Procédé structurel de lecture optique multipolice

Nicolas Tripon

ECOLE NATIONALE
SUPERIEURE DES MINES
DE SAINT-ETIENNE

INSTITUT NATIONAL
POLYTECHNIQUE
DE GRENOBLE

N° d'ordre : 24 II

THESE

présentée par

Nicolas TRIPON

pour obtenir le grade de

DOCTEUR-INGENIEUR
"SYSTEMES ET RESEAUX INFORMATIQUES"

PROCEDE STRUCTUREL DE LECTURE OPTIQUE MULTIPOLICE

SOUTENUE A SAINT-ETIENNE LE 9 OCTOBRE 1981 DEVANT LA COMMISSION D'EXAMEN :

Président	**M. A.CHEHIKIAN**
Examinateurs	**MM. Ph.COUEIGNOUX**
	L.LUMBROSO
	R.MAHL

ECOLE NATIONALE
SUPERIEURE DES MINES
DE SAINT-ETIENNE

INSTITUT NATIONAL
POLYTECHNIQUE
DE GRENOBLE

N° d'ordre : 24 II

THESE

présentée par

Nicolas TRIPON

pour obtenir le grade de

DOCTEUR-INGENIEUR
"SYSTEMES ET RESEAUX INFORMATIQUES"

PROCEDE STRUCTUREL DE LECTURE OPTIQUE MULTIPOLICE

SOUTENUE A SAINT-ETIENNE LE 9 OCTOBRE 1981 DEVANT LA COMMISSION D'EXAMEN :

Président	M. A.CHEHIKIAN
Examinateurs	MM. Ph.COUEIGNOUX
	L.LUMBROSO
	R.MAHL

Je remercie Monsieur Alain CHEHIKIAN, Directeur du laboratoire du Signal Bioélectrique et de la Reconnaissance des Formes de l'ENSERG, pour avoir accepté de juger cette thèse et d'en présider le Jury.

Que Monsieur Philippe COUEIGNOUX, Maître de Recherche à l'ENSMSE, initiateur de cette étude, trouve ici le témoignage de ma profonde gratitude pour ses conseils et l'aide qu'il ne m'a jamais refusée.

Je tiens à remercier Monsieur Lucien LUMBROSO, responsable adjoint du département Téléinformatique à la SECRE pour l'intérêt qu'il a manifesté à l'égard de ce travail et pour avoir accepté de participer à ce Jury.

Je remercie Monsieur Robert MAHL, Directeur du secteur Energies et Matières Premières à la DGRST, qui, au cours de cette étude, s'est toujours intéressé à nos problèmes.

Je remercie Madame BONNEFOY pour l'excellent travail de frappe et de mise en page qu'elle a fourni et Messieurs BROSSARD, DARLE, LOUBET et VELAY pour la réalisation de cet ouvrage.

S O M M A I R E

I - INTRODUCTION

INTRODUCTION

Presque aussi ancienne que l'informatique elle-même la lecture optique automatique a maintenant plus de trente ans. Comme l'informatique, elle a connu une évolution constante : les systèmes lourds, peu flexibles et entièrement câblés du début laissent peu à peu la place à d'autres beaucoup plus légers, flexibles (orientés multipolice) et plus faciles à mettre en oeuvre (logique programmée). Deux facteurs en sont responsables.

- Le premier a trait aux algorithmes employés : les premiers algorithmes, inspirés des méthodes connues de détection des signaux radio, ont laissé la place à des algorithmes basés sur l'analyse des caractères ; la structure de ceux-ci variant peu d'une police à l'autre, les machines deviennent du coup multipolice.

- Le second est un facteur proprement technologique : l'avènement du microprocesseur. Il permet d'augmenter l'intelligence des algorithmes en les distribuant sur les différentes parties d'un même système. Les machines construites autour d'un seul processeur sont ainsi remplacées par d'autres, aussi rapides mais plus fiables et plus compactes, ayant une structure multiprocesseur.

D'un autre côté, la diffusion des terminaux de saisie directe couplés à des disques magnétiques étouffaient le développement de la lecture optique. Mais la banalisation récente des moyens d'impression d'apparence typographique, découlant elle aussi de l'évolution

technologique, suscite désormais un regain d'intérêt pour les techniques de saisie des textes multipolices. La traditionelle machine à écrire sera en effet de plus en plus supplantée par des systèmes d'impression d'apparence typographique se distinguant de la machine à écrire classique par la chasse variable des caractères et, sur les terminaux imprimant par points, la possibilité de varier les polices en taille et en nature. Offrir une telle flexibilité en impression sera efficace dès que la résolution atteindra 12 points digitaux au mm. A l'heure actuelle elle est couramment de 8 points au mm, mais l'évolution vers une plus grande qualité est confirmée. Il est significatif en ce sens qu'un grand fabricant de matériel de bureau comme XEROX cherche à avoir accès à un catalogue de polices typographiques comme celui de MONOTYPE.

Le travail présenté ci-après espère contribuer à ces développements en proposant un système de lecture optique multipolice, structurel, basé sur un ensemble d'algorithmes nouveaux, validés par simulation sur un mini-ordinateur et destinés à être implantés sur un système à microprocesseurs en pipe-line.

Le lecteur trouvera d'abord une description succincte des approches possibles en lecture optique dans le chapitre II. Après avoir justifié le choix en faveur d'une reconnaissance structurelle, la présentation détaillée de la méthode commence par l'introduction des primitives principales et secondaires à partir desquelles tout caractère typographique peut être décrit.

La reconnaissance des primitives dans la ligne de texte fait l'objet du chapitre III. On y trouvera notamment une méthode de

réduction de la redondance verticale et d'exploitation de la structure horizontale d'une ligne de texte en vue de l'extraction des primitives principales. Les choix qui ont dû être faits à ce niveau se justifient par le but recherché : identification des primitives principales sans segmentation a priori en caractères.

Le chapitre IV traite deux problèmes supposés résolus dans le chapitre III. D'une part l'histogramme vertical de la page de texte permet de localiser les lignes de texte, solution classique, mais aussi de réduire la redondance verticale de chaque ligne de texte. D'autre part on traite l'apprentissage automatique d'une police inconnue : un modèle grossier des primitives, obtenu sur la base d'une description a priori en terme d'attributs fonctionnels, est fourni à l'étage d'extraction des primitives ; les paramètres de départ sont remplacés par d'autres directement mesurés sur l'image après identification des primitives qui se rapprochent le plus du modèle grossier. Une première lecture, lente et incomplète, permet ainsi d'effectuer la lecture définitive dont il est question au chapitre III.

La recombinaison des primitives pour reconstruire le texte de départ fait l'objet du chapitre V. Pour cela, une procédure de segmentation dynamique de la chaîne de primitives en caractères, suivie d'une synthèse des primitives de chaque segment selon les règles de production d'une grammaire probabiliste sont mises en oeuvre. Parmi toutes les segmentations essayées, celle qui représente véritablement le texte sera reconnue comme étant le chemin ayant un coût minimal dans un graphe orienté.

Une évaluation qualitative est proposée au lecteur lors du chapitre VI. On y trouvera aussi une synthèse des informations contenues dans les chapitres précédents, ainsi que des remarques sur le travail restant à faire pour améliorer les performances du système.

II - METHODES STRUCTURELLES EN LECTURE OPTIQUE

A - INTRODUCTION

B - TABLEAU DE LA LECTURE OPTIQUE

1 - Méthodes fonctionnelles

2 - Méthodes structurelles

3 - Justification des méthodes structurelles

C - CODAGE DES CARACTERES ET APPLICATION A LA LECTURE OPTIQUE

1 - Codage des caractères

2 - Application à la lecture optique

3 - Avantages

II

METHODES STRUCTURELLES EN LECTURE OPTIQUE

A - INTRODUCTION

Ce chapitre essaie de replacer notre travail dans le cadre des systèmes de lecture optique automatique.

Pour cela un tableau de la lecture optique sera esquissé dans la première partie ; il est destiné à décrire et comparer les deux méthodes de décision en lecture optique et à justifier le choix que nous avons été amenés à faire.

La deuxième partie est consacrée au codage de la forme géométrique du caractère afin d'obtenir une représentation plus facilement exploitable avec des moyens informatiques. Ce problème est bien connu des graveurs pour l'importance qu'il prend dans tout système moderne de photocomposition ; l'analyse de leur travail dans ce domaine nous a permis d'établir un codage répondant aux besoins de notre application. Les avantages de ce codage seront discutés en dernière partie.

B - TABLEAU DE LA LECTURE OPTIQUE

Les procédés mis en oeuvre pour la lecture optique de caractères peuvent être classés, au risque d'une certaine schématisation, en deux grandes familles :

- famille des méthodes non structurelles ou fonctionnelles,
- famille des méthodes structurelles.

1 - Méthodes non structurelles

Cette famille est aussi la plus ancienne. Les méthodes qui en font partie sont basées sur l'extraction d'un certain nombre de caractéristiques par des mesures appropriées, pratiquées sur la forme graphique du caractère.

Ces mesures transforment la représentation graphique du caractère en un ensemble de nombres que l'on peut concevoir comme un point dans un espace (espace des caractéristiques) de dimension égale au nombre des mesures. Dans cet espace tout caractère doit être représenté par un nuage de points isolés que l'on appellera classe et auquel on donnera le nom du caractère représenté.

Deux impératifs sont à prendre en considération lors du choix des caractéristiques :

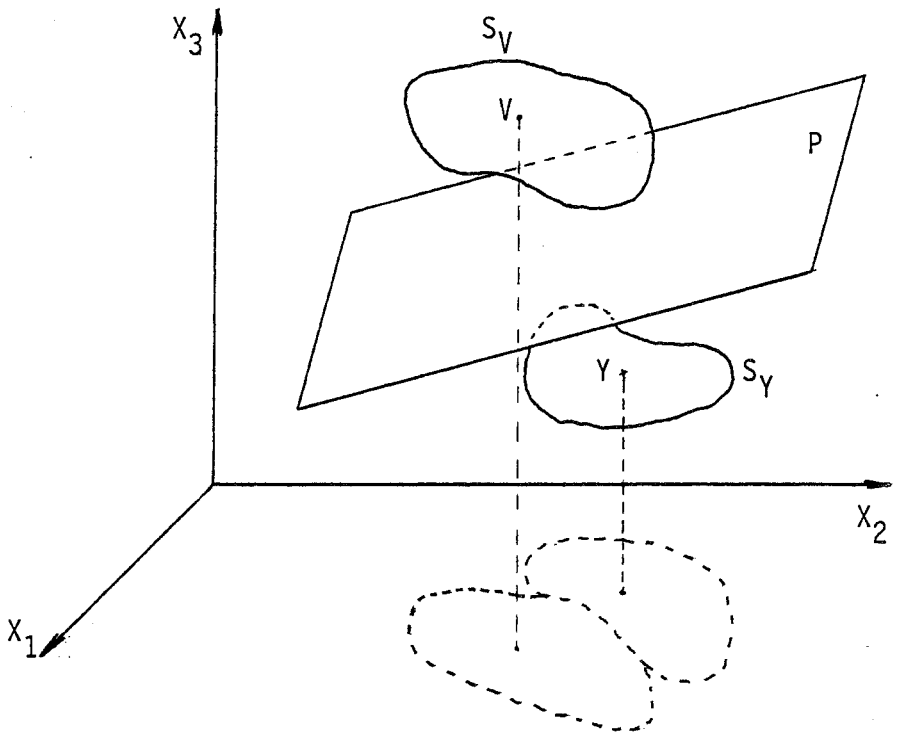
- deux classes distinctes ne doivent pas se recouvrir sous peine de confusion entre caractères différents,
- une règle de séparation des classes est nécessaire afin d'assigner sans ambiguïté **un point** (donc un caractère) à **une classe**

Dans le cas où les classes sont isolées la règle de séparation est l'équation de la surface de chaque classe.

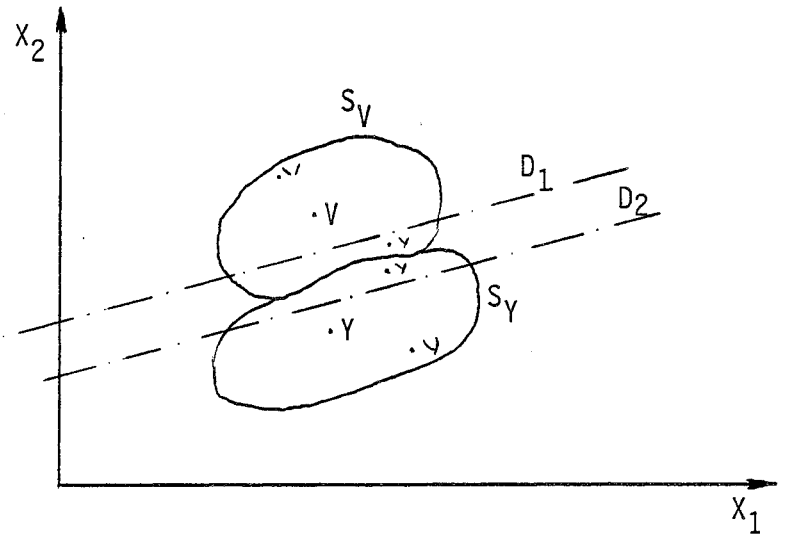
Exemple : Soit le cas à deux classes V et Y et l'espace R^3 des caractéristiques x_1, x_2 et x_3 permettant une classification non ambiguë. Dans le cas de la figure 1.a la surface S_V sépare le caractère V de tout caractère non-V (le prototype du caractère V se trouve au centre de la classe ; plus on s'approche des bords plus le caractère correspondant est loin du prototype) ; il en va de même pour Y. Précisons que dans ce cas simple une autre règle de séparation, le plan P, méthode linéaire beaucoup plus simple à mettre en oeuvre, permet d'arriver au même résultat.

Il est évident que la règle de séparation est d'autant plus simple qu'il y a moins de caractéristiques, mais dans ce cas le risque de recouvrement de deux classes est accru. Si l'on élimine x_3 dans l'exemple donné plus haut, on réduit la dimension de l'espace des caractéristiques mais on peut être amené à des cas analogues à celui de la figure 1-b où il est impossible de trouver une règle simple de séparation entre les classes ; les points situés d'un côté et d'autre et assez près de la frontière commune ont beaucoup de chance d'être mal classés.

Pour éviter cette difficulté, la dimension de l'espace des caractéristiques est volontairement agrandie pour augmenter les chances de disposer de l'information nécessaire à la décision (il n'y a aucun moyen de savoir à la conception du système si les caractéristiques choisies sont suffisantes pour le classement ; dans le cas négatif on en rajoute à la phase de test).



a) L'espace R^3 des caractéristiques : les classes sont bien séparées.



b) L'espace R^2 des caractéristiques : il y a risque de classement erroné pour les points compris entre les droites D_1 et D_2.

Exemple à deux classes : V et Y.

Figure 1

Ces méthodes ne sont pas propres à la lecture optique ; elles sont plutôt des méthodes générales de reconnaissance des formes et les mesures pratiquées sont neutres vis à vis de la structure intime des caractères. Le lecteur peut consulter avec profit l'article bibliographique de G. Gaillat et M. Berthod /2.1/ qui propose une classification en fonction de la notion physique exploitée. Ces méthodes sont très performantes pour un système monopolice (dans ce cas on peut trouver assez facilement un ensemble de caractéristiques qui transforment les signes de l'alphabet en classes bien distinctes), mais presque inexploitables pour des systèmes multipolice à cause de leurs performances médiocres. Les variations dans la forme des caractères d'une police à l'autre rendent la séparation des classes moins évidente, voire impossible ; des caractéristiques supplémentaires doivent être introduites, ce qui a pour effet de compliquer la règle de décision.

Jusqu'ici les caractéristiques ont été trouvées en comparant les caractères différents de bonne qualité ; nous avons aussi précisé que des difficultés apparaissent pour des caractères de moins bonne qualité, comme ceux compris entre les droites D1 et D2 de la figure 1.b. Une démarche originale a été définie par Blesser et al./2.2, 2.3, 2.4/ ; elle consiste en l'analyse des caractères ambigus se trouvant à proximité de la surface de séparation entre deux classes dans le but de définir des caractéristiques nouvelles, appelées **attributs fonctionnels** , capables de faire un classement des cas difficiles et, à plus forte raison, des cas normaux.

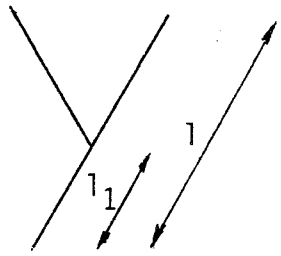
L'exemple de la figure 2a sert à expliciter cette notion. L'attribut fonctionnel analysé s'appelle jambage ; son existence est définie en fonction de la valeur du rapport l_1/l où l_1 est la longueur

	V	y	y
jambage physique	0	1	1
jambage fonctionnel	0	0	1
classe	V	V	Y

0 - absent

1 - présent

a)



Le jambage fonctionnel existe si $l_1/l \geq 0{,}17$
(valeur obtenue à la suite de tests psychologiques).

b)

Attributs fonctionnels et caractéristiques.

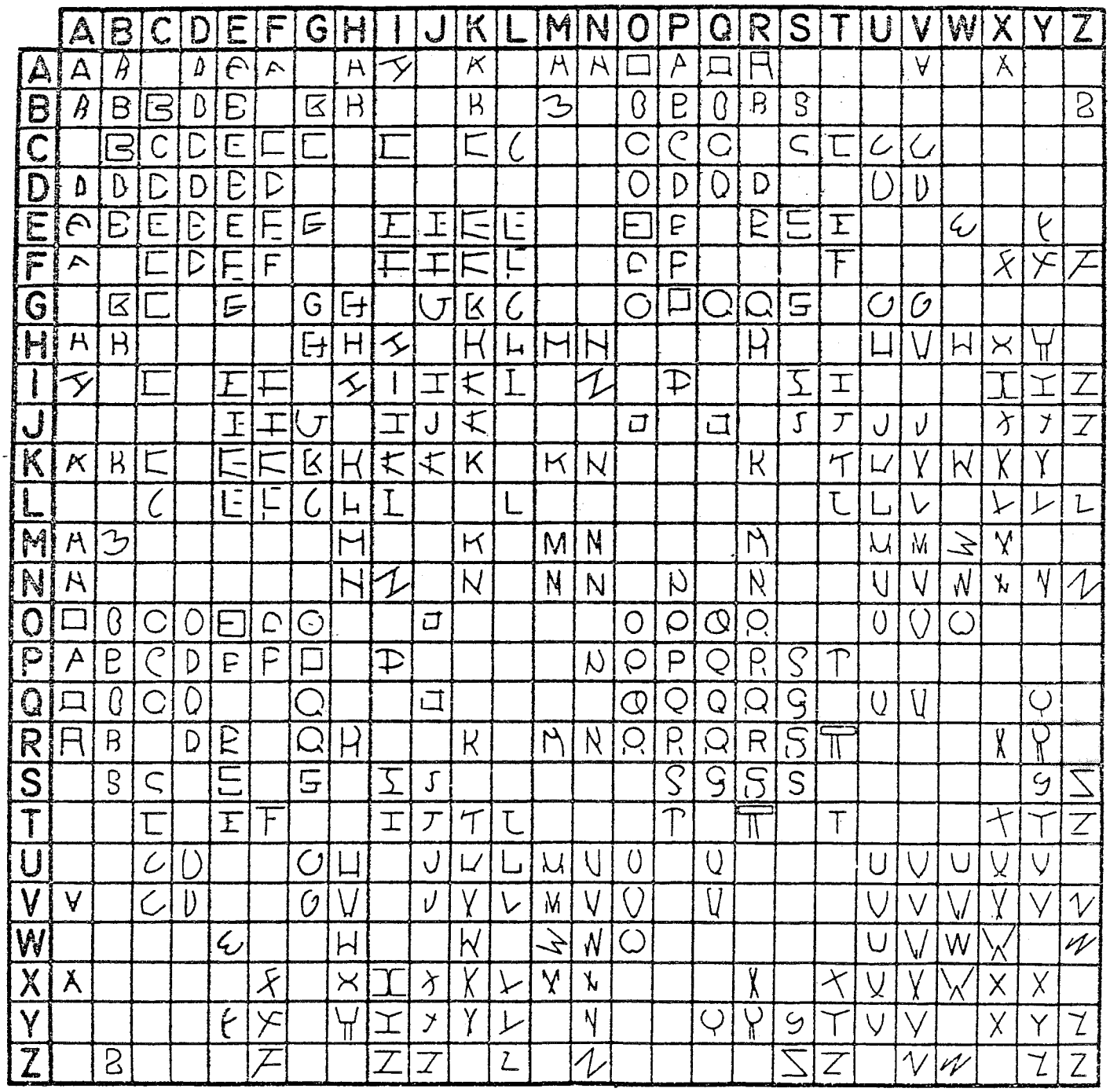
Figure 2.

du jambage physique, l étant la longueur totale de la barre oblique (voir figure 2.b). La valeur du seuil de décision a été obtenue à la suite de tests psychologiques ; pour l'attribut fonctionnel analysé elle est de 0.17 environ. En dessous de cette valeur une majorité de personnes interrogées ont classé le caractère de la figure 2.b comme V ; au-delà de ce seuil il a été classé comme étant un Y par les mêmes sujets.

Les avantages d'une telle démarche découlent de la manière dont les attributs fonctionnels ont été définis ; contrairement aux caractéristiques, dont le nombre était ajusté à la phase d'essais, les attributs fonctionnels peuvent être parfaitement définis à la phase de conception. Ainsi, les auteurs indiquent que 7 attributs fonctionnels seulement sont nécessaires pour décider dans 90 % des 180 cas ambigus de la figure 3.

2 - **Méthodes structurelles**

Une analyse attentive des signes d'un alphabet met en évidence un nombre réduit de parties ayant une géométrie identique et que nous appellerons primitives par la suite ; la figure 4 en donne un exemple : la barre verticale apparaît dans tous les signes représentés. L'ensemble des signes de l'alphabet peut être construit à partir d'une base de primitives, un caractère apparaissant sous la forme d'une structure reliant des primitives selon quelques règles bien définies. Des règles analogues ont été étudiées par la théorie des langages dont le but est l'analyse de la structure d'une phrase composée à partir d'éléments constitutifs élémentaires (mots). Nous voyons donc



Matrice d'ambiguïtés

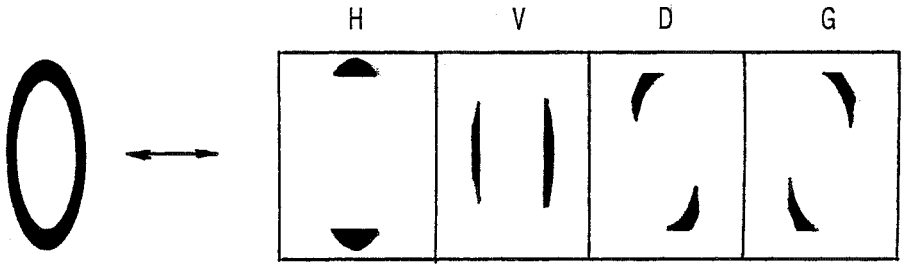
(tiré de /2.2/)

Figure 3

E L F T B D

Ces caractères sont construits autour d'une primitive verticale |.

Figure 4



Décomposition du caractère O en primitives selon la méthode proposée par Chehikian.

Figure 5

se dessiner la démarche à suivre en lecture automatique : analyse du texte pour mettre en évidence les primitives, suivie par une synthèse de celles-ci selon des règles analogues aux grammaires de la théorie des langages.

Le choix des primitives n'est pas rigide. Voici un exemple différent du nôtre : Chehikian /2.5, 2.6, 2.7/ bâtit son lecteur autour de quatre familles de primitives : horizontales (H), verticales (V), obliques de pente positive (D), obliques de pente négative (G). L'extraction des primitives se fait en classant chaque point du caractère inconnu dans une des quatre familles selon une procédure de décision basée sur l'orientation du segment dans le voisinage du point analysé. Ainsi le caractère O se décompose comme indiqué sur la figure 5. La reconstitution du caractère est opérée par une grammaire chargée de faire la synthèse des quatre familles de primitives.

3 - Justification des méthodes structurelles

L'écriture est un signal conventionnel ; elle a été et restera pour encore quelque temps le premier moyen de transmission et de stockage de l'information. De par son universalité, l'écriture a été normalisée afin de répondre à des besoins esthétiques ainsi que de rapidité et aisance de lecture. Cette standardisation lui a conféré une structure variant peu d'une police à l'autre. Une méthode structurelle, basée sur un choix judicieux des primitives, peut englober les différences entre les différents alphabets et devenir multipolice.

Dans le passé, plusieurs tentatives ont été faites pour normaliser davantage l'écriture afin de répondre aux exigences aiguës des premiers lecteurs optiques. Nous pouvons ainsi mentionner la norme

OCR-A, dépourvue de toute esthétique et difficile pour les humains, et la norme OCR-B, venue pour remplacer la première et combinant les exigences esthétiques, propres à l'homme, et techniques, propres à la machine de lecture. Dans l'état actuel de la lecture optique, ces normes sont devenues inutiles à cause du perfectionnement des machines, capables d'identifier les types courants de caractères d'imprimerie. La figure 6 montre les normes OCR-A, OCR-B et deux autres polices courantes d'imprimerie.

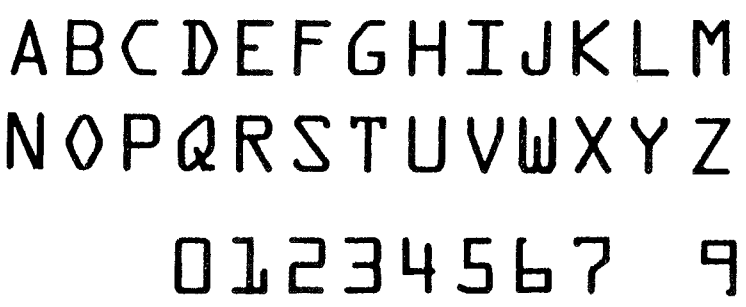
La standardisation dont nous avons fait état met en évidence une liaison entre les deux familles de méthodes de reconnaissance discutées. En effet, les attributs fonctionnels se trouvent maintenant inclus de façon définitive dans la structure du caractère et peuvent être retrouvés en décodant cette structure.

Pour toutes ces raisons, le choix fait est évident : nous bâtissons notre lecteur sur une méthode structurelle dont les primitives sont choisies de manière à refléter le mieux la structure du caractère.

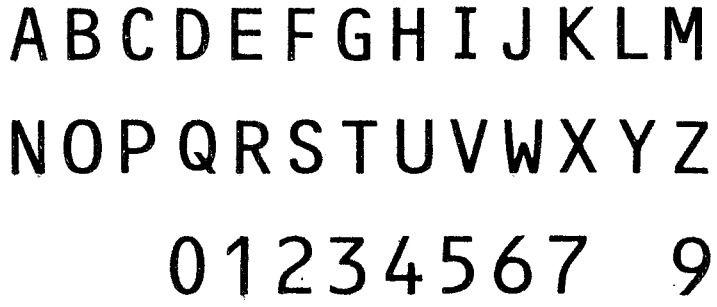
C - CODAGE DES CARACTERES ET APPLICATION A LA LECTURE OPTIQUE

1 - Codage des caractères

Pour cela, nous avons d'abord besoin d'approfondir nos connaissances quant à la structure des caractères. Nous nous intéresserons donc au codage employé par ceux qui ont généré le caractère, les graveurs. Le lecteur qui veut acquérir une vue d'ensemble des techniques employées doit consulter Coueignoux /2.8/.



OCR-A



OCR-B

Dexterity in the vocation of typesetting may be acquired by every judicious and zealous worker
Becoming expert in the vocation of typesetting
ABCDEFGHIJKLMNOPQRSTUVWXYZ*AI*

TIMES BOLD 334

Razor-sharp knives are required
Six reams of quad crown paper
ABCDEFGHIJKLMNOPQRSTU

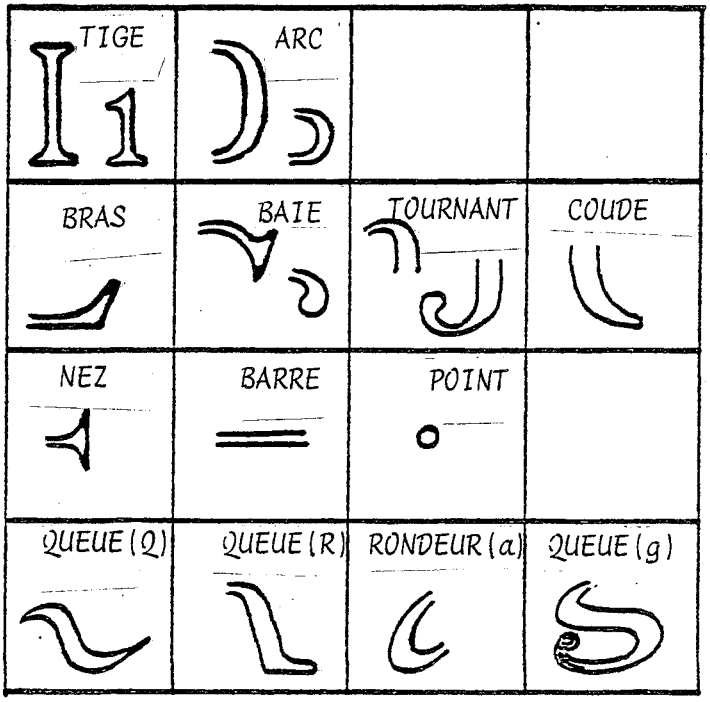
UNIVERS BOLD 693

Figure 6

Nous éliminons de notre discussion les techniques générales de codage par plage et par contour parce qu'elles ne prennent pas en compte la structure du caractère, mais nous retiendrons et discuterons dans la suite le codage structurel parce qu'il fournit un code intrinsèquement lié à la structure du caractère.

A la base du codage se trouve l'observation déjà faite lors de la présentation des méthodes structurelles : les caractères d'un alphabet sont constitués d'un nombre restreint de blocs géométriquement identiques. Le codage structurel spécifie le caractère sous la forme d'une liste de primitives, à chaque primitive étant associée une liste de paramètres indiquant sa position relative dans le caractère, l'échelle, la graisse. Pour les besoins de la génération Coueignoux /2.9/ trouve nécessaires et suffisantes 13 primitives ou familles de primitives (figure 7). Une famille de primitives est composée de formes étroitement liées mais non identiques ; elle dépend donc d'un certain nombre de paramètres (hauteur, pente, épaisseur) à l'aide desquels toute primitive de la famille peut être définie de manière complète. La figure 8 donne des exemples de primitives dans le contexte de quelques caractères. Les lecteurs intéressés par la potocomposition digitale trouveront dans /2.9/ tous les détails sur ces primitives, ainsi qu'une grammaire générative complète les mettant en oeuvre.

On peut considérer le caractère comme étant composé de deux classes d'éléments. La première classe comprend tous les éléments nécessaires à l'identification précise de la lettre représentée et qui constitue son squelette ; la deuxième classe est constituée de tous



Primitives de génération
(tiré de /2.9/).

Figure 7

Exemples de primitives dans leur contexte
(tiré de /2.9/).

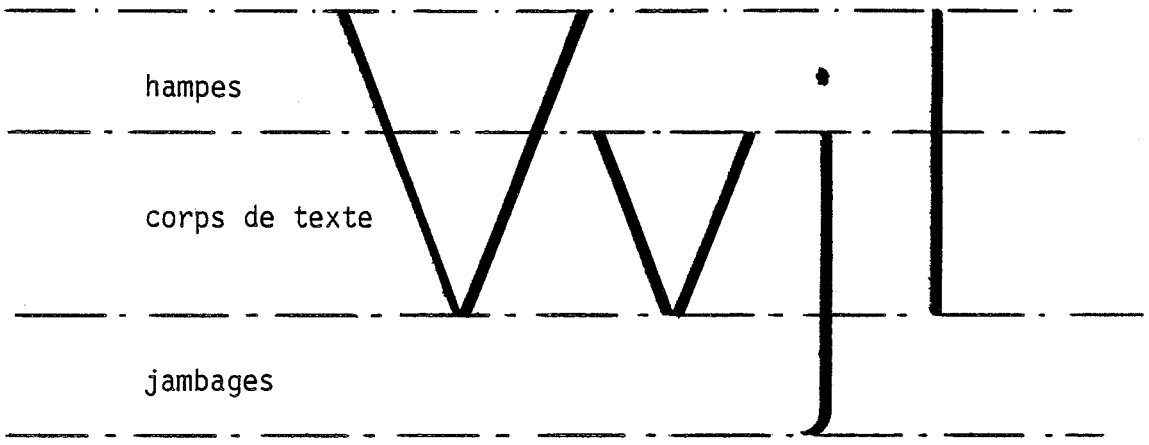
Figure 8

les éléments dont on pourrait se dispenser sans pour autant affecter l'identité de la lettre et qui ne sont là que pour donner un aspect esthétique au caractère. Pour la génération des caractères, les deux classes ont la même importance, le squelette et les embellissements étant propres à chaque police de caractères ; pour cette raison les primitives de génération ont été définies pour tenir compte autant du squelette que des embellissements.

Pour la lecture optique la seule classe indispensable est la première. C'est la raison pour laquelle l'ensemble des primitives de génération ne peut pas être employé comme tel en lecture automatique. Nous définirons dans la suite un ensemble de primitives pour la lecture issu de l'ensemble de génération en imposant une hiérarchie aux primitives de ce dernier, due à l'importance des primitives dans le squelette, les primitives d'embellissement étant éliminées.

2 - Application à la lecture optique

Trois zones peuvent être distinguées dans le sens de la hauteur d'un caractère : le corps de texte, hampes et jambages (figure 9). Parmi elles, la seule commune à tous les caractères est le corps de texte ; c'est la raison pour laquelle nous définissons les primitives sur le corps de texte seulement (avec la possibilité de vérifier leur prolongement a posteriori). Nous ne garderons donc que les deux premières primitives de celles représentées à la figure 7. On remarque qu'il s'agit en fait de familles de primitives puisque la primitive **tige** englobe des formes aussi différentes que \, | ,/ (dépendant de la pente d'inclinaison), alors que la primitive **arc** peut



Structure verticale d'une ligne de texte.

Figure 9

prendre les formes (et). Pour nous affranchir entièrement des paramètres nous abandonnons la notion de famille de primitives et définissons chaque variante comme primitive indépendante.

Une analyse de l'alphabet menée dans le but de définir les primitives de la famille **tige** montre qu'il y a quatre pentes possibles ; quatre primitives ont été donc définies :

primitive 1 : \ pente de - 20° environ

primitive 2 : / pente de 20° environ

primitive 5 : | pente nulle

primitive 7 : / pente comprise entre 20° et 45° environ

De la même manière la famille **arc** est scindée en deux :

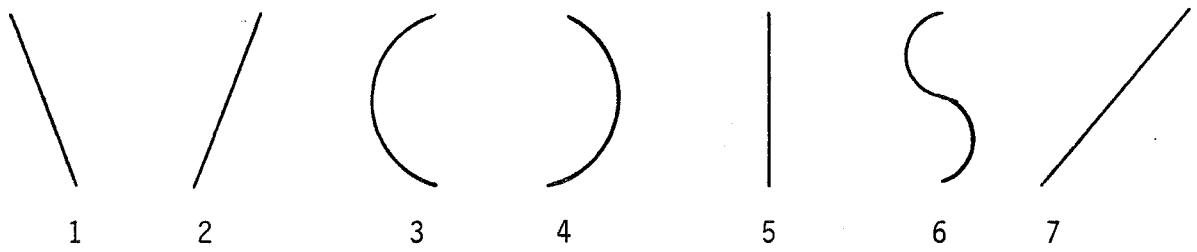
primitive 3 : (

primitive 4 :)

La primitive 7 diffère des autres dans le sens qu'elle n'apparaît que dans deux caractères : Z et z ; il s'agit d'une primitive spécialisée. Une autre primitive spécialisée est la primitive 6 ; celle-ci apparaissant dans seul le s minuscule (figure 10).

Les représentations graphiques données indiquent en fait le squelette de la primitive ; la primitive elle-même s'obtient en donnant à ce squelette une épaisseur égale à la graisse moyenne de la police.

Ces primitives, représentées à la figure 10, seront désormais appelées primitives principales parce qu'elles forment l'ossature des caractères : n'importe quel caractère alphabétique en

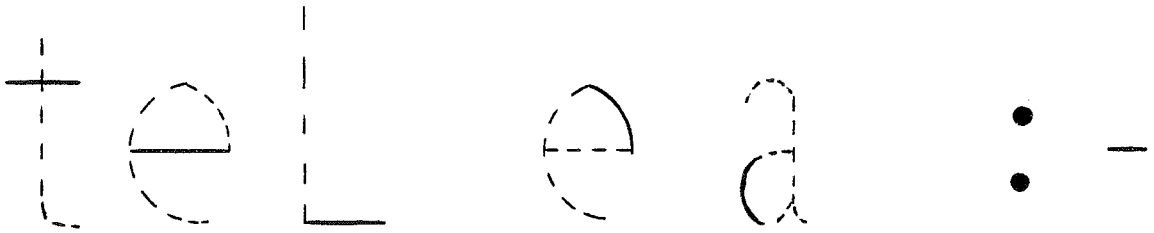


Primitives principales.

Figure 10

Importance des primitives secondaires pour la classification.
(heurtoir et barre).

Figure 11



a) barres b) heurtoirs c) points

Primitives secondaires.

Figure 12

contient au minimum une et au maximum quatre, mais elles ne sont pas suffisantes pour la tâche de reconnaissance. Il suffit pour s'en apercevoir de comparer les caractères e et c minuscules : ils sont tous les deux composés de la même primitive principale 3 mais la différence entre eux est donnée pour les éléments qui se greffent sur cette ossature (fig. 11).

Un nouveau type de primitives est donc nécessaire pour tenir compte des éléments qui se trouvent reliés aux primitives principales. Puisque maintenant nous vérifions seulement la liaison à une primitive principale, ces nouvelles primitives porteront le nom de secondaires. Elles sont au nombre de trois :

1 - **Barres** . La figure 12a en donne quelques exemples. Il est très important de mentionner de quel côté de la primitive elles se trouvent ainsi que la position dans le corps de texte

2 - **Heurtoirs.** Ce sont des segments à orientation prépondérante verticale couvrant la moitié basse ou haute du corps de texte. Ils sont illustrés dans la figure 12b.

3 - **Points** . Voir la figure 12c.

Les primitives secondaires forment une base naturelle de vérifications nécessaires pour valider les décisions basées sur la synthèse des primitives principales.

3 - AVANTAGES

Plusieurs avantages découlent des choix que nous venons de faire. Les plus importants sont à nos yeux les suivants :

a) **Description naturelle**

En effet, nous nous sommes basés sur le travail des graveurs pour définir nos primitives qui sont un sous ensemble des primitives employées par eux ; la description d'un caractère se fait donc dans le même langage que celui des graveurs.

b) **Description unitaire**

Ceci est une conséquence qui découle du point a). En effet, les caractères gardent la même structure d'une police à l'autre, les différences qui apparaissent entre les polices n'étant pas de nature à modifier leur structure mais seulement quelques paramètres (graisse, hauteur, chasse, etc...) qui interviennent dans la définition des primitives.

Deux conséquences en découlent :

- les règles de production sont en nombre réduit (pour un caractère donné les primitives principales sont toujours les mêmes).
- l'apprentissage automatique des primitives principales se trouve simplifié par le nombre réduit de paramètres dont elles dépendent.

c) **Description unidimensionnelle**

En effet, les primitives employées par les graveurs se concatènent les unes aux autres sans se superposer. Il en va de même pour nos

primitives. La décomposition d'une ligne de texte en termes de primitives principales transforme donc un problème bidimensionnel en un autre unidimensionnel facilitant ainsi grandement la reconnaissance, basée sur des techniques grammaticales.

Nous aurons l'occasion de rappeler ces avantages lors des chapitres suivants qui détaillent notre méthode de lecture optique.

III - DESCRIPTION D'UNE LIGNE DE TEXTE EN TERMES DE PRIMITIVES

A - INTRODUCTION

1 - Hypothèses

2 - Problème

3 - Principe de la solution

B - EXPLOITATION DE LA REDONDANCE VERTICALE

1 - Intérêt d'une simplification

2 - Emploi de sondes horizontales

3 - Nombre de sondes

C - EXPLOITATION DE LA STRUCTURE HORIZONTALE

1 - Définition des groupes exploitables

2 - Repérage des groupes exploitables

3 - Prédiction sur la nature de la primitive

4 - Prédiction de la position de la primitive

D - DETERMINATION DES POINTS DE CONFIANCE

1 - Démarche première

2 - Démarche analytique

3 - Démarche heuristique

4 - Forme définitive de la fonction de corrélation

E - EXTRACTION DES PRIMITIVES SECONDAIRES

F - ILLUSTRATION

III

DESCRIPTION D'UNE LIGNE DE TEXTE EN TERMES DE PRIMITIVES

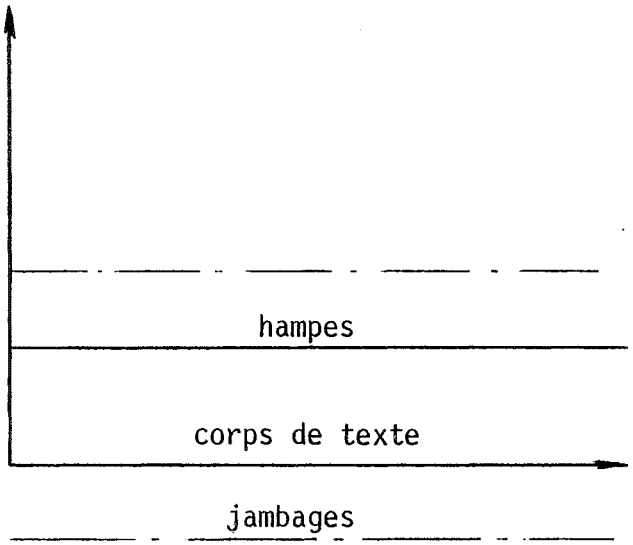
A - INTRODUCTION

1 - Hypothèses

Ce chapitre est consacré au codage d'une ligne de texte en terme de primitives définies dans le chapitre II. Nous admettons comme hypothèses de travail les faits suivants :

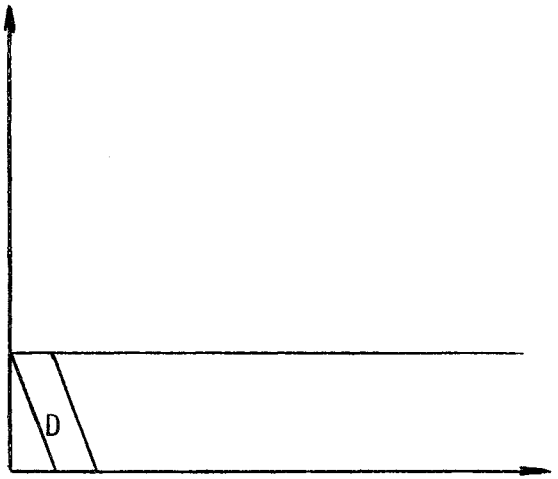
1°) La position de la ligne de texte, notamment son orientation, est connue ainsi que la position du corps de texte dans la ligne de texte. Cette hypothèse suppose résolus deux problèmes non triviaux : isoler d'abord une ligne de texte dans la page, trouver ensuite sa structure : corps de texte, hampes, jambages (fig. 1). Nous reviendrons plus en détails sur ces problèmes dans le chapitre IV.

2°) Le texte est monopolice et l'identité de la police est connue. Il en résulte que le modèle des primitives est parfaitement défini (le lecteur se rappelle que la primitive, en tant qu'élément structurel, englobe la représentation graphique des attributs fonctionnels, différents d'une police à l'autre). Dans le chapitre IV nous exposerons une méthode qui permet de trouver de manière automatique ces primitives lorsque l'identité de la police est inconnue a priori.



Structure verticale d'une ligne de texte.

Figure 1



Support d'une primitive 1.

Figure 2

2 - Problème

Dans le cadre des conditions exprimées plus haut, la difficulté à résoudre consiste en la décomposition de la ligne de texte en ses primitives constitutives. Ces primitives ne se superposent pas mais leur position est inconnue et il est impossible de les isoler a priori. En effet, la seule séparation qui pourrait être faite à ce stade de l'analyse se situe au niveau des mots (en identifiant les blancs séparateurs), mais il n'est pas aisé d'aller plus loin étant donnée la structure du mot : les caractères sont en principe isolés les uns des autres mais il arrive qu'ils se touchent, surtout s'ils sont pourvus d'empatements ; un caractère est formé d'une ou plusieurs primitives principales concaténées entre elles ou bien séparées par des primitives secondaires.

En termes plus généraux le problème à résoudre peut être énoncé de la sorte : identifier les signaux qui composent une suite de signaux, la frontière entre eux à l'intérieur de la suite n'étant pas connue a priori. Ceci se retrouve fréquemment en reconnaissance des formes (analyse de la voix par exemple).

3 - Principe de la solution

Simplifions le problème en nous intéressant à la découverte d'un seul signal à l'intérieur d'une suite.

La première solution qui vient à l'esprit consiste dans le balayage de la suite des signaux avec le signal que l'on recherche et

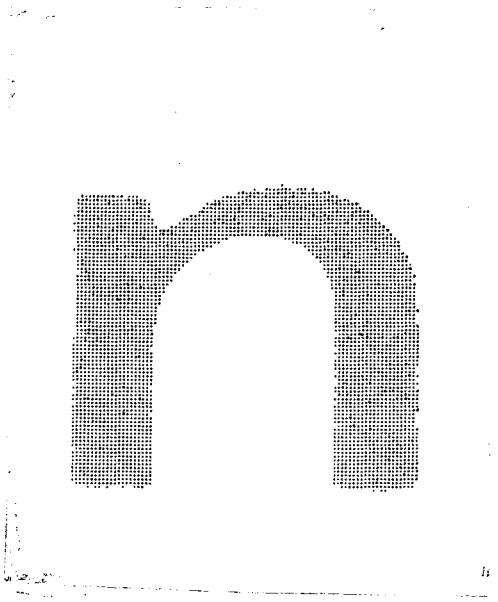
le repérage des endroits où la superposition est parfaite. Malheureusement, la réalité du problème précis qui nous préoccupe ne se laisse pas schématiser à ce point : on ne peut pas raisonnablement espérer une superposition parfaite entre le modèle des primitives et une partie de la ligne de texte tout simplement parce qu'une variation faible par rapport au modèle de certaines primitives peut apparaître du fait des liaisons dictées par la structure du caractère ou bien par son embellisement. La figure 3 montre une telle variation sur la primitive 5. Il faut donc définir un critère de classification basé sur une fonction de ressemblance à la place du critère tout-ou-rien que nous avons exposé.

On trouve dans la littérature /3.1/ plusieurs exemples de fonctions mesurant la ressemblance de deux images ; un bref aperçu sera donné dans la suite.

Soit f(x,y) la description (en termes d'intensité lumineuse par exemple) de la ligne de texte, p(x,y) la description analogue d'une primitive, D l'ensemble des points pour lesquels p(x,y) prend des valeurs non nulles (figure 2), et DT la translatée du domaine D d'une quantité u dans le sens horizontal :

$$(1) \quad DT = \{ (x,y) \mid (x-u,y) \in D \}$$

Voici une liste non exhaustive de fonctions mesurant la ressemblance entre la ligne de texte f(x,y) et la primitive translatée p(x-u,y) (il est inutile de translater la primitive selon l'axe Oy puisque l'on sait qu'elle est définie et calée sur le corps de texte) :



Variations dans la représentation d'une primitive : la deuxième primitive 5 se trouve recourbée du fait de la liaison avec la primitive précédente.

Figure 3

$$(2.1)\quad R_1(u) = \max_{DT} \left| f(x,y) - p(x-u,y) \right|$$

$$(2.2)\quad R_2(u) = \int_{DT} \left| f(x,y) - p(x-u,y) \right| dx\, dy$$

$$(2.3)\quad R_3(u) = \int_{DT} \left[f(x,y) - p(x-u,y) \right]^2 dx\, dy$$

$$(2.4)\quad R_4(u) = \int_{DT} f(x,y)p(x-u,y)\, dx\, dy = \int_{R^2} f(x,y)p(x-u,y)\, dx\, dy,$$

la ressemblance étant d'autant plus grande que R_1, R_2 ou R_3 sont petits ou R_4 grand. La fonction $R_4(u)$ est bien connue en traitement du signal pour ses applications au filtrage adapté ou détection d'un signal noyé dans le bruit (problème analogue au nôtre) ; il s'agit de la corrélation croisée des deux signaux réels f et p. Il est très aisé de démontrer que R_4 découle de R_3.

Nous allons étudier ce que représentent ces critères de ressemblance pour notre application. Pour le traitement sur ordinateur les images sont préalablement digitalisées : échantillonnées et quantifiées ; une quantification sur un seul bit (donc à deux niveaux : 0 et 1) est suffisante pour des images bien contrastées que sont les caractères. On négligera sciemment les unités de mesure pour ne s'intéresser qu'à la valeur de ses fonctions. Dans ces conditions :

$$(3.1)\quad f = f^2,\ p = p^2$$

$$(3.2)\quad |f-p| = (f-p)^2 = f \oplus p \quad (\oplus \text{ - ou exclusif logique})$$

La fonction $R_1(u)$ s'apparente à une décision par tout-ou-rien, sa valeur étant 0 en cas de superposition parfaite et 1 si les deux images diffèrent d'au moins un point ; elle est donc sans intérêt pour la suite.

Il y a une étroite relation entre R_2, R_3 et R_4. En effet, en appliquant (3.2) l'on obtient :

$$R_2(u)=R_3(u)= \int_{DT} f(x,y) \oplus p(x-u,y)dx\ dy$$
$$= \int_{DT} f(x,y)\bar{p}(x-u,y)dx\ dy + \int_{DT} \bar{f}(x,y)p(x-u,y)dx\ dy$$

(les opérateurs + logique et + arithmétique sont confondus parce que $f\bar{p}$ et $\bar{f}p$ ne peuvent pas prendre la valeur 1 simultanément). Par définition, $p(x-u,y)=1$ pour $(x,y) \in DT$, donc $\bar{p}(x-u,y) = 0$

D'un autre côté $\bar{f} = 1-f$. Il en résulte :

$$R_2(u)=R_3(u)= \int_{DT} \bar{f}(x,y)\ p(x-u,y)dx\ dy$$
$$= \int_{DT}[1-f(x,y)]p(x-u,y)dx\ dy$$
$$= \int_{DT} p(x-u,y)dx\ dy - \int_{DT} f(x,y)p(x-u,y)dx\ dy$$
$$= \int_{R2} p(x-u,y)dx\ dy - \int_{R2} f(x,y)p(x-u,y)dx\ dy$$

On reconnaît dans cette dernière formulation l'expression de l'énergie du signal p (rigoureusement l'énergie de p est :

$$E_p = \int p^2(x,y)dx\ dy$$

ce qui revient à l'expression indiquée en vertu de (3.2)), indépendante de u, et la corrélation croisée des signaux f et p.

$$(4) \quad R_2(u) = R_3(u) = E_p - R_4(u) \geq 0,$$

R_2 et R_3 étant des quantités positives ou nulles par définition. La fonction de ressemblance à laquelle nous nous sommes arrêtés découle de (4) et présente l'avantage d'exprimer la ressemblance de f et p en terme de pourcentage :

$$(5) \qquad r(u) = \frac{\int f(x,y)\,p(x-u,y)\,dx\,dy}{\int p(x,y)\,dx\,dy} \leq 1$$

sa valeur maximale étant atteinte dans le cas où :

$$f(x,y)=p(x-u,y),\ (x,y)\in DT$$

Dans la suite nous assimilerons souvent cette expression à une probabilité conditionnelle : la probabilité pour que le signal f représente la primitive p à l'abscisse u

$$(5') \qquad r(u) = \text{prob}\ (p/f)$$

(il ne s'agit pas d'une véritable probabilité parce que deux primitives p_i et p_j ne sont jamais orthogonales au sens de l'expression 5' : $\text{prob}(p_i/p_j) \neq 0$).

Les désavantages d'un tel choix sont faciles à constater : il faudrait pour chaque point u pris dans le sens de la ligne de texte calculer le coefficient r(u) pour ne prendre en compte que ses valeurs maximales et répéter ce processus autant de fois qu'il y a des primitives ; le temps de calcul serait prohibitif. Heureusement plusieurs améliorations peuvent être apportées à la méthode pour la rendre applicable dans la pratique. Ce chapitre est destiné à en exposer la nature. Nous verrons successivement comment exploiter la redondance verticale et la structure horizontale de la ligne de texte pour n'avoir à faire que des calculs en nombre très réduit pour déterminer la suite des primitives principales. En définitive le calcul de la fonction de corrélation se réduit à des mesures ponctuelles. Le passage le plus délicat de l'algorithme, la détermination de points de confiance, fait l'objet d'un paragraphe spécial. Après avoir dit un mot sur la détermination des primitives secondaires, nous terminons par une illustration de notre procédé.

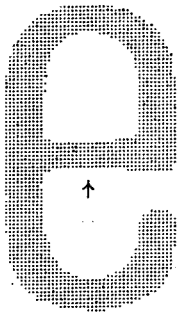
B - EXPLOITATION DE LA REDONDANCE VERTICALE

1 - Intérêt d'une simplification.

Calculer une intégrale du type de R_4 demande un temps proportionnel au domaine d'intégration. Si l'on veut accélérer le processus de calcul il faut donc diminuer le domaine de sommation qui est en fait le support de la primitive p. Ceci nous a paru possible en imposant une hiérarchie parmi les points qui constituent cette primitive : sera retenue une suite de points caractéristiques apportant le maximum d'information sur la forme analysée.

Cette idée de compression d'information (ou extraction de traits pertinents) n'est pas nouvelle en lecture optique, la littérature faisant état de nombreuses réalisations dans ce sens de principe très divers. Nous pouvons donc adapter au niveau des primitives les algorithmes étudiés au départ pour la reconnaissance de caractères entiers. Compte tenu de la simplicité relative des primitives par rapport aux caractères eux-mêmes, la méthode retenue sera évidemment beaucoup plus simple.

Ainsi Troxel /3.2/ emploie des sondes se déplaçant dans les quatre directions, chargées de découvrir les passages blanc-noir et noir-blanc, la décision de classification étant prise en fonction de cette information. La figure 4 montre comment une seule sonde de ce type peut faire la différence entre e et c : l'information graphique contenue dans la barre est réduite à un seul bit : présence ou absence d'intersection sur la sonde marquée. Il est nécessaire d'isoler au préalable le caractère.



Une seule sonde peut séparer **c** et **e**

(Troxel /3.2/)

Figure 4

Une autre méthode consiste en l'application d'un masque binaire sur un caractère après l'avoir isolé dans le contexte. On peut imaginer ce masque comme un écran opaque dans lequel on a pratiqué quelques trous à des endroits précis, apposé sur le caractère. Ce que l'on perçoit maintenant du caractère n'est plus sa description graphique mais son échantillonnage aux endroits où il y a des trous ; une comparaison entre cette description du caractère et les modèles déposés des signes de l'alphabet permet la classification (cette méthode est mieux connue sous le nom de "peephole mask").

D'autres réalisations mettent en oeuvre des méthodes basées sur le comptage du nombre d'intersections entre le caractère et des familles de droites verticales, horizontales ou bien choisies de façon aléatoire. Dans tous ces cas, une segmentation de la ligne de texte en caractères isolés est nécessaire au préalable. Une vue d'ensemble plus complète de ces méthodes est donnée dans l'article bibliographique de Berthod et Gaillat /2.1/.

2 - Emploi de sondes horizontales.

La méthode retenue est basée sur l'intersection avec des droites parallèles à la ligne de texte. A notre avis c'est en effet la mieux adaptée à une détection des primitives par corrélation. De plus, une telle méthode tient le mieux compte de l'orientation à prédilection verticale des primitives, ce qui assure une intersection avec la famille des droites à des abscisses pas trop dispersées, la présence simultanée d'une intersection sur toutes les droites à une même abscisse établissant d'autre part la continuité du trait représentant la primitive.

Le signal de départ lui-même discrétisé, se présentait déjà sous la forme d'un ensemble finement tramé de droites horizontales. La réduction de leur nombre traduit la redondance verticale des primitives principales que nous avons choisies.

Remplaçons la primitive $p(x,y)$ par son échantillonnée $p'(x,y)$ selon n droites horizontales, placées aux ordonnées y_i, i=1 à n. Dans ces conditions :

$$p'(x,y) = \sum_{i=1}^{n} p(x,y)\, d(y-yi)$$, d(y) étant la fonction de Dirac

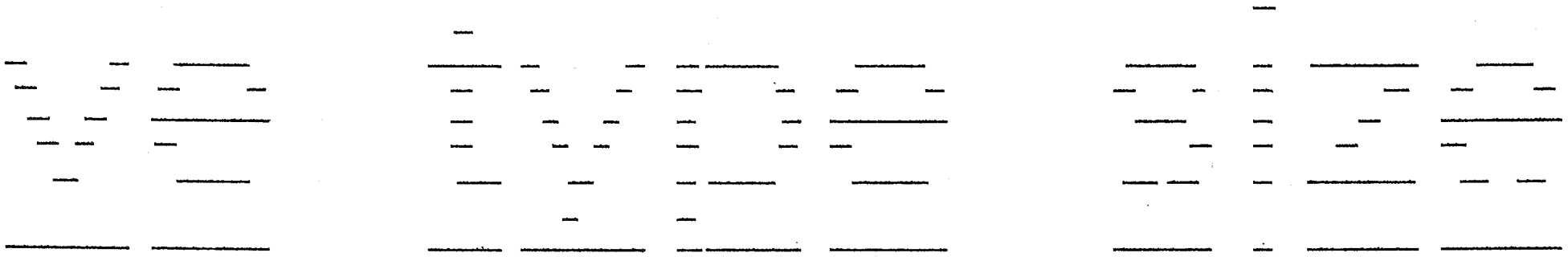
et la relation (5) devient :

$$r(u) = \frac{\sum_{i=1}^{n} \int f(x,y_i)p(x-u,y_i)dx}{\sum_{i=1}^{n} \int p(x,y_i)dx}$$

On remarque qu'il est inutile de connaître la description complète de la ligne de texte $f(x,y)$; seule suffit l'information se trouvant sur les droites $y=y_i$, i=1 à n. C'est sous cette forme que la ligne de texte sera désormais connue par notre système. La figure 5a montre l'image d'une ligne de texte sans compression de l'information graphique et 5b montre la même ligne après la compression (telle qu'elle est "perçue" par le lecteur optique) pour n=5 ; pour un humain cette information est encore suffisante pour permettre le déchiffrage du texte qu'elle représente.

A titre d'exemple les 512 x 80 bits de l'image 5a sont remplacés par 70*2*16 bits (il y a environ 70 segments dans la figure 5b, chacun codé par ses abscisses de début et de fin sous forme d'entiers

a) Texte original



b) Texte échantillonné avec 5 sondes sur le corps de texte, 2 dans la région des hampes et 1 dans la région des jambages. La ligne discontinue en dessous représente la projection des 5 sondes du corps de texte.

Compression d'information par échantillonnage avec des droites parallèles à la ligne de texte.

Figure 5

16 bits), donc un taux de compression de 16 (il peut être amélioré s'il y a des contraintes sur la mémoire, le choix de la représentation entière sur 16 bits, trop large pour l'application courante, ayant été fait pour des commodités de programmation).

L'analyse par sondes horizontales, appropriée pour la détection des primitives verticales, nous affranchit du besoin de segmentation dont pâtissent les autres méthodes. En effet, tout autre direction requiert un centrage sur le caractère pour que les mesures aient un sens ; il en va de même pour la méthode des masques (peephole mask). Avec notre méthode un centrage dans le sens vertical seulement est nécessaire (nous tirons profit de l'alignement des caractères dans le sens horizontal pour former des lignes de texte) mais nous savons résoudre ce problème (voir le chapitre IV).

D'un autre côté la direction horizontale est aussi le sens naturel de lecture. Il s'agit précisément de la direction dans laquelle les caractères subissent peu de variations de style (dont les empâtements) d'une police à l'autre. Cette disposition des sondes peut même permettre de traiter l'italique de la même manière qu'une police normale en le "redressant" d'abord de 13° environ.

3 - Nombre de sondes

Il reste à préciser le nombre de sondes et leur disposition relative au corps de texte. Une large place sera donnée à ces problèmes nontriviaux dans le chapitre IV. Pour garder l'unité de l'exposé nous donnons ici un simple aperçu.

Trois est le nombre de sondes sur le corps de texte minimum nécessaire pour préciser l'orientation d'une cavité et il serait suffisant pour définir de manière non ambiguë toutes les primitives principales et donc de les détecter. Il faut cependant tenir compte de la présence possible de primitives secondaires (surtout heurtoirs) qui ne s'étendent que sur une partie du corps de texte. Pour distinguer donc des cas comme celui de la figure 6 deux sondes supplémentaires, placées entre les sondes 1-2 et 2-3 sont nécessaires, leur rôle étant de vérifier les continuités dans le sens vertical.

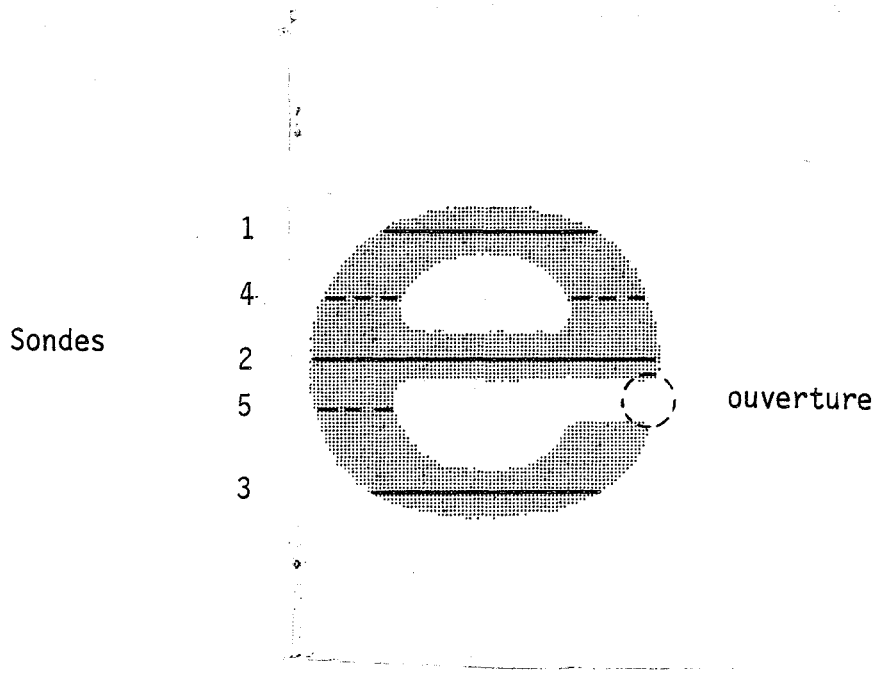
Pour conserver l'information se trouvant en dehors du corps de texte des sondes supplémentaires sont nécessaires. Ainsi dans la région des hampes deux sondes sont suffisantes pour détecter la présence et préciser la continuité des montants; pour les jambages une seule sonde est requise, ces derniers ne presentant pas de discontinuité.

Ces sondes sont placées de manière a fournir le maximum d'information sur la ligne de texte. La pratique a montré qu'un placement équidistant est acceptable pour des polices normales, mais échoue pour des polices condensées où les caractères, comme celui de la figure 6, ont des ouvertures relativement petites par rapport à leur taille. Une autre méthode, plus rationnelle, sera exposée dans le chapitre IV.

C - EXPLOITATION DE LA STRUCTURE HORIZONTALE

1 - Définition des groupes exploitables

La méthode la plus simple d'extraction des primitives de la ligne de texte consisterait en la corrélation de cette ligne avec



Avec seulement 3 sondes les côtés gauche et droit apparaissent identiques, malgré le manque de continuité à droite.

Figure 6

chaque modèle de primitive, identification des maxima (donc de la position de chaque primitive) sur chaque fonction de corrélation et en dernier le réordonnancement de ces primitives pour tenir compte du caractère unidimensionnel du texte décomposé en primitives dont nous avons fait état dans le chapitre précédent.

Un coup d'oeil à la figure 5b montre que l'espacement entre les segments successifs d'une sonde donnée est relativement grand par rapport à la longueur de ces mêmes segments. Comme une primitive exige la présence simultanée des cinq segments correspondants aux cinq sondes du corps de texte le calcul du coefficient de ressemblance (6) est une pure perte de temps pour les points u situés entre deux segments. On peut donc améliorer l'algorithme exposé plus haut en effectuant les calculs pour des points situés aux alentours des groupes de segments sur les cinq sondes.

En conclusion, la corrélation n'a de chance de dépasser un seuil significatif qu'en des endroits individualisables qui recèlent seuls les primitives cherchées. Mieux vaut repérer ces endroits dans l'ordre du texte et n'effectuer la corrélation qu'a posteriori. Il reste à repérer ces endroits, caractérisés par la présence d'un segment sur chacune des cinq sondes du corps de texte.

2 - Repérage des groupes exploitables

On peut encore réduire la recherche en faisant la remarque suivante : si un segment situé sur les sondes 1 ou 3 appartient très souvent à deux primitives principales en même temps (voir la figure 5b) il en est rarement de même pour les segments des sondes 4 ou 5 et

presque jamais pour les segments de la sonde 2. Il n'y a que deux exceptions à cette règle : x et k, le segment de la sonde 2 appartenant aux primitives de gauche et droite en même temps ; nous avons une solution à ce problème et nous l'exposerons plus loin. On peut donc affirmer que l'on a réussi à faire un centrage de la primitive sur le texte en prenant comme base la sonde 2 qui appartient à une primitive au maximum : sur la région déterminée par le segment de cette sonde on effectue la corrélation entre la ligne de texte et le modèle de chaque primitive. Pour résoudre les cas d'exception (x et k) et donner plus de souplesse au système les deux primitives les plus probables (celles dont la ressemblance est la plus grande avec la ligne de texte considérée sur le domaine de définition de la primitive) seront retenues à la place d'une seule ; de cette manière l'étage chargé de la synthèse des primitives pourra traiter et corriger les défauts de l'étage extracteur : **confusion** si la ligne de texte est assez ambiguë pour favoriser deux primitives avec des probabilités équivalentes ou bien **omission** si la sonde 2 appartient à deux primitives principales à la fois (x,k) ; dans ce cas la grande différence entre les deux primitives principales possibles attachées au même segment de la sonde 2 (4 suivi de 3 pour x, 5 suivi de 3 pour k) permet de détecter et corriger l'omission.

Nous avons insisté lors du chapitre II sur le caractère unidimensionnel de la ligne de texte si elle est décrite en termes de primitives telles les nôtres. Nous y reviendrons une dernière fois pour préciser qu'il se matérialise en pratique par une succession de segments sur la sonde 2, chacun pouvant appartenir à, au plus, une primitive.

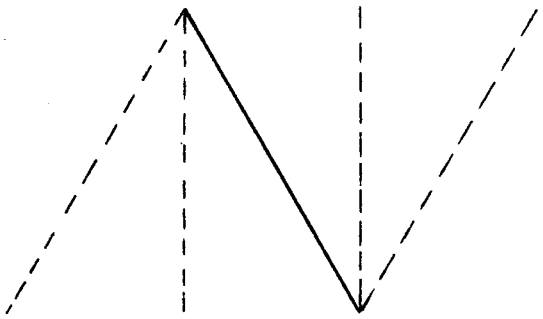
3 - Prédiction sur la nature de la primitive

D'après ce qui a été dit plus haut, pour chaque segment de la sonde 2, il faut calculer prob(p_i/f) où p_i est une des sept primitives principales de notre système (i=1 à 7) ; une étude exhaustive des hypothèses est donc faite. Il nous a semblé dans un premier temps possible de réduire le nombre d'hypothèses à vérifier en tenant compte d'une certaine affinité qui existe entre deux primitives successives. En effet, l'expérience a montré par exemple qu'une primitive 1 est souvent précédée ou suivie des primitives 2 ou 5 et plus rarement par d'autres ; il en va de même pour la primitive 2. Ces constatations doivent être mises en rapport avec le fait que souvent les segments des sondes 1 et 3 appartiennent à deux primitives à la fois : concaténation des primitives à l'intérieur d'un caractère. Le sens naturel de concaténation est donné dans la figure 7.

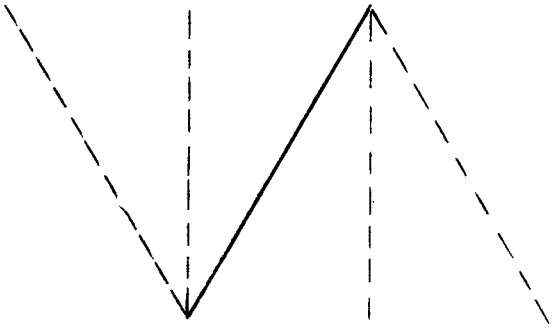
Nous avons abandonné par la suite cette idée parce que trop difficile à mettre en oeuvre ; en effet, les hypothèses de concaténation dont nous avons donné un exemple plus haut se rapprochent plus de l'étape de synthèse des primitives (qui sera abordée dans le chapitre V) que de celle d'extraction.

4 - Prédiction de la position de la primitive

Dans le chapitre II nous avons précisé la notion de primitive secondaire **barre** : un segment plus long que la graisse de la police résultant de l'approchement de deux primitives principales ou d'une primitive principale et une autre secondaire.



Primitive 1



Primitive 2

Quelques unes des concaténations possibles
des primitives 1 et 2.

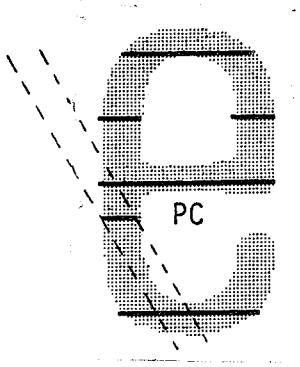
Figure 7

Pour la suite nous avons besoin d'une hypothèse qui se trouve largement vérifiée dans la pratique : parmi les cinq sondes composant une primitive principale trois au maximum peuvent présenter des barres sur un même côté. La figure 8 montre un tel cas, les barres étant sur le côté droit.

Dès lors une autre amélioration réduisant le temps de calcul de la corrélation vient à l'esprit : puisque au moins deux segments parmi les cinq ont leur origine (côté gauche) et au moins deux ont leur fin (côté droit) sur la primitive cachée dans la configuration des cinq sondes il suffit en principe d'en identifier un, du côté gauche ou du côté droit, de calculer le coefficient de ressemblance (8) dans ce point précis pour l'ensemble des primitives et de retenir les deux premières dans l'ordre décroissant des scores. Par la suite, ce point sera appelé **point de confiance** , la sonde sur laquelle il se trouve étant une **sonde de confiance** .

L'hypothèse introduite plus haut garantit l'existence d'un point de confiance sur les côtés gauche et droit. Sur la figure 8 la sonde de confiance côté gauche peut être une des cinq sondes (il n'y a pas de barre de ce côté) ; sur le côté droit le choix est plus restreint : sondes 4 ou 5.

A la lumière de cette dernière amélioration une remarque s'impose : le calcul fastidieux d'une corrélation entre une primitive et une zone de la ligne de texte centrée sur un segment de la sonde 2 a été remplacé par le calcul d'un seul coefficient de corrélation (la valeur de la fonction de corrélation dans un point précis), d'autant



PC : point de confiance

Détermination de la position de la primitive
à l'intérieur d'un groupement (en pointillé
une primitive 1 calée sur le point de confiance).

Figure 8

plus facile à obtenir que la primitive $p_i'(x,y)$ est définie avec des valeurs binaires : il s'agit donc d'un masquage. Le processus d'extraction peut être conçu comme la superposition du masque $p_i'(x,y)$ sur la ligne de texte de sorte qu'il y ait coïncidence entre les bords à la hauteur de la sonde de confiance.

Dans ce paragraphe, nous avons en fait progressivement remplacé la recherche aveugle d'un maximum de corrélation en nous guidant sur un modèle structurel du signal cherché : toute primitive est représentée par un groupement de cinq segments sur les sondes du corps de texte dont le segment situé sur la sonde médiane (sonde 2) est particulièrement représentatif. De plus, pour comparer un groupement inconnu à une primitive donnée, nous savons identifier a priori la position de cette primitive à l'intérieur du groupement si le groupement correspond bien à la primitive donnée. Dès lors la fonction de corrélation se réduit à une série de mesures ponctuelles.

D - DETERMINATION DES POINTS DE CONFIANCE

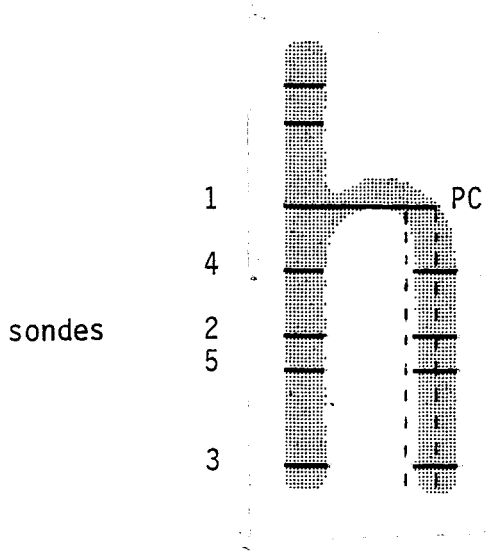
Il reste à préciser la manière dont un point de confiance est identifié. Pour fixer les idées nous allons nous intéresser au côté droit, le raisonnement étant analogue pour l'autre côté.

1 - Démarche première

La première solution que nous avons étudiée est basée sur la remarque suivante : puisque une barre déplace l'abscisse de fin du segment concerné au delà de l'endroit où elle devrait se trouver, il suffit de trouver la sonde dont l'abscisse de fin est la plus petite

pour être sûr d'avoir ainsi identifié la sonde et le point de confiance. Sur l'exemple de la figure 8 la sonde de confiance peut être 4 ou 5 (les deux abscisses sont égales). Choisissons 5 ; dans ce cas le point de confiance est l'extrémité droite du segment de la sonde 5. Il est facile de s'apercevoir par une construction graphique que la primitive qui s'approche le plus de cette configuration est la primitive 3 parce que la surface de leur superposition est la plus grande (pour comparaison la figure 8 montre la superposition de la forme et de la primitive 1).

Cette méthode donne d'excellents résultats dans presque tous les cas, avec cependant des difficultés qui tiennent à des variations dans l'allure de la primitive imposées par la concaténation à d'autres primitives à l'intérieur d'un caractère. La figure 9 en donne un exemple : la difficulté apparaît à l'analyse de la dernière primitive du caractère représenté (h) ; la sonde de confiance est 1 et le point de confiance PC. La primitive 5 donne un très mauvais résultat (à cause de la faible surface commune à la forme et à la primitive), dans ce cas précis moins bon que le résultat réalisé par la primitive 4. La cause de l'erreur est facile à détecter : le point de confiance droit ne se trouve pas sur la primitive 5 mais sur le lien de celle-ci à la primitive précédente. Il est très rare que ce genre de problème apparaisse des deux côtés en même temps ; dans le cas discuté une analyse centrée sur le côté gauche donnerait le bon résultat. Une démarche possible pour pallier à cette difficulté est donc la suivante : la forme doit être analysée sur les deux côtés sans distinction et assignée à la classe de la primitive la plus probable.



Le point de confiance PC provoque la substitution de la primitive 5 en primitive 4.

Figure 9

2 - Démarche analytique

Une autre démarche peut être envisagée : sur les cinq segments de la forme deux au minimum sont exempts de barres. Comme celui qui nous donnait jusqu'à présent le point de confiance peut être affecté de distorsions dans la forme de la primitive, il suffit de trouver le deuxième et baser l'analyse sur lui. Un tri dans l'ordre croissant des abscisses de fin des segments fournit la solution : le point de confiance cherché se trouve en deuxième position dans ce vecteur trié.

Il est aussi possible de faire une combinaison des deux procédés : analyser la forme sur les deux côtés en utilisant la stratégie que nous venons de décrire pour déterminer le point de confiance.

Nous donnons dans la suite une deuxième solution, moins intuitive que la première, qui évalue de manière plus fine la position du point de confiance :

Décrivons la forme par l'ensemble $F = \{fx_1, fx_2, fx_3, fx_4, fx_5\}$ des abscisses de début des cinq segments et l'ensemble $FL = \{fl_1, fl_2, fl_3, fl_4, fl_5\}$ des longueurs des mêmes segments. De la même façon $P = \{px_1, px_2, px_3, px_4, px_5\}$ et $PL = \{pl_1, pl_2, pl_3, pl_4, pl_5\}$ représentent la primitive. Signalons la propriété de la primitive d'avoir une épaisseur constante : $pl_i = g$, i=1 à 5, g étant la graisse moyenne de la police. La figure 10 illustre ces notations.

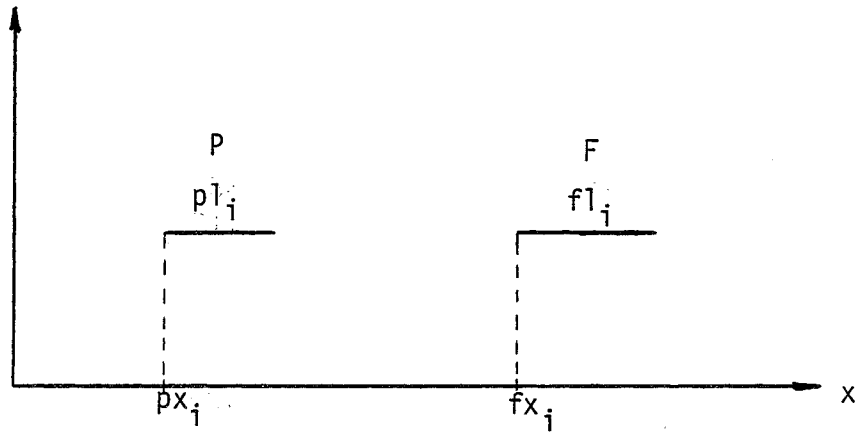


Figure 10

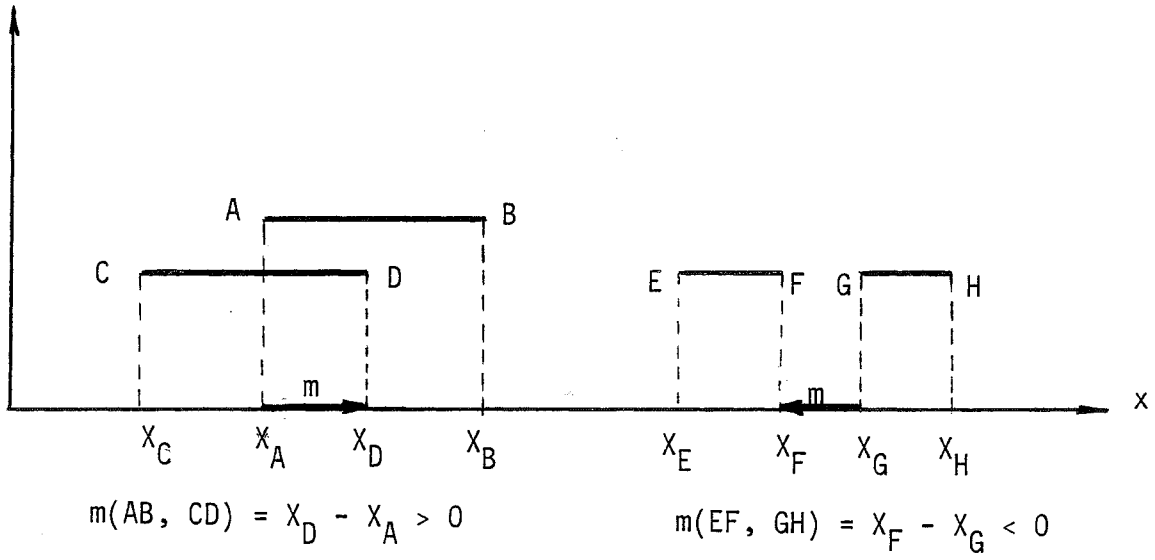


Figure 11

Trouver un point de confiance pour la forme F équivaut à trouver la valeur de u tel que r(u) donné par l'expression (6) soit maximum.

Nous modéliserons l'opération de masquage de deux segments AB et CD par la fonction :

$$(7) \quad m(AB, CD) = \min(x_B, x_D) - \max(x_A, x_C)$$

L'avantage que nous y voyons est que cette expression devient négative pour deux segments n'ayant pas de partie commune ; elle impose donc une pénalisation dans ce cas au lieu d'un apport nul (figure 11).

A la suite de ces considérations l'expression (6) devient :

$$(8) \quad r(u) = \frac{1}{\sum_{i=1}^{5} pl_i} * \sum_{i=1}^{5} [\min(fx_i+fl_i, px_i+pl_i+u) - \max(fx_i, px_i+u)]$$

Mais :

$$\min(fx_i+fl_i, px_i+pl_i+u) = pl_i + \min(fx_i+fl_i-pl_i, px_i+u)$$

et

$$pl_i = g, \; i = 1 \text{ à } 5,$$

donc

$$(8') \quad r(u) = 1 + \frac{1}{5g} * \sum_{i=1}^{5} [\min(fx_i+fl_i-g,\ px_i+u) - \max(fx_i,\ px_i+u)]$$

Une solution élégante existe pour (8') si l'on suppose $fl_i=g$, i=1 à 5. Dans ce cas :

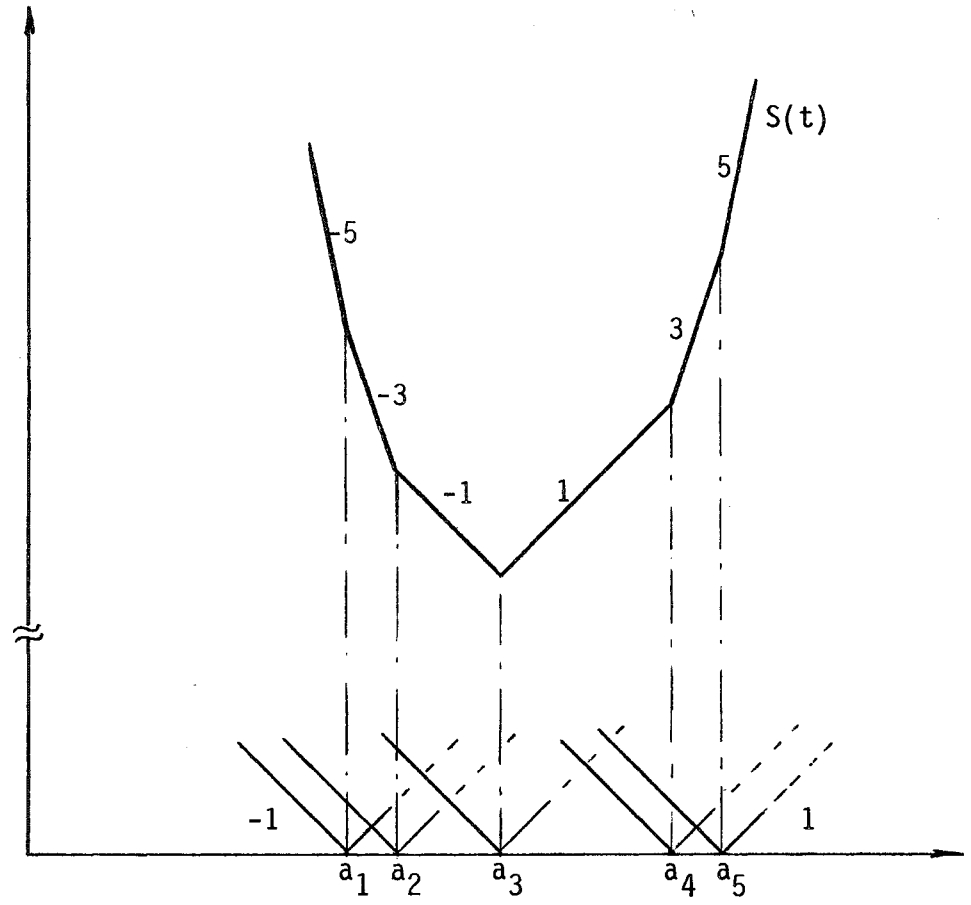
$$(9) \quad r(u) = 1 - \frac{1}{5g} * \sum_{i=1}^{5} |fx_i-px_i-u|$$

Etudions rapidement la fonction $S(t) = \sum_{i=1}^{5} |a_i-t|$, a_i et t étant des valeurs réelles. Supposons $a_1 \leq a_2 \leq a_3 \leq a_4 \leq a_5$ (hypothèse non restrictive pour cet exemple). La construction graphique de la figure 12 montre l'allure des courbes $|ai-t|$ et de la courbe S(t). On constate que S(t) présente un minimum pour $t=a_3$, c'est-à-dire la médiane de l'ensemble ai, i=1 à 5 (le lecteur aimant la rigueur mathématique arrivera aux mêmes résultats en montrant que S(t) est continue et dérivable par morceaux).

D'après ce développement r(u) passe par son maximum lorsque u est égal à la médiane de l'ensemble F-P

3 - Démarche heuristique

Cette dernière solution repose sur une condition trop restrictive (à savoir $fl_i=g$, i=1 à 5) pour être employée en pratique. Elle a cependant le mérite de suggérer une démarche à suivre dans le cas général. Etudions pour cela l'ensemble F-P.



$$S(t) = \sum_{i=1}^{5} |a_i - t|$$

Figure 12

I - La forme F représente une primitive P

Plusieurs cas sont possibles :

a) la forme F est exempte de barres et distorsions.
Dans ce cas $fx_i - px_i$ = const, i=1 à 5 et toutes les cinq valeurs sont représentées par un seul point dans la figure 13a.

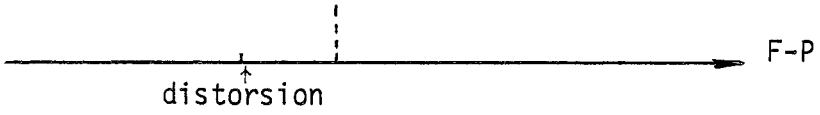
b) La forme F est exempte de barres mais peut présenter des distorsions comme celle de la figure 9.
Dans ce cas un point parmi les cinq se trouvera plus à gauche que les autres pour une analyse sur le côté droit (cas de la figure 9) ; voir la figure 13b (il servait de point de confiance pour la première solution).

c) La forme F peut présenter des barres et distorsions.
Par hypothèse maximum trois barres sont possibles simultanment ; trois points peuvent donc migrer à droite pour une analyse sur le côté droit (ou à gauche pour une analyse sur le côté gauche) ; voir la figure 13c.

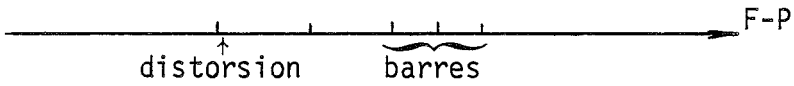
Nous constatons qu'un seul point est resté sur place ; il s'agit de la deuxième valeur dans l'ordre croissant pour le côté droit (décroissant pour le côté gauche) de l'ensemble F-P. Ce sera lui le point de confiance.

Toute cette analyse repose sur le fait généralement vérifié que, malgré les distorsions et les barres, il reste un point au moins sur la frontière de la primitive ; les distorsions affectent au plus une sonde et ont une action contraire aux barres.

a) Forme F pure



b) Forme F distordue



c) Distorsion et barres dans la forme F

Figure 13

II - La forme F ne représente pas une primitive P.

Dans ce cas le point trouvé selon la stratégie décrite plus haut n'est pas un point de confiance et la probabilité prob(P/F) sera très faible ; une comparaison avec un seuil élimine ces cas.

A ce point nous constatons une seule différence entre les deux solutions proposées : la première déterminait son point de confiance en se basant sur F, alors que la deuxième est basée sur l'exploitation des informations fournies pour F-P. Dans le premier cas le point de confiance était le même pour toutes les primitives du système, alors que dans le deuxième cas il change d'une primitive à l'autre (l'ensemble F-P est dépendant de P).

4 - Forme définitive de la fonction de corrélation

L'étude des formes F représentant réellement une primitive P a montré par ailleurs que certaines distorsions, celles qui sont provoquées par des empattements ou concaténations, sont toujours localisées sur les mêmes sondes et du même côté de P. Ainsi, une primitive 1 peut être reliée à une autre primitive au niveau de la sonde 1 pour une liaison à gauche ou au niveau de la sonde 3 pour une liaison à droite (figure 7) ; les primitives 3 et 4 peuvent être concaténées du côté de la concavité du niveau des sondes 1 et/ou 3, le côté convexe devant être entièrement libre.

Ces constatations nous ont amenés à compléter la description

de la primitive donnée en § D.2 (ensembles P et PL) par l'ensemble PS = $\{ s_1, s_2, s_3, s_4, s_5 \}$, indiquant pour chaque sonde i(i=1 à 5) les côtés qui doivent rester libres.

Ainsi, avec les conventions suivantes :

g - côté gauche libre

d - côté droit libre

0 - indifférent

la primitive 4 sera caractérisée par l'ensemble :

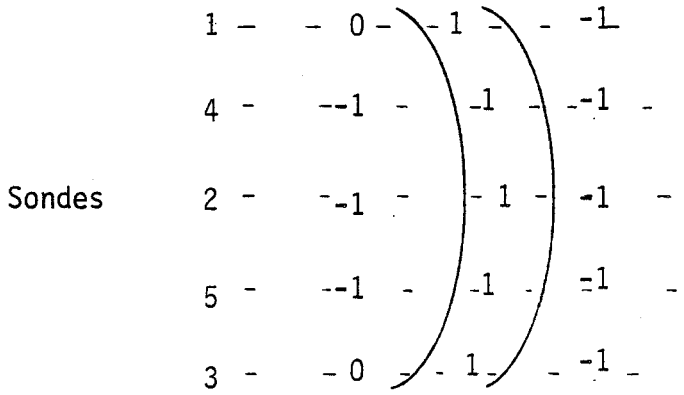
PS = {d, gd, d, gd, gd}

traduisant le fait que les concaténations possibles se situent sur les sondes 1 et 3, côté gauche, les sondes 2, 4 et 5 devant être libres des deux côtés.

La figure 14 illustre graphiquement la manière dont nous avons réalisé en définitive la fonction de corrélation : le recouvrement des segments marqués -1 diminuera la valeur finale du masquage proportionnellement à la longueur recouverte. Cette stratégie est censée diminuer encore le nombre des substitutions telles celle de la figure 9 ; la pratique nous a donné raison.

E - EXTRACTION DES PRIMITIVES SECONDAIRES

Cette tâche peut être résolue à peu d'effort après l'extraction des primitives principales. En effet, à ce stade de l'avancement l'identité et la position des primitives principales sont parfaitement connues. Les primitives secondaires seront cherchées dans l'espace



Modèle de la primitive 4.

Les chiffres indiquent le poids des segments correspondants lors de la corrélation.

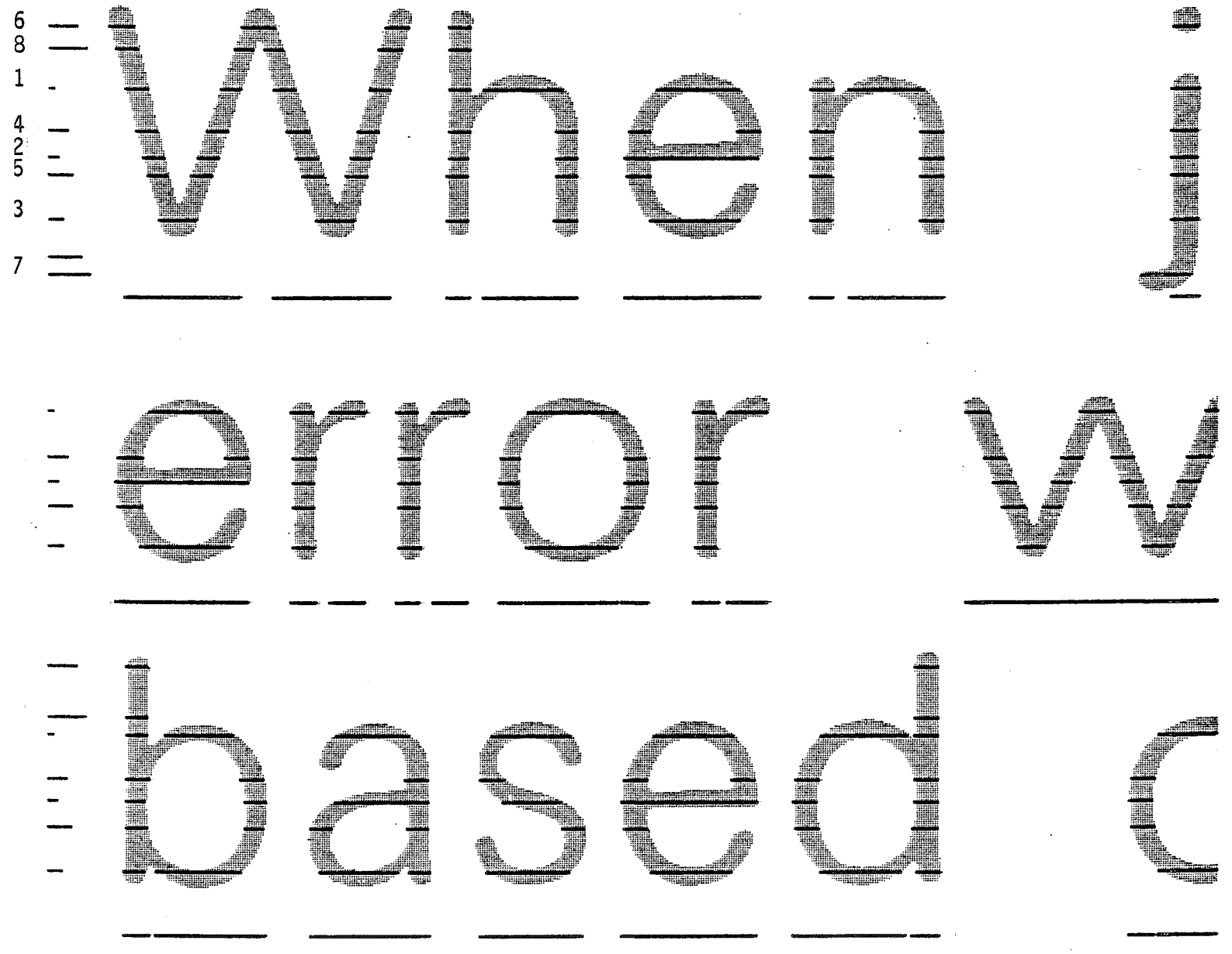
Figure 14

situé entre deux primitives principales successives. Leur nombre réduit et les formes simples qu'elles présentent facilitent le travail. Les informations qu'elles fournissent sont plus des marques qualitatives que des supports précis d'analyse. Beaucoup de précision n'est donc pas nécessaire dans leur détection. La tâche de détection est encore simplifiée par la position optimale des sondes sur le corps de texte, position définie en fonction de l'histogramme de la ligne de texte (voir chapitre IV). La forme accidentée de celui-ci est en grande mesure provoquée par les primitives secondaires : les barres provoquent des pics, les heurtoirs entraînent des creux à cause de leur manque de continuité.

Pour chaque segment trouvé entre deux primitives principales nous vérifions la continuité dans le sens vertical sur la moitié du corps de texte à laquelle il appartient pour déterminer les **heurtoirs** . La liaison d'un segment à la primitive de gauche ou de droite est une donnée essentielle pour la suite ; dans ce cas, le segment sera marqué comme **barre** . Si le segment ne peut être inclus en aucune de ces deux catégories, il sera introduit dans la base des données comme **point** .

F - ILLUSTRATION

Cette partie est destinée à l'illustration du principe d'extraction des primitives discuté précédemment. L'exemple traité est celui de la figure 15.



(Univers light 685)

Chaque ligne de texte comporte :
- au début de la ligne la position des sondes ;
- l'échantillonnage du texte avec les sondes (segments noirs) ;
- la projection horizontale des 5 sondes du corps de texte en dessous de la ligne.

Sondes : 1, 2, 3, 4, 5 - corps de texte
6, 8 - hampes
7 - jambages

Figure 15

Segm	1	2	3	4	5	6	7	8
1	33 43	41 50	48 64	38 47	43 51	26 36	478 499	29 38
3	75 83	65 74	117 133	69 78	62 71	84 98	-1 -1	81 89
5	98 107	108 117	174 183	104 114	111 120	147 157	-1 -1	94 102
7	140 149	132 140	221 231	135 144	130 139	175 184	-1 -1	144 154
9	175 184	175 183	263 302	175 184	174 183	492 502	-1 -1	175 184
11	190 224	222 231	333 342	222 231	222 231	-1 -1	-1 -1	-1 -1
13	266 301	252 310	381 391	254 263	253 263	-1 -1	-1 -1	-1 -1
15	333 343	333 342	491 502	301 311	333 342	-1 -1	-1 -1	-1 -1
17	350 382	381 391	-1 -1	333 343	381 391	-1 -1	-1 -1	-1 -1
19	491 503	491 502	-1 -1	381 391	491 502	-1 -1	-1 -1	-1 -1
21	-1 -1	-1 -1	-1 -1	491 502	-1 -1	-1 -1	-1 -1	-1 -1
23	-1 -1	-1 -1	-1 -1	-1 -1	-1 -1	-1 -1	-1 -1	-1 -1

a) *Ligne de texte échantillonnée*

NOM	1	4	2	5	3
4	-12 1	-1 11	0 11	-1 11	-11 1
3	12 10	1 10	0 10	1 10	11 10
2	10 0	4 0	0 0	-3 0	-9 0
1	-10 0	-4 0	0 10	3 10	9 0
5	0 0	0 0	0 0	0 0	0 0

b) *Modèle des primitives*

PRIM	ATTR	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
G(G)	( )	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	-1	0
1(1)	3(3)	84(82)	1(1)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	1(1)	-2(4)	40(42)	-1	0
2(2)	3(3)	86(84)	3(3)	3(3)	1(1)	3(3)	3(3)	3(3)	0(0)	3(3)	2(-1)	65(63)	-1	0
1(1)	3(3)	94(90)	5(5)	5(5)	3(3)	5(5)	5(5)	3(3)	0(0)	5(5)	2(-2)	108(107)	-1	0
2(2)	3(3)	86(84)	7(7)	7(7)	3(3)	7(7)	7(7)	5(5)	0(0)	7(7)	-4(2)	130(132)	-1	0
5(5)	3(3)	84(84)	9(9)	9(9)	5(5)	9(9)	9(9)	7(7)	0(0)	9(9)	2(-3)	175(173)	-1	1
5(5)	0(0)	80(76)	11(11)	11(11)	7(7)	11(11)	11(11)	0(0)	0(0)	0(0)	-5(4)	221(222)	-1	0
3(3)	0(0)	94(94)	13(13)	13(13)	9(9)	13(13)	13(13)	0(0)	0(0)	0(0)	4(-4)	253(252)	-1	4
5(5)	0(0)	94(90)	15(15)	15(15)	11(11)	17(17)	15(15)	0(0)	0(0)	0(0)	2(-3)	333(332)	-1	1
5(5)	0(0)	82(82)	17(17)	17(17)	13(13)	19(19)	17(17)	0(0)	0(0)	0(0)	3(-5)	381(381)	-1	0
5(5)	5(5)	100(100)	19(19)	19(19)	15(15)	21(21)	19(19)	9(9)	1(1)	0(0)	2(-4)	491(492)	-1	0

c) *Liste des primitives principales*

1	BARRE	SUR 1.	LIEN A DROITE
2	BARRE	SUR 1.	LIEN A GAUCHE
3	BARRE	SUR 2.	LIEN A GAUCHE
4	BARRE	SUR 3.	LIEN A GAUCHE
5	HEURTOIR	SUR 1.	LIEN A GAUCHE
6	BARRE	SUR 1.	LIEN A DROITE

d) *Liste des primitives secondaires*

Tableau 1

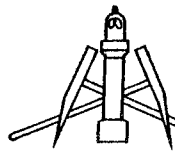
On trouvera dans Tableau 1a l'information retenue après l'échantillonnage de la première ligne de texte avec les 8 sondes placées automatiquement par le préprocesseur ; les segments résultants sont stockés par la paire (abscisse de début, abscisse de fin) dans les 8 colonnes marquées 1 à 8 correspondant aux 8 sondes de la ligne de texte, chacune étant organisée en liste linéaire. La première colonne **Segm** indique la position dans ces listes de l'abscisse de début du segment ; il servira désormais à repérer un segment dans la liste.

Nous avons indiqué dans Tableau 1b la liste des primitives nous ayant servi de modèle. La colonne **NOM** indique leur nom : les cinq colonnes qui suivent (portant l'en-tête **1, 4, 2, 5** et 3) définissent la primitive de la manière suivante :

- les premiers nombres de chaque colonne forment l'ensemble P, conformément à la partie § D.2.

- le deuxième nombre de chaque colonne indique le côté sur lequel la sonde correspondante doit être libre, conformément à la partie § D.4 :
  1 - libre sur la droite
  10 - libre sur la gauche
  11 - libre des deux côtés
  00 - indifférent

La figure 16 montre le détail de l'extraction pour les groupements centrés autour du segment 1 de la sonde 2.



```
---> EXTPRI      CONF.    1    1    1    1    1      (groupement / sondes 1, 2, 3, 4, 5)
   4    3    2    1    5                            (primitive)
  12   16  -10   82   48                            (probabilité / coté gauche)
   1    2    5    4    5                            (sonde de confiance / coté gauche)
   8   30  -12   84   48                            (probabilité / coté droit)
   2    4    4    2    4                            (sonde de confiance / coté droit)

PRIMS:        1    1                                (les deux meilleures primitives du groupement)
PROBS:       84   82                                (probabilités)
SONC :       -2    4                                (sondes de confiance)
POS2 :       40   42<---                           (position de la sonde 2 coté gauche)

---> EXTPRI      CONF.    1    1    1    1    3
   4    3    2    1    5
 -62  -20  -65   56    4
   3    2    3    1    3
 -50   -8  -50   56   10
   2    4    4    4    4

PRIMS:        1    1
PROBS:       56   56
SONC :        1   -4
POS2 :       43   41<---

---> EXTPRI      CONF.    1    1    1    3    1
   4    3    2    1    5
 -70  -26  -44   20   -2
   3    5    3    1    3
 -54  -20  -32   22    4
   5    5    2    2    2

PRIMS:        1    1
PROBS:       22   20
SONC :       -2    1
POS2 :       40   43<---

---> EXTPRI      CONF.    1    1    1    3    3
   4    3    2    1    5
 -28 -158  -42  -46  -34
   5    5    5    5    5
 -88  -58  -70    0  -34
   1    2    2    1    2

PRIMS:        1
PROBS:        0   -1
SONC :       -1    0
POS2 :       43    0<---

PTRS:  1  1  1  1  1  PRIM:1  PROB: 84  SONC :  -2  POS2 :  40    (résultat global concernant
PTRS:  1  1  1  1  1  PRIM:1  PROB: 82  SONC :   4  POS2 :  42    (le segment 1 de la sonde 2)
```

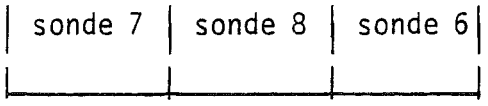
Figure 16

Le résultat global de l'extraction des primitives est donné dans Tableau 1c. Les quantités relatives au second choix de primitive sont placées entre parenthèses.

Nous trouvons :

- dans la colonne **PRIM** les deux meilleures primitives des groupements centrés autour d'un segment de la sonde 2.

- dans la colonne **ATTR** les attributs de ces primitives. Il faut entendre par attributs les montants et descendants qu'une primitive peut présenter et dont la détection est réalisée par les sondes 6 et 8 pour les hampes, 7 pour les jambages. Cette information est codée de la façon suivante :



bit : 2 1 0

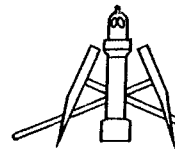
La présence d'intersection sur une quelconque de ces sondes étant représentée par 1, l'absence par 0.

- Dans la colonne **PROB** les résultats de la corrélation entre le groupement et le patron (assimilé à une probabilité).

- Dans les colonnes **PTS1** à **PTS8** les segments de Tableau 1 qui composent la primitive.
- Dans la colonne **SONC** la sonde de confiance qui a servi à la mesure de la ressemblance ; son signe définit le côté sur lequel se trouve le point de confiance : + pour gauche, - pour droit.
- Dans la colonne **POS2** la position de la primitive dans la ligne de texte ; obtenue à partir de la sonde et du point de confiance, ramenée à la sonde 2, côté gauche.
- Les colonnes **OUTSP** et **OUTSL** conservent des informations concernant les primitives secondaires se trouvant entre la primitive principale et son successeur (il s'agit des meilleurs choix) ; ils indiquent le début et respectivement la longueur de la zone de primitives secondaires dans la liste des primitives secondaires.

La liste des primitives secondaires se trouve dans Tableau 1d.

L'information concernant la deuxième et troisième ligne de texte se trouve dans Tableau 2 et Tableau 3 respectivement.



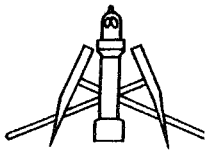
1	44	75	29	87	40	79	30	41	30	40	-1	-1	-1	-1	-1	-1
3	106	115	107	116	107	116	77	87	107	116	-1	-1	-1	-1	-1	-1
5	123	138	153	162	153	162	107	117	153	162	-1	-1	-1	-1	-1	-1
7	152	161	197	205	209	249	153	162	198	207	-1	-1	-1	-1	-1	-1
9	168	183	252	262	283	292	198	206	251	260	-1	-1	-1	-1	-1	-1
11	210	248	283	292	424	436	252	261	283	292	-1	-1	-1	-1	-1	-1
13	282	291	413	422	479	492	283	293	417	427	-1	-1	-1	-1	-1	-1
15	297	314	439	447	-1	-1	409	418	435	443	-1	-1	-1	-1	-1	-1
17	401	411	469	478	-1	-1	443	452	473	482	-1	-1	-1	-1	-1	-1
19	451	466	494	503	-1	-1	465	474	490	499	-1	-1	-1	-1	-1	-1
21	506	511	-1	-1	-1	-1	498	508	-1	-1	-1	-1	-1	-1	-1	-1
23	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1

NOM	1		4		2		5		3	
4	-13	1	-1	11	0	11	-1	11	-12	1
3	13	10	1	10	0	10	1	10	12	10
2	10	0	4	0	0	0	-3	0	-9	0
1	-10	0	-4	0	0	10	3	10	9	0
5	0	0	0	0	0	0	0	0	0	0

PRIM G(G)	ATTR ( )	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
		0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	-1	0
3(3)	0(0)	95(88)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	0(0)	0(0)	2(-4)	29(31)	1	4
5(5)	0(0)	97(97)	3(3)	3(3)	3(3)	5(5)	3(3)	0(0)	0(0)	0(0)	3(-5)	107(107)	5	1
5(5)	0(0)	97(97)	7(7)	5(5)	5(5)	7(7)	5(5)	0(0)	0(0)	0(0)	3(-5)	153(153)	6	1
3(3)	0(0)	95(88)	11(11)	7(7)	7(7)	9(9)	7(7)	0(0)	0(0)	0(0)	2(-2)	197(196)	7	2
4(4)	0(0)	97(97)	11(11)	9(9)	7(7)	11(11)	9(9)	0(0)	0(0)	0(0)	2(-3)	252(252)	-1	1
5(5)	0(0)	97(97)	13(13)	11(11)	9(9)	13(13)	11(11)	0(0)	0(0)	0(0)	3(-5)	283(283)	9	1
1(1)	0(0)	91(88)	17(17)	13(13)	11(11)	15(15)	13(13)	0(0)	0(0)	0(0)	-4(5)	413(414)	-1	4
2(2)	0(0)	82(77)	19(19)	15(15)	11(11)	17(17)	15(15)	0(0)	0(0)	0(0)	2(-5)	439(437)	-1	1
1(1)	0(0)	91(88)	19(19)	17(17)	13(13)	19(19)	17(17)	0(0)	0(0)	0(0)	-4(5)	469(470)	-1	0
2(2)	0(0)	84(80)	21(21)	19(19)	13(13)	21(21)	19(19)	0(0)	0(0)	0(0)	2(-1)	494(492)	-1	0

1	BARRE	SUR 1.	LIEN A GAUCHE
2	BARRE	SUR 2.	LIEN A GAUCHE
3	BARRE	SUR 3.	LIEN A GAUCHE
4	HEURTOIR	SUR 1.	LIEN A GAUCHE
5	POINT	SUR 1.	
6	POINT	SUR 1.	
7	BARRE	SUR 1.	LIEN A GAUCHE ET DROITE
8	BARRE	SUR 3.	LIEN A GAUCHE ET DROITE
9	POINT	SUR 1.	

Tableau 2



1	34	43	34	42	33	42	34	44	34	43	34	43	-1	-1	34	43			
3	51	80	86	95	47	84	84	94	86	94	379	389	-1	-1	379	389			
5	126	157	126	166	119	153	156	166	115	124	-1	-1	-1	-1	-1	-1			
7	194	228	199	224	158	167	189	199	157	166	-1	-1	-1	-1	-1	-1			
9	265	299	251	310	192	228	252	262	225	234	-1	-1	-1	-1	-1	-1			
11	338	375	326	336	264	299	299	309	252	262	-1	-1	-1	-1	-1	-1			
13	379	389	379	389	338	373	327	337	327	337	-1	-1	-1	-1	-1	-1			
15	486	510	473	483	380	390	379	389	378	389	-1	-1	-1	-1	-1	-1			
17	-1	-1	-1	-1	486	511	474	484	474	484	-1	-1	-1	-1	-1	-1			
19	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1			

NOM	1		4		2		5		3	
6	-13	10	-19	10	0	0	18	1	8	1
4	-15	1	-1	11	0	11	-1	11	-12	1
3	15	10	1	10	0	10	1	10	12	10
2	9	0	3	0	0	0	-4	0	-10	0
1	-9	0	-3	0	0	10	4	10	10	0
5	0	0	0	0	0	0	0	0	0	0

PRIM G(G)	ATTR ( )	PROB 0(0)	PTS1 0(0)	PTS2 0(0)	PTS3 0(0)	PTS4 0(0)	PTS5 0(0)	PTS6 0(0)	PTS7 0(0)	PTS8 0(0)	SONC 0(0)	POS2 0(0)	OUTSP -1	OUTSL 0
5(5)	3(3)	88(82)	1(1)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	1(1)	2(-2)	34(32)	1	2
4(4)	0(0)	94(90)	3(3)	3(3)	3(3)	3(3)	3(3)	0(0)	0(0)	0(0)	-4(2)	85(86)	3	4
5(5)	0(0)	76(72)	5(5)	5(5)	7(7)	5(5)	7(7)	0(0)	0(0)	0(0)	-5(5)	156(157)	7	1
6(6)	0(0)	94(94)	7(7)	7(7)	9(9)	7(7)	9(9)	0(0)	0(0)	0(0)	1(-4)	207(208)	8	1
3(3)	0(0)	98(98)	9(9)	9(9)	11(11)	9(9)	11(11)	0(0)	0(0)	0(0)	2(-4)	251(251)	9	4
3(3)	0(0)	98(98)	11(11)	11(11)	13(13)	13(13)	13(13)	0(0)	0(0)	0(0)	3(-4)	326(326)	13	2
5(5)	3(3)	98(98)	13(13)	13(13)	15(15)	15(15)	15(15)	3(3)	0(0)	3(3)	1(-4)	379(379)	-1	4
3(3)	0(0)	96(96)	15(15)	15(15)	17(17)	17(17)	17(17)	0(0)	0(0)	0(0)	2(-4)	473(473)	15	2

1	BARRE	SUR 1.	LIEN A DROITE
2	BARRE	SUR 3.	LIEN A DROITE
3	BARRE	SUR 1.	LIEN A DROITE
4	BARRE	SUR 2.	LIEN A DROITE
5	POINT	SUR 3.	
6	HEURTOIR	SUR 3.	LIEN A DROITE
7	BARRE	SUR 3.	LIEN A DROITE
8	BARRE	SUR 1.	LIEN A GAUCHE
9	BARRE	SUR 1.	LIEN A GAUCHE
10	BARRE	SUR 2.	LIEN A GAUCHE
11	BARRE	SUR 3.	LIEN A GAUCHE
12	HEURTOIR	SUR 1.	LIEN A GAUCHE
13	BARRE	SUR 1.	LIEN A GAUCHE
14	BARRE	SUR 3.	LIEN A GAUCHE
15	BARRE	SUR 1.	LIEN A GAUCHE
16	BARRE	SUR 3.	LIEN A GAUCHE

Tableau 3

IV - PREPROCESSEUR

A - INTRODUCTION

1 - Problèmes

2 - Hypothèses

B - IDENTIFICATION DES LIGNES DE TEXTE

1 - Localisation des lignes de texte

2 - Structure d'une ligne de texte

C - IDENTIFICATION AUTOMATIQUE DES PRIMITIVES

1 - Introduction

2 - Primitive 5

3 - Primitives 3 et 4

4 - Primitives 1 et 2

5 - Primitives 6 et 7

6 - Espaces

D - ILLUSTRATION

IV

PREPROCESSEUR

A - INTRODUCTION

1 - Problèmes

Nous avons vu lors des chapitres précédents que notre système de lecture optique est basé sur la reconnaissance d'éléments graphiques primitifs. Le chapitre III décrit en détail les algorithmes sur lesquels cette reconnaissance est faite. Pour travailler correctement ces algorithmes exigeaient la connaissance de :

- l'orientation et la structure de la ligne de texte.
- modèle des primitives.

Ce chapitre aura pour but de résoudre quelques uns de ces problèmes. Nous traiterons ainsi dans l'ordre le problème de l'identification de la ligne de texte, avec d'une part

- la détection des lignes de texte dans une page ;

et d'autre part

- l'identification de la structure d'une ligne de texte, à savoir : corps de texte, hampes, jambages ;

ainsi que

- l'identification automatique des modèles des primitives.

2 - Hypothèses

Les méthodes que nous proposons pour trouver l'information nécessaire reposent elles-mêmes sur les hypothèses suivantes :

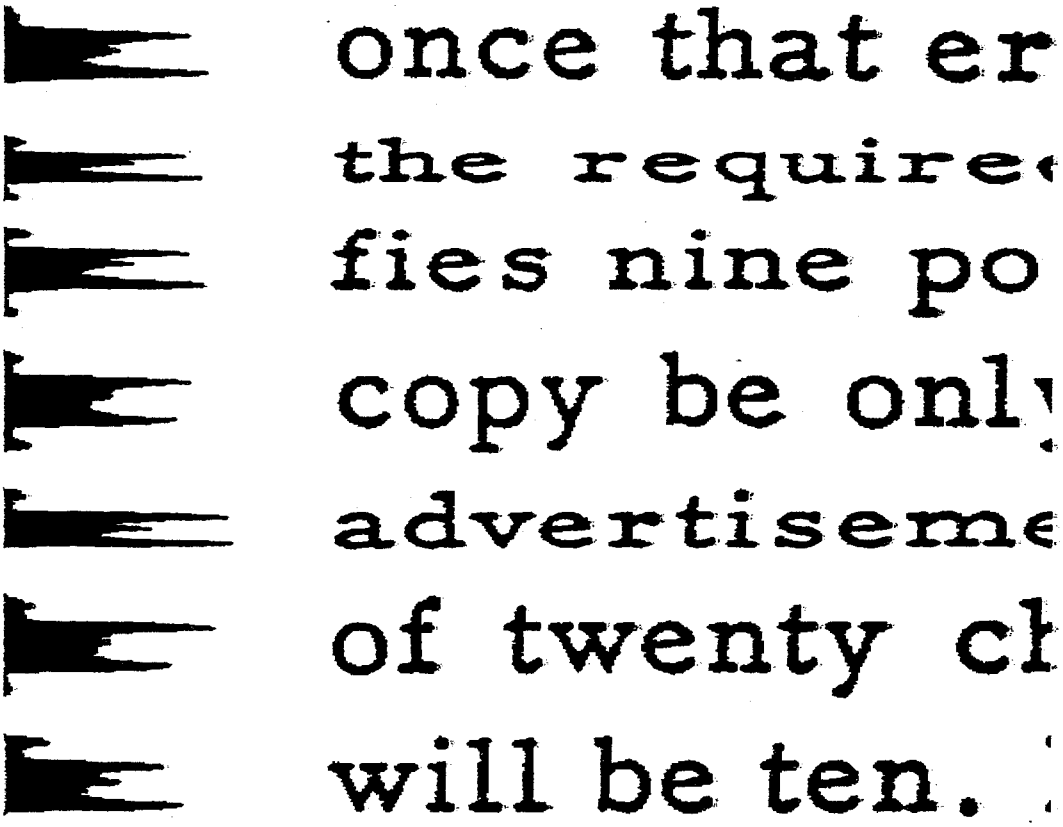
- l'orientation du texte est connue
- la page contient du texte monopolice à exclusion de toute autre représentation graphique (photos, schémas, etc...).

La deuxième hypothèse suppose réalisable la séparation entre texte et non-texte. Cette opération est en elle-même un problème non trivial dont la résolution dépasse les limites d'un exposé sur les méthodes de lecture optique. Le lecteur qui veut se familiariser avec ce sujet est invité à consulter l'article de E.G. Johnston /4.1/.

Le problème caché derrière la première hypothèse a été évoqué en partie lors du chapitre III. Trouver l'orientation du texte n'est pas un problème aisé, la difficulté résidant dans le fait que les lecteurs optiques sont souvent dépourvus d'une vue d'ensemble de la page typographiée. Une telle vue d'ensemble, même grossière, permet de saisir le texte sous forme de bandes parallèles, successives (les lignes de texte), sans se préoccuper des éléments qui les composent (caractères).

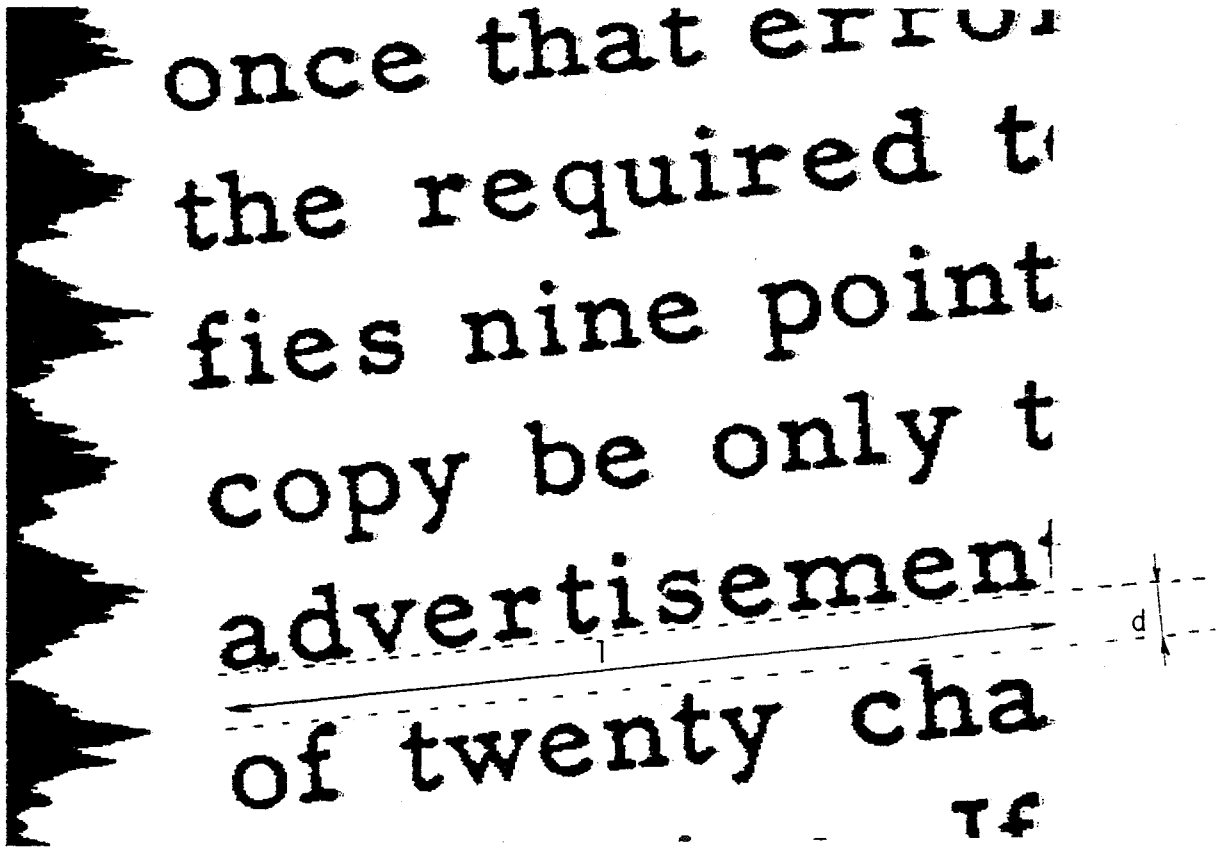
Cette structure périodique pourrait être mise en évidence par des techniques fréquencielles (analyse de Fourier) ou spatiales (histogramme selon une direction variable : lorsque la direction est perpendiculaire aux lignes de texte l'histogramme acquiert une structure périodique ; voir figures 1 et 2).

En général les lecteurs du commerce résolvent ces difficultés en imposant des contraintes qui les rendent triviales : strict alignement du texte avec les limites physiques de la page, interdiction de mélanger texte et graphique. Nous avons préféré les écarter du cadre de nos travaux.



Page de texte horizontale.

Figure 1



Page de texte inclinée.

$tg\ a_{max} = d/l$

Figure 2

B - IDENTIFICATION DES LIGNES DE TEXTE

1. - Localisation des lignes de texte

Détecter les lignes de texte lorsque l'on connaît leur orientation est un problème classique. Les solutions proposées font appel à la périodicité de la page de texte dans un sens perpendiculaire aux lignes de texte : celles-ci sont séparées par des interlignes (espaces ne contenant aucune information graphique) pour assurer une bonne lisibilité. Il suffit de localiser les interlignes pour savoir où se trouvent les lignes.

Notre méthode met en oeuvre un histogramme construit en comptant les noirs de chaque ligne digitale* de l'image ; l'interligne, dépourvu de message graphique, sera aisément reconnaissable à la valeur nulle de l'histogramme. Il faut, en réalité, tenir compte du bruit de fond introduit par le système d'acquisition ou dû à l'éclairement non uniforme du texte ; une comparaison avec un seuil peut être la solution à ce problème. La figure 1 illustre nos propos : on y remarque une page de texte et l'histogramme correspondant (à gauche).

La seule limite de cette solution se trouve dans l'erreur d'orientation du texte. La figure 2 montre une page de texte inclinée par rapport à la direction de discrétisation : le caractère périodique de l'histogramme disparaît pour des inclinaisons dépassant la valeur de seuil a_{max} définie dans la figure. Il est dès lors impossible de

---

* - Nous distinguons par la suite ligne digitale et ligne de texte, la première étant la ligne de discrétisation du système d'acquisition.

retrouver l'interligne par la méthode que nous avons décrite plus haut. Cette méthode reste cependant applicable si l'on réduit artificiellement la longueur de la ligne de texte par division en plusieurs morceaux de sorte que, pour une tolérance en inclinaison a_{max} donnée, la longueur l de tout morceau soit inférieure à $d/tg\ a_{max}$ où d est la hauteur de l'interligne; il faut toutefois assurer une longueur minimale à chaque morceau pour que l'histogramme obtenu soit significatif.

Signalons que le problème d'orientation est entièrement éliminé si l'on procède d'abord à la détection de l'orientation par la méthode esquissée en § A.2 sous le nom de spatiale et qui est basée sur la périodicité de l'histogramme de la page typographiée lorsque la projection a été faite parallèlement aux lignes de texte. De plus, par cette méthode l'erreur sur a est d'autant plus faible que la longueur l des lignes est grande, ce qui satisfait l'exigence d'une précision accrue dans de pareils cas :

$$tg\ a \leq d/l$$

2 - Structure d'une ligne de texte

La mise en évidence de la ligne de texte n'est malheureusement pas suffisante pour le lecteur optique. En effet, les primitives sont définies sur le corps de texte seulement ; une identification de celui-ci est absolument nécessaire.

Pour ce faire, Mason et Clemens /4.2/ mettent en oeuvre le même histogramme qui a été employé à la localisation des lignes de texte : la figure 3 montre l'histogramme correspondant à une seule

Discrimination du corps de texte par comparaison avec un seuil .

Figure 3

ligne de texte, comportant des minuscules avec hampes ; le corps de texte se détache du reste par une amplitude supérieure et la comparaison avec un seuil comme celui de la figure permet une mise en évidence facile.

Un inconvénient de cette méthode vient de la variation dans l'amplitude de l'histogramme d'une ligne de texte à l'autre (elle dépend du nombre de caractères sur la ligne mais aussi de la nature de ceux-ci : un i aura un apport plus faible qu'un w par exemple) ce qui implique un critère de choix du seuil. Il resterait ensuite à placer les sondes que nous avons introduites dans le chapitre III, problème indépendant du choix du seuil.

Lay et Bomsel /4.3/ ont montré que l'identification de la structure et le placement des sondes sont deux problèmes liés et qui peuvent être résolus en prenant comme point de départ l'information très riche contenue dans l'histogramme de la ligne de texte. Leur analyse est basée sur le fait que, malgré la diversité de leur aspect, les histogrammes présentent une structure commune. Ainsi, le corps de texte est encadré par deux pics principaux correpondant à sa base (alignement des empatements) et à son sommet (alignement des minuscules), séparés par un pic intermédiaire d'amplitude plus faible ou un plateau ; un plateau ou pic d'amplitude faible est visible au dessus du corps de texte si la ligne présente des hampes et en dessus si la ligne comporte des jambages. Les pics principaux sont dus à une épaisseur du trait plus grande que la normale provenant des empatements de certaines primitives, de leur concaténation (w, o, n, u, etc...) ou des barres (f, t, r). Le pic intermédiaire doit sa présence aux

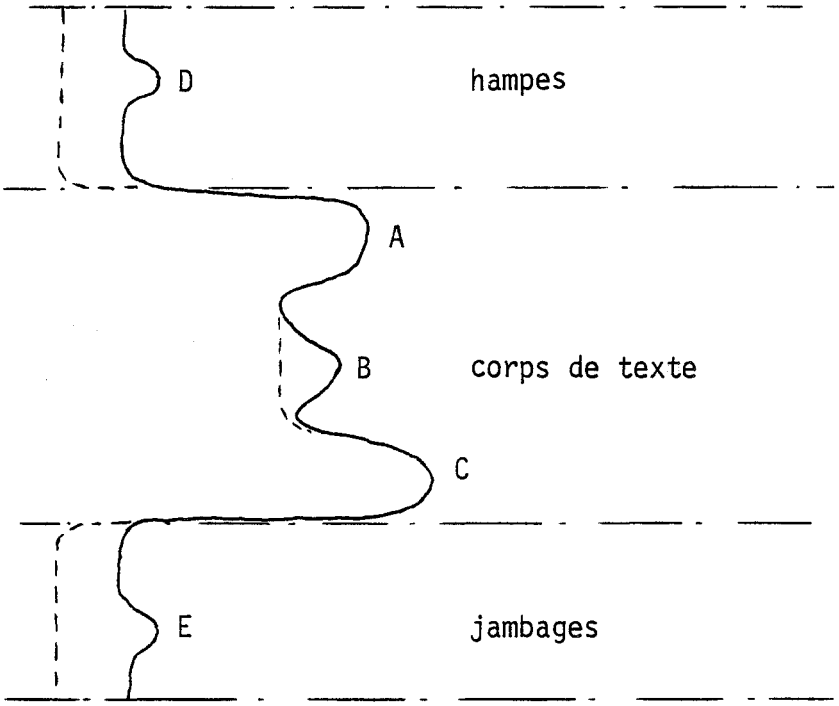
éléments transversaux à mi-hauteur (e,a) ou à une force accrue de la graisse de certaines primitives (s,(,) : l'épaisseur transversale est plus grande au milieu qu'aux extrémités).

Comme dans Mason et Clemens /4.2/, le cas d'une ligne composée seulement de majuscules est exclu parce que cette structure est entièrement bouleversée ; en effet la notion de corps de texte perd dans ce cas son sens.

La figure 4 explicite la structure que nous venons de discuter.

Dans le cas le plus fréquent le corps de texte présente trois pics séparés par deux creux. La théorie enseigne que plus les échantillons d'une épreuve sont dissemblables entre eux, plus ils contiennent d'information. Il faudra donc échantillonner la ligne de texte à la hauteur des pics et creux de l'histogramme, ce qui nous donne un total de cinq sondes pour le corps de texte. La pratique a montré que le nombre et la position des sondes, prédites par la théorie, étaient suffisantes pour la reconnaissance. La démarche intuitive que nous avons suivie lors du chapitre III concernant la compression de l'information sur le corps de texte se trouve ainsi justifiée. Précisons que cinq est le nombre minimum théorique de sondes sur le corps de texte ; pour une plus grande fiabilité il pourrait être augmenté dans une réalisation pratique.

Quelques primitives présentent des ascendants (1, 2, 5) ou descendants (2, 5) ; une sonde au-dessus du corps de texte et une autre en-dessous, placées à l'endroit correspondant aux pics secondaires



A : pic principal supérieur
B : pic intermédiaire
C : pic principal inférieur
D : pic secondaire supérieur
E : pic secondaire inférieur

Structure de l'histogramme d'une ligne de texte.

Figure 4

de l'histogramme, permettront de vérifier ces prolongements. Il existe un cas particulier de primitive qui ne présente pas de prolongements continus dans la région des hampes : i. Une sonde supplémentaire sera donc chargée de renseigner sur cette discontinuité ; elle sera placée dans le creux qui sépare le pic secondaire supérieur du pic principal supérieur.

Le problème d'identification de la structure d'une ligne de texte et le placement des sondes reposent donc sur l'identification des pics et de leur ordre d'importance dans l'histogramme correspondant. Pour y arriver, /4.3/ base son analyse sur la méthode de description des courbes par leurs pics et creux, proposée par Sankar et Rosenfeld /4.4/. En voici un résumé :

Un pic est jugé significatif s'il domine l'histogramme sur un intervalle de rayon r centré autour de lui. Pour une valeur très grande de r un seul pic de l'histogramme sera ainsi considéré significatif : le maximum maximorum. En diminuant progressivement r les autres pics sont mis en évidence par ordre de leur importance. Il y aura une valeur de r pour laquelle les pics principaux et les pics secondaires auront été repérés ; faire décroître encore r serait inutile parce qu'il n'y a pas d'autre information exploitable à faible coût dans l'histogramme.

Pour tous les cas particuliers (absence de pic intermédiare, secondaire supérieur ou inférieur) nous recommandons au lecteur de consulter l'ouvrage /4.3/ que nous avons repris tel quel dans le cadre de notre travail.

Il convient de remarquer que l'étude des histogrammes suivant un axe perpendiculaire aux lignes de texte est également utile pour la reconnaissance des caractères isolés (cf /4.6/, /4.7/). Mais ici nous nous refusons à considérer l'opération de segmentation a priori comme significative au niveau du caractère et nous la reportons au niveau de la primitive principale, comme l'a exposé le chapitre 3.

C - IDENTIFICATION AUTOMATIQUE DES PRIMITIVES

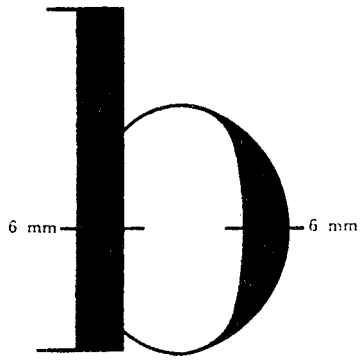
1 - Introduction

La dernière tâche du préprocesseur est de fournir à l'étage chargé de l'extraction des primitives les patrons qui serviront comme base de comparaison.

Dans le chapitre III nous avons défini les patrons par les abscisses de l'intersection des cinq sondes avec la forme graphique et par l'épaisseur de leur traversée. Nous avons aussi précisé que les cinq épaisseurs étaient égales entre elles. Ce fait n'est pas toujours vérifié dans la pratique. Ainsi, pour des exigences optiques, certaines classes de primitives sont plus épaisses que d'autres (figure 5a) ; quelquefois à l'intérieur de la même classe il y a des différences d'épaisseur entre deux primitives par ailleurs identiques (figure 5b). Il existe aussi des polices ayant deux épaisseurs principales du trait : les pleins et les déliés (figure 5c).

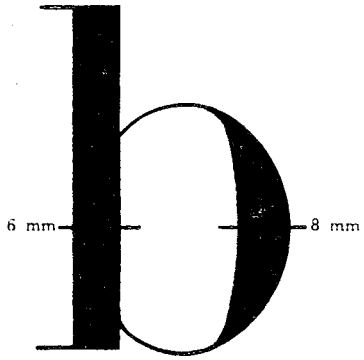
Toutefois ces phénomènes ne sont pas significatifs en première approximation et le fait d'utiliser une graisse uniforme pour toutes les primitives d'une même classe apporte au système une

MANQUE



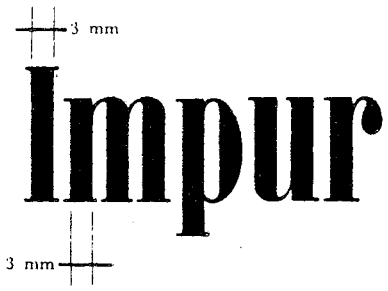
UNIFORMITÉ DE MESURE

RENFORCEMENT



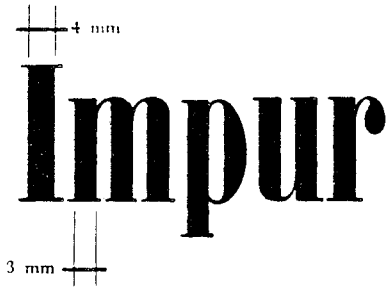
UNIFORMITÉ OPTIQUE

MANQUE



UNIFORMITÉ DE MESURE

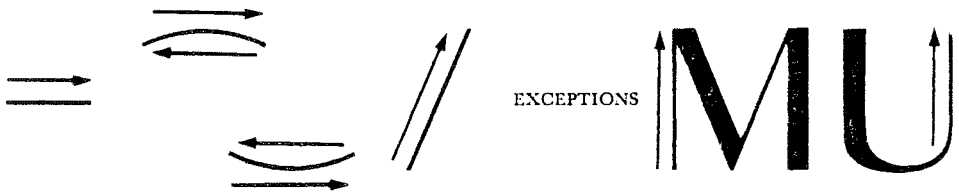
RENFORCEMENT



UNIFORMITÉ OPTIQUE

ORIENTATION DES PLEINS

ORIENTATION DES DÉLIÉS



Variation de l'épaisseur du trait à l'intérieur de la même police (tiré de /4.5/).

Figure 5

simplification qui dépasse en importance les erreurs qu'il entraîne. D'un autre côté on peut négliger sans trop de risque les différences entre les graisses des sept classes de primitives, exception faite pour la classe de la primitive 2 dans une police ayant des pleins et déliés. Par la suite, nous indiquerons une méthode pour trouver la graisse des déliés.

Une bonne estimation pour la graisse normale (l'épaisseur des primitives dans une police à graisse uniforme, l'épaisseur des pleins dans une police à pleins et déliés) s'est avérée être la moyenne des segments situés sur une des deux sondes placées dans les creux de l'histogramme sur le corps de texte. C'est en effet à ce niveau que le nombre de barres et d'empatements affectant les sondes est le plus faible, donc les segments retenus par les sondes considérées représentent presque toujours la section d'une primitive principale, égale à la graisse à cause de la faible proportion des déliés dans un texte statistiquement normal. La présence de déliés fausse théoriquement cette mesure mais sans l'invalider dans la pratique.

Parmi les primitives principales introduites au chapitre II, nous nous proposons d'identifier automatiquement les cinq suivantes : 5, 1, 2, 3, 4.

A ce stade d'avancement le système connaît :

- la position des lignes de texte;
- la structure de chaque ligne, donc les sondes;
- la graisse moyenne de la police;

et nous supposons que nous disposons librement du programme de recherche des primitives défini au chapitre III à ceci près que les modèles sont inconnus.

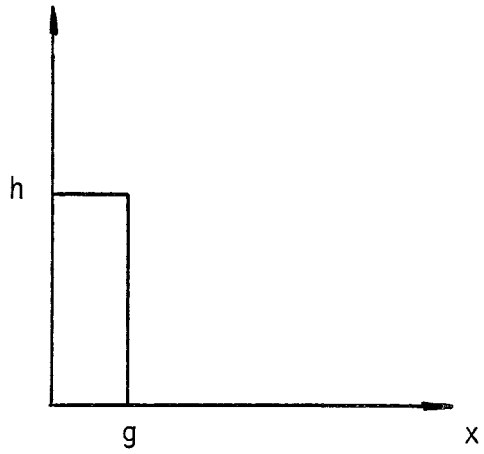
2 - Primitive 5

Le cas de la primitive 5 est trivial car nous avons décidé de ne pas tenir compte des empattements qu'elle peut présenter. En tenir compte aurait en effet multiplié les cas possibles et par suite les mesures de corrélation. Bien que la qualité des mesures en soit diminuée, le squelette de la primitive 5 s'est avéré suffisant. Le modèle est dès lors réduit à un rectangle de l'épaisseur de la graisse moyenne (figure 6).

3 - Primitives 3 et 4

Il s'agit de deux primitives entre lesquelles existe une relation de symétrie ponctuelle ; nous pouvons supposer en première approximation que cette symétrie est axiale, par rapport à un axe vertical. La connaissance de l'une apporte donc tous les renseignements demandés sur l'autre.

Pour faire un apprentissage de ces primitives, il faut d'abord identifier un caractère en contenant au moins une. Le caractère le plus facile à trouver à la lumière des propriétés énumérées plus haut est le o . En absence de toute information précise quant à sa représentation graphique, nous nous baserons sur sa description fonctionnelle en terme d'attributs essentiels : symétrie et fermeture (voir Shillman et al. /2.3/, Cox et al. /2.4/). Nous chercherons donc de façon systématique dans les lignes de texte les formes symétriques par rapport à un axe vertical (placé à mi-chemin entre deux segments successifs de la sonde 2). Pour chaque forme ainsi trouvée nous vérifions sa fermeture, test qui élimine les autres caractères symétriques



h : hauteur du corps de texte

g : graisse

P = {0, 0, 0, 0, 0} - début des sondes

PL= {g, g, g, g, g} - longueur des sondes

PS= {0, 0, 0, 0, 0} - côtés libres des sondes

Modèle de la primitive 5.

Figure 6

(v, x, h, etc...) ou les formes qui doivent leur symétrie au hasard : un i suivi d'un autre sont symétriques par rapport à un axe les séparant mais ne forment pas un caractère o .

Une analyse statistique des différents caractères retenus donne un modèle suffisant pour les primitives 3 et 4. Rappelons que la fréquence d'apparition du o dans un texte courant est de 8% en anglais et 5% en français, ce qui place la probabilité de ne trouver aucun o en 100 lettres à 0.02% en anglais et 0.6% en français. A la limite nous pourrions imposer la contrainte d'imprimer une ligne spéciale d'apprentissage en haut de la page.

4 - Primitives 1 et 2

Malheureusement le cas des primitives 1 et 2 ne peut pas être traité aussi simplement. La raison en est la grande variabilité de ces primitives et l'absence d'attributs fonctionnels aussi généraux que la symétrie ou la fermeture pour les caractériser.

Voici quelques unes de ces difficultés :

a) Dans les polices à pleins et déliés la primitive 2 est caractérisée par l'épaisseur faible alors que la primitive 1 a toujours la graisse normale ; une primitive 2 n'apparaît plus comme la symétrique de la primitive 1.

b) La pente des primitives 2 peut être plus accentuée que celle des primitives 1, de façon à compenser le déséquilibre provoqué par l'emploi des pleins et déliés dans une figure au départ symétrique.

c) Le rapport chasse-hauteur peut varier d'une police à l'autre. Il se situe aux alentours de 4/5 pour une bonne lisibilité (pour des caractères typiques : H, u) mais peut beaucoup varier dans des polices fantaisistes : voir la figure 7.

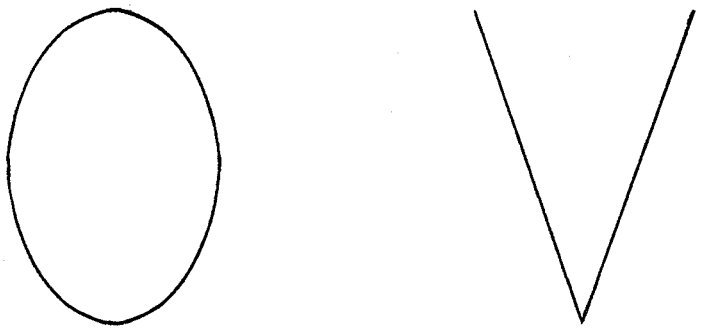
Nous n'avons pas encore trouvé de solution satisfaisante au problème b) mais notre système peut tenir compte des problèmes a) et c). Voici notre méthode :

Dans une police normale la pente des primitives 1 et 2 est d'environ 21°. Nous avons constaté expérimentalement que, quelque soit la police, les caractères o et v avaient la même hauteur et la même chasse (à moins de 5% près). Il est dès lors possible d'évaluer la pente d'une primitive 1 en partant de la connaissance des paramètres graphiques du o : suivant la figure 8, nous corrigeons la pente nominale de 21° par la formule :

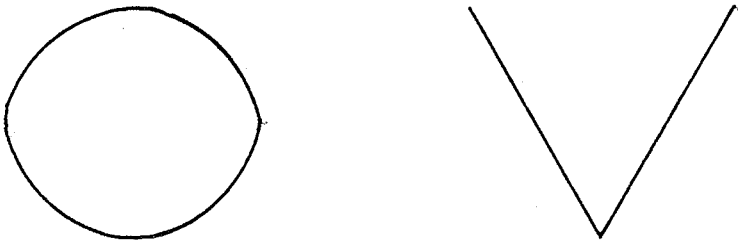
$$tga = \frac{l_o}{h_o} * tg\ 21°$$

où a est la nouvelle pente, l_o et h_o étant respectivement la largeur et la hauteur du o . Signalons que cette estimation est quelquefois mise en défaut par la perte d'information lors de l'échantillonnage ; en particulier nous ne connaissons pas la hauteur du corps de texte mais la distance entre les sondes 1 et 3.

L'idée de base de cette partie d'apprentissage est la suivante : fournir à l'étage d'extraction des primitives des modèles aussi précis que le permettent nos estimations de la primitives 1, lui



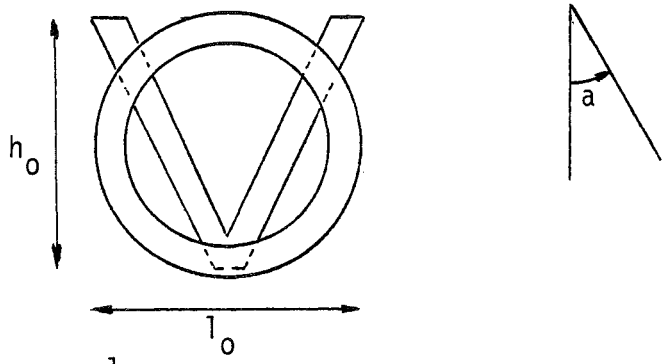
a) police étirée



b) police condensée

Variation du rapport chasse/hauteur

Figure 7



$$\text{tg } a = \frac{l_o}{h_o} \times \text{tg } 21°$$

Construction du modèle approximatif de la primitive 1

Figure 8

demander d'identifier la forme qui s'est rapprochée le plus d'un des modèles et faire les mesures précises de définition sur cette forme. Deux modèles peuvent être construits : le premier aura une pente de 21° (la pente de la primitive dans une police normale ; ceci se justifie par la très grande fréquence de telles polices), le deuxième aura comme pente l'estimation obtenue à partir de la comparaison o-v (figure 8).

Les primitives 2 que nous pouvons identifier ont la même pente (au signe près) que les primitives 1 mais peuvent avoir une graisse différente (si la police présente des déliés). La procédure d'extraction basée sur la corrélation pénalise fortement toute forme ayant une épaisseur plus faible que la graisse de la primitive ayant servi de modèle ; par contre corréler une forme avec une primitive d'épaisseur plus faible ne modifie pas le résultat de la vraie primitive représentée par la forme mais accroît le risque de confusion avec les autres primitives. Nous employons donc la même procédure que celle mise en oeuvre pour la primitive 1, mais avec, au départ, une épaisseur plus faible : la moitié de la graisse de la police s'est avérée être un bon paramètre de départ. Ce paramètre sera ensuite correctement estimé lors de l'identification de la meilleure primitive 2 trouvée avec nos modèles.

Nous devons remarquer qu'en abandonnant l'hypothèse de symétrie entre pente de 1 et pente de 2 nous pouvons encore essayer d'identifier une primitive 2 dans la mesure où toute primitive 1 est statistiquement souvent suivie d'une primitive 2 dans le corps de texte.

5 - Primitives 6 et 7

La grande variation dans la forme graphique des primitives 6 et 7 nous a empêché de trouver un modèle théorique. Rappelons que, selon les polices, l'inclinaison de la primitive 7 peut varier entre 21° (primitive 2) et 45°, alors que la primitive 6 peut présenter des courbures allant de faibles à très prononcées.

Nous nous sommes cependant aperçu à l'expérience que la primitive 6 ne demandait pas une grande précision au niveau de la définition du modèle pour être bien reconnue, mais pour l'instant nous n'avons pas essayé d'explorer cette hypothèse.

6 - Espaces

Le préprocesseur inclut une tâche secondaire, chargée d'identifier les blancs entre les mots. Pour nous ils dépassent en importance le simple fait qu'ils constituent un caractère comme les autres (code ASCII '20). En effet, l'étape de reconstitution des caractères repose sur la génération d'hypothèses de segmentation possibles de la chaîne de primitives ; un certain nombre de ces hypothèses sont absurdes et seront éliminées par la suite. Pour assurer de bonnes performances ces hypothèses doivent être générées dans la quantité la plus restreinte possible. Les blancs séparateurs entre les mots forment des sortes de goulots d'étranglement parce qu'ils obligent le générateur d'hypothèse à cesser son travail pour un mot donné ; sans eux, le texte serait reconstitué tout de même mais après avoir passé en revue beaucoup d'hypothèses inutiles.

Un histogramme est encore à la base de notre méthode. Mais au lieu de le construire de toute pièces nous avons tenu compte du fait qu'à ce niveau de l'analyse les sondes ont été déjà placées ; Une projection des sondes sur une droite horizontale (à la place de la projection de la ligne de texte entière) nous fournira à peu de frais la position des espaces. En effet, les interstices qui séparent les segments ainsi formés (figure 9) peuvent être de deux types : espaces entre les caractères d'un mot et espaces entre les mots. Un histogramme de ces interstices en fonction de leur longueur fait apparaître deux classes, très bien séparées dans tous les cas que nous avons analysés.

D - ILLUSTRATION

L'exemple qui nous servira comme base de discussion est celui de la figure 9.

Sur cette figure l'image de départ (3 lignes de texte) est représentée en une teinte grisée. Les informations trouvées par le préprocesseur sont visualisées en noir. Nous trouvons ainsi :

- en début de chaque ligne l'histogramme correspondant.

- les segments générés par l'échantillonnage de l'image initiale avec les 8 sondes, placées en fonction de l'histogramme de la ligne. Il est facile de constater que la stratégie de placement des sondes a répondu aux exigences imposées : les sondes 1, 2 et 3 ont souvent des barres, renseignant ainsi sur une possible concaténation ; les sondes 4 et 5 sont exemptes de barres, ce qui justifie notre choix

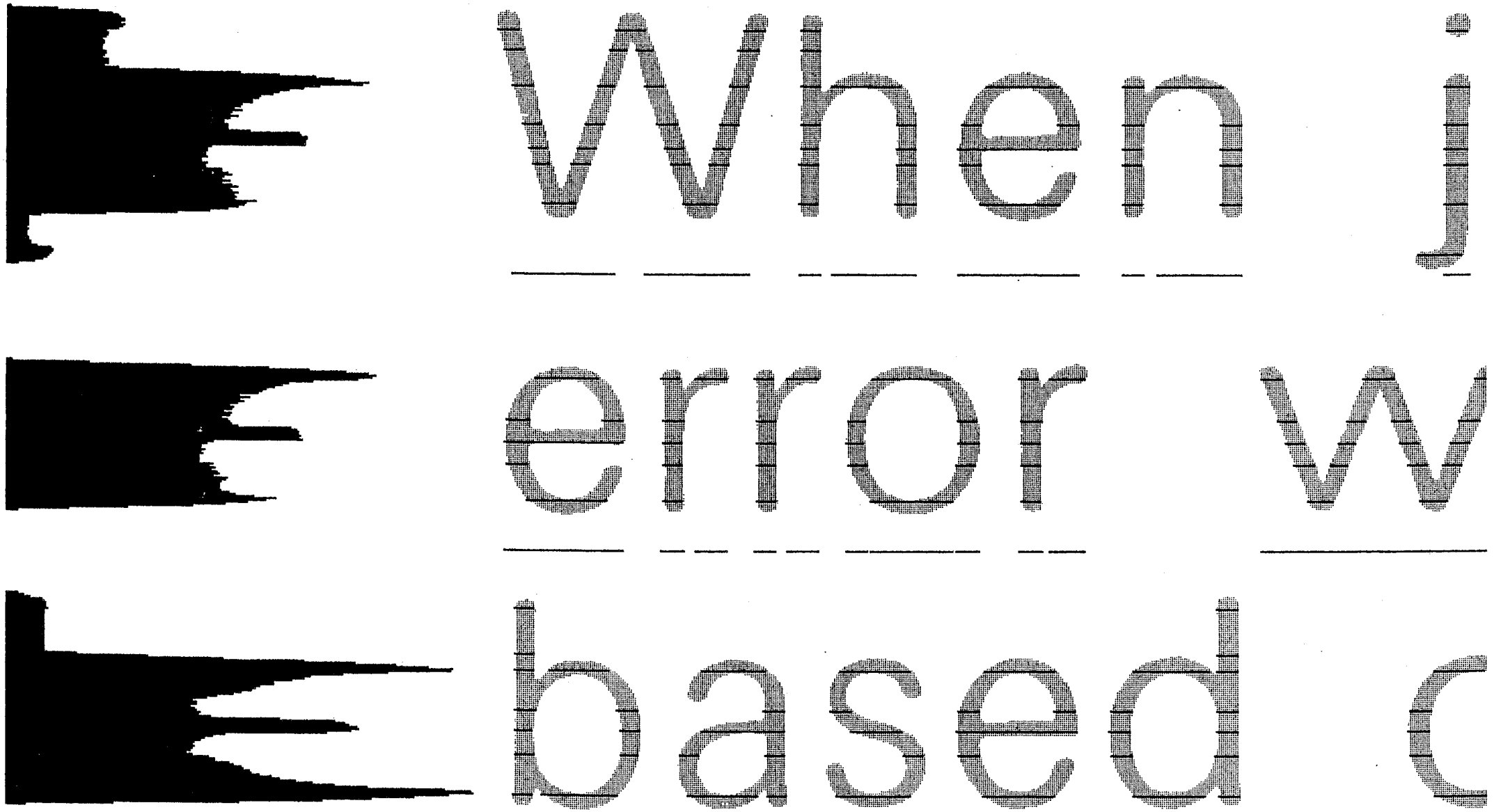


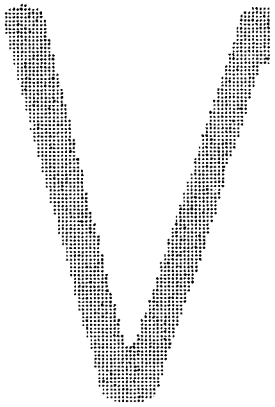
Figure 9

pour le calcul de la graisse.

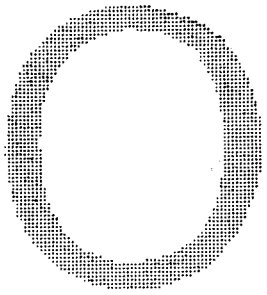
- en-dessous de chaque ligne de texte la projection horizontale des cinq sondes du corps de texte. La distance entre les segments ainsi construits sert à la détection des espaces.

L'identification automatique des primitives 3 et 1 selon la stratégie décrite a fourni les résultats de la figure 10.

Ces images sont stockées en mémoire de masse ; à chaque ligne de texte nouvelle elles sont échantillonnées à la hauteur des sondes correspondant à cette ligne pour fournir ainsi les patrons des primitives (de cette façon, les différences entre la position des sondes d'échantillonnage d'une ligne à l'autre sont rattrappées automatiquement).



Modèle de la primitive 1.



Modèle de la primitive 3

Ces formes sont échantillonnées à chaque changement de ligne, le premier segment sur chaque sonde étant pris comme modèle de la primitive respective.

Figure 10

V - RECOMBINAISON DES PRIMITIVES EN CARACTERES

A - INTRODUCTION

1 - Problème

2 - Principe de la méthode

B - SEGMENTATION

C - IDENTIFICATION DU CARACTERE DANS UN SEGMENT

D - RECOMBINAISON

E - ILLUSTRATION

RECOMBINAISON DES PRIMITIVES EN CARACTERES

A - INTRODUCTION

1 - Problème

Ce chapitre a pour objet la description de la dernière couche de notre système de lecture optique. Cette partie du système, dont la figure 1 montre le schéma général, se trouve en aval de l'étage d'extraction des primitives. Son entrée est constituée par les données de sortie de l'étage extracteur, à savoir une suite de primitives principales entre lesquelles se trouvent des primitives secondaires. Cette suite est supposée segmentée en mots par la reconnaissance des blancs les séparant. Sa sortie représente le texte reconstitué à partir des primitives, sous forme codée (ASCII dans notre cas). Les règles de passage d'une représentation à l'autre du texte (des primitives au code ASCII) sont fournies par une grammaire, chargée en plus d'évaluer en terme de probabilité la combinaison de primitives et capable de corriger quelques erreurs transmises par l'étage extracteur.

2 - Principe de la méthode

La solution mise en oeuvre pour atteindre le but fixé se décompose en trois parties :

1° génération d'hypothèses découpant la chaîne de primitives en segments.

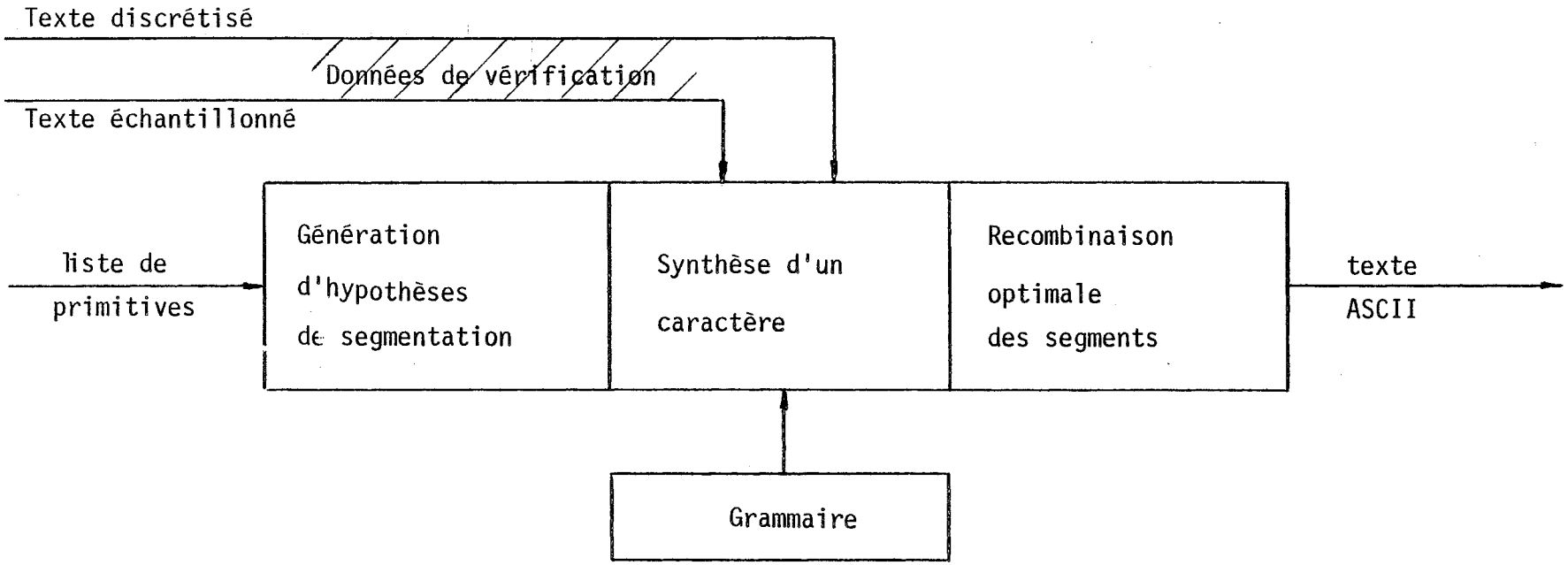


Schéma général de l'étage de recombinaison

Figure 1

2° identification du caractère dans un segment (avec évaluation de sa probabilité d'existence en fonction des primitives le composant).

3° recombinaison optimale des segments pour former le texte final.

La première partie a pour but de fournir pour chaque mot une segmentation plausible de la chaîne d'entrée de sorte que chaque segment ne contienne pas plus d'un caractère ; il y aura donc entre une et quatre primitives par segment (les caractères les plus "courts" sont c, e, f, i, j etc..., les plus "longs" étant M, W, w). Une analyse exhaustive de toutes les combinaisons de segments possibles est exclue au départ à cause de leur nombre exagérément élevé (à titre d'exemple un mot de 10 primitives peut se décomposer de 400 façons différentes en segments d'au plus quatre primitives). Nous décrirons plus loin une méthode de génération dynamique d'un ensemble d'hypothèses de segmentation comprenant, outre la vraie segmentation, un certain nombre d'autres hypothèses, en nombre réduit de sorte que le coût en temps de traitement ne devienne pas prohibitif. Le compromis ainsi atteint évite donc d'engager la reconnaissance des caractères suivant une segmentation a priori unique. L'hypothèse correcte se trouve enserrée dans un faisceau d'hypothèses plausibles, ce qui assure une plus grande fiabilité à la segmentation et, partant, à la reconnaissance. Cette méthode n'a évidemment de valeur que s'il est facile de découvrir a posteriori la meilleure segmentation dans le faisceau des hypothèses retenues. Tel est effectivement le cas dans notre application, où une erreur de décodage tend à en entraîner d'autres, réduisant à rien la plausabilité initiale des hypothèses erronées.

La seconde partie est chargée de travailler sur les primitives d'un seul segment à la fois. Sa tâche est de préciser le caractère représenté par le segment et de l'accompagner d'une estimation concernant la combinaison de primitives le composant. Cette évaluation est nécessaire pour distinguer les bonnes combinaisons des moins bonnes ou absurdes (l'on sait qu'un certain nombre d'hypothèses sont erronnées mais ceci n'est pas connu au moment de leur émission) ; elle permet aussi de corriger des erreurs concernant les primitives, transmises par l'étage extracteur : **omission** d'une primitive (parce que la meilleure probabilité était encore assez faible pour être prise en compte) ou **substitution** (la forme était assez ambiguë pour favoriser deux primitives, leur ordre étant souvent inversé). Dans le premier cas la décision est prise de rajouter une primitive pour compléter le segment (donc le caractère), alors que dans le deuxième il faut faire le choix entre les deux primitives gardées pour la forme (le lecteur se rappelle que les deux meilleures décisions au lieu d'une sont stockée pour chaque forme) ; dans les deux cas la confiance dans le caractère ainsi formé est diminuée pour préciser que la combinaison des primitives dans le segment n'était pas parfaite. C'est au niveau de cette partie qu'intervient la grammaire de caractères dont nous avons fait état plus haut.

La troisième partie travaille sur l'ensemble des segments ; son objet est de séparer dynamiquement les bonnes hypothèses des mauvaises. Pour y arriver elle calcule de façon incrémentale une estimation globale de tous les segments compatibles* couvrant de proche en proche l'ensemble de la chaîne de primitives ; sera retenu l'ensemble de segments dont l'estimation est la meilleure.

---

* Nous appelons compatibles deux segments ayant une frontière commune.

D'une manière plus formelle :

- la première partie doit produire des segmentations du type :

$$S^k = s_1^k, s_2^k, \ldots s_n^k ,$$

où s_i^k précède s_{i+1}^k et les deux segments sont compatibles, S^k couvrant en entier la chaîne des primitives (k indique la k-ième segmentation possible) ;

- la deuxième partie associe à chaque segment s_i^k le couple (c_i^k, p_i^k), où c_i^k est le caractère représenté par le segment s_i^k, p_i^k étant l'estimation de la vraisemblance entre s_i^k et c_i^k ;

- la troisième partie associe à S^k le couple (C^k, P^k), où :

$$C^k = c_1^k, c_2^k, \ldots, c_n^k ,$$

$$P^k = p_1^k \circ p_2^k \circ \ldots \circ p_n^k,$$

o étant une loi de composition que nous définirons plus tard. Elle fournit comme résultat le couple (C^k, P^k) pour lequel P^k est maximum.

Un travail semblable au nôtre a été réalisé par S. Peleg /5.1/ sur la reconnaissance de l'écriture manuscrite ; il s'agit d'une méthode de segmentation suivie du calcul d'un facteur de mérite de chaque segment et d'un facteur de mérite global de la segmentation.

Une approche similaire a été proposée par P. Mermelstein /5.2/ pour la reconnaissance de la parole continue : le rôle de la primitive, plus petit élément identifiable a priori, est alors joué par la syllabe, le caractère, que l'on cherche à isoler et à identifier, devenant le mot et le mot, unité macroscopique du texte devenant la clause, unité macroscopique du discours.

B - SEGMENTATION

Notre méthode de segmentation est basée sur la remarque fondamentale suivante : bien que le nombre théorique de combinaisons des primitives principales et de leurs dérivées (primitives principales avec hampes et/ou jambages) soit très grand, il y a peu d'associations admises dans la pratique.

Dans la suite nous appellerons :

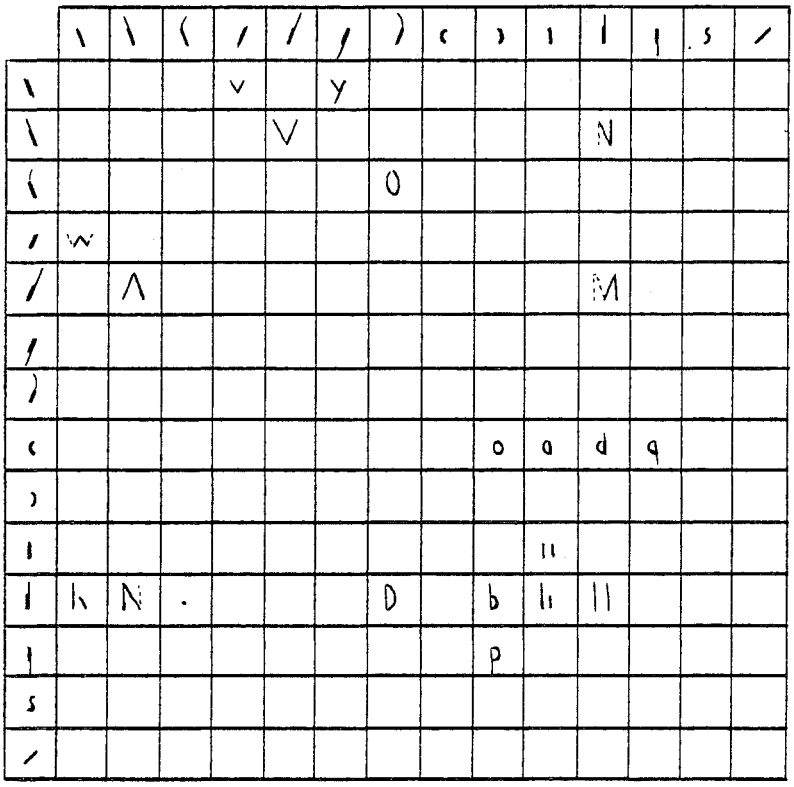
- digramme l'association de deux primitives principales ;
- trigramme l'association de trois primitives pricipales ;
- quadrigramme l'association de quatre primitives principales.

Les caractères les plus "longs" contenant quatre primitives, il n'y a pas d'association possible supérieure à quatre.

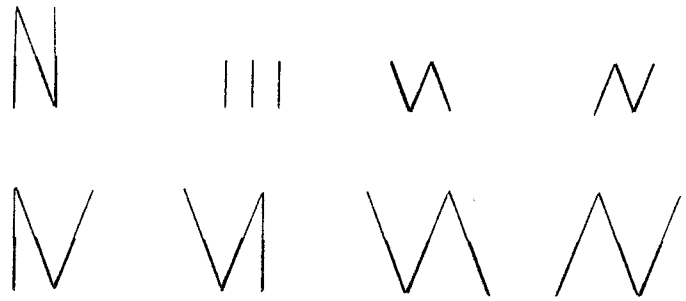
La figure 2a montre le tableau des digrammes ; sur les 196 cas possibles il n'y a que 20 combinaisons admises, soit environ 10 %.

La figure 2b montre les trigrammes légaux et 2c les quadrigrammes admissibles ; les proportions sont donc de $8/14^3$ (moins de 0,3%) pour les trigrammes et de $3/14^4$ (environ 8×10^{-5}) pour les quadrigrammes.

D'un autre côté plus les caractères contiennent des primitives moins ils sont nombreux : parmi les minuscules (de loin les plus fréquentes), il y en a 11 composés d'une seule primitive (a, c, e, f, i, j, l, r, s, t, z), 13 de deux primitives (b, d, g, h, k, n, o, p, q, u, v, x, y), 1 de 3 primitives (m) et 1 de quatre primitives (w).



a) digrammes légaux



b) trigrammes légaux

c) quadrigrammes légaux

N-grammes légaux (corps de texte en trait gras).

Figure 2

Il est donc évident que la chance pour qu'une combinaison de primitives successives représente un caractère décroît très vite avec leur nombre. C'est sur cette observation que nous avons bâti le générateur d'hypothèses.

Une telle situation, particulièrement favorable, ne se rencontre pas toujours lors des autres applications de cette méthode ; dans la reconnaissance de la parole par exemple la compatibilité des syllabes successives est beaucoup plus forte.

Il est évidemment impossible de pratiquer à coup sûr les coupures qui engendrent les bons segments ; nous contournons la difficulté en nous contentant de savoir que dans un bloc de n primitives successives il y a au moins une coupure. En vertu de ce qui a été dit plus haut, plus le bloc est long, plus la probabilité d'avoir une coupure est grande, mais malheureusement moins l'on sait où elle se trouve à l'intérieur du bloc.

Pour n=5 la probabilité d'avoir une coupure au moins devient certitude ; il y a quatre coupures possibles, chacune pouvant exister ou non ; il faudrait donc analyser 2^4 hypothèses pour déterminer au maximum 4 coupures, ce qui nous donne, dans le meilleur des cas, une efficacité de $4/2^4 = 0.25$.

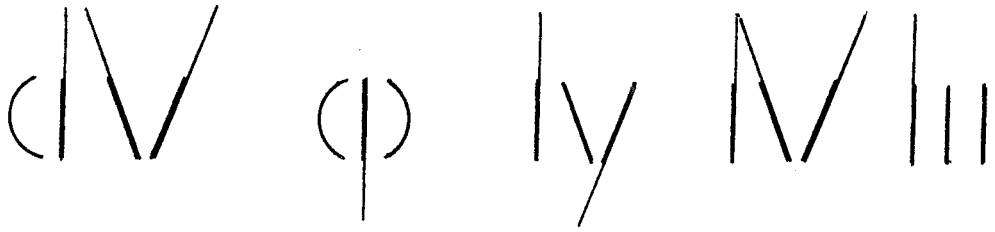
Pour $n \leq 4$ la certitude d'avoir une coupure demeure dans le cas où le bloc de n primitives ne fait pas partie du n-gramme correspondant. Dans ces cas nous avons une efficacité de $3/2^3 = 0.37$ pour des blocs de quatre primitives, $2/2^2 = 0.5$ pour des blocs de trois

primitives. Le bloc de deux primitives demande une analyse plus détaillée. Dans ce cas la coupure, si elle existe, est parfaitement localisée. Cependant, la fréquence relativement élevée des digrammes rend très important le risque de générer des segments comportant plus d'un caractère, que l'on appellera **frises** par la suite, composés de primitives compatibles deux à deux, comme ceux de la figure 3.

Ce risque est beaucoup diminué par une segmentation basée sur trigrammes. Dans ce cas l'ensemble des frises admises est un sous-ensemble des frises générées par digrammes et se compose des 5 classes de la figure 4 dont les trois premières sont relativement plus fréquentes (**m** , **w** , **W**).

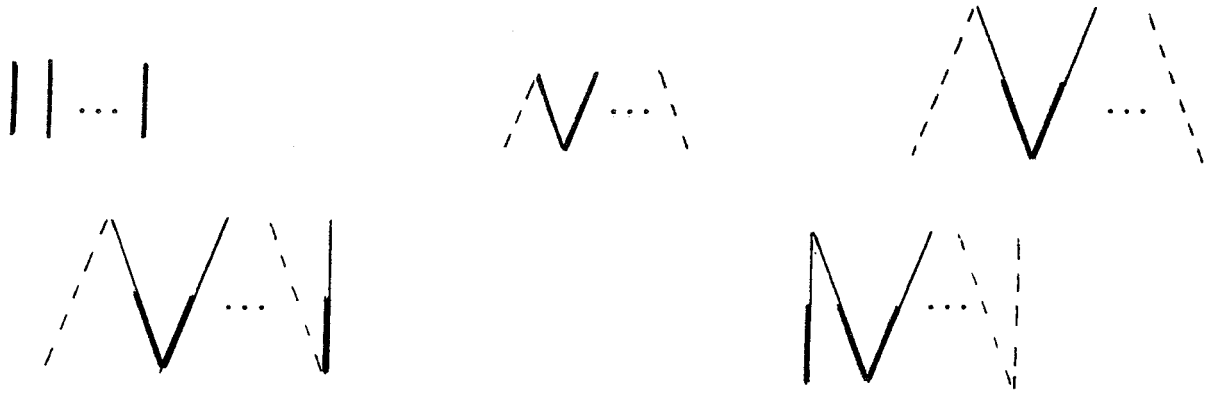
Il nous a paru préférable de baser notre segmentation sur les trigrammes, moins précises en ce qui concerne la localisation des coupures mais réduisant de beaucoup le nombre et le risque de frises, plutôt que sur les digrammes, très précises quant à la localisation de certaines coupures mais générant beaucoup de frises dont l'analyse est malaisée parce que fondée exclusivement sur les primitives secondaires ou sur la base de données brutes concernant les sondes.

Analysons en détail un bloc de trois primitives p_i p_{i+1} p_{i+2}, incompatible au sens des trigrammes ; il y a donc au moins une coupure, placée à droite de la primitive p_i (nous allons la noter X_i) ou à droite de p_{i+1} (notée X_{i+1}) ou bien les deux en même temps. Nous représenterons cette situation par le diagramme de la figure 5 ; les deux états en haut indiquent la présence de la coupure respective, les états inférieurs indiquant son absence. Ces états sont reliés par des arcs représentant des segments possibles, commençant ou finissant sur une primitive du bloc.



Exemples de frises générées par digrammes.

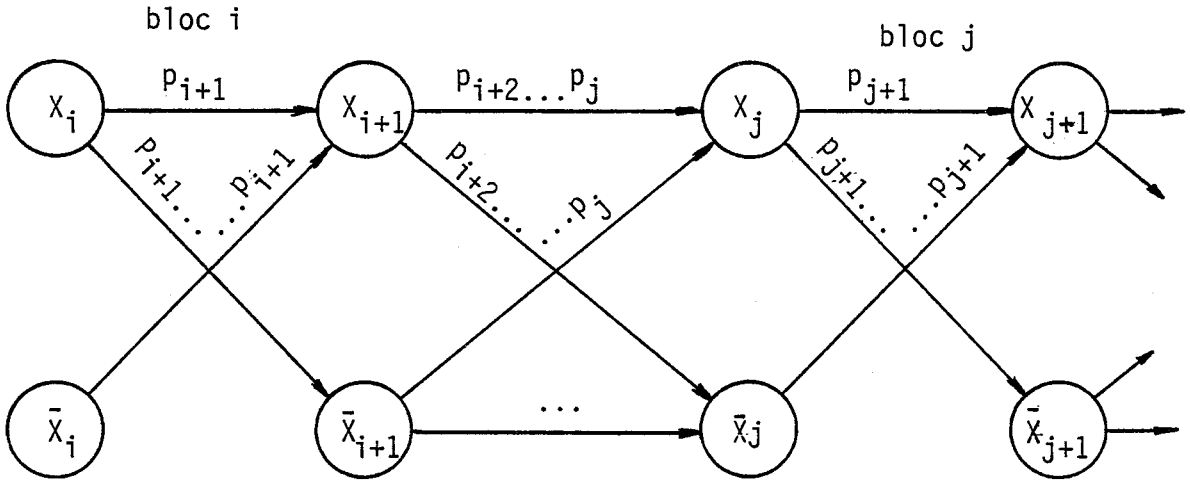
Figure 3



Frises admises par les trigrammes

(les primitives en pointillés sont optionnelles, les points indiquent la répétition indéterminée du motif formé par les 2 primitives précédentes)

Figure 4

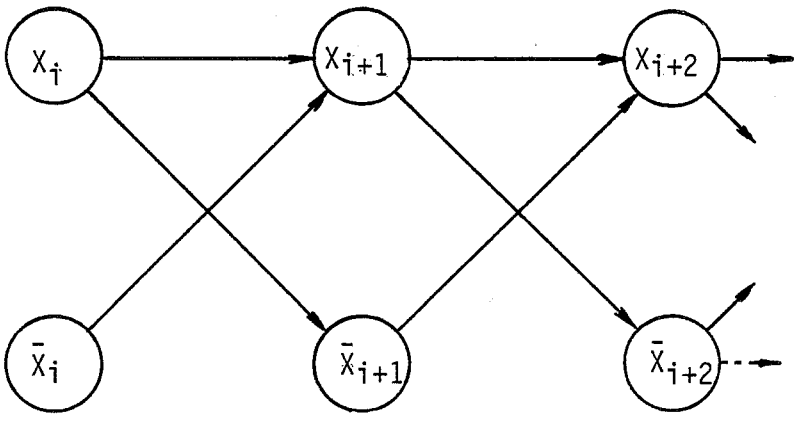


coupures	X_i X_{i+1}	X_j X_{j+1}
primitives	p_i p_{i+1} p_{i+2}	p_j p_{j+1} p_{j+2}

Graphe orienté des segmentations possibles

Cas général (j>i+1)

Figure 5



Cas particulier du graphe de la figure 5
j=i+1

Figure 6

Ainsi l'arc $X_i\ X_{i+1}$ indique un segment délimité par les deux coupures X_i et X_{i+1} et formé par la seule primitive p_{i+1} ; l'arc $X_i\ \bar{X}_{i+1}$ indique un segment ayant l'origine dans le bloc i, mais son extrémité doit être cherchée dans le bloc suivant ; le segment $\bar{X}_i\ X_{i+1}$ indique l'absence de la coupure X_i et dans ce cas la coupure X_{i+1} est obligatoire (il faut au moins une coupure dans ce bloc). Cette structure périodique doit être répétée autant de fois qu'il y a des blocs nécessitant des coupures (illégaux comme trigrammes). Dans la figure 5 un autre bloc, j, a été représenté pour montrer les liaisons qui s'établissent de l'un à l'autre.

Nous commençons déjà à voir l'intérêt de la méthode et son aspect dynamique : l'analyse démarrant sur un noeud supérieur (X_{i+1} par exemple) est indépendante des chemins y aboutissant ; de tous ces chemins nous retiendrons celui qui a la meilleure estimation. Les noeuds supérieurs constituent ainsi des goulots d'étranglement qui rendent la méthode attrayante parce qu'ils décomposent les chemins globaux en sous-chemins indépendants. Une décision globale n'est donc plus le produit mais la somme de décisions partielles portant sur des ensembles d'hypothèses restreints. Par analogie avec des méthodes connues de calcul rapide, nous dirons que la complexité de l'optimisation a été diminuée de façon logarithmique.

Il reste à régler la question concernant la définition des blocs, c'est-à-dire la manière dont i et j ont été trouvés. D'après ce qui a été dit plus haut, une seule solution est possible : balayer la chaîne des primitives avec une fenêtre de largeur 3 et passer d'un bloc à l'autre en avançant d'une primitive. Dans le cas où deux blocs successifs sont tous les deux illégaux au sens trigramme, plusieurs

chemins dans le graphe de la figure 5 se trouvent confondus : si $j=i+1$ alors le chemin $X_{i+1}\ X_j$ définit un segment vide, les chemins $X_i\ \overline{X}_{i+1}\ X_j$ et $X_i\ X_{i+1}$ définissent le même segment, de même que $X_{i+1}\ \overline{X}_j\ X_{j+1}$ et $X_j\ X_{j+1}$; le graphe de la figure 5 se transforme ainsi en un autre, représenté à la figure 6.

C - IDENTIFICATION DU CARACTERE DANS UN SEGMENT

L'hypothèse selon laquelle un segment engendré d'après la procédure décrite plus haut contient un seul caractère est vérifiée en pratique dans presque la totalité des cas. Dans la théorie, il faut traiter différemment les segments de 1 et 2 primitives des segments de 3 primitives et plus, les derniers pouvant être des frises. Le traitement des frises nécessite la division du segment en sous-segments pour satisfaire à l'impératif d'un caractère par segment ; cette nouvelle segmentation ne pouvant pas être faite sur la base des primitives principales, doit être effectuée en s'appuyant sur les informations contenues dans les primitives secondaires (heurtoirs, points, barres) ainsi que sur l'absence de liaison entre les primitives principales (ce qui peut indiquer une coupure possible). Dans la suite nous envisagerons seulement les cas des segments sans frise (ce qui est justifié par la fréquence moindre de cette dernière possibilité dans une segmentation).

Plusieurs cas sont envisageables :

a) le segment est complet : toutes les primitives d'un caractère sont présentes.

b) le segment est incomplet.

c) le segment est absurde.

L'identification de chaque segment est faite par une grammaire. Supposons dans un premier temps que l'étage extracteur est parfait : il reconnaît toutes les primitives sans en oublier aucune. Dans ce cas la grammaire doit être capable de classer des segments du type a) et rejeter ceux du type b) et c). Pour ce faire sa structure peut être très simple : une table contient la description de chaque caractère sous forme de primitives principales ; une recherche du segment en cours d'analyse dans cette table fournit un compte-rendu positif s'il s'y trouve et négatif dans le cas contraire. Dans cette dernière hypothèse le segment est rejeté parce qu'appartenant au type b) ou c).

Dans le premier cas le compte-rendu indique une classe de confusions regroupant tous les caractères ayant la même décomposition en primitives. Une analyse plus détaillée est nécessaire pour bien séparer les éléments de la classe. Contrairement aux méthodes classiques de lecture optique, nous ne considérons pas la présence de classes de confusion comme une infirmité de notre modèle de description des caractères. La structure de celui-ci est telle que leur confusion est inévitable à un certain niveau de reconnaissance. Ce niveau est celui qui nous sert à segmenter les caractères entre eux et repose sur leur description en terme de primitives principales. A l'intérieur de chaque classe de confusion, dont la théorie définit les membres a priori, l'analyse sera basée sur les primitives secondaires (heurtoirs, barres, points). Un exemple : e et c ont la même décomposition en primitives principales : 3 ; les deux caractères se trouveront donc dans la même classe et leur séparation sera faite sur la base des primitives secondaires : heurtoir sur la sonde 1 et barre sur la sonde 2 pour

e , rien pour c . Il n'est pas exclu de devoir recourir à ce niveau de l'analyse à des vérifications directes sur les bases de données brutes representant le mot sous forme d'une matrice de points ou d'une liste de sondes (les **données de vérification** sur la figure 1).

Malheureusement, l'étage extracteur n'est pas parfait. Il laisse passer des erreurs du type **omission** (une primitive est éliminée parce qu'ayant une probabilité trop faible) ou **substitution** (l'ordre des deux meilleurs choix est inversé parce que la forme est ambiguë). Dans ce cas quelques-uns des segments de type b) et c) peuvent représenter un caractère et doivent donc être corrigés.

Pour y arriver nous allons introduire dans la table de la grammaire la description incomplète (pour les omissions) ou modifiée (pour les substitutions) des caractères ; nous allons aussi fournir des pouvoirs plus grands à l'étage chargé de faire l'analyse détaillée du segment. Il commencera par assigner à tout segment une probabilité initiale de 100 %. Pour chaque élément qui manque (primitive principale ou secondaire) il retire de cette confiance une certaine quantité, définie empiriquement, en fonction de l'importance de l'élément manquant.

Dans le cas où il y a substitution, il vérifie qu'il y a bien une compatibilité entre les primitives qui ont changé de place et rejette les substitutions absurdes : une primitive ayant des ascendants ne peut pas se transformer en une autre avec des descendants par exemple. Dans ce cas aussi la confiance de départ sera diminuée.

En conclusion, cet étage fournit comme résultat le code ASCII d'un caractère, accompagné de la confiance que l'on a en la combinaison des primitives le composant.

D - RECOMBINAISON

A ce niveau de l'analyse nous possédons une estimation de la confiance que l'on peut avoir en chaque segment du graphe de la figure 5. Le texte final est représenté par un certain chemin dans ce graphe. Nous avons précisé plus avant que les chemins partant d'un noeud supérieur étaient indépendants des chemins arrivant à ce noeud. Une estimation du noeud est possible en fonction des chemins y aboutissant : ce sera l'estimation du chemin ayant fait le meilleur score. A partir de ce moment les autres chemins sont éliminés : c'est ce que nous avons appelé goulot d'étranglement. D'après ce que nous venons de dire, une procédure de composition des segments est nécessaire.

Soit X_{n-1} et X_n deux noeuds, $X_{n-1}\ X_n$ le segment défini par eux, $p(X_{n-1})$ et $p(X_n)$ l'estimation des noeuds (donc des meilleurs chemins arrivant à eux) et $p(X_{n-1}\ X_n)$ l'estimation du segment $X_{n-1}\ X_n$. Dans ces conditions nous avons la relation récursive (1) :

$$(1) \qquad p(X_n) = p(X_{n-1}) \circ p(X_{n-1}\ X_n)$$

o indiquant une loi de composition que nous définirons par la suite.

Soit aussi l_{n-1}, l_n et $l_{n-1\ n}$ les longueurs des chaînes de primitives arrivant au noeud n-1, n et respectivement la longueur du segment $X_{n-1}\ X_n$.

Nous avons voulu, pour des raisons évidentes, que la loi de composition o soit associative. La fonction qui nous a semblé la plus efficace à l'expérience est la suivante :

$$(2)\quad p(X_n) = \frac{p(X_{n-1})*l_{n-1} + p(X_{n-1}\ X_n)*l_{n-1\ n}}{l_{n-1} + l_{n-1\ n}}$$

Son efficacité décroît cependant avec la longueur de la chaîne totale. En effet, la variation de $p(X_n)$ en fonction de $p(X_{n-1}\ X_n)$ est :

$$\Delta p(X_n) = \Delta\ p(X_{n-1}\ X_n) * \frac{l_{n-1\ n}}{l_{n-1} + l_{n-1\ n}}$$

et dans le cas $l_{n-1} >> l_{n-1\ n}$:

$$\Delta p(X_n)_{max} = \frac{l_{n-1\ n}}{l_{n-1}+l_{n-1\ n}} \simeq \frac{l_{n-1\ n}}{l_{n-1}} << 1,$$

En passant donc de l'extrème $p(X_{n-1}\ X_n) = 0$ à l'extrême $p(X_{n-1}\ X_n) = 1$ la variation de la probabilité finale reste faible.

Le remède consiste en une limitation de la longueur de la chaîne analysée, d'où l'intérêt d'une analyse par mots, ce qui implique la détection des blancs séparateurs.

E - ILLUSTRATION

Nous donnons dans la suite le détail de l'analyse concernant la première ligne du texte nous ayant servi de base de discussion lors des chapitres précédents. La figure 7 reprend cet exemple, avec, au-dessus de chaque primitive, son numéro d'ordre (en début de ligne

PRIM	ATTR	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
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
1(1)	3(3)	84(82)	1(1)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	1(1)	-2(4)	40(42)	-1	0
2(2)	3(3)	86(84)	3(3)	3(3)	1(1)	3(3)	3(3)	3(3)	0(0)	3(3)	2(-1)	65(63)	-1	0
1(1)	3(3)	94(90)	5(5)	5(5)	3(3)	5(5)	5(5)	3(3)	0(0)	5(5)	2(-2)	108(107)	-1	0
2(2)	3(3)	86(84)	7(7)	7(7)	3(3)	7(7)	7(7)	5(5)	0(0)	7(7)	-4(2)	130(132)	-1	0
5(5)	3(3)	84(84)	9(9)	9(9)	5(5)	9(9)	9(9)	7(7)	0(0)	9(9)	2(-3)	175(173)	-1	1
5(5)	0(0)	80(76)	11(11)	11(11)	7(7)	11(11)	11(11)	0(0)	0(0)	0(0)	-5(4)	221(222)	-1	0
3(3)	0(0)	94(94)	13(13)	13(13)	9(9)	13(13)	13(13)	0(0)	0(0)	0(0)	4(-4)	253(252)	-1	4
5(5)	0(0)	94(90)	15(15)	15(15)	11(11)	17(17)	15(15)	0(0)	0(0)	0(0)	2(-3)	333(332)	-1	1
5(5)	0(0)	82(82)	17(17)	17(17)	13(13)	19(19)	17(17)	0(0)	0(0)	0(0)	3(-5)	381(381)	-1	0
5(5)	5(5)	100(100)	19(19)	19(19)	15(15)	21(21)	19(19)	9(9)	1(1)	0(0)	2(-4)	491(492)	-1	0

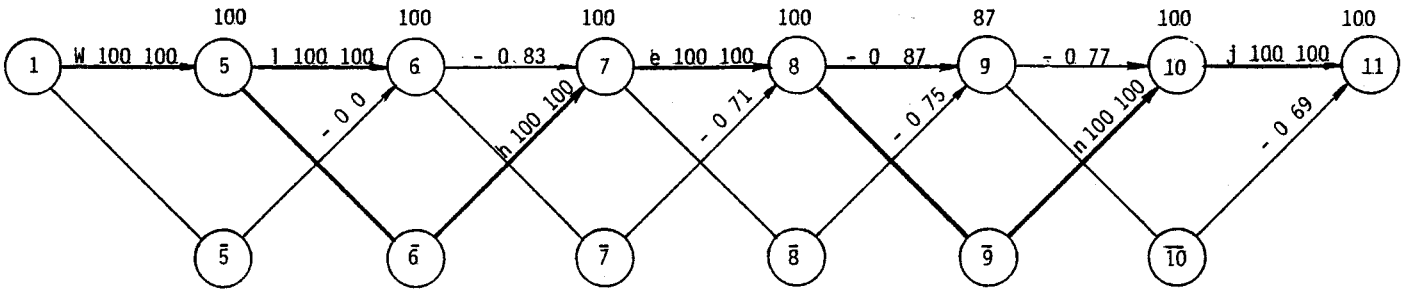
Liste des primitives principales

1	BARRE	SUR 1.	LIEN A DROITE
2	BARRE	SUR 1.	LIEN A GAUCHE
3	BARRE	SUR 2.	LIEN A GAUCHE
4	BARRE	SUR 3.	LIEN A GAUCHE
5	HEURTOIR	SUR 1.	LIEN A GAUCHE
6	BARRE	SUR 1.	LIEN A DROITE

Liste des primitives secondaires

Ligne n° 1

Tableau 1



a) Graphe de recombinaison de la ligne n° 1

```
COUPURES :

   1   1
   5   6
   6   7
   7   8
   8   9
   9  10
  10  11
   0   0

NOEUD :  5  NR PRIM :  4  ADR CH :  1  LONG CH :  1  PROB CH : 100W

NOEUD :  6  NR PRIM :  5  ADR CH :  4  LONG CH :  2  PROB CH : 100Wl

NOEUD :  7  NR PRIM :  6  ADR CH :  8  LONG CH :  2  PROB CH : 100Wh

NOEUD :  8  NR PRIM :  7  ADR CH : 13  LONG CH :  3  PROB CH : 100Whe

NOEUD :  9  NR PRIM :  8  ADR CH : 19  LONG CH :  3  PROB CH :  87Whe

NOEUD : 10  NR PRIM :  9  ADR CH : 26  LONG CH :  4  PROB CH : 100When

NOEUD : 11  NR PRIM : 10  ADR CH : 30  LONG CH :  5  PROB CH : 100WhenJ
```

b) Détail de la recombinaison

Figure 8

La grammaire nous ayant servi à la reconnaissance de chaque segment est la suivante :

W = 13 + 23 + 13 + 23

h = 53 + b1d + 50

e = 30 + h1g

n = 50 + b1d + 50 ! 50 + p1 + 50

j = 55

c = 30

où :

- les nombres de deux chiffres représentent une primitive principale et son attribut.
- les éléments commençant par une lettre sont relatifs aux primitives secondaires et doivent être interprétés comme suit :
  * le premier caractère représente le type (b-barre, h-heurtoir, p-point).
  * le chiffre qui suit indique la sonde sur laquelle il se trouve.
  * les caractères suivants indiquent la liaison exigée ou interdite : d(g) pour liaison obligatoire à la primitive de droite (gauche), d(g) pour liaison prohibée à droite (gauche).

L'on s'aperçoit en analysant le graphe de la figure 8a que parmi tous les segments il y a une seule frise ; nous avons par ailleurs eu la possibilité de constater que les frises générées avec notre procédé d'analyse par trigramme étaient en nombre très réduit, ce qui confirme en pratique notre choix pour les trigrammes. Voici à titre d'exemple comment l'analyse de cette frise s'est déroulée :

Le segment 1-5 comporte 4 primitives principales ; une recherche dans la table des quadrigrammes légaux indique une classe de confusions, formée dans le cas présent d'un seul élément : **W** . Dans cette classe, comme d'ailleurs dans toutes les classes de frises, l'analyse doit être basée sur les primitives secondaires et/ou les liaisons qui existent entre les primitives principales. Dans le cas présent, les primitives étant concaténées au niveau des sondes 6 et 3, la décision a été **W** , elle aurait été **VV** si la liaison sur la sonde 6 entre les primitives 23 et 13 avait manqué.

Dans le cas des frises à nombre de primitives supérieures à 4 l'analyse se déroulera de la même manière que plus haut à une seule différence près : un sous-segment de quatre primitives sera d'abord analysé comme plus haut pour déterminer le caractère qu'il représente (les caractères sont composés d'au plus 4 primitives) ; le sous-segment suivant de 4 primitives, commençant à la fin du caractère déjà déterminé, sera ensuite analysé, le processus étant repris jusqu'à la fin de la frise.

L'on remarque en analysant la figure 8a un aspect intéressant de notre méthode : bien que le noeud 6 ait une probabilité associée de 1, il n'appartient pas au chemin final parce que générant un segment illégal. Ceci peut être assimilé à un retour en arrière, caractéristique des automates à pile employés à la reconnaissance de phrases générées par des grammaires à contexte libre, plus puissantes que les grammaires régulières.

Notre méthode n'a cependant pas la puissance d'un tel automate, le retour en arrière étant ici limité à 2 niveaux : en supposant que le noeud X_j de la figure 6 a une probabilité associée élevée mais que les segments $X_j\ X_{j+1}$ n'est pas une phrase de la grammaire des caractères, le chemin le plus long évitant ce noeud est $X_i\ \bar{X}_{i+1}\ \bar{X}_j\ X_{j+1}$, représentant un retour en arrière de 2 noeuds par rapport à X_j.

VI - EVALUATION

A - INTRODUCTION

B - EVALUATION

1 - Schéma général

2 - Exemples

C - CONCLUSIONS

VI

EVALUATION

A - INTRODUCTION

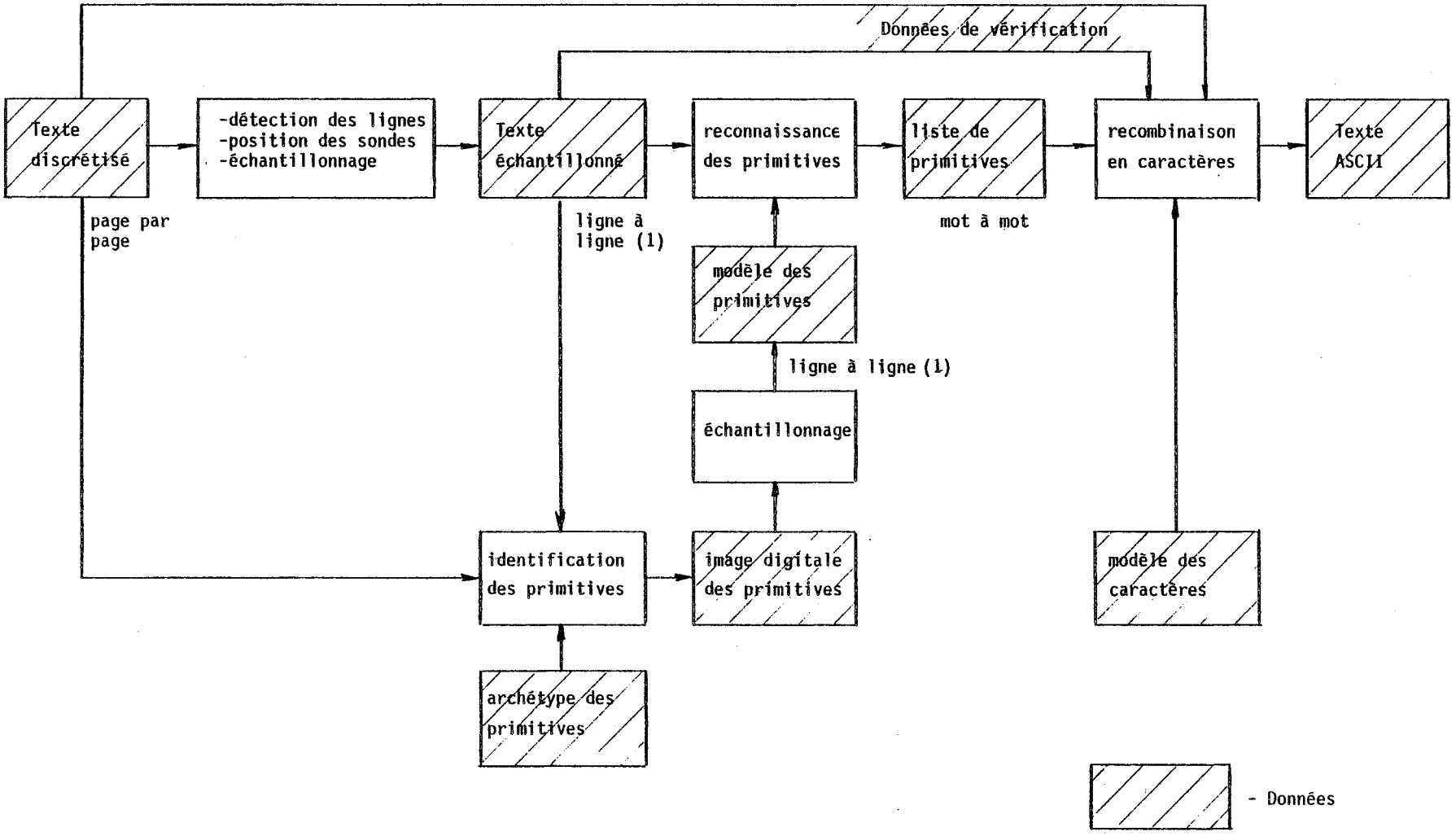
Les lignes qui suivent seront consacrées à la critique en son ensemble de la méthode que nous proposons. Nous n'avons pas pu, dans le cadre de notre travail, affiner notre méthode, très nouvelle et très générale dans sa conception, jusqu'à l'évaluer statistiquement. Toutefois nous présentons ci-dessous un échantillon de textes aussi différents les uns des autres que possible afin de révéler au mieux les avantages de notre système, ainsi que ses limitations dans l'état actuel d'avancement. Basée sur cette analyse qualitative, une discussion portant sur le travail nécessaire aux développements futurs conclura ce chapitre.

B - EVALUATION

1 - Schéma général

Le moment est venu de présenter au lecteur un synoptique regroupant toute l'information dont nous avons fait état lors des chapitres précédents. Ainsi, l'on trouvera sur la figure 1 :

- la détection des lignes dans la page de texte et la position des sondes sur une ligne de texte, décrites au chapitre IV ;



(1) peut être mot à mot pour plus de flexibilité aux désorientations angulaires.

Synoptique général

Figure 1

- la reconnaissance des primitives, exposée au chapitre III ;
- la recombinaison des primitives en caractères, faisant l'objet du chapitre V ;
- l'identification automatique de certaines primitives, traitée elle aussi lors du chapitre IV ;

de même que la position relative de ces blocs et la circulation des données.

Cette structure peut être matérialisée en une machine ayant une architecture du type pipe-line. Pour l'instant, une simulation sur un mini-ordinateur a été réalisée afin de valider les algorithmes présentés. La configuration matérielle dont nous disposions était la suivante :

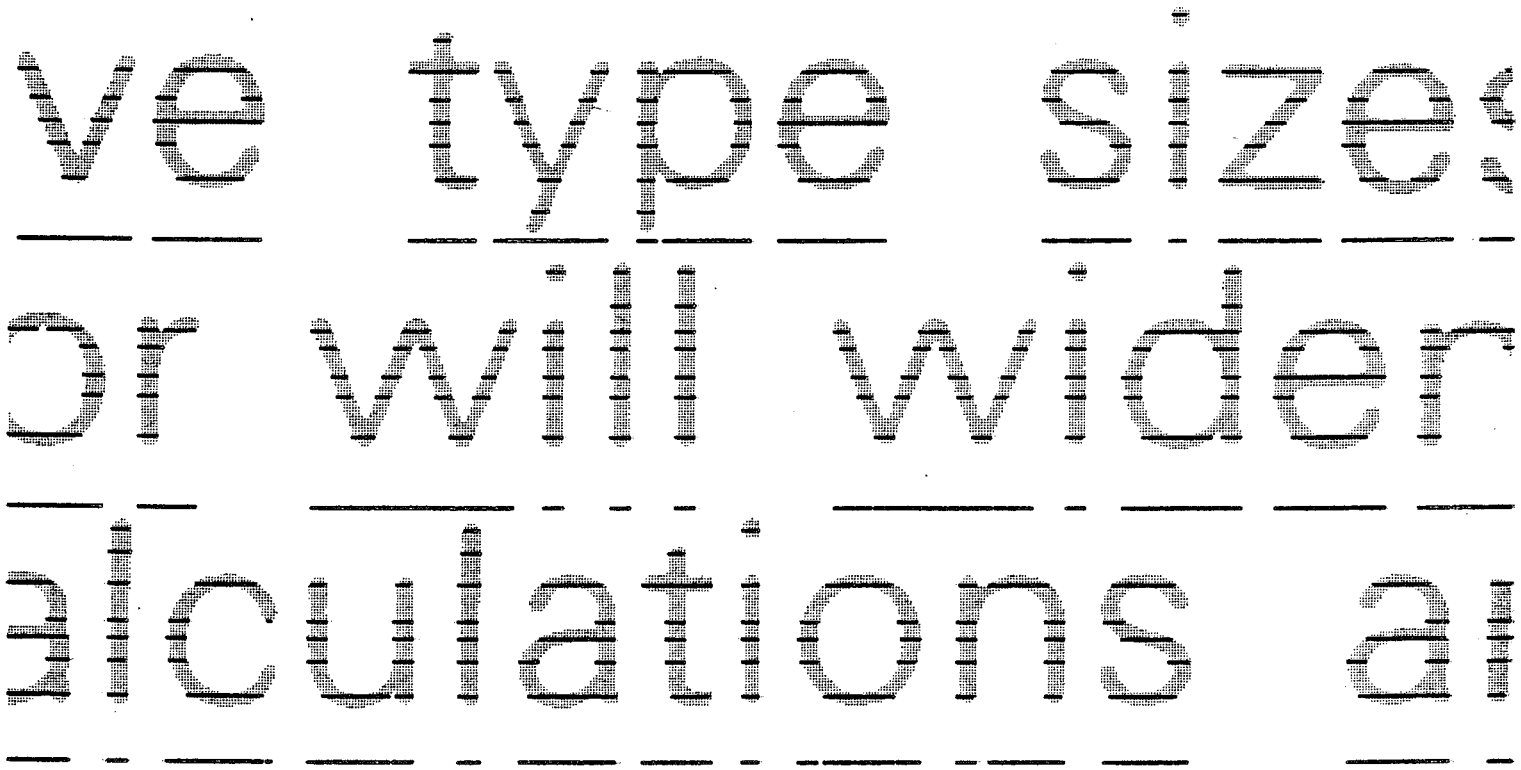
- mini-ordinateur PDP-11/40, ayant une mémoire vive de 64 Ko et une mémoire de masse de 5 Mo ;
- système d'acquisition Hamamatsu, comprenant une caméra de prise de vue et un échantillonneur, capable de discrétiser une image en 512 x 512 points sur 256 niveaux de gris ; pour l'application présentée deux niveaux sont utilisés ;
- mémoire d'images Matrox de 512 x 512 points de 8 bits ; dans notre application trois bits sont employés effectivement, un pour stocker la page de texte, les deux autres pour afficher des informations de contrôle ;
- moniteur de télévision en couleurs ;
- imprimante électrostatique Gould.

2 - Exemples

Le lecteur n'aura aucune difficulté à interpréter les résultats qui suivent ; nous avons en effet gardé la présentation adoptée lors des illustrations des chapitres précédents. De plus, de toute l'information disponible sur une ligne de texte nous avons retenu celle qui nous semblait utile pour illustrer l'aspect discuté.

a) La Planche 1 montre le comportement du système dans le cas d'une **police normale, à graisse uniforme, sans empattements** . L'apprentissage et l'extraction des primitives se sont déroulés sans aucun problème particulier ; le résultat de la recombinaison de ces primitives se trouve au-dessous du texte original. Dans la deuxième et troisième ligne des caractères ont été tronquées lors de la prise de vue. Ce fait explique la probabilité de la chaîne finale :

- pour la deuxième ligne, la primitive 4 se trouvant au début n'a pu être englobée en aucun caractère, fournissant un segment de probabilité nulle ;
- pour la troisième ligne la primitive 5 qui débute le texte n'était pas accompagnée de primitives secondaires nécessaires pour former un caractère complet (a=50+h3dḡ ; r=50+p1!50+b1 selon le formalisme expliqué page 5.24) ; il en va de même pour la dernière primitive.



LONG CH : 10 PROB CH : 100vetypesize

LONG CH : 10 PROB CH : 92rwillwider

LONG CH : 11 PROB CH : 841culationsa

(Univers light 685)

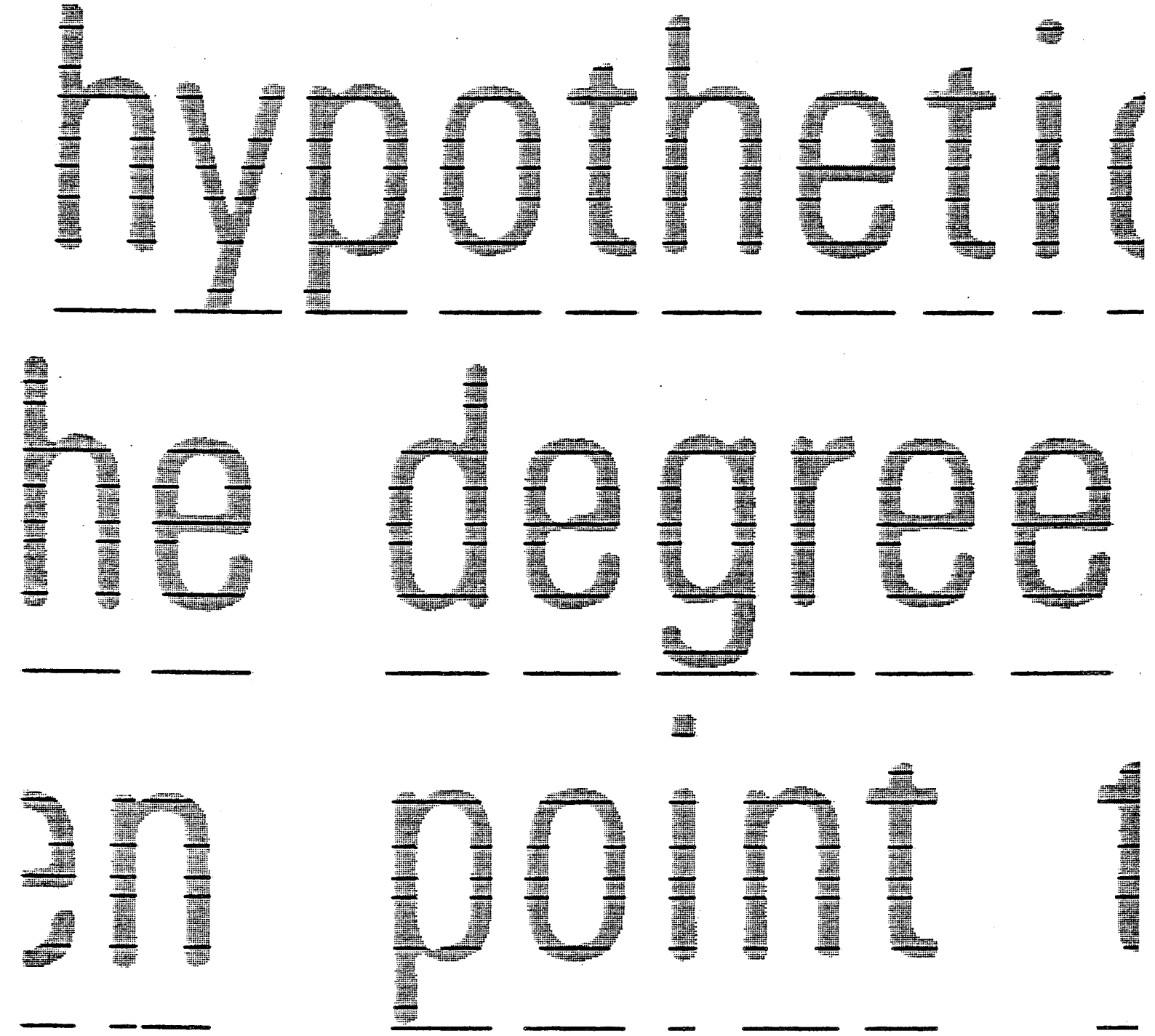
Planche 1

b) Le texte de la Planche 2 appartient à une police à **graisse uniforme, sans empattements mais relativement condensée** (faible rapport chasse/force de corps). La difficulté qu'une telle police peut poser, dans laquelle les **o** sont moins arrondis que rectangulaires, est la confusion entre les primitives 3 ou 4 d'une part et 5 d'autre part : les sondes 2, 4 et 5 ont en effet sensiblement la même position pour ces 3 primitives. C'est un des cas où la notion de côté libre, définie au chapitre 3 § D.4 se révèle très intéressante.

Une erreur apparaît à la deuxième ligne : **g** est devenu q : la séparation dans la classe de confusions (g, q) doit se faire par la longueur du segment de jambage.

c) Le texte de la Planche 3 présente plusieurs difficultés. Il s'agit cette fois d'une police **d'épaisseur de trait uniforme mais relativement importante, avec des empattements, légèrement condensée .** D'autre part, la taille du corps de texte est relativement petite. Aucune erreur n'est apparue à l'extraction des primitives malgré la présence d'empattements qui dénaturent le modèle des primitives 1 et 2 obtenues par apprentissage automatique (voir figure 2). A l'heure ou nous écrivons ces lignes l'étage de recombinaison n'est pas entièrement au point. Ceci explique les erreurs des deux premières lignes :

- g s'est transformé en q (déjà vu) ;
- la description de k ne figurait pas dans le dictionnaire.



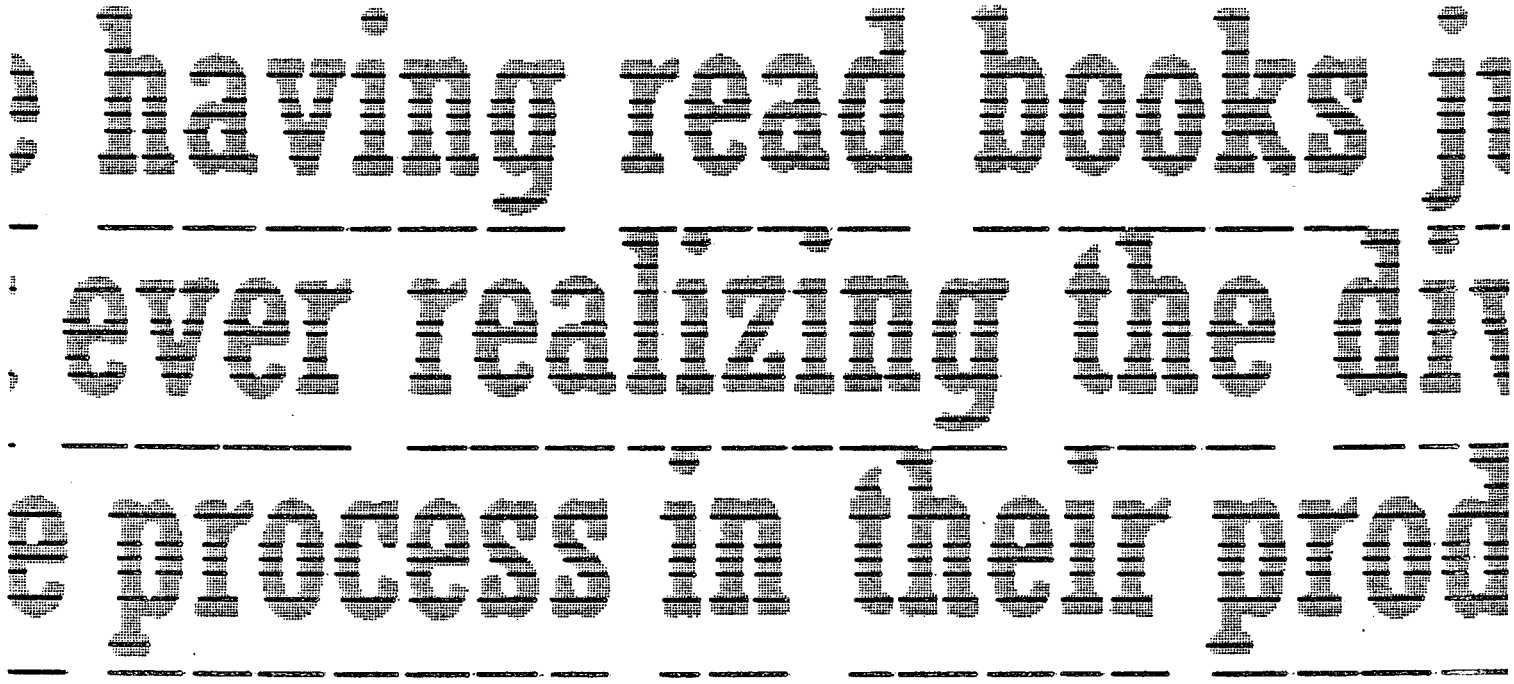
LONG CH : 10 PROB CH : 100hypothetic

LONG CH : 8 PROB CH : 100hedegree

LONG CH : 7 PROB CH : 100npointt

(Univers light condensed 686)

Planche 2

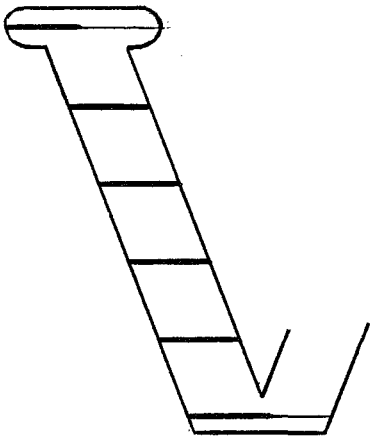


LONG CH : 15 PROB CH : 87havingreadboosJ

LONG CH : 18 PROB CH : 95everrealizingthedi

LONG CH : 19 PROB CH : 100rrprocessintheirprod

Planche 3



Lors de l'apprentissage automatique de la primitive 1, cette image est stockée en mémoire de masse et échantillonnée à chaque ligne de texte pour rattrapper la dérive des sondes ; la graisse du modèle étant constante, le segment de la sonde 1 apparaît en dehors de l'alignement des autres sondes.

Figure 2

La première primitive de la dernière ligne est perçue, du fait de la troncature, comme une primitive 5, ce qui explique l'erreur sur le premier caractère reconnu.

d) L'intérêt de la Planche 4 réside dans la présence de **pleins et déliés** . Une méthode pour la détection des primitives 2, seules affectées de déliés, a été exposée au chapitre 4. Elle exigeait la symétrie entre les primitives 1 et 2 et était capable de préciser l'épaisseur de la primitive 2 par mesure directe sur les primitives 2 trouvées à la suite d'une première passe avec un modèle approximatif. Dans cet exemple les pentes légèrement différentes des primitives 1 et 2 sont à l'origine du score relativement modeste réalisé par ces dernières ; de plus la corrélation devient très sensible aux imperfections inhérentes à l'échantillonnage (déplacement d'une unité à gauche ou à droite d'un segment) pour des graisses de plus en plus faibles. L'erreur rencontrée dans cette ligne ne vient pas de la présence de déliés mais de la forme assez fantaisiste de la police : la barre du e se trouve suffisament haut pour fausser l'échantillonnage de la partie gauche du a , créant ainsi une primitive ambiguë (prob(3/f)=0.75, prob(5/f)=0.72).

Deux solutions sont possibles pour palier à ce genre d'inconvénients :

- compléter la description du a par a=30+b1gd+50 ;
- tenir compte dans l'établissement des coupures et de la probabilité de chaque segment, de la probabilité des primitives mises en jeu.

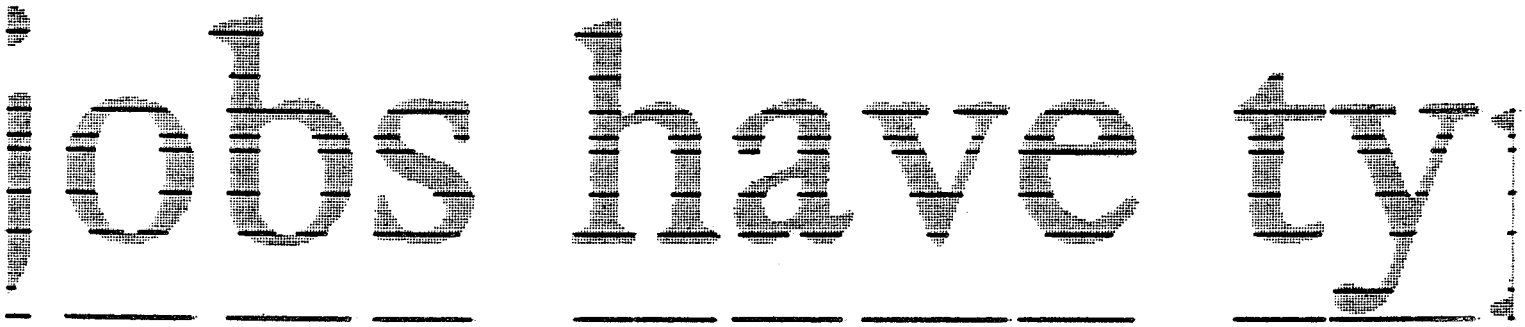


Planche 4

PRIM	ATTR	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
G(G)	( )	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	-1	0
5(5)	5(5)	87(87)	1(1)	1(1)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	2(-4)	0(-1)	-1	0
3(3)	0(0)	100(97)	3(3)	3(3)	3(3)	3(3)	3(3)	0(0)	0(0)	0(0)	2(-5)	20(21)	1	1
4(4)	0(0)	97(95)	3(3)	5(5)	5(5)	5(5)	5(5)	0(0)	0(0)	0(0)	3(-4)	53(54)	-1	0
5(5)	3(3)	95(92)	5(5)	7(7)	7(7)	7(7)	7(7)	3(3)	0(0)	1(1)	-4(2)	77(76)	2	1
4(4)	0(0)	95(95)	5(5)	9(9)	9(9)	9(9)	9(9)	0(0)	0(0)	0(0)	5(-3)	107(108)	3	1
6(6)	0(0)	90(87)	7(7)	11(11)	11(11)	11(11)	11(11)	0(0)	0(0)	0(0)	-3(2)	128(126)	4	2
5(5)	3(3)	100(100)	9(9)	13(13)	13(13)	15(15)	13(13)	5(5)	0(0)	3(3)	2(-4)	198(199)	6	1
5(5)	0(0)	90(87)	11(11)	15(15)	15(15)	17(17)	15(15)	0(0)	0(0)	0(0)	5(-5)	226(227)	-1	0
3(5)	0(0)	75(72)	13(13)	17(17)	17(17)	19(19)	17(17)	0(0)	0(0)	0(0)	-2(2)	247(249)	7	1
5(5)	0(0)	95(95)	13(13)	19(19)	19(19)	21(21)	19(19)	0(0)	0(0)	0(0)	2(-5)	269(270)	-1	0
1(1)	0(0)	95(95)	15(15)	21(21)	21(21)	23(23)	21(21)	0(0)	0(0)	0(0)	2(-4)	301(302)	8	2
2(2)	0(0)	67(45)	17(17)	23(23)	21(21)	25(25)	21(21)	0(0)	0(0)	0(0)	-2(4)	321(326)	-1	0
3(3)	0(0)	97(87)	19(19)	25(25)	23(23)	27(27)	23(23)	0(0)	0(0)	0(0)	1(-5)	344(346)	10	5
5(5)	2(2)	97(95)	21(21)	27(27)	25(25)	31(31)	25(25)	0(0)	0(0)	5(5)	-4(2)	425(422)	15	2
1(1)	0(0)	97(87)	23(23)	29(29)	27(27)	33(33)	27(27)	0(0)	0(0)	0(0)	2(-5)	459(461)	17	2
2(2)	4(4)	65(60)	25(25)	31(31)	27(27)	35(35)	29(27)	0(0)	3(3)	0(0)	-4(-4)	480(480)	19	4

LONG CH : 10 PROB CH : 92Jobshcvety

Planche 4 (suite)

La première solution, facile à mettre en oeuvre, se justifie par la fréquence relativement élevée des polices ayant ces caractéristiques (a au lieu de a). La deuxième, plus générale, permettrait du même coup le traitement plus aisé des classes de confusion, en réduisant les appels aux données de vérification (figure 1).

e) L'intérêt de la Planche 5, semblable à la Planche 1, réside en la **taille des caractères** . Nous avons en effet voulu savoir jusqu'à quelle limite on pouvait diminuer la hauteur du corps de texte sans affecter les performances de l'extracteur de primitives. Dans cet exemple, le corps de texte a une hauteur de 25 lignes digitales (soit une résolution de 12 points/mm pour analyser des caractères de 2 mm de haut sur le corps de texte) sans qu'aucune erreur apparaisse à l'extraction. Des erreurs sont par contre survenues pour des tailles plus petites, essentiellement substitution de 5 en 1 ; un choix plus judicieux des cotés libres de la primitive 1 permettrait d'éliminer ce problème pour des tailles encore plus petites.

Les erreurs rencontrées dans le texte reconstitué à partir de la Planche 5 sont dues elles aussi à l'imperfection de la description des caractères :

- dans la deuxième et troisième ligne on rencontre la frise |...| (n **un** et respectivement **ua**) qui n'a pas encore été programmée.
- le caractère f de la troisième ligne s'est transformé en l parce que la classe de confusions (l, t, f) n'est pas complètement décrite.

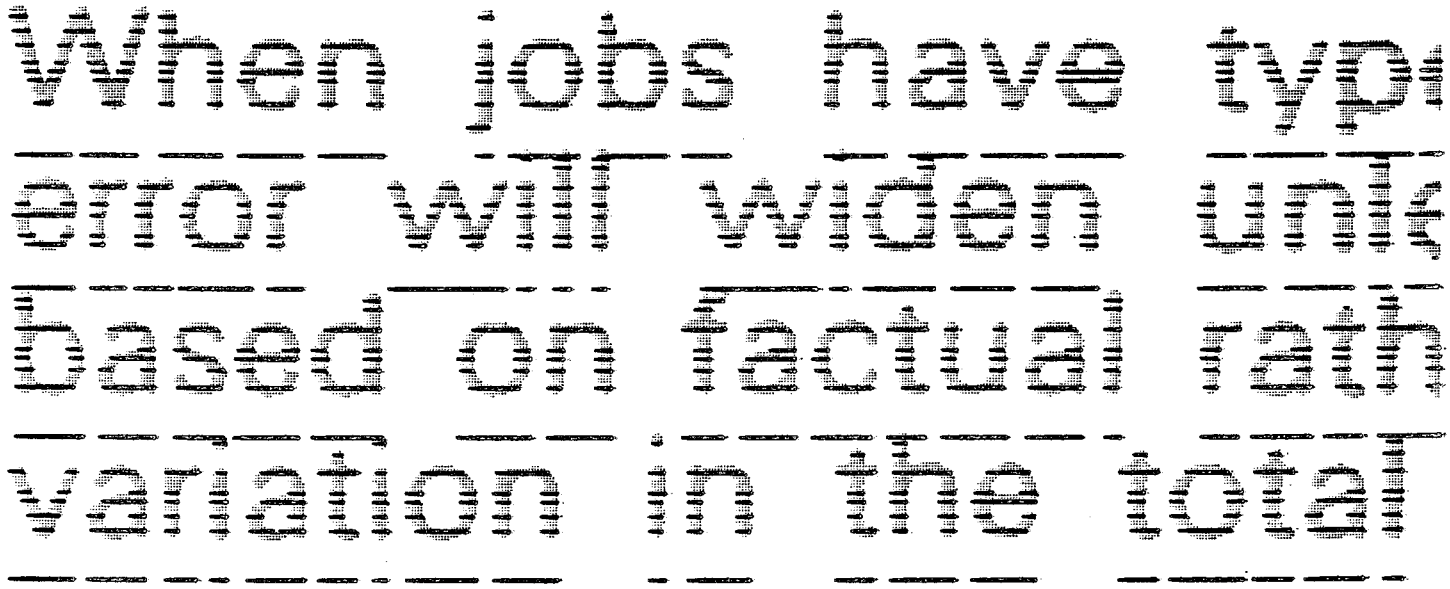

When jobs have typ
error will widen unk
based on factual rath
variation in the total

Planche 5

PRIM G(G)	ATTR ( )	PROB 0(0)	PTS1 0(0)	PTS2 0(0)	PTS3 0(0)	PTS4 0(0)	PTS5 0(0)	PTS6 0(0)	PTS7 0(0)	PTS8 0(0)	SONC 0(0)	POS2 0(0)	OUTSP -1	OUTSL 0
1(1)	3(3)	100(100)	1(1)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	1(1)	2(-4)	41(41)	-1	0
2(2)	3(3)	90(75)	3(3)	3(3)	1(1)	3(3)	3(3)	3(3)	0(0)	3(3)	-4(5)	49(50)	-1	0
1(1)	3(3)	100(100)	5(5)	5(5)	3(3)	5(5)	5(5)	3(3)	0(0)	3(3)	2(-4)	63(64)	-1	0
2(2)	3(3)	95(95)	7(7)	7(7)	3(3)	7(7)	7(7)	5(5)	0(0)	5(5)	4(-5)	72(72)	-1	0
5(5)	3(3)	100(100)	9(9)	9(9)	5(5)	9(9)	9(9)	7(7)	0(0)	7(7)	2(-3)	86(86)	1	1
5(5)	0(0)	85(85)	9(9)	11(11)	7(7)	11(11)	11(11)	0(0)	0(0)	0(0)	3(-5)	102(102)	-1	0
3(3)	0(0)	95(95)	11(11)	13(13)	9(9)	13(13)	13(13)	0(0)	0(0)	0(0)	1(-5)	112(113)	2	4
5(5)	0(0)	95(95)	13(13)	15(15)	11(11)	17(17)	15(15)	0(0)	0(0)	0(0)	2(-5)	139(140)	6	1
5(5)	0(0)	85(85)	13(13)	17(17)	13(13)	19(19)	17(17)	0(0)	0(0)	0(0)	3(-5)	155(155)	-1	0
5(5)	5(5)	95(95)	15(15)	19(19)	15(15)	21(21)	19(19)	9(9)	1(1)	0(0)	2(-5)	192(192)	-1	0
3(3)	0(0)	100(95)	17(17)	21(21)	17(17)	23(23)	21(21)	0(0)	0(0)	0(0)	2(-5)	202(203)	7	2
4(4)	0(0)	95(95)	17(17)	23(23)	17(17)	25(25)	23(23)	0(0)	0(0)	0(0)	2(-3)	221(221)	-1	0
5(5)	3(3)	100(100)	19(19)	25(25)	19(19)	27(27)	25(25)	11(11)	0(0)	9(9)	2(-4)	232(232)	9	2
4(4)	0(0)	100(95)	19(19)	27(27)	19(19)	29(29)	27(27)	0(0)	0(0)	0(0)	-4(2)	250(249)	11	2
6(6)	0(0)	100(100)	21(21)	29(29)	21(21)	31(31)	29(29)	0(0)	0(0)	0(0)	4(-4)	267(267)	13	1
5(5)	3(3)	100(100)	23(23)	31(31)	23(23)	33(33)	31(31)	13(13)	0(0)	11(11)	2(-3)	307(307)	14	1
5(5)	0(0)	85(85)	23(23)	33(33)	25(25)	35(35)	33(33)	0(0)	0(0)	0(0)	3(-5)	323(323)	15	4
5(5)	0(0)	90(85)	25(25)	35(35)	27(27)	37(37)	37(37)	0(0)	0(0)	0(0)	5(-5)	348(349)	-1	0
1(1)	0(0)	95(95)	27(27)	37(37)	29(29)	39(39)	39(39)	0(0)	0(0)	0(0)	2(-5)	362(362)	-1	0
2(2)	0(0)	90(90)	29(29)	39(39)	29(29)	41(41)	41(41)	0(0)	0(0)	0(0)	2(-5)	373(373)	-1	0
3(3)	0(0)	100(100)	31(31)	41(41)	31(31)	43(43)	43(43)	0(0)	0(0)	0(0)	4(-3)	385(385)	19	6
5(1)	2(0)	100(95)	33(33)	43(43)	35(35)	47(47)	45(45)	0(0)	0(0)	13(0)	-4(1)	438(437)	25	1
1(1)	0(0)	95(95)	35(35)	45(45)	37(37)	49(49)	47(47)	0(0)	0(0)	0(0)	2(-5)	453(453)	-1	0
2(2)	4(4)	90(90)	37(37)	47(47)	37(37)	51(51)	49(49)	0(0)	3(3)	0(0)	2(-5)	465(465)	-1	0
5(5)	4(4)	95(95)	39(39)	49(49)	39(39)	53(53)	51(51)	0(0)	5(5)	0(0)	2(-4)	477(479)	26	2
4(4)	0(0)	90(90)	39(39)	51(51)	41(41)	55(55)	53(53)	0(0)	0(0)	0(0)	3(-4)	495(496)	28	4

LONG CH : 15 PROB CH : 100WhenJobshavetyp

a) Ligne n° 1

PRIM G(G)	ATTR ( )	PROB 0(0)	PTS1 0(0)	PTS2 0(0)	PTS3 0(0)	PTS4 0(0)	PTS5 0(0)	PTS6 0(0)	PTS7 0(0)	PTS8 0(0)	SONC 0(0)	POS2 0(0)	OUTSP -1	OUTSL 0
3(3)	0(0)	100(100)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	0(0)	0(0)	3(-4)	37(37)	1	5
5(5)	0(0)	100(100)	3(3)	3(3)	3(3)	5(5)	5(5)	0(0)	0(0)	0(0)	2(-3)	63(63)	6	1
5(5)	0(0)	100(95)	5(5)	5(5)	5(5)	7(7)	7(7)	0(0)	0(0)	0(0)	-4(2)	79(78)	7	1
3(3)	0(0)	100(100)	7(7)	7(7)	7(7)	9(9)	9(9)	0(0)	0(0)	0(0)	2(-4)	93(93)	8	2
4(4)	0(0)	90(85)	7(7)	9(9)	7(7)	11(11)	11(11)	0(0)	0(0)	0(0)	-2(5)	112(113)	-1	0
5(5)	0(0)	95(95)	9(9)	11(11)	9(9)	13(13)	13(13)	0(0)	0(0)	0(0)	2(-3)	122(122)	10	1
1(1)	0(0)	95(95)	11(11)	13(13)	11(11)	15(15)	15(15)	0(0)	0(0)	0(0)	2(-4)	166(166)	-1	4
2(2)	0(0)	75(70)	13(13)	15(15)	11(11)	17(17)	15(15)	0(0)	0(0)	0(0)	-3(2)	173(175)	-1	0
1(1)	0(0)	90(85)	13(13)	17(17)	13(13)	19(19)	17(17)	0(0)	0(0)	0(0)	-2(4)	184(185)	-1	0
2(2)	0(0)	75(65)	15(15)	19(19)	13(13)	21(21)	17(17)	0(0)	0(0)	0(0)	-3(2)	191(193)	-1	0
5(5)	1(1)	100(100)	17(17)	21(21)	15(15)	23(23)	19(19)	1(1)	0(0)	0(0)	2(-4)	205(205)	-1	2
5(5)	3(3)	100(100)	19(19)	23(23)	17(17)	25(25)	21(21)	3(3)	0(0)	1(1)	2(-4)	217(218)	-1	0
5(5)	3(3)	100(100)	21(21)	25(25)	19(19)	27(27)	23(23)	5(5)	0(0)	3(3)	2(-4)	230(230)	-1	2
1(1)	0(0)	90(90)	23(23)	27(27)	21(21)	29(29)	25(25)	0(0)	0(0)	0(0)	1(-4)	270(270)	-1	2
2(2)	0(0)	75(70)	25(25)	29(29)	21(21)	31(31)	25(25)	0(0)	0(0)	0(0)	-3(2)	277(279)	-1	0
1(1)	0(0)	85(85)	25(25)	31(31)	23(23)	33(33)	27(27)	0(0)	0(0)	0(0)	4(-2)	289(288)	-1	1
2(2)	0(0)	85(80)	27(27)	33(33)	23(23)	35(35)	27(27)	0(0)	0(0)	0(0)	-4(2)	296(297)	-1	0
5(5)	1(1)	95(95)	29(29)	35(35)	25(25)	37(37)	29(29)	7(7)	0(0)	0(0)	2(-5)	309(310)	-1	0
3(3)	0(0)	100(100)	31(31)	37(37)	27(27)	39(39)	31(31)	0(0)	0(0)	0(0)	3(-4)	320(320)	11	2
5(5)	3(3)	100(100)	31(31)	39(39)	27(27)	41(41)	33(33)	9(9)	0(0)	5(5)	4(-3)	338(338)	-1	0
3(3)	0(0)	100(100)	33(33)	41(41)	29(29)	43(43)	35(35)	0(0)	0(0)	0(0)	4(-4)	349(349)	13	5
5(5)	0(0)	95(95)	35(35)	43(43)	31(31)	47(47)	39(39)	0(0)	0(0)	0(0)	2(-3)	376(376)	18	1
5(5)	0(0)	85(80)	35(35)	45(45)	33(33)	49(49)	41(41)	0(0)	0(0)	0(0)	3(-5)	392(393)	-1	0
5(5)	0(0)	90(90)	37(37)	47(47)	35(35)	51(51)	43(43)	0(0)	0(0)	0(0)	1(-5)	432(432)	19	1
5(5)	0(0)	95(95)	39(39)	49(49)	35(35)	53(53)	45(45)	0(0)	0(0)	0(0)	2(-5)	447(448)	-1	0
5(5)	0(0)	100(95)	41(41)	51(51)	37(37)	55(55)	47(47)	0(0)	0(0)	0(0)	-3(2)	461(460)	20	1
5(5)	0(0)	85(80)	41(41)	53(53)	39(39)	57(57)	49(49)	0(0)	0(0)	0(0)	3(-4)	476(477)	-1	0
5(5)	3(3)	100(100)	43(43)	55(55)	41(41)	59(59)	51(51)	11(11)	0(0)	7(7)	2(-4)	488(490)	-1	0
3(5)	0(0)	80(75)	45(45)	57(57)	43(43)	61(61)	53(53)	0(0)	0(0)	0(0)	-3(3)	500(502)	21	1

LONG CH : 15 PROB CH : 77errorwillwidele

b) Ligne n° 2

Planche 5 (suite)

PRIM	ATTR	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
G(G)	( )	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	-1	0
5(3)	3(0)	92(80)	1(1)	1(1)	1(1)	1(1)	1(1)	1(0)	0(0)	1(0)	2(4)	38(37)	1	2
4(4)	0(0)	84(84)	1(1)	3(3)	1(1)	3(3)	3(3)	0(0)	0(0)	0(0)	2(-2)	56(55)	3	4
5(5)	0(0)	84(84)	3(3)	5(5)	3(3)	5(5)	7(7)	0(0)	0(0)	0(0)	4(-5)	79(79)	7	1
6(6)	0(0)	96(96)	5(5)	7(7)	5(5)	7(7)	9(9)	0(0)	0(0)	0(0)	4(-5)	98(98)	8	1
3(3)	0(0)	96(96)	7(7)	9(9)	7(7)	9(9)	11(11)	0(0)	0(0)	0(0)	2(-4)	111(111)	9	4
3(3)	0(0)	88(76)	9(9)	11(11)	9(9)	13(13)	13(13)	0(0)	0(0)	0(0)	1(-4)	136(135)	13	2
5(5)	3(3)	100(100)	9(9)	13(13)	9(9)	15(15)	15(15)	3(3)	0(0)	3(3)	4(-4)	154(154)	-1	4
3(3)	0(0)	92(80)	11(11)	15(15)	11(11)	17(17)	17(17)	0(0)	0(0)	0(0)	2(-4)	185(184)	15	2
4(4)	0(0)	92(80)	11(11)	17(17)	11(11)	19(19)	19(19)	0(0)	0(0)	0(0)	-4(5)	204(205)	-1	0
5(5)	0(0)	92(92)	13(13)	19(19)	13(13)	21(21)	21(21)	0(0)	0(0)	0(0)	2(-5)	215(215)	17	1
5(5)	0(0)	84(84)	13(13)	21(21)	15(15)	23(23)	23(23)	0(0)	0(0)	0(0)	3(-5)	231(231)	-1	0
5(5)	3(3)	100(100)	15(15)	23(23)	17(17)	25(25)	25(25)	5(5)	0(0)	5(5)	3(-4)	264(264)	18	4
5(5)	0(0)	84(84)	17(17)	25(25)	19(19)	27(27)	29(29)	0(0)	0(0)	0(0)	4(-5)	289(292)	-1	0
3(3)	0(0)	92(80)	19(19)	27(27)	21(21)	29(29)	31(31)	0(0)	0(0)	0(0)	2(-4)	303(302)	22	2
5(5)	2(2)	92(92)	21(21)	29(29)	23(23)	31(31)	33(33)	0(0)	0(0)	7(7)	2(-4)	331(331)	-1	0
5(5)	0(0)	68(64)	23(23)	31(31)	25(25)	33(33)	35(35)	0(0)	0(0)	0(0)	-4(1)	345(347)	24	1
5(5)	0(0)	80(80)	25(25)	33(33)	25(25)	35(35)	37(37)	0(0)	0(0)	0(0)	2(-4)	362(361)	25	4
5(5)	0(0)	84(84)	27(27)	35(35)	27(27)	37(37)	41(41)	0(0)	0(0)	0(0)	4(-5)	387(387)	-1	0
5(5)	3(3)	92(80)	29(29)	37(37)	29(29)	39(39)	43(43)	7(7)	0(0)	9(9)	2(-3)	400(399)	-1	0
5(5)	0(0)	88(88)	31(31)	39(39)	31(31)	41(41)	45(45)	0(0)	0(0)	0(0)	3(-3)	433(432)	29	5
5(5)	0(0)	84(84)	33(33)	41(41)	33(33)	43(43)	49(49)	0(0)	0(0)	0(0)	4(-2)	462(462)	-1	0
5(5)	2(2)	96(96)	35(35)	43(43)	35(35)	45(45)	51(51)	0(0)	0(0)	11(11)	2(-4)	476(476)	-1	0
5(5)	3(3)	100(100)	37(37)	45(45)	37(37)	47(47)	53(53)	9(9)	0(0)	13(13)	2(-3)	491(491)	34	1
5(5)	0(0)	72(68)	37(37)	47(47)	39(39)	49(49)	55(55)	0(0)	0(0)	0(0)	-5(3)	506(507)	-1	1

LONG CH : 17 PROB CH : 100basedonlactmlrath

c) *Ligne n° 3*

PRIM	ATTR	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
G(G)	( )	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	-1	0
1(1)	0(0)	76(68)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	0(0)	0(0)	4(-2)	41(39)	-1	2
2(2)	0(0)	72(60)	3(3)	3(3)	1(1)	3(3)	1(1)	0(0)	0(0)	0(0)	-4(2)	48(50)	1	4
5(5)	0(0)	84(84)	5(5)	5(5)	3(3)	5(5)	5(5)	0(0)	0(0)	0(0)	5(-5)	76(76)	-1	0
5(5)	0(0)	84(80)	7(7)	7(7)	5(5)	7(7)	7(7)	0(0)	0(0)	0(0)	2(-4)	88(87)	5	1
5(5)	1(1)	88(88)	9(9)	9(9)	7(7)	9(9)	9(9)	1(1)	0(0)	0(0)	2(-5)	103(103)	6	4
5(5)	0(0)	84(84)	11(11)	11(11)	11(11)	11(11)	13(13)	0(0)	0(0)	0(0)	5(-5)	129(129)	-1	0
5(5)	2(2)	88(84)	13(13)	13(13)	13(13)	13(13)	15(15)	0(0)	0(0)	1(1)	3(-4)	143(142)	-1	0
5(5)	1(1)	72(72)	15(15)	15(15)	15(15)	15(15)	17(17)	3(3)	0(0)	0(0)	2(-5)	157(156)	-1	2
3(3)	0(0)	92(88)	17(17)	17(17)	17(17)	17(17)	19(19)	0(0)	0(0)	0(0)	-4(2)	167(168)	10	2
4(4)	0(0)	92(84)	17(17)	19(19)	17(17)	19(19)	21(21)	0(0)	0(0)	0(0)	-4(5)	186(187)	-1	0
5(5)	0(0)	92(84)	19(19)	21(21)	19(19)	21(21)	23(23)	0(0)	0(0)	0(0)	-4(2)	197(198)	12	1
5(5)	0(0)	84(84)	19(19)	23(23)	21(21)	23(23)	25(25)	0(0)	0(0)	0(0)	3(-5)	213(213)	-1	4
5(5)	1(1)	84(80)	21(21)	25(25)	23(23)	25(25)	27(27)	5(5)	0(0)	0(0)	2(-3)	249(248)	-1	0
5(5)	0(0)	84(80)	23(23)	27(27)	25(25)	27(27)	29(29)	0(0)	0(0)	0(0)	2(-4)	262(261)	13	1
5(5)	0(0)	72(68)	23(23)	29(29)	27(27)	29(29)	31(31)	0(0)	0(0)	0(0)	-5(3)	277(278)	-1	0
5(1)	2(0)	88(84)	25(25)	31(31)	29(29)	31(31)	33(33)	0(0)	0(0)	3(0)	2(1)	315(315)	-1	0
5(5)	3(3)	92(84)	27(27)	33(33)	31(31)	33(33)	35(35)	7(7)	0(0)	5(5)	-4(3)	329(330)	14	1
5(5)	0(0)	88(88)	27(27)	35(35)	33(33)	35(35)	37(37)	0(0)	0(0)	0(0)	2(-5)	345(345)	-1	0
3(3)	0(0)	100(100)	29(29)	37(37)	35(35)	37(37)	39(39)	0(0)	0(0)	0(0)	2(-4)	356(356)	15	5
5(5)	2(2)	96(96)	31(31)	39(39)	37(37)	41(41)	43(43)	0(0)	0(0)	7(7)	2(-4)	408(408)	-1	0
3(3)	0(0)	92(88)	33(33)	41(41)	39(39)	43(43)	45(45)	0(0)	0(0)	0(0)	-4(2)	421(422)	20	2
4(4)	0(0)	92(88)	33(33)	43(43)	39(39)	45(45)	47(47)	0(0)	0(0)	0(0)	4(-1)	441(440)	-1	0
5(5)	2(2)	96(96)	35(35)	45(45)	41(41)	47(47)	49(49)	0(0)	0(0)	9(9)	2(-4)	453(453)	22	4
5(5)	0(0)	84(84)	37(37)	47(47)	43(43)	49(49)	53(53)	0(0)	0(0)	0(0)	5(-5)	481(481)	-1	1
5(5)	3(3)	100(100)	39(39)	49(49)	45(45)	51(51)	55(55)	9(9)	0(0)	11(11)	2(-3)	493(493)	-1	0

LONG CH : 19 PROB CH : 100variationinthetotal

Planche 5 (suite)

f) Nous avons aussi voulu estimer la tolérance du système aux **désorientations** . La Planche 6 montre un texte incliné jusqu'à ce qu'une erreur apparaisse dans la liste des primitives principales (inclinaison de 2°30 environ).

Ces erreurs sont :

1 - **omissions** : dans la dernière ligne la déviation du texte par rapport à l'horizontale est telle que la sonde 1 passe au-dessus des 2 derniers caractères.

2 - **substitutions** :

* 4 devient 2 : troisième ligne, seconde primitive du **b** ;
* 5 devient **4** : seconde primitive du **n** , troisième ligne. La substitution est favorisée par le type de liaison à la primitive précédente. La configuration 50 + 40 ou 40 pris isolément n'étant pas admissible, la lettre n n'est pas reconnue.

Nous avons indiqué lors du chapitre IV une méthode pouvant prendre en compte des déviations angulaires relativement importantes : morcellement de la ligne de texte au niveau du mot de sorte que la déviation verticale des sondes ne dépasse un certain seuil pour une erreur angulaire donnée. Avec cette stratégie les erreurs du premier type seront éliminées et les erreurs du deuxième fortement diminuées.

A notre avis, une relation d'inverse proportionnalité existe entre la tolérance angulaire admise et le nombre de caractères sur la ligne de texte. En effet, soit a la tolérance admise pour une ligne de longueur l.

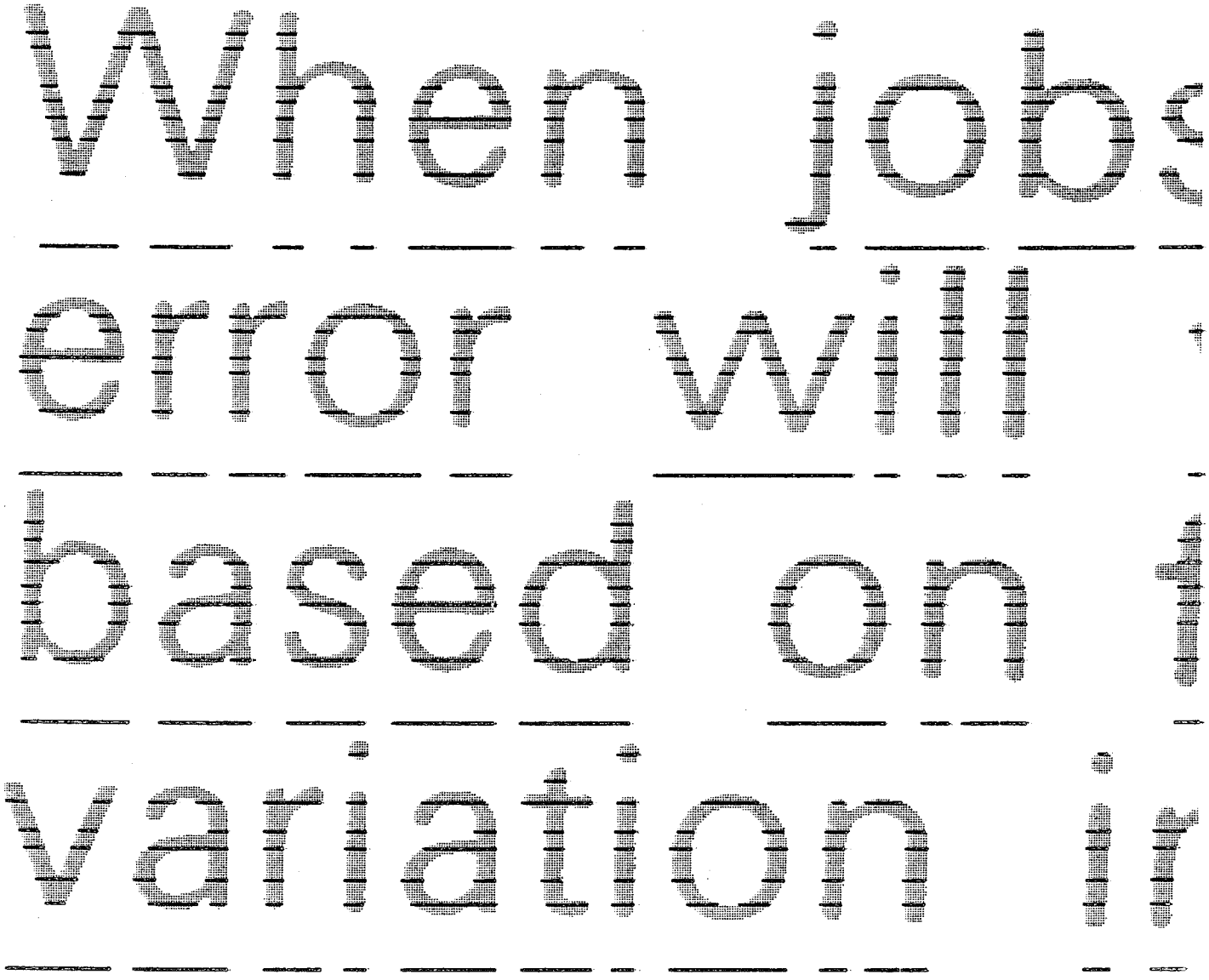


Planche 6

PRIM	ATTR	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
G(G)	( )	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	-1	0
1(1)	3(3)	88(88)	1(1)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	1(1)	4(-4)	52(51)	-1	0
2(2)	3(3)	82(77)	3(3)	3(3)	1(1)	3(3)	3(3)	3(3)	0(0)	3(3)	2(-3)	68(66)	-1	0
1(1)	3(3)	97(97)	5(5)	5(5)	3(3)	5(5)	5(5)	3(3)	0(0)	5(5)	2(-5)	97(97)	-1	0
2(2)	3(3)	91(88)	7(7)	7(7)	3(3)	7(7)	7(7)	5(5)	0(0)	7(7)	-4(2)	113(114)	-1	0
5(5)	3(3)	85(77)	9(9)	9(9)	5(5)	9(9)	9(9)	7(7)	0(0)	9(9)	2(-3)	143(141)	-1	0
5(5)	0(0)	94(94)	11(11)	11(11)	7(7)	11(11)	11(11)	0(0)	0(0)	0(0)	3(-5)	174(174)	-1	0
3(5)	0(0)	77(71)	13(13)	13(13)	9(9)	13(13)	13(13)	0(0)	0(0)	0(0)	2(1)	196(201)	1	4
5(5)	0(0)	97(97)	17(17)	15(15)	11(11)	17(17)	15(15)	0(0)	0(0)	0(0)	2(-5)	250(250)	5	1
5(5)	0(0)	94(94)	19(19)	17(17)	13(13)	19(19)	17(17)	0(0)	0(0)	0(0)	3(-5)	282(282)	-1	0
5(5)	5(5)	97(97)	21(21)	19(19)	15(15)	21(21)	19(19)	9(9)	1(1)	0(0)	2(-5)	357(357)	-1	0
3(3)	0(0)	100(100)	23(23)	21(21)	17(17)	23(23)	21(21)	0(0)	0(0)	0(0)	2(-4)	379(379)	6	1
4(4)	0(0)	100(100)	23(23)	23(23)	19(19)	25(25)	23(23)	0(0)	0(0)	0(0)	3(-4)	417(417)	-1	0
5(5)	3(3)	97(97)	25(25)	25(25)	21(21)	27(27)	25(25)	11(11)	0(0)	11(11)	2(-2)	440(440)	7	1
4(4)	0(0)	82(82)	27(27)	27(27)	23(23)	29(29)	27(27)	0(0)	0(0)	0(0)	2(-4)	475(475)	-1	0
3(2)	0(0)	51(48)	29(29)	29(29)	25(25)	31(31)	29(29)	0(0)	0(0)	0(0)	-4(3)	495(502)	8	1

LONG CH : 8 PROB CH : 100WhenJobc

a) Ligne n° 1

PRIM	ATTR	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
G(G)	( )	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	-1	0
3(3)	0(0)	82(77)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	0(0)	0(0)	2(-1)	41(39)	1	4
5(5)	0(0)	90(87)	5(5)	3(3)	3(3)	5(5)	3(3)	0(0)	0(0)	0(0)	2(-3)	94(93)	5	1
5(5)	0(0)	90(87)	9(9)	5(5)	5(5)	7(7)	5(5)	0(0)	0(0)	0(0)	2(-3)	125(124)	6	1
3(3)	0(0)	90(85)	13(13)	7(7)	7(7)	9(9)	7(7)	0(0)	0(0)	0(0)	2(-5)	155(154)	7	1
4(4)	0(0)	87(82)	13(13)	9(9)	9(9)	11(11)	9(9)	0(0)	0(0)	0(0)	-5(2)	192(193)	-1	0
5(5)	0(0)	90(85)	15(15)	11(11)	11(11)	13(13)	11(11)	0(0)	0(0)	0(0)	1(-3)	213(212)	8	1
1(1)	0(0)	77(77)	19(19)	13(13)	13(13)	15(15)	13(13)	0(0)	0(0)	0(0)	3(-4)	300(298)	9	1
2(2)	0(0)	65(55)	21(21)	15(15)	13(13)	17(17)	15(15)	0(0)	0(0)	0(0)	4(-3)	323(318)	-1	1
1(1)	0(0)	80(77)	21(21)	17(17)	15(15)	17(17)	17(17)	0(0)	0(0)	0(0)	-4(2)	335(337)	10	1
2(2)	0(0)	65(65)	23(23)	19(19)	15(15)	19(19)	19(19)	0(0)	0(0)	0(0)	4(-3)	360(356)	-1	0
5(5)	1(1)	75(75)	25(25)	21(21)	17(17)	21(21)	21(21)	1(1)	0(0)	0(0)	4(-1)	383(381)	-1	0
5(5)	3(3)	95(90)	27(27)	23(23)	19(19)	23(23)	23(23)	3(3)	0(0)	1(1)	2(-3)	408(407)	-1	0
5(5)	3(3)	97(97)	29(29)	25(25)	21(21)	25(25)	25(25)	5(5)	0(0)	3(3)	2(-4)	433(433)	11	1

LONG CH : 9 PROB CH : 100errorwill

b) Ligne n° 2

Planche 6 (suite)

	PRIM	ATTR	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
	G(G)	( )	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	-1	7
b	5(5)	3(3)	88(80)	1(1)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	1(1)	2(-2)	42(41)	1	3
	2(2)	0(0)	60(54)	3(3)	3(3)	3(3)	3(3)	3(3)	0(0)	0(0)	0(0)	-4(2)	75(77)	4	4
a	5(5)	0(0)	91(91)	7(7)	5(5)	7(7)	5(5)	7(7)	0(0)	0(0)	0(0)	4(-5)	126(126)	8	2
á	6(6)	0(0)	91(88)	9(9)	7(7)	9(9)	7(7)	9(9)	0(0)	0(0)	0(0)	4(-4)	162(161)	10	1
e	3(3)	0(0)	91(88)	13(13)	9(9)	11(11)	9(9)	11(11)	0(0)	0(0)	0(0)	2(-5)	190(191)	11	4
	3(3)	0(0)	97(97)	15(15)	11(11)	13(13)	13(13)	13(13)	0(0)	0(0)	0(0)	3(-4)	241(241)	15	2
d	5(5)	3(3)	100(100)	17(17)	13(13)	15(15)	15(15)	15(15)	3(3)	0(0)	3(3)	2(-4)	277(277)	-1	0
	3(3)	0(0)	97(97)	19(19)	15(15)	17(17)	17(17)	17(17)	0(0)	0(0)	0(0)	2(-4)	341(341)	17	1
o	4(4)	0(0)	97(97)	19(19)	17(17)	19(19)	19(19)	19(19)	0(0)	0(0)	0(0)	2(-2)	378(378)	-1	0
	5(5)	0(0)	91(88)	21(21)	19(19)	21(21)	21(21)	21(21)	0(0)	0(0)	0(0)	2(-1)	402(401)	18	1
n	4(4)	0(0)	80(80)	23(23)	21(21)	23(23)	23(23)	23(23)	0(0)	0(0)	0(0)	4(-2)	435(435)	-1	0
ó	5(5)	3(3)	100(100)	25(25)	23(23)	25(25)	25(25)	25(25)	5(5)	0(0)	5(5)	2(-3)	503(503)	-1	0

LONG CH : 6 PROB CH : 65asedol

c) *Ligne n° 3*

	PRIM	ATTR	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
	G(G)	( )	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	-1	0
v	1(1)	0(0)	80(77)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	0(0)	0(0)	-3(5)	42(44)	-1	3
	2(2)	0(0)	75(72)	3(3)	3(3)	1(1)	3(3)	1(1)	0(0)	0(0)	0(0)	2(-3)	62(60)	1	4
a	5(5)	0(0)	80(75)	7(7)	5(5)	5(5)	5(5)	5(5)	0(0)	0(0)	0(0)	-3(4)	115(117)	-1	0
r	5(5)	0(0)	87(82)	9(9)	7(7)	7(7)	7(7)	7(7)	0(0)	0(0)	0(0)	1(-5)	139(138)	5	1
i	5(5)	1(1)	92(85)	13(13)	9(9)	9(9)	9(9)	9(9)	1(1)	0(0)	0(0)	-5(4)	171(172)	6	4
a	5(5)	0(0)	87(87)	15(15)	11(11)	11(11)	11(11)	13(13)	0(0)	0(0)	0(0)	5(-5)	223(223)	10	1
t	5(5)	2(2)	87(87)	17(17)	13(13)	13(13)	13(13)	15(15)	0(0)	0(0)	1(1)	2(-4)	251(250)	11	2
i	5(5)	1(1)	87(85)	19(19)	15(15)	15(15)	15(15)	17(17)	3(3)	0(0)	0(0)	-4(2)	278(279)	-1	1
	3(3)	0(0)	92(90)	21(21)	17(17)	17(17)	17(17)	19(19)	0(0)	0(0)	0(0)	-5(4)	301(302)	13	1
o	4(4)	0(0)	95(95)	21(21)	19(19)	19(19)	19(19)	21(21)	0(0)	0(0)	0(0)	2(-5)	339(339)	-1	0
	5(5)	0(0)	77(75)	23(23)	21(21)	21(21)	21(21)	23(23)	0(0)	0(0)	0(0)	-3(4)	361(363)	-1	0
n	4(4)	0(0)	75(72)	25(25)	23(23)	23(23)	23(23)	25(25)	0(0)	0(0)	0(0)	-2(4)	395(397)	14	8

LONG CH : 8 PROB CH : 83variatio

d) *Ligne n° 4*

Planche 6 (suite)

$tg\ a = \frac{y}{l}$, où y est la déviation verticale maximale.

Cette déviation y est proportionnelle à la taille des caractères :

$$y = k_1 \times \text{taille}$$

Or d'une part, un rapport à peu près constant existe entre la taille et la chasse :

$$k'_2 \times \text{chasse} \leq \text{taille} \leq k''_2 \times \text{chasse}$$

et d'autre part, si n est le nombre de caractères composant la ligne de longueur l :

$$l = n \times \text{chasse}$$

On trouve après remplacement :

$$(1) \qquad n \times tg\ a \leq k_1\ k''_2$$

Un rapprochement peut être fait entre cette relation et celle établie lors du chapitre IV, concernant l'erreur de déviation angulaire tolérée par un algorithme de détection des lignes de texte basé sur l'analyse de l'histogramme.

$$tg\ a \leq \frac{d}{l}$$

d et l étant la valeur de l'interligne et de la ligne de texte respectivement.

Après remplacement, l'on obtient :

$$(2) \quad n \times tg\ a \leq \frac{d}{\text{chasse}}$$

Le cas :

$$\frac{d}{\text{chasse}} \leq k_1 k_2''$$

est très intéressant parce que la relation (1) est satisfaite automatiquement ; la détection des lignes à l'aide d'un histogramme pourrait donc satisfaire la précison requise par l'étage extracteur.

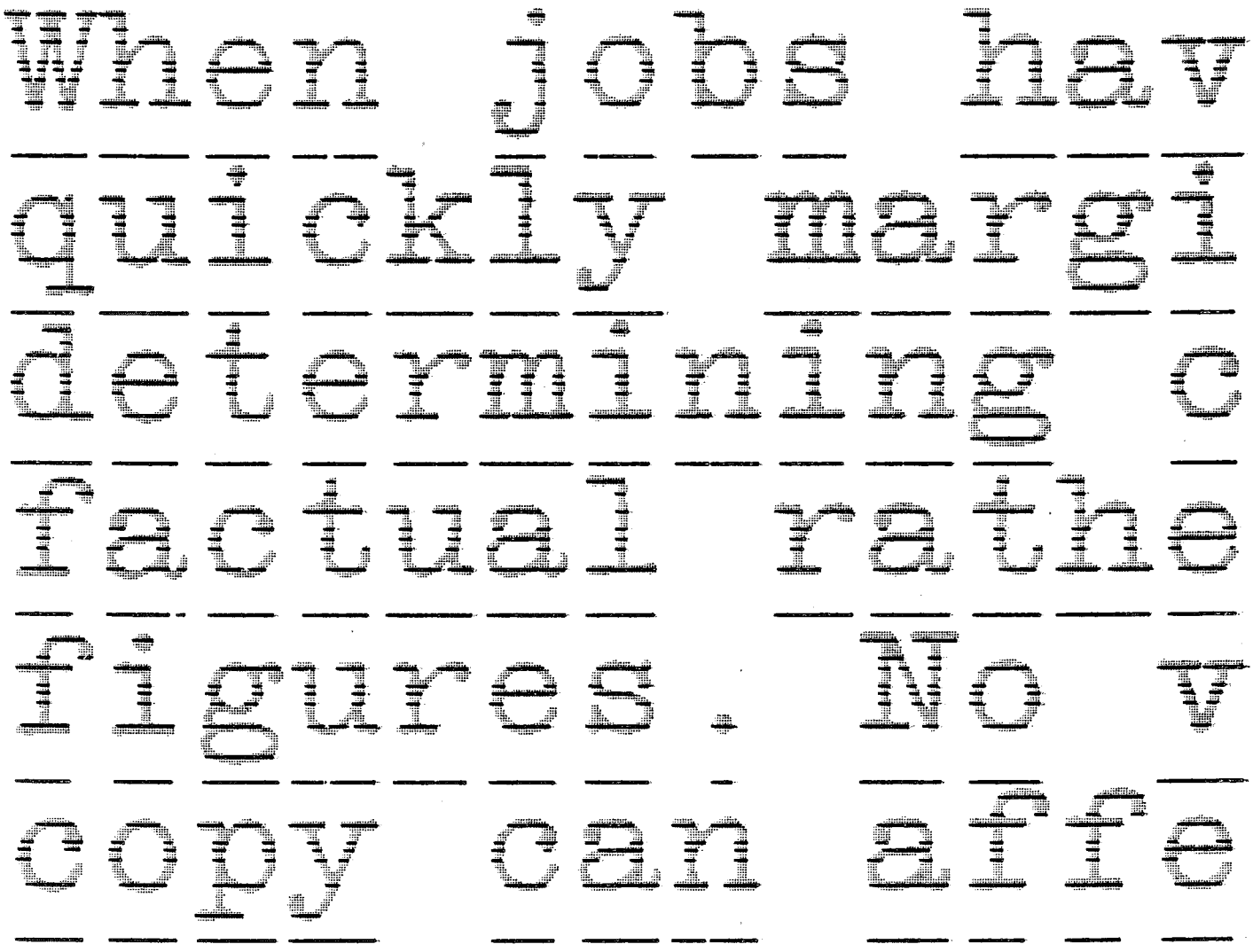
g) Enfin la Planche 7 est composée d'une police de **machine à écrire** se trouvant normalement en dehors du champs d'application de notre système. La spécificité de ces polices est la largeur toujours constante des caractères, quelque soit le nombre de primitives principales qui les composent : un **W** occupera le même espace que **N** et **i** . Ceci entraîne des déformations très importantes des primitives principales 1 et 2. De plus, pour combler l'espace entre les caractères et donner un aspect équilibré au texte, presque toutes les primitives sont pourvues d'empattements. A ces difficultés, inhérentes à la police, il faut ajouter le choix délibéré d'une taille petite des caractères et une graisse relativement faible. Passons en revue les erreurs détectées :

1 - **Omissions**

Les primitives 1 et 2 composant le caractère **W** ont disparu parce qu'ayant une pente par trop différente (pour les raisons invoquées plus haut) du modèle (il s'agit du caractère **v** à la fin de la première ligne).

2 - **Substitutions**

La majorité des substitutions concernent la primitive 5. Deux classes peuvent être distinguées :



(Type writer 100)

Planche 7

	PRIM G(G)	ATTR ( )	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
			0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	1	18
h	2(2)	0(0)	90(90)	9(9)	9(9)	5(5)	9(9)	9(9)	0(0)	0(0)	0(0)	1(-5)	79(79)	19	3
e	5(5)	0(0)	85(85)	9(9)	11(11)	7(7)	11(11)	11(11)	0(0)	0(0)	0(0)	4(-5)	97(97)	22	1
n	3(3)	0(0)	80(75)	11(11)	13(13)	9(9)	13(13)	13(13)	0(0)	0(0)	0(0)	-4(4)	114(115)	23	6
j	5(5)	0(0)	80(80)	13(13)	15(15)	11(11)	17(17)	15(15)	0(0)	0(0)	0(0)	4(-5)	152(151)	29	2
o	5(5)	0(0)	70(65)	15(15)	17(17)	13(13)	19(19)	17(17)	0(0)	0(0)	0(0)	-5(4)	170(171)	31	3
b	5(5)	5(1)	70(65)	17(17)	19(19)	17(17)	21(21)	19(19)	9(9)	1(0)	0(0)	-1(5)	238(240)	-1	0
s	3(3)	0(0)	85(80)	19(19)	21(21)	19(19)	23(23)	21(21)	0(0)	0(0)	0(0)	2(-4)	260(259)	34	2
h	4(4)	0(0)	95(95)	19(19)	23(23)	19(19)	25(25)	23(23)	0(0)	0(0)	0(0)	4(-4)	280(280)	-1	2
a	5(5)	3(3)	95(95)	21(21)	25(25)	21(21)	27(27)	25(25)	11(11)	0(0)	9(9)	2(-4)	300(300)	36	2
v	4(4)	0(0)	90(90)	21(21)	27(27)	21(21)	29(29)	27(27)	0(0)	0(0)	0(0)	4(-5)	320(320)	38	3
	6(6)	0(0)	90(90)	23(23)	29(29)	23(23)	31(31)	31(31)	0(0)	0(0)	0(0)	4(-4)	346(346)	41	4
	5(2)	3(0)	90(90)	25(27)	31(31)	25(25)	35(35)	33(33)	13(0)	0(0)	11(0)	-2(1)	409(409)	45	3
	5(5)	0(0)	90(90)	27(27)	33(33)	27(27)	37(37)	35(35)	0(0)	0(0)	0(0)	4(-5)	429(429)	48	6
	5(5)	0(0)	70(70)	29(29)	35(35)	31(31)	41(41)	39(39)	0(0)	0(0)	0(0)	5(-2)	464(463)	54	1
	1(1)	0(0)	95(95)	31(31)	37(37)	33(33)	43(43)	41(41)	0(0)	0(0)	0(0)	2(-4)	489(489)	55	2
	2(2)	0(0)	80(80)	33(33)	39(39)	33(33)	45(45)	43(43)	0(0)	0(0)	0(0)	2(-4)	500(500)	-1	3

a) *Ligne n° 1*

	PRIM G(G)	ATTR ( )	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
			0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	-1	0
q	3(3)	0(0)	75(75)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	0(0)	0(0)	4(-2)	40(39)	1	2
u	5(4)	4(0)	75(70)	3(1)	3(3)	1(1)	3(3)	3(3)	0(0)	1(0)	0(0)	-5(-5)	58(58)	3	1
i	5(5)	0(0)	80(70)	5(5)	5(5)	3(3)	5(5)	5(5)	0(0)	0(0)	0(0)	-5(2)	78(79)	4	2
c	5(5)	0(0)	95(95)	7(7)	7(7)	5(5)	7(7)	7(7)	0(0)	0(0)	0(0)	3(-5)	97(97)	6	3
k	5(5)	3(3)	95(95)	9(9)	9(9)	7(7)	9(9)	9(9)	1(1)	0(0)	1(1)	4(-5)	124(124)	9	1
l	3(3)	0(0)	85(85)	11(11)	11(11)	9(9)	11(11)	11(11)	0(0)	0(0)	0(0)	4(-2)	152(151)	10	5
y	5(5)	3(3)	85(80)	13(13)	15(15)	11(11)	15(15)	13(13)	3(3)	0(0)	3(3)	2(-4)	190(189)	15	7
	5(5)	3(3)	85(80)	17(17)	17(17)	15(15)	19(19)	17(17)	5(5)	0(0)	5(5)	2(-4)	234(233)	22	1
m	1(1)	0(0)	80(75)	19(19)	19(19)	17(17)	21(21)	19(19)	0(0)	0(0)	0(0)	2(-4)	265(264)	23	2
a	2(2)	4(4)	75(65)	21(21)	21(21)	17(17)	23(23)	21(21)	0(0)	3(3)	0(0)	-4(2)	275(276)	25	1
r	5(5)	0(0)	100(100)	23(23)	23(23)	19(19)	25(25)	23(23)	0(0)	0(0)	0(0)	4(-2)	332(332)	26	2
g	2(2)	0(0)	95(95)	25(25)	25(25)	19(19)	27(27)	25(25)	0(0)	0(0)	0(0)	2(-2)	345(345)	28	1
i	5(5)	0(0)	80(75)	25(25)	27(27)	19(19)	29(29)	27(27)	0(0)	0(0)	0(0)	-5(4)	357(358)	29	5
	5(5)	0(0)	75(75)	27(27)	29(29)	23(23)	33(33)	31(31)	0(0)	0(0)	0(0)	4(-4)	388(388)	34	2
	2(2)	0(0)	95(95)	31(31)	31(31)	25(25)	35(35)	33(33)	0(0)	0(0)	0(0)	5(-2)	414(414)	36	3
	3(3)	0(0)	75(75)	33(33)	33(33)	27(27)	37(37)	35(35)	0(0)	0(0)	0(0)	5(-2)	447(446)	39	2
	2(5)	4(4)	90(90)	33(33)	35(35)	27(27)	39(39)	37(37)	0(0)	5(5)	0(0)	4(4)	462(463)	41	3
	5(5)	3(3)	100(100)	35(35)	37(37)	29(29)	41(41)	39(39)	7(7)	0(0)	7(7)	4(-4)	494(494)	44	1

b) *Ligne n° 2*

	PRIM G(G)	ATTR ( )	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
			0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	-1	0
d	3(3)	0(0)	80(75)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	0(0)	0(0)	-5(4)	39(40)	1	2
e	5(5)	3(3)	80(80)	3(3)	3(3)	1(1)	3(3)	3(3)	1(1)	0(0)	1(1)	2(-4)	58(57)	3	1
t	3(3)	0(0)	90(90)	5(5)	5(5)	3(3)	5(5)	5(5)	0(0)	0(0)	0(0)	2(-4)	78(78)	4	5
e	5(5)	3(3)	75(70)	7(7)	7(7)	5(5)	9(9)	7(7)	3(3)	0(0)	3(3)	2(-4)	122(121)	9	2
r	3(3)	0(0)	85(85)	9(9)	9(9)	7(7)	11(11)	9(9)	0(0)	0(0)	0(0)	2(-4)	151(151)	11	6
	5(2)	0(0)	85(85)	11(13)	11(11)	9(9)	15(15)	11(11)	0(0)	0(0)	0(0)	4(-5)	192(193)	17	4
m	5(5)	0(0)	90(90)	15(15)	13(13)	11(11)	17(17)	13(13)	0(0)	0(0)	0(0)	4(-2)	223(222)	21	1
i	5(1)	0(0)	75(75)	15(15)	15(15)	13(13)	19(19)	15(15)	0(0)	0(0)	0(0)	4(-2)	235(234)	22	2
n	5(5)	0(0)	75(70)	17(17)	17(17)	13(13)	21(21)	17(17)	0(0)	0(0)	0(0)	-5(4)	246(247)	24	2
i	5(5)	1(1)	95(95)	19(19)	19(19)	15(15)	23(23)	19(19)	5(5)	0(0)	0(0)	4(-5)	271(271)	26	3
n	5(5)	0(0)	90(90)	21(21)	21(21)	17(17)	25(25)	21(21)	0(0)	0(0)	0(0)	2(-5)	298(298)	29	3
g	5(5)	0(0)	80(70)	23(23)	23(23)	19(19)	27(27)	23(23)	0(0)	0(0)	0(0)	-5(4)	317(318)	32	3
	5(5)	1(1)	95(95)	25(25)	25(25)	21(21)	29(29)	25(25)	7(7)	0(0)	0(0)	2(-4)	344(344)	35	3
	5(5)	0(0)	95(95)	27(27)	27(27)	23(23)	31(31)	27(27)	0(0)	0(0)	0(0)	4(-5)	372(372)	38	3
	5(5)	0(0)	85(80)	29(29)	29(29)	25(25)	33(33)	29(29)	0(0)	0(0)	0(0)	-5(4)	391(392)	41	1
	5(5)	4(4)	75(75)	31(31)	31(31)	27(27)	35(35)	31(31)	0(0)	1(1)	0(0)	2(-2)	410(409)	42	3
	2(2)	4(4)	90(90)	31(31)	33(33)	27(27)	37(37)	31(31)	0(0)	1(1)	0(0)	4(-1)	425(424)	45	1
	3(3)	0(0)	95(95)	33(33)	35(35)	29(29)	39(39)	33(33)	0(0)	0(0)	0(0)	1(-4)	484(484)	46	3

c) *Ligne n° 3*

Planche 7 (suite)

	PRIM	ATTR	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
	G(G)	( )	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	1	2
ó	5(5)	2(2)	95(95)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	0(0)	1(1)	4(-4)	49(49)	3	5
a	6(6)	0(0)	75(75)	3(3)	3(3)	5(5)	3(3)	5(5)	0(0)	0(0)	0(0)	4(-4)	87(88)	8	4
c	3(3)	0(0)	80(80)	5(5)	5(5)	9(9)	7(7)	7(7)	0(0)	0(0)	0(0)	4(-2)	116(115)	12	4
t	5(5)	3(3)	85(80)	7(7)	7(7)	13(13)	11(11)	9(9)	3(3)	0(0)	5(5)	-4(4)	158(159)	16	4
u	5(5)	0(0)	80(80)	9(9)	9(9)	15(15)	13(13)	13(13)	0(0)	0(0)	0(0)	2(-5)	188(188)	20	2
a	5(5)	0(0)	85(85)	11(11)	11(11)	17(17)	15(15)	15(15)	0(0)	0(0)	0(0)	3(-2)	207(206)	22	6
l	5(5)	0(0)	80(80)	13(13)	13(13)	19(19)	19(19)	19(19)	0(0)	0(0)	0(0)	5(-5)	241(241)	28	2
r	5(5)	3(3)	85(80)	15(15)	15(15)	21(21)	21(21)	21(21)	5(5)	0(0)	7(7)	2(-4)	271(270)	30	2
a	2(2)	0(0)	100(100)	19(19)	17(17)	23(23)	23(23)	23(23)	0(0)	0(0)	0(0)	5(-4)	339(339)	32	2
t	5(5)	0(0)	70(70)	21(21)	19(19)	25(27)	25(27)	25(27)	0(0)	0(0)	0(0)	2(-5)	372(388)	34	7
h	5(5)	3(3)	75(70)	23(23)	21(21)	29(29)	29(29)	29(29)	7(7)	0(0)	9(9)	2(-4)	416(415)	41	4
e	5(5)	3(3)	95(95)	25(25)	23(23)	31(31)	31(31)	33(33)	9(9)	0(0)	11(11)	2(-2)	446(446)	45	2
	1(5)	0(0)	80(80)	27(27)	25(25)	33(33)	33(33)	35(35)	0(0)	0(0)	0(0)	2(4)	466(466)	47	1
	3(3)	0(0)	90(85)	29(29)	27(27)	35(35)	35(35)	37(37)	0(0)	0(0)	0(0)	-4(4)	483(484)	48	5

d) Ligne n° 4

	PRIM	ATTR	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
	G(G)	( )	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	1	2
ó	5(5)	2(2)	100(100)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	0(0)	1(1)	4(-4)	49(49)	3	4
i	5(5)	1(1)	95(95)	3(3)	3(3)	3(3)	3(3)	3(3)	3(3)	0(0)	0(0)	4(-5)	88(88)	7	2
g	3(5)	0(4)	70(65)	5(5)	5(5)	5(5)	5(5)	5(5)	0(0)	0(1)	0(0)	-4(5)	115(116)	9	2
	5(2)	4(4)	75(70)	5(5)	7(7)	5(5)	7(7)	7(7)	0(0)	1(1)	0(0)	-4(4)	130(131)	11	2
u	5(5)	0(0)	70(70)	7(7)	9(9)	7(7)	9(9)	9(9)	0(0)	0(0)	0(0)	2(-5)	152(151)	-1	0
r	4(5)	0(0)	75(70)	9(9)	11(11)	9(11)	11(11)	11(11)	0(0)	0(0)	0(0)	2(4)	170(171)	13	2
e	5(5)	0(0)	95(95)	11(11)	13(13)	13(13)	13(13)	13(13)	0(0)	0(0)	0(0)	4(-5)	191(191)	15	2
s	3(3)	0(0)	85(85)	15(15)	15(15)	15(15)	15(15)	15(15)	0(0)	0(0)	0(0)	2(-4)	224(224)	17	6
	6(6)	0(0)	80(80)	17(17)	17(17)	17(17)	19(19)	17(17)	0(0)	0(0)	0(0)	4(-4)	271(271)	23	4
N	5(5)	3(3)	80(80)	19(19)	19(19)	21(21)	23(23)	19(19)	5(5)	0(0)	5(5)	2(-4)	372(371)	27	1
	1(4)	3(0)	55(55)	19(21)	21(21)	23(21)	25(25)	21(21)	5(0)	0(0)	5(0)	-5(1)	382(384)	28	1
o	5(5)	3(3)	75(75)	23(23)	23(23)	23(23)	27(27)	23(23)	7(7)	0(0)	7(7)	1(-4)	391(390)	-1	0
	3(3)	0(0)	85(85)	25(25)	25(25)	25(25)	29(29)	25(25)	0(0)	0(0)	0(0)	4(-2)	408(407)	29	2
v	4(4)	0(0)	95(95)	25(25)	27(27)	25(25)	31(31)	27(27)	0(0)	0(0)	0(0)	4(-1)	428(428)	-1	0
	1(1)	0(0)	95(95)	27(27)	29(29)	27(27)	33(33)	29(29)	0(0)	0(0)	0(0)	2(-4)	488(488)	31	2
	2(2)	0(0)	90(90)	29(29)	31(31)	27(27)	35(35)	31(31)	0(0)	0(0)	0(0)	2(-5)	499(499)	-1	3

e) Ligne n° 5

	PRIM	ATTR	PROB	PTS1	PTS2	PTS3	PTS4	PTS5	PTS6	PTS7	PTS8	SONC	POS2	OUTSP	OUTSL
	G(G)	( )	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	0(0)	-1	0
c	3(3)	0(0)	80(75)	1(1)	1(1)	1(1)	1(1)	1(1)	0(0)	0(0)	0(0)	-2(4)	42(43)	1	4
o	3(3)	0(0)	85(80)	3(3)	5(5)	3(3)	5(5)	3(3)	0(0)	0(0)	0(0)	-2(4)	77(78)	5	2
	4(4)	0(0)	75(60)	3(3)	7(7)	3(3)	7(7)	5(5)	0(0)	0(0)	0(0)	4(-1)	98(96)	7	1
p	5(5)	4(4)	90(80)	5(5)	9(9)	5(5)	9(9)	7(7)	0(0)	1(1)	0(0)	3(-1)	117(116)	8	2
	4(4)	0(0)	85(75)	7(7)	11(11)	5(5)	11(11)	9(9)	0(0)	0(0)	0(0)	2(-1)	136(135)	-1	0
y	1(1)	0(0)	75(55)	9(9)	13(13)	7(7)	13(13)	11(11)	0(0)	0(0)	0(0)	2(-2)	155(153)	10	2
c	2(2)	4(4)	75(60)	11(11)	15(15)	7(7)	15(15)	13(13)	0(0)	3(3)	0(0)	-3(4)	165(167)	-1	0
a	3(3)	0(0)	85(80)	13(13)	17(17)	9(9)	17(17)	15(15)	0(0)	0(0)	0(0)	-4(4)	224(225)	12	9
n	5(5)	0(0)	80(80)	15(15)	21(21)	11(11)	23(23)	19(19)	0(0)	0(0)	0(0)	5(-5)	277(277)	21	3
a	5(2)	0(0)	90(90)	17(19)	23(23)	13(13)	25(25)	21(21)	0(0)	0(0)	0(0)	4(3)	298(299)	24	1
ó	1(5)	0(0)	85(85)	19(19)	25(25)	15(15)	27(27)	23(23)	0(0)	0(0)	0(0)	1(4)	317(317)	25	6
ó	5(5)	0(0)	80(80)	21(21)	27(27)	17(17)	31(31)	27(27)	0(0)	0(0)	0(0)	5(-2)	388(387)	31	3
e	5(5)	3(3)	100(100)	23(23)	29(29)	19(19)	33(33)	29(29)	1(1)	0(0)	1(1)	4(-4)	415(415)	34	4
	5(5)	3(3)	100(100)	25(25)	31(31)	21(21)	35(35)	31(31)	3(3)	0(0)	5(5)	4(-4)	452(452)	38	2
	3(3)	0(0)	95(95)	27(27)	33(33)	23(23)	37(37)	33(33)	0(0)	0(0)	0(0)	1(-4)	482(482)	40	4

f) Ligne n° 6

Planche 7 (suite)

a) 5 transformée en 1 ou 2 (h , m, r)

L'erreur doit être cherchée dans le fait que le modèle des primitives 1 et 2 est dénaturé par des empattements sur la sonde 1 (voir figure 2). La substitution est aussi favorisée par la présence d'empattements sur la sonde 3 dans la ligne de texte, par les erreurs d'échantillonnage (visibles à l'oeil nu sur les primitives 5) et par l'épaisseur réduite du trait. Un choix plus judicieux des côtés libres pourrait apporter la solution à ce problème, mais nous n'avons pas exploré cette hypothèse.

b) 5 transformé en 4 (u)

Une seule substitution de ce genre est apparue dans le texte (deuxième primitive du caractère u se trouvant dans la cinquième ligne), justifiée par l'empattement sur la sonde 1 et le déplacement du segment de la sonde 3, dû à la courbure imposée par la liaison à la première primitive.

Il y a aussi une fausse primitive dans la quatrième ligne : la sonde 4, chargée d'établir la continuité du trait dans la moitié supérieure du corps de texte, à failli à sa tâche et donné ainsi naissance à une primitive 6 dans le caractère a .

Enfin, nous ne considèrons pas le cas du caractère g comme une erreur parce que son graphisme dans cet exemple ne le rend décomposable en aucune des primitives principales.

Nous ne donnons pas les résultats de la recombinaison pour une raison déjà invoquée : à l'heure actuelle le dictionnaire n'est ni complet ni tout-à-fait au point. Dans cet exemple les erreurs étaient crées par la présence d'empattements trop longs que notre système a interprété comme barres alors que le dictionnaire contenait une description exempte de barres.

C - CONCLUSIONS

Les remarques faites lors de l'évaluation de la méthode permettent au lecteur de se faire une idée suffisamment précise sur les limitations actuelles du système et ses améliorations possibles. Toutefois, tous les faits discutés n'ont pas la même importance. Il y en a parmi eux qui relèvent du détail d'implémentation : ainsi la détection des espaces n'est pas encore mise en oeuvre bien que son action soit bénéfique à plusieurs niveaux (plus de flexibilité quant aux inclinaisons, évaluation plus précise de la recombinaison des primitives). Les deux points développés ci-après apparaissent plus fondamentaux.

D'abord, le principal défaut, celui qui à l'heure présente limite le plus directement les performances du système, se situe au niveau du dictionnaire contenant la description des caractères, conçu dans une forme très simple et statique. Il s'est avéré à l'expérience qu'il était au contraire hautement souhaitable de le concevoir sous forme dynamique afin de compléter de manière incrémentale la description d'un caractère avec d'autres règles de production. Nous avons pu le constater lors de l'essai sur la Planche 7 (machine à écrire) où la presque totalité des primitives principales était détectée mais leur exploitation mise en défaut par l'étape de recombinaison. On devrait remédier à cet état de fait en ajoutant au système un étage interactif (en ligne) d'aide à la construction du dictionnaire. Il faut concevoir cette interaction comme une conception assistée par ordinateur plutôt qu'un apprentissage supervisé au sens classique du terme, bien que les deux notions soient finalement voisines.

Par ailleurs, l'information très riche sous forme de primitives pricipales et secondaires dont le dictionnaire dipose est pour l'instant mal exploité par l'étage de recombinaison. Celui-ci ne fait qu'à de rares exceptions près appel au deuxième choix de primitive concernant un groupement, alors que cette information est très significative chaque fois qu'un doute existe quant à la recombinaison. Pire, lorsque le premier choix d'une primitive principale n'a acquis qu'une note de confiance trop faible à l'issue de l'extraction, l'étage de recombinaison abandonne l'information contenue dans les primitives secondaires, en reportant sur la chaîne totale des caractères recombinés une estimation plus faible alors qu'il a suffisamment d'informations pour interpréter cette zone de mauvaise qualité. Une programmation plus poussée de cet étage s'impose donc.

Compte tenu de ces limitations présentes, nous n'avons pas la prétention d'avoir résolu de manière radicale tous les points faibles dont pâtissent les autres systèmes de lecture optique. Les améliorations que nous apportons sont néanmoins significatives à nos yeux. Ainsi :

* la segmentation a priori est remplacée par une phase de recombinaison qui offre plus de souplesse pour analyser de façon logique et exhaustive les solutions possibles sans que cela alourdisse réellement la recherche, grâce à son caractère dynamique ;

* la normalisation devient chez nous plus riche qu'une double affinité en x et en y comme la pratiquent les méthodes classiques du fait de l'exploitation de l'histogramme permettant l'ajustement des sondes aux caractéristiques internes des caractères (creux et barres) ;

* la squelettisation, technique classique de la lecture optique, est remplacée par les segments d'intersection des sondes avec les caractères. S'il y a biunivocité **segment d'intersection** sur les sondes - **squelette** l'épaisseur du trait nous apparaît en effet comme une perte d'information préjudiciable. Semblablement, bien que l'information apportée par les empattements ne soit pas exploitée pour l'instant, elle peut constituer une base solide pour certaines vérifications lors de la recombinaison dans la mesure où les empattements ont à ce niveau un effet discriminant intéressant ;

* enfin, la pauvreté remarquable des paramètres qui dépendent de la police dans notre modèle nous permet un apprentissage automatique omnipolice, simple et efficace.

Arrivé à ce point, le lecteur a sans doute saisi la principale caractéristique de cette méthode : sa grande simplicité. En effet, elle fait appel à peu de paramètres, emploie peu de primitives et se sert de peu de sondes pour les détecter. De plus, une liaison existant entre la génération des caractères et leur reconnaissance telle que nous la pratiquons, elle peut permettre non seulement la reconnaissance des caractères mais aussi l'identification de la police par mesures sur les formes réelles correspondant aux primitives au prix certes d'un accroissement de la résolution ordinairement admise (12 points/mm au lieu de 8 points/mm) mais tout à fait accessible aux capteurs d'aujourd'hui. Cette caractéristique est particulièrement intéressante dans le cas d'une application du type fac-similé où l'on

désire une restitution aussi voisine que possible du texte d'origine. Le taux de compression ainsi réalisé (codage des caractères et identité de la police) dépasserait de beaucoup celui réalisé par les systèmes classiques, sans dégradation notable de l'information graphique supportant l'information sémantique.

B I B L I O G R A P H I E

/2.1/ G. GAILLARD, M. BERTHOD.
Panorama des techniques d'extraction de traits caractéristiques en lecture optique des caractères .
2ème congrès AFCET-IRIA, 1979, tome III.

/2.2/ B. BLESSER, R. SHILLMAN, T. KUKLINSKI, C. COX, M. EDEN, J. VENTURA.
A Theoretical Approach for Character Recognition based on Phenomenological Attributes .
Document interne du MIT.

/2.3/ R. SHILLMAN, T. KUKLINSKI, B. BLESSER.
Experimental Methodologies for Character Recognition based on Phenomenological Attributes .
Second Joint Conference on Pattern Recognition, 1974, Copenhagen, pp. 195-201.

/2.4/ C. COX, Ph. COUEIGNOUX, B. BLESSER, M. EDEN.
Skeletons : a link between theoretical and physical letter description .
A paraître dans Pattern Recognition.

/2.5/ A. CHEHIKIAN
Conception et réalisation d'un automate de reconnaissance de caractères alphanumériques .
Thèse d'état, 1977, INP Grenoble.

/2.6/ A. CHEHIKIAN
Un algorithme de séparation des primitives verticales, horizontales et obliques d'une forme bidimensionnelle .
Congrès AFCET-IRIA, 1978, tome II, pp. 271-788.

/2.7/ A. CHEHIKIAN, F. HADJ HASSAN
Une description structurelle, peu redondante, des caractères alphanumériques, au moyen de variables binaires .
Congrès AFCET-IRIA, 1979, tome III, pp. 316-323.

/2.8/ Ph. COUEIGNOUX
Character Generation by Computer .
Computer Graphics and Image Processing, 16 - 1981, pp. 240-249

/2.9/ Ph. COUEIGNOUX
Generation of Roman Printed Fonts .
Ph.D. Thesis
MIT, dept. Elec. Eng. June 1975.

/3.1/ R.O. DUDA, P.E. HART
Pattern Classification and Scene Analysis .
Ed. Wiley-Interscience, 1973.

/3.2/ D.E. TROXEL
Feature Selection for low error rate OCR .
Pattern Recognition 1976, vol 8 pp. 73-76

/4.1/ E.G. JOHNSTON
Printed text discrimination .
CGIP, nr. 3, 1974, pp. 83-89

/4.2/ S.J. MASON, J.K. CLEMENS
Character Recognition in an Experimental Reading Machine for the Blind .
Recognising Patterns. Edité par P. A. Kolers et M. Eden, publié par MIT-Press, 1968.

/4.3/ B. LAY, O. BOMSEL
Lecture optique automatique de documents multipolice .
ENS des MINES de Saint-Etienne, dépt. Informatique, 1980.

/4.4/ P.V. SANKAR, A. ROSENFELD
Hierarchical Representation of Wave Forms .
IEEE on Pattern Analysis and Machine Intelligence, vol PAMI 1, nr. 1, Jan. 1979.

/4.5/ M. JACNO
Anatomie de la lettre .
Collection Caractère - Ecole Estienne, 1978.

/4.6/ G. BOUVIER
Système d'acquisition et de reconnaissance de caractères alphanumériques .
Thèse 3ème cycle, ENSER-INP Grenoble, 1977

/4.7/ A. KINDER
Procédé de reconnaissance syntaxique des caractères alphanumériques manuscrits .
Thèse de Docteur-Ingénieur, ENSER-INP Grenoble 1981.

/5.1/ S. PELEG
Ambiguity Reduction in Handwriting with Ambiguous Segmentation and Uncertain Interpretation .
CGIP, nr 10, 1979, pp. 235-245.

/5.2/ M. LENNING, P. MERMELSTEIN
Entraînement Lexical Semi-automatique d'un Système de Reconnaissance à Base Syllabique .
Bell Nothern Research, 3, Place du Commerce, Ile des Soeurs, Québec, Canada H3E 1H6.

dernière page de la thèse

AUTORISATION DE SOUTENANCE

VU les dispositions de l'article 3 de l'arrêté du 16 avril 1974,

VU les rapports de présentation de M. A. CHEHIKIAN
M. Ph.COUEIGNOUX

M. Nicolas TRIPON

est autorisé à présenter une thèse en soutenance pour l'obtention du diplôme de DOCTEUR-INGENIEUR, spécialité Systèmes et Réseaux Informatiques

Fait à Saint-Etienne, le 21 Septembre 1981

Le Président de l'INPG,

D. BLOCH
Président
de l'Institut National Polytechnique
de Grenoble

Le Directeur de l'EMSE,

M. MERMET
INGÉNIEUR EN CHEF DES MINES
DIRECTEUR
ECOLE NATIONALE SUPÉRIEURE DES
MINES DE SAINT-ÉTIENNE