COLLOQUE NATIONAL SUR LE TRAITEMENT DU SIGNAL ET SES APPLICATIONS

NICE du 26 au 30 AVRIL 1977

DESCRIPTION DE GRAPHISME EN VUE D'UNE RECONNAISSANCE STRUCTURELLE

S. CASTAN - G. PERENNOU

Laboratoire C.E.R.F.I.A. - I.U.T. Informatique - U. P. S. Avenue de Rangueil - 31077 TOULOUSE CEDEX

RESUME

Des caractéristiques topologiques telles que existence de boucles fermées, ouvertes, position relative, orientation et points extréma (en divers sens) de celle-ci jouant un rôle souvent primordial dans la reconnaissance structurelle de graphisme et plus généralement de formes visuelles.

L'objet de l'article est de décrire des algorithmes rapides de type local séquentiel, permettant la mise en évidence des boucles fermées, des boucles ouvertes, de leur orientation, de différents extréma.

Ils fournissent aussi certains paramètres de nature métrique, (longueur, hauteur, surface, positions relatives des boucles, etc...).

Ces différentes caractéristiques sont ensuite réunies dans un graphe relationnel traduisant leurs positions relatives et leurs incidences.

On applique ceci au cas des lettres manuscrites.

SUMMARY

Description of a graphism in view of a structural recognition.

Topological characteristics such as the existence of closed or open loops, relative position and extrema points (in various directions) of the latter often play a primordial role in the structural recognition of graphism and more generally of visual shapes.

The object of this article is to describe rapid flowcharts of local sequential type which reveal closed loops, open loops, their orientation and different extrema.

They also provide certain metric parameters (length, height, relative positions od the loops, etc...).

Those different characteristics are then put together in a relational graph showing their relative position and their incidences.

This is applied to handwritten letters.

DESCRIPTION DE GRAPHISME EN VUE D'UNE RECONNAISSANCE STRUCTURELLE

S. CASTAN - G. PERENNOU

1. INTRODUCTION

En traitement d'images les méthodes de reconnaissance structurelle (1), (2), (3), (4) sont particulièrement indiquées lorsqu'on veut reconnaître des formes d'après des caractéristiques de type topologique. (Nous utiliserons le terme pseudo-topologique pour qualifier des propriétés qui ne dépendent pas de métriques).

Il en est ainsi des boucles fermées, ouvertes, munies de leurs orientations et de leurs situations relatives. Les branches du graphismes, les points de croisement, les points extréma sont également intéressants.

L'objet de cet article est de donner des algorithmes de type traitement local séquentiel permettant la mise en évidence de ces différents traits caractéristiques. On montre ensuite comment utiliser les résultats pour constituer des graphes relationnels résumant l'essentiel de la forme.

2. ALGORITHMES

Nous nous intéresserons au cas des images binaires. Soit a, b, c, trois n x m matrices sur IN, la première contient la forme, c'est-à-dire :

$$a(ij) = \begin{cases} 1 \text{ sur le trait,} \\ 0 \text{ ailleurs} \end{cases}$$

Pour simplifier la présentation des algorithmes on supposera que la matrice "a" est bordée de zéros (première et dernière ligne à zéro, et de même pour les colonnes).

2.1. Mise en évidence, étiquetage et caractérisation des boucles

On voit sur la figure 1.a, les trois zones que l'on se propose d'étiqueter, la zone zéro est le fond, la zone 1 est ouverte à droite ((ie) connexe avec zéro sur sa droite), la zone 2 est fermée et située au dessous de 1.

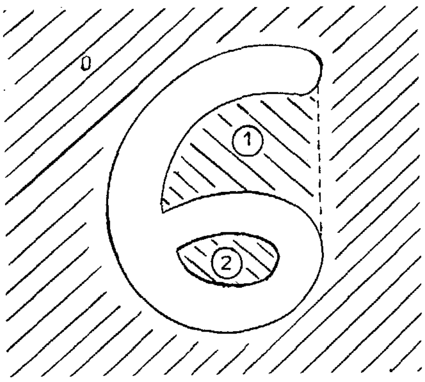


Figure 1

2.1.1. Algorithme A1

Conditions initiales : b ← 0 (mettre tous les termes de la matrice b à zéro).

CB ← 0 (le compteur de boucles est mis à zéro). Pour les colonnes j = 2,3,...,m-1 faire les traitements B1 puis B2.

Traitement B1

Pour toute ligne i = 2,3,...,n-1, faire le traitement L1 et passer à la ligne i+1 suivante si a(i,j) = 0, sinon passer à la ligne i+1 suivante.

Traitement L1

Distinguons les trois cas suivants :

cas 1 : a(i-1,j) = 0 et b(i-1,j) = 0

cas 2 : a(i,j-1) = 0 et b(i,j-1) > 0

cas 3 : a(i-1,j) = 0 et b(i-1,j) > 0

Le traitement L1 se formule comme suit.

Si cas 1 faire b(i,j) ← 0, et faire le traitement CNX, et fin de L1

Si cas 2 faire b(i,j) ← b(i,j-1) et fin de L1

Si cas 3 faire b(i,j) ← b(i-1,j) et fin de L1

Dans les autres cas faire : b(i,j) = - 1.

Traitement B2

Pour toute ligne i = n-1, n-2,...,2

Si b(i,j) = - 1 faire traitement L2 et passer à la ligne i-1 suivante, sinon passer à la ligne i-1 suivante.

Traitement L2

Si a(i-1,j) = 0 faire b(i,j) ← b(i-1,j) et fin de L2

sinon CB ← CB+1 et b(i,j) ← CB

et faire le traitement CNX.

Traitement CNX

(Il a pour but de noter les connexités entre régions).

Si a(i,j-1) = 0 faire ADJ (b(i,j-1), b(i,j)) = 1

La figure 2 donne un exemple de l'algorithme.

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	1	1	1	1	1	1	0	0
0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	1	0	0
0	0	0	1	1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

2-a : Forme initiale

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	-1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	-1	0	0	0	0	0	0
0	0	0	-1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	-1	0	0	0	0	0	0
0	0	0	-1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

2-b matrice b après traitement L_1 en colonne j=4

Figure 2 : Algorithme A_1 d'un chiffre 6 - La matrice ADJ obtenue par le traitement CNX est donnée en 4-d.

ADJ(2,0)←1

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	2	2	0	0	0	0
0	0	0	2	2	2	0	0	0	0
0	0	2	2	2	2	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	0	0	0	0
0	0	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

2-c : résultat final

ADJ	0	1	2
0	1	0	1
1	0	1	0
2	0	0	1

2-d : On observe que ADJ(2,0)=1 ce qui exprime que 2 est ouverte à droite.

2.2. Détermination et marquage d'un graphisme

On voit sur la figure 3 les trois branches délimitant les régions précédentes.

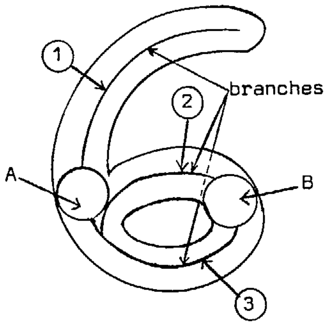


Figure 3

On veut marquer que les branches 2 et 3 sont issues d'une même zone et se rejoignent. On veut aussi pouvoir indiquer que 1 est au dessus de 2 et 2 au dessus de 3.

2.2.1. Algorithme A2

Conditions initiales

$C \leftarrow 0$ (mettre tous les termes de la matrice à zéro)

$CG \leftarrow 0$ (compteur de branches)

Pour toutes les colonnes $j = 2,3,\dots, m-1$ faire successivement les traitements G1 puis G2.

Traitement G1

$BLT \leftarrow 0$

Pour toutes les lignes $i = 2,3,\dots,n-1$

Si $a(i,j) = 1$ faire le traitement M1 et passer à la ligne i+1 suivante.

Si $a(i-1,j) = 1$ et $c(i-1,j) > 0$ faire $BLT \leftarrow C(i-1,j)$ et passer à la ligne i+1 suivante.

Traitement M1

On forme l'ensemble $\{C(i-1,j),\ C(i-1,j-1),\ C(i,j-1),\ C(i+1,j-1)\} = V(i,j)$.

Soit W une liste de longueur au plus égale à quatre des nombres distincts positifs mod 100 se trouvant dans le voisinage V(i,j).

$$W = (W_k) \qquad k = 1,\dots,4.$$

Distinguons les cas suivants :

cas 1 : $c(i-1,j) = -200$

cas 2 : $k = 1$ et $W_1 < 99$ et différent de BLT

cas 3 : $k \geq 2$

cas 4 : $c(i-1,j) = -100$

cas 5 : W est vide.

Le traitement M1 se formule comme suit :

Si cas 1 faire $C(i,j) \leftarrow -200$ et fin de M1

Si cas 2 faire $C(i,j) \leftarrow W$ et fin de M1

Si cas 3 faire $C(i,j) \leftarrow -200$ et fin de M1

Si cas 4 faire $C(i,j) \leftarrow -100$ et fin de M1

Si cas 5 faire $C(i,j) \leftarrow -100$ et fin de M1

Si autres cas faire $C(i,j) \leftarrow -1$ et fin de M1.

Traitement G2

Pour toutes les lignes $i = n-1,\ n-2,\dots,2$.

Si $c(i,j) < 0$ faire M2 et passer à la ligne i-1 suivante,

sinon passer à la ligne i-1 suivante.

Traitement M2

Si $C(i,j) < 0$:

a) $C(i+1,j) > 0$ faire $C(i,j) \leftarrow C(i+1,j)$

b) $C(i+1,j) = 0$, faire $CG \leftarrow CG - 1$

Si $C(i,j) = -1$ faire $C(i,j) \leftarrow CG$

Si $C(i,j) = -100$ faire $C(i,j) \leftarrow 100 + CG$

Si $C(i,j) = -200$ faire $C(i,j) \leftarrow 200 + CG$

Si $C(i,j) > 0$ et $C(i+1,j) > 200$ faire $C(i,j) \leftarrow C(I+1,j)$

Dans les autres cas, passer à la ligne i-1 suivante.

La figure 4 donne un exemple de l'algorithme.

Remarque -

1 - Les algorithmes A1 et A2 exposés séparément peuvent fonctionner simultanément lors de l'examen séquentiel de la forme.

2 - L'efficacité de l'analyse est plus grande sur des formes squelettisées (5).

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	1	1	1	1	1	1	0	0
0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	1	0	0
0	0	0	1	1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

4-a : Forme initiale

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
0	0	101	0	0	0	0	0	0	0
0	0	101	3	3	3	3	3	0	0
0	0	101	0	0	0	0	3	0	0
0	0	101	0	0	0	0	204	0	0
0	0	0	2	2	2	2	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

4-b : en cours de traitement colonne j=8

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0
0	0	101	0	0	0	0	0	0	0
0	0	101	3	3	3	3	204	0	0
0	0	101	0	0	0	0	204	0	0
0	0	101	0	0	0	0	204	0	0
0	0	0	2	2	2	2	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

4-c : résultat final

Figure 4

Algorithme A2 sur le chiffre 6.

3.1 GRAPHES RELATIONNELS ASSOCIES AUX FORMES

Un graphe relationnel est un quadruplet $<X,U,R,f>$ où X est l'ensemble des sommets, U l'ensemble des arcs, R un ensemble de relations et f une application de U dans R.

Autrement dit, chaque arête du graphe est muni d'une étiquette élément de R.

Deux graphes relationnels $<X,U,R,f>$ et $<X',U',R',f'>$ seront dits homomorphes lorsque l'on peut trouver trois applications α, β et γ telles que :

$\forall$ u si x est sommet initial de u, y son sommet terminal

(i) $\beta(u)$ a pour sommets initial $\alpha(x)$ et terminal $\alpha(y)$

(ii) $\gamma f(u) = f'\beta(u)$

Ces deux graphes sont de plus isomorphes si :

(iii) α, β, γ sont des bijections.

<u>Graphe relationnel associé aux régions délimitées par les boucles</u>

X sera l'ensemble des régions données par le premier algorithme auxquelles s'ajoutent deux sommets $\boxed{D}$ et $\boxed{G}$ (droite et gauche).

R sera constitué de deux relations

r_1 : au dessus de : x r_1 y sera représentée par

$$\boxed{x} \dashrightarrow \boxed{y}$$

r_2 : adjacent de : x r_2 y sera represénté par

$$\boxed{x} \longrightarrow \boxed{y}$$

U on créera un arc à chaque fois qu'un couple de sommets vérifie une relation r_i (i=1,2) et on lui donnera l'étiquette r_i.

Deux graphismes seront dits A_1-<u>équivalents</u> lorsque (i) leurs graphes seront isomorphes

(ii) γ application identique

(iii) $\alpha(D) = D$, $\alpha(G) = G$.

Ils seront dits A_1-<u>symétrique</u> lorsque (i), (ii) et (iii) : $\alpha(D) = G$, $\alpha(G) = D$.

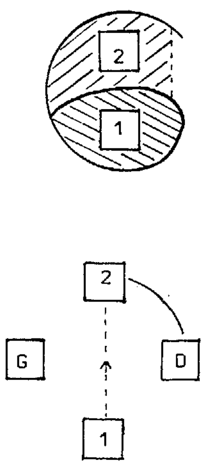


Fig. 5a

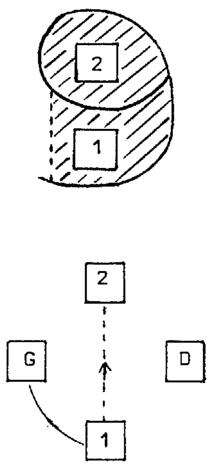


Fig. 5b

Le 6 et le 9 sont A_1-symétriques

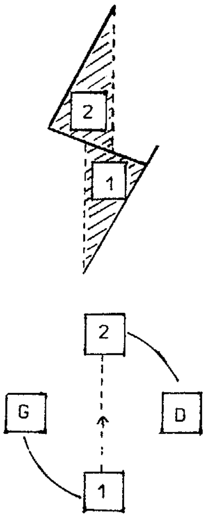


Fig. 5c

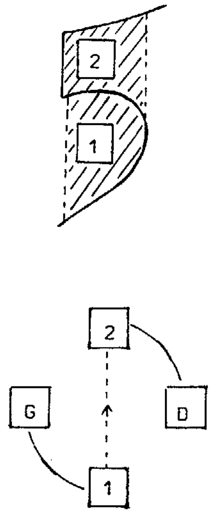


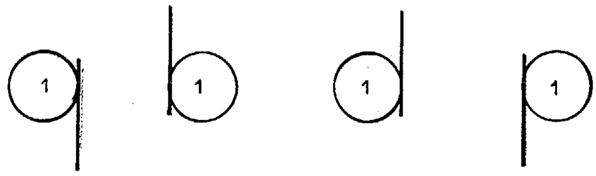
Fig. 5d

<u>Figure 5</u>

Le 4 et le 5 sont équivalents

La figure 5 illustre les principes exposés. On comprend alors que les chiffres 1, 2, 5, 6, 8, 9, 0 appartiennent en principe à des classes d'équivalence différentes.

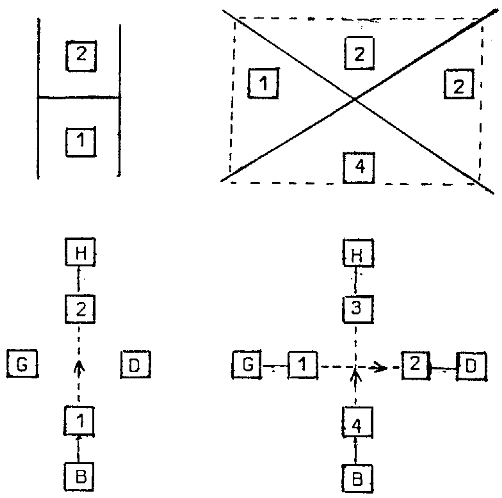
Cependant l'examen de la figure 6 montre la limite du procédé.



<u>Figure 6</u> - La seule région détectée est une région fermée.

$\boxed{1}$: les quatre graphismes sont A_1-<u>équivalents</u>.

On notera que l'algorithme A_1 peut s'appliquer à partir de l'un quelconque des côtés, ce qui donne en fait quatre possibilités qui peuvent se conjuguer. On introduit alors les points $\boxed{H}$ et $\boxed{B}$ pour haut et bas. La figure 7 illustre ceci.



<u>Figure 7</u>

Dans la figure 4, l'algorithme A_1 décrit ne donne aucune région pour le H. Dans l'algorithme A_1 avec rotation de 90° apparaît deux régions ouvertes vers le haut et le bas respectivement. Le X est représenté après utilisation de l'algorithme A_1 dans les deux orientations.

3.2. Graphes relationnels associés aux branches du graphisme.

Lorsque le trait ne délimite pas de région marquée par A_1 il faut utiliser l'algorithme A_2. Le graphe relationnel <X4, U4, R4 f'> correspondant est caractérisé par :

X' est l'ensemble des parties marquées du graphisme

R' est l'ensemble des relations suivantes :

r'_1 au dessus de : x r_1 y est représenté $x \dashrightarrow y$

r'_2 adjacent à gauche de : x r_2 y est représenté $x \rightarrow y$

(voir figure 8)

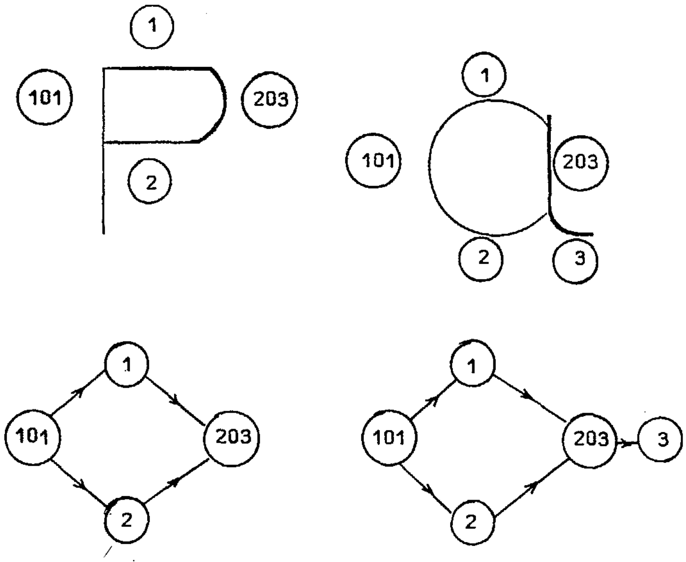


Figure 8 - Les graphes relationnels du 'p' et du 'a' ne sont pas isomorphes.

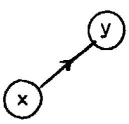
La figure 8 montre comment la branche 3 du 'a' non adjacente à une région donnée dans l'algorithme A_1, distingue ici les deux graphismes. On observe que A_2 est 'aveuglé' à la branche pendante du 'p' ce qui placera encore p, q, b, d dans la même classe d'équivalence.

Pour lever de telles ambiguités, il faut utiliser A_2 dans les quatre directions.

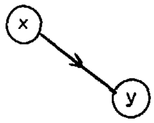
La fusion des diverses relations, les traits atomiques (ou minimaux non vides) intersection de traits obtenus au moyen de A_2 dans diverses directions, sont illustrés en figure 9. Les notations sont :

x est adjacent à gauche de y

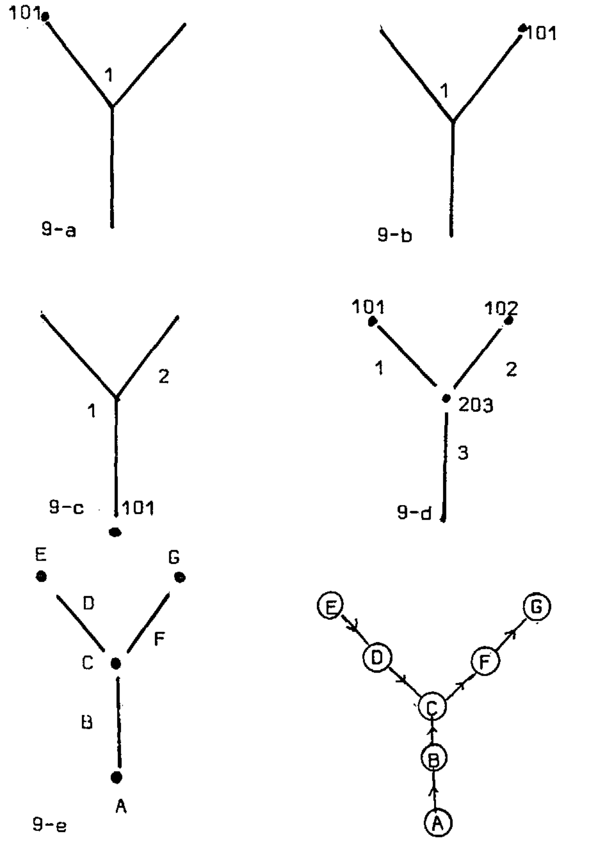
x est adjacent au dessous de y



si x est à gauche et au dessous



si x est à gauche et y au dessous.



9-f

Figure 9 - 9.a à 9.b analyse dans lesquatre directions 9.e synthèse des quatre analyses, 9.f graphe relationnel résumé.

3.3. Utilisation conjointe de A_1 et A_2. Simplification du graphe relationnel

En se basant sur la relation d'adjacence entre régions et traits fournis par A_1 et A_2 il est possible de compléter les graphes relationnels comme l'illustre la figure 10. Des simplifications apparaissent alors en supprimant les arcs bordant une région.

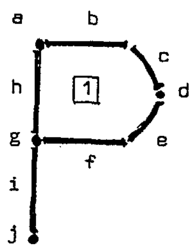
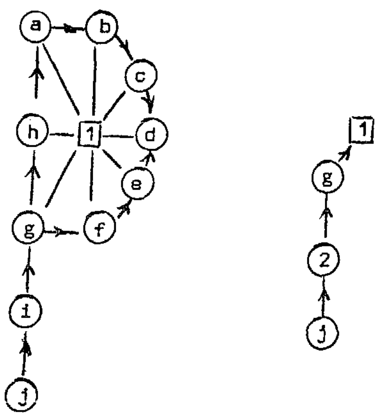


Fig. 10.a



10.b
graphe relationnel

10.c
graphe relationnel simplifié

~~Figure 10~~ Simplification du graphe relationnel du P

4. CONCLUSION

Les algorithmes A_1 et A_2 utilisés dans deux directions orthogonales, nous permettent donc de caractériser les boucles fermées, ouvertes dans les quatre directions, ainsi que l'étiquettage des branches, leurs positions relatives.

Les graphes relationnels permettent une description du graphisme liée à sa topologie en vue d'une reconnaissance basée sur les homomorphismes de graphes simplifiés.

Au stade actuel, les méthodes décrites permettent surtout d'illustrer l'utilisation et l'efficacité des méthodes structurelles en reconnaissance des formes.

Au plan de l'utilisation pratique, il est bien clair que des considérations métriques et quantitatives plus généralement doivent être ajoutées. Nous ne l'avons pas fait ici pour simplifier l'exposé.

BIBLIOGRAPHIE

(1) Y. FUJIMOTO, S. KADOTA and al.
Recognition of hand printed characters by non linearnear elastic matching
3th IJC PR Nov. 76 Coronado - Cal.

(2) E. HISDAL, Nils CHRISTOPHERSEN and al
Structural recognition of handwritten
3th IJC PR Nov 76. Coronado - Cal.

(3) M. NADLER
Structural codes for omnifont and handwritten characters
3th IJC PR Nov. 76 Coronado - Cal.

(4) T. PALVIDIS
Syntactic feature extraction for shape recognition
3th IJC PR Nov. 76 Coronado - Cal.

(5) S. CASTAN - A. NABONNE
Un algorithme de squelettisation opérant par tests locaux sur des suites de voisinages successifs.
Colloque national sur le traitement du signal et de ses applications
Nice 16-21 Juin 1975.