Proceedings of the Fourth Intl. Joint Conference on Pattern Recognition, Kyoto, Japan, Nov. 7-10, 1978

THE CONSISTENT LABELING PROBLEM AND SOME APPLICATIONS TO SCENE ANALYSIS

Linda G. Shapiro
Department of Computer Science
Kansas State University
Manhattan, Kansas 66506

Robert M. Haralick
Department of Electrical Engineering
Department of Computer Science
University of Kansas
Lawrence, Kansas 66045

I. INTRODUCTION

In this paper we formulate a general network constraint analysis problem which we call the labeling problem. The labeling problem is a generalization of specific problems from each of several different specialty areas. Some of these specific problems include the subgraph isomorphism problem [20], the graph homomorphism problem and the graph coloring problem [13], the relational homomorphism problem [8], the packing problem [3], the scene labeling problem [1], the shape matching problem [2], the Latin square puzzle [22], constraint satisfaction problems [4], and theorem proving [14]. The generalized problem involves a set of units which usually represent a set of objects to be given names, a set of labels which are the possible names for the units, and a compatibility model containing ordered groups of units which mutually constrain one another and ordered groups of unit-label pairs which are compatible. The compatibility model is sometimes called a world model. The problem is to find a label for each unit such that the resulting set of unit-label pairs is consistent with the constraints of the world model.

Before we can fully state the labeling problem, we need some additional concepts and definitions. Let $U = \{1,\ldots,M\}$ be a set of M units and let L be a set of labels. If $u_1,\ldots,u_N \in U$, a labeling of units $(u_1,\ldots,u_N)$ is a function $f:\{u_1,\ldots,u_N\} \rightarrow L$ that assigns a label in L to each of the units $u_1,\ldots,u_N$. The labeling problem is to use the world model to find a particular kind of labeling called a consistent labeling for all M units in U.

The problem of labeling is that not all of the labelings are consistent because some of the units are apriori known to mutually constrain one another. If an N-tuple of units $(u_1,\ldots,u_N)$ are known to mutually constrain one another, then not all labelings are permitted or legal for units $(u_1,\ldots,u_N)$. The compatibility model tells us which units mutually constrain one another N at a time and which labelings are permitted or legal for those units which do constrain one another. One way of representing this compatibility model is by a quadruple (U,L,T,R) where $T \subseteq U^N$ is the set of all N-tuples of units which mutually constrain one another and the constraint relation $R \subseteq (U \times L)^N$ is the set of all 2N-tuples $(u_1,\ell_1,\ldots,u_N,\ell_N)$ where $(\ell_1,\ldots,\ell_N)$ is a permitted or legal N-tuple of labels for the N-tuple of units $(u_1,\ldots,u_N)$. We call T the unit constraint relation and R the unit-label constraint relation.

A labeling $f:\{u_1,\ldots,u_p\} \rightarrow L$ is a consistent labeling of units $(u_1,\ldots,u_p)$ with respect to the compatibility model (U,L,T,R) if and only if $\{i_1,\ldots,i_N\} \subseteq \{1,\ldots,p\}$ and $(u_{i_1},\ldots,u_{i_N}) \in T$ imply the 2N-tuple $(u_{i_1},f(u_{i_1}),\ldots,u_{i_N},f(u_{i_N})) \in R$. When U and L are understood, such a labeling f is called a (T,R)-consistent labeling of $(u_1,\ldots,u_p)$. The consistent labeling problem is to find all consistent labelings of units $(1,\ldots,M)$ with respect to the compatibility model.

In this paper we give examples of the consistent labeling problem in scene analysis and define a general look-ahead operator ϕ_{KP} which aids in solving labeling problems by reducing the unit-label constraint relation R.

II. SOME EXAMPLES OF LABELING PROBLEMS IN SCENE ANALYSIS

II.1 Scene Labeling

The scene labeling problem arises in the context where a picture is taken of a scene (such as an office) that has objects (like chairs, tables, desks, file cabinets, and so on) which need to be identified. A low level computer vision system analyzes the picture, segments it and perhaps even assigns one or more labels to some of the objects in the picture. Given the world model information which describes allowable spatial relationships among pieces of office furniture, and given the spatial relationships that exist among segments in the image, the scene labeling problem is to use the world model to find labels for each segment in the picture. This involves labeling those segments which have not been given labels and reducing the labeling ambiguities for those segments which have been given tentative labels by the low level vision system.

Let $U = \{u_1,\ldots,u_M\}$ be the set of segments in the office image. By a spatial analysis of the image, we can produce a list of N-ary relationships that hold among the segments of U. Each item of the list can be expressed as a predicate and N possible segments for which the predicate holds. For example, with N = 2, the spatial analysis might discover that segment u_1 is above segment u_3, segment u_2 is on segment u_1, and segment u_5 is behind segment u_6. We write these kinds of relations in the shorthand form: ABOVE(u_1,u_3); ON(u_2,u_1); BEHIND(u_5,u_6).

The world model constraint is also a list of N-ary relationships. Each item of the list consists of a predicate and N possible object names for which the predicate holds. For example, with N = 2, the following constraints between object names may hold: pictures can be above chairs, books can be on desks, and chairs can be behind desks. In the shorthand form we write: ABOVE(PICTURE,CHAIR); ON(BOOK,DESK); BEHIND(CHAIR,DESK).

To put the scene labeling problem into the format of the general labeling problem, we must define the relation T of segments which mutually constrain one another and the relation R of constraints between segments and labels. We can construct the relations T and R as follows. Let P be a predicate. If for some segments $u_1,...,u_N$, a spatial analysis of the image shows $P(u_1,...,u_N)$ is true and if for some labels $\ell_1,...,\ell_N$ the world model constraint permits $P(\ell_1,...,\ell_N)$ to be true, and the low level vision analysis does not prohibit label ℓ_n for segment u_n, $n = 1,...,N$, then the N-tuple $(u_1,...,u_N)$ is a member of T and the 2N-tuple $(u_1,\ell_1,...,u_N,\ell_N)$ is a member of R. For example, if one of the labels that the low level vision system allows for segment u_1 is PICTURE and one of the labels it allows for segment u_3 is CHAIR and if P is the predicate ABOVE, since ABOVE(PICTURE,CHAIR) is true and ABOVE(u_1,u_3) is true, then T contains (u_1,u_3) and R contains $(u_1$,PICTURE,u_3,CHAIR). Each consistent labeling based on T and R is a possible labeling of the scene; and since consistent labelings are subsets of possible labelings, the number of labeling ambiguities will be reduced.

II.2 The Edge Orientation Problem

There are a variety of approaches to finding edges in a picture [18]. Most of them begin with the application of some local operator to determine the strength of an edge passing through each resolution cell in a particular direction. The problem with these local operators is that they tend to be noisy; their variance is high. Since most meaningful edges in real world images tend to be highly continuous with little curvature, it should be possible to combine the prior knowledge low curvature condition and the local gradient operator values to produce cleaner edges.

We define the orientation of an edge to be an angle between 0° and 360°. The edge lies along a line in the given angular direction and a person traveling along the edge in the angular direction of the edge will always find the darker side of the edge to his right. Low curvature edges mean that the maximum angle by which any small edge segment can bend with respect to its predecessor edge or successor edge segment is limited to some maximum bending angle which we call Δ (for example, 60°). With this kind of prior knowledge, we can formulate the edge orientation problem as a labeling problem.

Let $U = \{(i,j) \mid i = 1,...,I \text{ and } j = 1,...,J\}$ be the set of resolution cells of an I-row by J-column image. Let L be a set consisting of possible edge orientations (including the possibility of no edge). For example, L could be {none,0,45,90,135,225,270,315}. For each resolution cell (i,j) let $E(i,j) \subseteq L$ be the set of its possible edge orientations computed on the basis of the strength of some local edge operator. Let the neighborhood of resolution cell (i,j) be N(i,j). N(i,j) could be a 4-neighborhood, an 8-neighborhood, or perhaps something more complex. Only edge orientations of resolution cells in the neighborhood of a given resolution cell can constrain the edge orientation of the given resolution cell. On this basis we can define the compatibility model by (U,L,T,R) where

$$T = \{((i,j),(i',j')) \mid (i',j') \in N(i,j)\}$$

and

$$R = \{((i,j),\ell,(i',j'),\ell') \mid (i',j') \in N(i,j), \ell \in E(i,j), \ell' \in E(i',j') \text{ and } \ell,\ell' \neq \text{none implies } |\ell-\ell'| < \Delta\}$$

III. RELATED LITERATURE

One general technique that can be used to solve a labeling problem is a depth-first search. The search procedure fixes labels to units as long as it can find a label for each new unit that is compatible according to the constraint relation R with the labels already fixed to previous units. Whenever the procedure cannot find a label for a new unit, it backtracks to the previous unit and tries to find a different label for that unit. If the procedure finds a label for all M units, it has found a consistent labeling. If the procedure backs up all the way to the first unit without finding any consistent labelings and there are no more possible labels for the first unit, the procedure fails and there exist no consistent labelings.

The depth-first search procedure suffers from thrashing. A poor choice of labels for one of the first units can cause failure of all paths stemming from that choice. To make the depth-first search more efficient, we must eliminate those paths which terminate because they are not contained in any consistent labeling. To do an efficient search, we must first remove from R all N-tuples of unit-label pairs which do not participate in a consistent labeling. Montanari [16] showed that this problem itself is NP-complete. However, compatibility relations can be graded with respect to the difficulty of removing these N-tuples. The initial work by Waltz [22] on line labeling indicated that although the compatibility relation he employed was not minimal, the amount of work to make it minimal using a sequentially implementable look-ahead operator was small. Gaschnig [6] reported similar results. Hence, it may be that the worst case problems do not seem to arise very frequently in practice and look-ahead operators of low order complexity can be of great help. Other related work includes that of Rosenfeld [19], Rosenfeld et. al. [17], Ullman [20], Haralick and Kartus [8], Haralick [9,10], Mackworth [15], Zucker et. al [24], Vanderbrug [21], Hanson and Riseman [7], Barrow and Tennenbaum [13], Davis [2], Zucker and Hummel [25], Freuder [5], Haralick et. al. [11], and Haralick and Shapiro [12].

IV. FINDING (T,R)-CONSISTENT LABELINGS WITH THE HELP OF A GENERALIZED LOOK-AHEAD OPERATOR ϕ_{KP}

Let $U = \{1,...,M\}$ be a set of units, L be a set of labels, $T \subseteq U^N$, and $R \subseteq (U \times L)^N$. Let $K \leq N \leq P$ with $K < P$. The look-ahead operator ϕ_{KP} is defined by $\phi_{KP}R = \{(u_1,\ell_1,...,u_N,\ell_N) \in R \mid$ for every combination $j_1,...,j_K$ of $1,...,N$ and for every $u'_{K+1},...,u'_p \in U$, there exists $\ell'_{K+1},...,\ell'_p \in L$ such that the function f defined by $f(u_{j_i}) = \ell_{j_i}$, $i = 1,...,K$ and $f(u'_i) = \ell'_i$, $i = K+1,...,p$ is a (T,R)-consistent labeling of units $(u_{j_1},...,u_{j_K},u'_{K+1},...,u'_p)\}$.

The look-ahead operator ϕ_{KP} removes 2N-tuples from R which do not contribute to a consistent labeling. We illustrate its use with an example where N = 3, K = 2, and P = 4.

IV.1 Example

Let $U = \{1,2,3,4,5\}$, $L = \{a,b\}$, $T = \{(1,2,3), (1,2,4), (1,2,5), (2,3,4), (2,3,5), (3,4,5)\}$, N = 3, M = 5, K = 2, P = 4, and

R = {(1,a,2,a,3,a), (1,a,2,a,4,a), (1,a,2,a,5,a), (1,a,2,b,3,a), (1,a,2,b,4,b), (1,b,2,b,5,b), (2,a,3,a,4,a), (2,a,3,a,5,a), (2,b,3,a,4,b), and (3,a,4,a,5,a)}.

The results of examining three 2N-tuples (1,a,2,a,3,a), (1,a,2,b,3,a), and (1,a,2,b,5,b) are shown in Figure 1. The 2N-tuple (1,a,2,a,3,a) passed all tests and is therefore an element of $\phi_{2,4}R$. The 2N-tuple (1,a,2,b,3,a) passes its first test since with the fixed unit-label pairs (1,a) and (2,b) and free units 3,4 the function f defined by f(1) = a, f(2) = b, f(3) = a, f(4) = b is a consistent labeling of (1,2,3,4). However, with (1,a) and (2,b) still fixed and free units 3,5, there is no consistent labeling of (1,2,3,5). (There is no label x with 2N-tuple (1,a,2,b,5,x) in R). Thus (1,a,2,b,3,a) is not in $\phi_{2,4}R$. Similarly, the 2N-tuple (1,b,2,b,5,b) fails its first test and is not in $\phi_{2,4}R$.

After testing each 2N-tuple of R, we obtain
$\phi_{2,4}R$ = {(1,a,2,a,3,a), (1,a,2,a,4,a), (1,a,2,a,5,a), (2,a,3,a,4,a), (2,a,3,a,5,a), (3,a,4,a,5,a)}.

At this point every 2N-tuple of R contributes to the one consistent labeling f of (1,2,3,4,5) defined by f(i) = a, i = 1,...,5.

2N-Tuple	K = 2 Fixed unit-label pairs	P - K = 2 free units	P - K = 2 labels for the free units that contribute to a consistent labeling of all P = 4 units
(1,a,2,a,3,a)	1,a,2,a	3,4	3,a,4,a
		3,5	3,a,5,a
		4,5	4,a,5,a
	1,a,3,a	2,4	2,a,4,a
		2,5	2,a,5,a
		4,5	4,a,5,a
	2,a,3,a	1,4	1,a,4,a
		1,5	1,a,5,a
		4,5	4,a,5,a
(1,a,2,b,3,a)	1,a,2,b	3,4	3,a,4,b
		3,5	NONE
	No more combinations need be looked at for this 2N-tuple		
(1,b,2,b,5,b)	1,b,2,b	3,4	NONE
	No more combinations need be looked at for this 2N-tuple		

Figure 1 illustrates the application of the ϕ_{KP} operator to three 2N-tuples of R.

The theoretical results concerning the ϕ_{KP} operator may be found in Haralick and Shapiro [12]. We will briefly summarize the most important results here. A labeling f is a (T,R)-consistent labeling if and only if it is a $(T,\phi_{KP}R)$-consistent labeling. Thus ϕ_{KP} never removes any 2N-tuples of R that contribute to a consistent labeling. There is a minimum relation S_{TR} that has the same set of consistent labelings as R. The minimum relation S_{TR} is contained in $\phi_{KP}^m R$ for every positive integer m, and $\phi_{KP}(S_{TR}) = S_{TR}$. The operator ϕ_{KP} is more powerful (can remove more 2N-tuples) for larger values of p. Finally, for $T = U^N$, when $\phi_{K+1\ P}R = R$ ($\phi_{K+1\ P}$ cannot remove anything else from R), then $\phi_{KP}R = R$.

V. CONCLUSION

The labeling problem is an important problem in scene analysis, and the ϕ_{KP} operator is a generalized tool for helping to solve the labeling problem.

REFERENCES

1. Barrow, H.G. and J.M. Tenenbaum, MYSYS: A System for Reasoning About Scenes, SRI AI Technical Report No. 121, Stanford Research Institute, Menlo Park, California, 1976.

2. Davis, L.S., Shape Matching Using Relaxation Techniques, TR-480, Computer Science Center, University of Maryland, September 1976.

3. Deutsch, J.P.A., "A Short Cut for Certain Combinatorial Problems," British Joint Computer Conference, 1966.

4. Fikes, R.E., "REF-ARF: A System for Solving Problems Stated as Procedures," Artificial Intelligence, Vol. 1, 1970, pp. 27-120.

5. Freuder, E.C., Synthesizing Constraint Expressions, MIT AI Memo 370, July 1976.

6. Gaschnig, J., "A Constraint Satisfaction Method for Conference Making," 12th Annual Allerton Conference on Circuit and System Theory, University of Illinois, 1974.

7. Hanson, A. and E. Riseman, "Pattern-Directed Boundary Formation Via Relaxation," Machine Vision Workshop, University of Massachusetts, Amherst Massachusetts, June 1-3, 1977.

8. Haralick, R.M. and J. Kartus, "Arrangements, Homomorphsism, and Discrete Relaxation," to appear in IEEE Transactions on Systems, Man, and Cybernetics, 1978.

9. Haralick, R.M., "Scene Analysis, Homomorphisms, and Arrangements," Machine Vision, Academic Press, 1978.

10. Haralick, R.M., "The Characterization of Binary Relation Homomorphisms," International Journal General Systems, Vol. 4, 1978, pp. 113-121.

11. Haralick, R.M., L. Davis, A. Rosenfeld, and D. Milgram, Reduction Operations for Constraint Satisfaction, TR-560, Computer Science Center, University of Maryland, August 1977.

12. Haralick, R.M. and L.G. Shapiro, The Consistent Labeling Problem, University of Kansas Center for Research, Inc., Remote Sensing Laboratory, January 1978.

13. Harary, F., Graph Theory, Addison-Wesley Publishing Company, Massachusetts, 1969.

14. Kowalski, R., "A Proof Procedure Using Connection Graphs," Journal of the Association for Computing Machinery, Vol. 22, No. 4, October 1975, pp. 572-595.

15. Mackworth, A., "Consistency in Networks of Relations," Artificial Intelligence 8, 1977, pp. 99-118.

16. Montanari, U., "Networks of Constraints: Fundamental Properties and Applications to Picture Processing," Information Science 7, 1974, pp. 95-132.

17. Rosenfeld, A., R.A. Hummel, and S.W. Zucker, "Scene Labeling by Relaxation Operations," IEEE

Transactions on Systems, Man, and Cybernetics Vol. SMC-6, No. 6, June 1976, pp. 420-433.

18. Rosenfeld, A. and A. Kak, Digital Picture Processing, Academic Press, New York, 1976.

19. Rosenfeld, A., "Networks of Automata: Some Applications," IEEE Transactions on Systems, Man, and Cybernetics 5, 1975, pp. 380-383.

20. Ullman, J.R., "An Algorithm for Subgraph Homomorphisms," Journal of the Association for Computing Machinery, Vol. 23, No. 1, January 1976, pp. 31-42.

21. Vanderbrug, G., "Experiments in Iterative Enhancement of Linear Features," LARS Symposium Proceedings of Machine Processing of Remote Sensed Data, Purdue University, Lafayette, Indiana, June 29-July 1, 1976.

22. Waltz, D.L., Generating Semantic Descriptions from Drawings of Scenes with Shadows, MIT Technical Report A1271, November 1972.

23. Whitehead, E.G., Jr., Combinatorial Algorithms, Courant Institute of Mathematical Sciences, New York University, 1972.

24. Zucker, S.W., R.A. Hummel, and A. Rosenfeld, "An Application of Relaxation Labeling to Line and Curve Enhancement," IEEE Transactions on Computers, Vol. C-26, No. 4, April 1977, pp. 394-403.

25. Zucker, S.W. and R.A. Hummel, "Computing the Shape of Dot Clusters, I: Labeling Edge, Interior, and Noise Points," Technical Report TR-543, University of Maryland, College Park, Maryland, May 1977.