A STRUCTURAL METHOD OP SCENE ANALYSIS

Zdenek Zdrahal

Czech Technical University, Faculty of E.E..,,
Karlovo nam.13, 121 35 Prague 2,
Czechoslovakia

In this paper a method of scene analysis and pattern recognition is outlined. Relational structures are employed as a tool for describing scones and objects. Descriptive primitives are extracted from the TV image together with their interrelations and the recognition is defined as a task of matching this structural description against models. The method completely reported in [3] , has been developed for the vision subsystem of the GOALEM robot [2]. For comparison see [l] where another structural techniques are reviewed.

The problem can be split into two parts :

(A) The image of the scene to be recognized is converted into a relational structure called image structure IS. A model is stored as a model structure MS. Then we construct a function f which assigns to each element of IS an element of MS and evaluate the match. Usually the absolute match cannot be found since the real IS differs from the ideal one. Basically, the match can be assessed either in terms of the number of relations being preserved or in terms of the number of relations being different. Only the former case will be discussed in this paper although also the latter one yields interesting results. If MS is a substructure of the IS i.e. a monomorphism exists from MS to IS then the object described by MS is a part of the scene. All the monomorphisms should be found and indicated.

(B) Let an ensemble of model structures be given. Having solved the subproblem (A) we have to find the MS giving the best match over the whole ensemble. Since the essence of all the problem lies in the matching algorithm our attention will be focused on part (A).

<u>Def. 1.</u> A vocabulary V is a set of ordered pairs $\langle p,n \rangle$ where p is a predicate name and n is the number of its arguments. □

<u>Def. 2.</u> A relational structure S over a given vocabulary V is an ordered pair $S = \langle A,F \rangle$ where A is a set called the carrier and F is a function which assigns to each n-argument predicate name from V a subset of A^n (n-ary relation). Symbol $\langle a_{k_1},\dots,a_{k_n} \rangle$ stands for an ordered n-tuple of elements of A. □

For the sake of transparency we will demonstrate the principles of the method considering only unary relations (properties) and binary ones. Their predicate names will be p_i and r_j, respectively, $i,j = 1,2,\dots$. In order to quantify how the function f preserves the relations we define the following criterion :

<u>Def. 3.</u> Let $MS = \langle A,F \rangle$, $IS = \langle B,G \rangle$ be relational structures over a vocabulary V where $A = \{a_1,\dots,a_N\}$, $B = \{b_1,\dots,b_N\}$ are carriers,
let $f : A \rightarrow B$ be a bijective function,
let $w(p_i) > 0$, $w(r_j) > 0$ be weights of properties and binary relations, respectively.
Let us denote :

(i) $c_1(a_s,p_i,f) = 1$ if $a_s \in F(p_i)$ and $b_t = f(a_s) \in G(p_i)$,
$= 0$ otherwise

(ii) $c_2(a_s,r_j,f)$ ... the number of pairs $\langle a_{k_1}, a_{k_2} \rangle$ which satisfy conditions (iii),(iv), and either (v),(vi),(vii) or (viii),(ix),(x).

(iii) $\langle a_{k_1},a_{k_2} \rangle \in F(r_j)$

(iv) $\langle b_{l_1},b_{l_2} \rangle \in G(r_j)$

(v) $a_{k_1} = a_s$

(vi) $b_{l_1} = b_t = f(a_s)$

(vii) $b_{l_2} = f(a_{k_2})$

(viii) $a_{k_2} = a_s$

(ix) $b_{l_2} = b_t = f(a_s)$

(x) $b_{l_1} = f(a_{k_1})$

Criterion C is defined as :

$$C(MS,IS,f) = \sum_{s=1}^{N} \Big(\sum_j c_2(a_s,r_j,f).w(r_j) + \sum_i c_1(a_s,p_i,f).w(p_i) \Big).$$

□

We assume both carriers have the same number of elements. If this is not the case we add elements to the smaller carrier which do not participate in any relation. The optimal function f^* is to be found which maximizes the criterion C i.e. $C(MS,IS,f^*) = \max_f C(MS,IS,f)$. The value of $C(MS,IS,f^*)$ is used for solving the subproblem (B).

There are N! various functions $f : A \rightarrow B$. We cannot test all of them because of the combinatorial increase of time. It would be convenient if only the functions promising high values of the criterion C are checked. We will estimate the chances of each function in terms of an auxiliary criterion $\bar{C}$ whose evaluation is much simpler.

__Def. 4.__ Let the assumptions of Def. 3. hold with the exception of the conditions (vii),(x) which are omitted. The auxiliary criterion $\bar{C}$ is defined as :

$$\bar{C}(MS,IS,f) = \sum_s \Big(\sum_j \bar{c}_2(a_s,r_j,f).w(r_j) + \sum_i \bar{c}_1(a_s,p_i,f).w(p_i) \Big) = \sum_s x_{st}$$

where $\bar{c}_1$, $\bar{c}_2$ differs from the original c_1, c_2 due to the conditions (vii),(x). □

The expression $x_{st} = \sum \bar{c}_2(a_s,r_j,f).w(r_j) + \sum \bar{c}_1(a_s,p_i,f).w(p_i)$ depends only on the partial assignment $b_t = f(a_s)$ and thus can be calculated regardless of the rest of the function. Let us arrange the numbers x_{st} into a square matrix X. We calculate the value of $\bar{C}$ as a sum of elements x_{st} taking just one element from each column and row according to the bijective function f. Each x_{st} evaluates the local match of elements a_s and b_t without considering the rest of both structures and expresses the highest possible contribution of the partial assignment $b_t = f(a_s)$ to the main criterion C. Hence the following theorem can be proven :

__Theorem 1.__ For any given f the inequality $\bar{C}(MS,IS,f) \geq C(MS,IS,f)$ holds. □

The algorithm for matching the structures is based on this theorem.

__An outline of the algorithm__. Let us arrange the functions $f_1, f_2, \dots, f_{N!}$ in accordance with the decreasing values of $\bar{C}$ i.e. if $i > j$ then $\bar{C}(MS,IS,f_i) \leq \bar{C}(MS,IS,f_j)$. Since the $\bar{C}$ is calculated from the matrix X this task is a problem of linear programming (assignment problem) and, therefore, efficient methods are available.

In the i-th step of our algorithm $C(MS,IS,f_i)$ is evaluated. A variable q_i is defined as : $q_1 = C(MS,IS,f_1)$,

$q_i = \max \{ C(MS,IS,f_i), q_{i-1} \}$, $i \neq 1$.

From Theorem 1 follows :

After the i-th step all the functions for which $\bar{C}(MS,IS,f_j) < q_i$ are eliminated since they cannot become the optimal function f^* . It follows immediately from the inequality $C(MS,IS,f_j) \leq \bar{C}(MS,IS,f_j) < q_i$. The matching process is finished if $\bar{C}(MS,IS,f_{i+1}) < q_i$.

Obviously, we do not generate the complete decreasing sequence of $\bar{C}$ since we need only the next function in each step.

__Example__. Structures IS = ⟨A,F⟩ , MS = ⟨B,G⟩ are given in Fig.1.

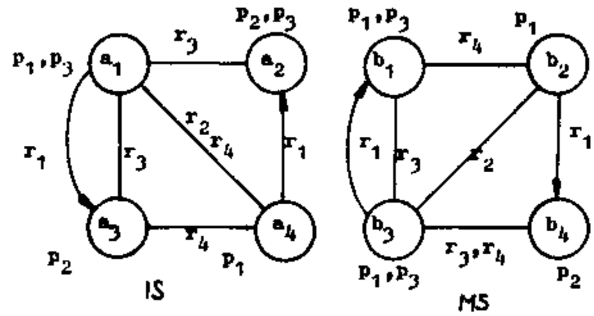

Fig. 1.

The vocabulary V consists of properties p_1,p_2,p_3 and binary relations r_1,r_2,r_3,r_4. Weights $w(p_1) = w(p_2) = w(p_3) = w(r_1) = 1$, $w(r_2) = w(r_3) = w(r_4) = 0.5$ are given. The matrix X is calculated,

$$X = \begin{Vmatrix} 4 & 4 & 7 & 2 \\ 3 & 0 & 2 & 3 \\ 3 & 1 & 2 & 4 \\ 2 & 4 & 4 & 1 \end{Vmatrix}$$

The evaluation of x_{21} is shown in Fig.2.

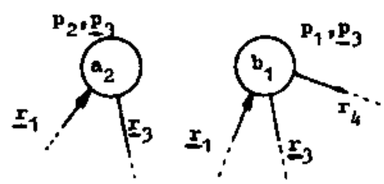

Fig. 2.

The underlined relations contribute to the value x_{21}.

Step 1 :

$\bar{C}_{max} = \bar{C}(MS,IS,f_1) = 18$ for $f_1 = \{\langle a_1,b_3\rangle, \langle a_2,b_1\rangle, \langle a_3,b_4\rangle, \langle a_4,b_2\rangle\}$
$q_1 = C(MS,IS,f_1) = 11$
For $f_2 = \{\langle a_1,b_3\rangle, \langle a_2,b_4\rangle, \langle a_3,b_1\rangle, \langle a_4,b_2\rangle\}$
$\bar{C}(MS,IS,f_2) = 17.$
Since $\bar{C}(MS,IS,f_2) > q_1$ the matching process continues.

Step 2 :

$C(MS,IS,f_2) = 16$, $q_2 = 16$, $\bar{C}(MS,IS,f_3) < q_2$, the optimal function $f^* = f_2$ has been found and thus the matching process is finished. For demonstration all the values of $\bar{C}$ and the evaluated ones of C are shown in Fig.3.

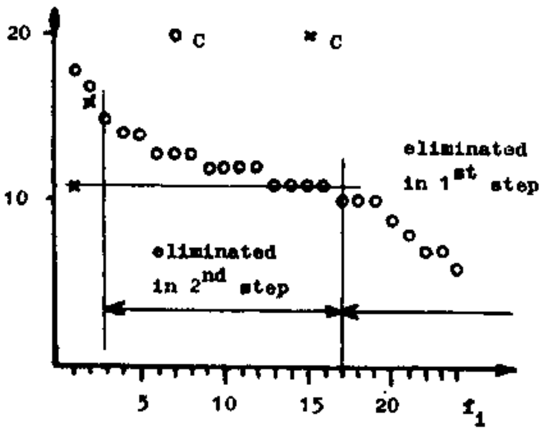

Fig. 3.

In our implementation of the algorithm improvements speeding up the convergence are included. Moreover, monomorphisms can be found by means of the following theorem :

<u>Theorem 2</u>. Let $MS = \langle B,G\rangle$, $IS = \langle A,F\rangle$, $A = \{a_1,\dots\dots,a_M\}$, $B = \{b_1,\dots\dots,b_N\}$, $N \leq M$, be given, let f be an injective function $f : A \to B$. Let us define the identical function $g : B \to B$, i.e. $g(b_j) = b_j$ for each b_j. If $\bar{C}(MS,MS,g) = C(MS,IS,f)$ then f is a monomorphism from MS to IS. □

It would be very difficult to compare our results with those of [1] because of the differencies in computers, programming languages etc. Just for brief Information, our algorithm was run on an ICL 2903 and took tens of seconds for structures with as many as 20 elements. We expeot we can slightly speed it up by improving the linear programming subroutine which is being prepared*

REFERENCES

[I] Barrow, H. G., Ambler, A. P., and Duretall, R. M. " Some Techniques for Recognizing Structures in Pictures", in <u>Frontiers of Pattern Recognition</u>. Academic Press, New York and London, 1972

[2] Kirchmann, B., Kopecky, P., and Zdrahal. Z. "GOALEM from Prague", in Proc,, IJCAI-77, Cambridge, MA, August, 1977, 771

[3] Zdrahal, Z. "Structural Methods in Pattern Recognition", CSc* (PhD.) thesis, Czech Technical University, Prague, 1979 (in Czech)