Ecole Nationale Polytechnique

Direction de l'Enseignement et de la Recherche
Génie Electrique et Informatique

Département d'Electronique



# THESE

en vue de l'obtention

du GRADE de MAGISTER en ELECTRONIQUE

Option : ACQUISITION et TRAITEMENT de l' INFORMATION

Theme

# Segmentation d'un Paragraphe de Texte Arabe Imprimé en Caractères

Par :

**Mme ZAÏT Malika Eps IDDIR**

Soutenue devant la commission d'examen

Président : Mr **A. FARAH** ..... Maître de conférence
Rapporteur : Mme **L. HAMAMI** ..... Chargée de cours
Examinateurs : Mlle **C. GUERTI** .....Maître de conférence
Mr D. **BERKANI** ..... Maître de conférence
Mr L. **SAADAOUI** ..... Chargé de cours

1997

# REMERCIEMENTS

☞ C'est à travers ces quelques mots que j'adresse mes remerciements à tous ceux qui ont fait qu'aujourd'hui je sois au bout de mon chemin, aussi long et difficile était il,et qui s'est soldé par la réalisation de cette modeste recherche. Un aboutissement qui est le fruit de nombreux encouragements reçus d'un entourage qui sans lui je ne serais peut être pas là aujourd'hui. Fruit aussi d'une formation qui m'a été très bénéfique.

☞ Mes remerciements vont à Madame L.Hamami, ma promotrice pour m'avoir confié ce travail, pour la grande confiance qu'elle a placé en moi, pour toute son aide morale, pour ses conseils ainsi que pour toute la documentation qu'elle a mis à ma disposition.

☞ Je remercie particulièrement mon mari tant pour ces encouragements que pour sa participation effective pour la réalisation de ce document.

☞ Je remercie également les membres du jury qui ont accepté de juger mon travail.

☞ Sans oublier mes amis et collègues qui ont montré un grand intérêt à mon travail et qui m'ont poussé à toujours aller de l'avant.

***Résumé :***

L'objectif de notre travail est principalement l'étude de la partie médiane d'un système de reconnaissance automatique de l'écriture arabe imprimée ou dactylographiée. Il s'agit du module de segmentation, qui reçoit en entrée un paragraphe de texte, et qui fournit en sortie des caractères séparés destinés au module de reconnaissance. Les méthodes étudiées (au nombre de deux) sont basées sur l'observation des histogrammes des lignes et des colonnes, et tiennent compte des caractéristiques propres de l'écriture arabe. La segmentation est effectuée en trois étapes, par localisation des lignes, des parties connexes et enfin des caractères. Constatant la présence de bruit dans les images étudiées, nous avons complété notre travail par l'étude du module de filtrage.
Les résultats obtenus nous ont permis de juger de la performance d'une méthode par rapport à l'autre.

---

ملخص

إن هدف دراستنا هو تقطيع ورقة تحتوي على نص مطبوع باللغة العربية إلى حروف . لهذا وبما أن الحروف العربية ترتبط ببعضها ، فقد قمنا بهذه العملية على ثلاث مراحل .
ففي المرحلة الأولى يتم التقطيع الأفقي و الحصول على السطور المكتوبة . في المرحلة الثانية يتم التقطيع الشاقولي و الحصول على "مقاطع حرفية " (وهي تسلسل حروف مرتبطة ببعضها البعض) وفي المرحلة الأخيرة يتم تقطيع المقاطع الحرفية إلى حروف منعزلة و لقد قمنا بهذه العملية بطريقتين مختلفتين لمقارنتهما.
و بالمقابل قمنا بدراسة عدد من المرشحات لأجل تقليص الإضطرابات التي تعرض لها النص خلال عملية المعالجة.

---

***Abstact:***

The aim of our work is a study of median part of AOCR (Arabic Optical Caracter recognition) system. In this work, we presented two methods of arabic text image segmentation, based on the use of histograms. These methods consider the caracteristics of arabic write, and have been effected in three stages. They permit first to locate the text lines, second to separate sub-word in the line. The third stage, the more importante, consists in separating the caracters in the sub-word. Finally, to reduce the noise and improve the rate of segmentation, we have completed our study by proposing a number of filters. The results obtained are different from method to other, and permitted us to compare performance of each one.

# INTRODUCTION GENERALE

# INTRODUCTION GENERALE



La reconnaissance de l'écriture, élargie depuis quelques années à l'analyse des documents composites devient de jour en jour un thème de recherche à part entière. Il ne s'agit plus de considérer le caractère comme la seule composante de l'image, mais il faut reconnaître la structure du document, séparer les parties textuelles des graphiques et photos, identifier les caractéristiques typographiques et reconnaître les caractères dans les textes, coder et interpréter les graphes, etc...

L'intérêt qu'a suscité le sujet de reconnaissance est justifié par ses nombreuses applications dont l'archivage de documents, la saisie automatisée en entreprise, la lecture automatique de chèques, etc... [1]

L'OCR, acronyme de la reconnaissance optique de caractères (Optical Character Recognition) consiste à transformer un document de texte se trouvant sur un support papier, en une représentation compréhensible par l'ordinateur, c'est une interface permettant de faciliter la communication homme - machine. C'est aussi un problème d'identification d'une forme donnée en l'affectant à une classe obtenue par apprentissage[2].

Trois familles de système OCR se distinguent par les propriétés du caractères pouvant être reconnu.

Les systèmes mono - fonte [2] reconnaissent les caractères d'une fonte bien déterminée, une fonte est caractérisée par un style, une taille, la graisse des lettres et la pente de l'écriture. Ces systèmes présentent l'inconvénient de nécessiter un nouvel apprentissage pour chaque fonte.

Les systèmes multi - fontes [2] permettent la reconnaissance de plusieurs fontes, toutes ayant fait l'objet d'un apprentissage unique.

Les systèmes omni - fontes [2] sont capables de reconnaître des caractères de n'importe quelle fonte, ils nécessitent un apprentissage limité aux caractères spéciaux.

Des recherches ont été entreprises, de part le monde, pour reconnaître l'écriture imprimée, dactylographiée, ou même manuscrite. Des systèmes très performants arrivent à des taux de reconnaissance très élevés pour l'écriture imprimée latine, chinoise et indienne. Quant au manuscrit, il pose encore quelques problèmes vu la cursivité des lettres et leur variabilité d'une personne à une autre.

L'écriture arabe, objet de notre étude, a bénéficié de quelques recherches, sa difficulté réside dans sa cursivité, sa large gamme de styles ainsi que les différentes formes d'un même caractère selon sa position dans le mot. En 1980, à l'université de Nancy, A. Amin a débuté ses travaux sur la reconnaissance de caractères arabes et a réalisé un système IRAC (Interactive Recognition of Arabic Characters). Depuis, plusieurs

recherches ont succédé sur l'arabe en France, en Syrie, en Algérie, en Tunisie, etc..., selon l'approche statistique ou structurelle de la reconnaissance.

Les techniques utilisées en OCR progressent vers des solutions de traitement entièrement automatiques du document. En effet, l'utilisateur est contraint dans beaucoup de cas à un travail interactif avec l'OCR apportant corrections et mises au point de la reconnaissance. Actuellement, les utilisateurs intéressés par des systèmes de reconnaissance sont ceux qui acceptent, moyennant un contrôle des résultats et des interventions manuelles fréquentes, une reconnaissance de qualité variable.

La reconnaissance d'un document passe d'abord par une phase de décomposition dont le but est de séparer la page en blocs et les blocs en composants qui varient selon la nature du bloc et l'entité à reconnaître, suivie d'une étape de reconnaissance et enfin de reconstitution du document original.

Pour la plupart des systèmes, la décomposition d'un page ou sa segmentation se fait selon le modèle suivant:

- analyse globale de l'image et identification des blocs
- segmentation de chaque blocs en composantes élémentaires (pour un bloc de texte, il s'agit de segmenter en lignes, en mots puis en caractères).

Dans le cas d'un bloc de texte, le grand problème de la segmentation est la réalisation d'un système multi-langues, qui permettrait de segmenter un texte en n'importe quelle langue en caractères. Cette difficulté est due aux caractéristiques propres de chaque écriture surtout son sens et sa cursivité. Un autre problème de la segmentation est justement la cursivité de l'écriture et la recherche du critère de liaison des caractères.

Dans notre étude, nous avons soulevé le problème de la segmentation d'un texte arabe imprimé ou dactylographié par des méthodes basées essentiellement sur la notion d'histogrammes.

Dans le chapitre I, nous présentons la description d'un système OCR. Nous abordons aussi le module d'acquisition en détaillant deux formats très utilisés de fichiers graphiques. Le chapitre II est consacré au module de prétraitement. Nous définissons quelques filtres auxquels nous avons soumis des textes bruités et nous présentons les résultats obtenus. Enfin la segmentation d'un texte arabe imprimé est étudiée au chapitre III. Les différentes étapes y sont détaillées . La méthode choisie est décrite au chapitre IV, au sein duquel, les résultats obtenus par expérimentation seront interprétés. Une conclusion générale clôturera notre étude.

# CHAPITRE I

## DESCRIPTION D'UN SYSTEME OCR

# CHAPITRE I
# DESCRIPTION D'UN SYSTEME OCR

## I.1. Objectif de la reconnaissance :

L'objectif de l'analyse d'un document est l'obtention d'une description synthétique des différents éléments qui le constituent, à partir de la masse d'informations qu'il contient à l'état brut. C'est d'abord un moyen économique qui permet de baisser considérablement le coût de la saisie et augmente sa vitesse, mais aussi l'image saisie en mode " image" peut être traitée par un logiciel de traitement de texte ou de PAO (Publication Assistée par Ordinateur).

La reconnaissance de caractères permet à une machine de simuler le comportement humain de lecture. Un document se trouvant au départ sur un support papier, et après passage par un système OCR (Optical Character Recognition) doit être reconstitué après avoir été segmenté.

De manière générale, un système de reconnaissance, comme le montre la figure (1) est composé de :

- un module d'acquisition et de prétraitement
- un module de segmentation
- un module de reconnaissance.

## I.2. Acquisition de l'image :

Le module d'acquisition permet le passage de l'information du monde réel au monde numérique de l'ordinateur. Ce passage qui est donc la transformation d'un ensemble de données analogiques, par un dispositif physique, en un ensemble de données numériques, s'appelle la numérisation.

Les systèmes d'acquisition actuels les plus courants, sont essentiellement des scanners ou des caméras linéaires. Les scanners actuels ont une résolution minimum de 300 dpi (dot per inch) et peuvent atteindre carrément 800 dpi [2]. Ils permettent en outre la saisie d'images en niveaux de gris, voire en couleur. Toutefois, ils sont relativement lents. Il faut souvent des temps de l'ordre de la minute pour faire une acquisition. Cette relative lenteur est liée à la quantité d'informations [2] (plusieurs Méga - octets de données) à transférer entre le scanner et le système de traitement.

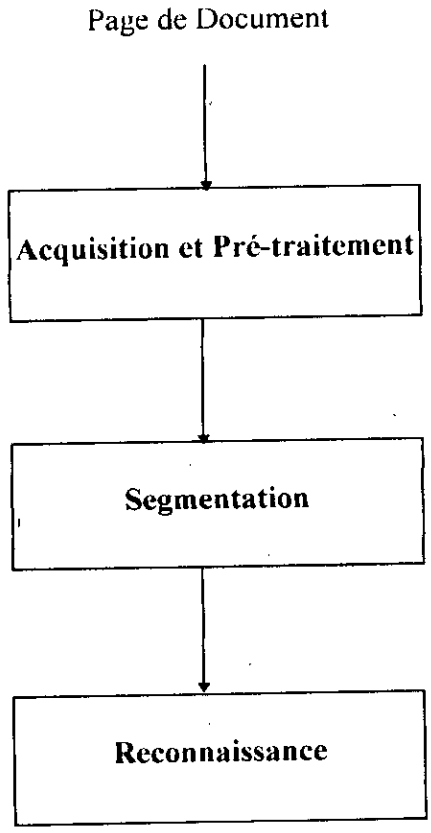



**Figure -1- Synoptique des étapes de l'analyse d'une image.**

Le principe de fonctionnement d'un scanner est une acquisition ligne par ligne par une barrette CCD. Trois types se scanners existent :

***Les scanners à plat*** : Le document à scanner repose sur une vitre et n'est pas déplacé par l'appareil. C'est la barrette CCD que se déplace. Ce type de scanner peut traiter des feuilles volantes, comme celles d'un livre sans avoir besoin de les détacher.

***Les scanners à rouleau*** : Pour lesquels, la barrette CCD est fixe, c'est le document à scanner qui est entraîné à travers une fente pour passer devant la barrette. Ce type de scanner n'accepte que des feuilles volantes.

***Les scanners à main*** : Pour lesquels, le système transmet deux informations, la valeur des pixels de la ligne, et une information de position. Il y a alors une phase supplémentaire de traitement confiée au système qui est la reconstitution de l'image.

Le système d'acquisition, le scanner en l'occurrence, est relié à l'ordinateur et est piloté par un logiciel. Ce dernier permet de digitaliser l'information analogique et stocke les données numériques dans un fichier graphique. Cette étape de l'acquisition est dite physique. Le deuxième niveau d'acquisition qui est dit logique, consiste à exploiter le fichier graphique pour en extraire l'image. Elle se présente alors sous forme d'un tableau M à n lignes et p colonnes. Chaque élément de la matrice M(i, j) représente un pixel de l'image dont l'intensité est une valeur numérique couramment appelée niveau de gris. Celle-ci est en général égale à 0 pour le noir et N pour le blanc, avec N = 255 le plus souvent. Ce choix de 256 niveaux de gris se justifie par le fait qu'un pixel est codé sur 8 bits, ce qui est une valeur commode [**3**].

Pour des images binaires, ce qui est le cas des textes que nous traitons, chaque pixel est codé sur un bit, soit "0" pour le noir et "1" pour le blanc. De ce fait, l'espace mémoire recquis pour stocker une image binaire est beaucoup plus faible que celui d'une image à plusieurs niveaux de gris. Dans le cas d'une image de 256 niveaux, il serait huit fois moins élevé.

L'image ainsi obtenue, peut être stockée sur disque dur sous forme de fichier, ou directement sur la mémoire RAM, sous forme de matrice, ce qui réduit considérablement le temps d'accès aux données par rapport à l'accès disque dur qui est relativement lent. Toute fois, on peut se heurter au problème de la taille de la RAM qui est beaucoup plus faible que celle d'un disque dur, et ce, dans le cas d'images contenant un grand nombre d'informations.

### I.2.1. Fichiers graphiques :

#### I.2.1.1. Introduction :

Plusieurs formats de fichiers graphiques sont disponibles sur PC (Personnal Computer). La différence entre un format et un autre, réside dans l'organisation de la structure, l'emplacement de l'image au sein du fichier, sa taille, etc...

L'information est inscrite dans un fichier selon des règles qui régissent la structure de chaque format. La structure peut être elle même décrite au sein du fichier. Certains formats sont plus flexibles que d'autres. Cependant, cette flexibilité est payée au prix d'une forme plus complexe.

L'accès à un fichier graphique nécessite la connaissance au préalable de sa structure, car la lecture suit les mêmes règles que l'écriture. Les principaux formats utilisés sur PC sont : . **BMP, . PCX, . TGA, . TIF, . GIF**.

### I.2.1.2. Format BMP [4] :

Schématiquement, un fichier BMP peut être décomposé en quatre blocs consécutifs :

- Bloc BitMapFileHeader
- Bloc BitMapInfoHeader
- Bloc codage des couleurs
- Bloc Bitmap

#### a- BitMapFileHeader:

La taille de cette entête est de 14 octets comme l'illustre la figure 2 :

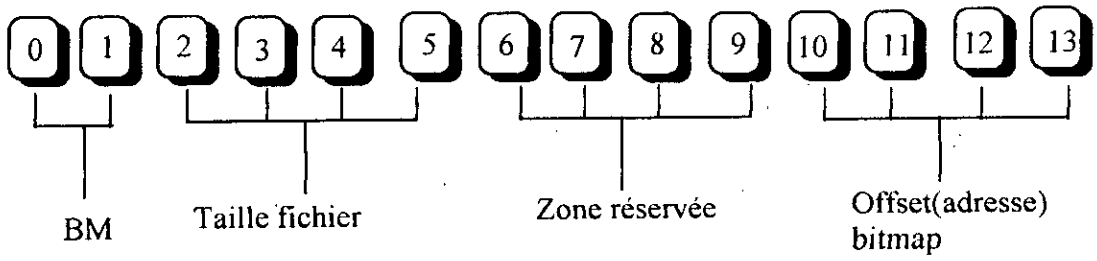



**Figure -2- Entête d'un fichier BMP (BitMapFileHeader)**

*Octets 0-1:*

Ils permettent l'identification du fichier BMP. Ils contiennent les codes ASCII des deux lettres « BM ».

*Octets 2-5:*

Ils contiennent la taille totale du fichier, ce qui permet d'avoir une taille maximale de 4 Giga Octets.

*Octets 6-9:*

Zone réservée et mise à zéro.

*Octets 10-13*:

Ils indiquent l'offset (l'adresse) des données bitmap.

## b- Informations sur l'image: (BitMapInfoHeader)

Pour un fichier BMP 3.0, la taille de ce bloc est de quarante octets. Pour une autre version cette taille est indiquée par les premiers octets de **BitmapInfoHeader**.

*Octets 0-3:*

Ils contiennent la taille en octets consacrée à ce bloc.

*Octets 4-7:*

Ils indiquent la largeur de l'image exprimée en pixels.

*Octets 8-11:*

Ils indiquent la hauteur de l'image exprimée en pixels.

*Octets 12-13:*

Ils sont toujours mis à Un (1).

*Octets 14-15:*

Ils indiquent le nombre de bits nécessaires pour coder un pixel. Quatre valeurs sont possibles: 1 pour les images monochromes, 4 pour les images à 16 couleurs, 8 pour des images à 256 couleurs et 24 pour des images en couleurs réelles(16 millions de couleurs).

*Octets 16-19:*

La valeur de ces quatre octets indique la méthode de compression utilisée. Si l'image n'est pas compressée, cette valeur est nulle.

*Octets 20-23:*

Ils indiquent la taille de l'image en octets.

*Octets 24-27:*

Ils indiquent la résolution horizontale en pixels par mètre.

*Octets 28-31:*

Ils indiquent la résolution verticale en pixel par mètre. Les résolutions horizontale et verticale en pixels par mètre donnent la possibilité à une application de choisir parmi plusieurs BMP celui qui convient le mieux au périphérique écran employé.

*Octets 32-35:*

Ils indiquent le nombre de couleurs utilisées. Si la valeur est nulle, alors toutes les couleurs sont utilisées.

*Octets 36-39:*

Ils indiquent le nombre de couleurs importantes. Si la valeur est nulle, alors toutes les couleurs le sont.

### c- Codage des couleurs :

Ce bloc comprend une liste de 2, 4, 16 ou 256 couleurs, décrites chacune par 4 octets. Le premier octet spécifie le bleu, le second le vert et le troisième le rouge. Le dernier octet est mis à zéro.

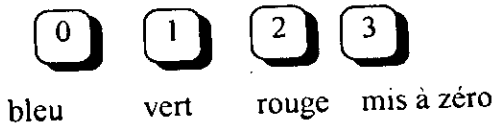



**Figure - 3 - Codage des couleurs**

La raison pour laquelle chaque couleur est codée sur 4 octets au lieu de 3, est qu'il est plus souple de manipuler des mots de 32 bits.

### d- Bloc Bitmap :

Chaque pixel de l'image est représenté par 1, 4, 8 ou 24 bits.

## I.2.1.3. Format TIFF (*TAGGED IMAGE FILE FORMAT*) [5]:

Un fichier TIFF, est actuellement l'un des formats graphiques les plus utilisés sous l'environnement Windows. Il est connu pour être très flexible et dynamique. Il peut contenir une image en couleur ou en échelle de gris, ou même plusieurs images. Il offre aussi la possibilité de compression pour des images de grande taille.

L'information dans un fichier TIFF, n'est pas figée à un emplacement spécifique, elle peut être placée à n'importe quelle position grâce à un pointeur qui indiquera son adresse.

Schématiquement, un fichier TIFF peut se décomposer en trois parties distinctes:

- L'entête (Header).
- Les IFD (*I*mage *F*ile *D*irectory).
- Les images proprement dites.

### a- L'Entête:

Les huit (8) premiers octets d'un fichier TIFF représentent l'entête comme l'illustre la figure (4) :

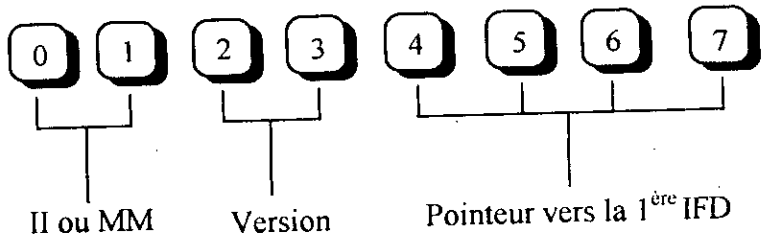



**Figure - 4 - Entête d'un fichier TIFF**

*Octets 0-1:*

Ils contiennent les valeurs 4949 ou 4D4D qui correspondent respectivement aux codes ASCII de II pour Intel, et MM pour Motorrola.

Les fichiers TIFF sont conçus pour fonctionner sur PC et sur Macintosh. Pour un fonctionnement sur PC, et pour des mots de 16 et 32 bits, l'octet le moins significatif est listé en premier. Sur Macintosh, l'octet le plus significatif est mis en premier. Tout programme destiné à lire un fichier TIFF doit prendre cette caractéristique en considération, et inverser les octets dans le cas d'utilisation d'un PC.

*Octets 2-3:*

Ces deux octets représentent le « numéro de version ». Ils indiquent la version du format (5.0) et contiennent la valeur 2A en Hexadécimal.

*Octets 4-5-6-7:*

Ces octets contiennent la valeur d'un pointeur qui indique la position de la première IFD (*I*mage *F*ile *D*irectory). Cette position s'exprime en nombre d'octets qui existent entre le début du fichier et la première IFD. Cette dernière aurait pu être placée juste après le numéro de version, mais au lieu de cela, le concepteur permet de la mettre à un emplacement quelconque pour donner une certaine flexibilité à cette structure.

## b- L'IFD (*I*MAGE *F*ILE *D*IRECTORY) :

L'IFD est une table contenant un nombre donné d'entrées, indiqué par les deux premiers octets de la table. A la suite, se trouve les différentes entrées, chacune sur 12 octets, comme l'illustre la figue (**5**).

Les quatre (4) derniers octets de l'IFD contiennent un pointeur vers l'emplacement de la prochaine IFD dans le cas où le fichier se compose de plus d'une image. Dans le cas contraire, ces octets sont mis à zéros.

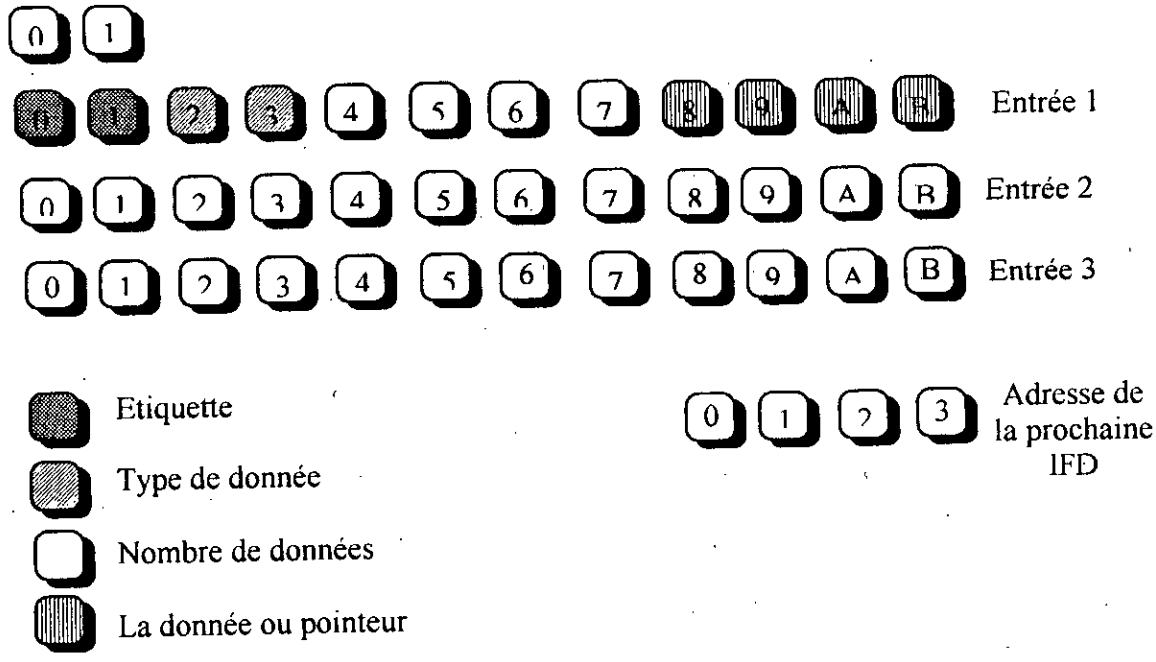



**Figure -5- IFD (Image File Directory)**

*Les entrées de l'IFD:* Chaque entrée est une suite de 12 octets.

*Octets 0-1:*

Ils contiennent une étiquette ou tag qui identifie le champ. Par exemple, le tag 257 représente la longueur de l'image.

*Octets 2-3:*

Ils contiennent une valeur qui indique le type des données qui suivront. Si leur contenu est 1, les données sont entières non signées sur 8 bits. Si le contenu est 3 par exemple, les données sont entières non signées sur 16 bits, etc...

*Octets 4-7:*

Ils indiquent le nombre de données dans le champ. S'il s'agit par exemple de la longueur de l'image, ce nombre serait égal à 1. Si la donnée est une chaîne de caractères, alors le nombre de données serait égal à la longueur de la chaîne.

*Octets 8-11:*

Ils contiennent la donnée elle même. Si celle-ci ne suffit pas sur 4 octets, ces derniers contiennent un pointeur vers la position où la donnée est stockée.

Dans une table IFD, les tags ne sont pas obligatoirement tous utilisés. Nous citons quelques tags les plus couramment utilisés:

**256 :** largeur de l'image.

**257 :** longueur de l'image.

**273 :** adresse de l'image proprement dite.

**262 :** si sa valeur est nulle, alors le blanc sera représenté par zéro et le noir par N. Si sa valeur est 1, le blanc sera représenté par N et le noir par zéro.

**Lecture d'un fichier TIFF :**

Comme le montrent les figures (**6**) et (**7**), l'accès à un fichier TIFF commence par la lecture des octets 4 et 5 de l'entête, qui représentent l'adresse de la première IFD. Les octets 6 et 7 sont généralement mis à zéro pour des fichiers mono - image. Le travail s'effectuant sur un PC, l'ordre des octets est rétabli par décalage.

Après positionnement sur la première IFD, le nombre d'entrées est lu. Les tag 256, 257 et 273 qui représentent respectivement la largeur, la longueur et l'adresse de l'image, sont recherchés séquentiellement.

L'image est extraite à partir du fichier octet par octet puis convertie en binaire pour les traitements ultérieurs. Pour notre programme, le " 0 " représente le blanc (le fond) et le " 1 " représente le noir (l'écriture).

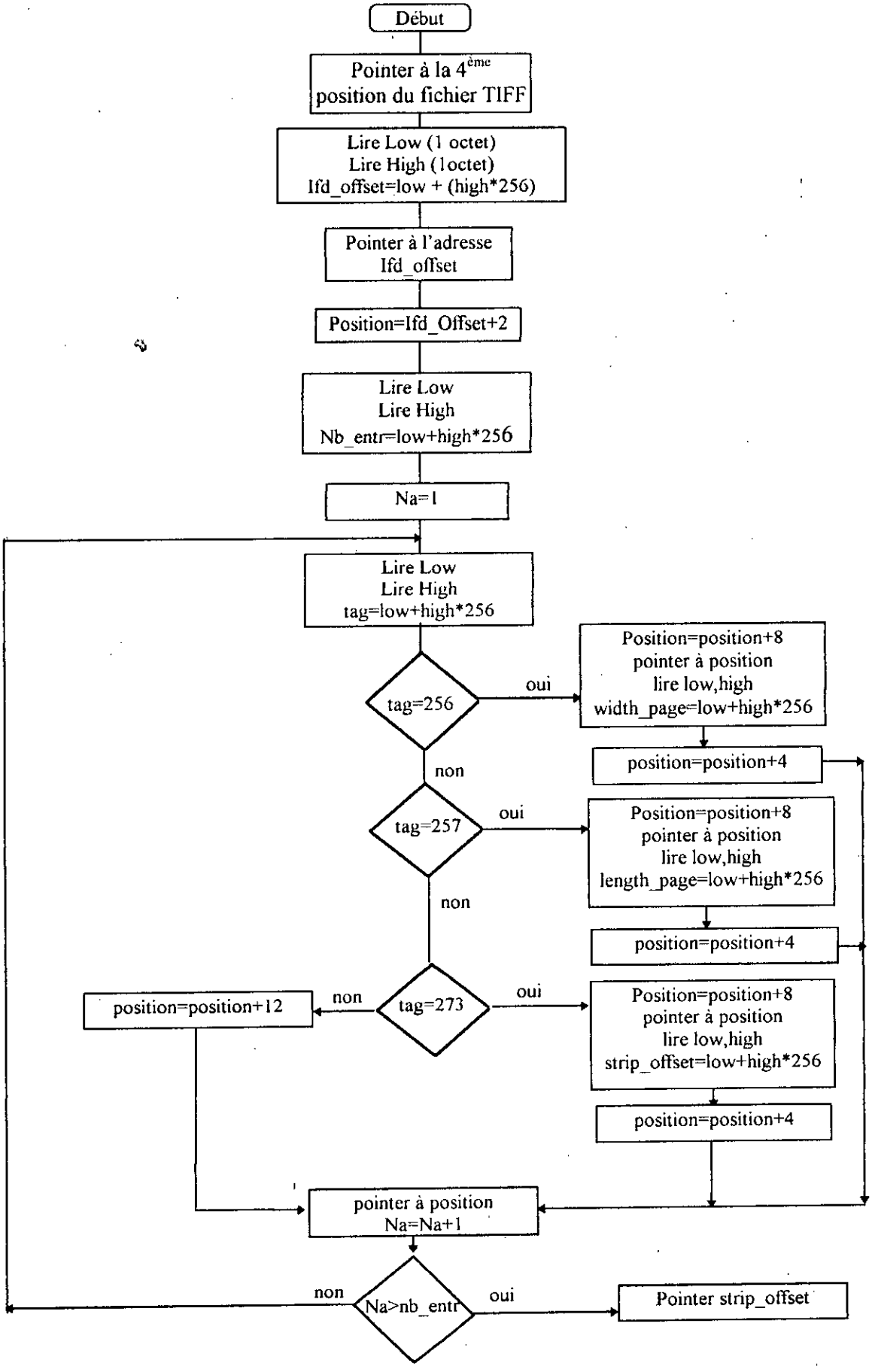



**Figure -6- Organigramme de lecture des informations dans un fichier TIFF**

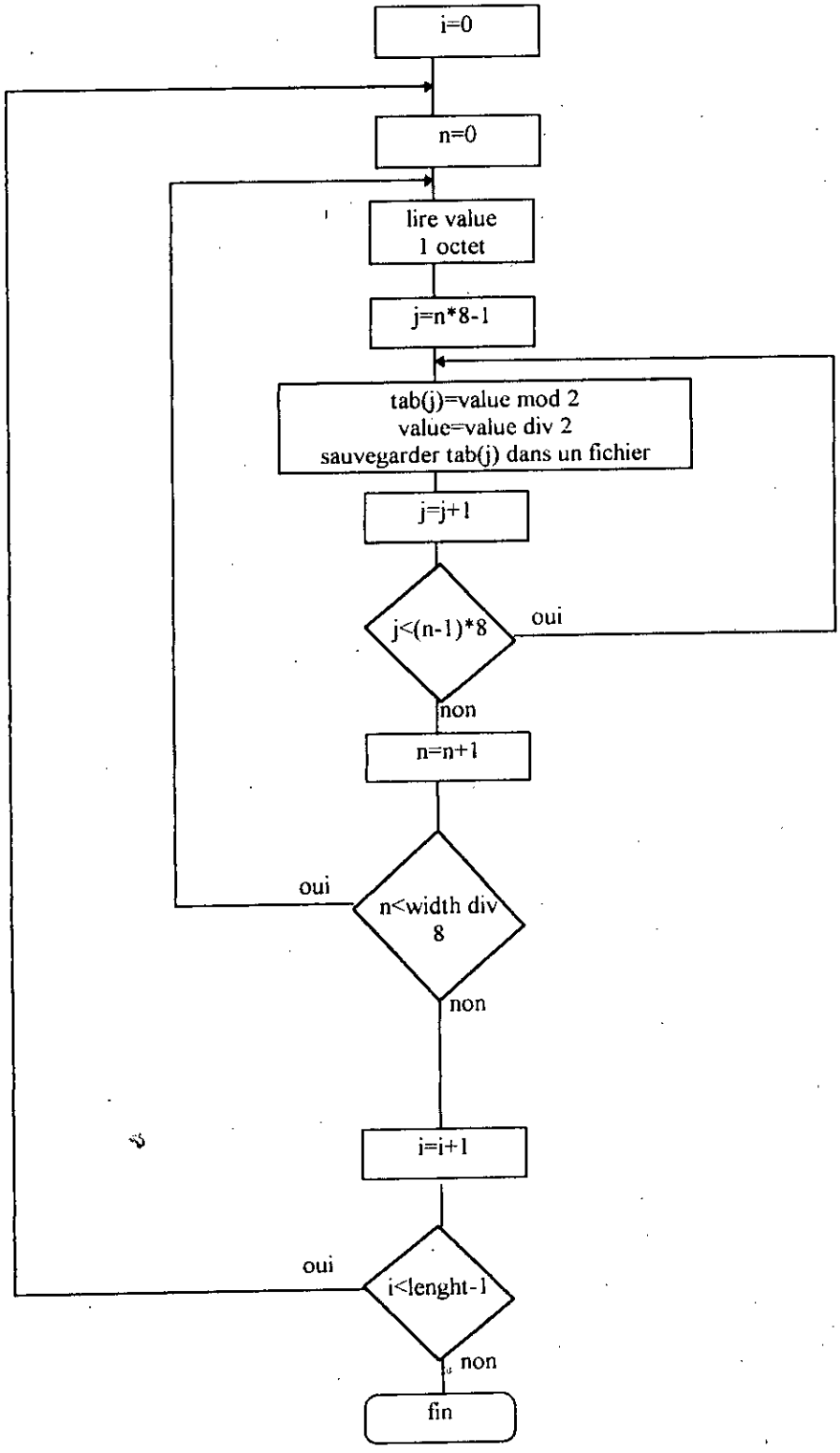



**Figure -7- Organigramme d'extraction de l'image**

## I.3. Prétraitement:

Lorsque l'acquisition est réalisée, la plupart des systèmes OCR comportent une étape de prétraitement. Généralement, ces prétraitements ne sont pas spécifiques de la reconnaissance de textes, mais sont des prétraitements classiques en traitement d'image.

Le prétraitement peut comprendre des dizaines d'opérations différentes effectuées sur l'image numérisée. Cette dernière peut présenter des distorsions ou des perturbations provoqués par divers facteurs, et qui sont principalement l'outil d'acquisition (qualité du digitaliseur) et l'état du papier (jaunissement de page) [3]. A cette phase de l'analyse, le but est de préparer les étapes ultérieures de la reconnaissance. Parmi ces opérations, on peut citer par exemple le problème de la réduction de bruit d'acquisition ou celui de l'élimination du fond dans le cas de la lecture d'un texte écrit sur un fond texturé (papier tramé, chèques, etc...).

On inclut aussi, généralement dans les prétraitements des opérations de type redressement des écritures penchées, détection et éventuellement redressement des lignes de base, etc...[6].

Dans le cas des images à plusieurs niveaux de gris, on peut recourir à une binarisation qui est considérée comme un traitement préliminaire, et ce, tant pour la facilité de manipulation d'images binaires que pour le faible espace mémoire requis pour les stocker. Le principal problème de la binarisation réside dans le choix d'un seuil et dans la détermination de la zone de validité de ce seuil de décision. Si la zone de validité représente l'image entière, le seuil est dit « global »; dans ce cas, qui est très courant en pratique, le risque que l'image comporte des zones pour lesquelles ce choix est mal adapté est grand. Si la zone de validité est locale, les techniques de binarisation sont alors à seuil adaptatif [2].

L'amélioration des images consiste en un ensemble de méthodes qui visent à atténuer les effets indésirables. Deux principales approches ont été envisagées [3]:

- Transformations ponctuelles : pour lesquelles chaque pixel est transformé indépendamment des pixels voisins.
- Filtrage : pour lequel la valeur de chaque pixel est évaluée par une combinaison des pixels voisins.

Pour éliminer les redondances dans un document, il peut être soumis à un traitement préliminaire appelé la squeletisation. Celle-ci consiste à réduire la largeur des traits constituant le caractère, à la valeur unitaire d'un pixel. la figure (8) illustre l'opération de squeletisation.

```
000001100000000000000   000000100000000000000
000001100000000000000   000000100000000000000
000001100000000000000   000000100000000000000
000001100000000000000   000000100000000000000
000001100000000001100   000000100000000000000
000001100001100011110   000000100001000001100
000001100010110010000   000000100010100010000
000001111111111111111   000000100010110010000
100001111101101111110   100000011101001101110
110001100000000000000   100000100000000000000
110001100000000000000   100000100000000000000
111111000000000000000   100001000000000000000
011110000000000000000   011110000000000000000
```

**Figure - 8 - : Exemple de caractères « عمل » squeletisés**

## I.4. Segmentation :

Les objets qui composent un document composite varient des plus simples (signes de ponctuation) aux plus compliqués (textes, tableaux, formules, graphiques). Ces objets sont appelés entités physiques et correspondent à des groupements de pixels délimités par un séparateur physique. L'entité de base est la composante connexe; le séparateur est alors la transition noir / blanc ou blanc / noir.

Les entités logiques correspondent à des groupements de pixels logiques délimités par un séparateur logique. Dans le cas de l'écriture, l'entité de base est le caractère, qui ne correspond pas directement à l'entité physique pour les lettres attachées.

La segmentation est une opération qui consiste à séparer un document en ses entités élémentaires, lesquelles sont destinées au module de reconnaissance. Deux approches différentes de segmentation existent [3]; Celle qui tient compte du contour des objets : on parle de segmentation par extraction de contour. Et celle qui privilégie le contenu de la région à segmenter, donc les pixels qui la composent.

On ne peut pas dire qu'une approche est meilleure que l'autre, leur différence réside dans l'information qu'elles mettent en valeur et qui est le contour pour l'une et le contenu pour l'autre.

## I.5. Reconnaissance :

Le module de reconnaissance de caractères peut se décomposer en 2 sous modules [7]:

- Extraction de caractéristiques.
- Classification et reconnaissance.

### I.5.1. Extraction des caractéristiques :

L'extraction des caractéristiques prend des formes différentes en fonction de la méthode utilisée. L'ensemble des caractéristiques le plus classique est obtenu en procédant à certaines mesures dont nous citons, pour le cas des méthodes structurelles [7]:

- recherche de la distance entre chaque point du caractère et le bord du rectangle qui l'encadre, le nombre des trous fermés.
- détection des angles.
- calcul des centres de gravité.
- recherche des points d'intersections, des points de branchement etc...

### I.5.2. Classification :

La finalité de la reconnaissance est l'identification d'un caractère après extraction de ses caractéristiques. Un système ne peut reconnaître un caractère que s'il le connaît déjà. L'apprentissage est donc une phase importante de la chaîne de reconnaissance.

L'apprentissage consiste à construire un dictionnaire qui contiendra un certain nombre de prototypes obtenus en consultant toutes les formes possibles des caractères pouvant être soumis à ce système [7]. Plus le dictionnaire est riche, meilleure sera la reconnaissance.

Pour l'écriture arabe imprimée ou dactylographiée, on peut compter une vingtaine de fontes comportant en moyenne 300 formes différentes chacune [8]. Le caractère à reconnaître sera donc comparé à tous les prototypes, et sera identifié à celui dont les caractéristiques sont les plus proches. Le taux de reconnaissance pourrait être élevé, cependant, cette identification coûterait cher en temps de calcul et en espace mémoire, du fait de la grande taille du dictionnaire.

Un moyen pour palier à cet inconvénient et de procéder d'abord à une classification des prototypes. Ainsi, le caractère à reconnaître ne sera plus comparé à tous les prototypes contenus dans le dictionnaire mais à un nombre restreint contenu dans une classe.

## I.6. Conclusion :

Dans ce chapitre, nous avons présenté une description générale d'un système de reconnaissance. Les différents modules d'un système OCR y sont décrits brièvement. Nous avons mis l'accent sur le module d'acquisition en décrivant les formats graphiques et notamment celui que nous avons utilisé dans notre étude et qui est le fichier TIFF. L'information recueillie dans ce fichier sera l'objet d'un prétraitement par filtrage avant de passer par les modules de segmentation et de reconnaissance.

# CHAPITRE II

## ETAPE DE PRÉTRAITEMENT : FILTRAGE

# CHAPITRE II
# ETAPE DE PRETRAITEMENT: FILTRAGE

## II.1. Introduction :

L'amélioration d'images est un ensemble de techniques dont le but est d'amoindrir les effets venus altérer l'information utile lors de l'étape d'acquisition. Dans une image numérisée, les pixels voisins possèdent pratiquement les mêmes caractéristiques physiques[3]. Un bruit est donc défini par une brusque variation d'un pixel par rapport à son voisinage. Ceci étant, les méthodes de filtrage ont été développées dans le sens de lutter contre les brusques variations d'un pixel en tenant compte de ses voisins. On peut compter deux grandes familles de filtres :

*Les filtres passe-haut:*

1. Gradient
2. Laplacien, etc...

*Les filtres passe bas:*

- Filtres linéaires pour lesquels la transformation d'un pixel est le résultat de la combinaison linéaire des pixels voisins. Exemple : filtre de Gauss, filtre moyen.

- Filtres non linéaires pour lesquels les pixels voisins interviennent selon une loi non linéaire. Exemple: filtre médian.

Pour des images binaires, telles que celles que nous étudions, il est possible de réduire le bruit en utilisant d'autres filtres appelés : opérateurs morphologiques. Ces derniers agissent sur un pixel de l'image en le combinant avec son voisinage par des relations logiques basées sur les opérateurs d'union "OU" et d'intersection "ET".

Schématiquement, on peut représenter le filtre par une boite noire qui reçoit en entrée l'image bruitée $I_B$ et fournit en sortie l'image filtrée $I_F$. Comme tout système électronique, un filtre est caractérisé par sa réponse impulsionnelle h ou sa transformée de fourrier H.

Ainsi, les signaux $I_F$ et $I_B$ sont reliés par les relations suivantes :

$I_F = I_B \otimes h$

$F(I_F) = F(I_B) * H$

ou $\otimes$ est le symbole du produit de convolution

et $F(I_F)$ est la transformée de Fourier du signal $I_F$

$F(I_B)$ est la transformée de Fourier du signal $I_B$

Pour tester l'efficacité des filtres sur un document écrit, nous avons considéré un texte dégradé par trois bruits réels différents. Le premier est un bruit additif représenté par un ensemble de pixels noirs isolés au milieu des parties blanches. La seconde image bruitée présente une distorsion visible directement sur le contour de l'écriture. Le dernier bruit est un ensemble de pixels noirs alignés selon un trait oblique qui traverse l'écriture.

## II.2. Filtres linéaires :

Comme indiqué précédemment, ce type de filtre utilise une loi linéaire du voisinage d'un pixel pour le modifier. Ceci est réalisé en faisant balayer tous les pixels de l'image ou une grande partie d'entre eux par un masque dont les coefficients dépendent du filtre utilisé. Le pixel central du masque est celui qui est affecté par la transformation.

### II.2.1. Filtre moyen [9] :

Partant d'une image initiale bruité $I_B$ de taille P x P , le filtre moyen consiste à générer une image filtrée $I_F$ , pour laquelle chaque pixel de coordonnées x, y est le résultat de la moyenne arithmétique non pondérée, d'un voisinage préalablement défini. Cette opération est réalisée par la relation suivante :

$$I_F(x, y) = \frac{1}{M} \sum_{m, n \in S} I_B(m, n)$$

où $x, y = 0 \ldots\ldots P - 1$

S : Ensemble des coordonnées du filtre.

M est la somme des éléments du masque (filtre).

Pour les pixels se trouvant sur le bord de l'image, le moyennage ferait appel à des pixels de coordonnées négatives n'appartenant pas à l'image.

Plusieurs solutions existent pour pallier à ce problème, dont celle que nous avons choisie et qui consiste à filtrer uniquement l'intérieur du filtre en ignorant les pixels pour lesquels le masque n'est pas applicable. La figure (9) illustre l'organigramme du filtre moyen.

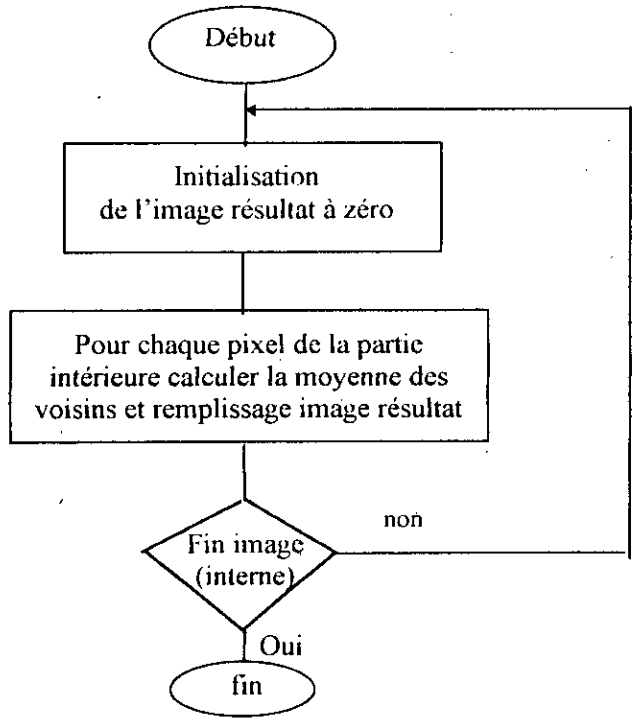



**Figure - 9 - : Organigramme du filtre moyen**

Nous avons soumis les trois images bruitées décrites précédemment au chapitre **II-1** à un filtre moyen de taille 3 x 3 , les résultats comme illustrés par la figure (**10**) montrent que ce filtre peut être efficace ou pas selon la nature du bruit. Il élimine effectivement les pixels noirs isolés mais affecte la taille de l'écriture en l'amincissant.

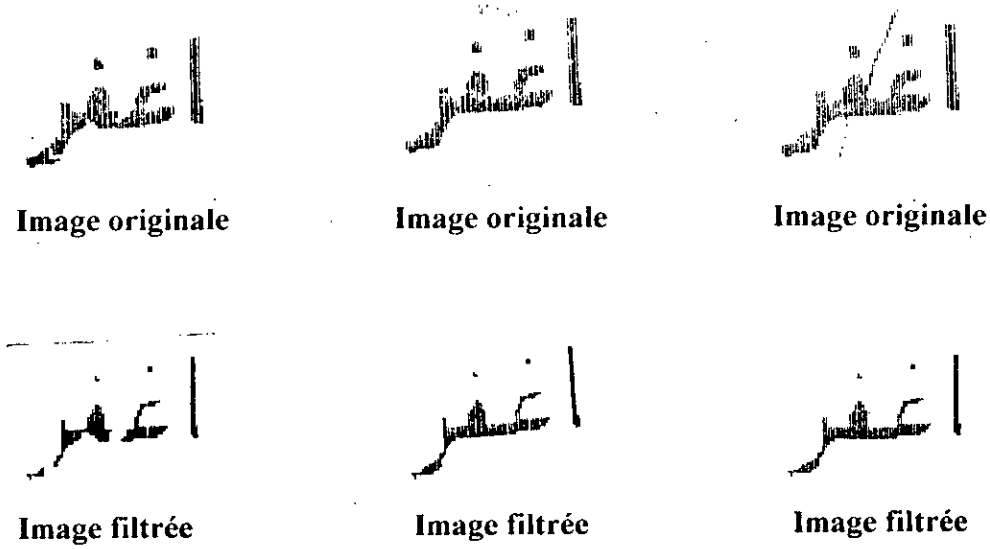


**Image originale** **Image originale** **Image originale**

**Image filtrée** **Image filtrée** **Image filtrée**

**Figure - 10 - : Filtre moyen**

### II.2.2. Filtre de GAUSS

Le filtre de Gauss fait partie des filtres linéaires les plus simples à mettre en oeuvre. L'image initiale est convoluée à une gaussienne G(x, y,σ) à deux dimensions donnée par la formule suivante **[3]** :

$$G(x, y, \sigma) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

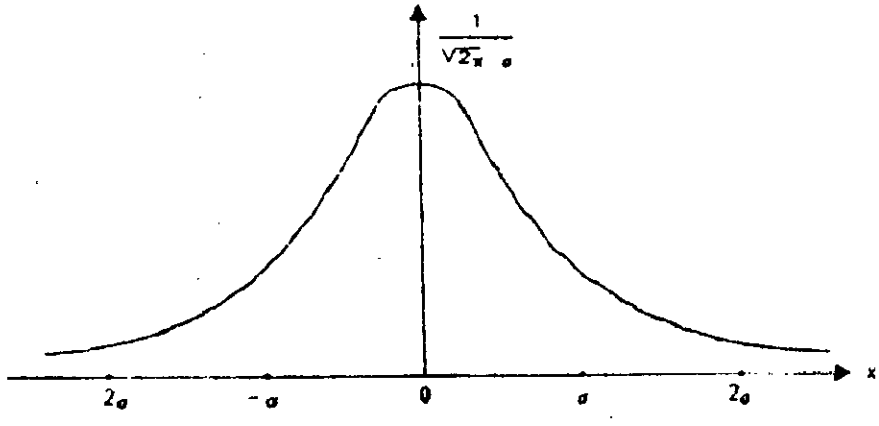



**Figure -11- Courbe de GAUSS à une dimension.**

L'opération est similaire au filtrage par moyennage, sauf que dans ce cas, la moyenne est pondérée par des coefficients qui sont les éléments discrétisés d'une gaussienne. Elle est effectuée en deux phases: la première consiste à convoluer l'image initiale par une gaussienne à une dimension suivant la direction x, le résultat est alors convolué à une autre gaussienne à une dimension selon la direction y.

La figure (**12**) illustre l'organigramme du filtre de GAUSS.

Les résultats obtenus sur des images bruitées, illustrés par la figure (**13**) nous poussent à tirer les mêmes conclusions que pour le filtre moyen.

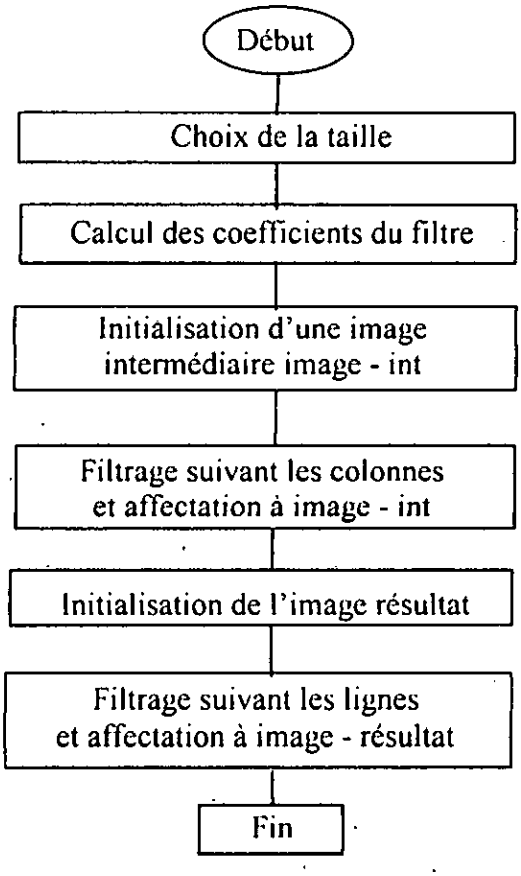



**Figure - 12 - : Organigramme du filtre de GAUSS**

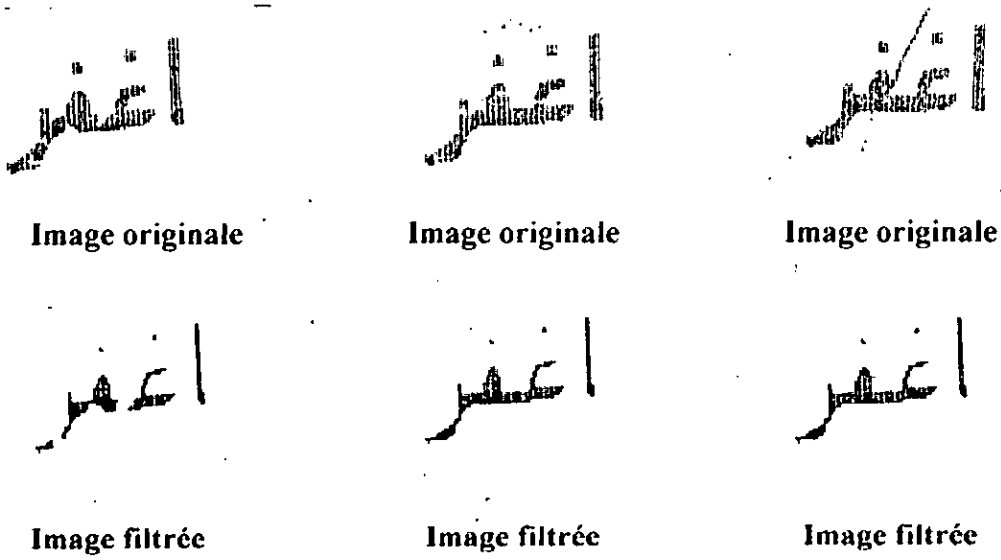



**Figure - 13 - Filtre de GAUSS**

## II.3. Filtres non linéaires :

### II.3.1. Filtre Médian [9], [3] :

Le filtre Médian de **Tuckey** est un filtre non linéaire qui agit sur le voisinage d'un pixel selon une loi non linéaire. Cette opération consiste à :

- classer les pixels voisins par ordre croissant,
- affecter la valeur médiane au pixel courant.

Les figures (**14**) et (**15**) illustrent le principe et l'organigramme du filtre Médian.

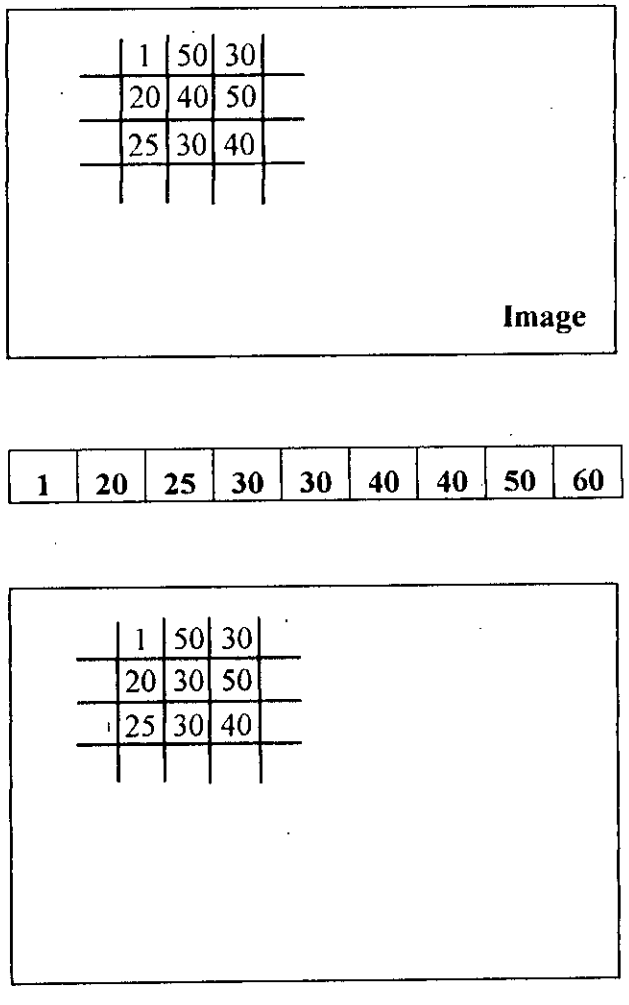



**Figure - 14 - : Principe du filtre Médian**

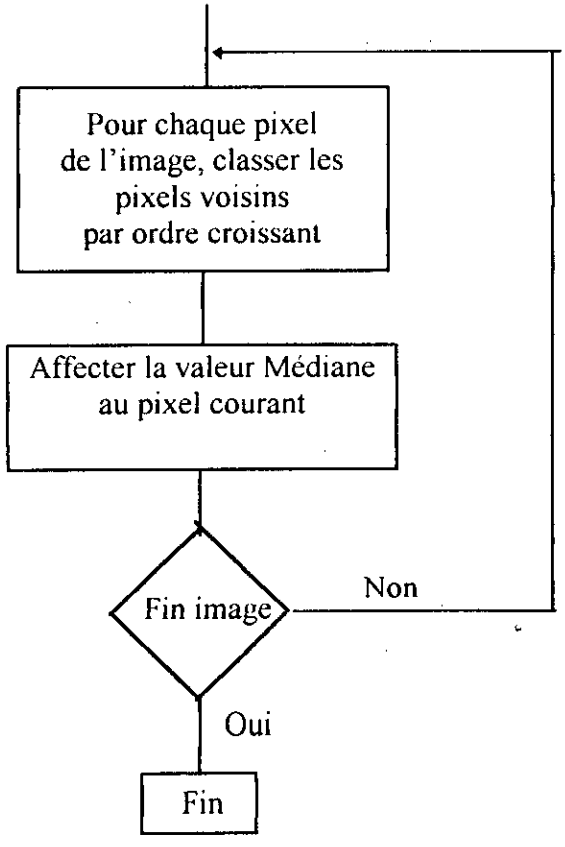



**Figure - 15 - : Organigramme du filtre Médian**

Le masque utilisé peut être de forme carré ou en croix (voir figure (16))

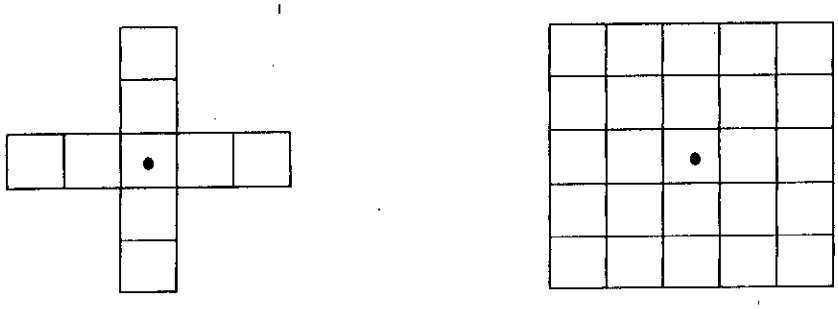


**Figure -16 - a- Masque en croix** **b- Masque carré**

Le filtre Médian semble très efficace pour une image dégradée par une source de bruit de type impulsionnel, donc pour des variations brusques de pixels isolés. La taille du filtre influe beaucoup sur la qualité de filtrage. Plus la taille est grande plus le filtre peut paraître efficace, mais plus il déforme l'image sans pour autant améliorer le contraste [3].

Pour notre part, nous avons choisi un masque en croix de taille 5 x 5, et nous avons réalisé un filtrage sur l'image altérée par trois bruits différents. Il a été noté selon la figure (17) que le filtre médian élimine les pixels noirs isolés dans l'image, effectue un lissage du contour de l'écriture, tout en gardant pratiquement la même épaisseur des caractères.

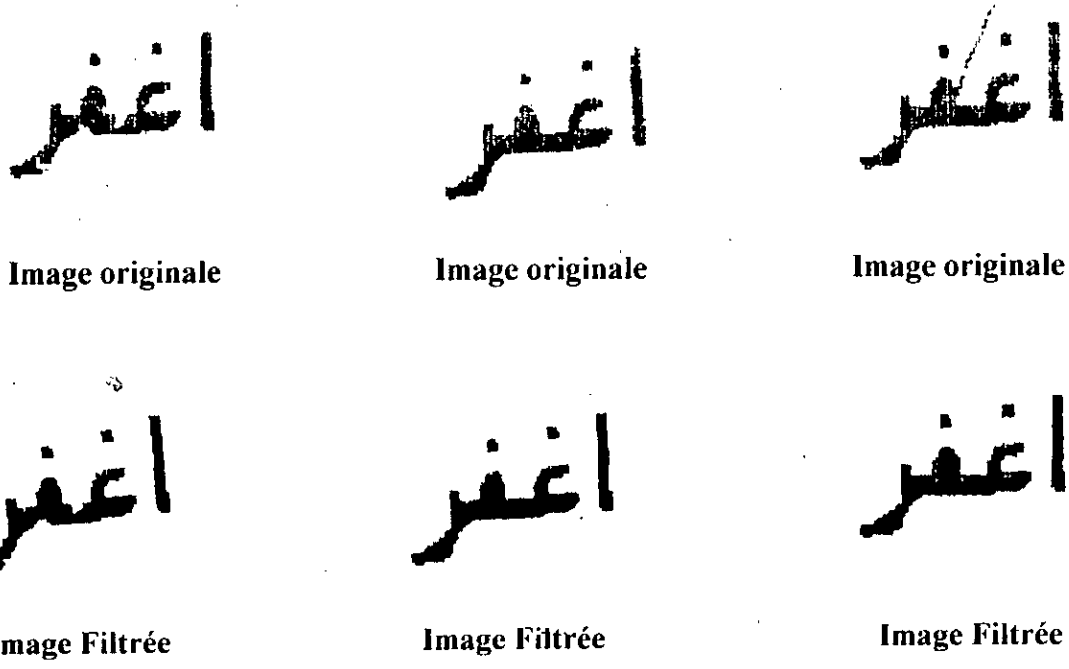


**Image originale** **Image originale** **Image originale**

**Image Filtrée** **Image Filtrée** **Image Filtrée**

**Figure -17 - Filtre Médian**

## II.4. Les filtres morphologiques

Le filtrage morphologique consiste à balayer l'image par un élément structurant qui n'est autre qu'un masque de forme quelconque. Ce dernier est mis en correspondance avec chaque pixel et son voisinage par une fonction logique plus ou moins complexe basée sur les opérateurs d'union et d'intersection [3].

La figure (18) illustre un exemple d'opérateur qui agit sur une image originale par intersection puis par union.

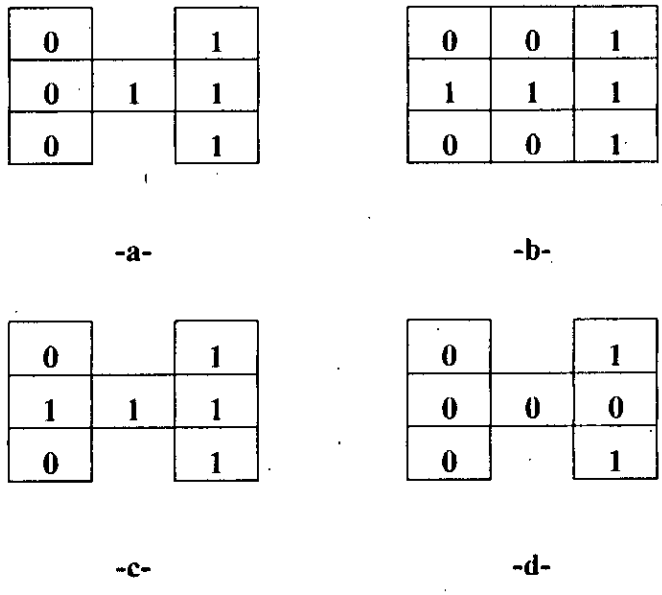



**Figure - 18 - : Exemple d'élément structurant**

**a- Elément structurant** **b- Image originale**

**c- Union** **d- Intersection**

### II.4.1. Dilatation [3]:

Elle consiste à effectuer le " ou " logique des huit pixels voisins à un pixel donné.

- Si le résultat vaut 1, alors le pixel courant est forcé à 1 dans l'image résultat
- S'il vaut 0, le pixel courant est recopié de l'image initiale vers l'image résultat.

Cette opération est réalisée sur tous les pixels de l'image. Son principe est illustré par la figure (**19- a, b**), ainsi que son organigramme par la figure (**20**). La dilatation comme le montre la figure (**21**) permet de dilater l'image, c'est à dire que les pixels noirs isolés au milieu de parties blanches sont éliminés.

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | | 0 | 1 | 1 | 1 | 0 | |
| 0 | 0 | 1 | 0 | 0 | ⟶ | 0 | 1 | 1 | 1 | 0 | -a- |
| 0 | 0 | 0 | 0 | 0 | | 0 | 1 | 1 | 1 | 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 | |
| 1 | 1 | 0 | 1 | 1 | ⟶ | 1 | 1 | 1 | 1 | 1 | -b- |
| 1 | 1 | 1 | 1 | 1 | | 1 | 1 | 1 | 1 | 1 | |
| 1 | 1 | 1 | 1 | 1 | | 1 | 1 | 1 | 1 | 1 | |
| | | | | | | | | | | | |
| 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 1 | 0 | 0 | ⟶ | 0 | 0 | 0 | 0 | 0 | -c- |
| 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 | 0 | 0 | 0 | 1 | |
| 1 | 1 | 0 | 1 | 1 | ⟶ | 1 | 0 | 0 | 0 | 1 | -d- |
| 1 | 1 | 1 | 1 | 1 | | 1 | 0 | 0 | 0 | 1 | |
| 1 | 1 | 1 | 1 | 1 | | 1 | 1 | 1 | 1 | 1 | |

**Figure - 19 - : Effets de dilatation et d'érosion**
**a- b- Dilatation c- d- Erosion**

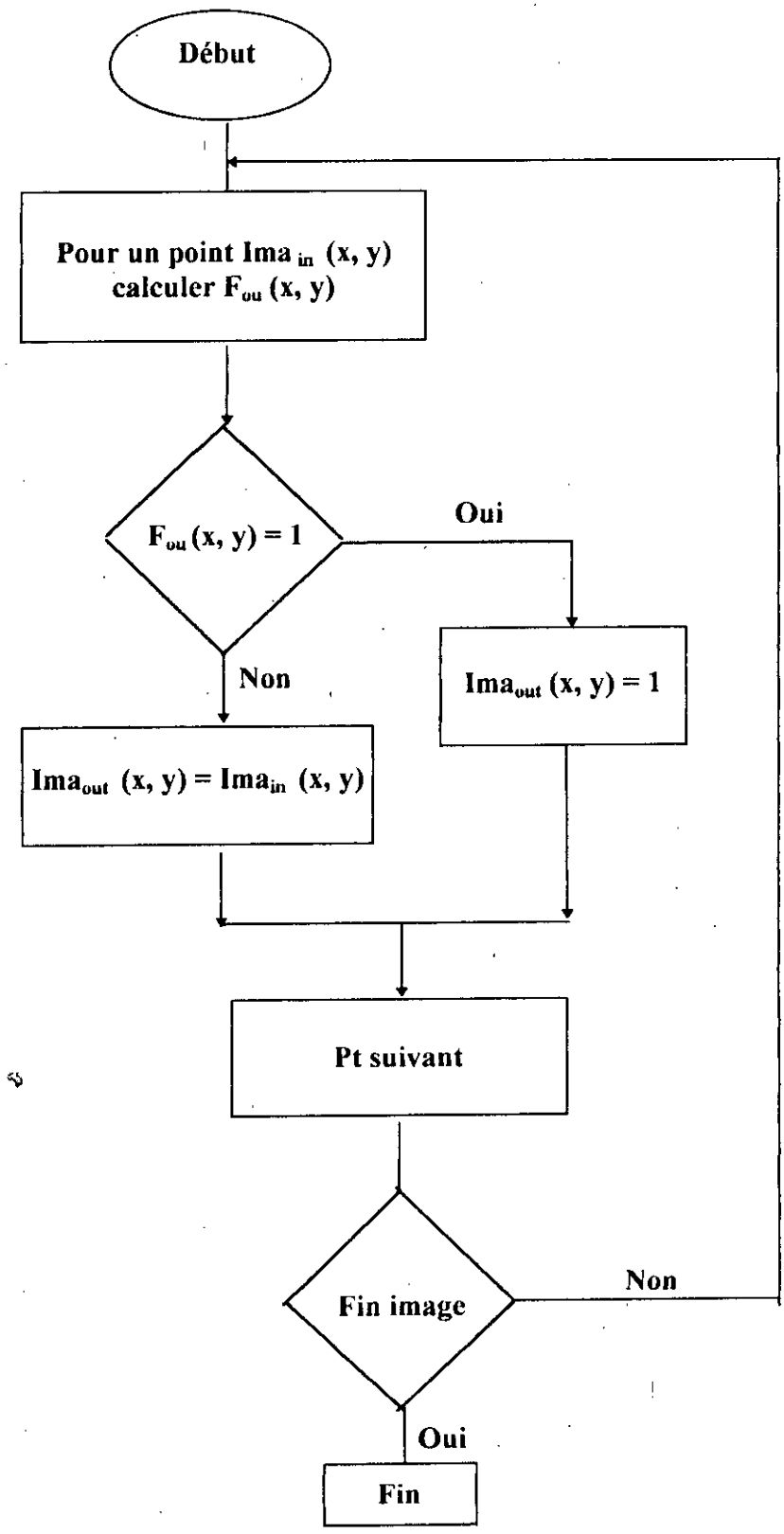



**Figure - 20 - : Organigramme de la dilatation**

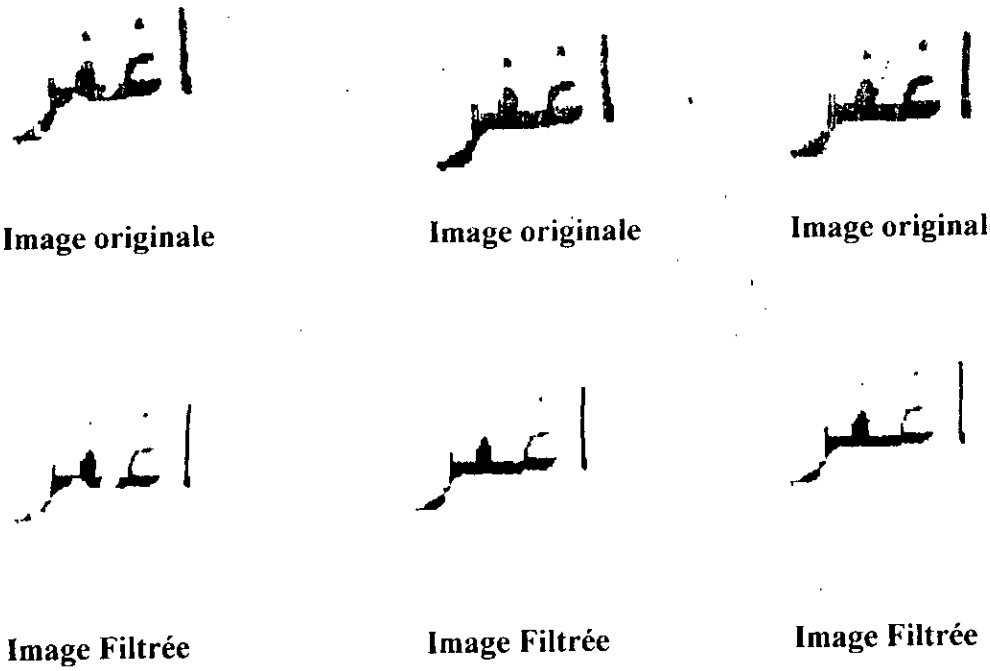


Image originale | Image originale | Image originale

Image Filtrée | Image Filtrée | Image Filtrée

**Figure - 21 - Dilatation**

### II.4.2. Erosion

Le " Et " logique du voisinage est calculé pour chaque pixel de l'image **[3]**.

- Si le résultat vaut 1, le pixel courant est recopié de l'image initiale vers l'image résultat
- S'il vaut 0, le pixel courant est forcé à 0 dans l'image résultat..

Les figures (**19 - c, d**) et (**22**) illustrent le principe et l'organigramme de l'érosion.

Il a été noté, d'après les résultats de l'expérimentation, de la figure (**23**), que l'érosion fait disparaître les points blancs au milieu de zones noirs, en lissant le contour. Les points noirs isolés ne sont pas réduits, mais sont amplifiés par la présence d'autres points autour d'eux. il faut noter aussi un épaississement de l'écriture.

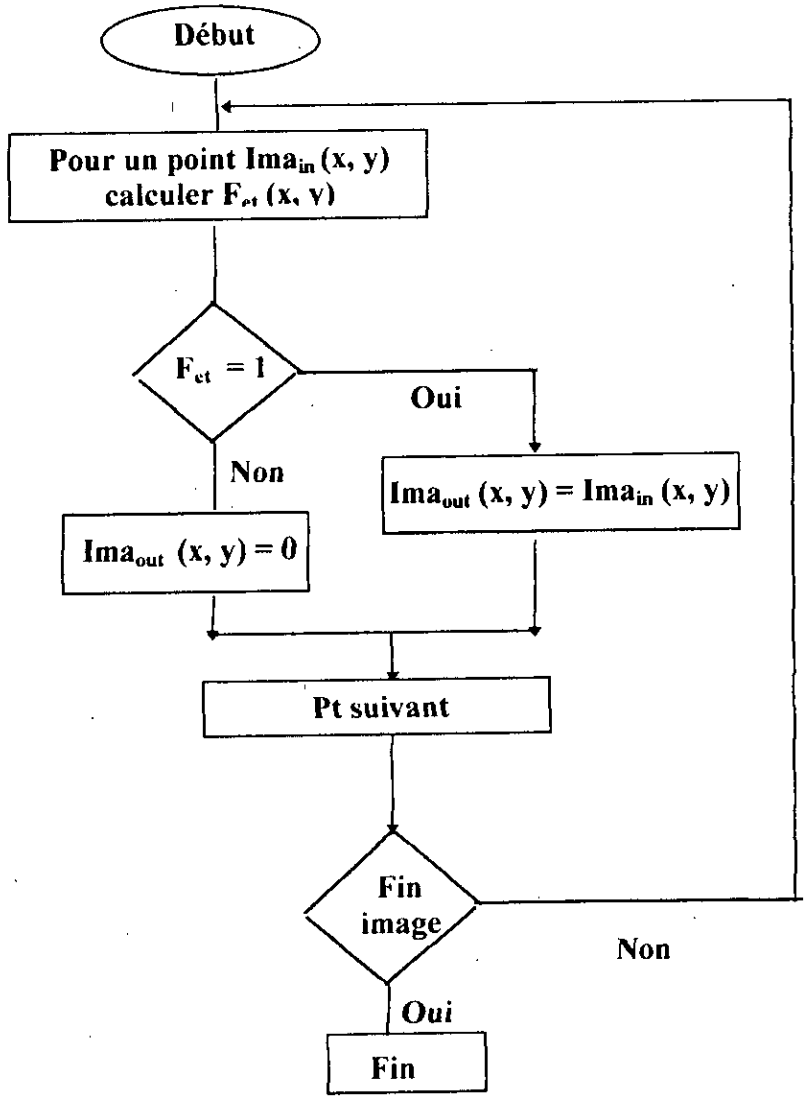



Figure - 22 - Organigramme de l'érosion.

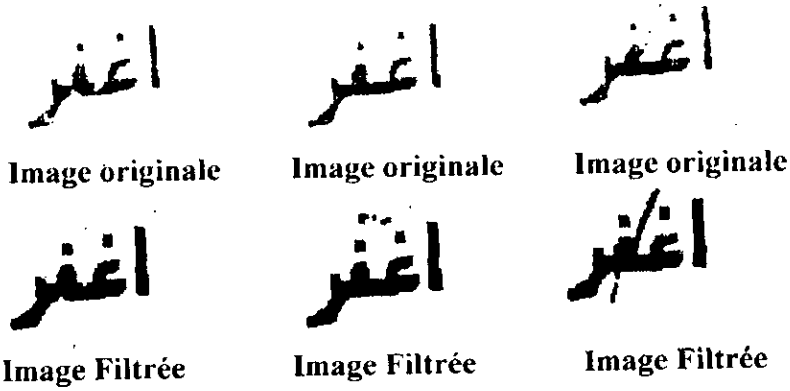



Figure - 23 - Erosion

### II.4.3. Ouverture

L'ouverture est réalisée par une érosion suivie d'une dilatation [3].

### II.4.4. Fermeture

La fermeture est réalisée par une dilatation suivie d'une érosion [3].

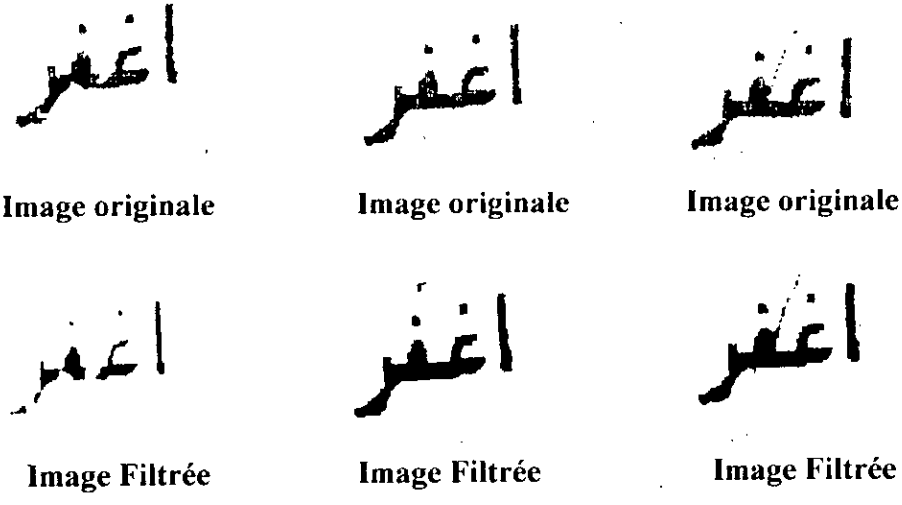


Image originale | Image originale | Image originale

Image Filtrée | Image Filtrée | Image Filtrée

Figure - 24 - Ouverture

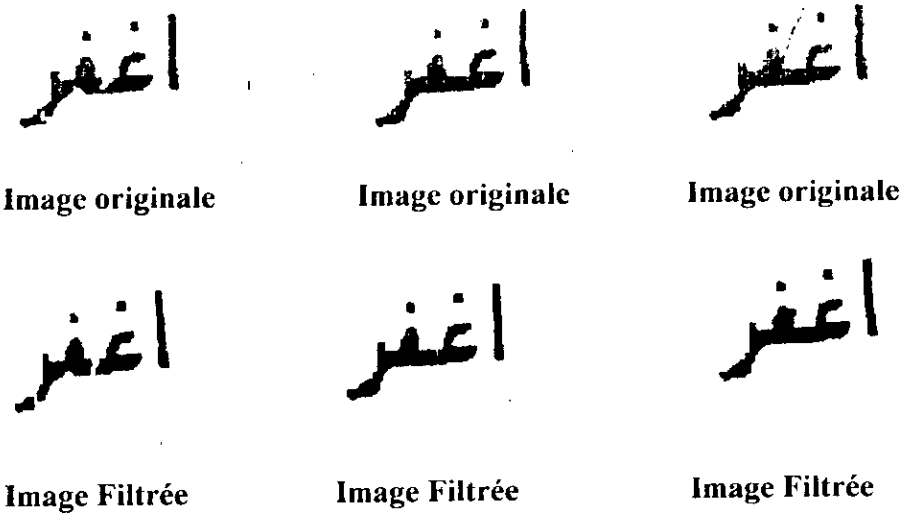


Image originale | Image originale | Image originale

Image Filtrée | Image Filtrée | Image Filtrée

Figure - 25 - Fermeture

## II.5. Conclusion :

Dans ce chapitre, nous avons mis en revue les différents filtres pouvant être appliqués à des images contenant un paragraphe de texte. Nous avons par ailleurs, soumis une image bruitée par trois bruits différents à ces filtres à savoir : les filtres linéaires (moyen, Gauss), les filtres non linéaires (médian) et les opérateurs morphologiques (dilatation, érosion, fermeture, ouverture)

Pour l'application des opérateurs morphologiques, nous n'avons pas eu à réaliser une binarisation, car les images traitées contenant des textes noirs sur un fond blanc, sont dès le départ sur deux niveaux de gris, donc binaires.

Notre but, pour cette partie de l'étude, était d'améliorer la qualité de l'image numérisée, en minimisant les bruits de différentes formes pouvant altérer l'aspect visuel de l'image. Nous avons constaté que selon la nature du bruit à éliminer, un même filtre pouvait être efficace dans un cas mais pas dans un autre. Le choix d'un filtre dépend donc essentiellement du bruit mais aussi de l'application associée à l'image. Dans notre cas, c'est à dire pour une application OCR, où l'image est un paragraphe de texte, et après expérimentation de tous les filtres cités plus haut, nous proposons le filtre médian de **Tuckey**, comme celui qui répond le plus à nos besoins.

# CHAPITRE III

## SEGMENTATION D'UN TEXTE EN CARACTERES

# CHAPITRE III
# SEGMENTATION D'UN TEXTE EN CARACTERES

## III.1. Introduction :

A la suite des prétraitements tels que ceux que nous avons abordé au chapitre II, il est nécessaire de segmenter l'image pour obtenir les entités élémentaires, lesquelles seront traitées dans une phase ultérieure par le module de reconnaissance.

De manière générale, la structure d'un document est composée de plusieurs couches:

- Une couche de graphiques : Celle-ci porte la majeure partie de l'information et est convertie en primitives par un module de vectorisation [**10**].
- Une couche de texte
- D'autres couches éventuelles (symboles et équations mathématiques).

Le premier niveau de segmentation consiste à séparer les différents blocs. Le second niveau est celui de la séparation d'un texte ou d'un graphique en éléments constitutifs. La couche de graphique peut être séparée en plusieurs couches distinctes selon des critères géométriques et structurels (traits forts et fins, lignes tiretées, traits mixtes, hachurage...) [**10**]. La couche de texte, objet de notre étude, est séparée en caractères.

Le module de segmentation constitue une phase intermédiaire entre le prétraitement et la reconnaissance. Il permet dans notre cas, de localiser les éléments du texte arabe, à savoir les lignes, les parties connexes, et de séparer ces dernières en caractères isolés. La segmentation d'une partie connexe en lettres est une opération naturelle, mais elle est extrêmement complexe, en particulier pour l'écriture manuscrite cursive [**11**]

La réussite de la reconnaissance de caractères dépend fortement de la bonne segmentation des lettres dans un mot. Nous pouvons donc dire que c'est une des étapes critiques pour la suite des traitements prévus pour les lettres. La segmentation de l'image se fait en trois étapes [**12**] et [**13**] :

*Segmentation horizontale:*

La page de texte est segmentée horizontalement dans le but de localiser les lignes de texte et de les séparer les unes des autres.

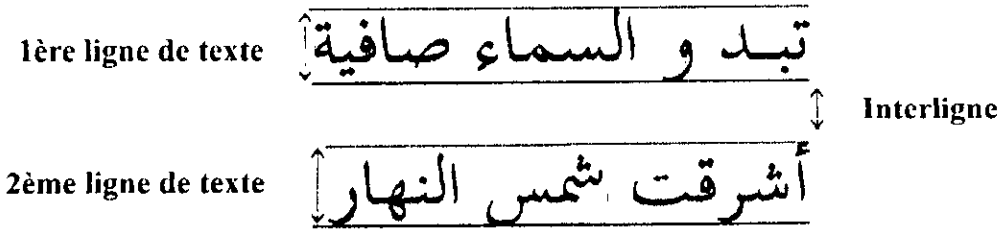



**Figure - 26 - Localisation des lignes de texte**

*Segmentation verticale:*

Chaque ligne de texte est segmentée en parties connexes ou éléments de mots. Une partie connexe contient par définition, soit un caractère, soit plusieurs reliés entre eux, et n'ayant pas forcément un sens dans la langue arabe. Un mot, qui est un ensemble de caractères attachés ou non, et ayant une signification dans la langue peut comprendre une ou plusieurs parties connexes.

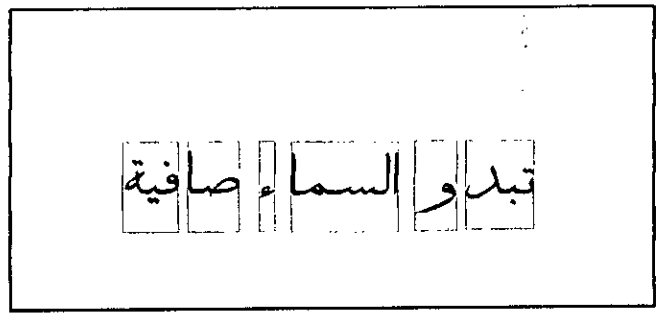



**Figure - 27 - : Délimitation des parties connexes**

*Segmentation en caractères:*

Chaque partie connexe est séparée en caractères isolés.

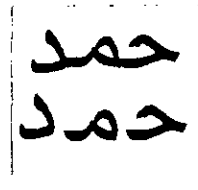



**Figure -28- : Exemple de segmentation en caractères**

### III.2. Problèmes de la segmentation :

La réalisation d'un système OCR se voit confrontée à un ensemble de problèmes rencontrés au niveau de la segmentation ou même au niveau de la reconnaissance. Dans certains cas, l'homme lui-même ne peut enlever quelques ambiguïtés que par une étude du contexte ou se trouve le symbole traité.

L'un des problèmes majeurs de la reconnaissance vient de la difficulté de séparer correctement les caractères et à gérer les problèmes de ligature et de Kerning **[15]**. Ces difficultés sont inhérentes à la typographie et peuvent être définis comme suit :

- *Ligature* : Deux caractères contigus sont reliés l'un à l'autre sur la même ligne ou de ligne en ligne **[15]**.
- *Kerning* : Deux caractères voisins peuvent empiéter l'un sur l'autre **[15]**.

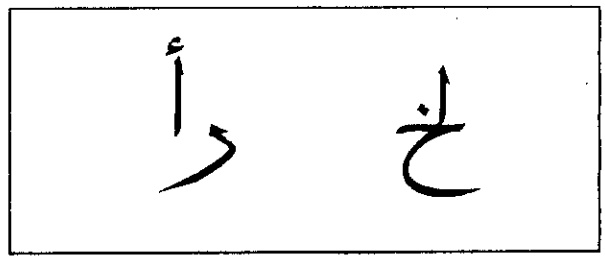


**Figure -29- a- Kerning b- Ligature**

La présence de bruit sur une page: tâches ou irrégularités de caractères, peut empêcher une segmentation efficace d'un texte en caractères. D'autres problèmes peuvent être liés aux caractéristiques propres de chaque langue, ce qui complique davantage la mise au point de solutions multilingues.

La langue arabe, de part sa grande richesse, pose jusqu'à nos jours des problèmes en OCR. Son traitement est presque dual au traitement du latin manuscrit, vu sa cursivité et d'autres propriétés que nous étudierons dans le paragraphe suivant.

### III.3. Caractéristiques de l'écriture arabe:

Contrairement aux caractères latins, les caractères arabes sont écrits de droite à gauche de manière cursive, c'est à dire que les lettres sont généralement liées entre elles. L'alphabet arabe comprend 29 caractères, ce qui ne veut pas dire qu'il comprend le même nombre de formes différentes. Ceci étant du au fait que certains caractères peuvent prendre plusieurs formes **[8]**.

Globalement, nous pouvons résumer les caractéristiques de l'écriture arabe par les points qui suivent :

- Chaque caractère peut prendre quatre formes différentes suivant qu'il se trouve au début, au milieu, à la fin d'un mot, ou isolé. La figure **(31),** illustre les formes des différents caractères arabes selon leur position.

- Certains caractères différents ont la même forme, mais se distinguent par la position et le nombre de points qui leur appartiennent.

- Les voyelles ne sont pas systématiquement utilisées dans l'écriture arabe; des signes qui correspondent aux voyelles (vocalisation) sont employés pour éviter des erreurs de prononciation. On peut distinguer deux types de textes: avec ou sans les signes de voyelles. Quelques textes arabes (le Coran et les livres d'apprentissage pour enfant) contiennent la vocalisation, les autres (les livres, les journaux, les publications) n'en contiennent pas.

- L'écriture arabe comprend une large gamme de fontes, toutes très utilisées.

- Certains caractères se chevauchent, c'est à dire qu'il est impossible d'encadrer un caractère dans un rectangle sans croiser son successeur, ou de faire passer une sonde verticale entre deux caractères successifs. Ce qui rend la séparation des caractères très difficile. On parle dans ce cas de ligatures. Par exemple dans le mot "Mohamed", les lettres "mim" et "Ha" se chevauchent. Le tableau de la figure **(30)** illustre quelques exemples de ligatures.

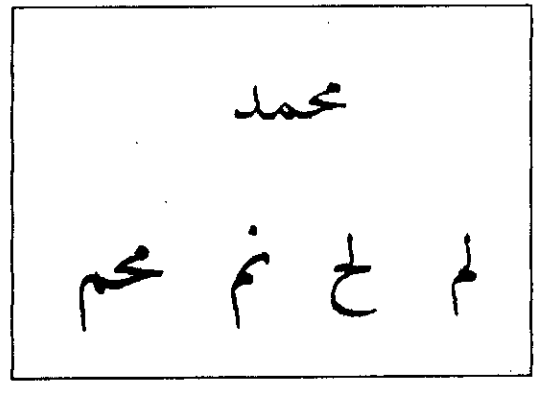


**Figure - 30 - : Quelques exemples de ligatures**

| Début | Milieu | Fin | Isolé | | Début | Milieu | Fin | Isolé |
|---|---|---|---|---|---|---|---|---|
| أ | ا | ا | أ | | ع | ع | ع | ع |
| ب | بـ | ب | ب | | غ | غ | غ | غ |
| ت | تـ | ت | ت | | ف | ف | ف | ف |
| ث | ثـ | ث | ث | | ق | ق | ق | ق |
| ج | جـ | ج | ج | | ك | ك | ك | ك |
| ح | حـ | ح | ح | | ل | ل | ل | ل |
| خ | خـ | خ | خ | | م | م | م | م |
| د | د | د | د | | ن | ن | ن | ن |
| ذ | ذ | ذ | ذ | | ه | ه | ه | ه |
| ر | ر | ر | ر | | و | و | و | و |
| ز | ز | ز | ز | | ي | ي | ي | ي |
| س | سـ | س | س | | ة | ة | ة | ة |
| ش | شـ | ش | ش | | ئ | ئ | ئ | ئ |
| ص | صـ | ص | ص | | ؤ | ؤ | ؤ | ؤ |
| ض | ضـ | ض | ض | | لا | لا | لا | |
| ط | طـ | ط | ط | | لأ | لأ | لأ | |
| ظ | ظـ | ظ | ظ | | | | | |

**Figure - 31 - Différentes formes des caractères arabes**

### III.4. Notion d'histogramme :

Les images binaires se présentent sous forme de matrice dont les éléments sont soit "0" pour la couleur de fond (blanc), soit "1" pour le texte (noir). Une image de taille Nx P, contient N lignes de pixels et P colonnes de pixels.

L'histogramme est une fonction, qui pour chaque ligne (respectivement colonne) d'indice i (respectivement j), fournit le nombre de pixels allumés. Ceci est réalisé au moyen de projections sur les deux axes horizontal et vertical selon les formules qui suivent [14]:

$$h(i) = \sum_{j} g(i,j)$$

$$v(j) = \sum_{i} g(i,j)$$

où h (i) est la projection horizontale de la ligne d'indice i.

v (j) est la projection verticale de la colonne d'indice j.

g (i, j ) représente la valeur du pixel de coordonnées (i,j).

i est l'indice de la ligne.

j est l'indice de la colonne.

L'histogramme est représenté par un graphe dans lequel les différentes positions de l'image sont portées en abscisses, et le nombre de pixels allumés par ligne (respectivement par colonne) en ordonnées.

La figure **(32)** montre les histogrammes horizontal et vertical d'un texte arabe.

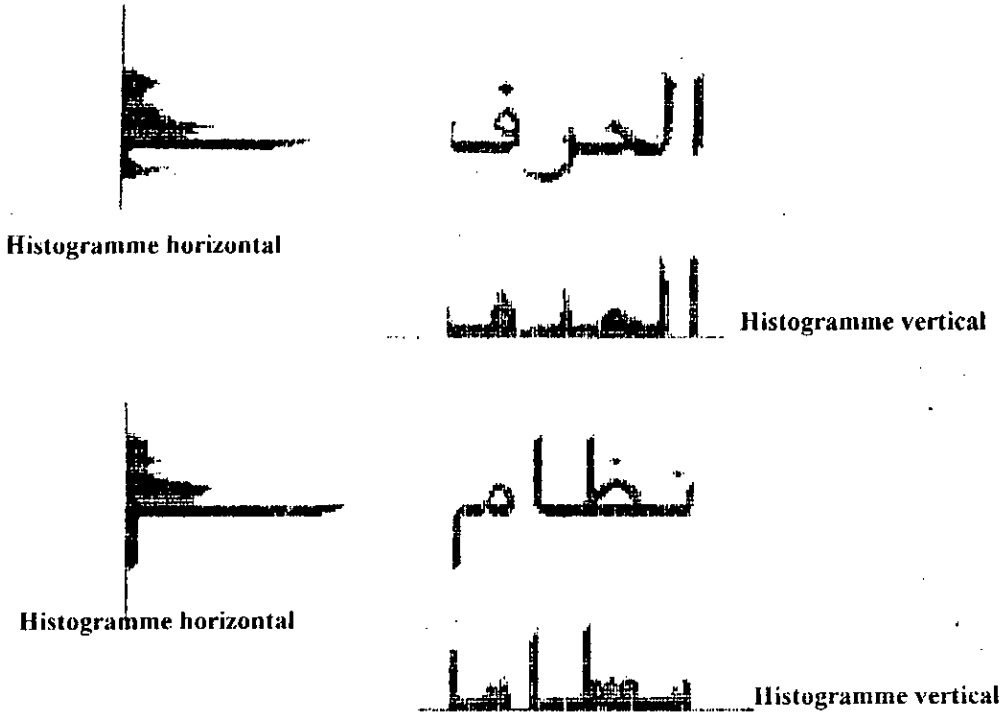



**Figure-32- : Histogrammes d'un texte arabe**

L'histogramme détient un grand nombre d'informations sur l'image, notamment sur la distribution des niveaux de gris. Il permet aussi de détecter certaines zones pour les images binaires telles que les zones à forte concentration de pixels allumés, les zones de silence pour lesquelles la concentration de pixels ne varie pas ou varie très peu, etc...

Les calculs des histogrammes horizontal et vertical sont réalisés selon les algorithmes suivants:

*Algorithme histogramme vertical*

**Début**

**Pour** chaque colonne j **Faire**

histv(j) ← 0

**Pour** chaque ligne i **Faire**

histv(j) ← histv(j) + g(i,j)

**Finpour**

**Finpour**

**Fin**

*Algorithme histogramme horizontal*

**Début**

**Pour** chaque ligne i **Faire**

histh(i) ← 0

**Pour** chaque colonne j **Faire**

histh(i) ← histh(i) + g(i,j)

**Finpour**

**Finpour**

**Fin**

## III.5. Quelques méthodes de segmentation

### III.5.1. Méthode de L. Boukined, B. Taconet, A. Zahour, A. Faure

En 1991, à l'université du Havre, ce groupe de chercheurs a développé une méthode de segmentation d'un document de texte latin imprimé en entités physiques simples (mot) de façon descendante **[16]**. L'opération est réalisée sur image contenant plusieurs blocs par rectangularisation et permet en premier lieu de séparer les blocs les uns des autres, ensuite, chaque bloc est séparé en lignes puis en mots.

Les espaces verticaux séparateurs sont les marges de gauche et de droite, et les espaces intercolonnes de textes. Les espaces horizontaux sont les marges supérieure et inférieure, et les espaces horizontaux interblocs.

Chaque élément (bloc) est localisé par les coordonnées de son enveloppe. La dimension des séparateurs horizontaux dépend de l'interligne et de la hauteur d'une ligne, une estimation du seuil à 1,5 unité convient dans la majorité des cas. Pour ce qui des séparateurs verticaux qui doivent être plus élevés que l'espacement entre les mots de la police la plus espacée, un seuil d'environ 2,5 unités est correct.

La segmentation d'un bloc en lignes est liée à la présence d'une ligne de pixels vide entre deux lignes de texte, ce qui rend cette opération sensible à la présence de points noirs isolés, et met en évidence la nécessité d'un prétraitement.

Pour l'extraction d'un mot à partir d'une ligne, le seuil vertical est la largeur du rectangle enveloppant la ligne, et le seuil horizontal a été choisi manuellement de 1 / 2,5 du seuil vertical.

Cette méthode de segmentation par rectangularisation a permit de séparer correctement une image de texte en mots, cependant, les difficultés principales sont l'automatisation des seuils, la segmentation d'un document contenant du graphique, ainsi que la séparation d'un mot en caractères.

### III.5.2. Méthode de K. Bouhlila, M.K. Hamrouni, N. Ellouze

Dans le cadre de la recherche sur l'écriture arabe, ces trois chercheurs ont développé au laboratoire des systèmes et de traitement du signal en Tunisie, une méthode de segmentation d'un texte arabe en caractères, en se basant principalement sur la notion d'histogramme définie au paragraphe **III-4**. La méthode traite des documents scannés avec une résolution de 300 dpi, n'ayant pas subit de traitements préliminaires tels que le filtrage**[12]**.

La séparation du paragraphe en lignes est d'abord réalisée par construction de l'histogramme horizontal de la page de texte. Celui-ci est une fonction multi-modale, pour laquelle les modes maximums correspondent à la ligne de texte et les modes minimums l'espacement entre deux lignes successives. Un retour arrière dans la programmation est nécessaire pour lever quelques ambiguïtés concernant la localisation d'une ligne qui, réellement correspond à la suite de la ligne précédente (points diacritiques), ou à un bruit.

L'histogramme vertical permet de localiser les parties connexes pour lesquels une ligne de référence qui correspond au maximum de l'histogramme horizontal est calculée. Le début d'un caractère qui se manifeste par une forte concentration de pixels, est détecté par l'équation V(i) > M x C ou V(i) est l'histogramme vertical de la partie connexe dans la zone voisine à la ligne de référence, et M est la moyenne de l'histogramme. Le coefficient C a été fixé expérimentalement à la valeur 1,5.

Des tests supplémentaires seront effectués selon la largeur du caractère afin de vérifier la séparation effective des lettres. L'avantage majeure de cette méthode, est l'utilisation d'un coefficient pouvant être ajusté à une fonte donnée, son taux de reconnaissance est estimé à 96 %.

### III.5.3. Méthode de K. Roméo Pakker, A. Ameur

Cette méthode, proposée à l'université de Rouen, est fondée sur les propriétés contextuelles propres à l'écriture arabe. D'abord la segmentation d'un texte en lignes est réalisée au moyen des projections horizontales, ensuite, la localisation des mots en considérant les espaces entre les lettres et les mots est effectuée.

La détection des caractères dans un mot, plus délicate que les deux autres, est fondée sur le trait de liaison entre les caractères. Le trait de liaison est caractérisé par sa faible épaisseur (en nombre de pixel) et sa position sur la ligne de base. Ce seuil de segmentation n'est en aucun fixe, mais varie selon l'écriture.

Cette méthode traite le cas des caractères qui se chevauchent par un suivi de contour selon les huit direction de Freeman en même temps qu'un étiquetage des contours détectés[17].

Le caractère « Sin » est sursegmenté en une succession de trois petits segments verticaux sans points qui seront reconnus comme tels. Le taux de segmentation est évalué à 98,9 % avec quelques caractères comme le « Ain », « Gain », « Sin » et « Noun » qui ont les appendices en fin de mot séparés du caractères.

### III.5.4. Méthode de ADNAN AMIN

En 1980, Adnan Amin suggère une méthode basée sur la morphologie de l'écriture arabe imprimée[18]. La séparation du texte en lignes et en parties connexes est effectuée par projections horizontale et verticale. Pour la dernière phase de la segmentation, il démarre de la constatation que les caractères arabes présentent un trait épais à leur naissance, et un amincissement à la fin, où ils sont liés à leurs caractères successeurs.

Cette méthode consiste à trouver la stabilité horizontale d'une partie connexe en utilisant la notion d'histogramme.

La première opération consiste à construire l'histogramme vertical de la partie connexe à segmenter, noté v ( j ) avec j varie de 1 à width où width représente la largeur de la partie connexe.

La moyenne des colonnes de l'histogramme Moy est ensuite calculée par :

$$Moy = \frac{1}{width} \sum_{j=1}^{width} v(j)$$

Partant de la première colonne de droite à gauche comme l'exige l'écriture arabe, on construit la matrice caractère tant que l'histogramme v est supérieur à la moyenne Moy. Dès qu'une colonne j d'histogramme inférieur à Moy est rencontrée, le remplissage s'arrête et toutes les colonnes possédant la même propriété sont ignorées.

La figure **(33)** illustre l'organigramme de segmentation d'une partie connexe en caractères par la méthode de **A.Amin**.

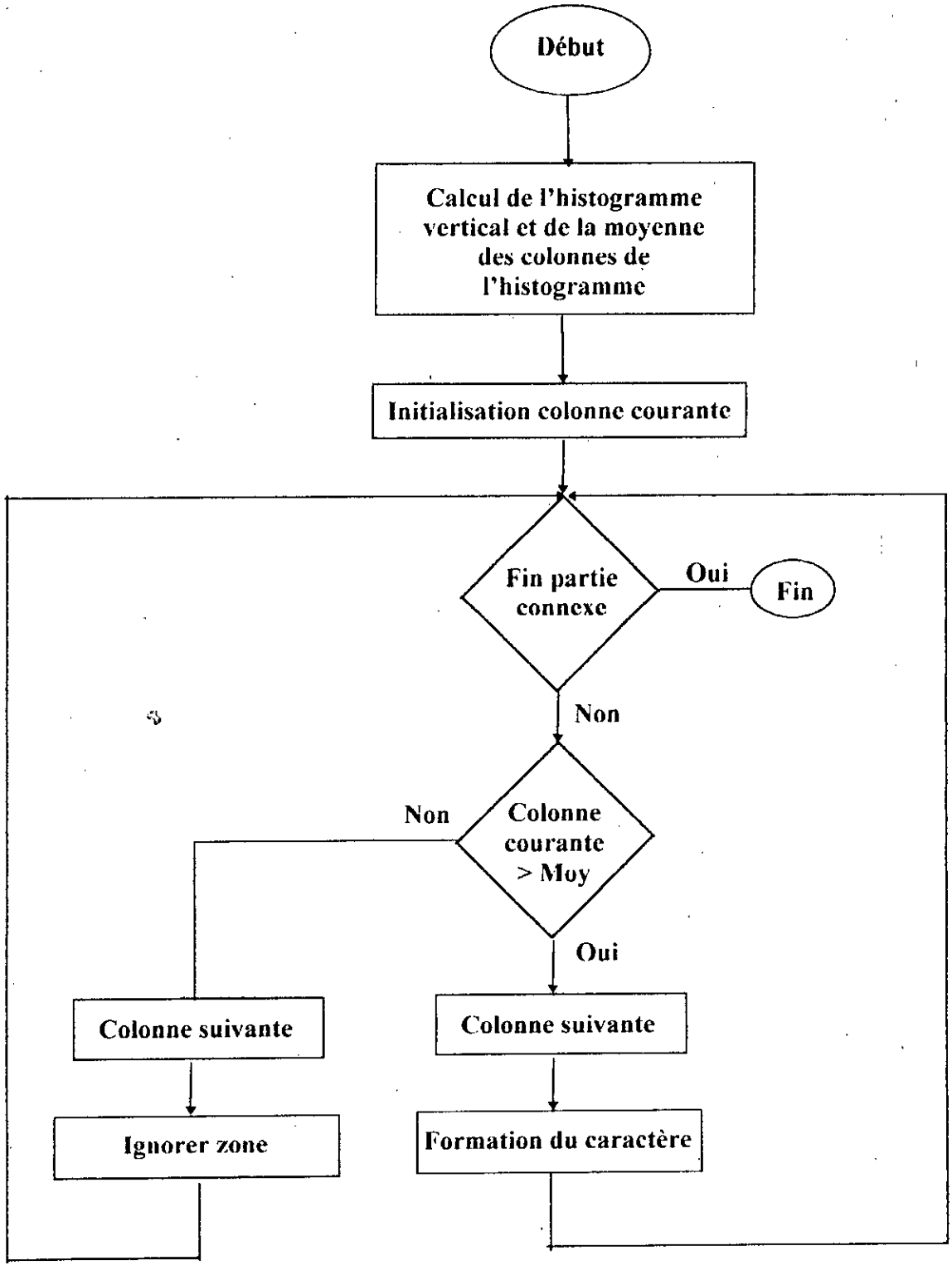



**Figure - 33 - : Organigramme de la méthode de A. AMIN**

## III.6. Conclusion

Les différentes méthodes de segmentation citées dans ce chapitre concernent les documents écrits, imprimés ou dactylographiés, et en particulier en langue arabe, qui se distingue du latin principalement par sa cursivité et le sens de l'écriture.

Ce qui complique encore la tâche de traitement d'un texte arabe est la grande variété de fontes , toutes très utilisées, et d'autres propriétés de cette langue.

La phase la plus difficile et la plus délicate de la segmentation est celle qui consiste à séparer les éléments de mots (ou parties connexes) en caractères. Cela est du au fait que les lettres peuvent être liées ou pas à leur successeur. L'opération est d'autant plus difficile lorsqu'on vise une segmentation sans faute qui nécessite un calcul très précis des seuils de segmentation. Ces derniers ne peuvent en aucun cas être fixes mais ajustables à la fonte étudiée.

Les taux de segmentation varient d'une méthode à une autre. Les problèmes aussi sont propres à chaque méthode.

Dans le chapitre suivant, nous abordons la segmentation d'un texte en caractères, en prenant en considération, en premier lieu la constatation faite par A. Amin, et en s'inspirant des caractéristiques de l'écriture arabe étudiées par Roméo Pakker et A. Ameur ainsi que K. Bouhlila, M.K. Hamrouni, N. Ellouze.

# CHAPITRE IV

## MÉTHODE UTILISÉE, RESULTATS ET INTERPRETATIONS

# CHAPITRE IV
# METHODE UTILISEE , RESULTATS ET INTERPRETATIONS

## IV.1. Introduction :

Dans ce chapitre, nous abordons le problème de la segmentation d'un paragraphe de texte arabe imprimé en caractères. Les documents traités sont scannés avec une résolution de 300 dpi, et comme mentionné au chapitre II, ont subit un prétraitement qui est le filtrage.

Nous avons à cet effet, expérimenté en premier lieu la méthode basée sur la constatation faite par A. Amin. En second lieu, nous avons essayé de rendre l'équation de détection du début et de la fin des caractères plus souple en introduisant un coefficient ajustable empiriquement [12].

Enfin, une méthode plus complète et proposée, elle tient compte des propriétés de l'écriture arabe et est fondée sur l'utilisation des histogrammes des lignes et des colonnes ainsi que d'autres notions que nous définirons plus loin dans ce chapitre.

## IV.2. Segmentation horizontale :

Avant d'aborder le problème de la segmentation horizontale, il est nécessaire de faire la distinction entre une ligne de texte et une ligne pixel.

Une ligne pixel, est un ensemble de pixels adjacents horizontalement pouvant prendre chacun la valeur "0" ou "1" selon qu'il s'agisse de l'écriture ou du fond [12].

Une ligne de texte est un ensemble de lignes pixels se trouvant les une au dessous des autres.

La segmentation horizontale, dont le but est de localiser les lignes de texte, est réalisée en projetant horizontalement la page de texte sur un axe verticale. L'histogramme ainsi obtenu, présente des zones non nulles qui représentent les lignes pixels qui rentrent dans la constitution de la ligne de texte, et des zones nulles qui représentent les interlignes.

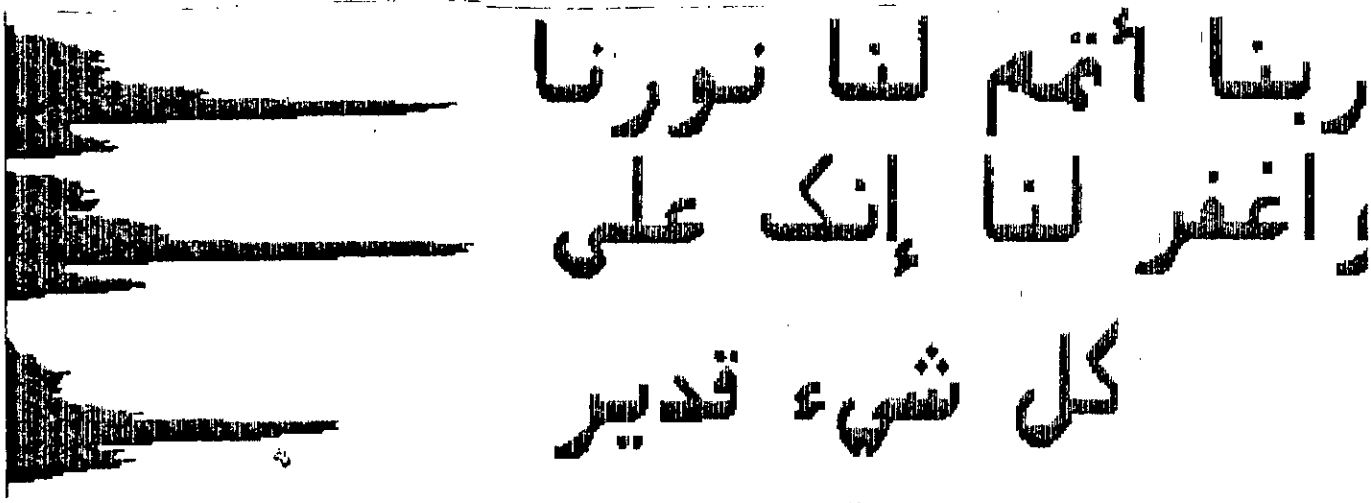



**Figure - 34 - : Délimitation des lignes**

Une première approche a consisté à supposer que deux lignes de texte sont séparées par au moins une ligne pixel vide. La segmentation est alors réalisée selon l'organigramme de la figure (35), en parcourant l'histogramme de haut en bas. Le début d'une ligne de texte correspond à une transition valeur nulle/valeur non nulle. Toutes les lignes pixels ayant une valeur non nulle de l'histogramme constituent la ligne de texte, dont la fin est détectée par une transition inverse, c'est à dire valeur non nulle/valeur nulle de l'histogramme.

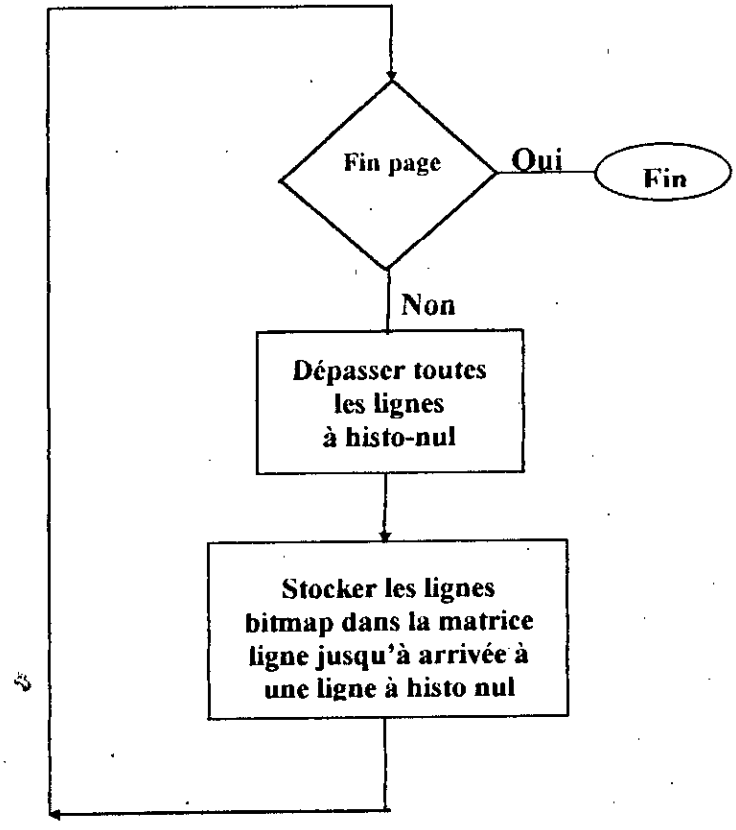



**Figure - 35 - : Organigramme de la segmentation horizontale**

Les résultats obtenus par cette opération montrent que les lignes d'un texte sont correctement segmentées lorsque l'histogramme présente effectivement des valeurs nulles au niveau de l'interligne. Néanmoins, elle reste très sensible au bruit et au problème de chevauchement des lignes.

Dans le cas de l'écriture arabe, deux lignes consécutives ne sont pas obligatoirement séparées par des lignes pixels ou tous les éléments sont éteints. Cela se produit comme l'illustre la figure (36), lorsque sur la ligne supérieure se trouve un caractère bas tel que "mim" et sur la ligne inférieure un caractère haut tel que "alif" ou "lam".

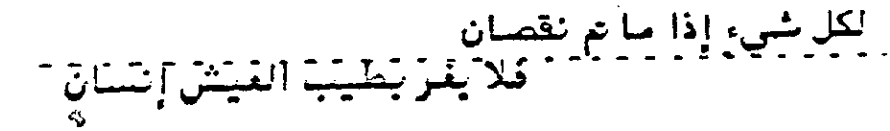



**Figure - 36 - : Exemple de deux lignes de texte se chevauchant**

Dans une telle situation, l'algorithme cité interpréterait les deux lignes comme une seule, ce qui fausserait tous les traitements ultérieurs. Pour corriger ce problème, il est nécessaire de fixer un seuil de segmentation expérimentalement. Ce dernier dépendra essentiellement de la taille de l'image ainsi de la fonte utilisée.

Un autre problème propre à l'écriture arabe se matérialise par le fait que les points diacritiques ou les signes de voyelles (pour les textes qui en possèdent) peuvent former à eux seuls une ligne de texte. Ce cas est rarement rencontré, car dans une ligne de texte arabe, la fréquence d'apparition des caractères longs tel que "alif" est assez importante, ce qui empêcherait d'interpréter les points de la lette "ta" par exemple comme une ligne.

Dans l'exemple أبي , les points au dessous de "ya" forment réellement une ligne de texte si ce mot se trouve seul surlligne écrite.

Pour éviter un tel défaut de segmentation, qui n'est pas très grave car au niveau de la reconnaissance, les points feront automatiquement objet d'un rejet, on juge qu'un ensemble de lignes pixels consécutives et remplissant les conditions de segmentation horizontale citées précédemment, constituent une ligne de texte si leur taille est comparable aux autres lignes de textes.

D'après une étude sur l'écriture arabe, il a été constaté que les lettres "alif" et "lam" présentaient la plus grande hauteur sur une ligne et qui serait cinq fois plus élevée que la hauteur d'une "nabira". Par conséquent, nous calculons la taille de chaque ligne qu'on comparera à celle possédant la plus grande hauteur. Une ligne dont les dimensions seraient cinq fois moins élevées que la taille maximale des lignes, ne sera pas considérée comme telle (comme une ligne de texte).

### IV.3. Segmentation verticale :

La séparation d'une ligne de texte en parties connexes ou élément s de mots est réalisée comme le montre l'algorithme de la figure (**38**), en parcourant l'histogramme vertical de la ligne colonne par colonne. Celui - ci présente des zones de silence à valeur minimale qui serviront à délimiter les parties connexes


**Figure - 37 - : Délimitation des parties connexes**

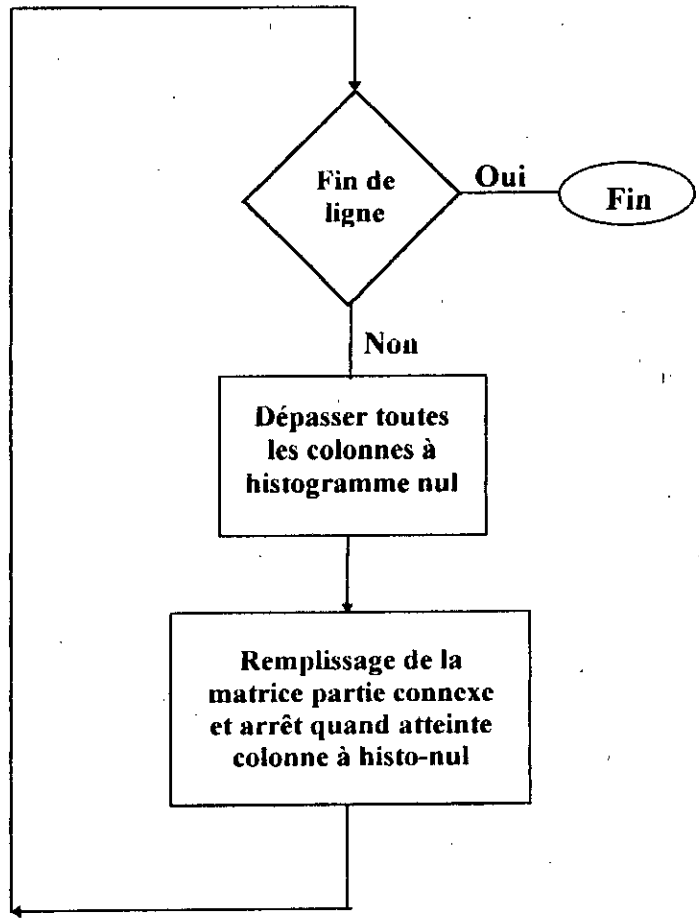



**Figure - 38 - : Organigramme de la segmentation verticale**

## IV.4. Segmentation en caractères :

### IV.4.1. Méthode de A. Amin :

Nous avons expérimenté la méthode de A. Amin décrite au chapitre III paragraphe (5-4) sur un ensemble de documents ne contenant qu'un paragraphe de texte. La localisation des caractères est réalisée en comparant les colonnes d'un élément de mots à leur moyenne. Cette méthode a l'avantage d'être simple à mettre en oeuvre. Cependant, les résultats obtenus montrent plusieurs insuffisances. La figure (**39**) illustre un exemple d'un texte segmenté par cette méthode.

Il est à noter que certains caractères, sont coupés en plusieurs parties (sur-segmentés), ceci est dû au fait que un amincissement n'annonce pas toujours la fin d'un caractère. Certaines lettres peuvent présenter des épaisseurs au dessous de la moyenne avant d'atteindre leur fin.

Pour des caractères à l'histogramme vertical uniforme, comme le montre la figure (**40**), la segmentation donne des résultats satisfaisants.

**Ligne** **Détection parties connexes** **Détection caractères**

**Figure - 40 - : Méthode de A. Amin**

Pour prendre en considération la remarque que certains caractères pouvaient présenter des épaisseurs au dessous de la moyenne avant d'atteindre leur fin, nous avons essayé de rendre cette méthode plus souple en introduisant un facteur multiplicatif (coefficient) qui permettrait d'éviter les fausses régions de segmentation. Nous sommes arrivé expérimentalement à un facteur de 0.8. Une valeur plus grande cause des cas de sur-segmentation. Une valeur plus petite des cas de sous-segmentation, c'est à dire que deux caractères ou plus peuvent être interprétés comme un seul. (figure (39-b))

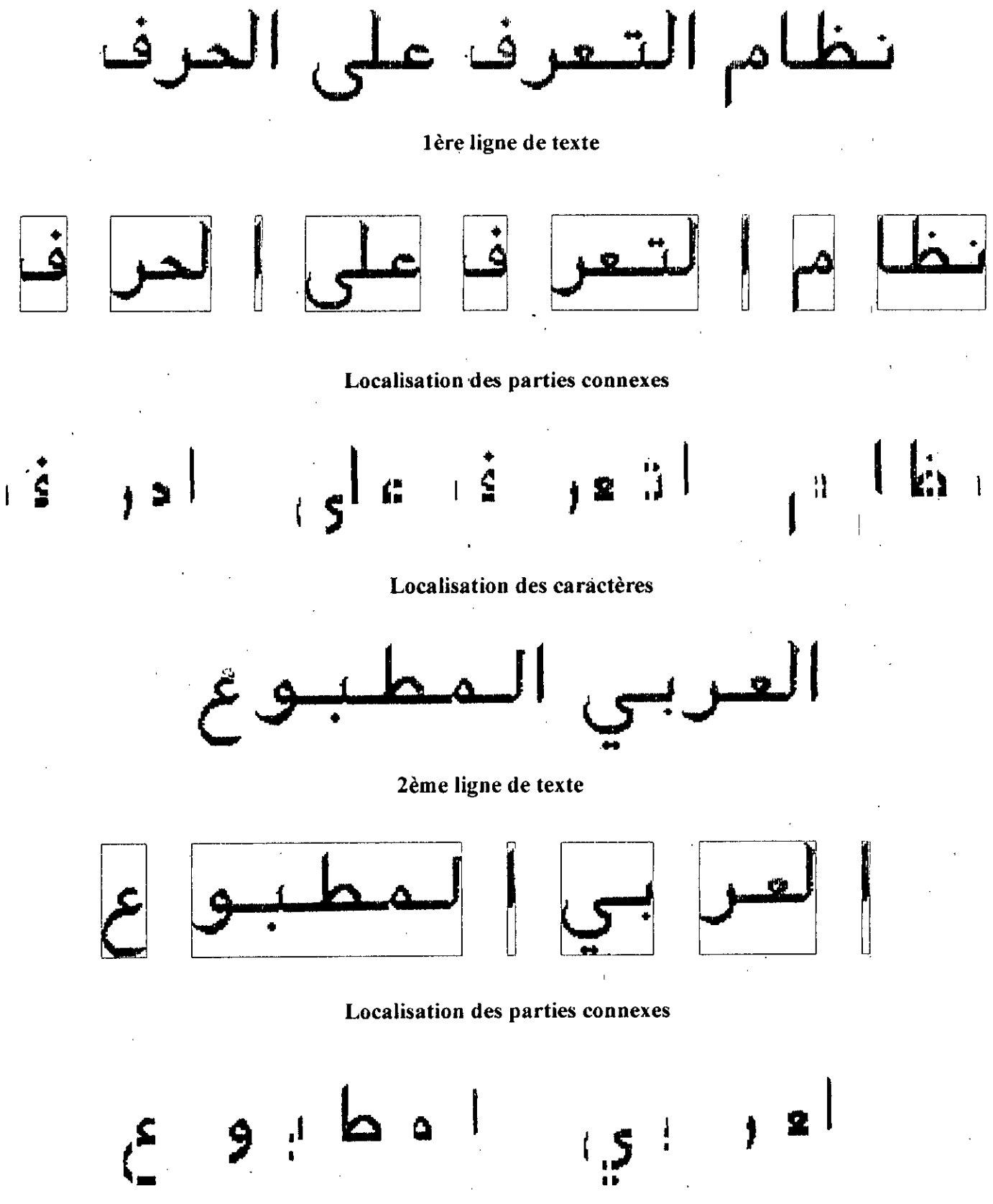
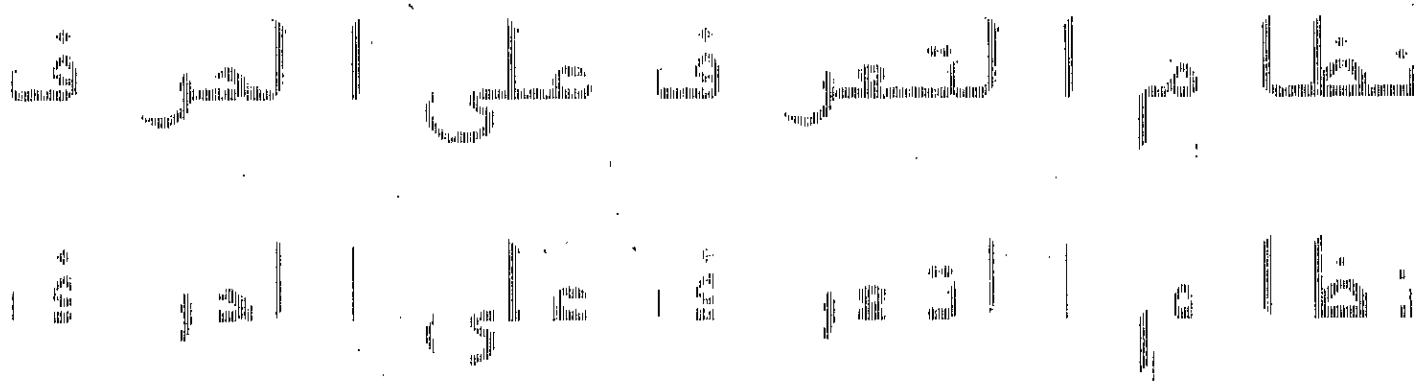



Figure - 39 - Exemple de texte segmenté par la méthode de A. Amin

نظام التعرف على الحرف

1ere Ligne de texte



العربي المطبوع

2ème Ligne de texte

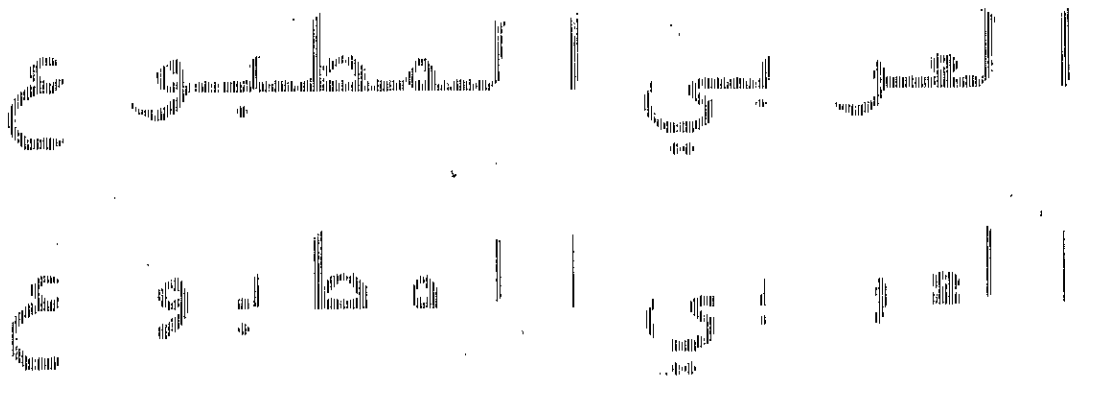


Figure 39_b Méthode A.Amin avec Coefficient 0.8

### IV.4.2. Méthode utilisée :

Cette méthode de séparation d'un élément de mot ou partie connexe en caractères est similaire, dans une certaine mesure, à la méthode citée précédemment. Cette similitude est liée au critère de détection de la fin et du début d'un caractère. Dans le paragraphe précédent, il s'agissait de détecter le début d'un caractère lorsque l'histogramme d'une colonne était supérieur à la moyenne des colonnes, et sa fin lorsque l'histogramme chutait au dessous de la même moyenne. Dans ce qui suivra, l'histogramme n'est plus comparé à une moyenne mais à un seuil calculée pour chaque partie connexe.

Nous utiliserons, aussi des tests supplémentaires pour s'assurer que le caractère est effectivement séparé. A cet effet, nous avons usé de notions que nous avons jugé utile de définir avant d'aborder la dernière phase de la segmentation.

#### IV.4.2.1. Ligne médiane :

La ligne médiane d'une partie connexe est la ligne à plus forte concentration de pixels. Sa localisation réalisée par parcours de l'histogramme horizontal et la recherche de la ligne qui présente la plus grande valeur de la fonction histogramme. La figure (41) illustre un exemple de la ligne médiane d'une ligne de texte.

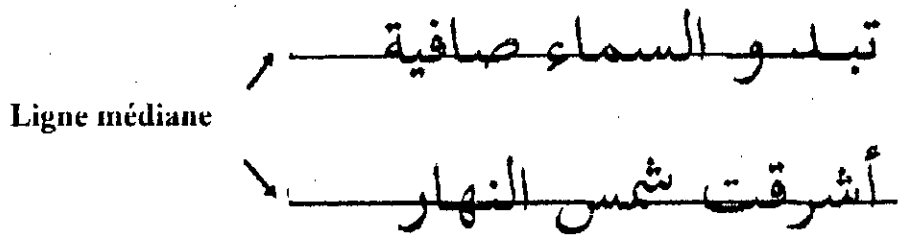
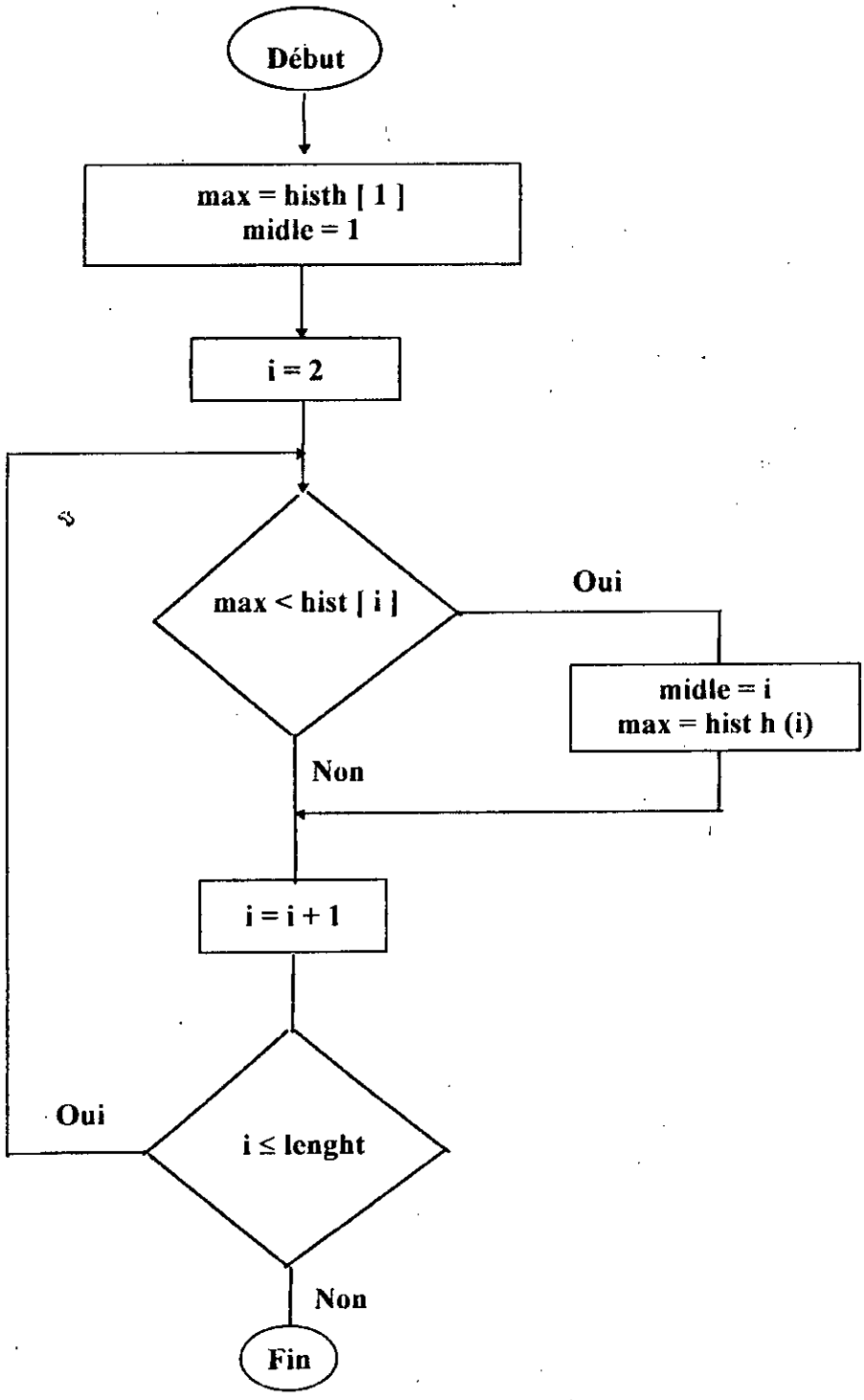



**Figure - 41 - : Ligne médiane d'une ligne de texte**

Il faut noter que la ligne médiane d'une ligne de texte ne correspond pas toujours à la ligne médiane d'une partie connexe, d'où la nécessité d'un réajustement.

Plusieurs lignes peuvent avoir la même valeur maximale de l'histogramme et donc à être qualifiées de médiane.

Pour notre part, nous commençons la recherche de haut en bas et nous nous arrêtons à la première ligne médiane rencontrée, comme l'illustre la figure (42).




**Lenght :** Hauteur de la partie connexe
**Histh (i) :** Histogramme horizontal de la ligne i
**Midle :** Ligne médiane

**Figure - 42 - : Organigramme de détection de la ligne médiane.**

### IV.4.2.2. Seuil de segmentation :

Le seuil de segmentation en caractères est la largeur du tracé et correspond dans l'histogramme vertical à la valeur minimale la plus redondante. Son calcul se fait selon l'organigramme de la figure (**43**). Ce seuil ne peut en aucun cas être fixe. Il varie d'une partie connexe à une autre, et pour la même partie d'une fonte à une autre.

### IV.4.2.3. Histogrammes des distances (haut et bas) :

Ces histogrammes représentent l'un, la distance entre la ligne médiane et le contour du coté haut, et l'autre la distance entre le contour du coté bas et la ligne médiane.

Pour le contour haut de la partie connexe, il s'agit de parcourir chaque colonne d'indice j de haut en bas. La rencontre de la première ligne i présentant un pixel non nul correspond au contour Vmax(j) = i. Le contour bas (inférieur) Vmin (j) est détecté de la même manière, en cherchant le premier pixel noir (non nul) sur la colonne en partant de la dernière ligne pixel de la partie connexe et en remontant vers le haut. La figure (**44**) montre le principe de calcul des contours.

Les histogrammes des distances sont ensuite calculés pour chaque colonne par différence de la ligne médiane et des contours.

### IV.4.2.4. Description de la méthode :

Des études sur les critères de liaison des caractères arabes sont arrivées expérimentalement à trouver des caractéristiques importantes que nous pouvons résumer en deux points :

- Il a été constaté que deux caractères consécutifs se relient entre eux dans la région centrale de la ligne de texte, on parle alors de la ligne de jonction (ligne médiane)[17].
- Il a été établi aussi que la liaison se fait généralement par des traits de plume avec une direction horizontale ou légèrement inclinée et une épaisseur constante qui représente l'épaisseur du tracé et que nous avons dénommé le seuil [**17**].

La méthode utilisée consiste alors, après calcul de toutes les caractéristiques propres à la partie connexe à segmenter (histogramme de lignes et de colonnes, ligne médiane, seuil, histogrammes des distances) à procéder à la recherche du début et de la fin d'un caractère, en se basant essentiellement sur l'équation de détection colonne courante > seuil.

L'expérimentation de ce critère montre que certains caractères à appendices finaux bas tels que "ya" peuvent présenter de fausses régions de segmentations. D'autres tests sont alors nécessaire pour détecter la fin effective des caractères à appendices finaux.

Les histogrammes des distances sont alors comparés au seuil et fournissent l'information sur l'existence de parties de caractères au dessus ou au dessous de la ligne de jonction. L'organigramme de la méthode est illustré par la figure (**45**).

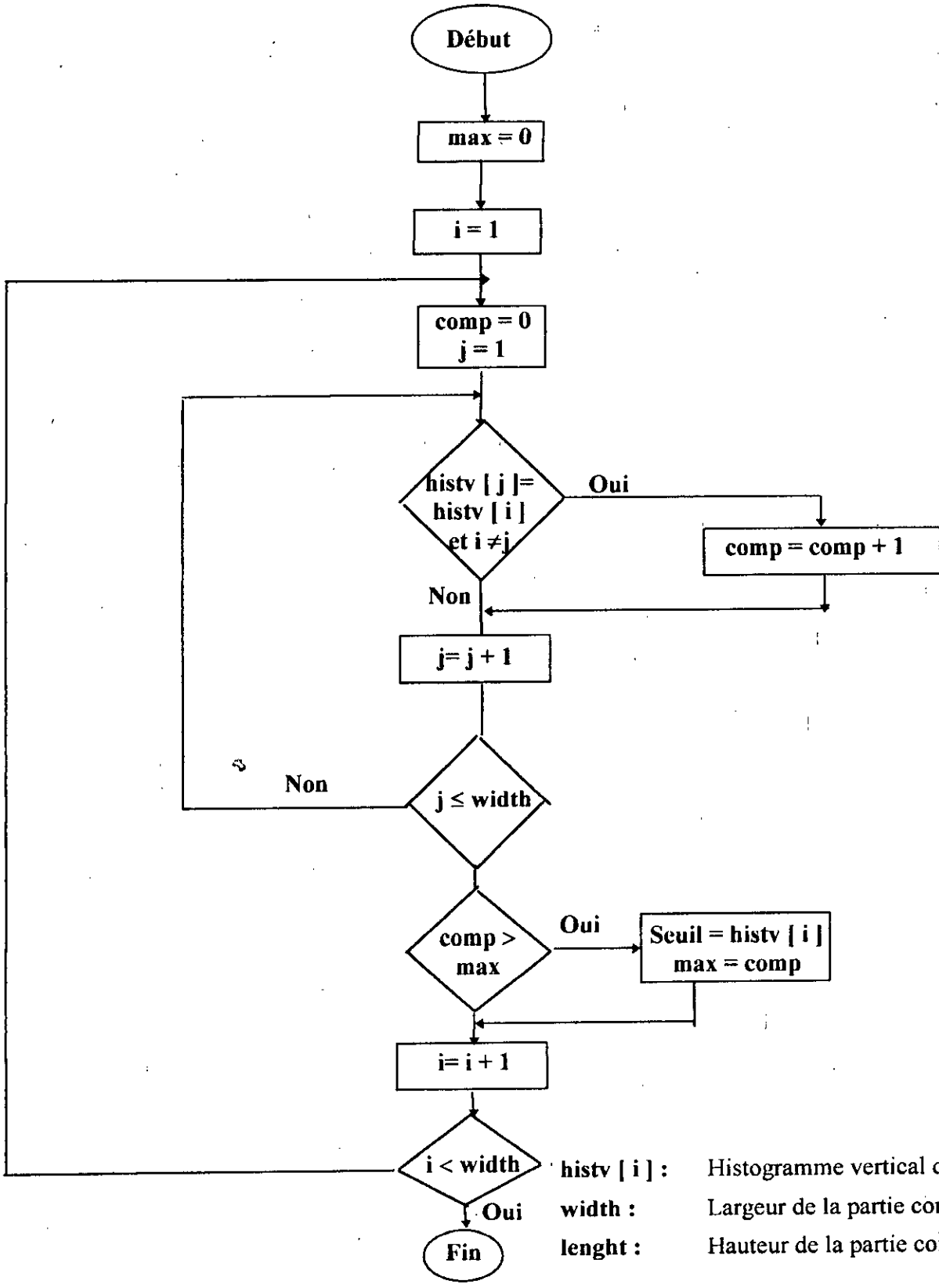
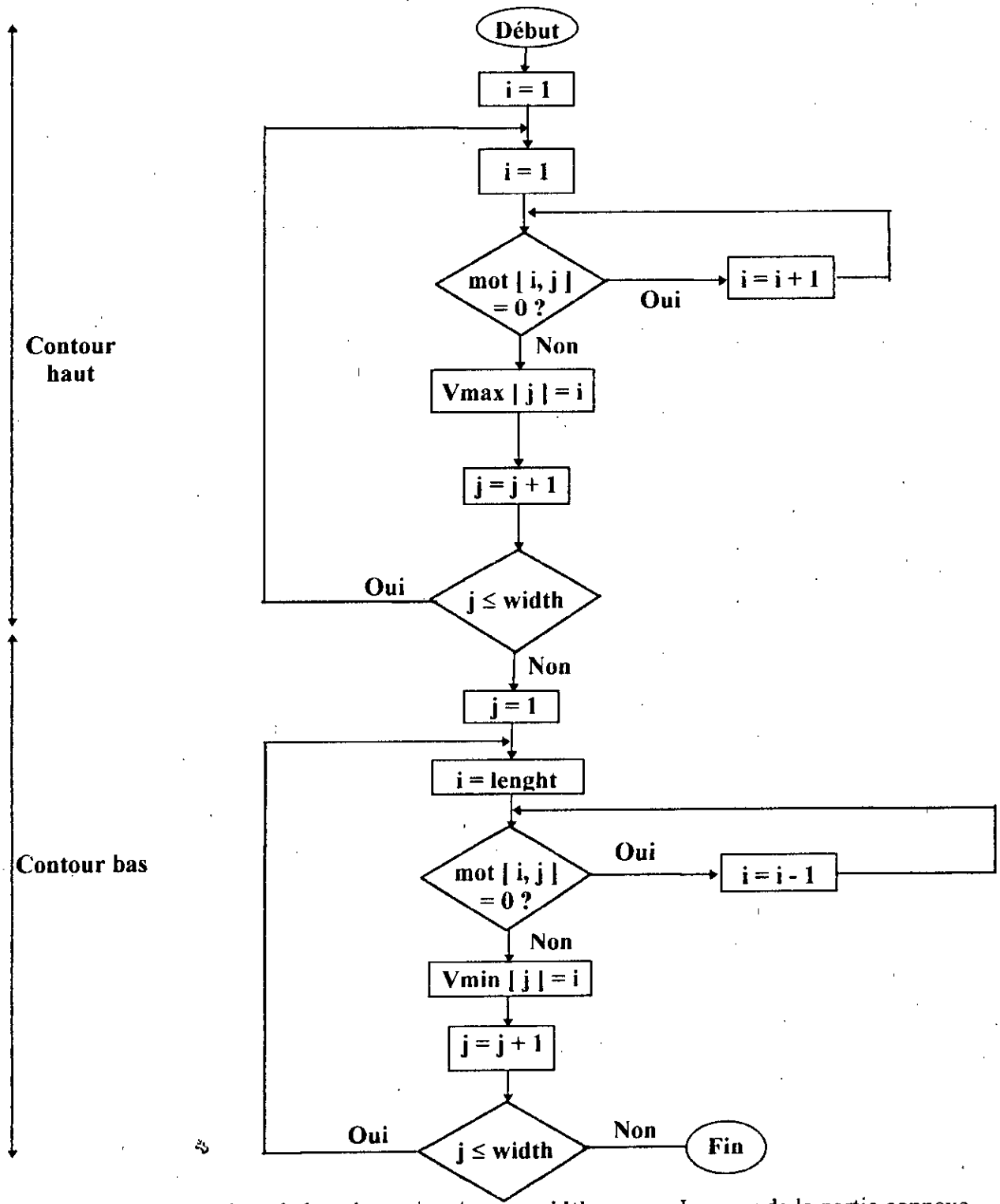



**Figure - 43 - : Organigramme de calcul du seuil.**

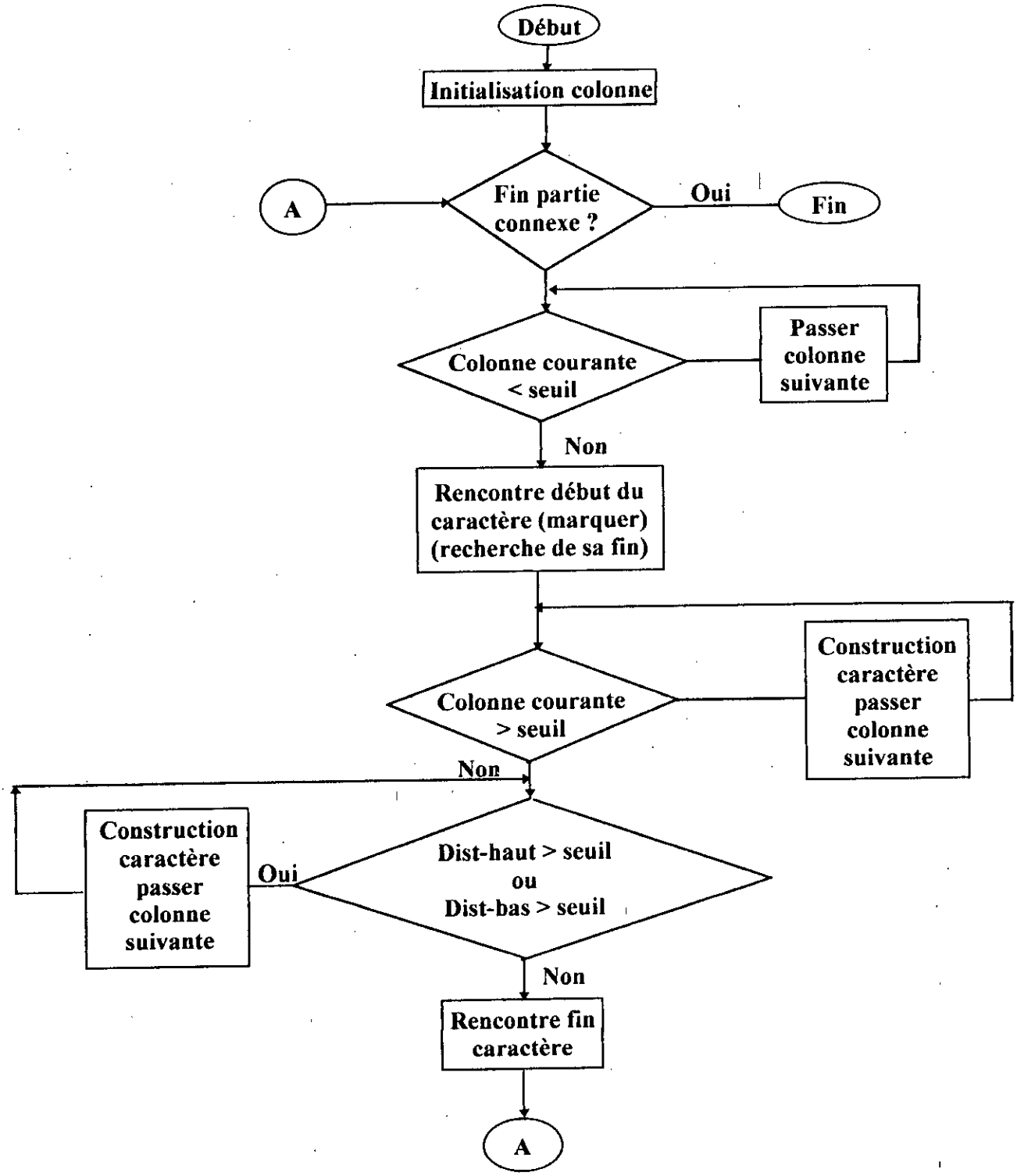



**Vmax [ i ] :** Contour supérieur de la colonne j / **width :** Largeur de la partie connexe

**Vmin [ j ] :** Contour inférieur de la colonne j / **lenght :** Hauteur de la partie connexe

**Mot [ i, j ] :** Valeur de pixel de la partie connexe situé à la ligne i et la colonne j

**Figure - 44 - : Organigramme de détection de contours**




**Dist-haut :** Histogramme des distances haut (Vmax (j) - midle)
**Dist-bas :** Histogramme des distances bas (midle - Vmin (j))

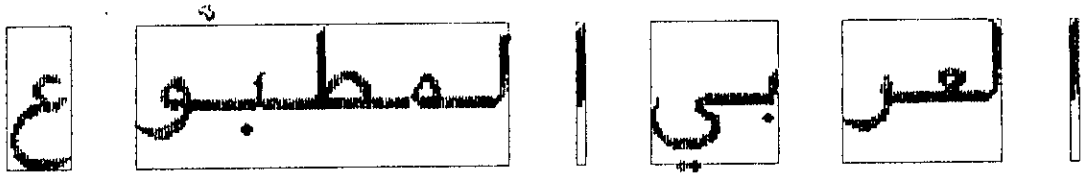
**Figure - 45 - : Organigramme de segmentation en caractères**

1ère ligne de texte

Localisation des parties connexes

Localisation des caractères

2ème ligne de texte



Localisation des parties connexes

Localisation des caractères

Figure - 46 - Exemple de texte segmenté par la méthode amélirée

## IV.5. Conclusion

La méthode utilisée, et après expérimentation sur un ensemble de textes, s'avère beaucoup plus efficace que celle décrite au paragraphe (**IV-4-1**). Les résultats ont montré que la majorité des caractères sont correctement séparés.

Comme nous l'avons prévu, nous ne pouvions nous attendre à une segmentation sans faute. Quelques caractères présentent des fausses régions de segmentation, telle que la lette "sin" qui est coupée entre trois formes identiques. Ainsi que la lettre "kaf" position fin ou isolée qui peut être interprétée comme une succession de la lettre "lam" et "hamza / nabira". Ceci est illustré par la figure (**47**).

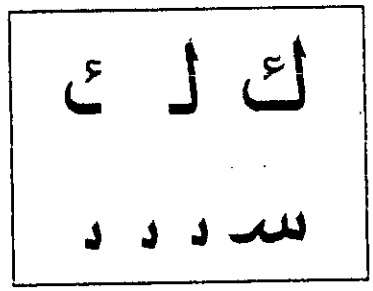
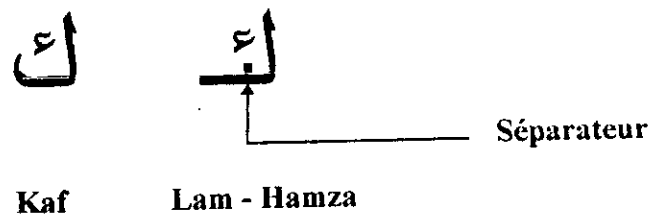


**Figure - 47 -**

L'ambiguïté due à la lettre "sin" peut être levée par le module de reconnaissance qui interprétera trois rejet successifs de la même forme comme étant la lettre "sin".

La lettre "kaf" position fin ou isolée, ne présente pas à la différence de "lam - hamza / nabira", un séparateur comme on le voit sur la figure suivante.




Ce séparateur, qui justement fera l'objet d'un rejet lors de la reconnaissance, sert à différencier les deux cas. L'idée est donc la suivante :

- Si une lettre "lam" reconnue est suivie de la lettre "hamza / nabira", alors on juge que ces deux caractères forment la lettre "kaf".

- Si une lettre "lam" reconnue est suivie d'un rejet ensuite de la lettre "hamza / nabira", alors on juge qu'il y a présence de deux lettres.

Enfin, des ligatures arabes sont traitées dans notre programme comme des lettres simples, nous supposons donc qu'elles feront l'objet d'une classification et seront intégrées dans le dictionnaire.

Le taux de segmentation est évalué à 93 % si l'on considère que les caractères ayant perdus une portion comme des caractères mal segmentés. Nous citons à titre d'exemple le caractère ڤ qui a été coupé en ڤ et ـ dans la figure (**46**). Mais étant donnée que la majorité des caractéristiques de la lettre "fa" est portée par la portion ڤ (la partie ـ fera objet d'un rejet lors de la reconnaissance), nous avons réévalué le taux de segmentation qui a été estimé à 98 %. La lettre "fa" a été citée comme exemple, mais la remarque est valable pour tous les caractères qui pouvaient être interprétés comme mal segmentés.

# CONCLUSION

# GENERALE

# CONCLUSION GENERALE

L'objectif de notre travail est l'étude du module de segmentation d'un texte arabe imprimé en caractères, celui-ci peut être sujet à des perturbations diverses, ce qui nous a amené à compléter notre analyse par l'étude du module de prétraitement.

La première opération réalisée, est l'acquisition d'un texte arabe imprimé en noir et blanc; on distingue deux niveaux d'acquisition : la première dite physique, consiste à faire passer le texte par un scanner et récupérer un fichier graphique; pour notre cas, nous avons choisi le format tiff pour sa flexibilité. L'acquisition logique quant à elle, permet de transformer un fichier graphique en un fichier binaire, c'est à dire contenant la matrice image.

La matrice image ainsi obtenue est un ensemble de pixels adjacents. La valeur de chaque pixel est liée à celle de ces voisins. Une brusque variation d'un pixel par rapport à son voisinage implique la présence d'un bruit que nous avons essayé d'éliminer ou du moins réduire. Nous avons utilisé plusieurs types de filtres : linéaire, non linéaire et morphologique, auxquels nous avons soumis des textes bruités. Notre choix s'est porté sur le filtre non linéaire médian qui donne les résultats les plus satisfaisants.

Par la suite, nous avons abordé la segmentation d'un texte imprimé arabe en trois étapes. La première vise à localiser les lignes de textes et les séparer les une des autres, la seconde consiste à séparer chaque ligne de textes en parties connexes. Ces deux opérations ont été réalisées par construction des histogrammes de lignes et de colonnes. Aucun problème pour la détection de lignes et de parties connexes n'a été noté. La troisième étape de la segmentation, la plus difficile et la plus délicate a pour but de séparer la partie connexe en caractères isolés

La difficulté de cette phase est due à la cursivité de l'écriture arabe. Nous avons adopté deux méthodes pour réaliser cette opération. La méthode de A. Amin étudiée, a montré plusieurs insuffisances que nous avons essayé de combler en adoptant une méthode basée sur le même principe, mais complétée par des critères supplémentaires de détection de début et fin de caractères.

Evidemment, quelques caractères demeurent encore mal segmentés, mais la plupart le sont correctement.

Le travail que nous avons réalisé peut être complété par un module de reconnaissance, qui tient compte des problèmes rencontrés au cours de la segmentation dans le but de les résoudre.

# BIBLIOGRAPHIE

[15] **P.Chauvet,** "Système d'analyse, reconnaissance et description des documents complexes", *Thèse doctorat d'état,* Paris 1992

[16] **L.Boukined, B.Taconet, A.Zahour, A.Faure**, "Recherche de la structure physique d'un document imprimé par rectangularisation", *Afcet, 8ème congrès Reconnaissance des formes et intelligence artificielle, Lyon-Villeurbanne,*Novembre 1991.

[17] **K.Roméo-Pakker, A.Ameur**, "Une méthode rapide de segmentation et de reconnaissance de caractères manuscrits arabes", *14 ème colloque sur l'ecrit GRESTI-JUAN-LES-PINS,* Rouen Septembre 1993.

[18] **A.Amin**, "Un système pour la reconnaissance et la compréhension de l'arabe écrit et imprimé", *Thèse Doctorat d'état,* Nancy Décembre 1985.

[19] **S.Al-Emami et M.Usher**, "On-Line Recognition of Handwritten Arabic Characters", *IEEE Transactions on pattern analysis and machine intelligence, Vol 12 NO 7,* July 1990.

[20] **Amar Benhouhou,** "Contribution à la conception et la réalisation d'un système de reconnaissance de caractères arabesimprimés multipolices", *Thèse de Magister,*INI Alger 1994.

## *BIBLIOGRAPHIE:*


**[1] E.Von Aschberg**, "La reconnaissance", *Revue Langage et système Info PC N°81*, Paris, 1990

**[2] A. Belaïd ,**"Traitement de l'écriture et de documents" *Colloque National sur l'Ecrit et leDocument*, Nancy, Juillet 1992

**[3] J.J. Toumazet**, "Traitement de l'image sur micro-ordinateur", Edition Sybex, France 1987.

**[4] J.F Duvier**, "Structure des fichiers BMP, décoder les fichiers BITMAP", Revue *Langage et système Info PC N°81,Paris, 1990*

**[5] A.Poor**, "Looking at the tiff specification from the inside", *PC Magazine*, Décembre 1991.

**[6] A.Belaid, K.Tombre**, "Analyse de documents: de l'image à la sémantique", *Colloque national sur l'écrit et le document*, Nancy, Juillet 1992.

**[7] T.L.Abegnoli**, "Quatre phases de reconnaissance de caractères", Revue *Electronique* N°29, France, Juin 1993.

**[8] J.D.Becker**, "Arabic word processing", *Communications of the ACM Vol 30 No 7*, Juillet 1987.

**[9] R.Gonzales, P.Wintz,** "Digital image processing", *Edition Addison-Wesley Publishing Company inc*, Université de Tennessee, 1987

**[10] R.Kasturi** "System for recognition and description of graphics", *Computer Engineering Technical Report TR 88-041, Université de Pensylvanie USA, Mars* 1988

**[11] S.Bercu, B.Deylor,** "Segmentation et reconnaissance en ligne de mots manuscrits", *Publication interne N° 700 IRISA/INRIA* Rennes France, Fevrier 1993.

**[12] K.Bouhlila, M.K.Hamrouni, N.Ellouze**, "Method of segmentation of arabic text image into characters",*Communication* , Tunisie, Mars 1989.

**[13] K.M.Bassam et A.Joukhadar**, "Multifont recognition system for arabic Characters", *Proceeding of the third international conference and exhibition on multilingual computing*, Durham U.K, Dec 1992.

**[14] H Al-Youcefi and SS Udpa** " Recognition of arabic characters",*'IEEE Transactions on pattern analysis and machine intelligence, Vol 14 N° 8 August 1992.*