixel (x, y) had the following three components in our experiments:

$$h(x, y|\lambda) = R^2 + \alpha D^2 + \beta G,$$

where

- R roughness measure, the number of neighboring pixels having labels different than λ,
- D discrepancy measure, the difference between the original gray level at (x, y) and the gray level associated with the label λ,
- G edge-matching component as described in the previous section.

The weights α and β were used to emphasize different components.

The initial segmentation is the result of thresholding. To show that the optimization process is independent of the initial segmentation, each image was processed twice, starting from two drastically different segmentations, one obtained from a very low threshold, the other from a very high threshold. In practice, we should ordinarily get better initial segmentations than the ones in our examples, and the optimization process will then converge faster.

Fig. 1 shows the process applied to an infrared image of a tank. The first row shows (from left to right) the initial image and the gradient image. Fig. 1(b) and (d) show initial, intermediate, and final segmentations with $\alpha = 0.5$, $\beta = 0$, but with different initial segmentations. Fig. 1(c) and (e) show the same stages for $\alpha = 0.2$, $\beta = 0$. The gradient data were not used since the image is very noisy, and high gradient values do not correspond to object boundaries. These figures also show that even with drastically different initial segmentations, the final segmentation is nearly the same.

Fig. 2 shows a white blood cell which is to be segmented into three classes. The first row is the same as in Fig. 1, while Fig. 2(b)

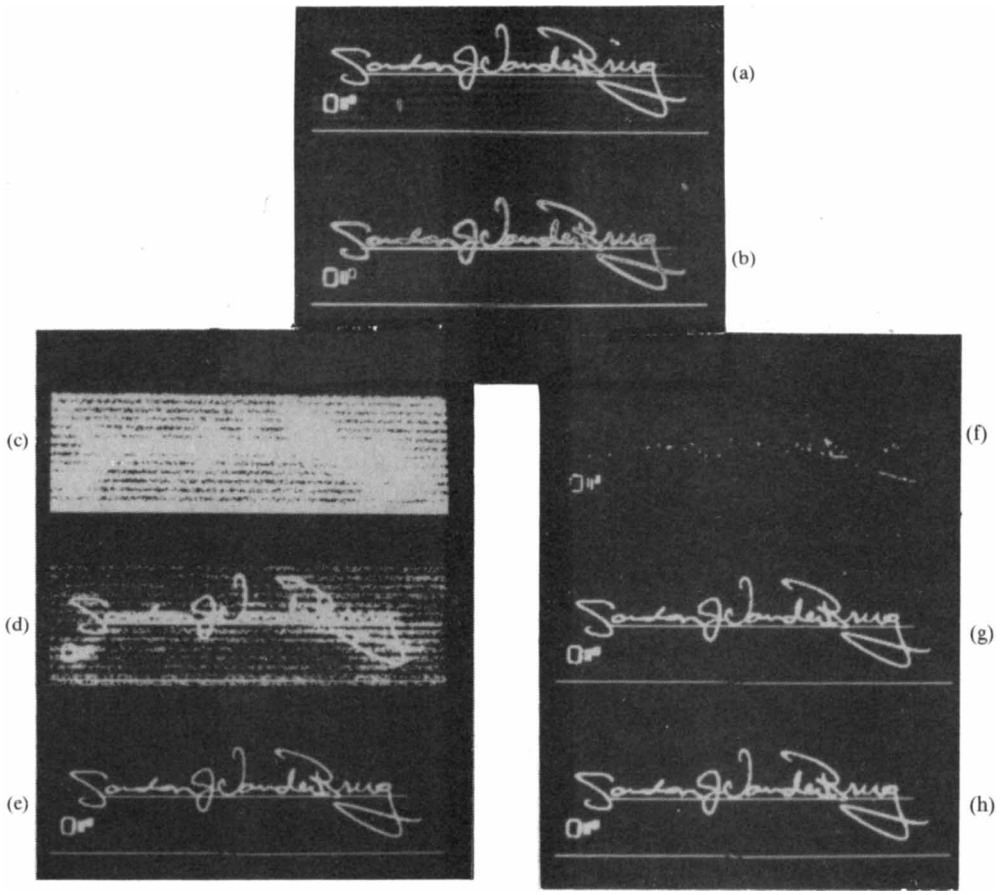
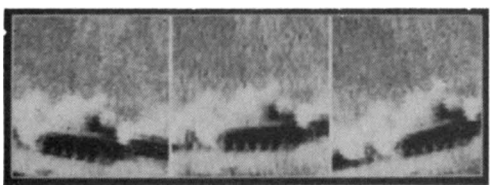
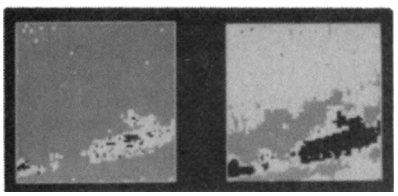


Fig. 4. Segmentation of signature. (a) Initial picture. (b) Gradient picture. (c), (f) Two different initial segmentations. (d), (g) Intermediate segmentations corresponding to (c) and (f). (e), (h) Final segmentations.



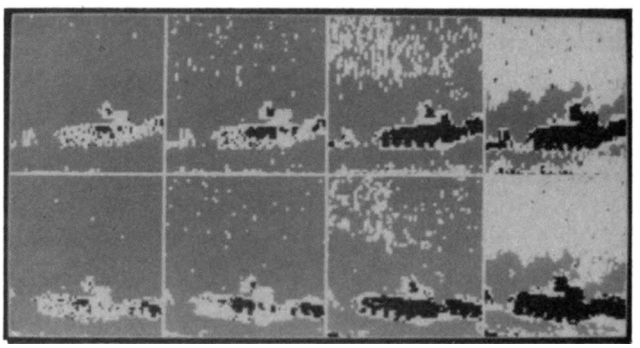
(a) (b) (c)

Fig. 5. Sequence of three frames, first frame on right.



(a) (b)

Fig. 6. (a) Initial segmentation of first frame. (b) Final segmentation of this frame after 12 iterations.



(a) (b) (c) (d)

Fig. 7. Segmentation of second (upper row) and third (lower row) frames. (a) Initial segmentation. (b) After two ordinary iterations. (c) After two iterations using second tracking scheme (computing displacement between frames). (d) After two iterations using first tracking scheme (preceeding frame as initial segmentation).

and (d) are with $\alpha = 1$ and $\beta = 0.2$. Fig. 2(c) and (e) are for $\alpha = 1$ and $\beta = 1$.

Fig. 3 shows some automobile parts. This picture is hard to segment by thresholding because of the global shading of the parts and the background. Fig. 3(b) and (d) show the segmentation results with $\alpha = 1$, $\beta = 20$. Fig. 3(c) and (e) display the results with $\alpha = 1$, $\beta = 0$.

Fig. 4 shows a signature. From top to bottom, we have the original image, the gradient, the initial segmentation, and intermediate and final results of the segmentation with $\alpha = 1$, $\beta = 1$.

In Figs. 1–4 updating was done in parallel at every iteration. This was done to demonstrate that the process improves segmentation even though the parallel updating does not guarantee reduction of cost. The process is relatively fast (faster than relaxation or steepest descent optimization), and final results were obtained after 10–20 iterations, when no more change is introduced into the labeling.

V. Object Tracking

The optimal segmentation scheme leads to simple object tracking methods by deriving the segmentation of a frame in a sequence from the final segmentation of the preceding frame. This introduces additional knowledge into the procedure, and only a few iterations are needed to reach final segmentation at each frame. Two tracking methods were tested and are described in this section.

In the first tracking scheme, the final segmentation of a frame is used as the initial segmentation for the following frame. This approach is appropriate when the movement of the objects from frame to frame is small relative to their size and is simple to implement.

The second tracking method compares the initial segmentation of a new frame to the final segmentation of the preceding frame. After computing the center of gravity for each label, a correspondence is established between labels in the two frames to minimize size and location differences. After the correspondence has been established, an estimate is available for the displacement of each object. In updating the labels of each frame, a weight is given to the labeling of the preceeding frame, shifted by the computed displacement estimate.

Fig. 5 shows a sequence of three frames. The initial and final segmentation of the first frame, after 12 iterations of ordinary optimization, are displayed in Fig. 6. Segmentation of the second and third frames with two iterations only, for different methods, is displayed in Fig. 7. The two tracking algorithms show much improved results compared with ordinary optimization.

VI. Concluding Remarks

An optimization method has been presented that can perform classification based on optimization of a cost function. Experiments were performed for image segmentation with cost functions that include roughness, discrepancy from original image, and fitting the gray level gradient. The general method is not limited to this class of functions only; any function appropriate to the problem can be used. This method can also be used for simple tracking algorithms in sequences of frames. It has also been applied to solve substitution ciphers [11], resulting in fast and accurate solutions.

References

[1] A. Rosenfeld, R. A. Hummel, and S. W. Zucker, "Scene labeling by relaxation operations," *IEEE Trans. Systems, Man, Cybern.*, vol. SMC-6, pp. 420–433, June 1976.

[2] S. Peleg, "A new probabilistic relaxation scheme," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-2, pp. 362–369, July 1980.

[3] ——, "Monitoring relaxation algorithms using labeling evaluation," in *Proc. 5th Int. Conf. Pattern Recognition*, 1980, pp. 54–57.

[4] O. D. Faugeras and M. Berthod, "Improving consistency and reducing ambiguity in stochastic labeling: An optimization approach," *IEEE Trans. Pattern Anal. Machine Intell.* vol. PAMI-3, pp. 412–424, July 1981.

[5] R. Hummel and S. Zucker, "On the foundations of relaxation labeling processes," Computer Vision and Graphics Lab., McGill Univ., TR-80-7, July 1980.

[6] S. Peleg and A. Rosenfeld, "Breaking substitution ciphers using a relaxation algorithm," *Commun. ACM*, vol. 20, pp. 598–605, Nov. 1979.

[7] A. Martelli and V. Montanari, "Optimal smoothing in picture processing: an application to fingerprints," in *Proc. IFIP Congr.*, 1971, Booklet TA-2, pp. 86–90.

[8] J. S. Weszka and A. Rosenfeld, "Threshold evaluation techniques," *IEEE Trans. Systems, Man, Cybern.*, vol. SMC-8, pp. 622–629, 1978.

[9] K. A. Narayanan, D. P. O'Leary, and A. Rosenfeld, "Image smoothing and segmentation by cost minimization," Computer Vision Lab., Computer Science Center, Univ. of Maryland, College Park, TR-1049, May 1981.

[10] R. M. Haralick, "Edge and region analysis for digital image data," *Comput. Graphics Image Proc.*, vol. 12, pp. 60–73, 1980.

[11] R. Kirby, Ph.D. dissertation, Univ. of Maryland, College Park, 1983.

Information Structures in Deterministic Decentralized Control Problems

R. H. LI AND MADAN G. SINGH

***Abstract*—The information structures associated with deterministic decentralized control are examined. Conditions are derived under which such structures are partially nested. The tuning regulator approach and the sequential optimization approach are used as examples to examine these conditions in the context of concrete decentralized controller design philosophies.**

I. Introduction

Since the publication of Witsenhausen's paper in 1968 [22] major interest has arisen in the information structures associated with decentralized decisionmaking under uncertainty (cf. [1]–[13]). Although the optimal solution to Witsenhausen's problem has not so far been found, much insight has been gained. For example, although in the general case of nonclassical information patterns, we know that separation results do not apply so that the optimal solution may be nonlinear and based on an infinite dimensional record of the measurements, certain special cases exist where simple solutions do apply. The most significant of these special cases is the one where the information structure is partially nested [3], since here linear solutions are possible.

In parallel with the developments on stochastic decentralized control, a significant amount of work has been done on decentralized deterministic problems [14]–[21], [23]. Here, conditions have been derived for stabilization for decentralized control [14], and design techniques have been developed [15]–[20], [24]. As Bismut [2] points out, the deterministic decentralized control problem can be viewed as a degenerate case for the more general stochastic formulation. However, aside from the paper of Bismust [2], no attempt has been made to establish the links, if any, between the two fields. The aim of the present correspondence is to examine the links.

One useful way of examining the links between the stochastic and the deterministic decentralized control problems is through the information structures. We examine some particular approaches in deterministic decentralized control and see under what conditions we can have a partially nested information structure. It should be noted that most of the work on deterministic decentralized controller design has used a stabilization as opposed to an optimization framework. However, since many stable decentralized controls may be possible [24], designers often also use some optimization criteria (e.g., speed of response, etc.). In order to do out comparison, we will consider deterministic decentralized control from an optimization point of view.

The rest of the correspondence is divided into three parts. In Section II we review briefly the notion of information structures in decentralized stochastic control. In Section III, we consider a deterministic decentralized control approach and examine its information structure. We state the conditions under which its information structure is partially nested. In Section IV we conclude by examining the tuning regulators approach of Davison as well as the sequential optimization approach of Davison and Gesing [20] from the point of view of information structures.

Manuscript received May 5, 1981; revised April 26, 1983.

R. H. Li is with the Control Systems Center, UMIST, Manchester, England and Sian Jaitong University, People's Republic of China.

M. G. Singh is with the Control Systems Center, UMIST, Manchester, England.

0018-9472/83/1100-1162$01.00 ©1983 IEEE