

وزارة التربية الوطنية
MINISTRE DE L'EDUCATION NATIONALE

ECOLE NATIONALE POLYTECHNIQUE

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE - المكتبة
Ecole Nationale Polytechnique

DEPARTEMENT d' ELECTRONIQUE

PROJET DE FIN D'ETUDES

SUJET

Etude De La Methode structurelle
Pour La Reconnaissance Des Formes
Appliquée Aux Caractères Arabes
Manuscrits.

Proposé par :

M^{ME} L. HAMAMI

Etudié par :

M^{MR} N. BENAIFA

Dirigé par

M^{ME} L. HAMAMI

PROMOTION

JUILLET 93

PROJET DE FIN D'ETUDE

PRESENTE A

l'ECOLE NATIONALE POLYTECHNIQUE

POUR OBTENIR LE TITRE :

de **INGENIEUR D'ETAT**

EN ELECTRONIQUE

PAR

NACER BENAIFA



**Etude De La Méthode Structurelle
_ Structure De Chaîne _ Pour La Reconnaissance Des Formes Appliquée Aux Caractères Arabes Manuscrits**



JUILLET 93

S O M M A I R E

CHABITRE 1 :

GENERALITES

| | |
|--|---|
| <i>I.1 introduction</i> | 1 |
| <i>I.2 processus de reconnaissance des formes</i> | 2 |
| <i>I.3 les caractères manuscrits : définitions et caractéristiques</i> | 4 |
| <i>I.4 propriétés des caractères arabes</i> | 5 |
| <i>I.5 environnement matériel et logiciel utilisé</i> | 6 |
| <i>I.6 champs d'application de la reconnaissance de forme</i> | 7 |

CHAPITRE 2 :

ACQUISITION

| | |
|---|----|
| <i>II.1 Introduction</i> | 11 |
| <i>II.2 Techniques d'acquisition</i> | 11 |
| <i>II.3 Acquisition des images</i> | 13 |
| <i>II.4 Système d'acquisition utilisé</i> | 17 |

CHAPITRE 3 :

PRETRAITEMENT

| | | |
|---------|--|----|
| III.1 | Traitement d'images : définitions et pourquoi de la classe | 19 |
| III.2 | Différents procédés technologiques de traitement d'images | 20 |
| III.3 | Techniques de prétraitement | 23 |
| III.3.1 | Suppression de bruit : | |
| a) | Le lissage | 24 |
| b) | Le filtrage | 26 |
| c) | Augmentation du pas de quantification | 28 |
| d) | Seuillage | 31 |
| III.3.2 | Techniques de normalisation de la luminosité | 33 |
| III.3.3 | Techniques de squelettisation | 35 |
| III.4 | Conclusion | 46 |

CHAPITRE 4 :

ANALYSE ET EXTRACTION DE PARAMETRES :

| | | |
|--------|---|----|
| IV.1 | Différentes approches de représentation | |
| IV.1.1 | Approche globale | 48 |
| IV.1.2 | Approche structurelle | 49 |
| IV.2 | Techniques de représentation | 50 |
| IV.3 | Développement sur la structure de chaînes | 59 |
| IV.4 | Stratégie de codage adopté | 63 |

CHAPITRE 5 :

APPRENTISSAGE ET DECISION :

| | |
|---|----|
| V.1 Introduction | 68 |
| V.2 L'apprentissage : | |
| V.2.1. Manuel | 69 |
| V.2.2. Automatique | 69 |
| V.2.3. Continu | 70 |
| V.3 Techniques d'apprentissages adaptés | 70 |
| V.4 La décision : | |
| V.4.1. Approche statistique..... | 72 |
| V.4.2. Approche structurale | 72 |
| V.5 Développement sur la méthode de comparaison de chaînes .. | 74 |
| V.6 Distance entre chaînes (algorithme de WIGNER & FISHER) .. | 77 |
| V.7 Les procédés de reconnaissances adaptés | 82 |
| | |
| CONCLUSION GENERAL | 87 |
| | |
| ANNEXE I . | |
| PRINCIPALES ORGANIGRAMMES REALISES | |
| | |
| ANNEXE II . | |
| BIBLIOGRAPHIE | |

DEDICACES

أهدي هذه الدراسة المتواضعة

إلى والدي العزيزين:-

« لخصر »

REMERCIEMENTS



Je ne saurai présenter la présente étude sans exprimé mes remerciements : -

- Avant tout à madame HAMAMI qui a bien voulu me faire honneur en acceptant de me diriger le long de cette étude.
- A monsieur A.AZIZI pour son soutien, ses conseils et ses encouragements. Pour tout et pour l'amitié qu'il m'a témoigné je lui suis profondément reconnaissant.
- A monsieur H.Med.LAID pour son aide bibliographique, sans cette dernière cette étude serait totalement différente.
- A messieurs A.ABDELKRIM et KRACHID et à tous ceux qui ont participé de près ou de loin dans l'élaboration de cette étude.

B. NACER

CHAPITRE I

GENERALITES

GENERALITEES

I.1 INTRODUCTION :

La reconnaissance des formes peut être définie comme l'ensemble des techniques informatiques de représentation et de décision permettant aux machines de simuler un comportement *sensible*; cette discipline permet par exemple de donner une capacité de lecture (vision) ou d'écoute (audition) à certaines machines. Fondamentalement il s'agit d'une part de doter l'ordinateur de capteurs (microphone suivi de CAN, scanner, caméra vidéo, tablette graphique, etc...); et d'autre part de le programmer de sorte qu'il soit capable d'interpréter les sensations reçues à travers ces capteurs.

Le mot *interprétation* signifie dans ce qui nous occupe une simple catégorisation du phénomène perçu, c'est à dire son affectation à une famille de phénomènes ressemblants; Autrement dit, son identification par comparaison avec des phénomènes analogues conservés en mémoire.

La reconnaissance des formes a donc pour objet général d'une part de capter et de d'écrire en mémoire des formes, c'est à dire les manifestations de l'univers extérieur auquel la machine a été rendue sensible et d'autre part, de prendre sur la représentation mémoire ainsi obtenue une décision d'identification par référence à un ensemble d'apprentissage décrit dans une représentation analogue.

GENERALITEES

La partie essentielle est évidemment le choix de cette représentation mémoire, celle-ci doit être à la fois assez informative pour permettre une bonne précision à l'identification et assez condensée pour éviter une inutile redondance qui se traduit par un temps de calcul exagéré au moment de la décision.

I.2 PROCESSUS DE RECONNAISSANCE :

La démarche classique suivie en reconnaissance de formes consiste à opérer selon l'organisation générale schématisée à la figure (I.1). Il est important de mentionner que ce schéma n'est pas dans la pratique purement linéaire. Des interactions peuvent apparaître entre les différents niveaux, pour d'éventuels retours en arrière.

Nous allons d'abord rappeler le rôle de chaque étape dans l'élaboration du processus complet de la reconnaissance, puis nous reviendrons dans les chapitres qui suivent plus en détail sur la fonction de chacune d'entre elles.

I.2.1 MONDE PHYSIQUE :

La chaîne part du monde physique qui est un espace analogique de dimension infinie appelée espace des formes (F). Les objets, dans cette espace, sont décrits de façons indifférentes avec une multitude de propriétés dont-il serait difficile de tenir compte de chacune à la reconnaissance des formes. La loi de passage au monde discret nécessite forcément une sélection et par conséquent une certaine simplification.

I.2.2 L'ACQUISITION :

C'est une opération de conversion numérique du monde physique continu vers un monde numérique discret. Ce dernier, appelé aussi

SCHEMA GENERAL D'UN SYSTEME DE RECONNAISSANCE

DES FORMES

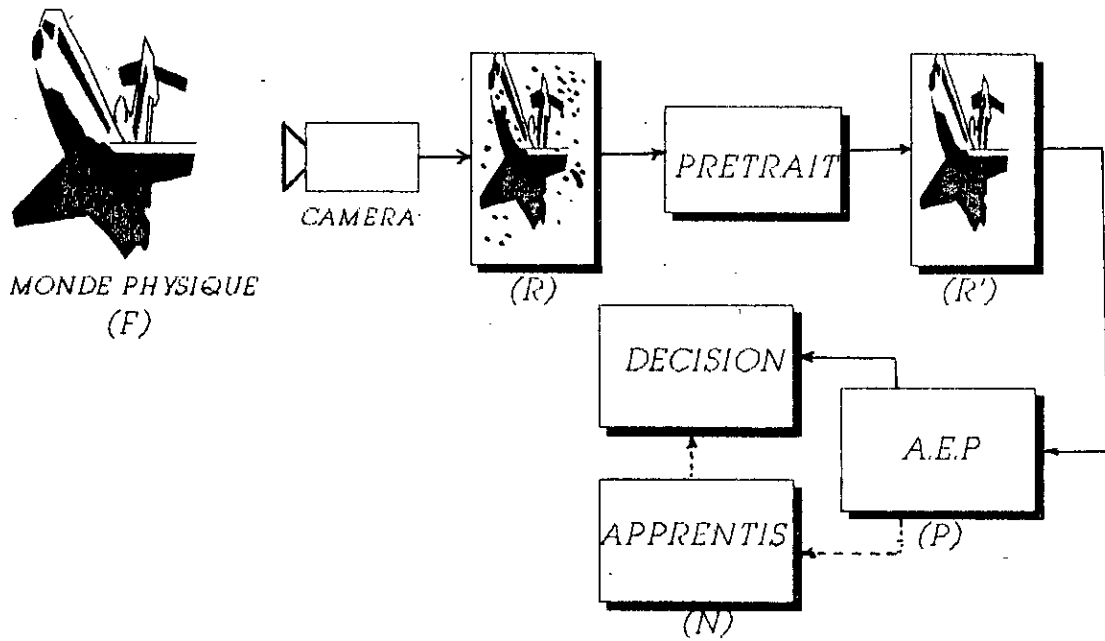


Fig I - 1

GENERALITEES

espace de représentation, à une dimension encore trop importante malgré qu'elle soit finie. La dimension (r) de cet espace est choisie volontairement grande de manière à pouvoir disposer d'un maximum d'informations sur la forme et à pouvoir y sélectionner des sous ensembles pour de multiples usages.

I.2.3 LE PRETRAITEMENT :

L'étape de prétraitement consiste à sélectionner dans l'espace de représentation l'information nécessaire à l'application. Cette sélection passe souvent par l'élimination du bruit due aux conditions d'acquisition, par la normalisation des données ainsi que par la suppression de la redondance. Le nouvel espace de représentation (R') à une dimension r' très inférieur devant r , mais demeure un espace de grande dimension et contient des informations encore assez primitives.

I.2.4 L'ANALYSE :

Lors de l'étape d'analyse les techniques de la reconnaissance de forme calculent un certain nombre de caractéristiques ou paramètres. Ces paramètres correspondant à des mesures de nature géométrique, topologique ou statistique et servent comme seules données représentant la forme; ils sont généralement limités en nombre, ainsi l'espace obtenu est l'espace des paramètres (P) de dimension p très petites par rapport à r' .

I.2.5 L'APPRENTISSAGE :

L'apprentissage ou modélisation est une étape clé dans la chaîne de reconnaissance. Son rôle est d'éclairer la décision à l'aide de connaissance a priori sur les formes. A partir de critères spécifiques aux formes, l'apprentissage tente de définir des modèles de référence ou de caractériser des classes de décision. L'espace (N) ainsi obtenu s'appelle l'espace des noms, puisqu'il contient les noms des modèles.

GENERALITES

ou classes qu'il a formé. Sa dimension n correspond globalement au nombre de modèles ou classes existantes.

1.2.6 LA DÉCISION :

La décision ou classement est l'étape de reconnaissance proprement dite. Son rôle est d'identifier la forme test à partir de l'apprentissage réalisé. La méthode de décision est souvent " exhibée " par l'apprentissage, ce qui veut dire que les critères utilisés pour la comparaison, sont les mêmes que ceux utilisés pour l'apprentissage. En effet, il est évident que le choix de critères différents pour la décision n'assurant pas l'uniformité de description, ne peut pas conduire avec certitude à un résultat cohérent.

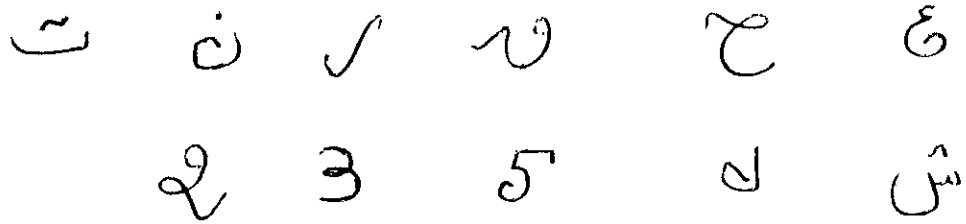
Parmi les techniques utilisées, certaines sont fondées sur la notion de proximité et nécessitent de calculer une distance ou une probabilité de ressemblance avec les modèles définis. D'autres sont fondées sur l'analyse de la structure de la forme et essaient plutôt de vérifier une certaine cohérence dans les relations entre les différentes sous-structures. La réponse de décision peut être, selon le cas, le nom de la forme en cas de bonne reconnaissance, plusieurs noms en cas d'ambiguïté ou bien le rejet de la forme en cas d'incompatibilité de description avec les formes de référence. Dans les deux premiers cas, la réponse peut être accompagnée d'un taux ou score de confiance.

1.3 LES CARACTERES MANUSCRITS , DÉFINITIONS ET CARACTÉRISTIQUES :

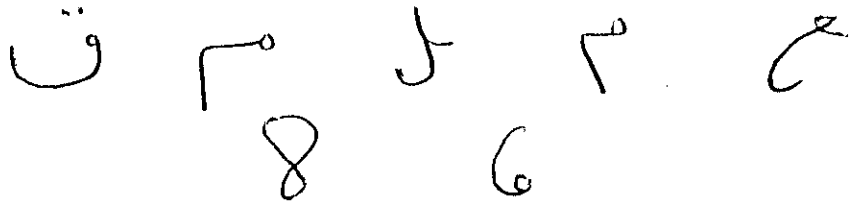
Tout caractère écrit à la main est, par définition, un manuscrit. Sa forme est souvent variable, traduisant le style d'écriture, l'état d'âme et la personnalité du scripteur, ce qui le rend difficile à caractériser. Parmi les variations qui l'entachent nous trouvons :

- La distorsion du contour qui conduit à la formation des boucles et arrondis :

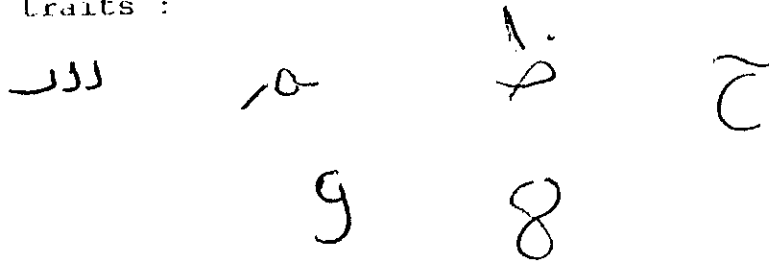
GENERALITEES



- L'inclinaison et la dissymétrie engendrant des déformations de la forme :



- Le mauvais raccordement des tracés engendrant des interruptions brutales des traits :



I.4 PROPRIETES DES CARACTERES ARABES :

L'alphabet arabe est constitué de 28 caractères, chaque caractère peut apparaître sous forme isolée ou fait partie d'un mot. La première situation a lieu dans quelques applications tels que le cas des courriers où les caractères incluent les lettres et les chiffres, la reconnaissance des équations mathématiques où les caractères incluent les lettres, les chiffres et les symboles mathématiques, dans cette première situation chaque caractère a une forme unique. L'autre situation a lieu avec les textes, la forme de chaque caractère dépend de sa position dans le mot (début, milieu, fin). En tout on a les formes de la figure (I.2).

آء ا ا ا ا ا ا ا

ئ ئ ئ ئ ئ ئ ئ

ب ب ب ب ب ب ب

ت ت ت ت ت ت ت

ث ث ث ث ث ث ث

ج ج ج ج ج ج ج

ح ح ح ح ح ح ح

خ خ خ خ خ خ خ

د د

ذ ذ

ر ر ر ر ر ر ر

ظ ظ ظ ظ ظ ظ ظ

ع ع ع ع ع ع ع

غ غ غ غ غ غ غ

ك ك ك ك ك ك ك

ل ل ل ل ل ل ل

م م م م م م م

و و و و و و و

ه ه ه ه ه ه ه

ف ف ف ف ف ف ف

س س س س س س س

ي ي ي ي ي ي ي

لا لا لا لا لا لا لا

fig - 2 -

GENERALITEES

Il est à noter que les caractères arabes sont constitués principalement de courbures, parmi eux qui contiennent plus qu'une seule composante connectée. En effet 16 parmi les 28 caractères contiennent une, deux, ou même trois points en dessous, en dessus ou sur la ligne d'écriture d'autre contient en plus du corps principal des barres horizontales ou verticales ou même des zigzag (Fig I.3 et I.4).

4 ~ : " .

Fig I.3

4 ✓ 4

Fig I.4

I.5 ENVIRONNEMENT MATÉRIEL ET LOGICIEL UTILISÉ :

La réalisation de nos programmes a bénéficié de l'environnement suivant :

I.5.1 LANGUAGE DE PROGRAMMATION :

Les programmes décrits ont été développés à l'aide du logiciel turbo pascal de BORLAND (version 5.5 /6.0)qui intègre les outils et

GENERALITEES

les fonctions suivantes :

- Un éditeur de texte;
- Un compilateur séparé;
- Possibilité de créer des unités s'appelant mutuellement.
- L'exécution pas à pas assurée par le débogueur intégré.
- Autorisation de la récursivité et la structure dynamique.

Ce langage s'est avéré d'une part puissant en temps de calcul, compilation et exécution; Et offre d'autre part un graphisme de qualité.

I.5.2 SUPPORT MATÉRIEL :

On a fait fonctionner le logiciel TURBO PASCAL sur un micro ordinateur SIEMENS. Ce dernier nous a servi de support pour la réalisation de nos programmes.

La configuration utilisée est comme suit :

- Un microprocesseur 80 286, et une horloge de 12 MHz.
- Une mémoire central RAM de 1 MO.
- Une unité de disquette (3" 1/2), et un disque dur de 42 MO.
- Un écran graphique équipé d'une carte (VGAHI) ayant une résolution de (640 x 480) de 16 couleurs.
- Un système d'exploitation MS-DOS version (5:00).
- Une imprimante graphique Epson FX 80 / FX 100.

I.6 CHAMPS D'APPLICATIONS :

Les applications auquel cette discipline est classiquement attachée sont les suivantes:

I.6.1 RECONNAISSANCE DE LA PAROLE :

Elle est liée à plusieurs disciplines comme :

- La reconnaissance de mots.
- La reconnaissance de phrases.
- La reconnaissance du locuteur.
- Le décodage acoustico-phonétique.
- La compréhension du dialogue oral homme-machine.

Parmi les applications les plus courantes, nous pouvons citer la commande vocale, la dictée automatique, la traduction en temps réel de langues étrangères et la rééducation des mal entendant.

I.6.2 RECONNAISSANCE DE L'ECRITURE :

Elle est attachée à :

- Reconnaissance des caractères imprimés et manuscrits.
- Reconnaissance des mots et des phrases.
- Reconnaissance du scripteur.
- Reconnaissance de documents.

Parmi les domaines d'applications de la reconnaissance de l'écriture signalant :

I.6.2.a) L'environnement bancaire :

Elle permet ici la saisie automatique des chèques pour la vérification de la somme d'argent donnée en chiffres et en toutes lettres, et la vérification de signatures. Actuellement les banques

GENERALITEES

applications utilisées avec succès sont celles qui font le relevé des chèques. Le jeu de caractères employé est assez réduit avec une grande tolérance à la mauvaise impression est acceptée.

I.6.2.b) La bureautique :

L'utilisation des lecteurs en bureautique est destinée essentiellement à la saisie et à l'archivage de documents.

Les lecteurs manipulent des documents de format A4 et les gèrent dans des environnements de traitements de textes. Ces lecteurs reconnaissent plusieurs types de fontes.

I.6.2.c) La poste :

Les lecteurs sont utilisés pour faire le tri automatique des courriers. Ils lisent l'adresse, localisent et identifient le code postal.

I.6.2.d) La lecture pour les non-voyants :

Les lecteurs peuvent aider les non-voyants à la compréhension des textes. Ils sont équipés d'un haut parleur avec un synthétiseur de parole. Le coût de tels lecteurs ne permet pas l'achat à titre individuel. Ils sont limités à l'utilisation dans des associations de non-voyants.

I.6.3 RECONNAISSANCE DES IMAGES - vision - :

La vision comprend le traitement, l'analyse, et l'interprétation des images. Les applications sont très nombreuses, on peut citer la sûreté avec la reconnaissance des empreintes digitales, la médecine avec l'analyse d'images de radiographies ou d'échographies, l'industrie avec l'analyse des défauts dans les pièces ou l'identification

GENERALITEES

d'objets (tri, ou, localisation d'objets), la surveillance en processus robotiques, la géophysique avec l'analyse du sol en pétrographie, l'armée avec la surveillance, le guidage des cibles...etc.

CHAPITRE II

ACQUISITION

A C Q U I S I T I O N

II.1 INTRODUCTION:

L'opération d'acquisition consiste à transformer un ensemble de données analogiques en un ensemble de données numériques de manière à pouvoir les traiter par ordinateur. Cette transformation doit se faire de la manière la plus fidèle possible c'est à dire sans perte d'informations pertinentes et en conservant les propriétés essentielles de l'objet physique. En reconnaissance de la parole, par exemple, le signal obtenu doit reproduire à la fois les différences de timbre de la voix et sa mélodie. En vision on doit être capable de retrouver les mêmes dispositions des objets de la scène, les mêmes rapports de dimensions et de distances entre objets ainsi que la même nuance de couleurs.

De manière générale on exige de l'acquisition que la forme obtenue soit la plus intelligible possible.

II.2 TECHNIQUES D'ACQUISITION :

La transformation numérique des données physiques, appelée aussi transduction, se fait par l'intermédiaire d'un couple (Capteur, Digitaliseur), la fonction du capteur est d'assurer la transformation de grandeur physique (énergie lumineuse, acoustique) en un signal

ACQUISITION

électrique analogique adapté au système de traitement; ce qui ensuite convertie par le digitaliseur en données numériques. La fonction du digitaliseur est subdivisé en échantillonnage et quantification.

II.2.1 LE CAPTEUR :

Le choix du capteur est important pour l'acquisition des données. Ces caractéristiques doivent être connues de l'utilisateur afin que ce dernier en tienne éventuellement compte lors du traitement. Les capteurs utilisés sont fonction du domaine traité. Pour les signaux temporels du type parole, il s'agit en général d'un microphone. Pour les signaux du type image, le choix est plus large et dépend du résultat souhaité. Nous présenterons les différentes possibilités dans la suite du chapitre.

II.2.2 LE DIGITALISEUR :

Afin de pouvoir traiter les données analogiques provenant d'un capteur, par le calculateur, il est indispensable de passer par un digitaliseur (numérisateur) qui effectue l'échantillonnage et la quantification.

II.2.2.a) L'ECHANTILLONAGE :

L'échantillonnage consiste en un prélèvement temporel ou spatial - suivant le domaine d'application - à des intervalles réguliers d'échantillons sur la fonction continue.

II.2.2.B) LA QUANTIFICATION :

Pour finir la digitalisation, les échantillons doivent être quantifiés, ceci est réalisé en divisant la gamme dynamique de l'amplitude des échantillons en un nombre fini d'intervalles; Le choix du nombre d'intervalle dépend de deux facteurs principaux, le premier

est subjectif (perception des organes de sensibilité de l'homme), le second est le support physique sur lequel on désire stocker l'image (facteur objectif).

II.3 ACQUISITION DES IMAGES :

II.3.1 IMAGE ANALOGIQUE :

A l'échelle de l'observation visuelle une image est une représentation bidimensionnels d'objets de nature diverse (pièces industrielles, vues aériennes, images médicales, scènes naturelles... etc), elle contient en chaque point l'intensité lumineuse perçue par une caméra (ou toute autre système de mesure) en ce point. Elle rend compte de la variation du contraste dans la scène. L'intensité lumineuse varie entre le blanc (beaucoup de réflexion de lumière due à un faible contraste) et le noir (réflexion nulle de la lumière due à la présence d'un fort contraste). C'est donc une fonction de brillance continue définie sur un domaine borné.

Sous cette forme l'image est inexploitable par la machine. Il est donc nécessaire de la numériser.

II.3.2 IMAGE DIGITALE :

On appelle image digitale, une matrice à deux dimensions dont les éléments représentent les niveaux de gris d'un faisceau de lumière monochromatique réfléchi par l'objet. Cette matrice est obtenue après une opération de numérisation des données à la sortie du capteur.

II.3.3 DIFFÉRENTS TYPES DE SYSTEMES D'ACQUISITION :

En résumé, un système d'acquisition peut être caractérisé par :-

- La présence ou non d'un dispositif optique complexe.
- Type du capteur (résolution incluse).
- Type du dispositif de balayage.

La partie échantillonnage, et quantification et mise en forme et éventuellement stockage est standard.

Le choix du système d'acquisition dépend essentiellement de la qualité d'image voulue et le temps d'analyse.

II.3.3.a) TABLETTE GRAPHIQUE :

L'image obtenue est un ensemble de points séparés par des levés d'un crayon. En effet, à partir de contact avec la surface de la tablette, un crayon spécial émet les coordonnées des points qu'il décrit à une fréquence constante. La rupture du contact interrompt la transmission d'un caractère donné. Ainsi, l'enregistrement sur la tablette produit un ensemble de coordonnées séparés par des lignes indiquant les moments où le crayon a cessé de toucher la tablette. Ce type d'enregistrement offre l'avantage d'une fidèle reproduction de la direction des lignes tracées et l'ordre dans lequel les parties du diagramme ont été dessinées.

II.3.3.b) LE MICRO-DENSIMETRE :

Le micro-densimètre est composé d'une platine porte objet qui se déplace mécaniquement, d'une source lumineuse, d'un système de focalisation de la lumière sur les points de l'image qui défilent et d'un photomultiplicateur qui recueille la lumière (flux de photons)

ACQUISITION

sur la photocathode, celle-ci délivre un courant qui est proportionnelle aux flux; ce courant est amplifié par un système de dynodes en cascade et en recueille également un courant d'anode proportionnelle au flux reçu; ce courant est moyenné par un circuit intégrateur aux borne duquel on prélève une tension qui constitue le signal de sortie.

Tous comme le flying spot, ce système est relativement lent mais précis, l'acquisition se fait d'une manière continue, point par point, c'est au système de mise en forme de l'image de retrouver les différentes lignes et colonnes de l'image.

II.3.3.c) LES CAMERAS :

LES CAMERAS A TUBES :

Elles sont composées d'une cible photoconductrice qu'explore un faisceau électronique. L'exploration s'effectue ligne par ligne, à chaque ligne correspond, en sortie, un signal analogique proportionnel à l'intensité.

LES CAMERAS C.C.D : (charged coupled device)

Elles sont constituées par un assemblage de photodiodes, chacune d'entre elles délivre une intensité proportionnelle à un point de l'image appelé pixel. Contrairement aux caméras à tubes qui fournissent un signal continu qu'il faut échantillonner spatialement pour isoler chaque pixel, les caméras C.C.D. fournissent directement l'intensité pour chaque point de l'image puisqu'à chaque point correspond une photodiode. On distingue deux familles de caméras C.C.D. :-

-Les caméras matricielles : Les photodiodes sont assemblées en une matrice de N lignes et M colonnes; cette matrice peut être carré, les résolutions courantes sont alors 100x100, ou 256x256 ou

bien rectangulaire 300x512.

-Les caméras unilignes : Comme leurs noms l'indique, les photodiodes y sont agencées suivant une seule ligne; elles ont l'avantage de posséder une résolution supérieure (2048 pixels); l'obtention d'une image nécessite cependant le déplacement de la caméra afin d'acquérir plusieurs lignes pour obtenir une image bidimensionnelle.

II.3.3.d) LE SCANNER :

L'image obtenue à partir du scanner est une image à plusieurs niveaux de gris. On distingue deux types de scanners : les scanners à miroir et les scanners de table qu'ils soient à défilement ou à plat. Le système comprend toujours une source de lumière constante fournie par des tubes luminescents; Plus la zone de document ainsi éclairé est moins sombre plus elle réfléchit la lumière. Par l'intermédiaire d'un système optique des capteurs (phototransistors de très faible diamètre ou cellule C.C.D.) recueillent la lumière réfléchi et donnent une tension proportionnelle à celle-ci.

L'image obtenue en sortie de ces différents capteurs est, soit directement, soit à l'aide d'un digitaliseur, numérisée et devient donc une image formée de pixels à plusieurs niveaux de gris.

II.3.3.e) LE FLYING SPOT :

Le flying spot est composé d'un spot mobile (faisceau) qui se déplace sur l'image. La formation du faisceau est assurée par des électrodes qui constituent avec la cathode émettrice et la grille de contrôle le canon à électrons. Un système magnétique réalise la focalisation et la déflexion du faisceau du spot. Chaque position du spot correspond à un point (x , y) dans l'image. L'amplification et la numérisation du faisceau représentent le résultat de la mesure de la lumière transmise par le point (x , y). Ce mode de balayage permet

ACQUISITION

d'obtenir une grande précision. Le balayage peut être séquentiel ou aléatoire. Vu la complexité du système de guidage du spot, les applications de ce système sont généralement limitées à l'exploration d'images de petits formats.

II.3.3.f) LE DISQUE DE NIPKOV :

Le disque de nipkov est un disque percé de trous. Ces trous sont disposés en cercles concentriques tels qu'entre deux trous consécutifs, la distance selon un rayon soit constante. Pendant la rotation du disque, chaque trou parcourt une partie du diaphragme rectangulaire. Le parcours circulaire de trou est assimilé à une bande rectiligne du diaphragme. Ainsi, lorsque la rotation du disque est synchronisée avec le défilement du convoyeur du support de l'image, le disque permet l'échantillonnage d'une colonne de l'image correspondant à une bande rectiligne du diaphragme.

II.4 SYSTEME D'ACQUISITION UTILISE :

Vue l'absence dans notre matériel utilisé d'un système d'acquisition on (voir ch-1) il est donc impossible de travailler sur des images réelles, pour résoudre ce problème et comme on ne dispose que d'un clavier, on a été amené à simuler une table à digitaliser, il est à noter que cette simulation n'a pas résolu tout le problème car les étapes de filtrage ne peuvent être appliquées pour la simple raison que l'acquisition d'un caractère point par point est une tâche extrêmement difficile surtout si on essaye de travailler à plusieurs niveaux de gris, (la seule manière d'avoir une simulation complète d'un bruit d'acquisition); ceci, bien sûr, est loin d'être un inconvénient vu que les étapes principales qui sont la normalisation de la taille et la squelettisation sont appliquées très normalement. Ainsi, dans notre système la fonction d'acquisition se réduit à une simple manipulation de quelques touches du clavier, dont le mode d'emploi peut être résumé ainsi :

ACQUISITION

-Après que l'opérateur_ayant choisi la rubrique " ACQUISITION " du menu principal; Un cadre délimitant la zone d'acquisition s'affiche sur l'écran;

-L'opérateur est ensuite invité a choisir une configuration parmi deux pour le curseur graphique.

-Le déplacement du curseur graphique dans la zone d'acquisition est assuré par les touches de directions.

-Et suivant que l'opérateur est en mode affichage ou en mode effacement (la permutation entre ces deux modes est assurée par la barre d'espacement) le déplacement du curseur est accompagnée avec l'allumage ou l'extinction des points formant sa trajectoire.

REMARQUE IMPORTANTE :

Pour saisir un caractère de la manière la plus fidèle possible, il a fallu d'abord l'écrire sur un papier transparent, pour le juxtaposé sur le cadre précédent et le reproduire point par point sur l'écran en faisant parcourir le curseur graphique le long de son contour.

CHAPITRE III

PRETRAITEMENT

P R E T R A I T E M E N T

III.1 TRAITEMENT D'IMAGE DÉFINITION ET POURQUOI DE LA CHOSE :

On peut dire que l'on traite une image dès lors que l'on extrait une information de cette image. Ceci étant dit, une question se pose pourquoi cherche-t-on à traiter une image ? :

Les raisons sont nombreux :-

1- On peut chercher à améliorer sa qualité, qu'il s'agit d'une qualité subjective (on veut, par exemple, la rendre plus agréable à l'oeil) ou d'une qualité objective, donc mesurable : on veut améliorer son contraste, accroître la perception de certains détails ou contours, faire ressortir plus nettement certaines zones ou certaines formes, diminuer les bruits ou les parasites d'origines diverses qui perturbent l'information utile,..etc. Il s'agit donc de techniques d'enhancement qui font appel à des procédés variés (modification de l'histogramme, filtrage linéaire, filtrage non linéaire) créant des lissages, accentuations des contours, rehaussements, ...etc.

2- On peut chercher à sa restauration : c'est à dire à retrouver une image idéale qui a été dégradée par divers processus : ce qui revient à corriger les distorsions géométriques ou photométriques d'un capteur, diminuer les fluctuations dues à des phénomènes de turbulence

PRETRAITEMENT

es atmosphérique, corriger le flou dû aux bougies de l'appareil de prise de vues, ...etc. Ici divers types de filtrage linéaire ou non sont utilisés.

3- On peut chercher à détecter la présence de certaines formes, certains contours ou certaines textures de modèle connu, sans vouloir préserver les autres informations contenues dans l'image. Il s'agit là du problème de détection classique en théorie de signal: on veut extraire un signal de gabarit connu noyé dans un "bruit" constitué par tout ce qui n'est pas signal utile.

4- Un aspect important de traitement d'image concerne l'énorme quantité d'informations qu'il faut manipuler quand on transmet une image et a fortiori plusieurs images. Il est intéressant de chercher à réduire cette quantité d'informations, à la "compresser" afin de gagner en vitesse de transmission, en encombrement des organes utilisés, en capacité de stockage, ...etc tout en dégradant le moins possible les images considérées.

5- Le traitement d'images concerne aussi l'analyse et la compréhension de l'image dont un des buts est de doter les machines de capacités visuelles (cas qui nous occupe).

Dans la partie suivante nous décrirons tout d'abord les différents procédés technologiques utilisés en traitement d'image, puis nous détaillons les différents algorithmes implémentés dans notre système.

III.2 DIFFÉRENTS TECHNIQUES DE TRAITEMENT D'IMAGE :

Suivent la nature du support de l'image plusieurs méthodes de traitement d'images ont été mise au point :-

III.2.1. PROCÉDES PHOTOGRAPHIQUES:

L'image est fixée à l'état latent sur un support physique de film

PRETRAITEMENT

photosensible : Par action sur les conditions de développement, par utilisation de bains spéciaux, de filtres chromatiques; il est possible de corriger certaines dégradations simples: surexposition, sous-exposition, affaiblissement ou renforcement d'une dominante.

Les techniques sont fastidieuses, souvent empiriques et très limitées dans leurs applications.

III.2.2 PROCEDES ANALOGIQUES :

L'image est représentée par un signal analogique comme c'est le cas, par exemple, du signal vidéo fourni par une caméra de TV. Des opérations électroniques simples peuvent être directement effectuées sur ce signal, tel que :

- Intégration.
- Dérivation.
- Amplification.
- Ou toute autre opération fonctionnelle.

L'avantage de ce procédé est la possibilité de traitement de l'image en temps réel.

Les inconvénients sont nombreux :

- Impossibilité de modifier la fonction d'un appareillage.
- Effet unidirectionnel de traitement dû au balayage en ligne de l'image

III.2.3 PROCEDES OPTIQUES :

L'image est enregistrée sur un faisceau lumineux après avoir fait traverser cette onde lumineuse dans un négatif photographique ou un diapositive.

La technique de traitement la plus utilisée est le filtrage spatial. fondé sur la théorie de la diffraction de la lumière, il est possible

PRETRAITEMENT

d'obtenir de façon immédiate une grande variété de traitement telle que :

- Réhaussement d'images floues.
- Mise en évidence des contours d'objets.
- ...etc.

Parmi les avantages de ces procédures nous pouvons citer :

- Le traitement se fait en temps réel.
- Pas de contraintes de mémorisation, les supports photographiques constituent une grande capacité de stockage.
- Les résultats sont directement interprétables par un observateur.

Les inconvénients suivants limitent cependant les applications de ces procédés :

- les filtres sont délicats à réaliser.
- Le positionnement spatial des filtres est difficile à mettre au point.
- Impossibilité d'extraction de paramètres significatifs de l'image.

III.2.4 PROCÉDES NUMÉRIQUE :

L'image digitale représentée par une matrice contenant MXN pixels est traitée par un calculateur numérique. Le traitement d'image numérique consiste à considérer une image comme un ensemble de nombres et à effectuer sur ces nombres des calculs numériques.

Les méthodes numériques connaissent un développement croissant dû à plusieurs facteurs :-

1. Développement des moyens informatiques :-

- Augmentation de la taille des mémoires centrales des calculateurs.

PRETRAITEMENT

- Augmentation de la vitesse de calcul grâce à des nouveaux processeurs rapides et des mémoires à temps d'accès court.

2. Développement de logiciels et algorithmes rapides.
3. Souplesse d'utilisation due à la possibilité de traitements itératifs et interactifs sur les images.

En tenant compte des avantages et inconvénients des différents procédés présentés, le choix de la méthode numérique dans la reconnaissance de forme par les chercheurs s'avère justifier.

III.3 TECHNIQUES DE PRETRAITEMENT :

A cause de la forte granularité de l'échantillonnage et les divers problèmes d'éclairage et de saisie, l'image du caractère peut subir des manquements ou des empâtements. Il convient de corriger si possible ces problèmes avant toute étape d'analyse. Par ailleurs, il n'est pas toujours utile d'utiliser tous les points de l'image du caractère pour extraire les propriétés caractéristiques. Une étape de réduction élimine les points redondants. Nous allons voir dans la suite quelques techniques de prétraitement relatives à la suppression du bruit, à la normalisation de la taille, et à la squelettisation.

III.3.1 SUPPRESSION DU BRUIT :

Le but de cette étape est de débarrasser les données de l'image du bruit de l'acquisition et de ne garder que l'information significatif de la forme.

III.3.1.a) TECHNIQUES DE LISSAGE :

L'image du caractère peut être entachée de bruit due aux artefacts de l'acquisition et souvent à la qualité du document, conduisant soit à des absences des points (trous) soit à des empâtements ou des excro-

PRETRAITEMENT

issances et donc à une surcharge de points. Les techniques de lissage permettent de résoudre ces problèmes par des opérations locales appelées nettoyage et bouchage.

Nous utiliserons les notations suivantes pour la description des masques de transformation de voisinage : " 1 " pour un point de la forme, " 0 " pour un point du fond et " X " pour un point quelconque.

L'opération de nettoyage conduit à supprimer les petites taches et les excroissances de la forme. Elle est réalisée de différentes manières suivant le type du bruit à enlever :-

- pour éliminer des points isolés de la forme ou situées à l'extrémité des contours, on applique sur l'image le masque suivant dans les 8 (huit) directions :-

$$\begin{array}{ccc} 0 & 0 & 0 \\ X & 1 & 0 \\ 0 & 0 & 0 \end{array}$$

- Pour éliminer les points formant des angles droits ou des excroissances du contour on applique le masque suivant dans les 8 (huit) directions :-

$$\begin{array}{ccc} X & 0 & 0 \\ 1 & 1 & 0 \\ X & 0 & 0 \end{array}$$

- Pour éliminer les points formant des coins, on applique le masque suivant dans les 4 (quatre) directions principales :-

$$\begin{array}{ccc} 0 & 0 & 0 \\ X & 1 & 0 \\ 0 & X & 0 \end{array}$$

PRETRAITEMENT

Pour le bouchage, il s'agit d'égaliser le contour et de boucher les trous interne à la forme du caractère en lui ajoutant des points. On distingue :-

- Le bouchage des trous isolés: si le voisinage d'un point du fond correspond au masque suivant alors ce point est mis à 1:-

$$\begin{array}{ccc} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{array}$$

- Pour la correction des irrégularités des contours dues à effritement de la forme, On applique sur l'image le masque suivant dans les 8 (huit) directions :-

$$\begin{array}{ccc} X & 1 & 1 \\ 0 & 0 & 1 \\ X & 1 & 1 \end{array}$$

UN MOT SUR L'EFFET DE BORD :

Avant d'aller plus loin dans les techniques de prétraitement, il est à noter que l'application d'un masque 3X3, par exemple, au point de l'image de coordonnées (0,0) pose un problème de fait que ce masque fait appel à 5 points qui n'appartiennent pas à l'image; et de manière générale, pour un filtre de taille $(n+1) \times (n+1)$, il y a tout autour de l'image un cadre d'épaisseur $n/2$ où le filtre n'est a priori pas applicable (fig III.1). Les différentes possibilités sont alors :

- Travailler uniquement avec la partie intérieure de l'image variant de $n/2$ à $p-n/2$; L'image résultante est alors de taille réduite (fig III.1.a).

- Réaliser un effet de miroir : on ajoute autour de

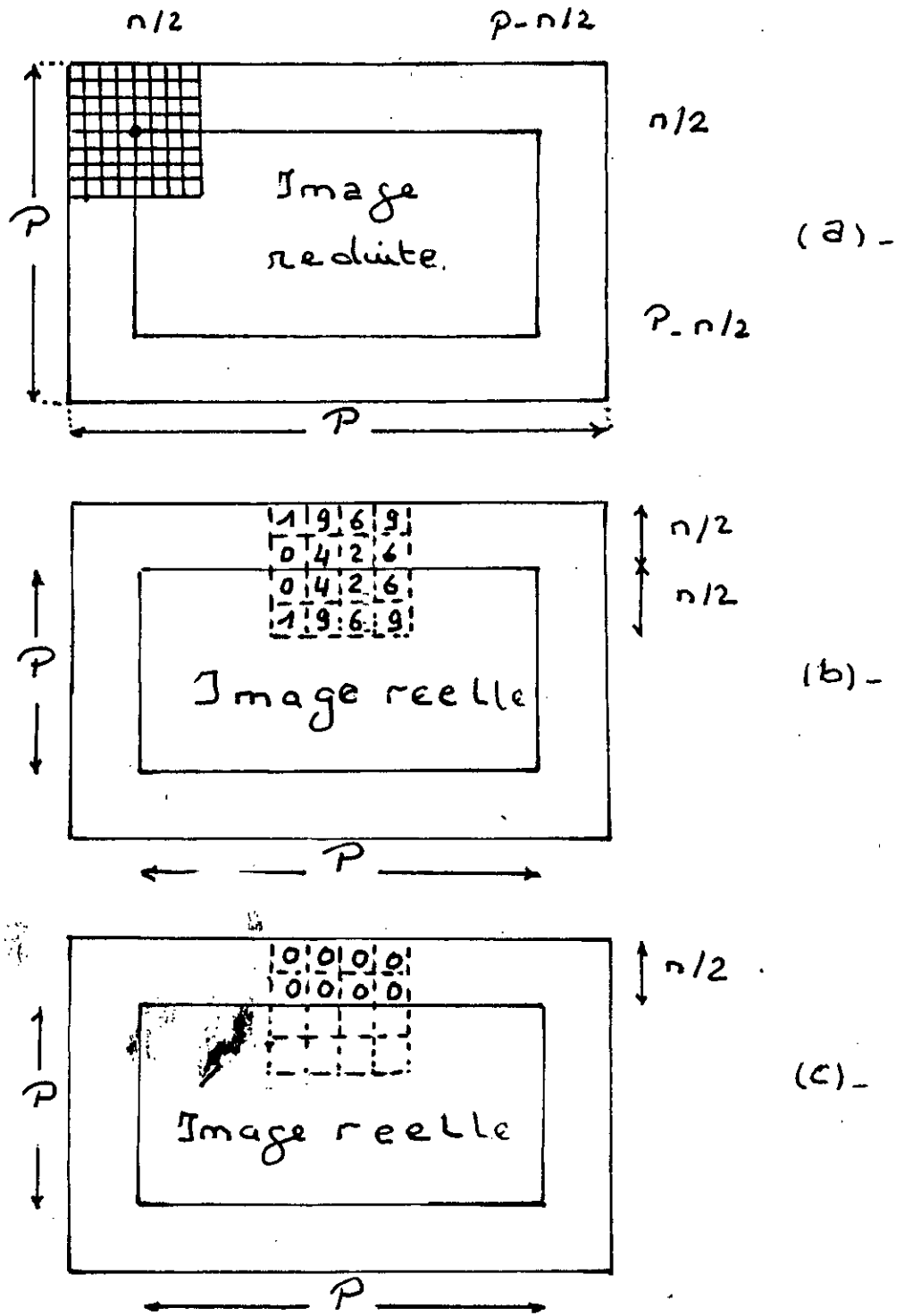


Fig III.1 Résolution des effets de bords
 (a) - Domaine d'utilisation du filtre, (b) - effet de miroir, (c) - Mise a zero de la couronne.

PRÉTRAITEMENT

l'image initial un cadre d'épaisseur $n/2$ dans lequel on recopie les $n/2$ lignes du bord de l'image comme illustrer sur la fig ; Dans ce cas la taille de l'image n'est pas touché (fig III.1.b).

-On reprend la méthode précédente mais on remplit le cadre de " 0 " (fig III.1.c).

III.3.1.b) TECHNIQUE DU FILTRAGE:

FILTRAGE PASSE BAS:

Le filtrage passe bas revient à faire un lissage sur les données de l'image et à pour intérêt d'éliminer une partie du bruit qui se trouve dans les hautes fréquences spatiales.

Comme conséquence les détails fins de l'image et les parties représentant les contours qui se trouvent dans les régions peu contrastées, seront éliminées, d'où l'apparition d'un certain effet de flou.

En général, on distingue deux types de filtres passe bas: les filtres linéaires, où la transformation d'un pixel est le fruit d'une combinaison linéaire des pixels voisins, et les filtres non linéaire où les pixels voisins interviennent suivant une loi non linéaire.

LE FILTRAGE LINÉAIRE:

L'exemple le plus classique de ce type de filtres est le filtre moyen: il utilise la moyenne non pondérée des voisins et peut être mis sous forme d'un masque tel que celui-ci :

$$H1 = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

On va alors déplacer ce masque sur toute l'image, le pixel affecté par

PRETRAITEMENT

la transformation étant le pixel central du masque, le facteur $1/9$ sert à normaliser le filtre de manière que celui-ci n'influe pas sur l'intensité globale de l'image. Nous avons donc, pour chaque pixel de coordonnées (x, y) :

$$G_f(x, y) = \frac{1}{9} \sum \sum H_1(i, j) G_1(x+i, y+j).$$

Ce filtrage est la technique la plus directe pour adoucir une image; d'autres filtres ont été réalisés en utilisant des coefficients de pondération différents, voici deux exemples :

$$H_2 = \frac{1}{10} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad H_3 = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Le filtre H_2 donne plus de poids aux pixels centrales, ce qui le rend moins actif que le filtre H_1 ; alors que le filtre H_3 privilège les directions x et y .

LE FILTRE NON LINÉAIRE:

L'exemple le plus classique des filtres non linéaires est le filtre médian. Son principe est illustré sur la figure III.2, et il peut être décomposé selon les étapes suivantes :

Pour chaque pixel de l'image :

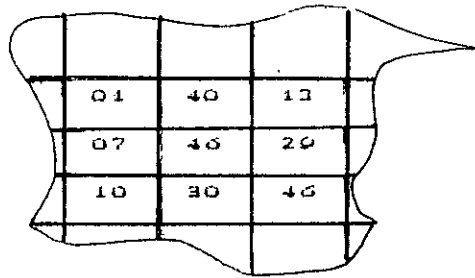
- On classe les pixels voisins du pixel courant par valeurs croissantes.
- On prend la valeur médiane des pixels classées et on l'affecte au pixel courant.

LE FILTRE NON LINÉAIRE:

L'exemple le plus classique des filtres non linéaires est le filtre médian. Son principe est illustré sur la figure III.2, et il peut être décomposé selon les étapes suivantes :

Pour chaque pixel de l'image :

- On classe les pixels voisins du pixel courant par valeurs croissantes.
- On prend la valeur médiane des pixels classées et on l'affecte au pixel courant.



| | | | | | | | | |
|----|----|----|----|----|----|----|----|----|
| 01 | 07 | 10 | 13 | 29 | 30 | 40 | 45 | 46 |
|----|----|----|----|----|----|----|----|----|

FIGURE III.2

III.3.1.c) HOMOGENEISATION DES DONNÉES:

Cette opération a pour but d'améliorer la qualité de l'image en réduisant principalement les effets de la sensibilité des capteurs à la non uniformité spatiale de l'éclairage.

Plusieurs techniques existent, en voici quelques unes

a) L'AUGMENTATION DU CONTRASTE:

La manipulation ponctuelle de l'intensité peut conduire à diminuer

ou à rehausser le contraste d'une image à la suite, par exemple, de mauvaises conditions de prise de vue. La figure III.3 donne trois exemples de fonctions de transferts utilisées respectivement pour le réhaussement des intensités des régions sombres, claires, et, centrales.

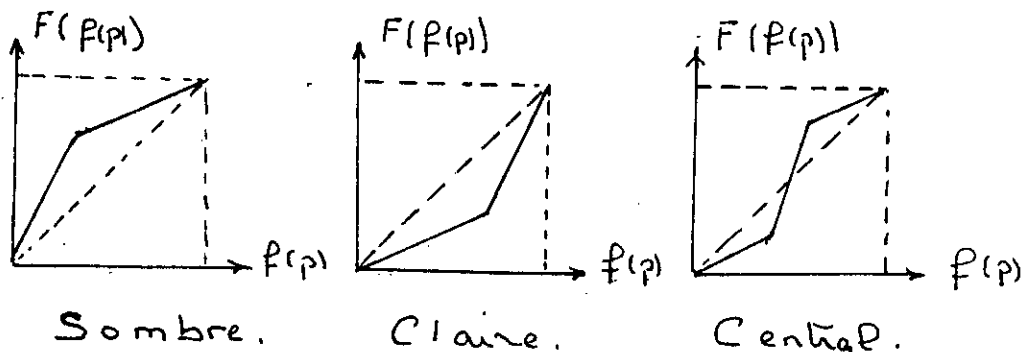


FIGURE III.3

B) AUGMENTATION DU PAS DE QUANTIFICATION :

L'œil humain ne distingue qu'un nombre limité de niveaux de gris dans une image. Pour améliorer l'interprétation visuelle, il est convenable quelque fois de modifier le nombre pas de quantification en appliquant la fonction de transferts suivante :

PRETRAITEMENT

Pour tous point $p \in E$ faire .

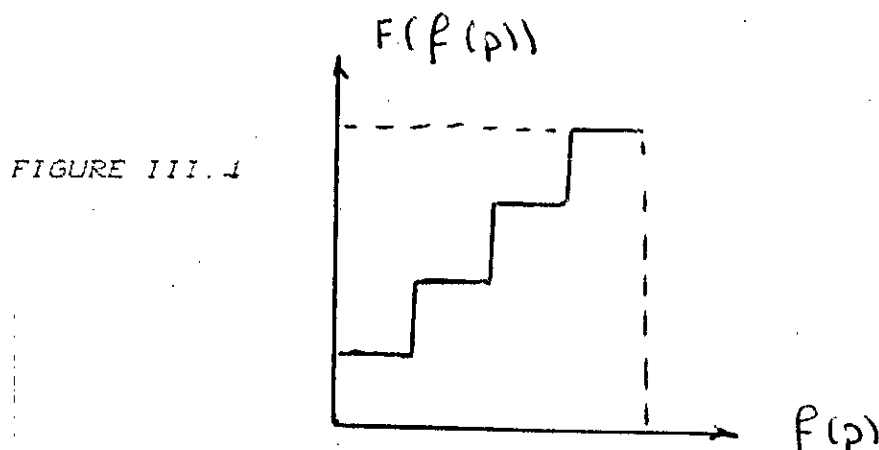
$$F(f(p)) = \left[\frac{f(p)}{\text{pas}} \right]$$

$$\text{avec } \text{pas} = \left[\frac{\text{nvgi}}{\text{nvgf}} \right] .$$

ou nvgi = nbre de niveau de gris initial.

nvgf = nbre de niveau final.

La figure III.4 donne un exemple de telle fonction de transfert.



REMARQUES :

1.

L'utilité de cette dernière procédure dans notre système, est de rendre compatible le nombre de niveau de gris du système d'Acquisition avec le TP 5.5 \ 6.0 qui manipule 16 niveaux de gris.

2.

Parmi les techniques de homogénéisation des données, il existe aussi le filtrage morphologique dont les principaux opérateurs sont

l'érosion et la dilatation pour plus de détail voir spécialement [1], [3], pour leur implémentation voir nos programmes.

III.3.1.d) LE SEUILLAGE:

Cette fonction consiste à examiner les pixels et à ne retenir que ceux dont le niveau de gris est compris entre deux seuils donnés. Ceci a pour but de sélectionner des plages de niveau de gris correspondant à des régions d'intérêt dans la scène étudiée. Selon le cas, l'utilisateur peut affecter un code différent à chaque région ou obtenir une image binaire où le blanc représente toute les régions et le noir représente le fond. Dans ce cas, l'opération de seuillage s'appelle binarisation. et voici un exemple de binarisation.:

$$\text{pour tout point } p \in E \text{ faire } F(f(p)) = \begin{cases} 0 & \text{si } f(p) < S1 \text{ et} \\ & f(p) > S2 \\ 1 & \text{sinon.} \end{cases}$$

Où $S1$ et $S2$ sont deux seuils de détermination d'une plage de niveau de gris. L'opérateur de seuillage F peut s'exprimer d'une manière plus générale par $F(p, f(p), v(p))$, où $v(p)$ désigne le voisinage autour du pixel p .

On peut distinguer deux catégories de seuillage en fonction du niveau de la dépendance de l'opérateur de seuillage F , le seuillage est dit globale lorsque F dépend uniquement de $f(p)$, et local lorsque F dépend de $v(p)$ et de $f(p)$.

a) LE SEUILLAGE GLOBAL:

Le seuillage est réalisé d'une manière identique sur tous les pixels

ls de l'image à partir des seuils déterminés à l'avance. Le problème réside dans le choix des seuils. La méthode la plus classique consiste à calculer les seuils à partir de l'histogramme de l'image. Quand l'image est homogène et bien contrastée, l'histogramme fait apparaître des pics correspondant aux différentes distributions de niveaux de gris et donc aux régions existantes. Les valeurs des seuils sont choisies parmi les vallées qui séparent les pics consécutifs

Cette méthode donne peu de résultat satisfaisant sur des images bruitées où la distribution des niveaux des gris est rarement homogène dans les régions.

De nombreuses méthodes consistent à faire ressortir les populations de pixels du fond ou des objets et à creuser les vallées de séparation. NALEL et RESENFELD réalisent d'abord sur l'image une différenciation qui rehausse les transitions existantes entre régions et qui conduit par la suite à une séparation plus franche des régions de l'image. NAGAO remplace le niveau de gris de chaque pixel par la valeur moyenne du voisinage le plus homogène entourant ce point, sachant que l'on entoure un point de neuf voisinages de neuf pixels chacun, comme suit :

```
222 333 444
222 333 444
222 333 444
555 111 666
555 111 666
555 111 666
777 888 999
777 888 999
777 888 999
```

b) LE SEUILLAGE LOCAL:

En examinant le voisinage des pixels, il est possible de contourner le problème d'hétérogénéité des images. L'approche la plus simple est

la comparaison de la valeur du niveau de gris du pixel étudié avec celles de quelque voisins. Cette approche ne nécessite plus de se fixer des seuils, il s'agit là d'une technique qui permet de décider de la valeur du niveau de gris d'un pixel en fonction de son environnement propre.

WALF [1] utilise deux opérateurs bidimensionnels de tailles différentes pour comparer les niveaux de gris des pixels. Le premier moyenne les niveaux de gris dans un voisinage 3X3. Sur cette image lissée, une comparaison est effectuée entre le pixel central p et l'ensemble des quatre pixels voisins séparés par un pas de trois soit $f(x+3,y+3)$, $f(x+3,y-3)$, $f(x-3,y+3)$ et $f(x-3,y-3)$. Si le niveau de gris du pixel p est plus clair que deux voisins symétriques par rapport à lui, ce pixel est considéré comme objet. Le second opérateur est similaire mais travaille sur un domaine plus vaste, ce qui lui permet de rattraper les formes épaisses.

III.3.2 TECHNIQUES DE NORMALISATION DE LA TAILLE:

La taille du caractère peut varier d'un scripteur à un autre, et peut causer ainsi une instabilité des paramètres. Une technique plus au moins naturelle de prétraitement consiste à ramener les caractères à la même forme.

Nous allons donner deux exemples d'algorithmes de normalisation de la taille.

III.3.2.a) ALGORITHME DE SRIHARI:

Cet algorithme opère en deux étapes. La première normalise le caractère en hauteur et le second en largeur. L'ordre de normalisation ainsi choisie évite que les caractères fins comme le '1' et le 'i' ne

se déforment par rapport à des caractères épais comme le 'm' ou le 's'.

Il s'agit de transformer l'image du caractère de dimensions (h_i, l_i) en une image de dimensions (h, l) . Une étape intermédiaire consiste à produire une image de dimension (h, l') avec :

$$p = \frac{h}{h_i} ; \text{ et } l' = p \times l_i ;$$

La normalisation en hauteur est exécutée en transformant chaque pixel (x, y) de l'image appartenant au caractère en (px, py) . La normalisation en largeur de l'image ainsi obtenue se fait par l'examen de deux cas, si l' est inférieur à l alors l'image normalisée en hauteur est centré dans un rectangle de dimensions (h, l) , si, par contre l' est supérieur à l alors l'algorithme balaye cette image et assigne à la forme tous pixel de coordonnées $((x/l')l, y)$ si le pixel (x, y) l'est aussi.

III.3.2.b) ALGORITHME DE HUSSAIN:

Soit (h_i, l_i) les dimensions du caractère présenté dans une image, il s'agit de ramener ces dimensions à (h_i, l_i) , Pour cela on calcule le rapport $r = h/l$, ensuite plusieurs cas se présentent :

Si $r \leq 3$ alors :

Si $h_i \geq h$ alors :

$nhd \leftarrow h_i - h,$

$nha \leftarrow h - h_i,$

supprimer toutes les lignes de numéros multiple de l'entier immédiatement supérieur à $(h+nhd)/nhd+1$

si $h_i \leq h$ alors :

$nd \leftarrow h / (nha-1);$

$ndi \leftarrow$ entier le plus proche de nd ;

insérer une nouvelle ligne immédiatement en dessous de la ligne de rang multiple de ndi (chaque ligne insérée est une copie de la ligne immédiatement supérieur)

(Les h premières lignes sont retenues pour la normalisation en largeur utilisant une procédure similaire à celle ci).

Si $r > 3$ alors :

Centrer le caractère horizontalement sur une grille de 1 colonne et normaliser à h la hauteur par une procédure identique à la précédente, ceci afin de ne pas trop déformer les caractères allongés de type 'l'.

III.3.3 TECHNIQUES DE SQUELETTISATION :

Le but de la squelettisation est de simplifier l'image du caractère en la ramenant aux résultats de l'écriture continue de la tablette graphique; et elle consiste à chercher une ligne médiane ayant le même ordre de connections que le caractère lui même.

Seulement le résultat n'est pas toujours fiable car il s'accompagne toujours d'une perte irréversible d'information ainsi que de barbelures. La plus part des algorithmes de squelettisation tentent de corriger cela en tentant de conserver les propriétés topologiques et métriques de la forme liées à la courbure, à l'orientation et à l'échelle.

Les critères retenus pour les méthodes de squelettisation sont les suivant :

- L'épaisseur de la squelette doit être de " 1 ".
- La squelette doit conserver les propriétés topologiques de la forme comme le nombre de parties, le nombre de trous et la connexité.
- La squelette doit respecter les propriétés métriques de la forme comme la longueur totale et la distance entre parties de la forme.

Dans la suite on parlera de 5 (cinq) de ces algorithmes, qu'on classera dans les deux grandes classes suivantes :

- Les algorithmes à critère topologiques.
- Les algorithmes à critère métrique.

III.3.3.a) LES ALGORITHMES A CRITERE TOPOLOGIQUE :

Ils sont aussi appelés algorithmes de pelage, ce sont des algorithmes itératifs supprimant à chaque étape, le long de la frontière de la forme, les points appelés inessentiels (c'est à dire n'appartenant pas à la squelette). Ils utilisent souvent des fonctions booléennes opérant sur des voisinages de points déterminant à chaque passage la validité des points frontières.

PRETRAITEMENT

ALGORITHME GENERAL :

Début

répéter

innessential ← faux

pour toute point de l'image faire

si pointinnessential(point) alors

innessential ← vrai

image(point) ← 0

fsi

fpour

jusqu'a innessential = faut

fin

dans cette classe on va voir deux algorithmes : le premier - celui de SERRA - de type séquentiel et le second - celui de THOME - de type parallèle.

a) ALGORITHME DE SERRA :

Pour déterminer si un point est innessential SERRA utilise les masques suivants :

| | | | | | |
|---|---|---|---|---|---|
| 0 | 0 | 0 | X | 0 | 0 |
| X | 1 | X | 1 | 1 | 0 |
| 1 | 1 | 1 | X | 1 | X |

L'ensemble des voisinages 3X3 d'un point innessential est obtenu à partir de ses configurations par rotation de 90°.

L'algorithme de SERRA - comme celui général - est de type séquentiel, il consiste à balayer toute l'image est à supprimer les points

PRETRAITEMENT

inessentiels au fur et à mesure qu'il sont rencontrés. Il ne peut donc pas tenir compte de l'environnement dans lequel il opère et a tendance à enlever trop de points d'un seul coup.

Un autre procédé de type parallèle, consiste à décomposer la suppression en sous cycles permettant d'approcher la squelette par raffinement successif.

EXEMPLE :

soit l'exemple suivant :

| | | | |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 |

Tous les points sont des points frontières et peuvent être considérés chacun à part inessentiel.

Par contre si l'on supprime d'abord les deux points allumés de dessus, on s'aperçoit que les points restant sont essentiels et ne doivent pas être ainsi supprimés.

L'idée d'un algorithme parallèle est donc d'imposer progressivement des contraintes sur les points de manière à dégager petit à petit le vrai squelette.

b) ALGORITHME DE THOME:

C'est un algorithme parallèle, décomposant la suppression en quatre sous cycles :

| | | | | | | | | | | | | | | |
|--------|---|---|---|--------|---|---|---|--------|---|---|---|--------|---|---|
| X | 0 | X | X | 1 | X | X | 1 | X | X | 1 | X | | | |
| E1 = X | 1 | X | ; | E2 = X | 1 | X | ; | E3 = X | 1 | 0 | ; | E4 = 0 | 1 | 1 |
| X | X | X | X | 0 | X | X | 1 | X | X | 1 | X | | | |

PRETRAITEMENT

L'algorithme consiste alors à chercher les points inessentiels suivant E1 et à les supprimer, et refaire ceci successivement pour E2, E3 et E4, et ensuite recommencer jusqu'il n'y aura plus de points à supprimer.

EXEMPLE :

Soit la forme initiale suivante : E =

| | | | | |
|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |

Les résultats successives seront les suivant :

| | | | | | |
|-----------|-----------|-----------|-----------|-----------|-----------|
| 0 0 0 0 0 | 0 0 0 0 0 | 0 0 0 0 0 | 0 0 0 0 0 | 0 0 0 0 0 | 0 0 0 0 0 |
| 1 1 1 1 1 | 1 1 1 1 1 | 1 1 1 1 1 | 1 1 1 1 1 | 0 1 1 0 0 | 0 1 1 0 0 |
| 1 1 0 1 1 | 1 1 0 1 1 | 1 0 0 1 0 | 1 0 0 1 0 | 1 0 0 1 0 | 1 0 0 1 0 |
| 1 1 0 1 1 | 1 1 0 1 1 | 1 1 0 1 1 | 1 1 0 1 1 | 1 1 0 1 0 | 0 1 0 1 0 |
| 1 1 1 1 1 | 0 0 1 0 0 | 0 0 1 0 0 | 0 0 1 0 0 | 0 0 1 0 0 | 0 0 1 0 0 |
| E1 | E2 | E3 | E4 | E1 | E2 |

Schéma d'une squelettisation utilisant
l'algorithme de thome.

c) L'ALGORITHME DE BELAIDE :

L'idée de cette algorithme est l'extraction de la squelette par amincissement répète.

L'élément structurant utilisé est :

| | | |
|-------|---|-----|
| 0 | 0 | 0 |
| B = X | 1 | X ; |
| 1 | 1 | 1 |

et par rotation en construit la séquence suivante :

PRETRAITEMENT

$$\begin{array}{cccc}
 0 & 0 & 0 & X & 0 & 0 & 1 & X & 0 & 1 & 1 & X \\
 B1 = X & 1 & X & , & B2 = 1 & 1 & 0 & , & B3 = 1 & 1 & 0 & , & B4 = 1 & 1 & 0 & , \\
 1 & 1 & 1 & & 1 & 1 & X & & 1 & X & 0 & & X & 0 & 0
 \end{array}$$

$$\begin{array}{cccc}
 1 & 1 & 1 & X & 1 & 1 & 0 & X & 1 & 0 & 0 & X \\
 B5 = X & 1 & X & , & B6 = 0 & 1 & 1 & , & B7 = 0 & 1 & 1 & , & B8 = 0 & 1 & 1 & , \\
 0 & 0 & 0 & & 0 & 0 & X & & 0 & X & 1 & & X & 1 & 1
 \end{array}$$

La squelette est alors obtenue par élimination des points dite inessentiels de B1 sur toute l'image et refaire cela successivement pour B2, B3, ..., B8, puis en recommence dès le début et ainsi de suite. On s'arrête quand aucun point n'est plus enlevé.

Cette opération s'accompagne souvent par l'application d'autre élément structurant qui enlèvent les barbales ou branches restantes,

d) ALGORITHME DE ZHANG et SUEN :

Cet algorithme est itératif, chaque itération comprend deux passages par la forme (ou deux balayages); le premier passage force à zéro un point, P_i , de la forme satisfaisant les conditions suivantes :

- a) $2 \leq N(P_i) \leq 6$;
- b) $S(P_i) = 1$;
- c) $P_2 * P_4 * P_6 = 0$;
- d) $P_4 * P_6 * P_8 = 0$;

où $N(P_i)$ est le nombre de pixels allumés voisins de P_i et $S(P_i)$ est le nombre de transition 0-1 dans l'ordre P_2, P_4, \dots, P_8 . Par exemple $N(P_i) = 4$, et, $S(P_i) = 3$ pour la figure III.5.

| | | |
|----------------|----------------|----------------|
| P ₀ | P ₂ | P ₃ |
| P ₈ | P ₄ | P ₄ |
| P ₇ | P ₆ | P ₅ |

| | | |
|---|----------------|---|
| 0 | 0 | 1 |
| 1 | P ₁ | 0 |
| 1 | 0 | 1 |

FIGURE III.5

le second passage force a zero un point de la forme satisfaisant les conditions a) et b) précédente et c) et d) suivante :

$$c) P_2 * P_4 * P_8 = 0;$$

$$d) P_6 * P_6 * P_8 = 0;$$

L'algorithme s'arrête lorsqu'il n'est plus possible d'éliminer des points.

Il est a noter que cette algorithme présente certains grave défauts qui sont :

- 1- On perd des pixels qui doivent faire partie de la squelette.
- 2- Certaines* formes disparaissent totalement, c'est le cas des formes carrées.

REMARQUE :

Pour plus de détails concernant cet algorithme prière se rapporter a [8].

III.3.3.b) LES ALGORITHMES A CRITERE METRIQUE :

A l'envers des algorithmes précédents, on s'intéresse plus ici aux points formant la squelette, c'est à dire aux points essentiels. Ces points sont normalement situés aux centres de la forme, c'est à dire à égal distances de la frontière, cette notion de distance est la clé de toutes ces méthodes métriques.

On définit la distance d'un point x à un ensemble de points H par :

$$d(x,H) = \text{Min} (d(x,y)) \text{ pour tout } y \in H;$$

Soit B la frontière de la forme, la squelette S est alors défini par :

$$S = \{ X/\exists (Y,Z) \in B^2; y \neq z : d(x,B)=d(x,y)=d(x,z) \}.$$

Cette définition de la distance ne doit pas être appliquée de manière stricte car la distance entre les points de l'image est influencée par l'échantillonnage.

Dans le paragraphe qui suit nous détaillons l'algorithme de MARTON.

ALGORITHME DE MARTON :

C'est un algorithme local qui sélectionne les points qui sont des maximums pour leur distance à la frontière du caractère. Cette algorithme ne considère que les voisinages de type 3×3 ; Pour chaque point il calcule la somme des vecteurs joignant ce point à ces voisins; Les coordonnées des points voisins prennent leur valeurs dans l'intervalle $[-1, 0, +1]$ suivant leur position par rapport au point central, considéré comme origine, la longueur est calculée en sommant les valeurs des composantes X et Y du vecteur somme; elle est comprise entre 0 et 4.

Considérons le point ayant les 4 voisinage $V1, V2, V3, ET V4$ disposés

comme suit: V3 . V2

. . V1 dans ce cas le vecteur somme a pour composantes:

. . V4

$$\begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} -1 \\ +1 \end{pmatrix} + \begin{pmatrix} +1 \\ +1 \end{pmatrix} + \begin{pmatrix} +1 \\ 0 \end{pmatrix} + \begin{pmatrix} +1 \\ -1 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \end{pmatrix};$$

$$\Rightarrow |X| + |Y| = 3;$$

Intuitivement, nous pouvons dire que les points squelette sont associés a une longueur $|X| + |Y|$ petite comme le montre l'exemple de la figure III.6. L'algorithme de squelettisation élimine les points dont la valeur est égale a 4 et retient ceux dont la valeur est inférieure ou égale a 2, les dans la valeur est égale a 3 nécessite d'abord une étude de voisinage.

Par exemple pour la configuration suivante :

| | | |
|---|---|-----|
| V | . | V |
| . | P | V ; |
| . | . | V |

on a :

$$\begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} -1 \\ +1 \end{pmatrix} + \begin{pmatrix} +1 \\ +1 \end{pmatrix} + \begin{pmatrix} +1 \\ 0 \end{pmatrix} + \begin{pmatrix} +1 \\ -1 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \end{pmatrix};$$

$\Rightarrow |X| + |Y| = 3$; et le point P doit être gardé pour éviter une disconnection de la

forme.

Par contre, dans le cas ci-dessous :

| | | |
|---|---|----|
| . | . | V |
| . | P | V; |
| . | . | V |

on a:

$$\begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} -1 \\ +1 \end{pmatrix} + \begin{pmatrix} +1 \\ +1 \end{pmatrix} + \begin{pmatrix} +1 \\ 0 \end{pmatrix} + \begin{pmatrix} +1 \\ -1 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \end{pmatrix};$$

PRETRAITEMENT

$\Rightarrow |X| + |Y| = 3;$
et P doit être éliminé.

Pour l'ensemble des points la décision peut être prise en incluant une information concernant le nombre de points voisins (nvois); Il y a cependant deux exceptions a cette règle correspondant aux deux cas de la ligne double ou de colonne double.

DESCRIPTION DE L'ALGORITHME :

L'algorithme est itératif, chaque itération comprend deux passages par la forme ; le premier passage calcule les caractéristiques locales de chaque point ($|X|$, $|Y|$ et nvois), le deuxième élimine ou non les points en fonction de leur caractéristiques; l'algorithme s'arrête lorsqu'il n'est plus possible d'éliminer des points.

Premier balayage :

```

pour chaque point P à traiter faire
calculer |X|, |Y|, et nvois,
si |X|+|Y| ≠ 3 alors faire |P| ← |X|+|Y|
sinon
    si nvois = 2 ou nvois = 6 ou
        (nvois = 3 et max(|X|, |Y|) = 3 )
        ou (nvois = 5 et Max(|X|, |Y|) = 3)
    alors :
        si Configuration ligne double ou
            colonne double alors |P| ← 3
            sinon |P| ← 5
        fsi
    sinon |P| ← 3
    fsi
fsi
fpour.
    
```

Deuxième balayage :

éliminer tous les point P tel que $|P| \geq 4$.

La squelette obtenue contient des ébarbules et parfois une épaisseur supérieure a 1. un dernier passage par l'image du caractère élimine ces défauts en supprimant les points ayant une des configurations suivantes :

- 1) $|X| = 1, |Y| = 1$ et $nvois = 2$;
 - 2) $|X| = 0, |Y| = 2$ et $nvois = 3$;
 - 3) $|X| = 2, |Y| = 0$ et $nvois = 3$;
-

III.4 CONCLUSION :

Les points originaux les plus important traités dans ce chapitre sont, sans doute, les techniques de lissages, de normalisation de la taille et de squelettisation, le reste peut être plus ou moins estimé réductementaires.

Pour le lissage les résultats obtenus avec les différents masques utilisés sont très satisfaisant; en effet, les irrégularités telles que :

- points isolées ou situées à l'extrémité de la forme;
- points formants des excroissances;
- Absence de points et trous internes à la forme;
- Irrégularités des contours de la forme;
- ...etc.

sont toutes totalement éliminées, ceci d'une part, d'autre part on a constaté que ces opérations sont peu coûteuse en temps d'exécution ce qui permet de les utilisées en grands nombres.

Pour la normalisation de la taille les résultats obtenues avec les deux algorithmes implémentés sont extraordinaires, seul deux remarques sont à noter :

- 1- Comme on s'attend, l'algorithme de SRIHARI introduit un découpage dans les formes de petites tailles à normaliser; Ce découpage peut être dans une seule direction ou suivant les deux directions, bien sure pour les caractères à grand taille ce découpage n'aura pas lieu. Du point de vu temps d'exécution on a constaté que cet algorithme est extrêmement rapide.
- 2- Pour l'algorithme de HUSSAIN, le problème de découpage, par principe même, ne se pose pas, mais par contre il exige un temps d'exécution plus grand que dans le

cas précédant.

Ces deux remarques peuvent être vue comme des critères de choix entre les deux algorithmes.

Pour le cas qui nous occupe le second algorithme (de HUSSAIN) est évidemment le plus adopté ceci car la méthode d'approximation angulaire de FREEMAN utilisé pour le codage de nos caractères suppose que la squelette ne présente aucune discontinuité artificielle.

Pour la squelettisation est malgré le grand nombre d'algorithmes qu'on a implémenté, les critères retenus ne sont toujours pas totalement obtenus. En effet si parmi les algorithmes implémentés, il y a quelques uns qui élimine les défauts tels que:

- Epaisseur supérieure à " 1 ".
- perd de pixels qui doivent faire partie de squelette.
- Disparitions total de certaines formes.

- Discontinuités.
- ... etc.

Le problème des ébarbules reste irrésolvable.

Pour nous une comparaison expérimentale nous a permis de préférer l'algorithme de THOME pour notre application, et ceci pour les raisons suivantes :

- Il conserve d'une manière très satisfaisante les propriétés topologiques du caractère.
- Il respect les propriétés métriques du caractère.
- Les ébarbules engendré ne déforme pas trop le caractère.
- Temps d'exécution très satisfaisant.

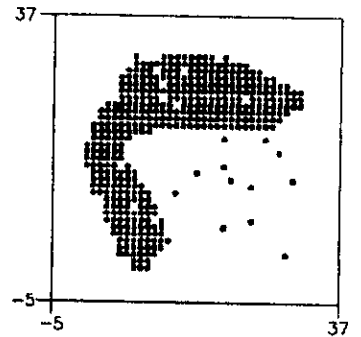
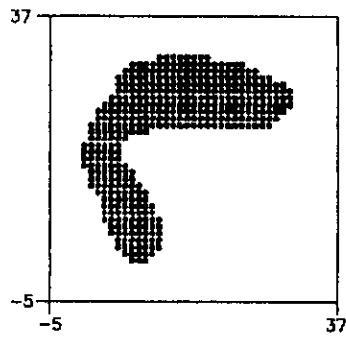
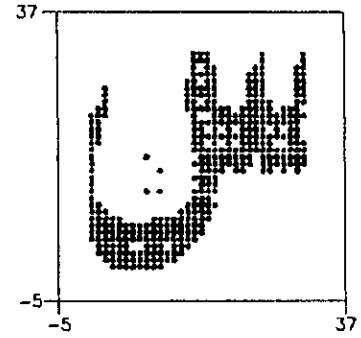
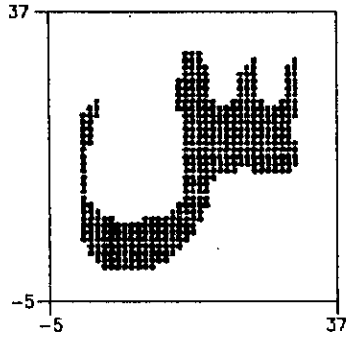
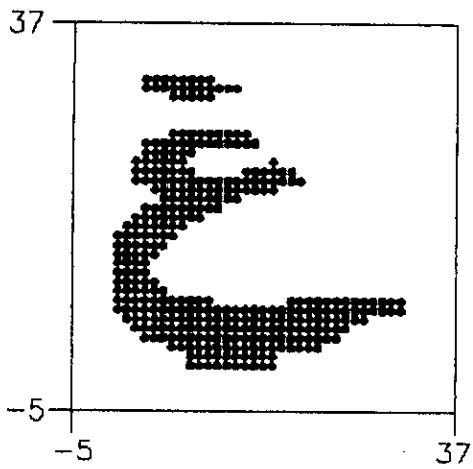
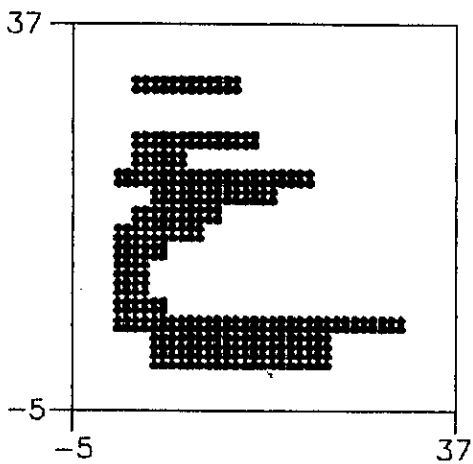
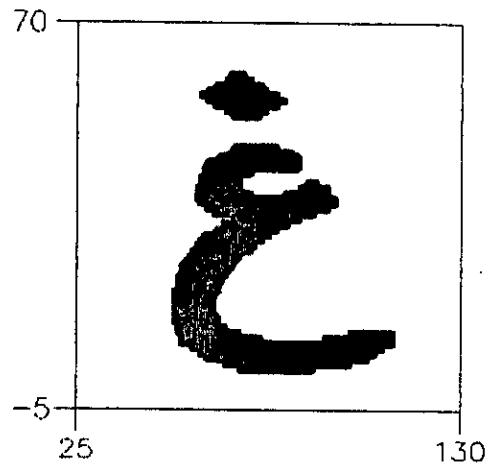


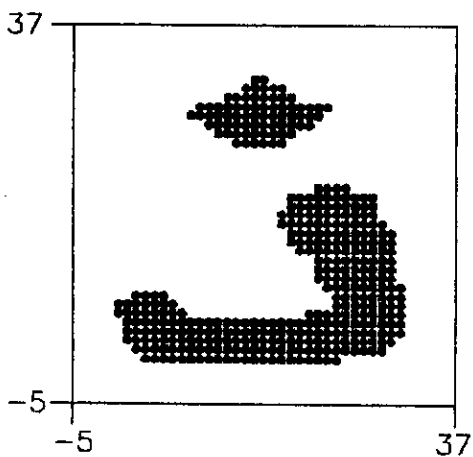
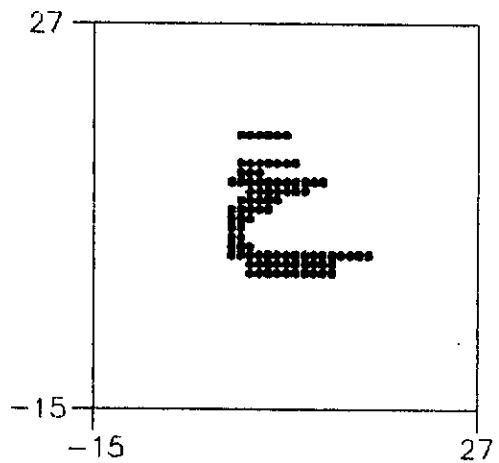
Image 1 :- Exemples de lissage de caractères
Par les filtres présentés



-(a)-



-(b)-



-(c)-

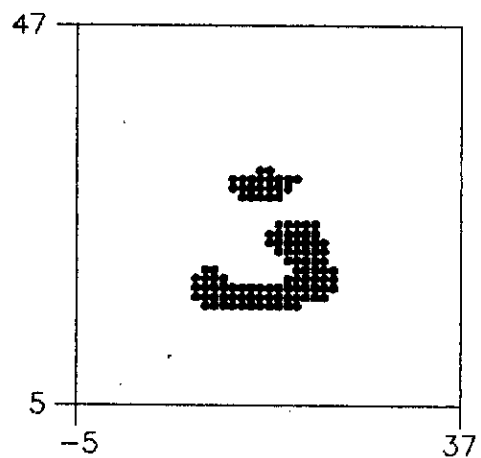
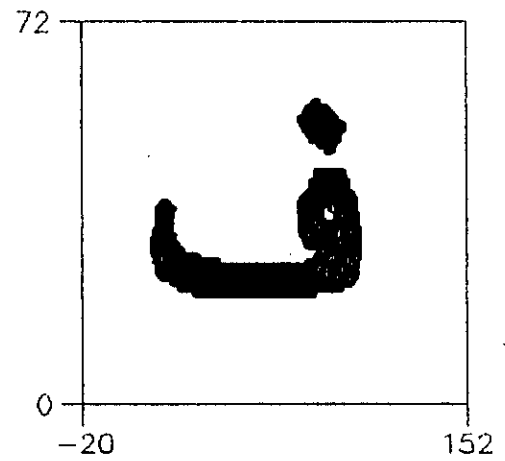
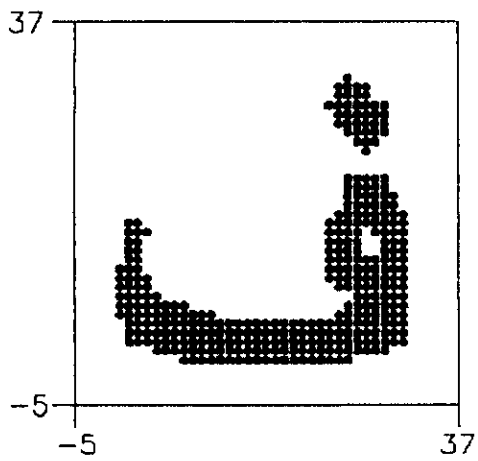
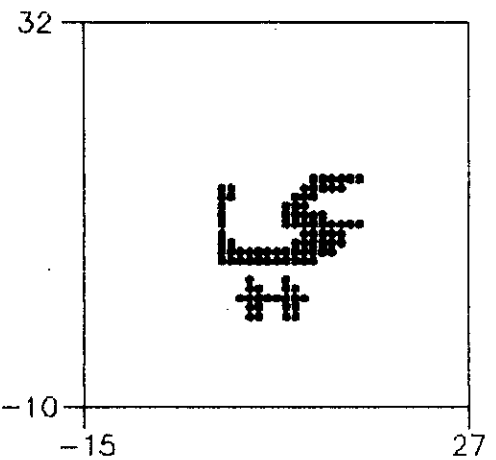
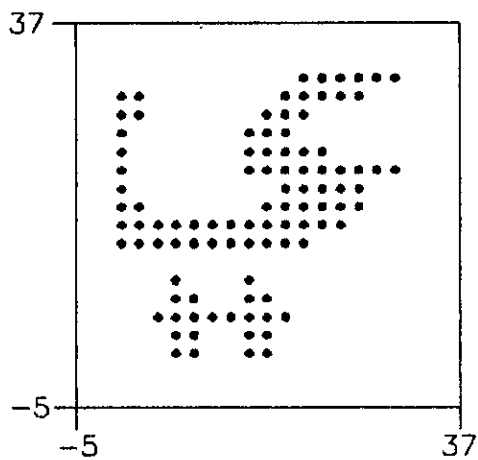


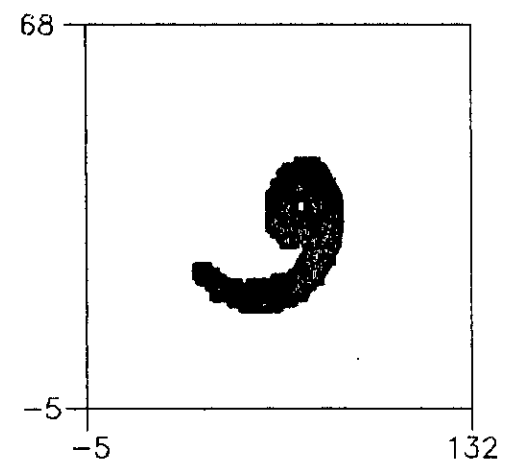
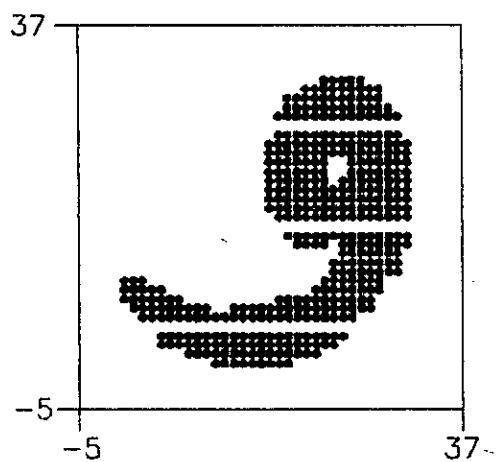
Image 2 :- Techniques de Normalisation de la taille. (a) & (b) : Normalisation avec l'algorithme de HUSSAIN. (c) : Norm. avec celui de SRIHARI.



-(d)-



-(e)-



-(f)-

Image 3 :- [Suite.]

-(d), (e), (f) : Normalisation avec l'algorithme de SRIHARI.

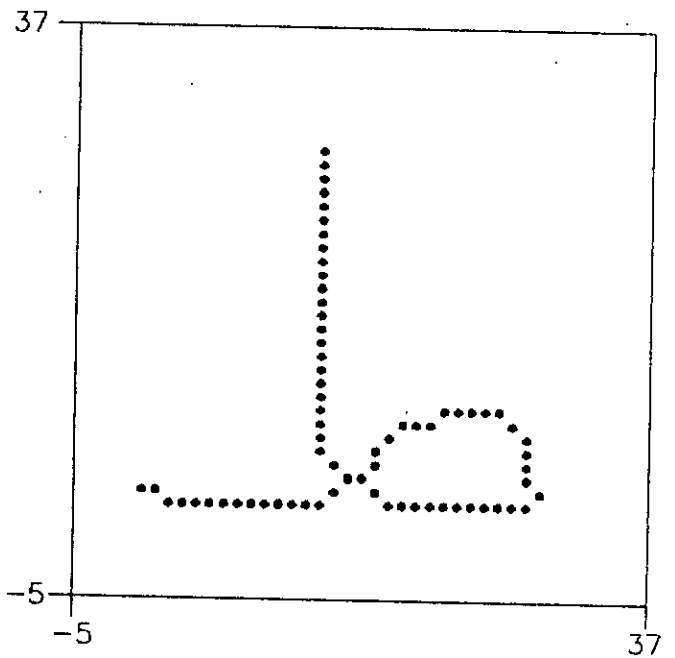
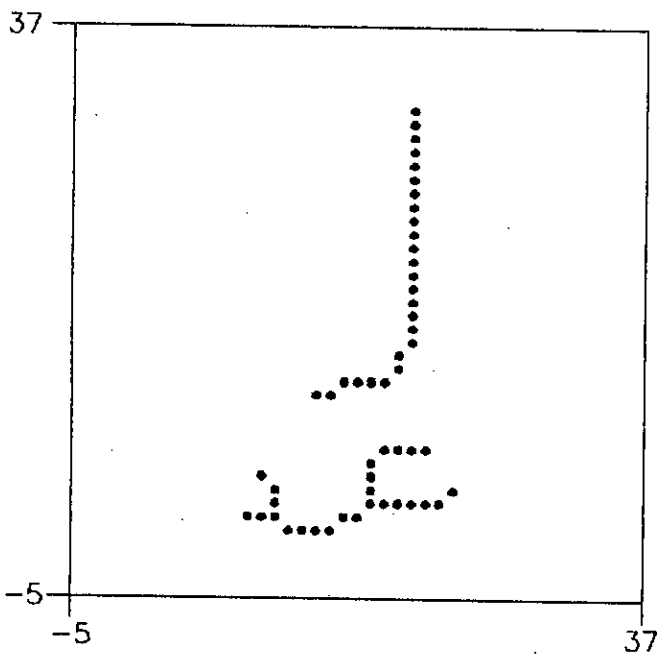
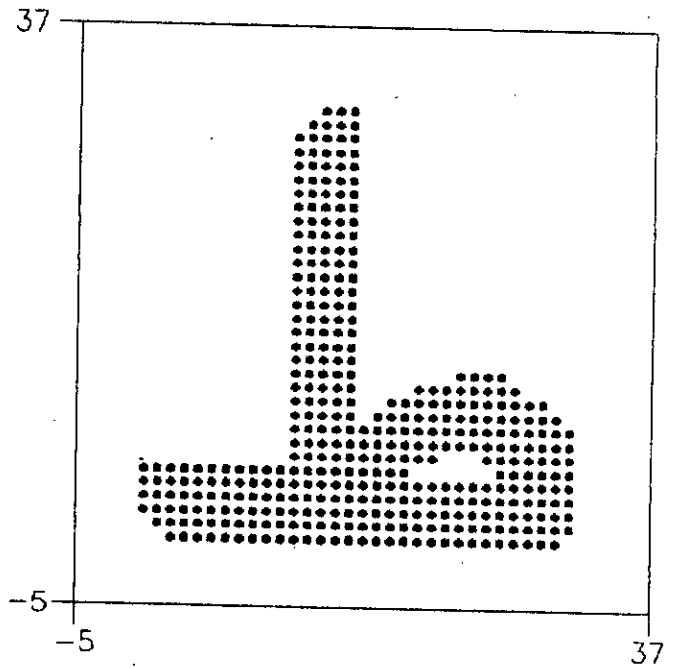
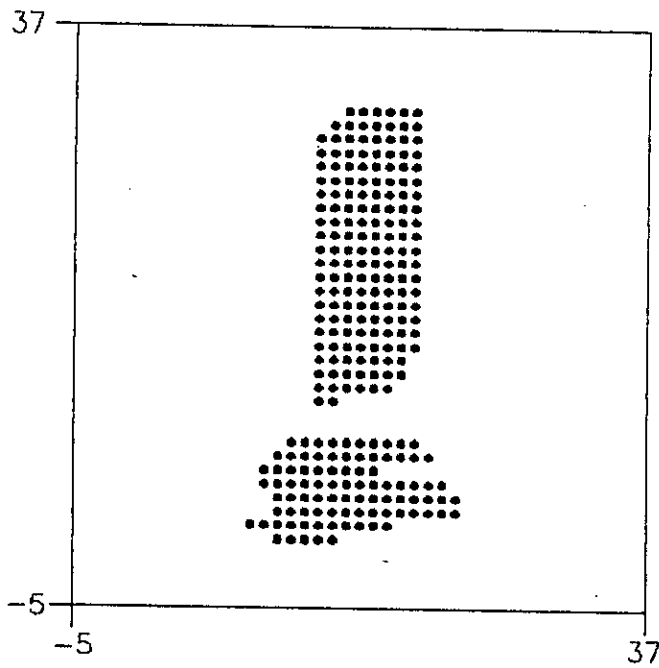


Image 4 : Techniques de Skeletonisation.

(a) : Skeletonisation avec l'Algorithme de SERRA.

(b) : Skeletonisation avec l'algorithme de Thomé

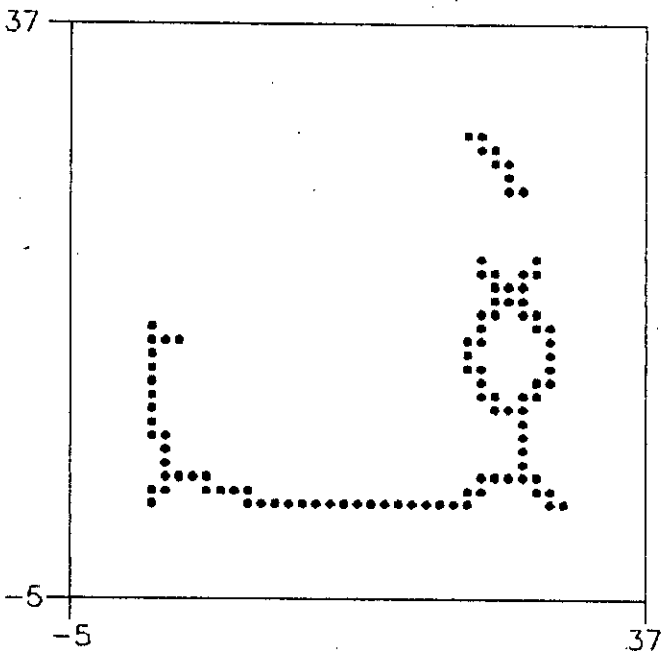
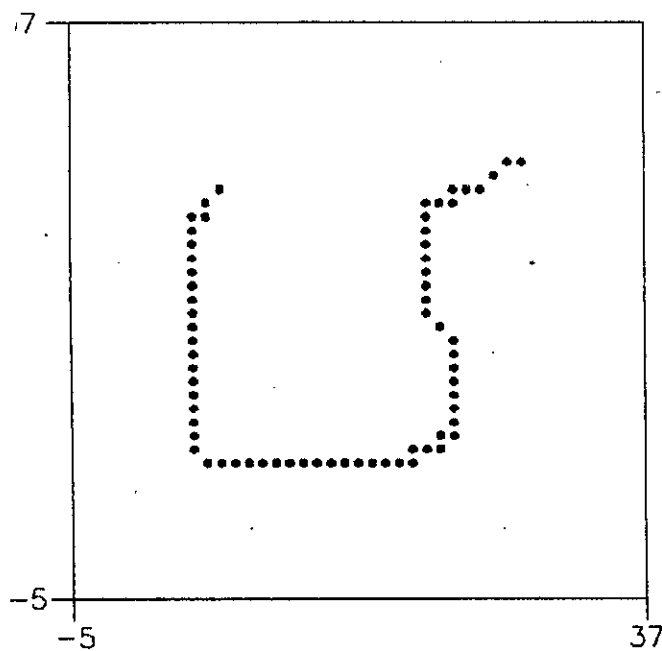
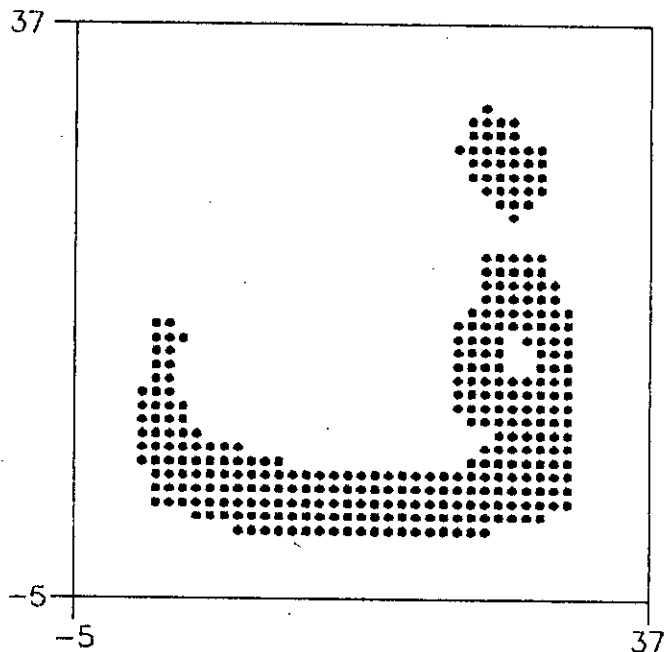
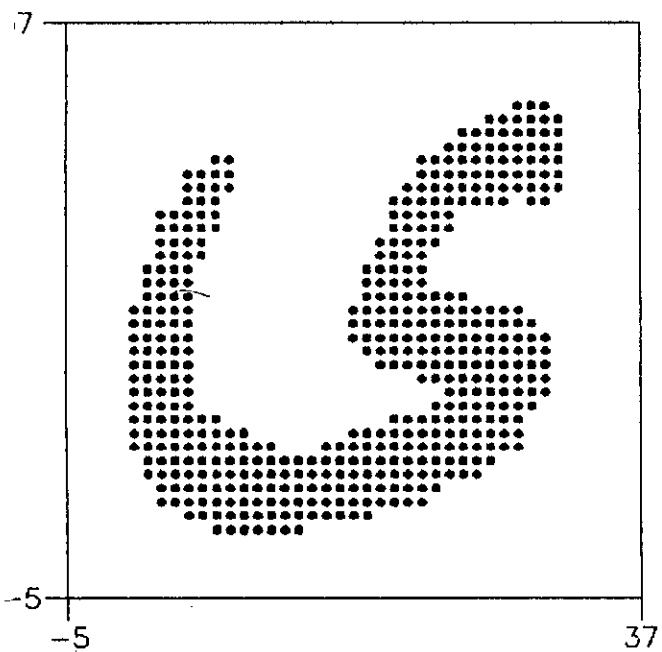


Image 5 : [Suite.]

(c) : Squelettisation avec l'algorithme de MARTON.

(d) : Squelettisation avec l'algorithme de BELAD.

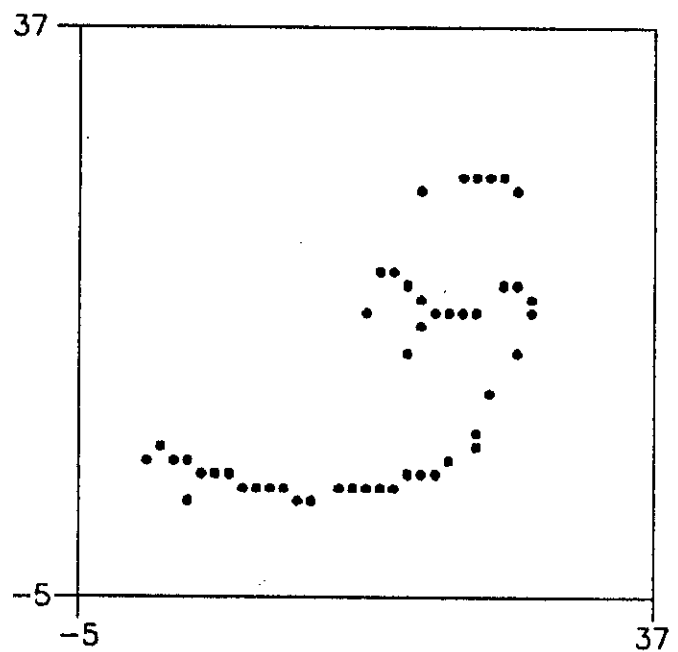
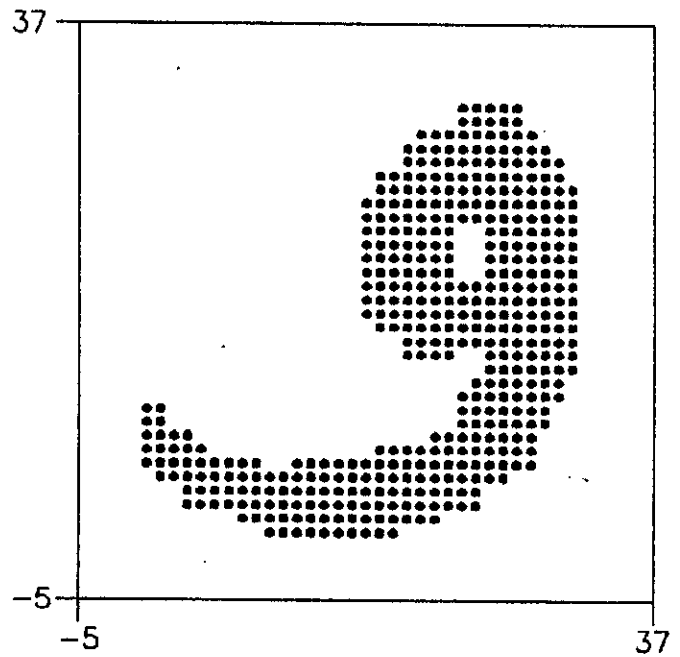


Image 6: [Suite.]
(e) : Squelettisation avec l'Algorithme
de ZHANG et SUEN.

CHAPITRE IV

ANALYSE ET EXTRACTION

DES PARAMETRES

ANALYSE ET EXTRACTION
DES PARAMETRES

IV.1 DIFFÉRENT APPROCHE DE REPRÉSENTATION :

Il existe deux grandes méthodologies utilisées en reconnaissance de forme : l'approche dite globale et l'approche dite structurale.

IV.1.1 L'APPROCHE GLOBALE :

Cette approche est fondée sur l'étude des propriétés de la forme sans distinction de composition ou de structure. De manière générale, on trouve deux grands types de mesures :-

IV.1.1.a) MESURE NUMÉRIQUE :

Les mesures correspondent à des calculs quantitatifs, tels que le diamètre, l'aire, le périmètre, coefficients de FOURIER, ...etc.

IV.1.1.b) MESURE LOGIQUE :

Les mesures correspondent à des calculs qualitatifs. L'objet est représenté par un vecteur binaire où chaque composante indique la présence ou l'absence d'une certaine propriété.

IV.1.2 L'APPROCHE STRUCTURELLE :

L'approche globale est très efficace sur les formes simples. Elle devient totalement inutile sur des formes riches en informations structurelles. C'est le cas en analyse d'images où souvent la décision n'est pas uniquement la détermination d'un seul nom ou le calcul d'une seule mesure mais plutôt la description d'une certaine situation; par

exemple, les objets se trouvant dans l'image et leurs dispositions mutuelles. Cette description se réalise souvent par l'analyse de l'objet en termes de ses formes primitives et de leurs dispositions. C'est ce que l'on appelle l'approche structurelle.

IV.2 TECHNIQUE DE REPRÉSENTATION :

Dans les paragraphes qui suivent; on commence par détailler les différentes techniques et stratégies d'extraction de paramètres utilisées en reconnaissance des formes dans les deux méthodologies précédentes; et on terminera par la représentation et le codage adapté pour notre application.

IV.2.1 TECHNIQUE DE L'APPROCHE GLOBALE :

Parmi ces mesures il y a :

IV.2.1.a) LES MESURES TOPOLOGIQUES :

Parmi ces mesures élémentaires on trouve :

- La surface égale, par exemple, au nombre de pixels situés à l'intérieure de la frontière de l'objet.
- Le périmètre égale au nombre de pixels situés le long de la frontière de l'objet.
- Le nombre de cavités égale au nombre de trous ou zones composées de pixels n'ayant pas les mêmes niveaux de gris que les pixels de l'objet n'ayant pas les mêmes niveaux de gris que les pixels de l'objet.

IV.2.1.b) LES MESURES DE FORME :

Il existe différentes méthodes utilisées pour caractériser la forme. En voici deux exemples de nature globale :

- La compacité : $C = \frac{S}{P^2}$,
où S est la surface et
R le périmètre.

-L'allongement : $A = \frac{\text{long}}{\text{larg}}$;

où long correspond à la plus grande dimension de la région, et larg à la plus petite.

IV.2.1.c) LES MESURES STATISTIQUES :

Pour l'image, cela revient à analyser la distribution statistique des données en vue de la préclassification des sous ensembles.

L'HISTOGRAMME :

Une image peut être considérée statistiquement comme une distribution particulière de la brillance. Dans ce modèle statistique, les mesures des pixels sont considérées comme la réalisation des variables aléatoires et les distributions correspondantes peuvent être estimées par les histogrammes de ces mesures.

L'histogramme dans l' image est une fonction de R dans R qui à toute valeur ou mesure dans l'image, associe son nombre d'occurrences et de ce fait, l'histogramme résume au mieux l'information globale que contient l'image et donc sa structure statistique. La forme de l'histogramme est significative de la manière de la distribution des niveaux de

gris dans l'image. Elle donne des indications précises sur la caractè-
re et la nature de l'image dont elle est issue. Si l'histogramme est
étroit, l'image à une faible contraste lumineux. Si l'histogramme est
bi ou tri-modal, l'image peut être segmentée en deux ou trois régions
de radiométrie très différentes, ...etc

IV.2.2 TECHNIQUES de L'APPROCHE STRUCTURELLE :

Pour commencer introduisons la notion de vecteur de forme : soit le
problème de la reconnaissance des caractères; Une typique approche
sera de projeté le caractère sur une matrice photosensible dont la
réponse de chaque cellule est '1' si elle contient une aire suffisam-
ment large du caractère et '0' sinon; Un exemple de résultat est donné
à la figure IV.1 où une cellule ombrée correspond à une réponse '1'.
Par commodité d'utilisation il convient en général d'arranger ceci
sous forme d'un vecteur appelé vecteur de forme dont la dimension n
est égale au nombre de cellule de la matrice utilisée.

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} ; \text{ avec } x_i = 1 \text{ ou } 0 \text{ suivant l'état de la cellule} \\ \text{correspondante.}$$

IV.2.2. a) STRUCTURE DE CHAINE :

Considérons maintenant la structure en escalier de la figure IV.2, il est bien sur possible de la coder sous forme d'un vecteur avec une technique similaire à la précédente; Mais il est à noter qu'une méthode (représentation) plus efficace résultera si on tire avantage des relations structurelles existantes: la répétition géométrique dans ce cas précis; par définition de deux primitives élémentaire a et b figure IV.2.b, on peut coder la figure IV.2.a, sous la forme montrée à la figure IV.2.c.

On obtient ainsi une chaîne de symbole $w = abab...$ qui caractérise complètement la structure considérée.

Contrairement à la description numérique précédente; ce type de représentation produit une forme composé de primitives élémentaires dont la signification reste reliée à l'image originale.

Un autre exemple de la représentation en chaîne est donné à la figure IV.3 ; dans ce cas les structures sont des triangles équilatéraux

base horizontale de différentes tailles. Si on considère que les primitives sont à longueur égales; On peut représenter de tel triangle avec des chaînes de la forme $aaa...bbb...ccc...$ ou tout simplement $a^n b^n c^n$.

D'autre part il est toujours possible d'augmenter la capacité de description de la représentation en chaîne par définition d'opérateurs qui permettent d'exprimer sous forme d'une chaîne des figures et formes plus complexes.

Quatre de tels opérateurs sont donnés à la figure IV.4 dans cette représentation des primitives sont des segments de lignes orientés et les opérations permettent leurs connexions uniquement entre leur bout ou queue; Comme exemple de cette approche considérons les primitives définies dans la figure IV.5.a , la figure IV.5.b montre la primitive d et la figure IV.5.c est le résultat de l'application de l'opérateur '+' avec les primitives c et d (le symbole \sim indique une inversion dans la direction de la primitive). Notons les positions du bout et queue de la structure composé $c+(\sim d)$; l'ajout de la primitive de cette structure au moyen de l'opérateur '+' donnera comme résultat la struc-

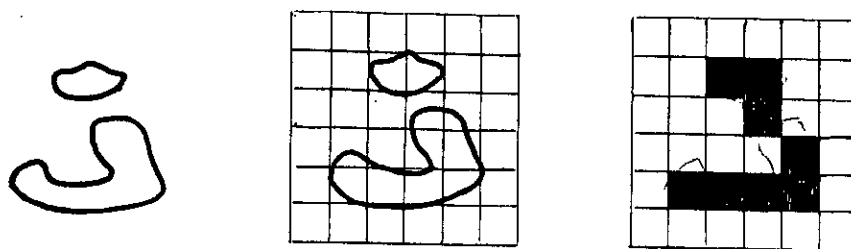


Fig. 5

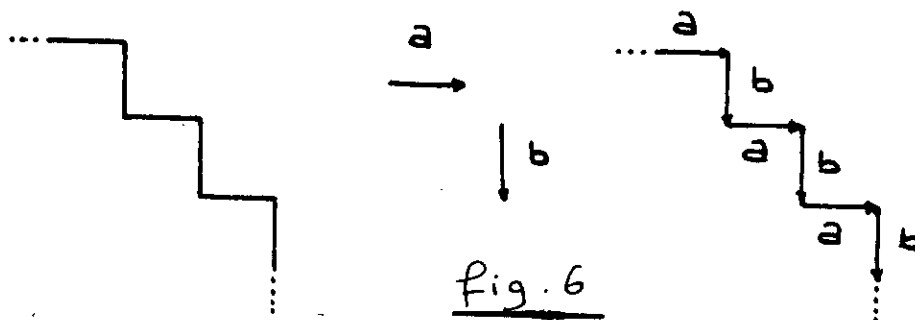


Fig. 6

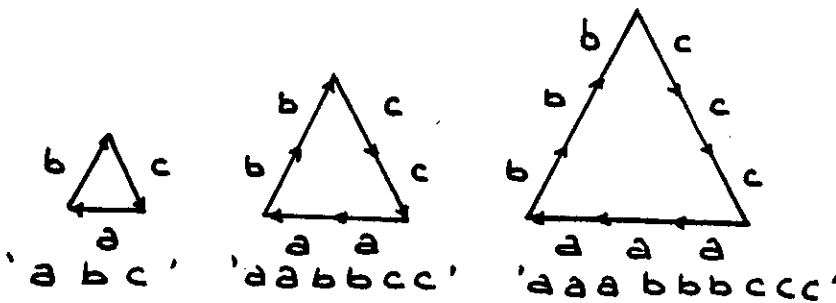


Fig. 7

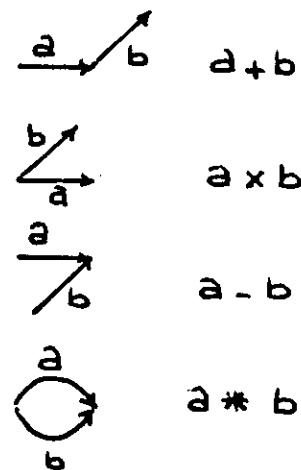


Fig. 8

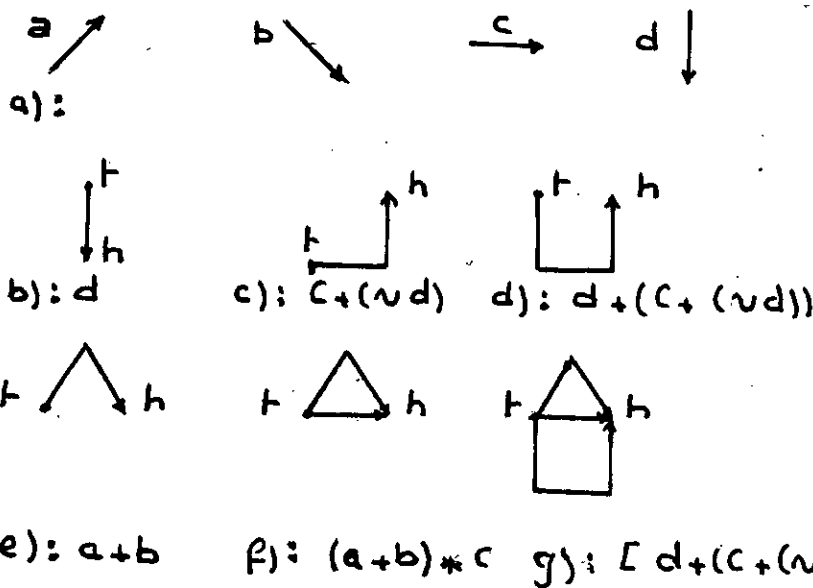


Fig. 9

ture de la figure IV.5.d. Une procédure similaire produira la structure composée montrée à la figure IV.5.f; Finalement l'application de l'opérateur '*' entre les deux structures précédentes nous font aboutir à la forme de la figure IV.5.g.

La représentation de chaîne est adéquate pour la description d'objet ou toute autre entité dont la structure est basé sur la connexion de primitive relativement simple

Une étude plus pratique sera donnée à la fin de ce chapitre.

IV.2.2.b) STRUCTURE D'ARBRE :

Une plus puissante approche pour beaucoup d'applications est réalisée autour de la représentation (structure) d'arbre.

Fondamentalement, tout système ordonné hiérarchiquement conduit à une structure d'arbre, considérons par exemple l'image simple de la figure IV.6. labelons l'image entière avec le symbole \$; La représentation d'arbre montré à la figure IV.7 est obtenu par l'utilisation de la

relation "A l'intérieur de".

La figure IV.8 donne un deuxième exemple qui consiste en une simple scène composée de bloc. A la figure IV.9 on voit son représentation à l'aide de l'utilisation de la relation "composé de".

Pour une étude plus détaillée voir [5]; et pour une application originale voir [3].

IV.2.2.c) STRUCTURE DE GRAPHE :

La représentation chaîne est un cas spécial de la représentation d'arbre qui consiste en une seule branche dont les noeuds sont labelés dans un ordre similaire au symbole dans la chaîne. De même, la représentation d'arbre est un cas spécial d'une classe de représentations plus général appelé graphe. L'utilisation des graphes pour la reconnaissance de forme est illustré à la figure IV.10, la partie (a) de cette figure montre un simple objet, la figure IV.10.b montre son approximation par des polygones; l'interconnexions des polygones produit la représentation en graphe de la figure IV.10.c; Pour une deuxiè-

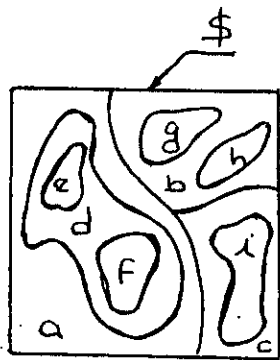


Fig 10

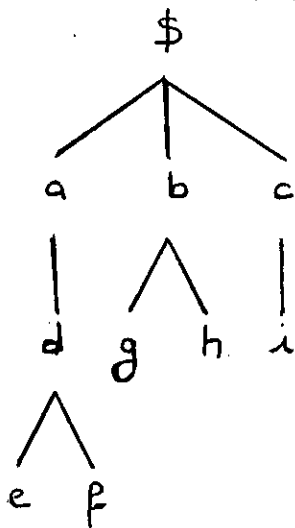


Fig 11

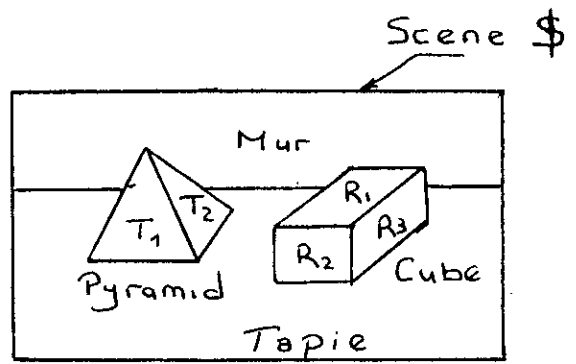


Fig 12

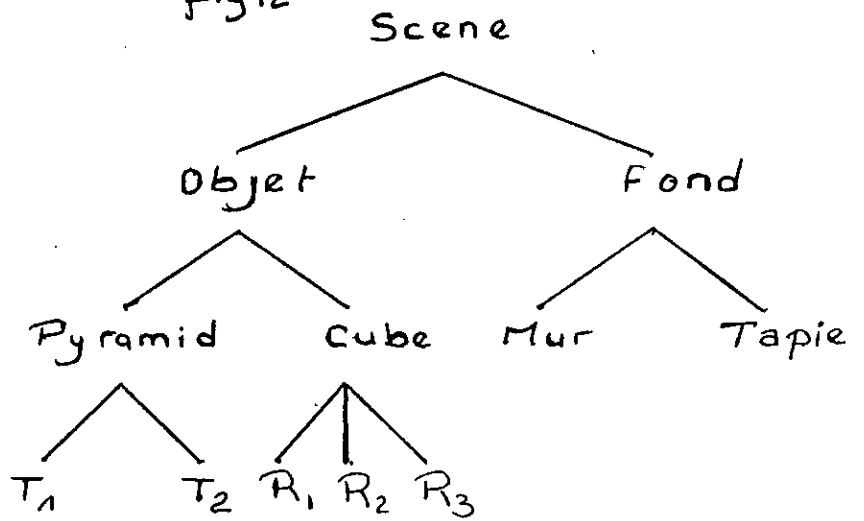
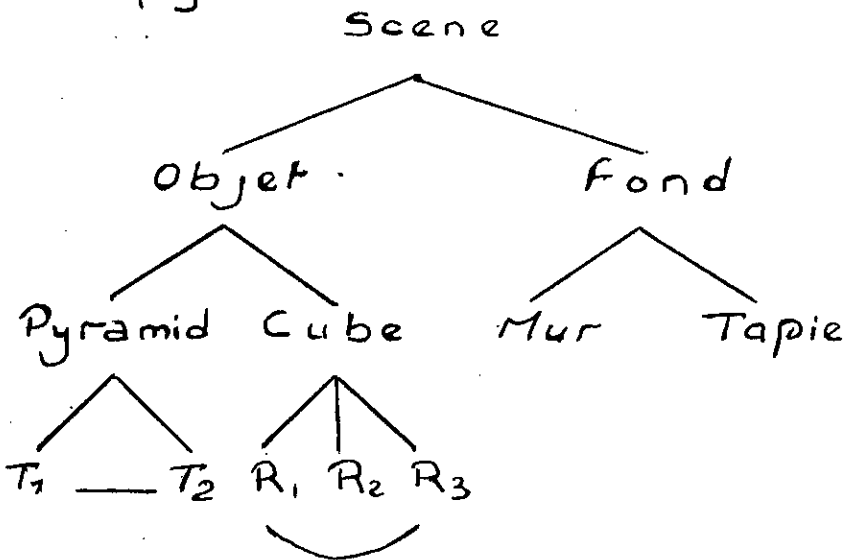
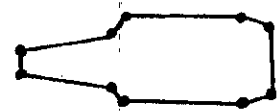
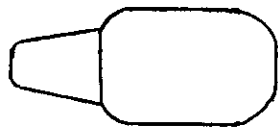


Fig 13



me illustration considérons une autre fois la figure IV.8; Si on ajoute à la relation "composé de" la relation "connecté à" on obtient le graphe de la figure IV.10.d; Ainsi on remarque que les composantes triangulaires T_1 et T_2 sont reliées par un arc indiquant qu'ils sont connectés de même pour R_1 , R_2 , et R_3 d'une part et le mur et le tapis d'autre part.

D'après ce qui précède on constate que les graphes sont des outils de descriptions beaucoup plus que modèles permettant réellement une apprentissage et une décision; ceci est lié à la constatation d'évidence que plus un outil est général, et moins sa manipulation est rapide pour un cas précis; malgré ceci un large aspect algorithmique de la théorie des graphes qui peuvent se révéler utiles en reconnaissance de forme existe.

IV.3 DÉVELOPPEMENT SUR LA STRUCTURE DE CHAÎNE:

QUELQUES DÉFINITIONS ET MOTS CLÉS :

DÉFINITION 1 :

On appellera alphabet un ensemble fini noté X . ses éléments, appelés lettre, seront notés : a, b, c, \dots ; en reconnaissance des formes, les lettres de l'alphabet seront les formes élémentaires distinctes issues du prétraitement.

DÉFINITION 2 :

On définit une phrase sur X comme une suite ordonnée d'éléments de X représentée par simple juxtaposition (concaténation) de ces éléments.

Exemple : Alphabet: $X = \{a, b, c, \}$

Phrase sur X : $X = bcaab.$

DÉFINITION 3 :

La longueur d'une phrase x est le nombre d'éléments qu'elle comporte. on la note $|x|$.

Dans l'exemple précédent : $|x| = 5$.

REMARQUE :

La terminologie est assez peu fixée : en théorie des langages, une lettre peut être appelée un " *symbole* " ; Une phrase peut aussi s'appeler *mot* (ce qui est plus naturel. Puisqu' il est composé de lettres). Les informaticiens utilisent les termes de *chaîne*, voir de *liste* dans certains cas. Dans la suite les termes de chaîne et de phrase seront utilisés indifféremment.

DÉFINITION 4 :

La phrase vide notée λ est la suite de lettres de longueur nulle

Notation :

On note par convention la phrase :

$$x = \underbrace{a a a \dots a a a}_{n \text{ fois}} \quad \text{par } a^n;$$

DÉFINITION 5 :

L'ensemble des phrases sur X est noté X^* ; L'ensemble des phrases non vides X^+ .

DÉFINITION 6 :

X^* possède l'opération interne associative de concaténation, définie par :

$$\text{Soit } x, y \in X^* \text{ avec } x = x_1 \dots x_n.$$

$$y = y_1 \dots y_m.$$

$$\text{Alors } z = xy = x_1 \dots x_n y_1 \dots y_m \in X^*.$$

Cette opération est en général non commutatif; et possède λ pour élément neutre.

DÉFINITION 7 :

$y \in X^*$ est dite sous phrase de la phrase $x \in X^*$ quand il existe dans X^* deux phrases μ et ν telles que : $x = \mu\nu$;

μ : appelé préfixe.

ν : appelé suffixe.

EXEMPLES :

-1. Une ligne dans une image peut être approchée de façon discrète par une suite de vecteurs de taille élémentaire et de direction choisie dans un ensemble fini. Un exemple en est le codage de FREEMAN.

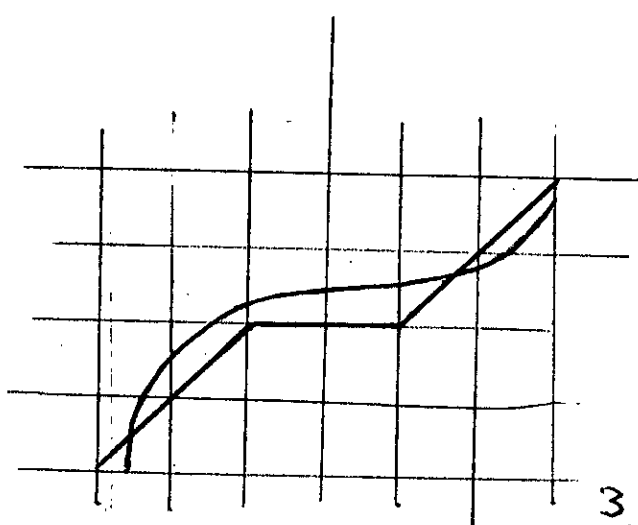
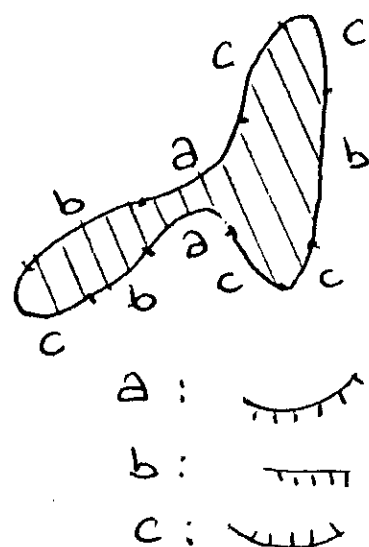
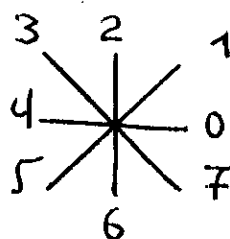


Fig IV. II



La courbe de la figure IV.11 est représenté sur l'alphabet de FREEMAN $X = \{ 0, 1, 2, \dots, 7 \}$ par la phrase $x = 110011$.

-2. A un niveau plus élevé de prétraitement, la frontière d'un objet isolé, dans une image peut être représentée par une succession de parties concaves (notées a), de parties convexes (notées c) et de parties à peu près plates (notées b).

Ici on se libère de certaines contraintes de taille, puisqu'une lettre ne correspond pas toujours à la même longueur de frontière.

IV.4 STRATÉGIE DE CODAGE ADOPTE :

Comme on a vu précédement, l'image en général et la squelette en particulier se présente sous forme d'une matrice image $I(M,N)$, chaque point $P(i,j)$ de cette matrice à 4 voisins horizontaux et verticaux et 4 voisins diagonaux.

Schématiquement, la position de $p(i,j)$ par rapport à ses voisins est

comme suit :

$$\begin{array}{ccc} P(i-1, j-1) & P(i, j-1) & P(i+1, j-1) \\ P(i-1, j) & P(i, j) & P(i+1, j) \\ P(i-1, j+1) & P(i, j+1) & P(i+1, j+1) \end{array}$$

Cette position particulière et régulière de n'importe quelle point $P(i, j)$ appartenant à la matrice image nous a amener à adopter la méthode, d'approximation angulaire de FREEMAN vu précédemment.

Les paragraphes qui suivent permet d'avoir une idée sur l'application de cette technique de codage et les modifications qu'on a apporter pour l'adapter à notre application.

Soit un caractère manuscrit prétraité et stocké, dans une matrice de pixel, (voir exemple à la fin de ce chapitre).

Pour la coder il faut suivre les étapes suivantes:-

Etape 1 : Balayage horizontal de la matrice de gauche à droite (sens conventionnelle).

Etape 2 : Arrêt du balayage après la détection du premier pixel allumé. Mémorisation des coordonnées de ce point comme étant point noeud et point courant en même temps.
On marque le début de cette branche ou groupe de pixel par l'insertion d'un caractère spécial \$ dans une chaîne code.
code := code + '\$';

Etape 3 : Scrutation des voisins de ce pixel suivant la convention de direction suivant:

Etape 4 : Mémorisation des coordonnées de chaque pixel que l'on trouve allumé lors de la scrutation dans une chaîne conçue pour cela ; tout en incrémentant un compteur comptabilisant le nombre de pixels allumés

Etape 5 :

Test de l'état du compteur :

rs'il est égal à 1 alors faire sortir les coordonnées du point allumé a partir de la chaîne; aller vers ce point

après avoir éteints le pixel courant (pour éviter le risque de le prendre en compte une autre fois).

-S'il est supérieure à 1 : ceci implique que nous sommes devant une dérivation ; On marque ceci par l'insertion d'un caractère spécial '#'.
code := code + '#'.
Sous-écriture de coordonnées du pixel courant dans une pile de structure FIFO , éteindre le pixel courant.

Aller vers le noeud courant et refaire 3 - 4 - 5.

-S'il est nulle; alors tester l'état de la pile

-Si elle n'est pas vide alors :

Eteindre le point courant. Dépiler la pile et changer le point noeud courant et le point courant avec son contenu et refaire 3 - 4 - 5.

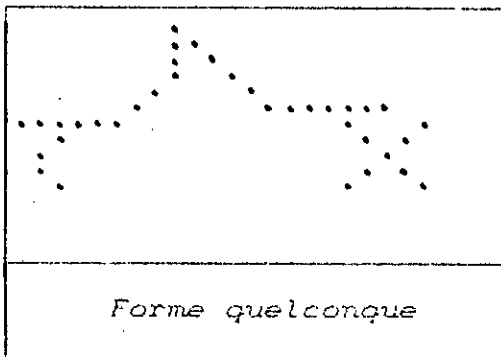
-Si elle est vide alors le codage de cette branche est terminé.

ANALYSE ET EXTRACTION DES PARAMETRES

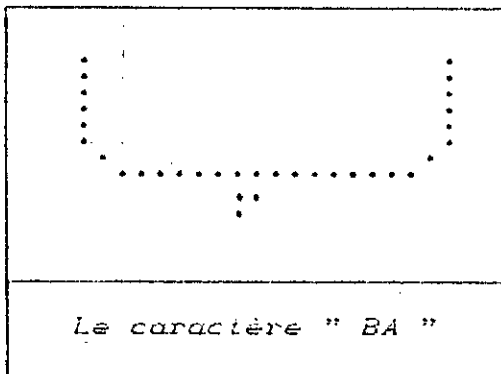
Aller a l'étape 1 pour d'autre éventuelle branche.

Remarque : Pour une description plus complète de ces étapes voir l'organigramme correspondant .

EXEMPLE :



Le code correspondant est le suivant :
\$#e#d#eef#ddddccc##fffgg#ddca#e#d##f#
gggncndd#fedn#ffn#bbnddn



Le code correspondant est le suivant :
\$eeeeedddccccccccccccccccbbbaaaan\$#cen

CHAPITRE V

RECONNAISSANCE

A P P R E N T I S S A G E E T D E C I S I O N

V.1 INTRODUCTION :

La reconnaissance regroupe les deux tâches d'apprentissage et de décision qui jouent des rôles assez proches, dans les systèmes de reconnaissance des formes.

En effet, à partir de la même description de la forme en paramètres, elles tentent, toutes les deux, d'attribuer cette forme à un modèle de référence. Le résultat de l'apprentissage est, soit le renforcement des modèles existants en tenant compte de l'apport de la nouvelle forme, soit la création d'un nouveau modèle représentant la forme entrée. Le résultat de la décision est un " oui " sur l'appartenance de la forme aux modèles de l'apprentissage. Nous allons revenir dans la suite sur la fonction de ces tâches en développant quelques approches classiques qui les réalisent.

V.2 L'APPRENTISSAGE :

Il s'agit, lors de cette étape d'apprendre au système les propriétés pertinentes des caractères du vocabulaire utilisé. L'idéal serait de lui apprendre autant d'échantillons que de forme d'écriture différente, mais cela est impossible à cause de la trop grande variabilité de l'écriture qui conduirait à une combinaison explosive des modèles de représentation. La tendance consiste alors à remplacer le nombre par une meilleure qualité des traits caractéristiques et de bonne réparabilité des classes d'apprentissage.

Nous donnons dans la suite quelque exemples de procédés d'apprentissage.

V.2.1 L'APPRENTISSAGE MANUEL:

Il est réalisé lors d'une étape préliminaire à la reconnaissance, en introduisant un grand nombre, d'échantillons de référence. L'utilisateur joue le rôle de professeur pour indiquer le nombre de chaque échantillon. Le choix des caractères de référence est fait à la main en fonction de l'application (échantillon le plus représentatif possible de la typographie des caractères de l'application). Le nombre d'échantillons peuvent varier de quelques unités à une dizaine par caractère.

Dans le cas spécial qui nous occupe (apprentissage de caractères manuscrits), ce choix est délicat à cause de la variabilité de l'écriture et des scripteurs. Il n'est pas souvent facile d'exprimer le

APPRENTISSAGE ET DECISION

représentativité des échantillons sélectionnés.

V.2.2 L'APPRENTISSAGE AUTOMATIQUE :

Ce deuxième type d'apprentissage consiste à doter le système d'un mécanisme automatique lui permettant de trouver les classes de référence avec une assistance minimale. Les échantillons sont introduits en grand nombre par l'utilisateur sans indication sur leur classe. Ce type d'apprentissage est intéressant car il permet de renseigner l'utilisateur sur les ambiguïtés entre les caractères afin d'agir en conséquence, par exemple en ajoutant des échantillons pour renforcer la représentativité d'une classe.

V.2.3 L'APPRENTISSAGE CONTINU :

Afin de pouvoir s'adapter à un nouveau texte, le système peut compléter ou enrichir sa base de données (d'apprentissage) en s'ajoutant des échantillons extraits du texte étudié. L'apprentissage se fait ainsi de manière continue au fur et à mesure de la reconnaissance chaque fois que le système bute sur la reconnaissance d'un nouveau caractère. Le nom du caractère non reconnu peut lui être communiqué par l'utilisateur.

V.3 TECHNIQUE D'APPRENTISSAGE ADOPTE :

APPRENTISSAGE ET DECISION

Pour construire notre base de données on a procédé ainsi :

Dans une première étape l'apprentissage était manuel. En effet, chaque classe était assigné à un fichier, et d'une manière manuel on introduit, dans chaque fichier le code de quelques échantillons, qu'on pense suffisamment représentatif de la classe correspond.

Après ceci; on a doté notre système d'une technique d'apprentissage continu; Ceci, afin d'enrichir vraiment notre base : en effet lors de l'utilisation de notre système, chaque fois qu'il bute sur la reconnaissance d'un caractère inconnu une possibilité d'introduire son code dans la base est offert; dans le cas du caractère non reconnu le nom peut lui être communiqué par l'opérateur.

Ainsi au fur et à mesure de l'utilisation de notre système, sa base se trouve complète de plus en plus.

V.4 LA DÉCISION :

La décision est l'ultime étape de reconnaissance, à partir d'une description en paramètres, elle recherche, parmi les modèles d'apprentissage en présence, ceux qui sont les plus " proches ". La notion de proximité à un sens différent en fonction de la nature de la représentation et du type de la méthode. La décision peut conduire à un succès si la réponse est unique (un seul modèle répond à la description de la forme). Elle peut conduire à une confusion si la réponse est multiple (plusieurs modèles correspondants à la description). Enfin, la

décision peut conduire à un rejet de la forme si aucun modèle ne correspond à sa description. Dans les deux premiers cas, la décision peut être accompagnée d'une mesure de vraisemblance appelée aussi score ou taux de reconnaissance.

Les méthodologies de reconnaissance sont nombreuses. On va parler, dans la suite, des deux méthodologies les plus utilisées en reconnaissance de forme .

V.4.1 APPROCHE STATISTIQUE :

L'approche statistique est une approche classique en reconnaissance des formes; elle est fondée sur l'étude statistique des mesures que l'on a effectué sur les objets à reconnaître. L'étude de leur répartition dans un espace métrique et la caractérisation statistique des classes permet de prendre une décision de reconnaissance de type " *plus forte probabilité d'appartenance à une classe* ". Ces méthodes s'appuient en général sur des hypothèses concernant la description statistique des familles d'objets analogues dans l'espace de représentation.

V.4.2 APPROCHE STRUCTURELLE :

Elle est fondée sur la structure propre du caractère. La structure est exprimée en terme de composant primitifs correspondant à des formes

es élémentaires du tracé.

Parmi ces méthodes structurelles, Nous trouvons :

V.4.2.a) LES MÉTHODES DE TESTS :

Elles consistent à appliquer sur chaque caractère entré des tests de plus en plus fins sur la présence ou l'absence de certain primitives, de manière à répartir les échantillons en catégories ou en classes.

V.4.2.b) LES MÉTHODES DE COMPARAISONS DE CHAINES :

Les caractères sont représentés par des chaînes de primitives. La comparaison du caractère test avec un modèle de référence consiste à mesurer la ressemblance entre deux chaînes et à se prononcer sur celle-ci, la mesure de ressemblance peut se faire par calcul de distance ou par examen de l'inclusion de tout ou partie d'une chaîne dans une autre.

V.4.2.c) LES MÉTHODES SYNTAXIQUES :

Chaque caractère est représenté par une phrase dans un langage où le vocabulaire est constitué des primitives. Les chaînes d'une même famille sont représentées par une grammaire. La reconnaissance consiste à déterminer si la phrase de description du caractère peut être générée par la grammaire.

V.5 DEVELOPEMENT SUR LA METHODE DE COMPARAISON DE CHAINE :

PROGRAMMATION DYNAMIQUE :

La programmation dynamique est utilisée en reconnaissance des formes pour chercher la ressemblance entre deux chaînes par le calcul d'une distance. Avant de détailler cet aspect, nous allons d'abord évoqué le principe général de la programmation dynamique avec un exemple introductive.

PRINCIPE GENERALE :

La programmation dynamique est une méthode de recherche d'optimum fondée sur le principe d'optimalité locale de BELLMAN. "Dans une séquence optimale de décision, quelle que soit la première décision prise, les décisions subséquentes forment une sous séquence optimale, compte tenu des résultats de la première décision". Ce principe indique que tout chemin optimal est constitué de portions de chemin elles mêmes optimales. En effet, si on considère un chemin optimal reliant A à B et un point quelconque M de ce chemin, les chemins de A à M et de M à B sont tous deux optimaux.

EXEMPLE INTRODUCTIVE :

On désire construire une route entre deux villes A et L. Les chemins possibles sont représentées, dans le graphe de la figure V.1. Chaque chemin est accompagné d'un coût noté $C(x_i, x_{i+1})$ qui correspond au coût de la construction d'une route entre les villes x_i, x_{i+1} . Ce coût est nulle lorsque la route existe déjà.

On cherche le chemin optimal joignant A et L c'est à dire celui qui minimise le coût de la construction.

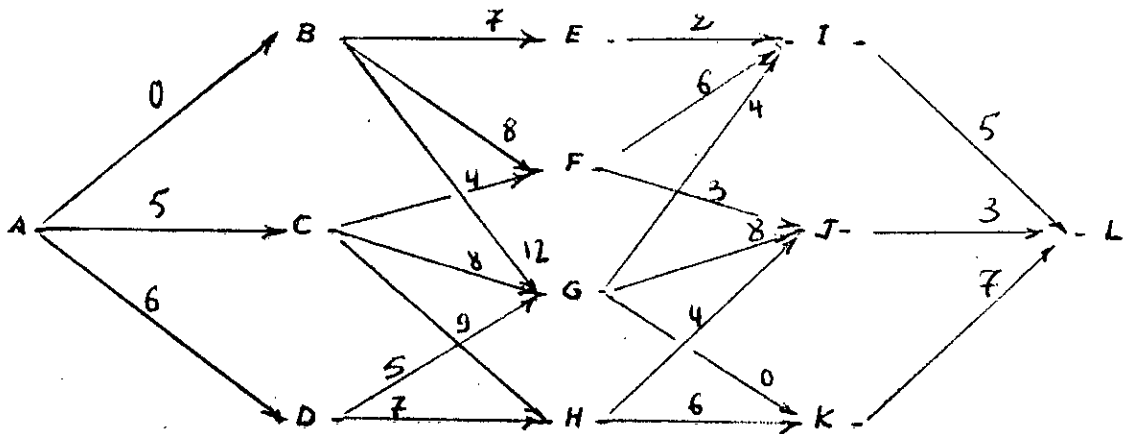


Fig V.1

Le problème consiste donc à chercher les villes intermédiaires x_1, x_2, x_3 qui minimisent :

α

$$C(A, L) = C(A, x_1) + C(x_1, x_2) + C(x_2, x_3)$$

avec

$$x_1 \in \{ B, C, D \} = E_1,$$

$$x_2 \in \{ E, F, G, H \} = E_2,$$

$$x_3 \in \{ I, J, K \} = E_3;$$

On calcule:

$$C(A, x_2) = \min_{x_1 \in E_1} [C(A, x_1) + C(x_1, x_2)], \forall x_2 \in E_2 \text{ puis}$$

$$C(A, x_3) = \min_{x_2 \in E_2} [C(A, x_2) + C(x_2, x_3)], \forall x_3 \in E_3 \text{ et on déduit}$$

$$C(A, L) = \min_{x_3 \in E_3} [C(A, x_3) + C(x_3, L)];$$

Les calculs sont résumés dans le tableau ci-dessous, dans lequel $C(A, x_i)$ est remplacé par $C(x_i)$ Car l'origine A est la même pour tout les chemins, chaque coût est suivi du chemin correspondant :

| | | | |
|---------------|-----------------|------------------|-------------|
| $C(B) = 0$ AB | $C(E) = 7$ ABE | $C(I) = 9$ ABET | $C(L) = 14$ |
| $C(C) = 5$ AC | $C(F) = 8$ ABF | $C(J) = 11$ ABFJ | |
| $C(D) = 6$ AD | $C(G) = 11$ ADG | $C(K) = 11$ ADGK | |

| |
|----------------------------|
| $C (H) = 13 \text{ ADH}$ |
|----------------------------|

Le coût minimal est $\llbracket 14 \rrbracket$ et il existe deux chemins possibles "ABE-IL", et "ABEJL";

V.6 DISTANCE ENTRE CHAINES (ALGORITHME DE WIGNER ET FISHER) :

La distance entre chaînes peut être calculée par minimisation de distances locales entre les composantes des chaînes et optimisation donc de la ressemblance.

Si l'on prend l'exemple de deux chaînes de caractère { abc } et { abde }. On s'aperçoit que la comparaison peut se faire en termes de transformations pouvant faire passer d'une chaîne à l'autre. Par exemple, pour passer de la première chaîne à la deuxième, il faut conserver les deux premiers caractères " ab ", changer la troisième, remplacer la lettre " c " par la lettre " d " et ajouter un e. On remarque que ces opérations peuvent être effectuées à l'aide des trois transformations élémentaires suivantes :

- la substitution :

$$a \longrightarrow b \quad \text{coût } \gamma (a, b)$$

- l'insertion :

$$\lambda \longrightarrow a \quad \text{coût } \gamma (\lambda, a)$$

- la destruction :

$$a \longrightarrow \lambda \quad \text{coût} : (a, \lambda)$$

où λ est un caractère vide.

Ainsi, la phrase $x = abc$ peut être transformé en $y = abde$ par la suite d'opérations élémentaires :

$$(a \longrightarrow a),$$

$$(b \longrightarrow b),$$

$$(c \longrightarrow d),$$

$$(\lambda \longrightarrow e).$$

Cette transformation totale a un coût de :

$$r(c, d) + r(\lambda, e)$$

car $r(a, a) = r(b, b) = 0$;

On définit la distance d'édition $\delta(x, y)$ entre deux phrases x et y comme le coût de la suite de transformations élémentaires la moins coûteuse pour transformer x en y (Cette distance est connue en codage sous le nom de " distance de LEVENSTEIN ").

Pour calculer $\delta(x, y)$ en évitant une inutile explosion combinatoire-

e, il faut se donner une contrainte naturelle :

$$\delta(a, b) = \gamma(a, b)$$

Autrement dit, la façon la moins chère de transformer la chaîne 'a' en la chaîne 'b' doit être simplement de changer la lettre a en la lettre

b. Ceci est équivalent à l'inégalité triangulaire sur γ :

$$\gamma(a, b) + \gamma(b, c) \geq \gamma(a, c)$$

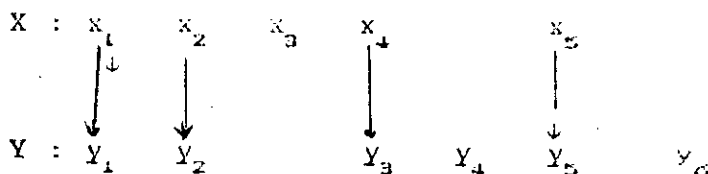
La condition imposée est donc que la fonction γ soit fondamentalement une vraie distance. Le schéma de transformation, appelé trace, reproduit par des flèches les transformations concernées.

Soit

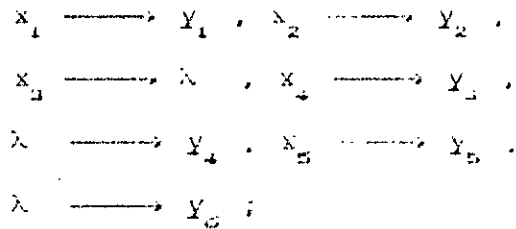
$$X = x_1 x_2 x_3 x_4 x_5 ;$$

$$Y = y_1 y_2 y_3 y_4 y_5 y_6 ;$$

Supposons que la trace de transformation de x en y soit le suivant:



Elle peut donc s'écrire comme :



Notons : $x = a_1 \dots a_n$
 $y = b_1 \dots b_n$

$$\begin{array}{l}
 x(i) = a_1 \dots a_i \\
 y(j) = b_1 \dots b_j \\
 D(i,j) = \gamma(x(i), y(j))
 \end{array}$$

On a alors :

$$D(i,j) = \text{Min} \begin{cases} D(i-1, j-1) + \gamma(a_i, b_j) ; \\ D(i-1, j) + \gamma(a_i, \lambda^j) ; \\ D(i, j-1) + \gamma(\lambda^i, b_j) ; \end{cases}$$

L'additivité du critère de coût par rapport aux coûts élémentaires assurent que l'on peut alors utiliser une méthode de programmation dynamique, traduite par la formule de récurrence indiquée. L'algorithme de calcul de γ est donc :

Début

$$D(0,0) = 0$$

APPRENTISSAGE ET DECISION

Pour i de 1 à n faire

$$D(i,0) \leftarrow D(i-1,0) + \gamma(a_i, \lambda)$$

Ppour

Pour j de 1 à n faire

$$D(i,0) \leftarrow D(0, j-1) + \gamma(\lambda, b_j)$$

~~~~~Fpour

Pour i de 1 à n faire

Pour j de 1 à n faire

$$m_1 \leftarrow D(i-1, j-1) + \gamma(a_i, b_j)$$

$$m_2 \leftarrow D(i-1, j) + \gamma(a_i, \lambda)$$

$$m_3 \leftarrow D(i, j-1) + \gamma(\lambda, b_j)$$

$$D(i, j) \leftarrow \text{Min}(m_1, m_2, m_3)$$

Distance d'edition  $\delta(x,y) = D(n,m)$

Exemple :

$X = \{a, b, c, d\}$

$x = aabacd$

$y = abd$

$\gamma(\alpha, \beta) = 1$  pour tout  $\alpha, \beta \in X \cup \{\lambda\}$  et  $\alpha \neq \beta$

$\gamma(\alpha, \alpha) = 0$  pour tout  $\alpha \in \{x\}$



Tableau D(i,j)

|           | $\lambda$ | a | a | b | a | c | d |
|-----------|-----------|---|---|---|---|---|---|
| $\lambda$ | 0         | 1 | 2 | 3 | 4 | 5 | 6 |
| a         | 1         | 0 | 1 | 2 | 3 | 4 | 5 |
| b         | 2         | 1 | 1 | 1 | 2 | 3 | 4 |
| d         | 3         | 2 | 2 | 2 | 2 | 3 | 3 |

trace correspondant à une distance de 3.

a    a    b    a    c    d  
 ↓        ↓        ↓  
 a        b        d

V.7 PROCÉDE DE RECONNAISSANCE :

Pour comparer le caractère inconnu avec les éléments de la base de données en vue d'une reconnaissance, on a procédé ainsi :

comparer le caractère inconnu avec tout les éléments de la même classe; le coût minimal de comparaison est mémorisé; La décision d'appartenance du caractère à une telle ou telle classe ne sera prise que lorsque toutes les classes seront sollicités; finalement nous disons que le caractère appartient à la classe ayant un coût minimal de comparaison.

Cette technique est très optimale du point de vue précision; mais cette optimalité crée un problème de temps de reconnaissance - surtout si la base de données est grande - qui est principalement due au fait que la lecture de la base s'effectue dans le disque.

Pour résoudre ce problème, nous proposons les deux solutions suivantes

a) PREMIERE SOLUTION :

Cette solution consiste à créer un disque virtuel avec la commande externe de MS-DOS : *VDISC* ou l'équivalent de la version utilisée, et l'introduction dans l'*AUTOEXEC.BAT* d'un ensemble d'instructions permettant le chargement automatique de la base de données dans ce disque virtuel au début de chaque session de travail.

Ceci va permettre de diminuer considérablement le temps de reconnaissance car le système n'a plus besoin d'aller vers les unités disques pour accéder aux fichiers de la base; ces derniers seront maintenant disponibles dans la *RAM* qui a un temps d'accès énormément court.

Cette solution est contrainte seulement par la capacité de la *RAM* qu'occupe notre micro.

un autre point aussi important que ce qui précède est le suivant : comme nos programmes compte plus que sept ( 7 ) unités il est donc intéressant d'en profiter de la manière suivante : sachant que le compilateur *TURBO PASCAL* reprend une unité dans un programme lorsqu'il rencontre la directive *USES* de la manière suivante : il cherche l'unité dans le fichier *TURBO.TPL*, s'il ne l'y trouve pas, il cherche le fichier *TPU* correspondant sur la disquette ou le disque dur; Or le fichier *TURBO.TPL* figure en permanence en mémoire vive où la recherche est extrêmement rapide; par contre la recherche d'un fichier sur la disquette ou le disque dur prend évidemment plus de temps; il serait donc avantageux d'enregistrer dans le *TURBO.TPL* tous nos unités et ceci peut être effectué à l'aide du *TPUNOVER.EXE* de *BORLAND* qui est justement fait pour ça.

Ainsi avec ces deux techniques on peut estimer que le temps de reconnaissance va être considérablement optimisé.

#### B) DEUXIEME SOLUTION :

Cette solution aura un sens seulement si la base de données est suffisamment riche et que les éléments qu'elle contient soient suffisamment représentatives.

Elle consiste à calculer pour chaque classe la distance max qui peut exister entre deux de ces éléments, ainsi on affecte à chaque classe un seuil caractéristique.

Théoriquement, un caractère à reconnaître appartient à une telle ou telle classe si la distance entre cet élément et un élément quelconque de la classe est nulle. Démarrons de cette idée, un caractère appartient à une classe si la distance entre ce caractère et tous ces éléments est inférieur ou égal à son seuil caractéristique.

Ainsi, si la distance entre le caractère inconnu et un des éléments de la classe au cours de son sollicitation s'avère supérieur à son seuil, cette comparaison est arrêtée et la classe est ainsi rejetée. (C'est à dire que le caractère à reconnaître ne peut appartenir à cette classe).

Trois cas se présente :-

- 1). Si tous les classes sont rejetées  $\implies$  CARACTERE NON RECONNU.
- 2). Si plus qu'une classe est acceptée  $\implies$  CARACTERE PRESENTANT UNE AMBIGUITE.
- 3). Si une classe et une seule est acceptée  $\implies$  CARACTERE RECONNU.

REMARQUE :

Pour tenir compte de la grande variabilité des formes des caractères manuscrits et du fait que la représentativité d'un élément ne peut être estimée avec exactitude, il est donc impératif de prendre une

certaine marge dans le choix des seuils précédents.

L'amplitude de cette marge dépendra bien sûr de la tolérance à la mauvaise impression acceptée par le système.

zzzzz

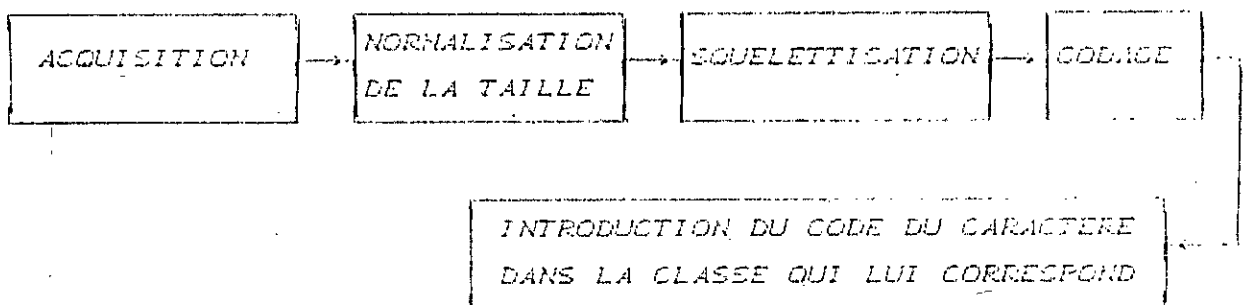
*CONCLUSION GENERAL*

CONCLUSION GENERALE :

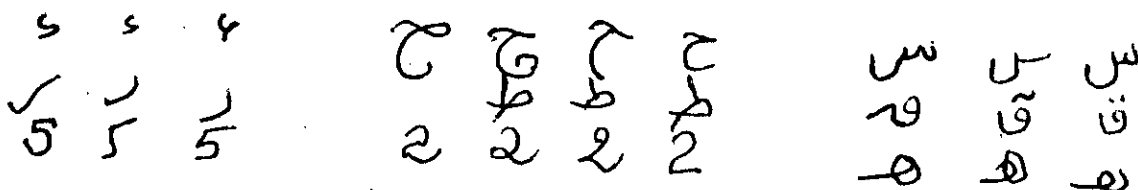
Le but que nous nous sommes fixé dans cette étude est de concevoir et réaliser du logiciel pour la reconnaissance des caractères arabes manuscrits, évidemment ceci impliquera que la reconnaissance des caractères arabes dactylographiés serait acquise de même coup. Pour cela on a été amené à parcourir la chaîne de reconnaissance des formes de bout à bout : de l'acquisition vers la décision et l'apprentissage en passant, bien sur, par le prétraitement, l'analyse et l'extraction des paramètres; ceci nous pensant, nous a permis de voir les principales techniques de traitement d'image et de la reconnaissance de l'écriture existantes et les manipuler.

Notre application s'inscrit dans la méthodologie structurale à apprentissage continu; en effet notre système est structurel dans sa représentation, son apprentissage et sa décision; le caractère continu de son apprentissage nous permettra de s'adapter facilement à d'autre texte, affiner et compléter notre base.

Lors de la construction de la base; notre apprentissage était manuel, et la chaîne parcourue était la suivante :

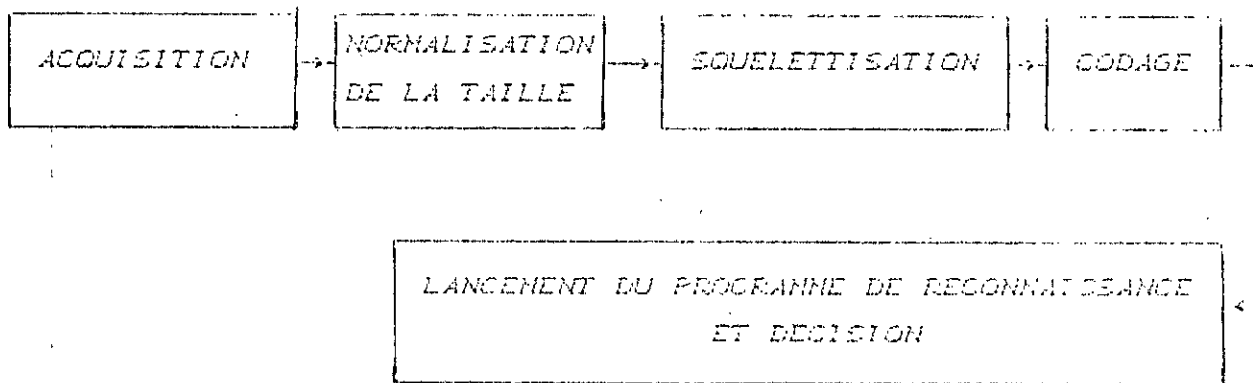


Ici, on s'est limité, faute de manque de moyens, à l'acquisition de 32 caractères en plus des 10 chiffres, en voici un extrait d'eux :



Il est à noter que la seule demande aux différents scripteurs était simplement d'écrire lisiblement.

Lors de la reconnaissance la chaîne parcourue est la suivante :



Le problème essentiel que nous avons rencontré lors de la reconnaissance était dû au fait que notre système ne tient pas compte de la différence de points entre différents caractères par exemple il mélange entre le ... etc, c'est à dire que sa comparaison repose principalement sur le corps principal du caractère.

La résolution de ce problème était pour nous trivial; en effet, il faudrait tout simplement donner un poids plus fort au point lors du



codage ce qui s'obtient par le remplacement du sigle '\$' par '\$\$' ou même par '\$\$\$', réaliser ceci est évidemment simple, mais on sera obligé de refaire toute la base de données, ceci a exclu radicalement cette solution ( problème de temps ). En pensant on a pu trouver une idée équivalente a la précédente du point de vu résultat avec le seul défaut qu'elle augmente légèrement le temps de reconnaissance, cette idée consiste principalement en une intervention extérieure sur la distance entre caractère, pour plus de détail voir la différence entre l'algorithme de W & F et son implémentation.

Les résultats finales obtenues avec notre système a l'état actuelle sont largement satisfaisants ceci témoigne que le processus que nous avons adopté pour notre démarche est très efficace et adéquat aux problèmes posés.

On dit ceci et on rappelle que le nombre d'échantillons par classe est extrêmement maigre; il varie entre 3 et 5.

Ce qui précède nous permis d'affirmer avec certitude qu'en améliorant la taille de notre base ( ce qui exige un matériel d'acquisition spécialisé ) on aura un appareil micro informatique industrielle pour la reconnaissance des caractères et chiffres de notre langue.

Des améliorations et extensions du présente étude sont toujours possibles, nous proposons les directions suivantes :

- 1- Bien sur, et avant tous, enrichir la base et son extension à tout les caractères arabes ( isolé, début, milieu, fin ).
- 2- Optimiser le temps de reconnaissance en vue d'une reconnaissance en temps réel par l'application des procédés déjà cités et l'utilisation, s'il est possible, d'un micro plus puissant en terme de la fréquence de son horloge.

# ANNEXE II

## BIBLIOGRAPHIE

*bibliographie:*

---

- [1]. A.Belaid & Y.Belaid ; " RECONNAISSANCE DES FORMES : METHODES ET APPLICATION " ; Inter\_edition , Paris , 1992 .
- [2]. A.Amin ; " CHARACTER RECOGNITION " ; AFCET/IAPR , eighth international conference on pattern recognition ; oct 27-31 1986 .
- [3]. F.Ait boudaoud & M.Belhandouz ; " METHODE DES ARBRES POUR LA R.F APPLIQUEES AUX CARACTERES DE L'ALPHABET ARABES " ; PFE ENP juillet et 1992 ;
- [4]. G.Rosenbaum & R.Scholles ; " LE GRAND LIVRE DE TURBO PASCAL " Micro-edition 1989 .
- [5]. L.Miclet ; " METHODES STRUCTURELLES POUR LA R.F " ; Edition cyrolles 1984 .
- [6]. L.Saadaoui ; " TECHNIQUES DE TRAITEMENT NUMERIQUES D'IMAGE EN VUE DE LA R.F. " ; these de magister ; ENP , avril 1991 .
- [7]. L.Saadaoui & M.Latreche ; " TRAITEMENT ET ACQUISITION D'IMAGE: APPLICATION A L'EXTRACTION DES CARACTERISTIQUES DES IMAGES " PFE ENP , juin 1987 .
- [8]. Merrouche et Abdou ; " ETUDE DE LA METHODE DES MOMENTS POUR LA R.F APPLIQUEE AUX CARACTERES ARABES " ; PFE ENP 1990 .

- [9]. O.Abdelghani & M.Abdelkrim ; " RECONNAISSANCE DES FORMES : APPLICATION POUR LA RECONNAISSANCE STRUCTURELLE DES CARACTERES MANUSCRITS MAJUSCULES " PFE UNIV ANNABA sep 1991 .
- [10]. R.C.Gonzalez & P.Wintz " DIGITAL IMAGE PROCESSING " ;Addison-wesley publishing company , 1977 .
- [11]. R.C.Gonzalez & M.Thomason ; "SYNTACTIC PATTERN RECOGNITION : AN INTRODUCTION " ; addison-wesley publishing company ; 1982 .
- [12]. W.K.Pratt ; " DIGITAL IMAGE PROCESSING " ; korean student edition 1978 .

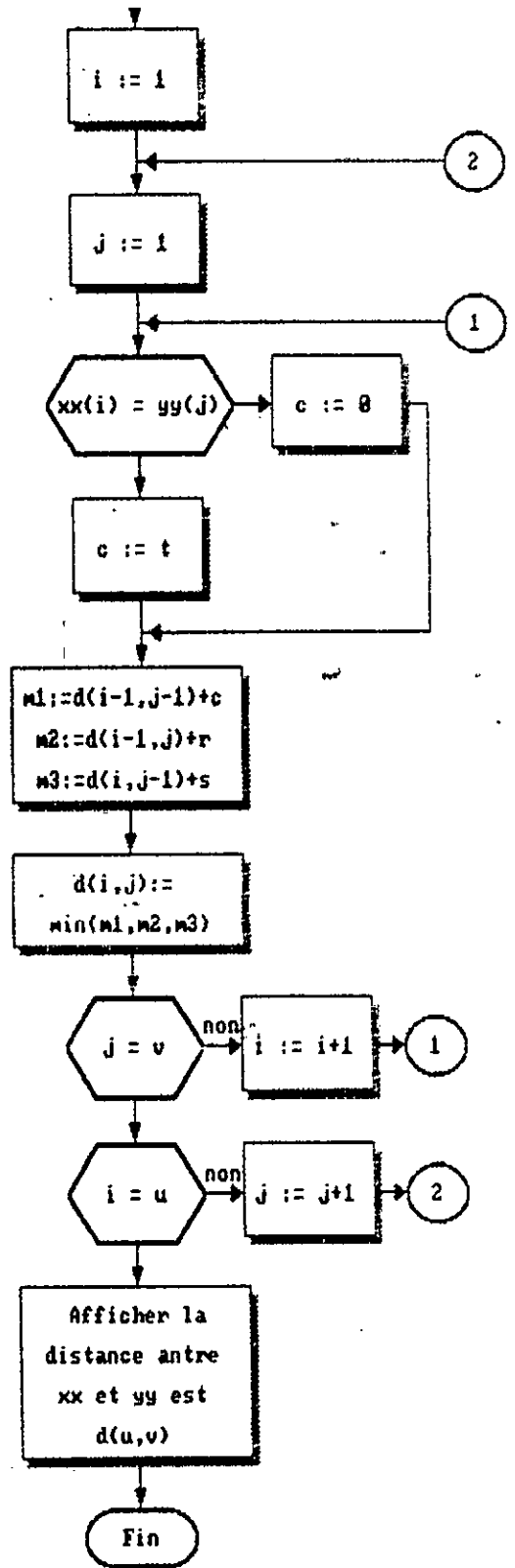
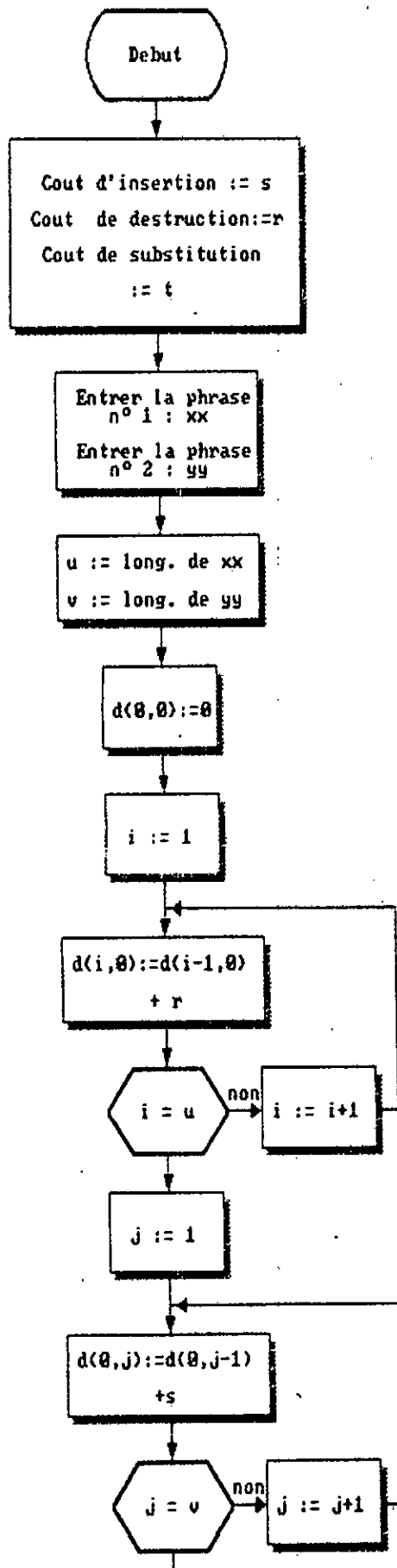


Fig 1 : Algorithme de Wagner & Fisher.

Fig : Organigram. de l'Algorithme de Codage.

