

50376
1988
73



50376
1988
73

N° ordre 1429

UNIVERSITÉ DES SCIENCES ET TECHNIQUES DE LILLE
FLANDRES ARTOIS

THÈSE

présentée à

**L'UNIVERSITÉ DES SCIENCES
ET TECHNIQUES DE LILLE**

pour obtenir le grade de

DOCTEUR DE 3^e CYCLE

SPÉCIALITÉ :

AUTOMATIQUE & INFORMATIQUE INDUSTRIELLE

par

Farhat YATIM

**RECONNAISSANCE DE CARACTÈRES MULTIFONTES
PAR UNE STRUCTURE PLURI-PROCÉDURES**

soutenue le 13 juin 1988 devant la commission d'examen

MM. P. VIDAL

Professeur, Université de Lille I

Président

G. STAMON

Professeur, Université Paris V

Rapporteur

A. FAURE

Professeur, Université du Havre

J.-G. POSTAIRE

Professeur, Université de Lille I

Membres

H. EMPTOZ

Maître de Conférences, I.N.S.A., Lyon

B. TACONET

Maître de Conférences, Université du Havre



AVANT-PROPOS

Ce travail a été effectué au Laboratoire d'Analyse et de Commande des systèmes (L.A.CO.S.-I.T.E.P.E.A.) de l'université du Havre, dirigé par Monsieur le Professeur Alain FAURE.

La direction de recherche est sous la responsabilité de Monsieur Bruno TACONET Maître de Conférences.

Le matériel sur lequel nous avons travaillé a été obtenu grâce au contrat de plan Etat-Région Haute Normandie.

REMERCIEMENTS

Je remercie vivement Monsieur Pierre VIDAL Professeur à l'Université des Sciences et Techniques de Lille Flandres Artois d'avoir bien voulu me faire l'honneur d'accepter la présidence de ce jury.

Mes remerciements et ma reconnaissance iront également au Professeur Georges STAMON, de l'Université Paris V, pour avoir accepté d'examiner attentivement ce mémoire.

Je tiens à remercier particulièrement Monsieur Alain FAURE directeur du Laboratoire d'analyse et de commande des systèmes de l'Université du Havre de m'avoir accueilli au sein de son équipe de recherche et participer à ce jury.

Je remercie vivement le Professeur Jack-Gérard POSTAIRE Professeur à l'Université des Sciences et Techniques de Lille Flandres Artois d'avoir accepté de participer à ce jury.

Je suis très honoré de la présence dans ce jury de Monsieur Hubert EMPTOZ Maître de Conférences à l'INSA de Lyon. Je le remercie pour l'intérêt qu'il a porté à mon travail.

Je suis très heureux de pouvoir exprimer ma sincère reconnaissance à Monsieur Bruno TACONET Maître de Conférences à l'IUT du Havre, directeur de ma thèse pour son dévouement et les conseils qu'il m'a apportés et ce, dans un esprit fraternel.

Je tiens à exprimer ma gratitude à l'égard de Monsieur Paul BIZET directeur technique de la société AI VISION systèmes pour m'avoir prêté une carte d'acquisition d'images, pour m'avoir reçu pendant quelques jours dans les locaux de la société et pour avoir bien voulu tester et juger la réalisation de ce travail.

Mes remerciements iront également aux personnels de la société AI VISION pour leur disponibilité et leur gentillesse.

Enfin, je tiens à remercier tous les membres du L.A.CO.S pour avoir contribué à maintenir une excellente ambiance.

INTRODUCTION

Plusieurs travaux en reconnaissance de caractères ont déjà vu le jour et des systèmes de reconnaissance sont commercialisés [1]. Ces systèmes sont conçus pour l'automatisation de certaines tâches telles que la saisie de données pour le secteur bancaire ou l'acquisition de texte pour le secteur de l'édition.

Un système de reconnaissance de caractères multifontes est performant si :

- 1) Le taux de reconnaissance est supérieur à 98%
- 2) La vitesse de reconnaissance est supérieure à 10 caractères par seconde

Nous avons cherché, dans ce travail, à concevoir une méthode générale de reconnaissance multifonte, rapide et performante; puis à accroître la vitesse de reconnaissance par l'introduction de deux méthodes complémentaires, tout en maintenant un taux de reconnaissance élevé.

La méthode générale, de type structurel, s'appuie essentiellement sur la détection et la localisation des points caractéristiques de la forme squelettisée.

Pour les méthodes rapides, seul le premier niveau de reconnaissance a été pris en compte, ce qui assure un taux de substitution très faible, mais en contrepartie quelques caractères seulement sont reconnus.

Cinq procédures, qui mettent en jeu la méthode générale couplée éventuellement à l'une ou à l'autre ou aux deux méthodes rapides, sont disponibles.

Nous sélectionnons automatiquement la procédure de reconnaissance la plus rapide en minimisant le temps moyen de reconnaissance. Le choix de la procédure dépend de la fréquence d'apparition des caractères : à titre d'exemple, la fréquence de la lettre E dans un texte littéraire en langue française est de 17% environ [2].

Nous résumons dans un premier temps les principales approches existantes en reconnaissance de caractères; nous exposons ensuite les méthodes de reconnaissance développées ou retenues, et justifions enfin la structure de reconnaissance adoptée.

CHAPITRE I

**APPROCHES UTILISEES EN RECONNAISSANCE
DE CARACTERES**

Le problème en reconnaissance de formes (RF) est de donner un nom connu à une forme inconnue.

Soient X l'espace de représentation et Ω l'espace d'interprétation. Le but de la RF est de trouver un opérateur de reconnaissance qui effectue une application de X dans Ω [3],[4]. Dans le cas de la reconnaissance de caractères, l'espace de représentation est la description des caractères d'un texte écrit sur un papier par exemple. L'espace d'interprétation est le résultat de la reconnaissance qui peut être le code ASCII de ces caractères.

On distingue trois phases (figure 1) dans un système de reconnaissance de caractères [5]:

- Le prétraitement où une information exploitable par la deuxième phase est extraite.
- L'extraction de caractéristiques ou primitives.
- La décision.

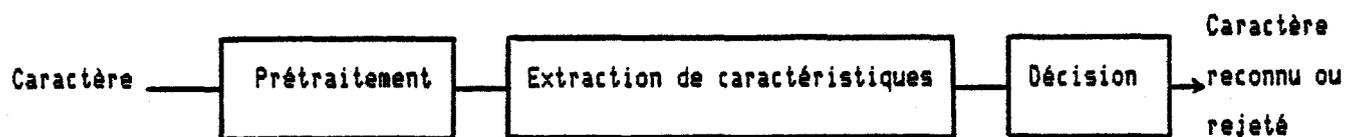


Figure 1. Organisation d'un système de reconnaissance de caractère

Après avoir présenté les méthodes usuelles de prétraitement, nous exposons les principales approches en reconnaissance de caractères.

I-1 LE PRÉTRAITEMENT

Le prétraitement est une phase très importante dans un système de reconnaissance de caractères puisqu'il conditionne la phase suivante qui est l'extraction des caractéristiques (primitives) décrivant le caractère.

Nous donnons ci-dessous les méthodes les plus utilisées en prétraitement.

I-1-1 Seuillage et binarisation:

Cette opération permet de faciliter l'exploitation de l'image et réduit le nombre de points de celle-ci.

Soient $G(i, j)$ le niveau de gris d'un pixel de l'image et $B(i, j)$ sa valeur binarisée.

Si $G(i, j) > S$ alors $B(i, j)=1$ sinon $B(i, j)=0$.

S étant le seuil de binarisation. Si la valeur du seuil est faible, le caractère a tendance à s'empâter. Dans le cas contraire, le caractère peut se couper.

Ce seuil est généralement obtenu après exploitation de l'histogramme .

KINDER [6] choisit le seuil de binarisation donnant la meilleure description du caractère . Dans ce but, il étudie la variation de la densité du caractère en fonction du seuil d'acquisition (figure 2) .

La densité est définie par=
$$\frac{\text{Nbre de points } \epsilon \text{ au caractère } \times 100}{\text{Nbre total de points encadrant le caractère}}$$

Le seuil choisi est inférieur au seuil S_d pour lequel la première coupure apparaît.

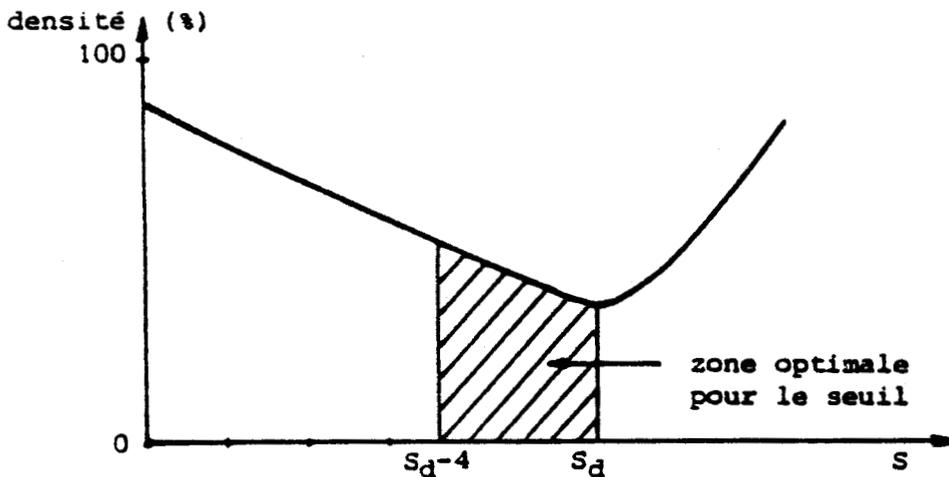


Figure 2, Détermination de la zone optimale pour le seuil

Le seuil de binarisation n'est pas facile à trouver quand l'image n'est pas très contrastée. Ce n'est pas le cas dans notre application puisque nous travaillons sur des

document imprimés (écriture noire sur fond blanc). Les techniques de restauration d'image, de lissage du bruit ne seront pas alors développées.

I-1-2 Extraction et analyse du contour:

Le contour d'une image est la zone de transition séparant deux plages dont les éclaircissements répondent à des modèles statistiques différents [26]. La plupart des opérateurs de détection de contour sont de type dérivée première (méthode du gradient) ou de dérivée seconde (méthode du laplacien). Ces opérateurs donnent de bons résultats quand l'image est très contrastée (nous donnons comme exemple: Sobel, Roberts...). Lorsque l'image est binaire, une simple étude de voisinage permet l'obtention du contour.

L'analyse du contour est une opération qui consiste à extraire l'information donnée par l'exploration du contour du caractère. Ce contour est approximé en segments de droites de directions différentes puis codé. Le codage le plus utilisé est celui de FREEMAN (figure 3).

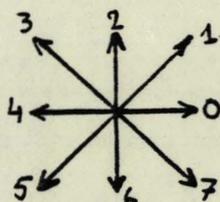


Figure 3. Code de FREEMAN

Dans [7], GOKANA utilise une variante de ce codage qui ne privilégie que cinq directions puisqu'il ne s'intéresse qu'aux profils gauche et droit du chiffre manuscrit (figure 4). Ces profils sont obtenus après approximation des segments du polygone circonscrit du caractère (figure 5).

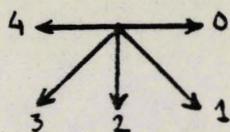


Figure 4. Codage utilisé par GOKANA

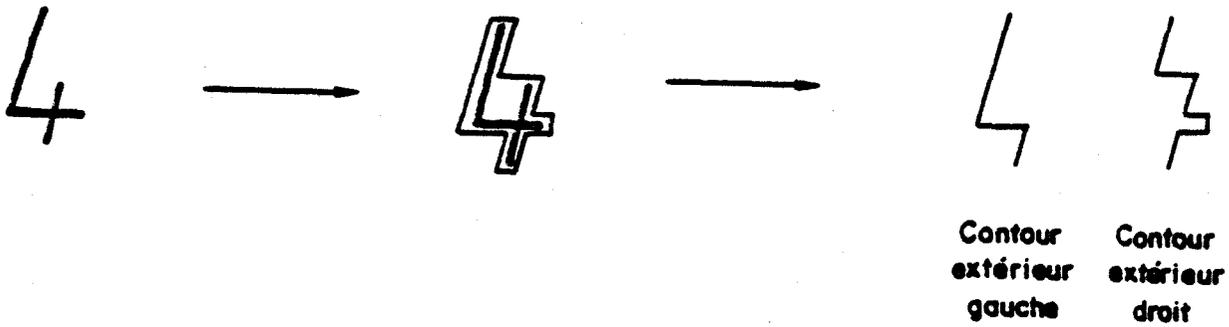


Figure 5. Contour gauche et droit du chiffre 4

I-1-3 Squelettisation

La squelettisation est une opération très intéressante pour des formes d'épaisseur sensiblement constante et faible (exemple: les caractères) d'une part parce que l'information utile est contenue dans le squelette et d'autre part parce qu'elle réduit notablement le nombre de points de l'image et permet ainsi de diminuer le temps des traitements ultérieurs. Cependant, elle n'est pas très utilisée à cause de son coût en temps de calcul.

L'opération de squelettisation sur un caractère consiste à chercher une ligne médiane ayant le même ordre de connexion (le nombre de composantes connexes) que le caractère lui-même.

Soit P une partie de R^2 représentant le caractère.

Soit F la frontière du caractère.

Si pour $x \in P-F$, $d(x, F) = \inf_{y \in F} \{ d(x, y) \}$; $d(x, y)$ étant une distance dans R^2

L'ensemble E tel que :

$$E = \{ x \in P-F; \exists y \in F, \exists z \in F, y \neq z \text{ tels que } d(x,y) = d(x,z) = d(x,F) \}$$

est une solution squelette.

Cette solution existe et elle est unique dans le cas continu mais elle peut ne pas exister dans le cas discret. Cette méthode est intéressante puisque le squelette est très proche du caractère initial mais elle est coûteuse en temps de calcul.

Plusieurs algorithmes de squelettisation sont disponibles dans la littérature [11], [12]. On demande le plus souvent à la squelettisation de respecter la connexité, d'assurer l'épaisseur unitaire du trait et de ne le privilégier dans aucune direction.

Selon l'algorithme utilisé, le squelette obtenu sera plus ou moins centré dans la forme initiale et plus ou moins ramifié. On peut classer les algorithmes de squelettisation en deux grandes familles:

a) Les algorithmes d'amincissement où plusieurs balayages de l'image sont effectués et à chacun d'eux, une étude de voisinage de chaque point permet de déterminer si celui-ci peut ou non être supprimé. Ces points sont éliminés en série ou en parallèle. Ceci s'effectue sous contrainte de préservation des caractéristiques topologiques de la forme initiale. Le nombre de balayages de l'image est croissant avec l'épaisseur de la forme.

Pour squelettiser le chiffre manuscrit, TOHME [8] applique un algorithme de type parallèle. Au cours d'une itération (lecture complète de l'image), l'algorithme

examine successivement les quatre frontières de l'image et supprime tous les points inessentiels d'une même frontière. Deux mémoires sont nécessaires; une contenant les points du squelette et une deuxième contenant les points inessentiels.

Le squelette obtenu (figure 6) est d'épaisseur unitaire et la connexité est conservée.

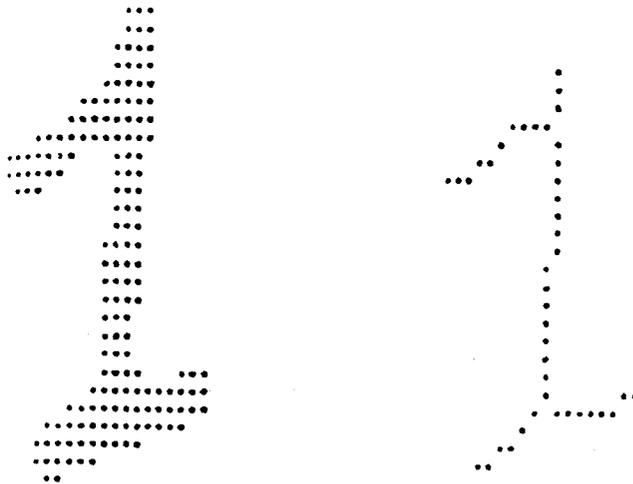


Figure 6. Squelettisation par TOHME

NACCACHE et SHINGAL [9] ont comparé l'algorithme proposé par STEFANELLI et ROSENFELD à celui proposé par HILDITCH.

Le squelette obtenu par le premier algorithme n'est pas d'épaisseur unitaire et dans certains cas, il ne représente pas la forme initiale. Les points éliminés doivent appartenir à la frontière et ne doivent pas couper la forme.

L'algorithme proposé par HILDITCH, conserve les points terminaux. Le nombre de balayages par cette méthode est de 3,3 en moyenne alors qu'il est de 4,3 par la méthode de Stefanelli sur le caractère de la figure 7.



Figure 7. Résultats de la squelettisation (d'après [9])

b) Les algorithmes fondés sur la transformation de distances. Le nombre de balayages de l'image est indépendant de l'épaisseur de la forme. Ces algorithmes sont plus rapides que ceux de la première famille quand l'épaisseur est importante.

ARCELLI et SANNITI DI BAJA [10] ont réduit le nombre de balayages de l'image complète à six (indépendamment de l'épaisseur de la forme) et ont affecté, à chaque point, tous les paramètres nécessaires : degré d'intériorité du contour et des couches internes, nombre de voisins directs, degré de connexité, comptage des transitions, etc...

Deux balayages successifs, de sens contraire (par exemple du haut vers le bas et de gauche à droite puis du bas vers le haut et de droite à gauche) permettent d'évaluer la profondeur du pixel. On calcule le degré de profondeur d'un pixel d_i :

$$d_i = \inf (d_j + 1)$$

d_j est le degré de profondeur d'un voisin déjà étiqueté (8-voisinage). Un pixel ayant un voisinage incomplet appartient à la première couche (étiquette 1).

Un troisième balayage permet d'extraire les pixels du noyau selon les critères suivants:

a) Son degré de profondeur est un maximum local dans son 8-voisinage.

b) Le pixel est un centre de symétrie.

Un quatrième balayage assure la connexité du squelette en adjoignant les pixels qui obéissent à certaines conditions de voisinage (voir détails dans [10]).

Un cinquième balayage fusionne les chemins adjacents.

Au cours du dernier balayage, les pixels superflus sont éliminés afin d'obtenir une épaisseur unitaire.

Parmi les travaux en reconnaissance de caractères utilisant la squelettisation, citons ceux de KAHAN [13] qui extrait de la forme squelettisée les segments de droites constitutifs et en calcule les caractéristiques : coordonnées du centre, longueur, inclinaison.

Les différents algorithmes de squelettisation que nous avons présenté sont plus ou moins rapides mais restent coûteux en temps de calcul. Nous présentons au chapitre II un algorithme de squelettisation rapide que nous avons développé .

I-2 EXTRACTION DE CARACTERISTIQUES (OU PRIMITIVES)

Nous donnons dans ce paragraphe un aperçu sur les différentes approches de reconnaissance de caractères [5], [27].

Les premières méthodes développées sont les méthodes statistiques où des mesures localisées sont effectuées sur le caractère et en fonction de ces mesures, le caractère sera affecté à la classe la plus probable. Ces méthodes donnent de bons résultats en monospace mais leur extension au multispaces ou au manuscrit a été un échec.

C'est l'approche utilisée par DE CARCENAC DE TORNE [15]. Les caractères sont décrits par des vecteurs à T composantes binaires représentant la détection ou l'absence de caractéristiques géométriques.

Après décomposition syntaxique de ces vecteurs et calcul de distances à chaque classe de comparaison, le caractère est affecté à la classe la plus proche au sens de ces mesures.

Les méthodes géométriques consistent à appliquer une série de tests portant sur des traits distinctifs du caractère et de comparer leurs réponses à celles données par apprentissage sur des caractères de référence.

GAILLAT [16] propose une structure de décision "intelligente" qui permet d'exploiter au maximum le pouvoir discriminant de chaque test.

les méthodes structurelles ou syntaxiques ont fait l'objet de nombreuses études et sont les plus utilisées. Le caractère est représenté par ses composantes primitives et leur composition .

CHEHIKIAN [17] décompose le caractère en segments (primitives) verticaux, horizontaux, obliques à pente positive et oblique à pente négative (figure 8). Cette décomposition est obtenue après étude du voisinage de chaque point. Le caractère est ensuite décrit par le mode de concaténation de ses primitives.

Cette méthode tolère une rotation de 20° mais les obliques de grande pente ou de pente faible sont considérées comme des segments verticaux ou horizontaux.

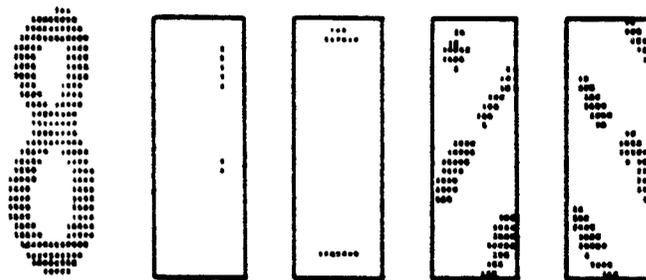


Figure 8. Décomposition du chiffre 8 en ses primitives verticales, horizontales et obliques.

BELAID [18] décrit le tracé sur une tablette graphique par une séquence temporelle de primitives. Ces primitives sont:

- 
- Trait ou segment de droite (T)
 - Courbe ou arc de cercle (C)
 - Lever de plume (L)

Le caractère R de la figure suivante peut-être codé par:

$$\text{Code (R)} = T+L+C+T+L$$

Cette description ne suffit pas pour distinguer chacun des caractères alphanumériques latins c'est pourquoi d'autres informations sont extraites: le sens d'une courbe, sa longueur relative par rapport à la longueur totale du tracé sont des informations qui permettent de lever certaines ambiguïtés. Ces primitives sont ensuite comparées à tous les chemins d'un arbre de décision construit par apprentissage. Le chemin de l'arbre qui décrit la même liste de primitives que la liste entrée donne le nom du caractère.

KINDER [19] balaie horizontalement et verticalement le caractère et lors du balayage, tous les points contigus sont attribués à un même élément d'ordre k . Les points séparés par un espace jugé suffisant seront attribués à un élément d'ordre $k+1$ (figure 9).

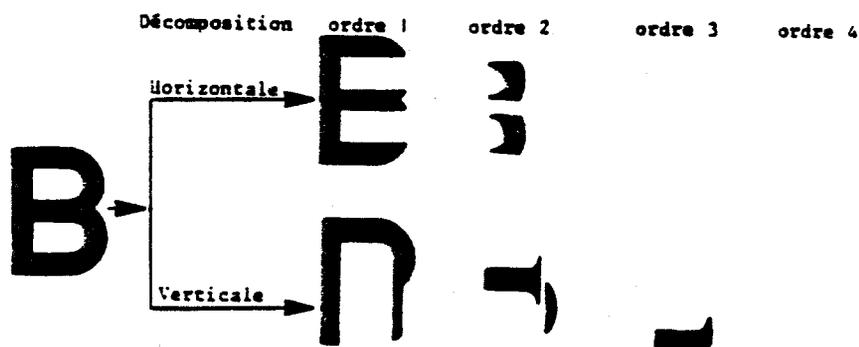


Figure 9, Décomposition du B en éléments d'ordre K

Chaque élément est ensuite décrit par une chaîne de caractères significative de la structure de cet élément. Cette description est obtenue après projection sur un axe vertical pour un élément horizontal et sur un axe horizontal pour un élément vertical.

Un caractère alphanumérique latin est composé de 4 éléments au maximum donc il sera décrit par 8 chaînes de caractères.

PAKKER [20] présente une méthode de reconnaissance structurelle où l'analyse du caractère est de type "hétérarchique".

Après projections partielles du caractère (figure 10), les positions des segments horizontaux et verticaux sont déterminés. A la suite de cette opération, une classification est faite pour regrouper les caractères en sous-classes.

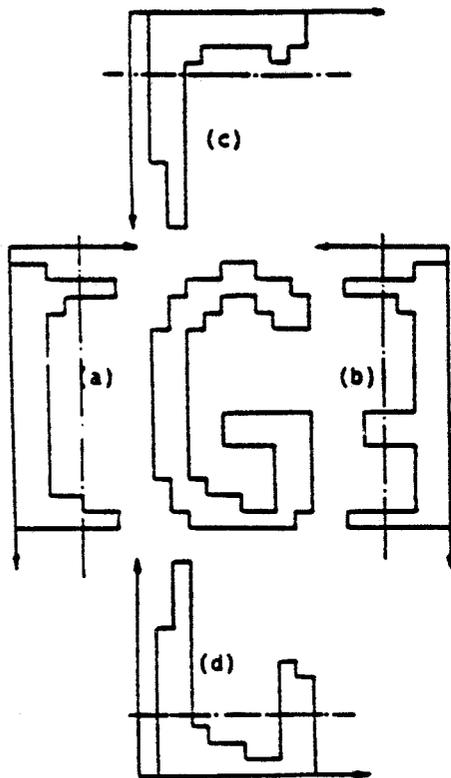


Figure 10

Les quatre projections partielles de la lettre G.

- (a) projection horizontale à gauche
- (b) projection horizontale à droite
- (c) projection verticale en haut
- (d) projection verticale en bas.

Une information de la structure est ainsi obtenue et à l'aide d'une "sonde" bien positionnée, une information sur la classe d'appartenance du caractère est élaborée. Cette information peut-être suffisante pour la reconnaissance;

dans le cas contraire, elle sert à guider une analyse plus détaillée jusqu'à aboutir à une décision.

La méthode utilisée par VO VAN [21] consiste à étudier les intersections du caractère avec un ensemble de droites .

C'est la même méthode utilisée dans le dispositif électronique de lecture de texte pour aveugles [22] : Trois droites horizontales et trois verticales sont positionnées de façon à extraire l'information la plus riche possible (figure 11).

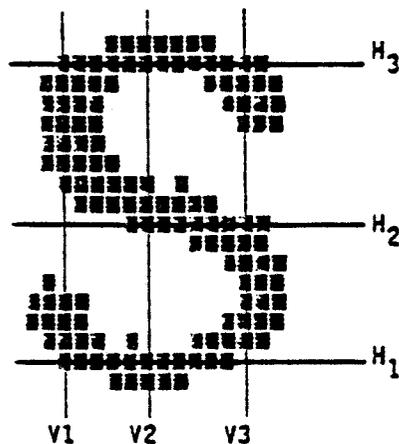
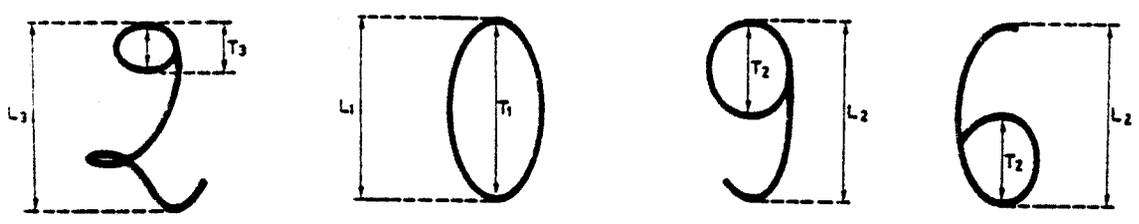
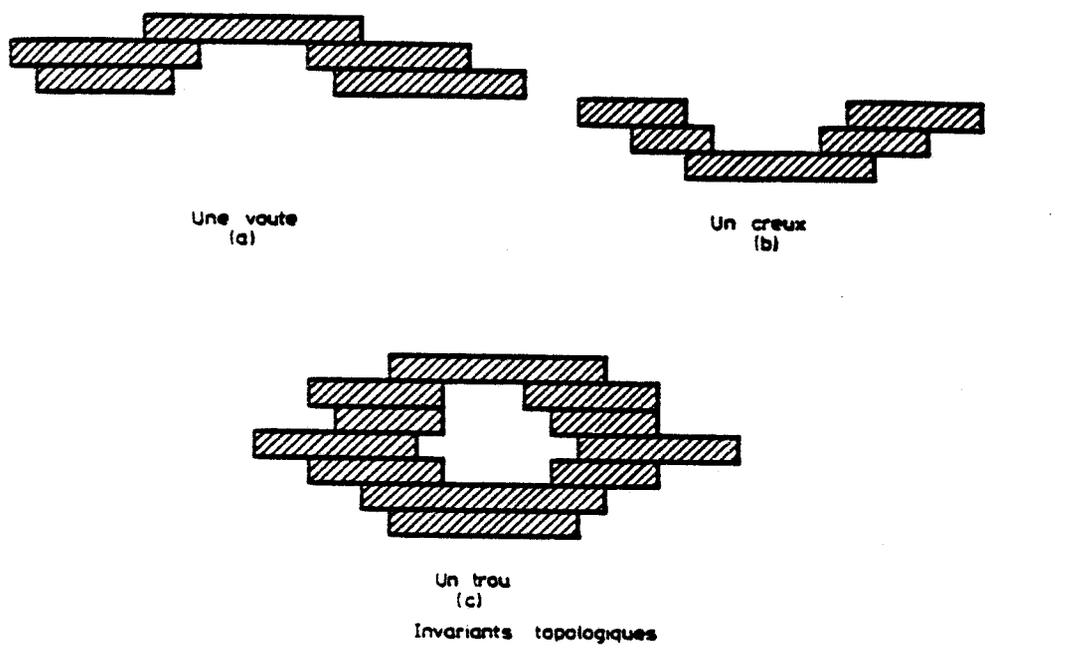


Figure 11. Positionnement nominal des lignes et colonnes

Les caractères à empattement sont automatiquement détectés et par un décalage approprié des colonnes (V_1, V_3) le nombre d'intersections correspondant à la forme pure du caractère est retrouvé. Si le nombre de zones obtenues par intersection d'un caractère avec chacune des droites $V_1, V_2, V_3, H_1, H_2, H_3$ dépasse 9 (nombre maximum pour un caractère latin), un décalage approprié de la droite concernée permet de rechercher une intersection plus stable.

Dans [7], après avoir présenté un algorithme de suivi de contour, GOKANA propose une méthode de reconnaissance du chiffre manuscrit à plusieurs niveaux : au premier niveau, le

caractère est décrit par ses profils gauche et droit obtenus par approximation en segments de l'enveloppe extérieure du caractère et les niveaux suivants (figure 12) utilisent une arborescence basée sur les invariants topologiques (nombre de trous, de voûtes, de creux) et algébriques (rapports entre certaines dimensions significatives du caractère).



Les différents rapports Trou sur Taille

$$R1 = \frac{T_1}{L_1} = 1 \quad R2 = \frac{T_2}{L_2} = 0.5 \quad R3 = \frac{T_3}{L_3} < 0.25$$

Figure 12. Invariants topologiques et algébriques

BENALI [23] propose une méthode de reconnaissance à deux niveaux. Au premier niveau, le caractère est décrit par le nombre de concavités relatives à chacun des quatre côtés

du rectangle ex-inscrit (le dénombrement de ces concavités est obtenu après le suivi de la fonction distance correspondant à chaque côté (figure 13) ainsi que le nombre de boucles. Le caractère est ainsi affecté à une sous-classe de population restreinte (6 au maximum).

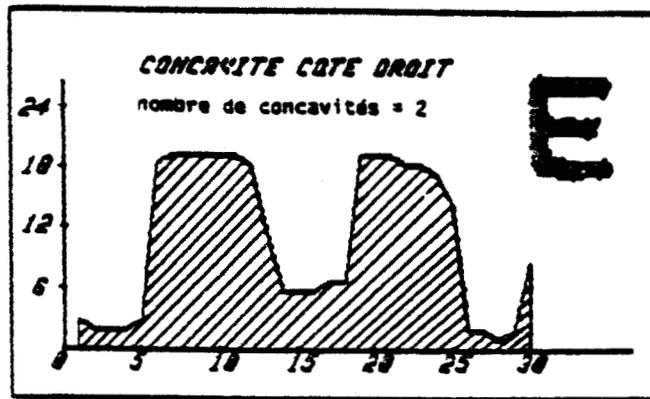
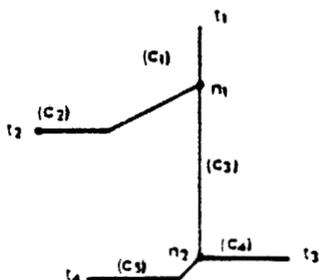


Figure 13. Fonction distance correspondant au côté droit du rectangle ex-inscrit

Le deuxième niveau de reconnaissance utilise des informations complémentaires telles que la position des concavités, leur profondeur et les convexités.

Après avoir squelettisé le chiffre manuscrit et extrait les points caractéristiques du squelette (extrémités et noeuds), TOHME [14] donne une représentation syntaxique où la topologie du caractère est bien intégrée.

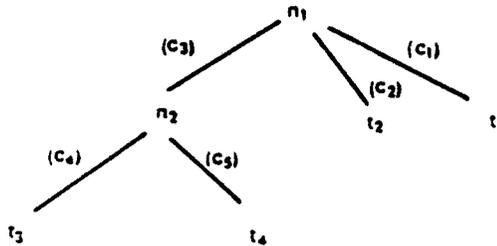
Exemple:



$$N = \{n_1, n_2\}$$

$$T = \{t_1, t_2, t_3, t_4\}$$

N étant l'ensemble des noeuds et T celui des extrémités.
Le squelette est partitionné en cinq classes C_1, C_2, C_3, C_4
et C_5 .



Cet arbre représente syntaxiquement le caractère où les branches ont une signification topologique bien précise.

I-3 NOTRE APPROCHE

Nous proposons une structure de reconnaissance pluri-procédures. Chacune de ces procédures, est une concaténation de méthodes parmi lesquelles une méthode (générale) est toujours présente. Le premier niveau de décision de cette méthode est basé sur la position des points caractéristiques (extrémités et noeuds) du squelette.

La procédure de reconnaissance choisie est celle pour laquelle le temps de reconnaissance est minimum.

CHAPITRE II
METHODE GENERALE DE RECONNAISSANCE

II-1 DISPOSITIF DE RECONNAISSANCE (figure 14):

L'acquisition d'un caractère est effectuée à partir d'une caméra CCD qui est reliée à une carte d'acquisition AI VISION comportant un logiciel permettant :

- l'acquisition d'une partie de l'image matérialisée par une fenêtre en surbrillance. Cette fenêtre est de dimensions réglables.
- la segmentation et l'isolement des caractères. La segmentation est obtenue après projection des caractères contenus dans la fenêtre sur un axe horizontal.
- leur binarisation
- leur normalisation

Les dimensions normalisées sont de 21 lignes sur 15 colonnes soit 315 pixels de résolution ou de 24 lignes sur 16 colonnes soit 384 pixels de résolution .

Le logiciel a été développé au LACOS en langage assembleur 68000 de Motorola.

b-éliminer les points non significatifs sans couper la forme

x	1	x	0	0	0
0	p	0	0	p	0
0	0	0	0	0	0

Dans ce cas p sera mis à 0.

La figure 15 ci-dessous montre le résultat du lissage sur un caractère.

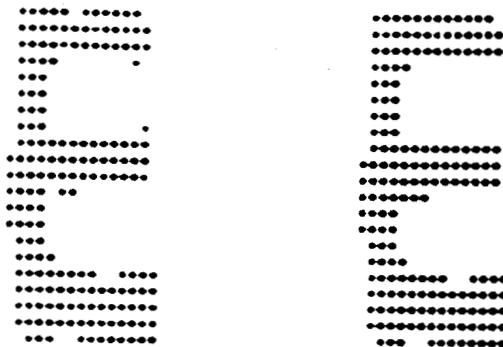


Figure 15. Résultat du lissage

II-2-2 La squelettisation [24]

Nous exposons dans ce paragraphe notre algorithme de squelettisation, qui permet d'accroître notablement la rapidité d'exécution en conservant la connexité et qui assure l'épaisseur unitaire du trait .

Nous avons repris la structure générale présentée par ARCELLI et SANNITI DI BAJA [10], en deux étapes :

- Obtention d'un noyau d'épaisseur faible.
- Affinement du noyau en squelette .

Cependant, pour ces deux étapes, nous avons développé des algorithmes différents de ceux proposés par les auteurs de l'article, en vue d'accroître la vitesse de traitement .

a) Obtention d'un noyau d'épaisseur faible

ARCELLI et SANNITI DI BAJA ont réduit le nombre de balayages de l'image et ont affecté, à chaque point, tous les paramètres nécessaires : degré d'intériorité du contour et des couches internes, nombre de voisins, nombre de voisins directs, degré de connexité, comptage des transitions, etc ...

Notre approche est autre : nous avons cherché à exprimer les conditions d'appartenance d'un point du contour au noyau, sous forme de configurations de voisinage, aisément transposables en opérations logiques élémentaires .

Définitions de voisinage

3 2 1 Voisinage : $V(p) = \{ 0, 1, \dots, 7 \}$

4 p 0 Voisinage direct: $V_0(p) = \{ 0, 2, 4, 6 \}$

5 6 7

Partition en contour et masse interne

Contour : $p \in C$ Card $V(p) < 8$

Masse interne: $p \in I$ Card $V(p) = 8$

Points essentiels du contour

Un point du contour est essentiel, s'il vérifie l'une des conditions suivantes :

1- Aucun voisin n'appartient à la masse interne (ou encore tout voisin est un point du contour : $V(p) \ll C$)

2- Le voisinage du point correspond à un des 12 cas de figure générés par les trois représentations ci-dessous (les autres se déduisent par rotation de 90°).

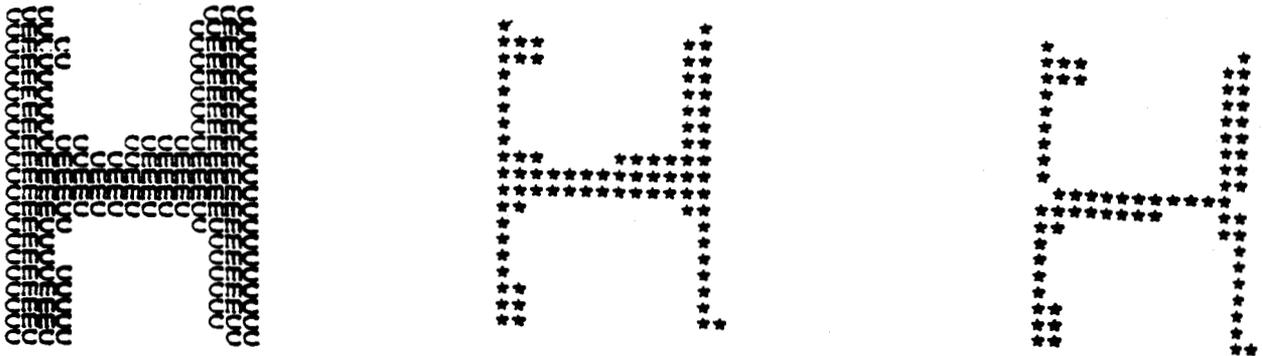
X	C	X		X	C	X		X	C	X
C	p	C		X	p	C		X	p	X
X	X	X		C	X	X		C	X	C

Où X vaut 0 ou 1

A chaque itération, on conserve les points du contour essentiels ; ils sont alors adjoints aux points de la masse interne restante pour former la nouvelle masse initiale . L'algorithme se termine lorsque la masse restante est vide. On peut donc caractériser un noyau ainsi : tout point d'un noyau appartient à son contour, $C(N) = N$.

Par construction, l'algorithme préserve la connexité et respecte parfaitement l'isotropie . Deux mémoires sont requises : l'une pour le contour, l'autre pour la masse interne.

Le nombre de balayages est, à une unité près, la demi-épaisseur maximale de la forme. La figure 16 montre le résultat d'une itération et l'obtention d'un noyau sur un caractère binarisé de dimensions 21 lignes * 15 colonnes, le temps d'une itération est inférieur à 5 ms (programmation en langage assembleur 68000).



Forme initiale

masse restante après une itération

noyau

Figure 16. Résultat d'une itération et obtention d'un noyau

b) Affinement du noyau en squelette

Etant donné la faible épaisseur du noyau, il s'agit d'une opération limitée en temps de calcul. Nous avons envisagé l'algorithme parallèle de TOHME [14] qui supprime plusieurs points inessentiels à la fois.

Un point inessentiel a au moins un zéro dans son voisinage direct. Il faut en plus que l'ensemble des 1 de $V(p)$ soit δ -connexe et l'ensemble des zéros de $V(p)$ soit 4-connexe. Pour arrêter l'algorithme sur une courbe discrète d'épaisseur égale à l'unité, on impose, en plus, qu'au $V(p)$ d'un point inessentiel il y ait au moins deux 1.

Les points inessentiels ont pour configuration :

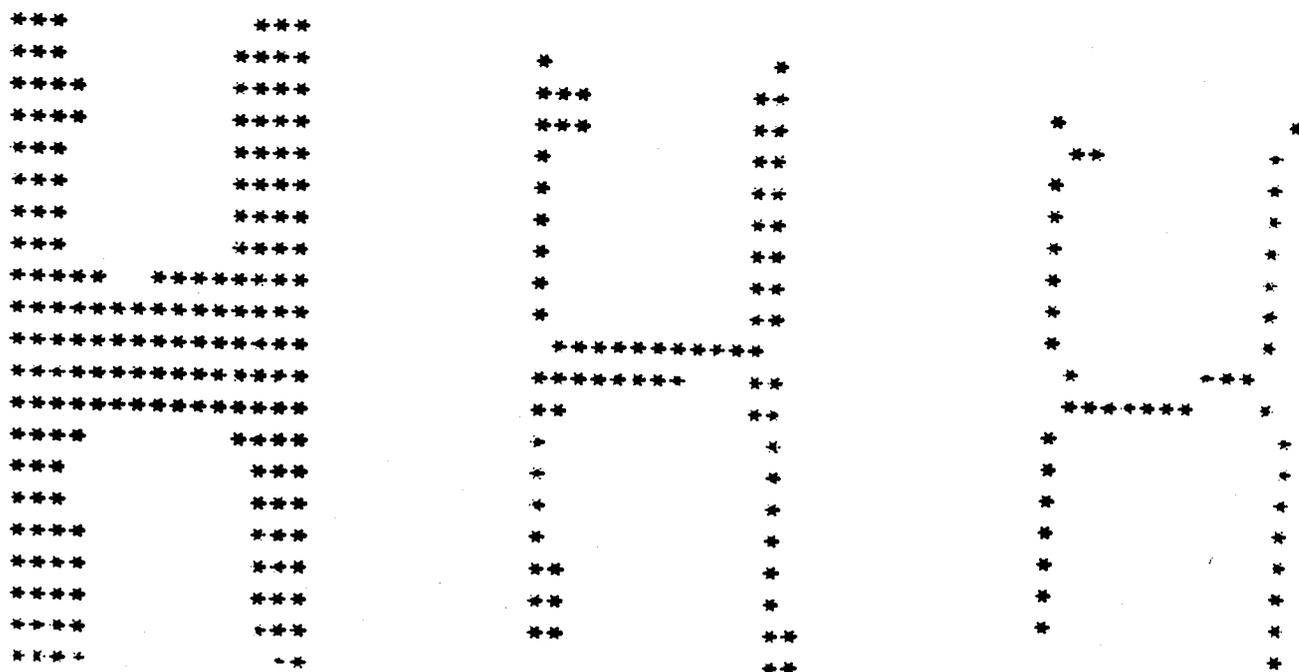
X	1	X	X	0	X	X	1	X	X	1	X
1	p	0	1	p	1	0	p	1	1	p	1
X	1	X	X	1	X	X	1	X	X	0	X
0	0	0	0	0	0	1	1	X	X	1	1
0	p	0	0	p	0	0	p	0	0	p	0
X	1	1	1	1	X	0	0	0	0	0	0

0	0	X	0	0	1	X	0	0	1	0	0
0	p	1	0	p	1	1	p	0	1	p	0
0	0	1	0	0	X	1	0	0	X	0	0
X	0	0	0	0	X	X	1	X	X	1	X
1	p	0	0	p	1	0	p	1	1	p	0
X	1	X	X	1	X	0	0	X	X	0	0

X vaut 0 ou 1

Deux itérations suffisent pour squelettiser le noyau. Une itération comporte quatre balayages, un par direction principale. On a modifié cet algorithme pour réduire le nombre de balayages à quatre au lieu de huit.

Le temps d'exécution est de 10 ms environ pour cette seconde étape. La figure 17 récapitule les deux étapes.



Caractère brut

Noyau

Squelette

Figure 17. Les deux étapes de la squelettisation

II-2-3 Points caractéristiques du squelette

Les points caractéristiques du squelette sont les extrémités et les noeuds.

Une extrémité a un voisin dans $V(p)$ (8 voisins).

Un noeud est détecté après dénombrement des transitions 0-1 dans $V(p)$. Le nombre de ces transitions doit être de 3 au minimum.

Exemple:

```
      *                *  *
     * * *            *
      *                *
```

Le cas suivant, rare mais possible par exemple pour le caractère X, nécessite un examen particulier. Un noeud équivalent est détecté.

```
      *      *
       * *
       * *
      *      *
```

Le premier niveau de reconnaissance sera basé sur ces points caractéristiques.

II-2-4 Ebarbulage et élimination des sérifs

Puisque les sérifs (ou empattements) et le bruit ajoutent des extrémités et des noeuds non significatifs pour la reconnaissance, nous éliminons systématiquement les sérifs puis les branches parasites.

Les sérifs sont facilement détectés puisqu'ils sont situés, pour les caractères imprimés, en haut et en bas du caractère.

Après avoir extrait les points caractéristiques du squelette, nous détectons les sérifs non symétriques (une extrémité est à gauche d'un noeud en haut ou en bas du squelette) puis les sérifs symétriques (un noeud est situé sensiblement au milieu de deux extrémités).

La longueur d'un sérif peut dépasser la demi-largeur du caractère c'est pourquoi les sérifs symétriques de longueur inférieure au 2/3 de la largeur du squelette seront éliminés.

Pour l'ébarbulage, le nombre maximum de points éliminés est trois suivant l'horizontale et un suivant la verticale.

Cependant, la squelettisation peut conduire à privilégier l'empatement dans une seule direction; les branches correspondantes ne sont pas éliminées par l'opération élimination des sérifs. Dans ce cas, ces branches, qui possèdent une extrémité dans la partie centrale supérieure ou inférieure, sont écourtées de trois points (figure 18). Ceci se justifie pour obtenir une bonne description du caractère (cf. paragraphe II-3).

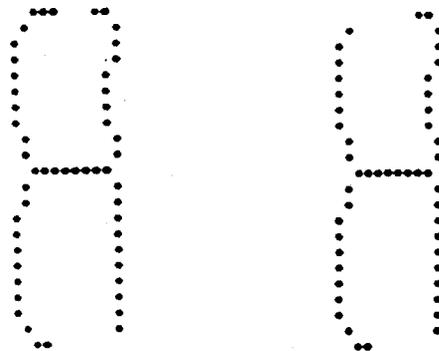
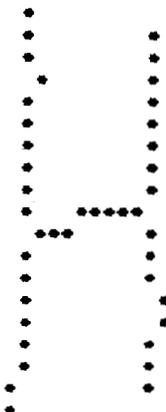
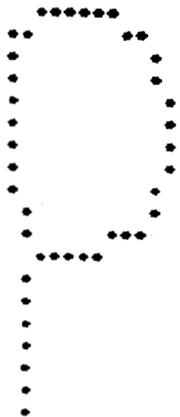
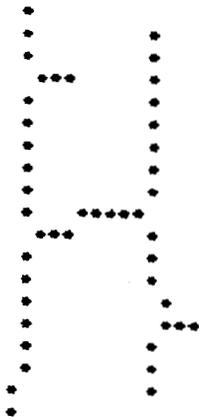
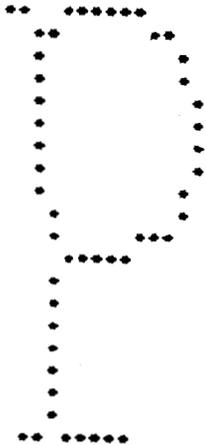
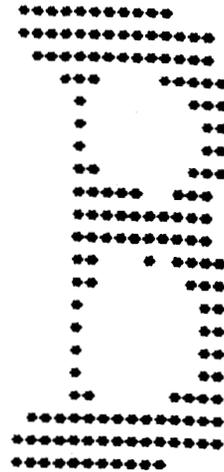
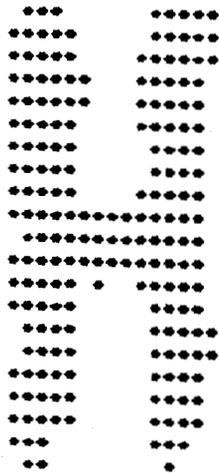
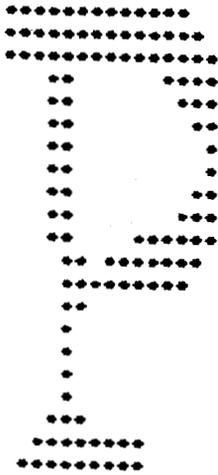
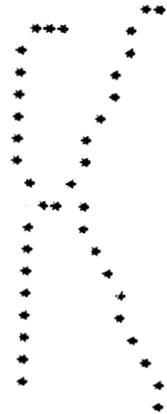
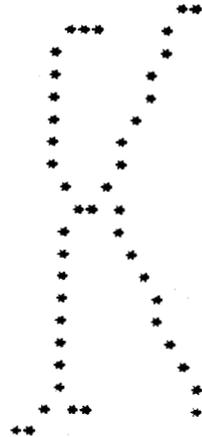
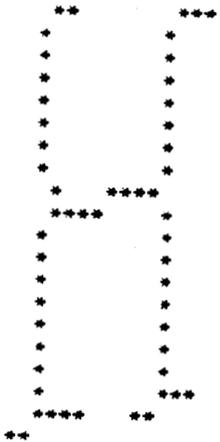
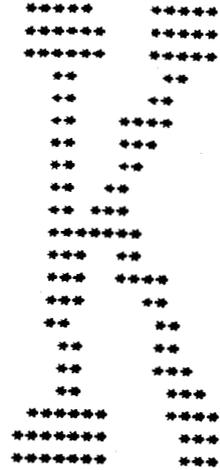
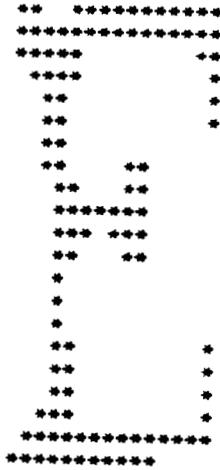
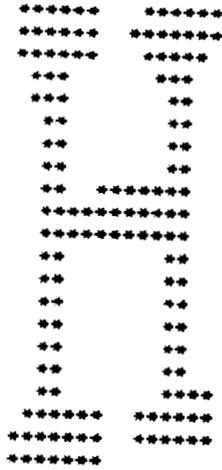
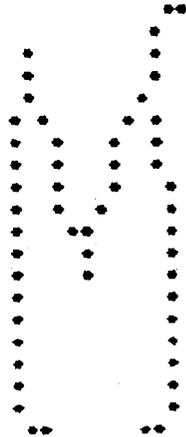
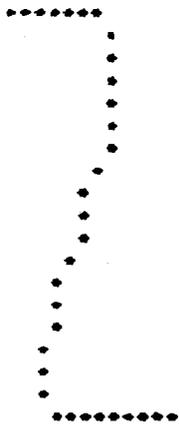
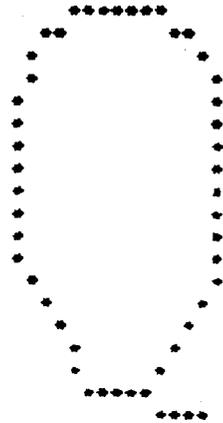
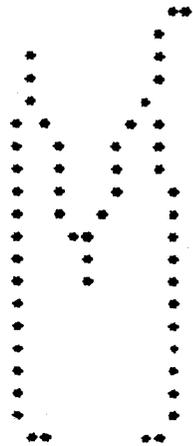
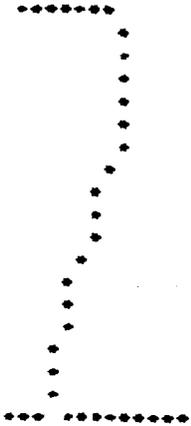
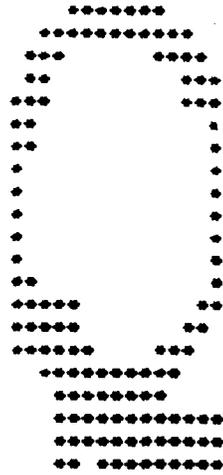
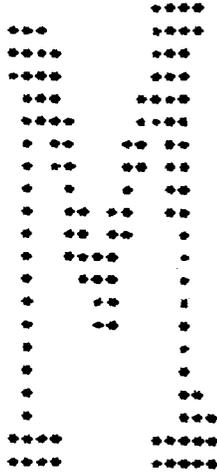
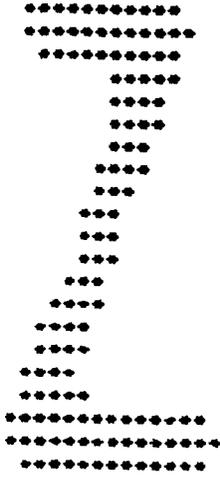


Figure 18. Traitement du squelette du caractère H

Nous donnons ci-dessous le résultat de la squelettisation puis celui de l'ébarbulage et l'élimination des sérifs sur quelques caractères.







II-3 DESCRIPTION DU CARACTERE

Nous avons développé une méthode de reconnaissance à plusieurs niveaux de décision. Le premier niveau est basé sur les points caractéristiques du squelette (extrémités et noeuds). Le test qui consiste à dénombrer ces points n'est pas très discriminant c'est pourquoi, nous nous sommes intéressés à la position des noeuds et des extrémités dans le squelette.

Dans le but de coder la position relative de ces points, nous avons effectué une étude statistique de la position des extrémités et des noeuds par rapport à la hauteur et à la largeur du squelette (figure 19). Cette étude a été faite sur les 36 caractères alphanumériques (lettres majuscules et chiffres) provenant de fontes droites diverses.

Fréquence en %

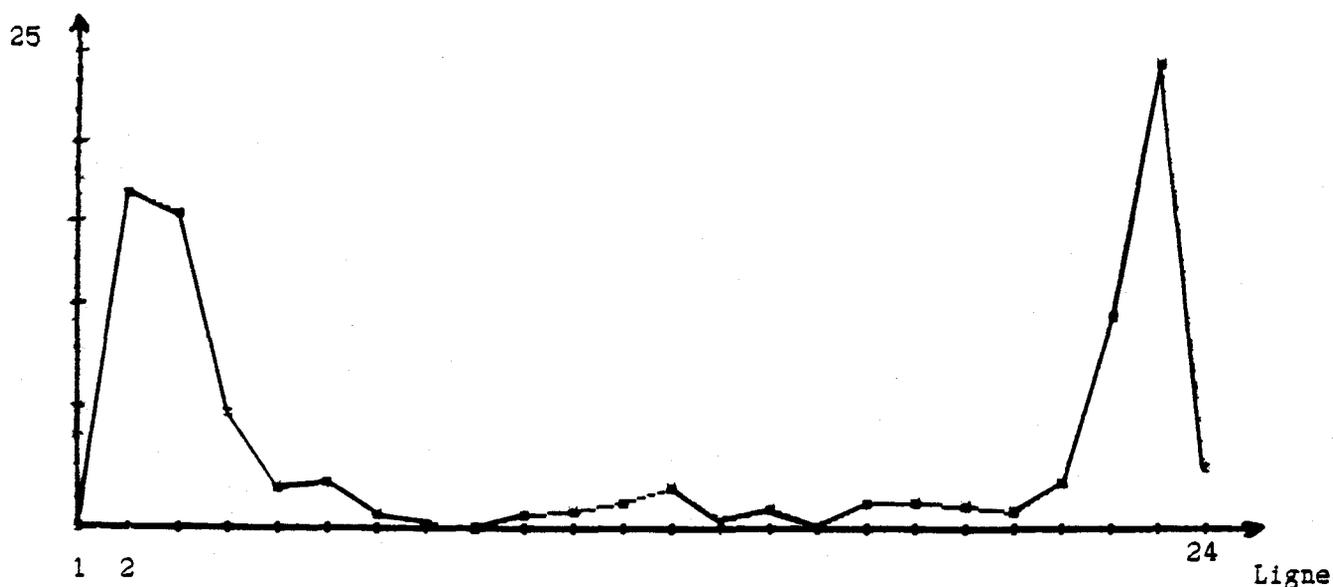


Figure 19a . Extrémités: Histogramme vertical

Fréquence en %



Figure 19b. Extrémités: Histogramme horizontal

Fréquence en %

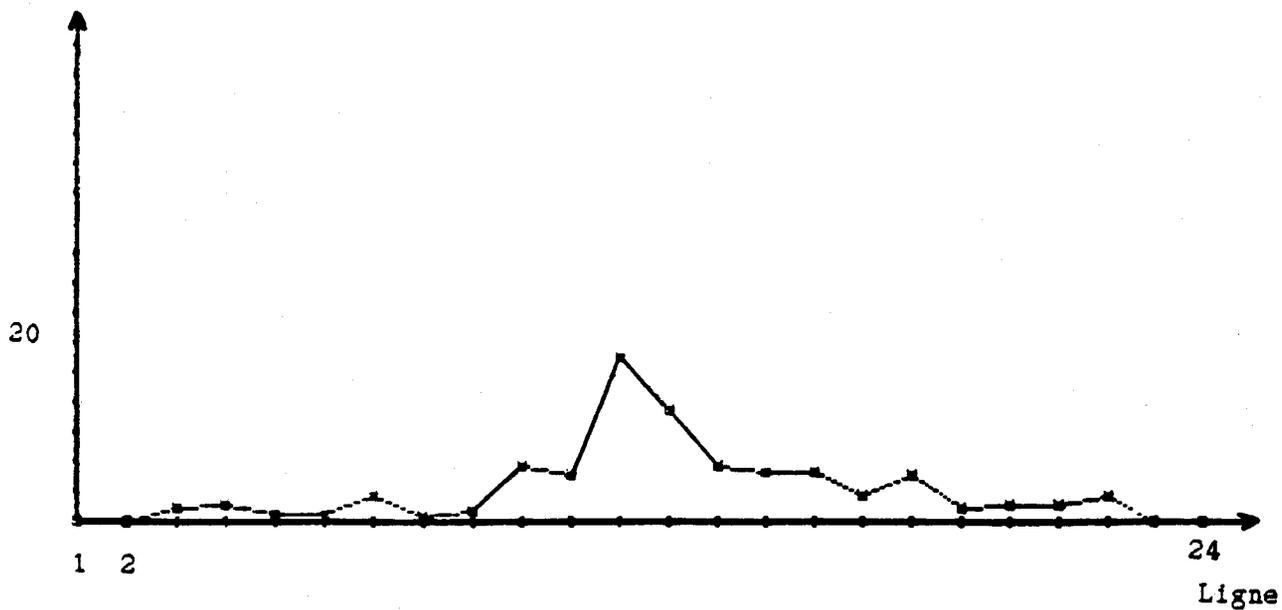


Figure 19c. Noeuds: Histogramme vertical

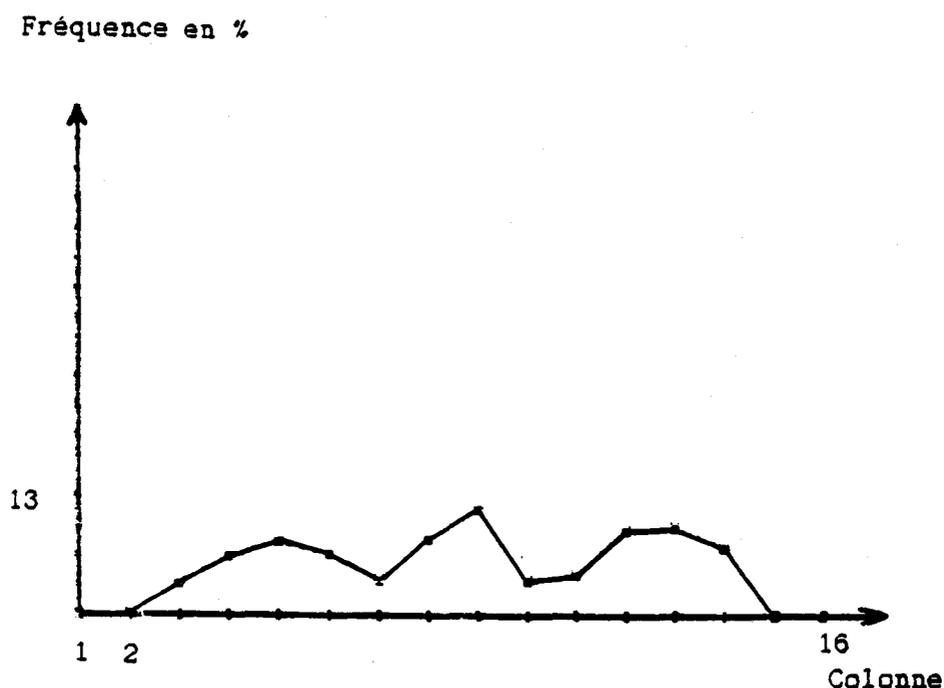
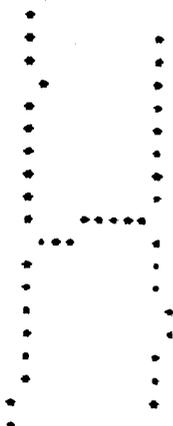


Figure 19d. Noeuds: Histogramme horizontal

En examinant ces histogrammes, on remarque d'abord que pour les caractères alphanumériques, il y a beaucoup plus d'extrémités que de noeuds et que ces extrémités sont situées plutôt en haut ou en bas du caractère, à sa gauche ou à sa droite alors que les noeuds sont plus nombreux au milieu du caractère.

Cette étude statistique nous a mené à partitionner le rectangle enveloppe du caractère squelettisé en neuf zones (figure 20) et à coder la position des points caractéristiques dans ces zones. Une confirmation à posteriori de ce codage nous a été suggérée par le cadre normalisé pour le manuscrit guidé (): les 4 points, les extrémités de la croix et son centre montrent les 9 positions imposées pour les points caractéristiques.

Le caractère sera donc décrit par un code sur 18 bits, 9 pour les extrémités et 9 pour les noeuds. Le code du chiffre 0 et de la lettre O est le zéro puisqu'il n'y a ni extrémité ni noeud dans leur squelette.



Code (H)=00000101000101000101
=5145 (Hex)

Figure 20. Codage de la position des points caractéristiques
du squelette

Le nombre de codes obtenus après apprentissage sur différentes fontes est de 200 environ.

Certains caractères ont un seul code tel que le chiffre 0 et la lettre O qui ne peuvent pas être discriminés en multifontes sauf si on tient compte du contexte, d'autres en ont plusieurs; exemple: le chiffre 4 qui peut prendre des structures graphiques différentes (4, 4, 4, ...).

II-4 PREMIER NIVEAU DE CLASSIFICATION

Le code du caractère inconnu sera comparé à tous les codes rangés en mémoire suivant l'ordre croissant de leur valeur ASCII. Il sera donc comparé avec le code du chiffre 0 puis ceux du 1 et ainsi de suite.

Si ce code est rencontré une seule fois, le caractère sera reconnu (c'est le cas pour les caractères F et P par exemple). Dans le cas contraire, nous serons en présence d'une classe dont le nombre d'éléments est celui des codes rencontrés. Grâce au pouvoir très discriminant de cette première classification, le nombre de classes à étudier est limité et la classe la plus nombreuse est composée de quatre caractères; exemple (J,N,5,S). Notons que le caractère N n'appartient pas toujours à cette classe (si le squelette a un ou deux noeuds); il est le seul représentant de sa classe et peut être reconnu immédiatement à la suite de ce codage.

En outre, des classes inattendues apparaissent telle que la classe (C,E): en effet l'ébarbulage peut éliminer la barre médiane du E si celle-ci est courte (figure 21).

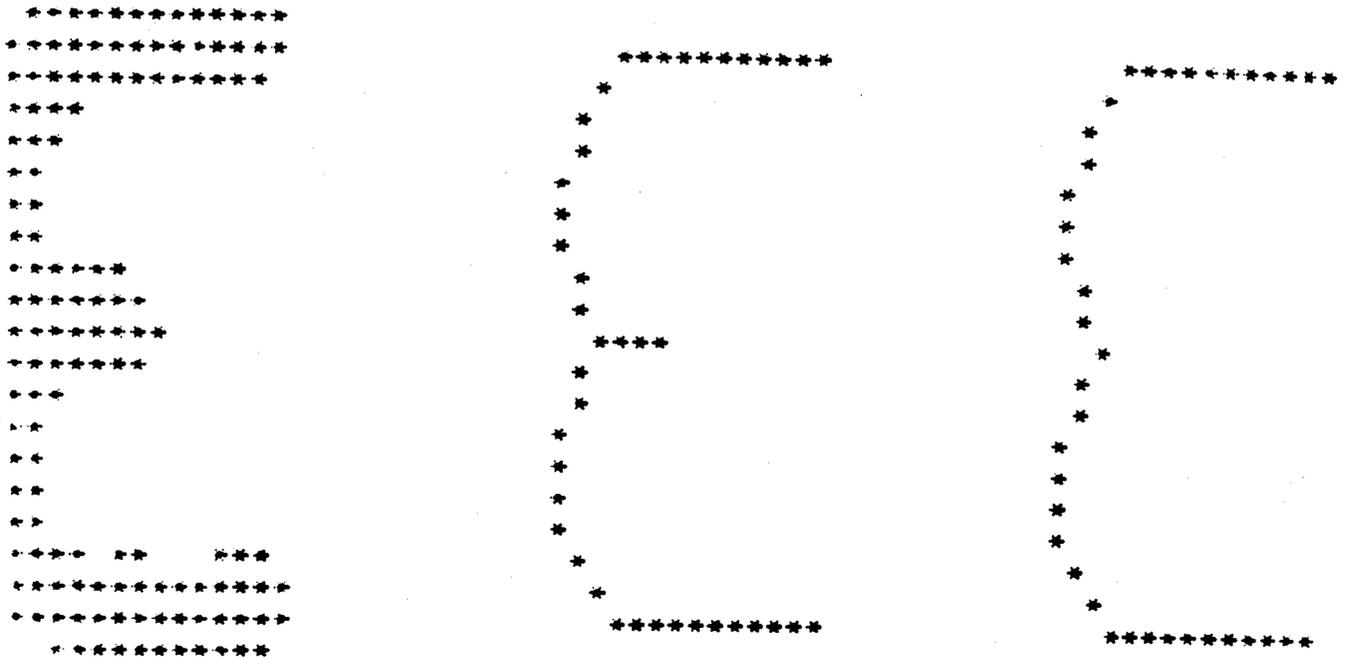


Figure 21. Résultat de l'ébarbulage sur le caractère E

Aussi, un même caractère d'une même fonte peut avoir différentes descriptions pour différentes acquisitions. Ceci est vrai surtout quand la largeur du caractère squelettisé n'est pas importante. Dans ce cas, le codage est sujet à de fréquentes variations; le déplacement d'un point caractéristique (extrémité ou noeud) sur une colonne voisine provoque une nouvelle répartition des zones et peut donc induire un nouveau code.

Nous donnons, à titre d'exemple, les différentes descriptions (code en binaire) obtenues pour le caractère *N* et les classes engendrées par ces descriptions.

Description	Classe
000000000100000001	(5, J, N, S)
000101000101000101	(H, M, N, W)
001000000101000001	N
001000100101000101	N
001100000101000101	N
000000100100000101	N
000100000100000101	N
000001000101000001	N

Pour les six dernières descriptions, le caractère sera reconnu dès le premier niveau de classification.

II-5 DEUXIEME NIVEAU DE CLASSIFICATION

A la suite de l'examen de la position des extrémités et des noeuds du squelette, nous avons effectué une première classification affectant le caractère à une sous-classe de population réduite (dans laquelle il y a 4 caractères au maximum).

Le second niveau de reconnaissance consiste à affiner la description du caractère de façon à le différencier des autres éléments de la même sous-classe.

Les tests utilisés pour la classification finale sont appliqués sur le caractère lissé non squelettisé sauf dans quelques cas très rares où des informations supplémentaires sur les points caractéristiques suffisent. Ces tests sont spécifiques à chaque classe et la discrimination se fait facilement sauf dans le cas des doublets classiques (D,0), (S,5), (B,8), (Z,2).

La majorité de ces tests est basée sur des projections horizontales et verticales du caractère (ceci n'est valable que pour des fontes droites). Cette opération consiste à faire un comptage du nombre de points de l'image sur une colonne ou sur une ligne.

Le caractère R est différencié du A par la présence d'un grand segment vertical à gauche du caractère (figure 22). Si le résultat de la projection verticale est supérieur au $\frac{2}{3}$ de la hauteur du caractère, celui-ci sera reconnu comme un R.

Le caractère U est différencié du V par la présence d'un grand segment horizontal en bas du caractère (figure 23). Si le résultat de la projection horizontale est

supérieur à la moitié de la largeur du caractère, celui-ci sera reconnu comme un U.

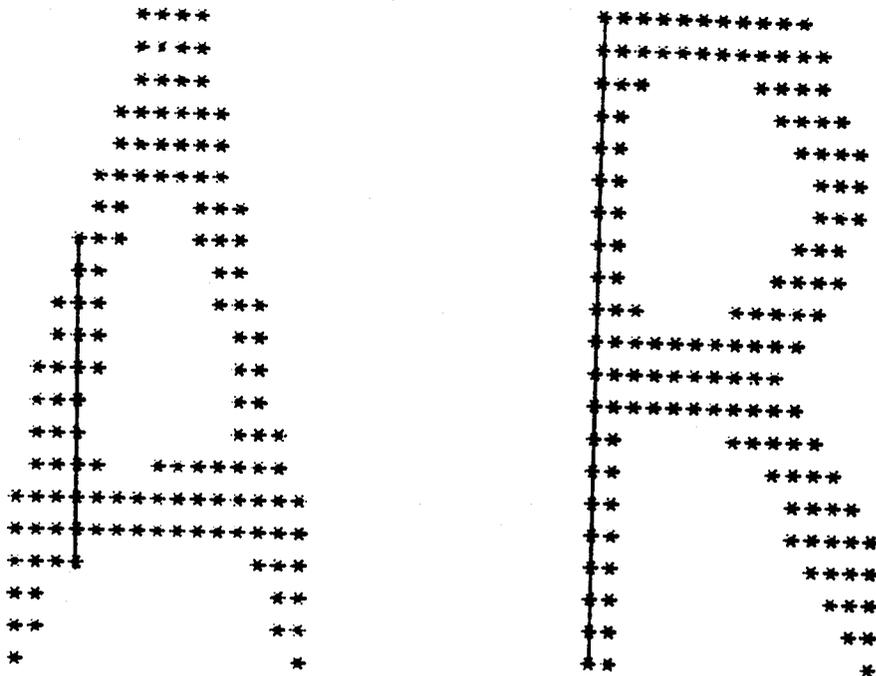


Figure 22. Classe (A, R)

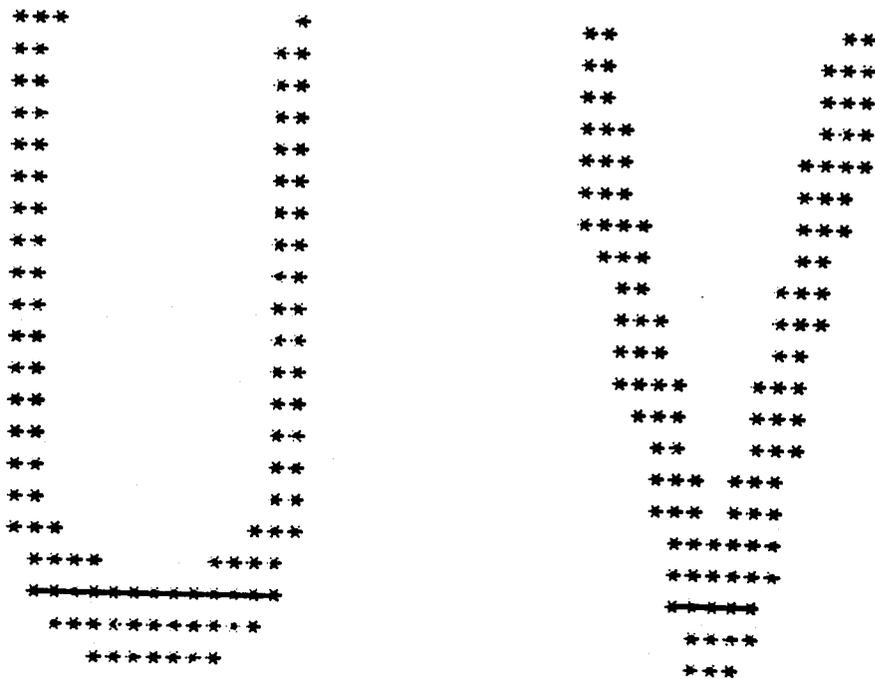


Figure 23. Classe (U, V)

Dans les deux cas cette opération est rapide puisque la projection n'est effectuée que sur une partie du caractère (les premières colonnes à gauche du caractère pour le premier cas et les dernières lignes en bas pour le deuxième cas).

Pour différencier les caractères D et O nous procédons à une projection verticale (sur un axe horizontal) du caractère. Si une dissymétrie à gauche est détectée et si le maximum de cette projection à gauche est sensiblement égal à la hauteur du caractère, celui-ci sera reconnu comme un D (Figure 24).

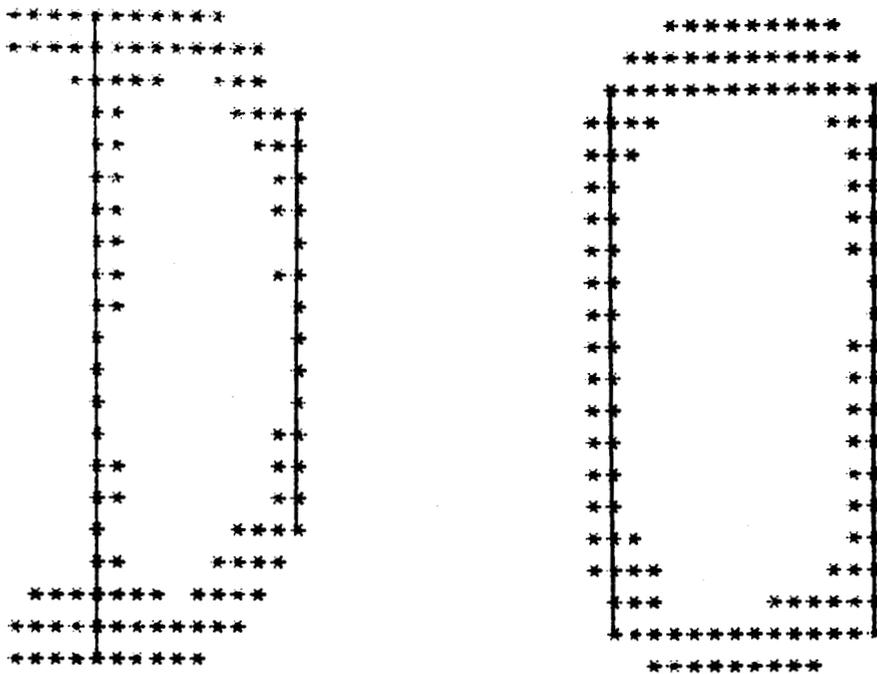


Figure 24. Classe (D, O)

Une deuxième famille de tests a pour fondement l'analyse des concavités suivant certaines directions parmi les quatre principales (dessus, dessous, gauche, droite). Cette méthode sera détaillée plus loin lorsqu'on parlera des méthodes rapides de reconnaissance.

Dans certains cas, nous ne nous intéresserons qu'au nombre de ces concavités (le W sera différencié du U et du V par la présence d'une concavité en bas du caractère), dans d'autres cas, des mesures seront effectuées sur cette concavité.

Par exemple, le caractère 5 peut être différencié du S par la présence d'un angle au milieu à gauche; nous étudions la concavité à gauche du caractère : si on détecte une pente importante (supérieure à la moitié de la largeur), le caractère sera reconnu comme un 5 (Figure 25).

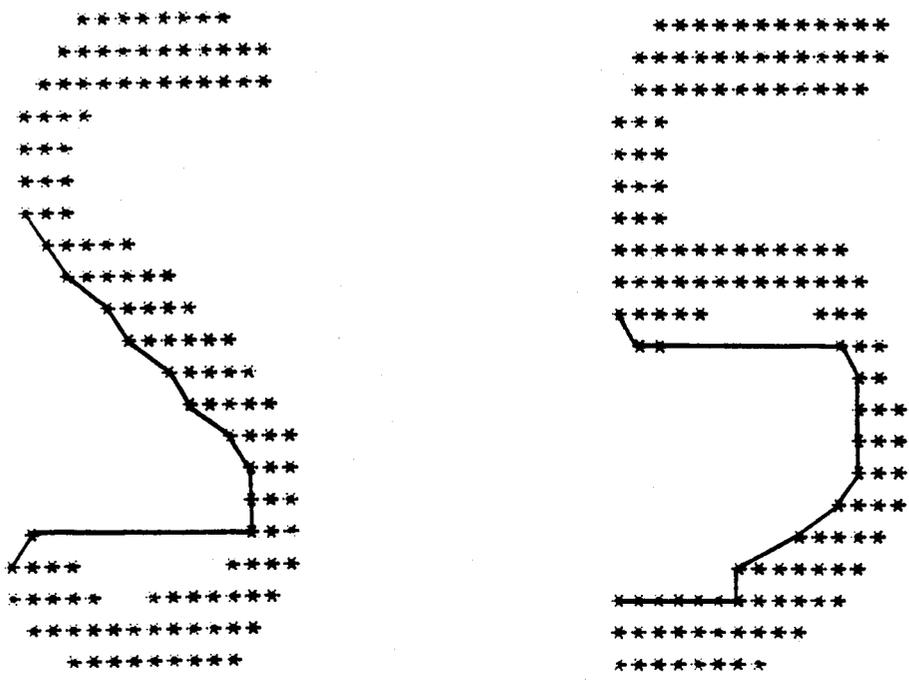
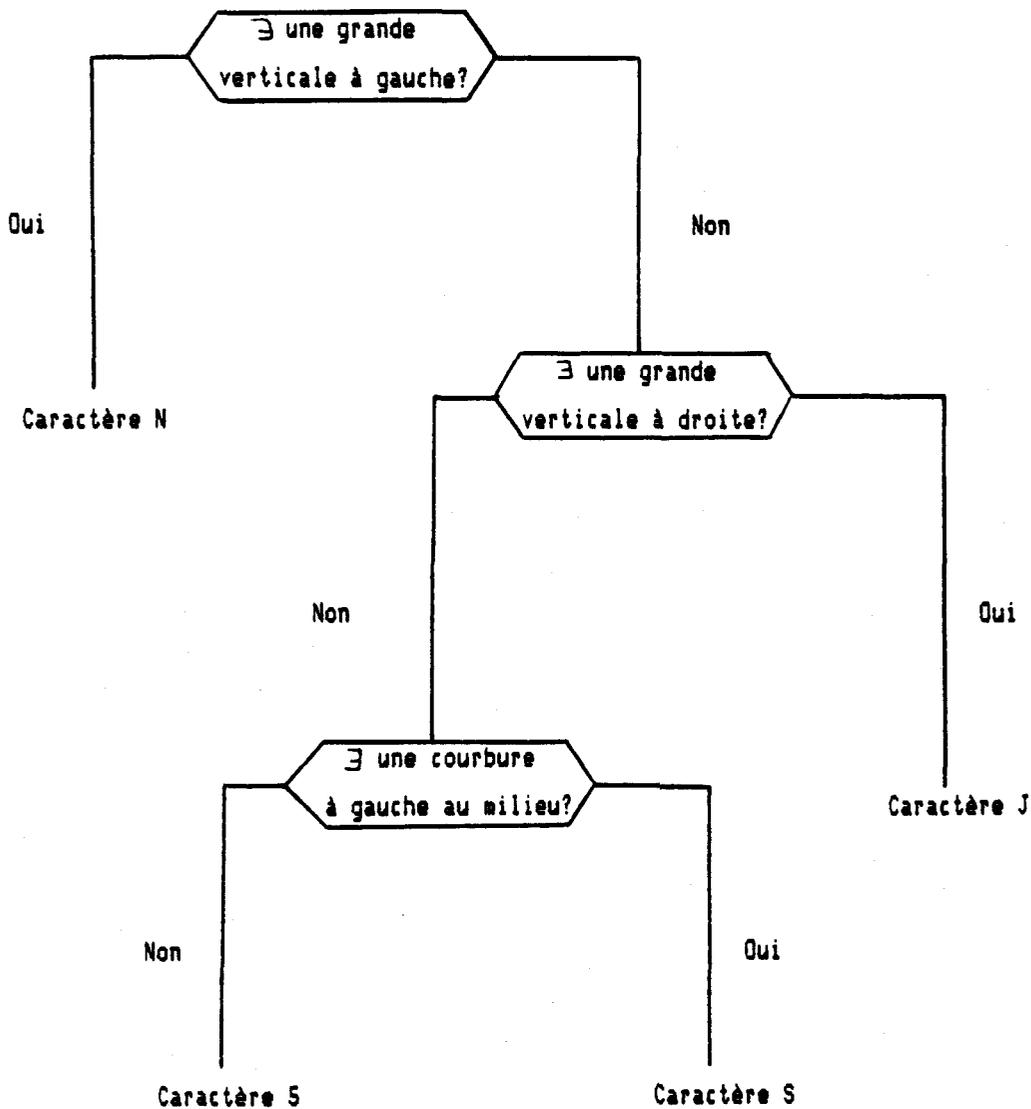


Figure 25. Classe (S, 5)

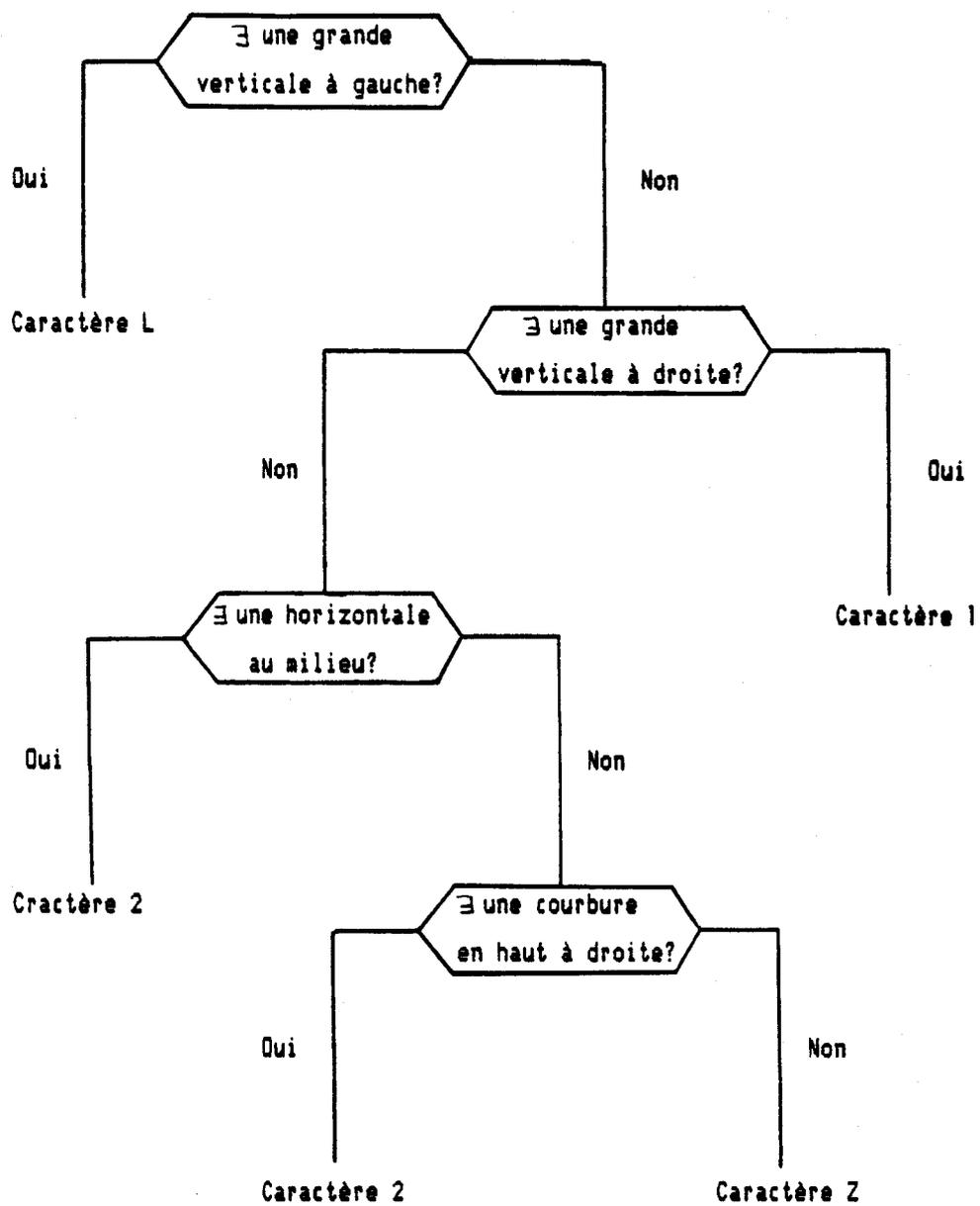
Lorsque nous sommes en présence d'une classe de population importante (3 ou 4), une classification arborescente est effectuée.

Exemple 1 : Cas de la classe (5, J, N, S)

Les deux premiers tests utilisés sont basés sur une projection verticale du caractère et le troisième sur l'examen de la concavité côté gauche du caractère.



Exemple 2 : Cas de la classe (L, Z, 1, 2)



Nous donnons ci-dessous un tableau récapitulant les classes obtenues après le premier niveau de décision et les tests utilisés pour la classification finale.

	Projection horiz.	Projection vert.	Concavités	Autres *
0,0,D,Q		-		-
1,2,L,Z	-	-		-
H,M,N,W				-
5,J,N,S		-	-	
K,X,W		-	-	
3,7	-			
9,Q				-
4,Q		-		
8,B		-		-
C,E	-			
T,Y	-			
A,R		-		
U,V,W	-		-	

* Autres:

- calcul d'angle
- comptage de points dans une partie du caractère
- position d'une extrémité par rapport à un noeud dans une des 9 zones

II-6 Résultats

Nous avons sélectionné pour l'apprentissage des fontes droites diverses d'épaisseur et de dimensions variables: Des fontes comportant des sérifs (exemple:Courrier) et d'autres pas (exemple:OCR). Nous avons effectué des essais sur des documents de qualité moyenne et dans des conditions moyennes d'éclairage (nous n'avons pas utilisé un dispositif d'éclairage).

Les fontes choisies pour l'apprentissage sont les suivantes:

OCR-A	ABCDEFGHIJKLMN OP QRSTUVWXYZ	1234567890
OCR-B	ABCDEFGHIJKLMN OP QRSTUVWXYZ	0123456789
1403 OCR modifie	ABCDEFGHIJKLMN OP QRSTUVWXYZ	0123456789
IBM 1403	ABCDEFGHIJKLMN OP QRSTUVWXYZ	1234567890
Farrington 12L2/12F2	ABCDEFGHIJKLMN OP QRSTUVWXYZ	1234567890

GOTHIC :
 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
 1 2 3 4 5 6 7 8 9 0

COURRIER :
 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
 1 2 3 4 5 6 7 8 9 0

ADVOCATE :
 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
 1 2 3 4 5 6 7 8 9 0

Le taux de reconnaissance par cette méthode générale est de l'ordre de 99,1% en estimant la lettre O et le chiffre 0 confondus (nous affichons 0 dans ce cas).

La vitesse de reconnaissance est de 25 caractères par seconde.

Cette méthode tolère une variation des dimensions du caractère à condition que celles-ci restent comprises dans un intervalle fixé par le logiciel de normalisation des dimensions.

Elle tolère aussi une variation d'épaisseur et une légère rotation du caractère. Nous donnons ci-dessous des caractères qui ont subi une rotation et qui ont été reconnus.

BCDEFGHIJKLMNOPQRSTUVWXYZ1234567890

CDEFGHIJKLMNOPQRSTUVWXYZ1234567890

La méthode de reconnaissance que nous avons développée est basée sur la position des points caractéristiques du squelette. C'est pourquoi elle n'est pas valable pour des caractères coupés puisque le squelette comporte des extrémités supplémentaires. Ce problème pourra être résolu si la discontinuité n'est pas très importante en effectuant un traitement approprié pendant la squelettisation (on pourra par exemple faire une dilatation après la première étape qui est l'obtention du noyau).

Cette méthode peut-être étendue à des fontes nouvelles. Deux cas se présentent:

1) Des codes nouveaux apparaissent. Si la classe associée au code ne contient qu'un seul caractère, ce

dernier est reconnu immédiatement. Dans le cas contraire, il est nécessaire de construire de nouveaux critères de séparation.

2) Le code existe déjà. Si la classe correspondante n'a pas été modifiée, on s'assure de la validité du critère de séparation et on affine dans le cas échéant. Dans le cas où la classe a été étendue, on complète le critère de séparation.

Dans les pages suivantes, nous donnons quelques résultats obtenus sur des caractères imprimés et manuscrits.

CHAPITRE III

STRUCTURE DE RECONNAISSANCE PLURI-PROCEDURES

III-1 METHODES RAPIDES DE RECONNAISSANCE

Dans le but de diminuer le temps de reconnaissance, nous avons adjoint deux méthodes très rapides, en comparaison avec la méthode générale; mais en contrepartie quelques caractères seulement sont reconnus par celles-ci. Le choix de ces méthodes a été guidé par la nécessité de décider de la classe d'appartenance du caractère le plus rapidement possible et sans substitution.

Ces méthodes seront appliquées sur le caractère lissé non squelettisé.

III-1-1 Première méthode : Recherche de concavités

Nous nous sommes inspirés des travaux de BENALI [23] déjà exposés au début de ce rapport. Nous n'avons retenu que le premier niveau de son approche : c'est à dire le dénombrement des concavités relatives à chacun des quatre côtés du rectangle englobant le caractère (nous n'avons pas dénombré les boucles).

Le dénombrement des concavités s'effectue par le suivi de la courbe de la fonction distance entre un côté du rectangle et le caractère. Une concavité est définie par une abscisse de début, une abscisse de fin et une amplitude.

Le début de concavité est obtenu en cherchant la première pente positive significative (non due au bruit).

Lorsque cette abscisse est trouvée, on continue le suivi pour déterminer une fin éventuelle de concavité donnée par la deuxième pente négative significative (figure 26). La différence entre la valeur maximale atteinte par la fonction distance entre ces deux abscisses et la plus grande des deux valeurs de cette fonction en ces deux points donne l'amplitude de la concavité. Pour éviter une substitution, nous avons été sévères dans le choix de l'amplitude (ou profondeur). Nous avons pris le tiers de la largeur du caractère.

Pour les caractères alphanumériques, le nombre de concavités correspondant à chacun des quatre côtés est de 0, 1 ou 2. Le nombre et la position de ces concavités seront donc codés sur 8 bits; 2 pour chaque côté.

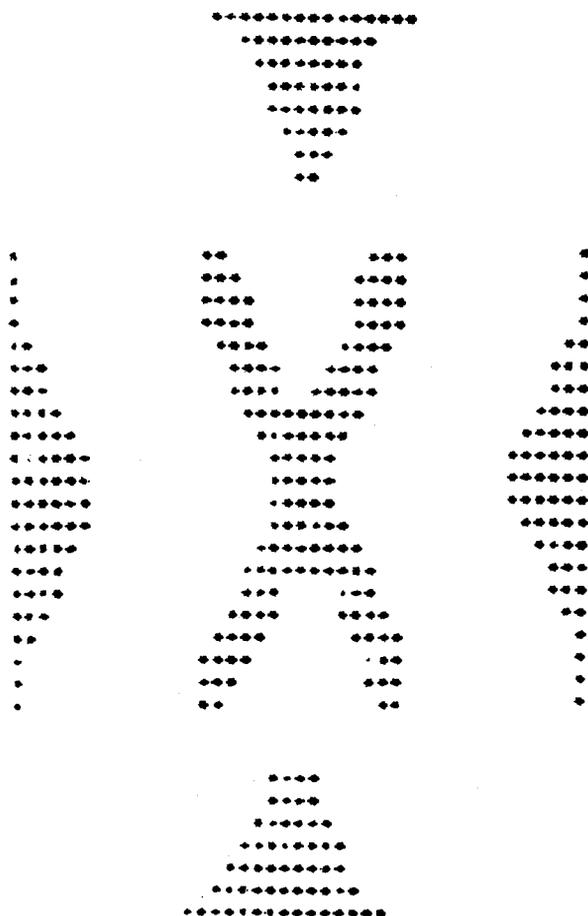


Figure 26. Les quatre concavités du caractère X

Nous nous sommes intentionnellement limités à ce niveau de reconnaissance. L'étape suivante, qui consisterait à faire des mesures supplémentaires décrivant avec une plus grande précision ces concavités (profondeur, largeur, position), a été volontairement ignorée, ce pour réduire au maximum le taux de substitution.

En examinant les caractères alphanumériques majuscules, on remarque que certains peuvent-être reconnus à la suite du dénombrement des concavités par rapport au rectangle englobant le caractère.

Ces caractères sont : E, K, M, R, W, X, 3.

Notons que ces caractères peuvent ne pas être reconnus (ceci dépend de la fonte); par exemple le caractère E de la figure 27 n'est pas reconnu à la suite de cette méthode puisque nous ne dénombrons qu'une seule concavité à droite. Le caractère sera donc rejeté par cette méthode. Il conviendra alors de le soumettre à une autre méthode de reconnaissance (cf. III-2).

Le temps de reconnaissance par cette méthode est d'environ 8 ms.

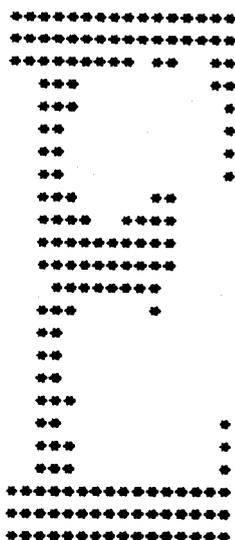


Figure 27. Caractère rejeté par la première méthode

III-1-2 Deuxième méthode : Projections horizontale et verticale

Les caractères alphanumériques majuscules latins sont essentiellement composés de segments horizontaux et de segments verticaux; c'est pourquoi nous nous sommes intéressés à la position des segments de longueur suffisante.

On procède à une projection partielle du caractère suivant les deux axes (horizontal et vertical): la projection est interrompue dès qu'une discontinuité est rencontrée (figure 28). La présence d'un segment est détectée quand le résultat de la projection est supérieur à un seuil bien choisi.

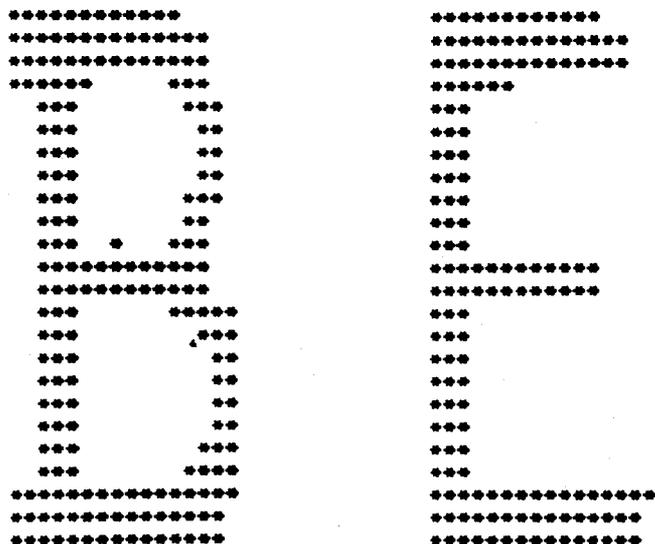


Figure 28. Projection horizontale partielle

PAKKER [20] a mené une étude statistique de la position de ces segments; cette étude l'a conduite à considérer trois zones pour les segments horizontaux et deux zones pour les segments verticaux (figure 29). La position

de ces segments a été codée respectivement sur trois bits et deux bits. Nous adoptons cette démarche dans notre étude.

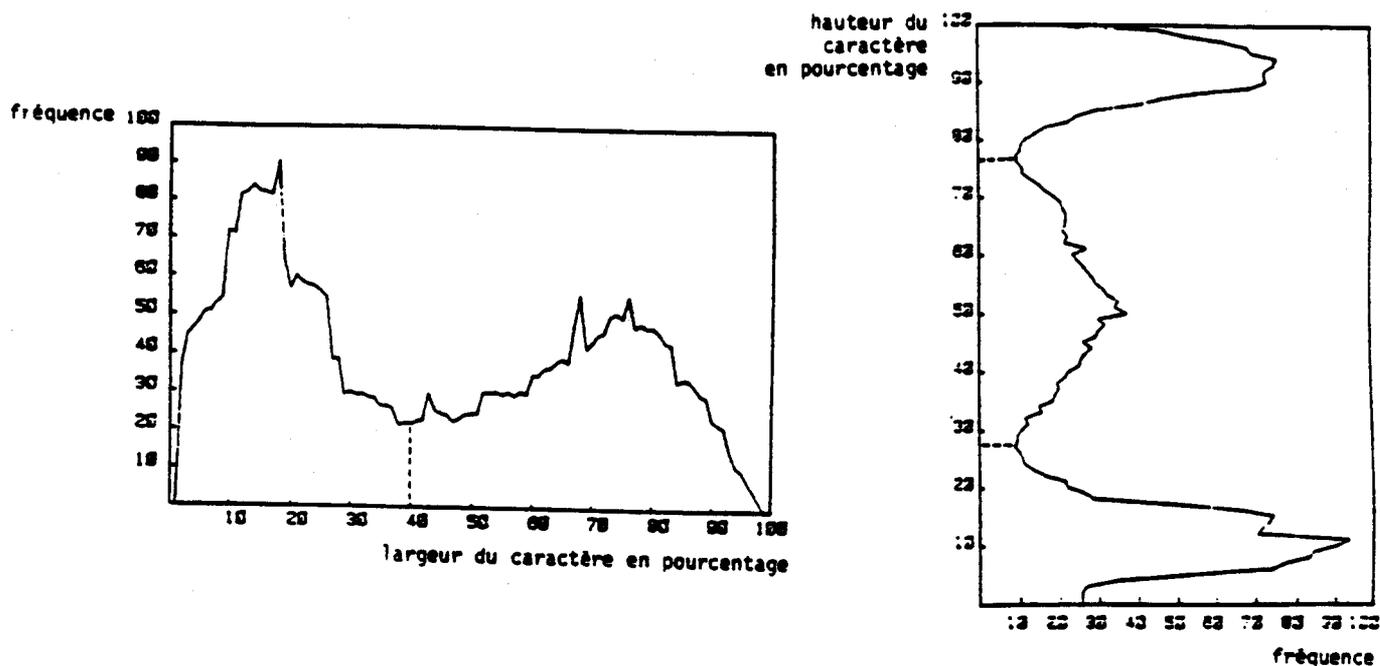


Figure 29. Etude statistique de la position des segments horizontaux et verticaux

Notre objectif demeure, comme dans le cas de la première méthode, de reconnaître rapidement et sans substitution quelques caractères. Pour cela, nous avons choisi un seuil (celui de la détection d'un segment) égal au $2/3$ de la hauteur pour les segments verticaux et à la moitié de la largeur pour les segments horizontaux. Ce choix a été dicté par le souci de discriminer au mieux les caractères.

Voici la liste des caractères reconnus pour la plupart des fontes dont nous disposons: J, L, T, 7.

Le temps de reconnaissance est de 4 ms environ.

III-2 STRUCTURE DE RECONNAISSANCE PROPOSEE [25]

Nous avons exposé une méthode de reconnaissance générale basée sur la squelettisation capable de reconnaître tous les caractères et deux méthodes plus rapides capables de reconnaître quelques caractères seulement. La structure de reconnaissance que nous proposons donnée par la figure 30 tient compte de la fréquence d'apparition des caractères. Cette fréquence estime la probabilité d'apparition ultérieure.

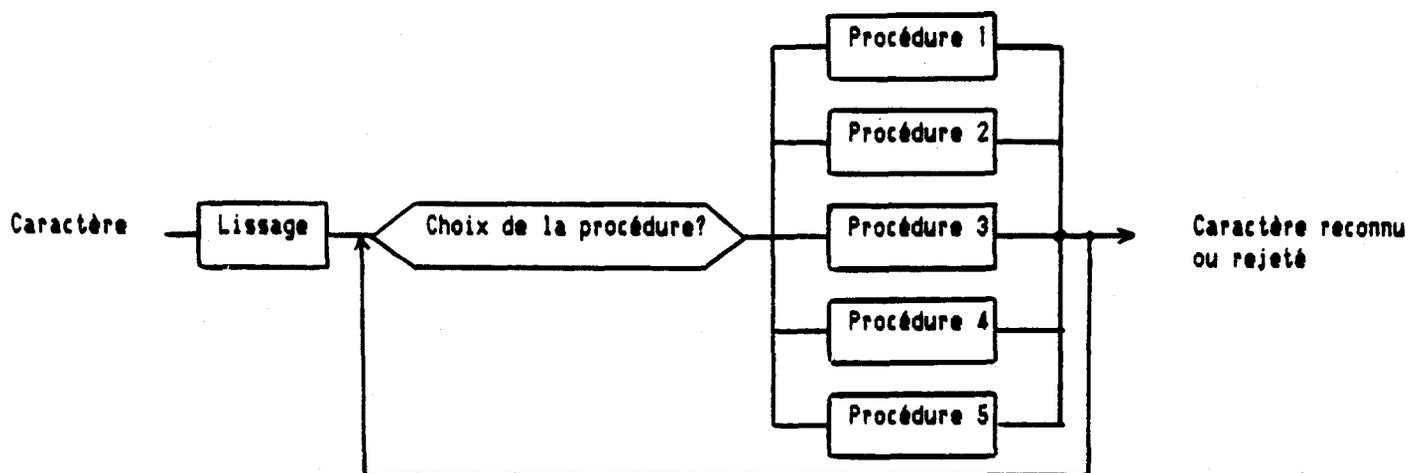


Figure 30. Structure de reconnaissance proposée

Nous avons le choix entre cinq procédures de reconnaissance.

P_1 : Méthode I + Méthode II + Méthode générale

P_2 : Méthode II + Méthode I + Méthode générale

P_3 : Méthode I + Méthode générale

P_4 : Méthode II + Méthode générale

P_5 : Méthode générale

La méthode I étant le dénombrement et la localisation des concavités dans les quatre directions privilégiées, la méthode II la recherche du nombre et de la position des segments horizontaux et verticaux de longueur significative et la méthode générale celle basée sur la position des noeuds et des extrémités du squelette.

Il est à noter que la méthode générale est présente dans toutes les procédures. En effet, si le caractère est rejeté par la ou les méthodes rapides, il sera examiné par la méthode générale.

Soient t_1 le temps de reconnaissance par la méthode I, t_2 le temps de reconnaissance par la méthode II et t_a le temps de reconnaissance par la méthode générale.

Soient n_1 le taux de caractères pouvant être reconnus par la méthode I, n_2 le taux de caractères pouvant être reconnus par la méthode II.

Nous calculons le temps moyen de reconnaissance correspondant à chacune des cinq procédures:

$$T_1 = n_1 * t_1 + n_2 * (t_1 + t_2) + (1 - n_1 - n_2) * (t_1 + t_2 + t_a)$$

$$T_2 = n_2 * t_2 + n_1 * (t_1 + t_2) + (1 - n_1 - n_2) * (t_1 + t_2 + t_a)$$

$$T_3 = n_1 * t_1 + (1 - n_1) * (t_1 + t_a)$$

$$T_4 = n_2 * t_2 + (1 - n_2) * (t_2 + t_a)$$

$$T_5 = t_a$$

La procédure choisie automatiquement est celle pour laquelle le temps de reconnaissance calculé est minimum.

Exemple: Dans le cas d'un texte littéraire en langue française, $n_1 = 27\%$ et $n_2 = 13\%$. La procédure de reconnaissance choisie est la première avec un gain d'environ 5ms par caractère, par rapport à la méthode générale seule (procédure P_s).

CONCLUSION

Nous avons contribué à la reconnaissance de caractères imprimés multiformes en élaborant une structure de reconnaissance tenant compte de la fréquence d'apparition des caractères dans le texte. Cette fréquence va déterminer le choix d'une procédure permettant de diminuer le temps moyen de reconnaissance.

Les logiciels d'acquisition, de binarisation et de normalisation nous ont été fournis par la société AI VISION

Pour la phase de prétraitement, nous avons développé un algorithme de squelettisation rapide conservant la connexité du caractère et assurant l'épaisseur unitaire du trait.

Nous avons développé ensuite une méthode générale de reconnaissance dont le premier niveau de décision basé sur la position des points caractéristiques (extrémités et noeuds) dans le squelette, se révèle très discriminant. Cette méthode a de bonnes performances pour l'imprimé multiforme (fontes droites). Le taux de reconnaissance est de 99,1% et la vitesse est de 25 caractères par seconde.

Dans le but de diminuer ce temps de reconnaissance, nous avons retenu puis adapté deux méthodes plus rapides capables de reconnaître quelques caractères seulement.

La première méthode utilise des projections horizontales et verticales du caractère et la deuxième l'examen des concavités par rapport aux quatre côtés du rectangle enveloppe du caractère. Seul le premier niveau de décision des deux méthodes a été retenu et ce, pour limiter au maximum le taux de substitution.

Cinq procédures de reconnaissance sont alors disponibles dans lesquelles la méthode générale est toujours présente. Nous choisissons, automatiquement, la procédure pour laquelle le temps moyen de reconnaissance est minimum.

Cette structure pluri-procédures nous permet un gain de temps appréciable dans le cas où les caractères reconnus par les méthodes rapides apparaissent fréquemment dans le texte à reconnaître tout en maintenant une bonne fiabilité grâce à la méthode générale. Nous pensons notamment au domaine de la productique dans lequel fiabilité et rapidité sont les deux critères principaux.

Dans un avenir proche, il est envisagé, pour diminuer encore le temps de reconnaissance de paralléliser notre algorithme de reconnaissance. Le parallélisme sera structuré comme suit:

- un processeur maître pour l'acquisition et la gestion des périphériques.

- un processeur esclave pour la reconnaissance

De plus, il serait intéressant d'étendre le domaine d'application et d'évaluer les performances de la structure de reconnaissance pour les minuscules imprimées puis pour le manuscrit guidé.

BIBLIOGRAPHIE

- [1] C. REMY
Lecture optique et reconnaissance de textes
Micro-systèmes, Octobre 1985

- [2] R. ROUBATY
ABC de cryptographie
Masson, 1984

- [3] J. C. SIMON
La reconnaissance des formes par algorithmes
Masson, 1984

- [4] A. FAURE
Perception et reconnaissance des formes
Editests, 1985

- [5] G. GAILLAT, M. BERTHOD
Panorama des techniques d'extraction de traits
caractéristiques en lecture optique des caractères
Congrès AFCET-IRIA, Reconnaissance des formes et
intelligence artificielle, Toulouse 1979

- [6] A. KINDER
Procédé de reconnaissance syntaxique de caractères
alphanumériques manuscrits. Réalisation micro-informatique.
Thèse de Docteur-Ingénieur, I. N. P. Grenoble, 1981

[71] **D. GOKANA**

Contribution à la reconnaissance automatique de caractères manuscrits. Application à la lecture optique de caractères sur supports mobiles.

Thèse de Doctorat. Université de Paris-Sud. Mars 86.

[81] **S. TOHME**

Prétraitement du chiffre manuscrit

Congrès AFCET-IRIA Reconnaissance des formes et traitement des images, Châtenay-Malabry, Février 1978

[91] **H. J. NACCACHE, R. SHINGHAL**

An investigation into the skeletonization approach of Hilditch

Pattern recognition, vol. 17, N°3, pp. 279-284, 1984

[101] **C. ARCELLI, G. SANNITI DI BAJA**

A width-independent fast thinning algorithm

IEEE Trans. on Pattern Recognition and Machine Intelligence, vol. 7, N°4, Juillet 1985

[111] **T. PAVLIDIS**

Algorithms for graphics and image processing

Computer Science Press, 1982

[121] **J. SERRA**

Image analysis and mathematical morphology

Academic Press, London, 1982

[131] **S. KAHAN, T. PAVLIDIS**

On the recognition of printed characters of any font and size

IEEE Transactions on pattern analysis and machine intelligence, vol. PAMI. 9, N°2, Mars 1987

[14] S. TOHME

Reconnaissance du chiffre manuscrit
Thèse de Docteur-Ingénieur, ENST, Octobre 1979

[15] B. CARCENAC de TORNE

Identification des adresses postales
Congrès AFCET-IRIA. Reconnaissance des formes et traitement
des images, pp.601-611, Châtenay-Malabry, Février 1978

[16] G. GAILLAT

Une procédure statistique de décision avec apprentissage et
son application à la reconnaissance de caractères
manuscrits.
Thèse de 3ème cycle, Paris VI, Juin 1975

[17] A. CHEHIKIAN, F. HADJ HASSAN

Une description structurelle, peu redondante, des caractères
alphanumériques, au moyen de variables binaires
Congrès AFCET-IRIA. Reconnaissance des formes et
intelligence artificielle, Toulouse 1979

[18] A. BELAID

Reconnaissance structurelle de caractères manuscrits et de
formules mathématiques
Thèse de 3ème cycle, Université de Nancy 1, 1979

[19] A. KINDER, G. BOUVIER, A. CHEHIKIAN

Codage des caractères alphanumériques par des chaînes de
caractères
Congrès AFCET-IRIA. Reconnaissance des formes et
intelligence artificielle, Toulouse 1979

[20] K. PAKKER

Reconnaissance des caractères alphanumériques par une
analyse structurelle hétérarchique
Thèse de Docteur-Ingénieur, I.N.P. Grenoble, Mars 1985

[21] T. VO VAN

Une méthode de reconnaissance de caractères (latins et chinois) non stylisés

Congrès AFCET-IRIA. Reconnaissance des formes et traitement des images, Châtenay-Malabry, Février 1978

[22] J. CONTER, M. VALOBRA, A. BRUEL

D.E.L.T.A. Un Dispositif Electronique de Lecture de Textes pour Aveugles

Congrès AFCET. Reconnaissance des formes et intelligence artificielle, Nancy, Septembre 1981

[23] M. BENALI

Du choix des mesures dans des procédures de reconnaissance des formes et d'analyse de texture

Thèse de Doctorat, Ecole Nationale Supérieure des Mines de Paris, Décembre 1986

[24] B. TACONET, F. YATIM, A. FAURE

Un nouvel algorithme de squelettisation rapide appliqué à la reconnaissance de caractères

2ème Atelier Scientifique: TIPI, Aussois, Avril 1988

[25] F. YATIM, B. TACONET, P. BIZET, A. FAURE

Système d'acquisition et de reconnaissance de caractères alphanumériques multifontes

Congrès AFCET Reconnaissance des formes et intelligence artificielle, Antibes, Novembre 1987

[26] J. L. JACQUOT, T. COURT, D. DAVID, A. PIRSON

Une classe d'extracteurs de contours performants adaptés à une implantation matérielle fonctionnant à cadence vidéo

2ème Atelier Scientifique: TIPI, Aussois, Avril 1988



[27] C. Y. SUEW, M. BERTHOD, S. MORI

*Automatic Recognition of Handprinted Characters-The State of
the Art*

Proceedings of the IEEE, vol. 68, N°4, Avril 1980

Nom : Farhat YATIM

Titre : Reconnaissance de caractères multifontes par une structure pluri-procédures

Résumé :

L'étude statistique d'un texte permet d'estimer la fréquence d'apparition des caractères dans le texte. Dans le but de réduire le temps de reconnaissance, nous proposons une structure pluri-procédures. Dans un premier temps, nous avons développé une méthode (générale) de reconnaissance multifonte, dont le premier niveau de décision est basé sur la position des points caractéristiques du squelette. Dans un second temps, pour accroître la vitesse de reconnaissance, nous avons retenu puis adapté deux méthodes plus rapides que la méthode générale mais en contrepartie quelques caractères seulement sont reconnus.

Cinq procédures, qui mettent en jeu la méthode générale couplée éventuellement à l'une ou à l'autre ou aux deux méthodes rapides, sont disponibles. La procédure, choisie automatiquement, est celle pour laquelle le temps moyen de reconnaissance est minimum. Ce dernier est fonction de la fréquence d'apparition des caractères dans le texte. Cette structure nous permet de réduire le temps moyen de reconnaissance tout en maintenant un bon taux.

Mots clés : Fréquence d'apparition - Reconnaissance multifonte - Squelette - Structure pluri-procédures.

Abstract :

A text statistic study allows to estimate the frequency of characters apparition in the text. In the goal to decrease the recognition time, we propose a pluri-procedures structure. Firstly, we have developed a general multifold recognition method in which the first level of decision is based upon the position of skeleton characteristic points. Secondly in order to increase the recognition speed, we have used two methods faster than the general method after their adapting to our case, but only some characters are recognized.

Five procedures are available. These procedures are determined by the general method which is eventually coupled to one or two of faster methods. The procedure is chosen automatically according their average recognition time. This time which is in function of frequency of characters apparition in the text has to be minimum.

This structure permits us to decrease the average recognition time by keeping a good rate of recognition.

Key words : Frequency of apparition - Multifold recognition - Skeleton - Pluri-procedure structure.