

N° d'ordre :
N° de série :

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



UNIVERSITE ECHAHID HAMMA LAKHDAR - EL OUED
FACULTÉ DES SCIENCES EXACTES
Département D'Informatique



Mémoire de Fin D'étude
Présenté pour l'obtention du Diplôme de

MASTER ACADEMIQUE

Domaine : **Mathématique et Informatique**
Filière : **Informatique**
Spécialité : **Systemes Distribués et Intelligence Artificielle**

Présenté par :

▪ **GHETTAS Ouassila**

Thème

**Accélération d'extraction des lignes
du texte à base des modèles
de cross résolution**

Soutenu le 29-05-2017 Devant le jury:

M.	BELILA Khaoula	MCA	Président
M.	HAMOUD Meriem	MAA	Rapporteur
M.	ZAIZ Faouzi	MAA	Encadreur

Année Universitaire: 2016-2017

Remerciements

Nous remercions Allah le tout puissant, qui nous a donné la force et la patience pour l'accomplissement de ce travail.

Nous remercions notre encadrant :

Mr. (ZAIZ Faouzi)

son précieux conseil et son aide durant toute la période du travail.

Nos vifs remerciements vont également aux membres du jury pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail Et de l'enrichir par leurs propositions.

Nos remerciements s'étendent également à tous nos enseignants durant les années des études.

Enfin, nous tenons également à remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail.

Merci

Résumé

Plusieurs recherches ont été faites durant la dernière décennie dans le but d'améliorer les techniques de segmentation de texte en lignes. Souvent ces techniques concentrent sur l'amélioration de la qualité et la précision d'extraction des lignes. D'une part, ces méthodes utilisent souvent des scans afin de réaliser les traitements. D'autre part, elles utilisent des résolutions de 150 ou 300 dpi.

Dans ce travail nous proposons une méthode qui permet d'accélérer l'opération d'extraction des lignes en utilisant des modèles de cross résolutions ou des résolutions basses afin de gagner du temps de traitement.

La méthode proposée est tester sur trois méthodes de segmentation descendantes : projection horizontale, projection partielle et méthode hybride, et dans les deux cas, avec résolution normale et avec résolution basse. Elle permet d'avoir un gain de temps considérable de l'ordre de 3 à 4 fois.

Mots clés : OCR, Segmentation de texte en lignes descendante, résolution.

ملخص

خلال العشر السنوات الأخيرة أجريت عدة بحوث بهدف تحسين و تطوير تقنيات تقسيم النصوص إلى أسطر و هذه التحسينات تمس نوعية و دقة استخراج هذه الأسطر. في الغالب هذه الطرق تستخدم صور بدقة مسح تتراوح بين 150 و 300 نقطة في البوصة.

في هذا العمل اقترحنا تقنية تسمح بتسريع عملية استخراج الأسطر نستخدم فيها نموذج جديد يعتمد على أقل دقة للصور و هذا بهدف ربح الوقت.

قمنا بتجربة التقنية المقترحة على ثلاثة تقنيات و هي: الإسقاط الأفقي الإسقاط الجزئي، الإسقاط المتعدد أولاً بدون استخدام النموذج و ثانياً باستخدامه لكل تقنيه و لاحظنا ربحا للوقت من ثلاثة إلى أربع مرات .

الكلمات المفتاحية : التعرف على الكتابة، تجزئة النصوص إلى سطور، دقة الصور.

Abstract

Several researches have been done through the last years in the aim of enhancing text lines extraction techniques. Usually, those techniques concentrate on the quality and the precision of text lines extraction. On the one hand, these methods use scans to achieve their processing steps. On the other hand, they use resolutions of 150 or 300 dpi.

In this work we propose a new method that allows accelerating the operation of text lines extraction using a set of cross resolution models (XRM) or low resolution to gain processing time.

The proposed method is tested on three descendant segmentation methods: horizontal projection, partial projection method and a hybrid method, and in the cases, with normal resolution and low resolution. It allows achieving a considerable gain of the order of 3 to 4 times

Key words: OCR, descendant text lines Segmentation, image resolution.

Liste des figures

FIGURE 1-1 MODE D'ACQUISITION	4
FIGURE 1-2 STYLE D'ÉCRITURE	6
FIGURE 1-3 DIFFÉRENTS SYSTÈMES, REPRÉSENTAT ET APPROCHES DE RECONNA	7
FIGURE 1-4 PROCESSUS GÉNÉRALE D'UN SYSTÈME OCR	8
FIGURE 1-5 OPÉRATION DE PRÉTRAITEMENT D'IMAGE	9
FIGURE 1-6 EXEMPLE DE LISSAGE	10
FIGURE 1-7 CORRECTION DE L'INCLINAISON ON DES LIGNES	11
FIGURE 2-1 SEGMENTATION DE LA PAGE	15
FIGURE 2-2 SEGMENTATION DE TEXTE EN LIGNES	16
FIGURE 2-3 SEGMENTATION DE LIGNE EN MOTS	16
FIGURE 2-4 APPROCHE DE SEGMENTATION DES MOTS	17
FIGURE 2-5 SEGMENTATION PAR CONTOUR D'UN MOT	17
FIGURE 2-6 SEGMENTATION PAR HISTOGRAMME D'UN MOT	17
FIGURE 2-7 SEGMENTATION PAR FENÊTRE GLISSANTE	18
FIGURE 2-8 TYPES DES LIGNES	19
FIGURE 2-9 PROJECTION HORIZONTALE D'UN LETTRE	20
FIGURE 2-10 LE PROJECTION D'UN COLONNE	20
FIGURE 2-11 RÉSULTAT D'EXTRACTION DES LIGNES BASÉ SUR PROJECTION PARTIEL.....	21
FIGURE 2-12 RÉSULTAT D'EXTRACTION DES LIGNES BASÉ SUR COMPOSANTS CONNEXES 1	22
FIGURE 2-13 RÉSULTAT DE L'EXTRACTION DES LIGNES BASÉ SUR COMPOSANT CONNEXE 2	22
FIGURE 2-14 RÉSULTAT D'EXTRACTION DES LIGNES BASÉ SUR LE TRANSFORMÉE DE HAUGH ..	23
FIGURE 3-1 ARCHITECTURE DE MODÈLE PROPOSÉ (CROSS RÉOLUTION FAIBLE).....	25
FIGURE 3-2 CARACTÉRISTIQUE D'IMAGE.....	26
FIGURE 3-3 TRANSFORMATION D'ÉCHELLE D'IMAGE	29
FIGURE 3-4 FONCTION DE MAPPING	35
FIGURE 4-1 INTERFACE D'APPLICATION.....	38
FIGURE 4-2 INFORMATION SUR L'IMAGE ET LES BUTTONS	38
FIGURE 4-3 EXTRACTION DES LIGNES PAR PROJECTION HORIZONTALE	39
FIGURE 4-4 EXTRACTION DES LIGNES PAR PROJECTION PAR FENÊTRE	39
FIGURE 4-5 EXTRACTION DES LIGNES PAR MÉTHODE HYBRIDE.....	40
FIGURE 4-6 ILLUSTRATION DE TEMPS D'EXECUTION SANS RESOLUTION	42
FIGURE 4-7 ILLUSTRATION DE TEMPS D'EXÉCUTION DE MÉTHODE PAR FENÊTRE.....	42
FIGURE 4-8 ILLUSTRATION DE TEMPS D'EXÉCUTION DE MÉTHODE HYBRIDE.....	43

Liste des tableaux

TABLEAU 3-1 TABLEAUX DE CHOIX DE RÉOLUTION PAR TAILLE DE STYLO	28
TABLEAU 4-1 RÉSULTATS DE TEST	41
TABLEAU 4-2 LES MOYENNE DE RÉSULTATS SANS RÉOLUTION	43
TABLEAU 4-3 LES MOYENNE DES RÉSULTATS AVEC RÉOLUTION BASSE	44

Liste des algorithmes

ALGORITHME 3-1 TAILLE DE STYLO.....	26
ALGORITHME 3-2 TRANSFORMATION D'ÉCHELLE D'IMAGE	28
ALGORITHME 3-3 PROJECTION HORIZONTALE	30
ALGORITHME 3-4 PROJECTION PAR FENÊTRE	31
ALGORITHME 3-5 PROJECTION HYBRIDE	32
ALGORITHME 3-6 LOCALISATION DE LIGNE DE BASE	33
ALGORITHME 3-7 LOCALISATION DE LIGNE HAUTE.....	33
ALGORITHME 3-8 LOCALISATION DE LIGNE INFÉRIEURE.....	34

SOMMAIRE

LISTE DES FIGURES	I
LISTE DES TABLEAUX	II
LISTE DES ALGORITHMES	III
SOMMAIRE	IV
INTRODUCTION GENERALE	1
1 RECONNAISSANCE OPTIQUE DE CARACTÈRE	3
INTRODUCTION.....	4
1.1 DIFFERENTS ASPECTS DE L'OCR	4
1.1.1. Mode d'acquisition En lignes ou Hors ligne.....	4
1.1.2. Approches de reconnaissance.....	5
1.1.2.1. Approche globale:	5
1.1.2.2. Approche analytique:	5
1.1.3. Reconnaissance de l'imprimé ou du manuscrit.....	5
1.1.4. Applications de la Reconnaissance.....	8
1.2 PROCESSUS GENERALE D'UN SYSTEME DE RECONNAISSANCE	8
1.2.1. Acquisition	9
1.2.2. Prétraitements	9
1.2.2.1. Binarisation (seuillage):	10
1.2.2.2. Lissage:.....	10
1.2.2.3. Normalisation:.....	10
1.2.3. Segmentation	11
1.2.4. Extraction des caractéristiques	12
1.2.5. Classification	12
1.2.6. Post-traitement.....	13
CONCLUSION.....	13
2 SEGMENTATION ET EXTRACTION DES LIGNES	14
INTRODUCTION.....	15
2.1 SEGMENTATION DE DOCUMENT.....	15
2.1.1. Segmentation de la page	15
2.1.2. Segmentation de texte en lignes	15
2.1.3. Segmentation de lignes en mots	16
2.1.4. Segmentation de mot en caractères	16
2.1.4.1. Segmentation explicite :	16
2.1.4.2. Segmentation implicite :.....	17
2.2 TECHNIQUES DE SEGMENTATION D'ECRITURE.....	17
2.2.1. Segmentation par contour.....	17
2.2.2. Segmentation par histogramme	17
2.2.3. Segmentation par fenêtre	18
2.3 EXTRACTION DES LIGNES	18
2.3.1. Caractéristiques des lignes de texte.....	18

2.3.2. Méthodes d'extraction des lignes	19
2.3.2.1. Méthodes descendantes	19
2.3.2.2. Méthodes ascendantes :	21
CONCLUSION	23
3 CONCEPTION ET ARCHITECTEUR	24
INTRODUCTION	25
3.1 DESCRIPTION GENERALE DE MODELE PROPOSE	25
3.1.1. Taille de stylo	26
3.1.2. Choix de résolution cible	27
3.1.3. Transformation d'échelle	28
3.1.4. Extraction des lignes	29
3.1.4.1. Projection horizontale	29
3.1.4.2. Projection par fenêtre	31
3.1.4.3. Méthode hybride (horizontale et par fenêtre) [22]	31
3.1.4.4. Extraction des caractéristique de ligne	32
3.1.5. Fonction de mapping	35
CONCLUSION	35
4 TEST ET RÉSULTATS	36
INTRODUCTION	37
4.1 LANGAGE UTILISE	37
4.2 INTERFACES DE L'APPLICATION	37
4.3 TEST	40
4.3.1. Comparaison entre les résultats :	42
4.3.2. Comparaison entre les méthodes :	43
CONCLUSION	44
CONCLUSION GENERALE	45
BIBLIOGRAPHIE	46

Introduction générale

Jusqu'aujourd'hui, la réalisation des systèmes de reconnaissance de l'écriture, que ce soit manuscrite ou imprimée, présente une tâche très importante vue les grandes bases de documents existantes, et l'importance de tels systèmes comme une interface homme machine. Deux classes de systèmes de reconnaissance existent dans ce domaine : systèmes hors-lignes et systèmes en-lignes. Dans le premier cas, l'image du texte est d'abord numérisée à l'aide d'un scanner ou caméra, ensuite, donnée à un système OCR (Optic Character Recognition) afin de transformer le contenu de l'image en texte dans un fichier numérique. Par contre, dans le second cas, la reconnaissance est réalisée au fur et au mesure que le texte est écrit à l'aide d'un stylo électronique, ce qui permet d'avoir des informations supplémentaire par rapport au premier cas. Le processus de reconnaissance passe par les phases suivantes : acquisition, prétraitement, segmentation, post-traitement.

La segmentation est une phase très importante dans le processus de reconnaissance. Elle consiste à extraire les différents segments nécessaires aux phases suivantes. Plusieurs techniques de segmentation commencent par l'extraction des lignes du texte, ensuite, chaque ligne est segmentée en un ensemble de mots et caractères. Les stratégies d'extractions des lignes se divisent en deux approches : descendants et ascendants. Les méthodes descendantes sont basées sur la projection et les méthodes ascendantes sont basées sur des traitements au niveau de pixels ou des composantes connexes. La déférence entre ces méthodes réside dans le nombre des lignes extraites.

Dans cette étude nous proposons une contribution qui se présente par une méthode à base de modèles à résolution faible (XRM : Cross Resolution Models) permettant d'accélérer la tâche d'extraction des lignes de texte d'une image de texte dans le cas des systèmes hors-lignes.

Le reste du travail est organisé comme suit:

- Le premier chapitre présente un état de l'art sur les systèmes de reconnaissance de l'écriture manuscrite. D'abord, nous allons présenter la situation et les problèmes liées à ces systèmes. Ensuite, nous allons donner un aperçu sur l'architecture d'un OCR.
- Le deuxième chapitre présente un panorama sur la segmentation de l'écriture manuscrite. Dans ce chapitre, nous allons voir les différentes approches et stratégies de segmentation. Puis, nous décrivons les différentes méthodes d'extractions des lignes proposées.

- Le troisième chapitre est consacré à la conception et la mise en œuvre du système qui décrit en détail la méthode d'accélération proposée.
- Le chapitre quatrième présente les résultats obtenus par notre méthode ainsi qu'une comparaison, discussion et évaluation de la méthode.

Enfin, nous terminons le travail par une conclusion sur les résultats obtenus par la méthode proposée, et des perspectives de ce travail.

Chapitre I

Reconnaissance Optique de caractère

Introduction

La reconnaissance optique de caractères est le processus automatisé de transformer une image de document donné en un fichier de texte symbolique. Le format de l'image du document peut être numériquement Créées, télécopiées, numérisé, imprimées à la machine ou manuscrits, etc.

Dans ce chapitre, nous présentons une description générale des différents éléments et aspects concernant le processus de reconnaissance de l'écriture.

1.1 Différents aspects de l'OCR

Les systèmes de reconnaissance optique de caractères sont classifiés selon différentes catégories, on peut citer:

- Selon le mode d'acquisition : en ligne ou hors ligne.
- Selon le mode de traitement : globale que le traitement se fait sur la totalité de mot ou analytique que le traitement se fait sur le caractère après segmentation .

1.1.1. Mode d'acquisition En lignes ou Hors ligne

Dans le cas des systèmes **en ligne** se fait en temps réel (pendant l'écriture), ces systèmes peuvent être utilisés seulement pour la reconnaissance d'écriture manuscrite [1]. Le texte est saisi par un stylo sur une surface ou un écran digitale, la reconnaissance est réalisée pendant le processus de saisie [2].

Par contre, dans le cas des systèmes **hors-lignes** l'image du texte écrit est numérisée par un scanner ou un caméra, ce mode peut être considéré comme le plus général de la reconnaissance d'écriture [3], il consiste à reconnaître un texte préalablement écrit ou imprimé sur une image de texte acquise par un scanner ou un caméra, ou des images créées à travers une écriture en ligne.

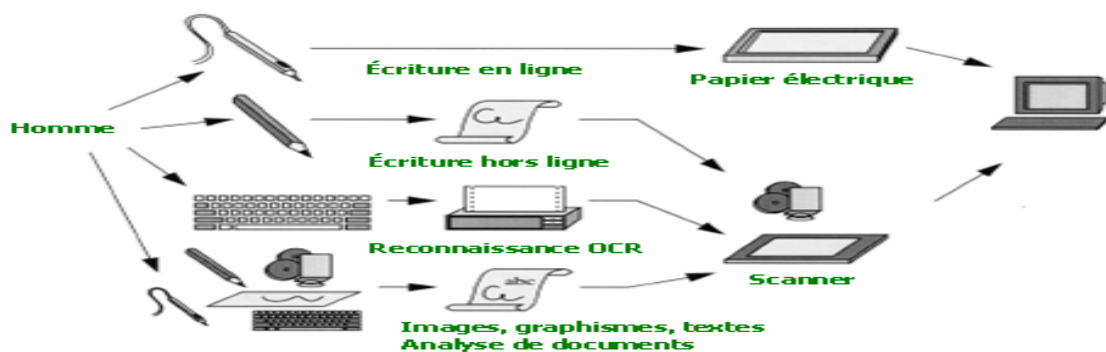


Figure 1-1 Mode d'acquisition [4]

1.1.2. Approches de reconnaissance

Les systèmes de reconnaissances de l'écriture est basé sur le mode d'application et le segmentation d'image ou de mot, si le but de reconnait le mot comme une seule entité ou reconnait le mot par les caractères composé.

Il y'à deux approches pour le reconnaissance des mots :

- Approche globale ;
- Approche analytique.

1.1.2.1. Approche globale:

Dans l'approche **globale** le mot dans un document est comme un seul entité, cette approche est efficace et simple car elle ne nécessite pas de reconnaissance en niveau de caractère ou de graphèmes qui composant le mot [1].

1.1.2.2. Approche analytique:

Contrairement de l'approche globale l'approche **analytique** décompose le mot en une séquence de caractères intermédiaires..Le mot est segmenté en caractère ou en fragment morphologiques significatifs inférieurs au caractère appelés graphèmes [5].

1.1.3. Reconnaissance de l'imprimé ou du manuscrit

L'OCR peut être subdivisé en reconnaissance de l'écriture manuscrite et de reconnaissance de l'écriture imprimée. La reconnaissance de caractères manuscrits est plus difficile à appliquer que la reconnaissance de caractères imprimés.

- Dans le cas de l'imprimé, les images des caractères à traiter sont mise en forme par des polices standard qui sont bien alignés et souvent bien séparés verticalement, ce qui simplifie la phase de lecture [6]. De plus, le graphisme des caractères est conforme à un style de fonte qui constitue un modèle pour l'identification, la reconnaissance peut être mono-fonte, multicolore ou omnicolore .

- Dans le cas du manuscrit, il existe divers styles d'écriture humaine dont les caractères sont souvent ligaturés et leur graphisme est inégalement proportionné. Cela nécessite l'emploi de techniques de délimitation très spécifiques et souvent des connaissances contextuelles pour guider la lecture.

Il existe plusieurs facteurs qui pilotant un système de reconnaissance d'écriture:

1. Types et styles d'écriture [7]:

On distingue principalement trois types d'écritures le script qui est un mot séparées, et le cursif correspond à un mot ou toutes les lettres sont attachées, et l'écriture mixte.

- écriture bâton (en majuscules bien séparées) ;
- écriture scripte en lettres isolées ;
- écriture scripte jointive avec des lettres mal segmentées ;
- écriture cursive en lettres liées ;
- écriture mixte cursive et scripte.

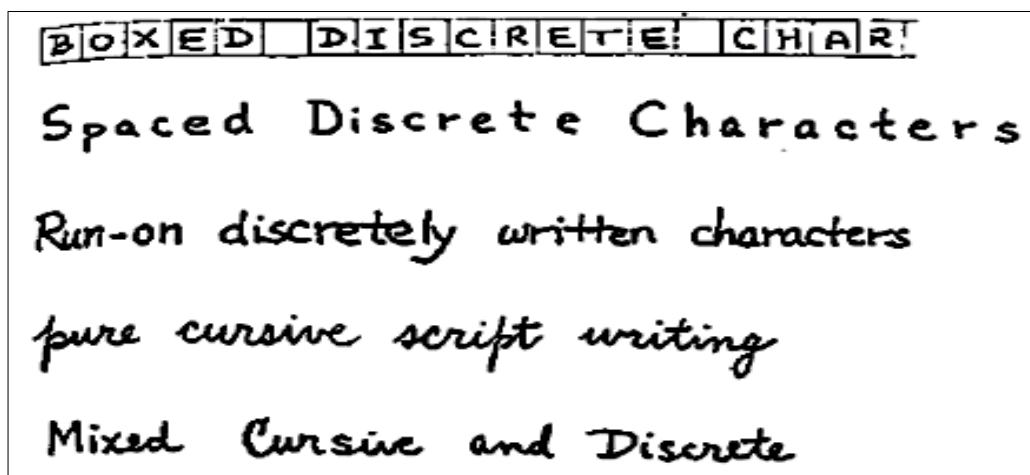


Figure 1-2 Style d'écriture

2. Disposition spatiale du texte:

L'écriture **contrainte** correspondant à une écriture guidée par des cadres (les formulaires par exemple), et l'écriture **non-contrainte** correspondant à une écriture guidée exclusivement par le scripteur donc extrêmement variable.

3. Nombre de scripteurs:

La difficulté de reconnaissance croit avec ce nombre, divisant l'échelle en trois ; mono-scripteurs, multi- scripteurs et omni-scripteurs.

4. Taille de vocabulaire:

Lorsque la taille du vocabulaire augmente, la tâche de reconnaissance devient de plus en plus complexe, car des mots ressemblants ont plus de chances d'être présents dans le vocabulaire. La complexité calculatoire est également liée à la taille du vocabulaire.

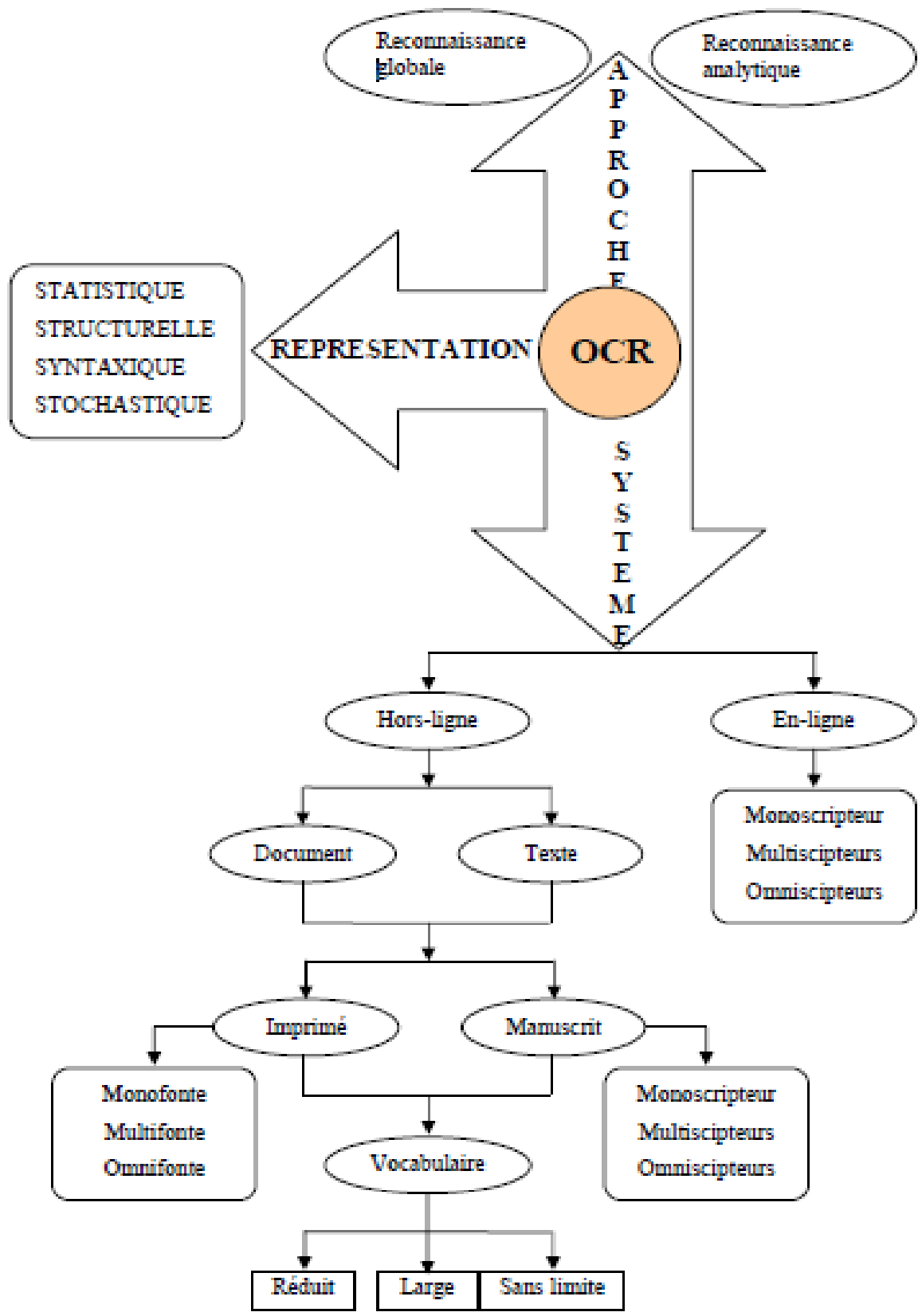


Figure 1-3 Différents systèmes, représentations et approches de reconnaissance [8]

1.1.4. Applications de la Reconnaissance

Les applications qui plus visent actuellement :

- Dans la reconnaissance **en ligne** telle que :

L'Ordinateurs sans clavier pour ambition de remplacer le clavier et la souris de l'ordinateur par un stylo. Cette modification a pour but de rendre les ordinateurs plus conviviaux en permettant de les utiliser dans des situations très diversifiées (prises de notes, rédaction de commandes, de constats d'accidents, enseignement de l'écriture, etc.) [4].

- Dans la reconnaissance **hors ligne** telle que [9] :

La lecture automatique de documents manuscrits: lecture d'adresse postale manuscrite et tri du courrier, lecture des montants de chèques, la numérisation des collections patrimoniales, des archives, documents anciens.

1.2 Processus générale d'un système de reconnaissance

Le processus général d'un système de reconnaissance est caractérisé par ces étapes suivantes : acquisition, prétraitement, classification/reconnaissance, post-traitement.

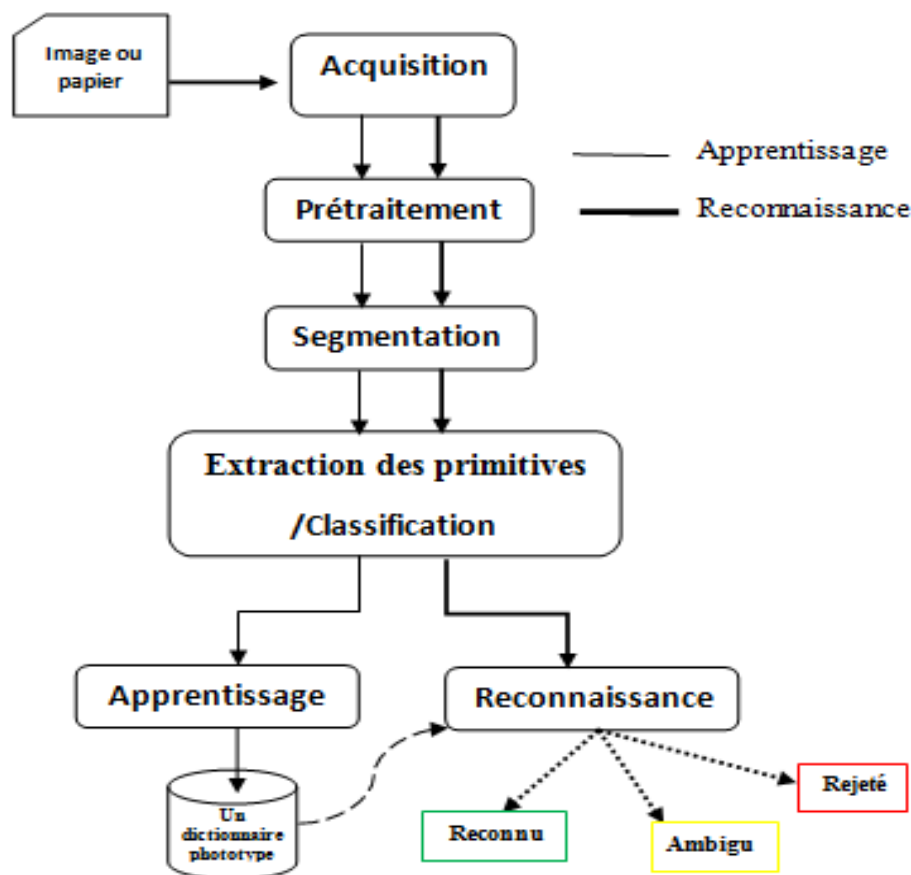


Figure 1-4 Processus générale d'un système OCR [10]

1.2.1. Acquisition

La première opération consiste à acquérir l'image numérique de la page avec un scanner ou caméra. Cependant, les documents à traiter sont de contenu et de qualité divers ; il est donc quelquefois nécessaire d'acquérir les images en niveaux de gris. Pour éviter l'encombrement de la mémoire, des images noir et blanc sont préférables. Cette phase est caractérisée par deux paramètres important :

✓ **Résolution :**

La résolution optimale d'une image dépend de l'épaisseur du trait d'écriture. Ainsi, pour que les traitements ultérieurs puissent s'appliquer correctement, il faut que le trait d'écriture ait une épaisseur minimale de 3 pixels. La résolution souvent utilisée est de 300dpi.

✓ **Niveau d'éclairage:**

Un éclairage élevé (du numériseur) réduit le bruit, mais fait disparaître les traits minces. Donc il faut choisir un niveau d'éclairage optimal selon la qualité du document physique.

1.2.2. Prétraitements

Le but de prétraitement est de réduire le bruit superposé aux documents, le bruit peut être du aux conditions d'acquisition (éclairage, mise incorrect du document, ... etc.) ou encore à la qualité du document l'origine [11] [12].

Plusieurs opérations de prétraitement sont utiliser on peut citer : binarisation, normalisation, redressement, squelettisation, lissage. mais pas nécessairement appliquer toutes ces opérations.

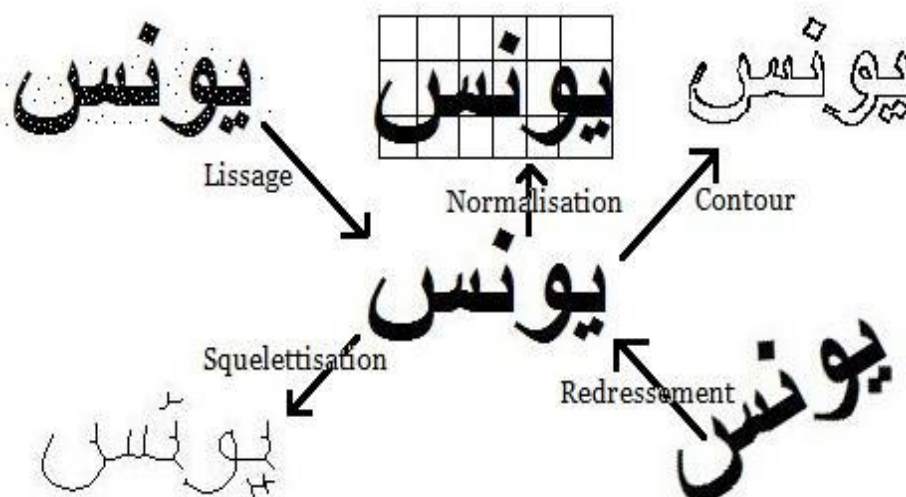


Figure 1-5 Opération de prétraitement d'image [1]

1.2.2.1. Binarisation (seuillage):

Transformer une image(entré) à niveaux de gris en une image binaire (noir et blanc) composé des valeurs 0 et 1.

Pour effectuer la transformation nous utilisons un seuil, tous les pixels dont les valeurs au-dessus de ce seuil sont considérés comme étant des pixels d'arrière plan, et tous les autres pixels comme étant des pixels de texte.

1.2.2.2. Lissage:

L'image de caractères peut être entachée de bruit dû aux artefacts de l'acquisition et souvent à la qualité du document, conduisant soit à des absences de points (trous) soit à des empâtements ou des excroissances 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 utilisons les notions suivantes pour la description des masques de transformation de voisinage : « 1 » pour un point de la forme et « 0 » pour un point du fond et « X » pour un point quelconque [4].

L'opération de nettoyage conduit à supprimer les petites tâches et les excroissances de la forme, elle est réalisée de différentes manières.



Figure 1-6 Exemple de lissage [13]

1.2.2.3. Normalisation:

L'objectif de cette technique est de rendre l'écriture la plus indépendante possible du scripteur, il y'a trois techniques de normalisation [5] :

- ✓ La correction de l'inclinaison des lignes:

La correction de l'inclinaison des lignes de texte (également appelée correction de "skew"), consiste à redresser horizontalement les lignes d'écriture obliques.

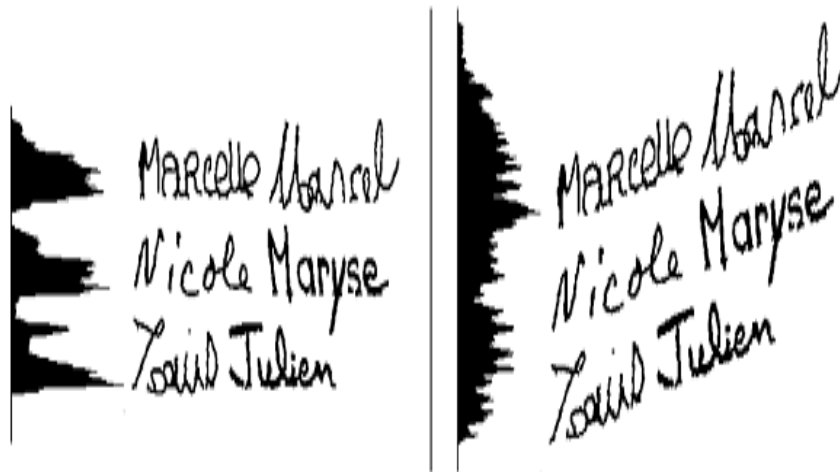


Figure 1-7 Correction de l'inclinaison des lignes [5]

- ✓ La correction de l'inclinaison des lettres:

Certains scripteurs écrivent leurs lettres de façon inclinée par rapport à l'axe vertical. Cette inclinaison de l'écriture est également appelée « slant ». Les lettres peuvent être inclinées vers la droite ou vers la gauche. Pour la même raison que dans le paragraphe précédent, il convient de corriger cette inclinaison de l'écriture pour la rendre la plus indépendante possible des spécificités d'écriture du scripteur [14].

- ✓ La normalisation des caractères.

Une autre manière de diminuer la variabilité est de normaliser la taille des caractères, en les ramenant tous à une même taille standard prédéfinie.

1.2.3. Segmentation

C'est une phase essentielle dans les systèmes de reconnaissance, l'objectif de cette phase est d'isoler les différentes composantes d'un bloc de texte en lignes, en mots, et en caractères, avant la phase de reconnaissance de caractères [1]. Le résultat de cette opération est une forme isolée à partir d'une image et qui pourrait être un caractère ou non.

1.2.4. Extraction des caractéristiques

L'extraction de caractéristiques ou primitives, est une étapes essentielle, les images dans cette phase transformer à un ensemble des caractéristiques dans un vecteur de taille n , qui représente l'image dans la phase de classification, qui peuvent être extraites par plusieurs manière directement par l'image complète, ou par segmentation explicite, ou par fenêtre glissante.

Les types des caractéristiques peuvent être aux trois niveaux, Niveaux bas tel que les petites lignes, les courbes, les traits, les barres,..., et niveau moyenne basés sur la segmentation explicite, et niveau haut on peut citer : les boucles, les ascendants, les descendants, en plus des jonctions, les points finaux et traits et les points diacritiques. en peuvent être classé en cinq groupes [13] :

- **Caractéristiques topologiques ou métriques:** Ce type de primitives est basé sur des densités de pixels, Le terme métrique désigne la mesure d'une distance.
- **Caractéristiques locales ou structurelle:** ce type extraites non pas de l'image brute mais à partir du squelette ou du contour de la forme en donnant ses propriétés globales et locales.
- **Caractéristiques statistiques:** Elles sont extraites en considérant la distribution des pixels noirs de l'objet (caractère, mot, chiffres).
- **Transformations globales:** La transformation consiste à convertir la représentation en pixels en une représentation plus abstraite pour réduire la dimension des caractères, tout en conservant le maximum d'informations sur la forme à reconnaître.
- **Superposition des modèles (template matching) et corrélation:** La méthode de "template matching" appliquée à une image binaire (en niveaux de gris ou squelettes), consiste à utiliser l'image de la forme comme vecteur de caractéristiques pour être comparé à un modèle (template) pixel par pixel dans la phase de reconnaissance, et une mesure de similarité est calculée [10].

1.2.5. Classification

Cette phase regroupe deux taches : l'apprentissage qui consiste à apprendre au système les propriétés pertinentes du vocabulaires utilisé et des modèles de références [3], et le deuxième qui est la reconnaissance pour identifier (reconnaître) les nouvelles exemples.

1) Apprentissage :

Le mécanisme d'apprentissage a pour objectif d'extraire d'une base d'apprentissage l'ensemble des connaissances nécessaires à la modélisation du problème et à les structurer si besoin.

2) Reconnaissance:

La reconnaissance peut conduire à un *succès* si la réponse est unique (un seul modèle répond à la description de la forme du caractère). Les étapes peuvent être regroupées en des quatre approches principaux [8] :

- ✓ approche statistique:
 - ✓ approche structurelle;
 - ✓ approche stochastique;
 - ✓ et l'approche hybride.

La différence essentielle entre ces approches réside dans la représentation de la forme. La reconnaissance se termine par une décision qui peut être [7] :

- **Caractère reconnu** : Si le système arrive à associer un et un seul prototype au caractère à reconnaître, il prend une décision unique.
- **Caractère ambigu** : Si le système associe plusieurs prototypes au caractère à reconnaître, il propose ainsi plusieurs choix avec des confiances de même ordre.
- **Caractère rejeté** : Si le système n'arrive pas à associer aucun prototype au caractère à reconnaître, il ne prend aucune décision de classification.

1.2.6. Post-traitement

Cette phase utilisé pour la vérification et l'amélioration du taux de reconnaissance, [15] par un dictionnaire, un lexique bien développé d'un ensemble de règles orthographiques utilisées pendant ou après la phase de reconnaissance.

Conclusion

Dans ce chapitre, nous avons présenté les concepts de base pour les systèmes de reconnaissance d'écriture et les différentes étapes de celles-ci: acquisition, prétraitement, segmentation, classification, post-traitement.

Dans le deuxième chapitre, nous allons décrire les étapes de segmentation, et les méthodes d'extraction des lignes.

Chapitre II

Segmentation et Extraction des lignes

Introduction

La segmentation est le processus de partitionnement du document en des entités homogènes (lignes, mots ou caractères), elle est très importante puisque le taux de reconnaissance dépend fortement d'elle, et nécessaire dans le développement d'un système de reconnaissance automatique de l'écriture.

Dans ce chapitre, nous présentons les différentes techniques de segmentation et quelques méthodes d'extraction des lignes.

2.1 Segmentation de document

La segmentation d'une page (document) se fait à l'aide de quatre étapes à savoir :

- Segmentation de page en zone ;
- Segmentation en lignes ;
- Segmentation en mot et pseudo-mots ;
- et segmentation en caractères/graphèmes.

2.1.1. Segmentation de la page

Dans cette phase consiste à déterminer les zones d'informations dans chaque page, La segmentation de document permet d'étiqueter la nature du média représenté dans chaque zone (texte, graphique, photographie,...etc.). De ce fait, ces approches permettent la séparation et l'extraction des zones multi structurelles dans les documents.

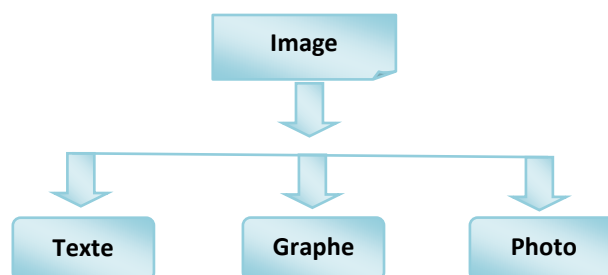


Figure 2-1 Segmentation de la page

2.1.2. Segmentation de texte en lignes

Consiste à séparer le texte en différentes lignes pour en extraire les mots puis les caractères, selon le besoin ou le but de système de reconnaissance.

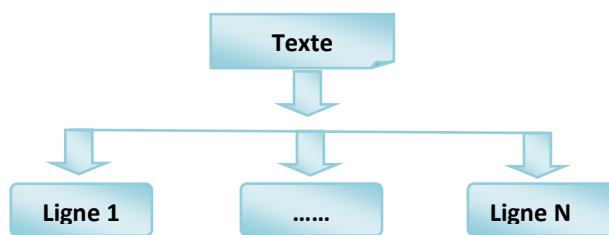


Figure 2-2 Segmentation de texte en lignes

2.1.3. Segmentation de lignes en mots

Il existe plusieurs méthodes pour découper les lignes en groupes des mots, telle que: l'histogramme de projection verticales, le suivi du contour, détermination des squelettes ou la détermination des composantes connexes; Mais dans certain cas la technique est pas efficace (par exemples l'écriture arabe si les mot se chevauchent).

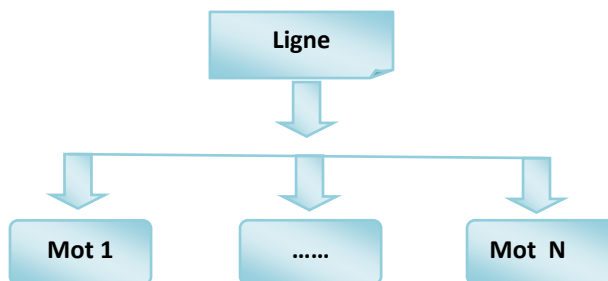


Figure 2-3 Segmentation de ligne en mots

2.1.4. Segmentation de mot en caractères

La segmentation des mots en séquence des caractères est une opération qui permet de découper le mot ou le texte selon les lettres ou les graphèmes qui le composent, mais, elle est très difficile, pour l'écriture arabe soit manuscrits ou imprimé.

Il existe deux approches permettant la mise en œuvre de la segmentation en mot :

1. Segmentation explicite ;
2. Segmentation implicite.

2.1.4.1. Segmentation explicite :

Dans cette approche la segmentation est effectuée par la sélection des points les plus proches (ex: a partir du contour), elle s'appuie sur un découpage à priori de l'image en sous-unités qui peuvent être des lettres ou des graphèmes.

2.1.4.2. Segmentation implicite :

Cette approche consiste à segmenter le mot exactement en caractère en utilisant des propriété général de l'écriture d'un pseudo-mot [16], La segmentation est effectuée pendant la reconnaissance.

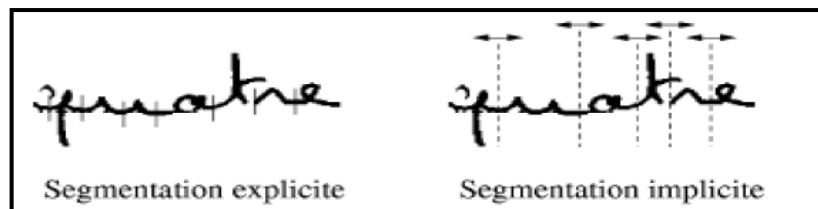


Figure 2-4 Approche de segmentation des mots [13]

2.2 Techniques de segmentation d'écriture

La segmentation des mot en caractère si l'écriture imprimé ne pose pas des problèmes, mais le cas de l'écriture cursive la segmentation est plus complexe, parce que impossible de segmenter directement le mot en des lettres.

Les littérateur ont développés des techniques pour facilité la segmentation. A partir du détection de contour, histogramme, squelette, réservoir et fenêtre glissante, on peut citer quelque exemples des ce techniques:

2.2.1. Segmentation par contour

Utilise des filtres et des techniques de seuillage pour la détection d'objets et de certaines régions homogènes, le point faible de cette technique est le temps d'exécution qui est très grand.

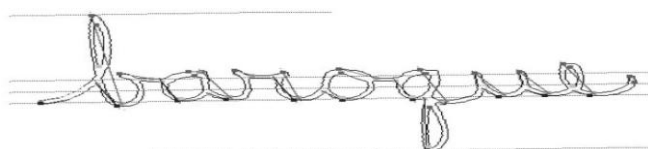


Figure 2-5 Segmentation par contour d'un mot

2.2.2. Segmentation par histogramme

C'est l'approche la plus utilisée dans les systèmes de reconnaissance en raison de sa vitesse élevée de récupération de résultats, mais elle est sensible au problème du chevauchement entre des lignes adjacentes.

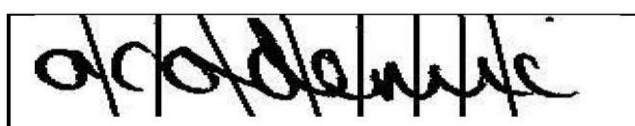


Figure 2-6 Segmentation par histogramme d'un mot [17]

2.2.3. Segmentation par fenêtre

Le principe de cette segmentation; qu'elle est fondée sur le parcourt de l'image par une fenêtre de largeur fixé, et un hauteur égale à l'hauteur d'image, mais existe certain système qui utilisent cette fenêtre avec une taille variable selon la taille de chaque caractère dans l'image.

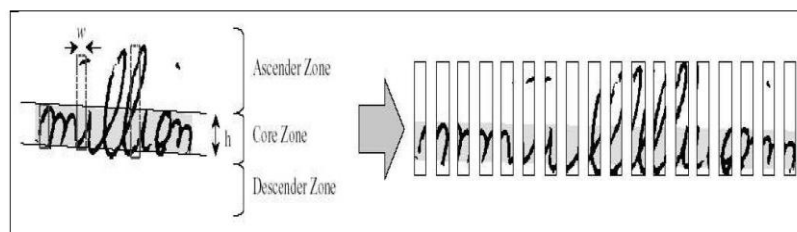


Figure 2-7 Segmentation par fenêtre glissante [17]

2.3 Extraction des lignes

L'extraction des lignes de textes est une étape principale à tous les processus d'analyse et reconnaissance de mots ou des caractères. Elle a comme objectif d'assigner chaque composant du texte à une ligne appropriée ; ce qui permet de préparer les données pour les traitements ultérieurs tel que la segmentation en mots, en caractères et l'extraction des caractéristiques.

Cette opération est relativement facile quand le texte est régulière, non incliné, mono-orienté, et ne comportant pas de chevauchement. Ces conditions sont sans doute réunies pour des textes imprimés mais pas souvent pour des textes manuscrits.

2.3.1. Caractéristiques des lignes de texte

Essentiellement, nous distinguons pour une telle ligne de texte, quatre lignes de références : la ligne haute, la ligne de base supérieure, la ligne de base, la ligne de base inférieure.

Ces lignes définissent les limites de trois zones significatives d'une ligne de texte : la zone ascendante, la zone médiane et la zone descendante.

La ligne de base est définie comme la ligne sur laquelle repose les lettres ne possédant pas de dépassement bas, qui encadre, en général la bande des minuscules, et utilisé cette ligne dans l'étape de prétraitement pour l'extraction et la normalisation des primitives.

Ces lignes sont illustrées par la Figure ci-dessous.

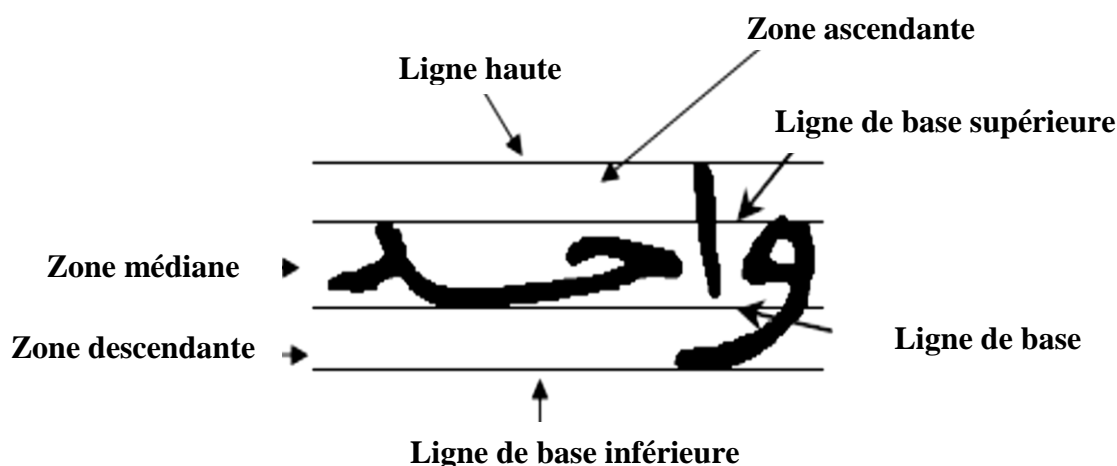


Figure 2-8 Types des lignes

2.3.2. Méthodes d'extraction des lignes

Les techniques d'extraction des lignes de texte se divisent comme toutes les méthodes de segmentation en deux classes : descendante et ascendante, nous définissons les exemples, ces méthodes et quelques notions utilisées pour l'extraction des lignes.

➤ Histogramme:

Représente le nombre de points dont l'intensité a une valeur donnée ; il donne donc, la fréquence d'apparition de chaque niveau de gris.

➤ Composantes connexes:

Une composante connexe est un ensemble de points dans le plan [1]. Elle peut correspondre à un point diacritique, un accent, au corps d'un caractère ou d'une chaîne de caractères....

➤ Histogrammes de projection:

Le principe de cette méthode est de calculer le nombre de Pixels placés en lignes et colonnes en se servant respectivement de l'histogramme de projection vertical et l'histogramme de projection horizontal.

2.3.2.1. Méthodes descendantes

Les méthodes descendantes commencent par le niveau le plus élevé à savoir la page et descendent d'un niveau à un autre jusqu'à arriver au niveau des composantes connexes ou au niveau pixel.

a) Méthodes basées sur la projection:

1. Dans la figure suivant le résultat d'application de projection horizontale sur une lettre [18].

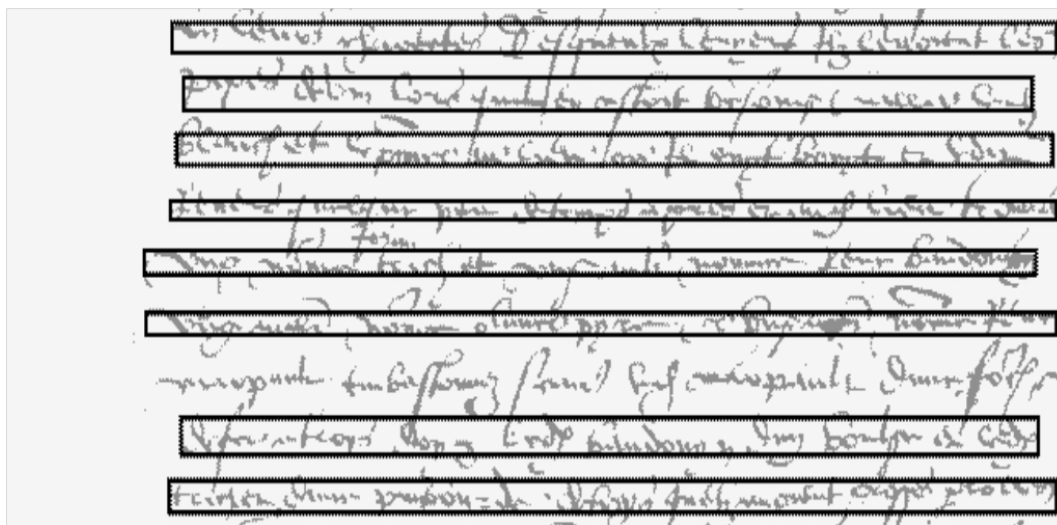


Figure 2-9 Projection horizontale d'un lettre

2. Dans [19] Bennisri et al. ont proposé une méthode d'extraction des lignes d'un texte manuscrit arabe basée sur la projection partielle.

Cette méthode nécessite les étapes suivantes:

- Subdiviser le texte en colonnes, la largeur d'une colonne est celle d'un mot environ, car l'inclinaison d'un mot est moins importante que celle d'une ligne.

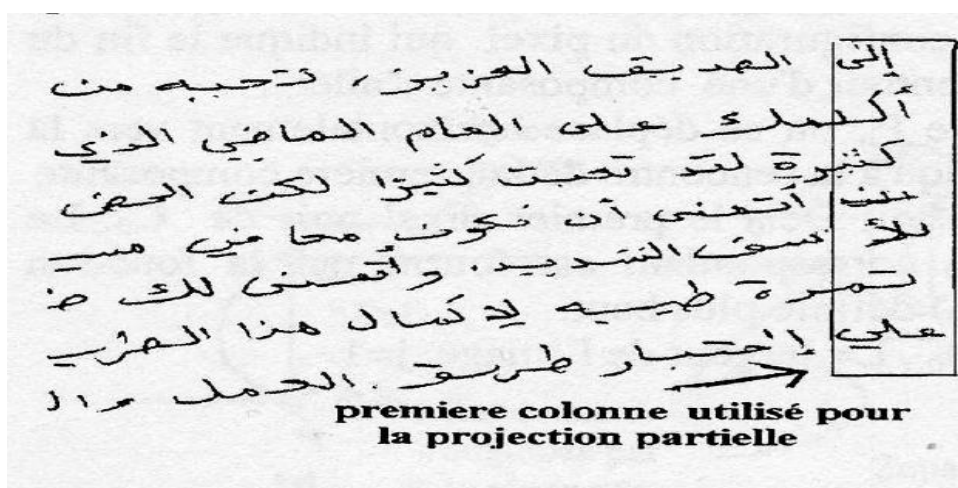


Figure 2-10 Le projection d'un colonne

- Déterminer les minima des histogrammes résultant des projections horizontales pour chaque colonne. Ces minima correspondent à la zone de séparation entre deux lignes adjacentes.

- Représenter chaque minima par un trait horizontal de même longueur que la colonne. La liaison de ces traits entre eux permet d'avoir la séparatrice de deux lignes adjacentes.

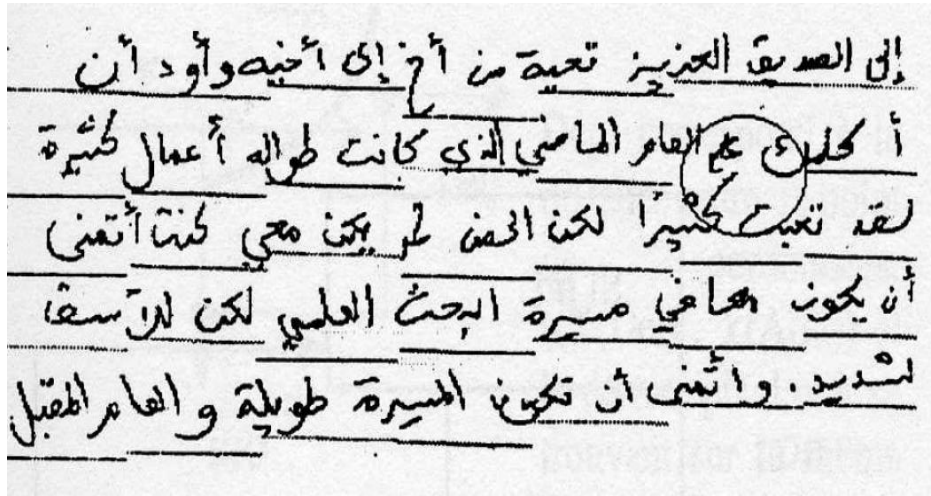


Figure 2-11 Résultat d'extraction des lignes basé sur projection partiel

b) Méthodes utilisant un modèle de document

Parmi les techniques les plus connues qui utilisent des modèles, nous trouvons le découpage en XY.

2.3.2.2. Méthodes ascendantes :

Les méthodes ascendantes commencent par le niveau le plus bas et remontent d'un niveau à un autre jusqu'à compléter la page. En effet, elles se basent sur l'analyse des composantes connexes. Ces dernières sont obtenues en scannant une image pixel par pixel, et en regroupant les pixels en des composants en se basant sur la connexité des pixels qui peut être en 4 voisins ou en 8 voisins.

a) Méthodes basées sur le regroupement de composantes connexes

1. Une approche pour l'extraction des lignes des anciens manuscrits arabes a été proposée par Zahour et al. dans [20].

- 1) Divisé le document en plusieurs colonnes de même taille.
- 2) Puis, chaque colonne du document est segmentée en trois types de blocs de texte : les petits blocs qui représentent généralement les symboles diacritiques, les blocs moyens qui correspondent au corps du texte et les grands blocs qui reflètent le chevauchement entre les mots des lignes voisines.

b) Méthodes basées sur la transformée de Haugh

Ce sont des composantes connexes ayant une direction privilégiée

1. Dans [18], Likforman-Sulem et al. ont proposé une autre méthode basée sur la transformée de Haugh pour l'extraction des lignes des documents manuscrits.

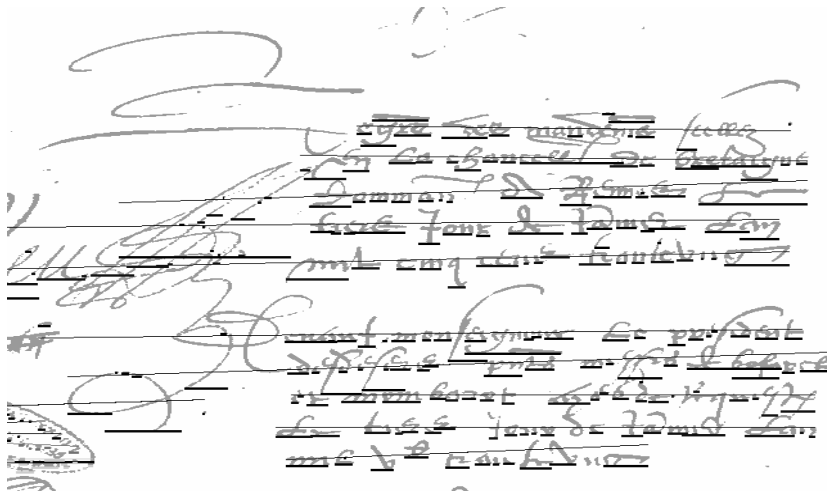


Figure 2-14 Résultat d'extraction des lignes basé sur le transformée de haugh

c) Méthodes utilisant d'autres techniques

Plusieurs autres méthodes basées sur diverses techniques ont été proposées dans la littérature.

Conclusion

Comme un petit résumé de cette chapitre, pour extraire la déférente caractère, mots, et lignes d'un texte il doit être segmenter par deux approches.

L'approche descendante segmenter le page en zone et sélection la zone de texte, en suite utilisé un méthodes d'extraction des lignes selon le caractéristique de texte, et en fin la segmentation de lignes en mots et les mots en caractère, et l'inverse dit mode ascendante.

Chapitre III

Conception et Architecteur

Introduction

Dans les chapitres précédents nous avons fait un survol sur les processus des reconnaissances de l'écriture, ainsi qu'un panorama sur la segmentation du document en général, et la segmentation du texte en lignes, mots et caractères en particulier.

Dans ce chapitre, nous allons évoquer voir notre contribution qui se présente par une méthode d'accélération d'extraction des lignes du texte, en utilisant des résolutions basses au lieu des résolutions souvent utilisée de 150 à 300 dpi.

3.1 Description générale du modèle proposé

D'abord, le système commence par l'acquisition de l'image à traiter. Ensuite, prépare l'image aux traitements futur en appliquant des opérations de prétraitement, ce génère une image binaire. Par la suite, avant d'appliquer l'un des algorithmes de segmentation des lignes le système approxime la taille du stylo de l'écriture. Selon la taille calculée, il peut définir la résolution basse cible. Puis, il applique l'échelle choisit dans le but de générer une image avec une résolution basse (XRM : Cross Résolution Model).

Ce modèle est utilisé dans la phase de segmentation de lignes dans le but d'accélérer le processus de segmentation.

L'architecture de ce modèle est présentée sur dans la figure ci-dessus .

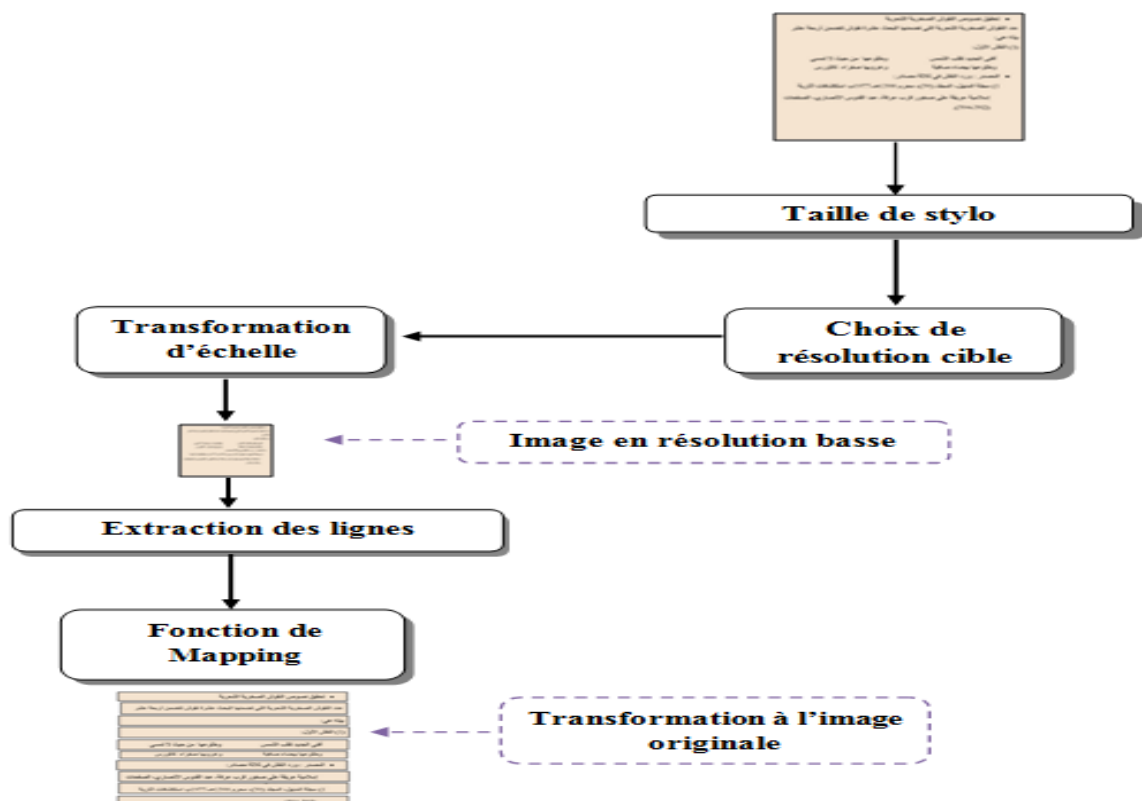


Figure 3-1 Architecture de modèle proposé (cross résolution faible)

3.1.1. Taille de stylo

Comme nous le savons, les images de texte varient de la taille de la police et le type de police du texte, l'écriture ou le texte écrit dans ces images est une collection de mots et de lettres, qui se représentent par un ensemble de pixels. Alors, il est nécessaire de connaître (approximer cette taille) cette taille afin de bien choisir la résolution cible voulut.

Dans le but d'approximer cette taille, le système lance un balayage horizontale, et pour chaque étape horizontale lance un autre balayage vertical afin de connaître pour chaque colonnes combien de pixels noirs successifs. Ensuite, considère la taille la plus fréquente comme la taille du stylo. La figure et l'algorithme ci-dessous illustre cette étape.

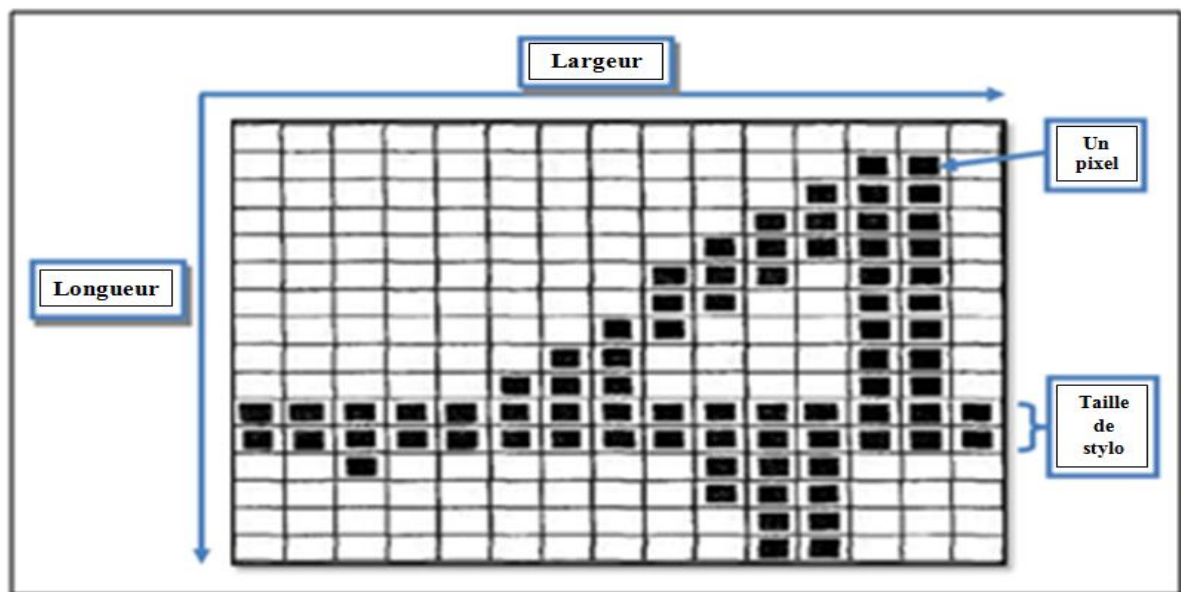


Figure 3-2 Caractéristique d'image

Algorithme 3-1 Taille de stylo

Image : Image noire et blanche ;

Début

M , F = tableau ;

Somme , taille ; // initialisant par 0

Pour chaque pixel P de Image **faire**

Si P = 1 alors somme = somme +1 ;

Sinon

 charger la somme dans le tableau M ;

 somme = 0 ;

Fin si

Fin pour

Pour chaque somme de **M** faire
calculer le fréquents de somme ;
charger la fréquents dans un tableau **F** ;
Fin pour
Pour chaque fréquents de **F** faire
Si fréquents > taille ;
taille = valeur de fréquents ;
Fin si
Fin pour
Fin

3.1.2. Choix de résolution cible

Le choix d'une résolution basse cible ou du modèle XRM voulu, nécessite la connaissance de la longueur et largeur de l'image, ainsi que la taille du stylo plus des informations sur des valeurs de teste empirique déjà réalisés sur des images de teste manuellement.

Il existe deux types de résolution, la première est la résolution faible (basse) et la seconde est la résolution haute, comme nous savons que toute modification de l'image changera leurs propriétés et conduit parfois à la perte de certaines informations.

Alors, cette étape est une étape cruciale dont laquelle nous redimensionnons l'image à une taille plus petite. Mais, comment peut-on choisir la plus petite résolution tout en gardant les propriétés importantes de l'image avec les variations des tailles et des longueurs.

Nous avons fait expérience, dont laquelle nous avons choisis un ensemble d'images, ensuite, nous avons appliqués les étapes suivantes:

- a) Calculer la taille du stylo de texte pour chaque image de teste;
- b) Appliquer les résolutions spécifiée 0,1 jusqu'à 0,9 sur chaque image;
- c) Choisir la plus petite résolution pour chaque taille de stylo, qui garde les informations importantes du texte.

Le tableau ci-dessous représente la taille de chaque stylo et la résolution qui peut être sélectionnée indépendamment de la taille de l'image.

Taille de stylo (pixel)	1	2	3	4	5	6 - 7 - 8
Résolution	0,8	0,6	0,4	0,3	0,2	0,1

Tableau 3-1 Tableaux de choix de résolution par taille de stylo

3.1.3. Transformation d'échelle

Après avoir choisir l'échelle de la résolution basse cible ou le modèle XRM voulu (ERB), il reste à redimensionner l'image en utilisant cette résolution, et ceci fait en suivant l'algorithme ci-dessous:

Algorithme 3-2 Transformation d'échelle d'image

Image1 : Image noire et blanche ;

Image2 : Image noire et blanche ;

Début

L1 , L2 = longueur ;

H1, H2 = largeur ;

l1, h1, l2 , h2 ;

scale ; // initialisé par 0

$L2 = scale * L1$; $H2 = scale * H1$;

Pour chaque **l2** de Image2 à L2 **faire**

Pour chaque **h2** de Image2 à H2 **faire**

$l2 = l1 / scale$; $h2 = h1 / scale$;

Copier le valeur(rgb) de pixel(h1, l1) de **Image1** dans (h2, l2) **Image2** ;

Fin pour

Fin pour

Fin

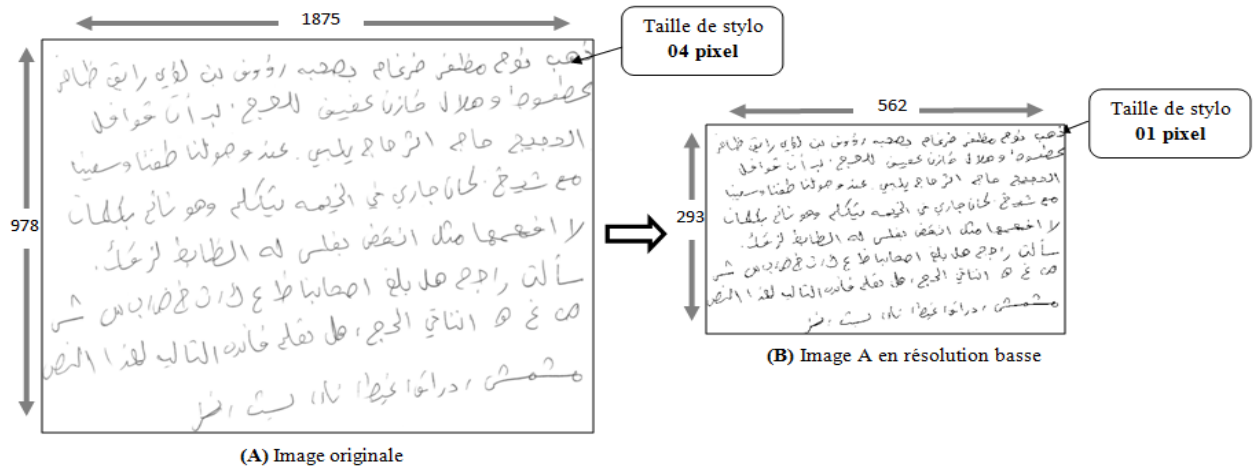


Figure 3-3 Transformation d'échelle d'image

3.1.4. Extraction des lignes

La méthode d'accélération à base de modèles de cross-résolution ne peut être appliquée que sur des méthodes de segmentation de texte en lignes descendantes. Pour cela, nous avons choisis trois méthodes de segmentation descendantes, à savoir:

- Méthode de projection horizontale ;
- Méthode de fenêtre glissante ;
- Méthode hybride.

3.1.4.1. Projection horizontale

Cette méthode est bien utilisée si le texte non incliné, ne contient pas de chevauchement entre les caractères, ou les documents imprimé. La méthode prend comme entrée l'image binaire (résultat de l'étape de prétraitement), puis, calcul une projection horizontale et utilise des seuils afin d'identifier les lignes de texte. Elle utilise les étapes suivantes :

- Calculer le somme de pixel noire de chaque ligne de l'image ;
- Calculer le seuil, qui est égale à le moyenne de la somme de tout les résultats des lignes ;
- Calculer les maximum et minimum de cette projection, si le nombre de pixel d'une ligne n'égal pas la valeur « seuil », la ligne est comme une maximum ;
- Tracer les lignes qui avant de chaque ligne nombre de pixel moins ou égal à valeur « seuil ».

Algorithme 3-3 Projection horizontale

```
Image : Image noire et blanche ;  
Début  
  L = Longueur de l'image;  
  H = Largeur de l'image;  
  Lp = Longueur de projection ;  
som , somp, seuil , x, le, he; // initialisé par 0  
  // calculer le somme de pixel (projection)  
Pour chaque le de Image à L faire  
  som = 0 ;  
Pour chaque he de Image à H faire  
  Si le valeur de pixel P =1 alors  
  som ++ ;  
  Charger le somme dans la liste de projection LP ;  
Fin si  
Fin pour  
Fin pour  
  // calculer le seuil  
Pour chaque valeur x dans LP faire  
  somp ++ ;  
  seuil = somp / Lp ;  
Fin pour  
  // calculer les minimum et les maximum  
Pour chaque x dans LP faire  
  Si le valeur de LP(x) > seuil alors  
  Charger x dans liste de maximum ;  
Fin si  
Sinon  
  Charger x dans liste de minimum ;  
  LP(x) = 0 ;  
Fin sinon  
Fin pour  
Fin
```

3.1.4.2. Projection par fenêtre

Appelée aussi projection partielle, elle est consiste à diviser l'image du texte en des colonnes, et par la suite applique localement une projection horizontale dans chaque colonne. Elle suit les étapes suivantes:

- Subdiviser l'image en plusieurs colonnes selon le choix, dans cette étude le nombre est 08 colonnes, si le nombre était grand le résultat était meilleur, mais plus de temps de calcul).
- Pour chaque colonne appliqué une projection horizontale.
- Le résultat final (nombre des lignes extrait) est le résultat plus grande des parties.

Algorithme 3-4 Projection par fenêtre

Image : Image noire et blanche;

Début

HE = largeur de l'image;

part ; // initialisé par 0

 // partitionner la largeur de l'image en 08 parties

part = HE/8 ;

Pour chaque part **faire**

 calculer le somme de pixel (projection) ;

 calculer le seuil ;

 calculer les minimum et les maximum ;

Fin pour

FIN

3.1.4.3. Méthode hybride (horizontale et par fenêtre) [22]

La méthode mixte consiste à divisé l'image en quatre partie, le nombre des lignes sur quatre. Dans la première et la troisième partie en applique une projection horizontale. Mais, dans la deuxième et quatrième une projection partielle. Elle suit les étapes suivantes :

- Subdiviser la longueur de l'image en quatre parties ;
- Dans la première et la troisième partie on applique une projection horizontale ;
- Dans la deuxième et la quatrième on applique une projection partielle.

Algorithme 3-5 Projection hybride

```

Entrée : Image noire et blanche bImage2;
Début
LE = longueur de l'image;
HE = largeur de l'image ;
partl , part; // initialisé par 0
// partitionner longueur de l'image en 04 parties
partl = LE/4 ;
Pour le partie « 1 » et partie « 3 » faire
    // appliqué le projection horizontale ;
    calculer le somme de pixel (projection) ;
    calculer le seuil ;
    calculer les minimum et les maximum ;
Fin pour
Pour le partie « 2 » et partie « 4 » faire
    part = HE/8 ;
    // appliqué le projection partiel pour chaque part ;
Fin pour
FIN

```

3.1.4.4. Extraction des caractéristique de ligne

Pour localiser les lignes de texte il faut savoir pour quoi utiliser cette ligne, et quelle zone il faut extraire selon l'application de reconnaissance. Chaque zone de ligne est divisée en trois sous zone, zone médiane, zone supérieur, et zone inférieur. Ces zones se divisent elles-mêmes en quatre lignes: ligne de base, ligne de base supérieur, ligne de base inférieur, et ligne de haut.

a. Localisation de ligne de base (ligne de référence) :

La localisation des lignes de référence peut s'accompagner d'une étape de normalisation qui implique le redressement de l'écriture et éventuellement le redressement des caractères qui peuvent également présenter une certaine inclinaison par rapport aux lignes de référence.

Pour localiser cette ligne à partir d'histogramme de projection calculer le maxima de chaque ensemble des lignes situé avant le ligne de minima.

Algorithme 3-6 Localisation de ligne de base

Image : Image noire et blanche;

Début

Lm = liste des maximum ;

Lp = liste de projection ;

max , x, y; // initialisé par 0

Pour chaque x dans Lp **faire**

Pour chaque y dans Lm **faire**

Si Lm(y) = x **alors**

Si Lp(x+1) = 0 **alors**

Si Lm(x) > max **alors**

max = Lm(x) ; // le ligne de base est « x »

Charger x dans la liste des lignes de base LB ;

Fin si

Fin si

Fin si

Fin pour

Fin pour

Fin

b. Localisation de ligne haut (supérieur) :

Elle nous permet de détecter les hampes et les points diacritiques hauts. Pour localiser la zone ascendante ou supérieure, Nous balayons l'image ligne par ligne, colonne par colonne et de haut en bas, dès que nous trouvons le premier pixel noir nous considérons cette ligne comme une ligne haute. Enfin, la zone supérieure est cernée entre la ligne haute et la ligne de base haute.

Algorithme 3-7 Localisation de ligne haute

Image : Image noire et blanche ;

Début

Lm = liste des minimum ;

Lp = liste de projection ;

lh , x, y; // initialisé par 0

Pour chaque x dans Lp **faire**

Pour chaque y dans Lm **faire**

Si Lm(y) = x **alors**

```

Si  $Lp(x+1) > 0$  alors
    lh = x ;
    // le ligne de haut est « lh »
    Charger lh dans la liste des lignes de haut LH ;
    Fin si
Fin si
Fin pour
Fin pour
Fin

```

c. Localisation de ligne inférieure (inferieure):

Elle nous permet de détecter les jambages et les points diacritiques bas. Pour localiser la zone descendante, nous balayons l'image ligne par ligne, colonne par colonne mais cette fois ci de bas en haut, dès que nous trouvons le premier pixel noir nous considérons sa ligne comme une ligne basse. Nous obtenons alors la zone inférieure qui est cernée entre la ligne de base et la ligne inférieure.

Algorithme 3-8 Localisation de ligne inférieure

```

Image : Image noire et blanche;
Début
    Lm = liste des minimum ;
    Lp = liste de projection ;
    li =0 ;
Pour chaque x dans Lp faire
    Pour chaque y dans Lm faire
        Si  $Lm(y) = x$  alors
            Si  $Lp(x+1) = 0$  alors
                li = x ;
                // le ligne de haut est « li »
                Charger li dans la liste des lignes de l'inférieure LI ;
            Fin si
        Fin si
    Fin pour
Fin pour
Fin

```

3.1.5. Fonction de mapping

Cette phase consiste à retrouver les coordonnées des cadres des lignes en divisant les coordonnées des cadres trouvées dans les étapes précédentes par le facteur d'échelle choisit dans la troisième étape (Transformation d'échelle).

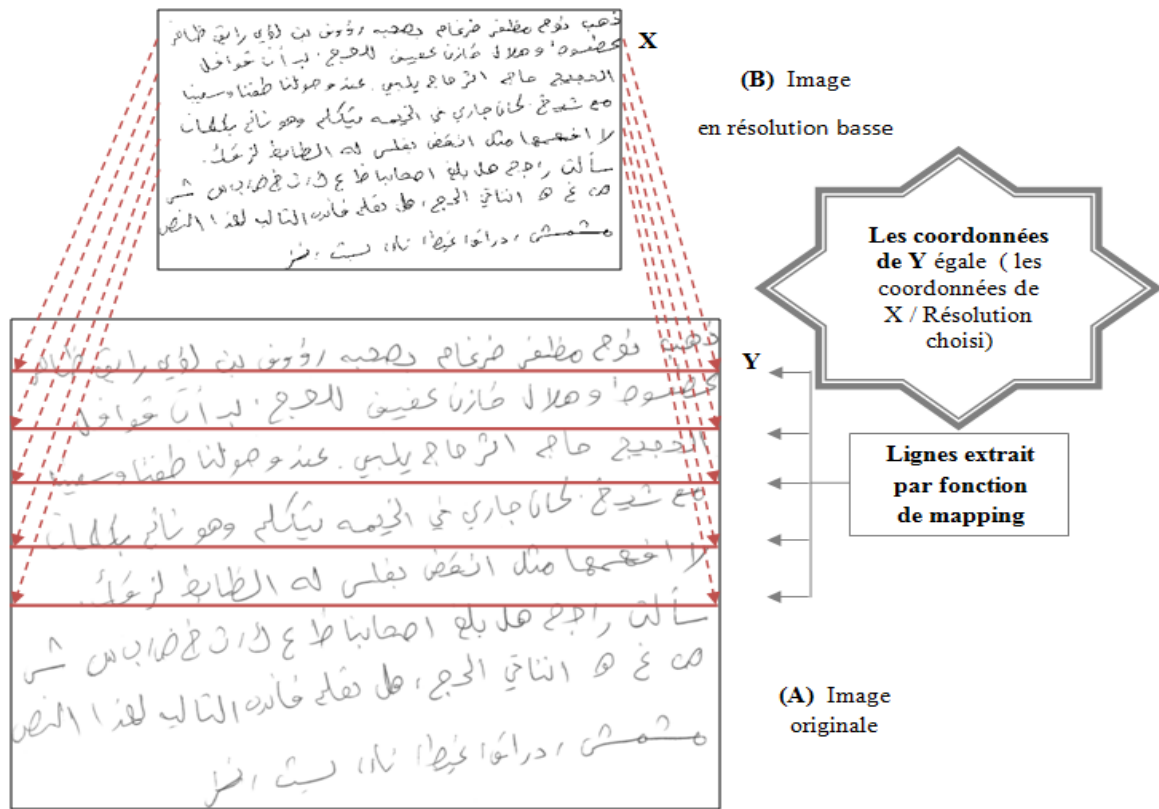


Figure 3-4 Fonction de mapping

Conclusion

Dans ce chapitre, nous avons commencé par illustrer l'architecture globale du système proposé. Dans laquelle nous avons présenté en détail notre contribution, qui se présente en une méthode d'accélération d'extraction des lignes de texte en utilisant des modèles de cross résolutions (avec résolutions basses). Par la suite, le système est capable de retrouver les coordonnées des cadres des lignes dans l'image originale en utilisant des fonctions de mapping.

Le prochain chapitre, présente les résultats obtenus par notre méthode, ainsi qu'une validation en comparants nos résultats avec trois méthodes différentes.

Chapitre IV

Test et Résultats

Introduction

Dans le chapitre précédent, nous avons vu les détails de conception du système et de la méthode d'accélération proposée. Elle consiste à trouver un modèle en résolution basse de l'image originale dans le but de trouver les coordonnées des cadres des lignes dans ce modèle, ensuite, utilise des fonctions de mapping afin de retrouver les coordonnées de ces cadres dans l'image originale.

Dans ce chapitre, nous allons voir d'une part une petite présentation sur l'interface de l'application. D'autre part, nous allons présenter une comparaison des résultats obtenus de trois méthodes de test.

4.1 Langage utilisé

Nous avons utilisé le langage de programmation Java, parce que c'est un langage efficace et appropriée pour ce travail et riche pour le traitement d'image, y compris plusieurs caractéristiques tel que:

- La portabilité des logiciels ;
- La réutilisation de certaines classes déjà développées ;
- La possibilité d'ajouter à l'environnement de base des composants fournis par l'environnement lui même ;
- La quasi-totalité de contrôle de Windows (boutons, boites de saisies, listes déroulantes, menus ...etc.) qui sont représentés par classes;

4.2 Interfaces de l'application

En lançant l'application, l'interface principale présentée par la figure ci-dessous s'affiche. Elle contient un panneau d'image, et des informations sur l'image, le panneau d'image et afficher chaque modification sur l'image.

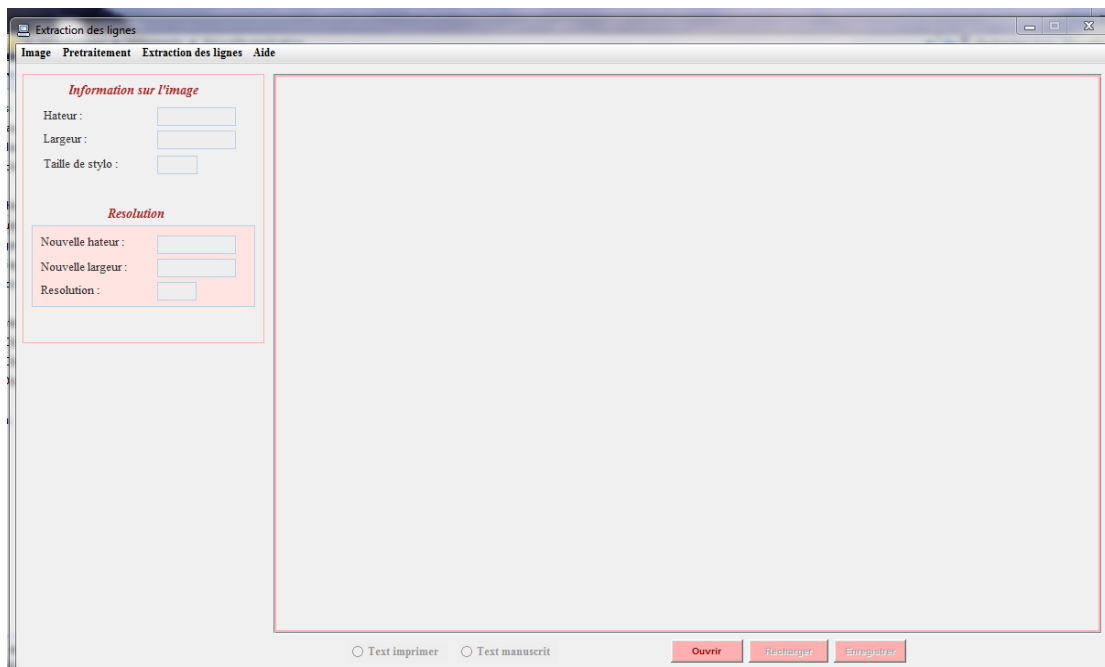


Figure 4-1 Interface d'application

a. Prétraitement :

Dans ce menu nous effectuons la transformation de l'image en forme binaire en cliquant sur **binarisation**, le choix de résolution et les choix de lignes. Pour la **résolution**, le programme calcule la taille de stylo et choisir la résolution à appliqué sur cette image.

Le programme affiche les informations de traitement, la **taille de stylo** et la **résolution** choisir.

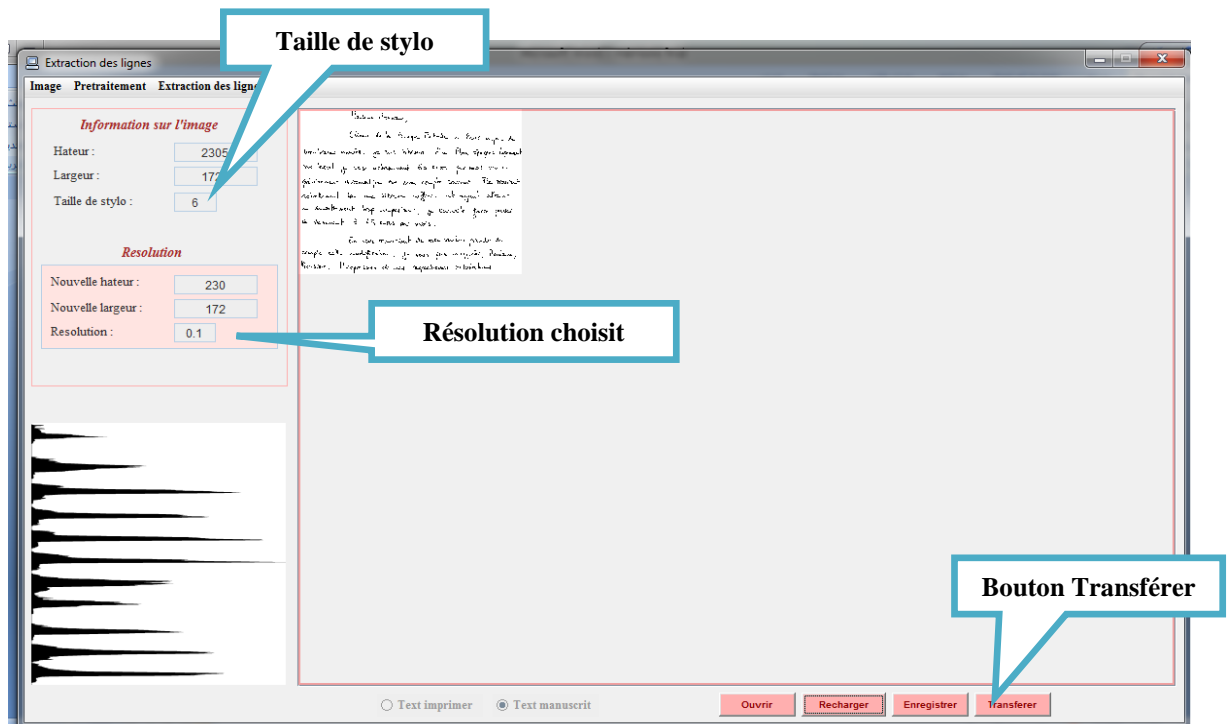


Figure 4-2 Information sur l'image et les buttons

b. Extraction des lignes

Dans ce menu on choisit la méthode d'extraction à appliquer, soit projection horizontale, ou projection partielle ou la méthode hybride. Les lignes détectées dans sont la ligne de base, la ligne inférieure et la ligne haute.

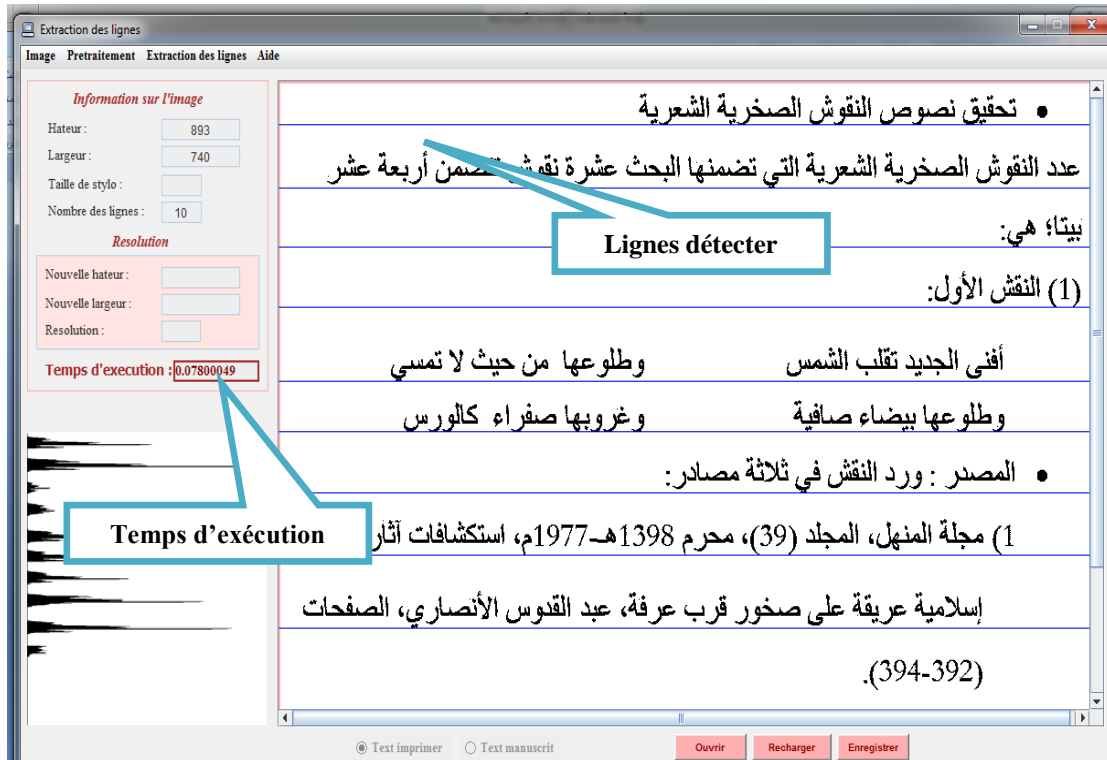


Figure 4-3 Extraction des lignes par projection horizontale

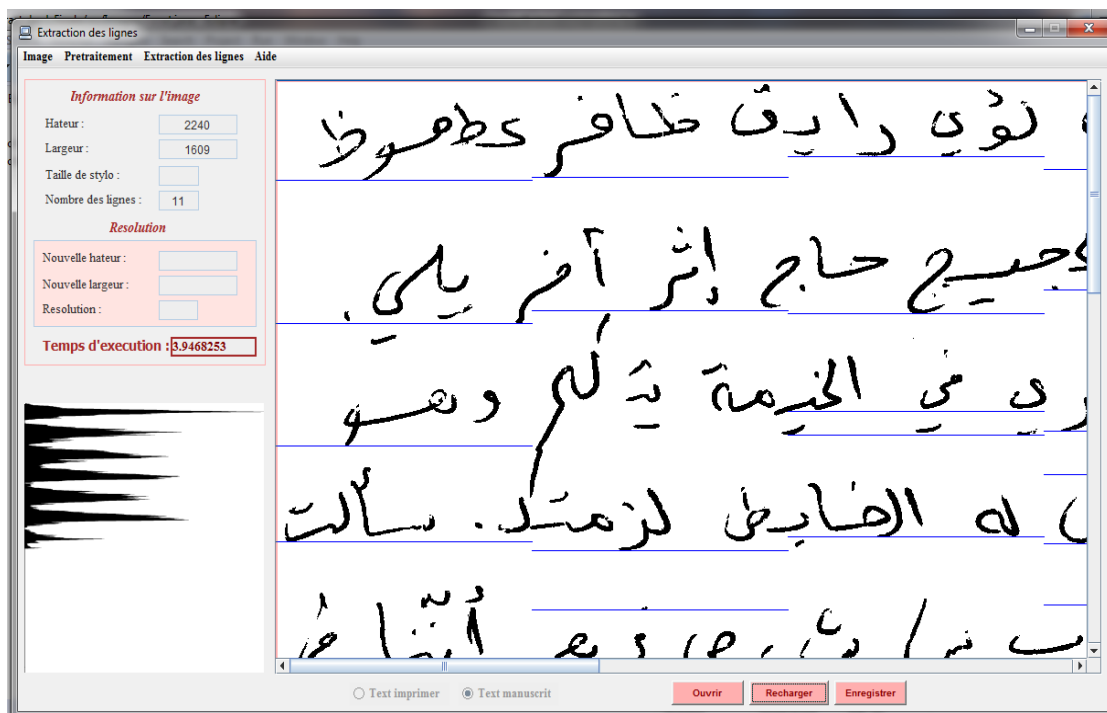


Figure 4-4 Extraction des lignes par Projection par fenêtre

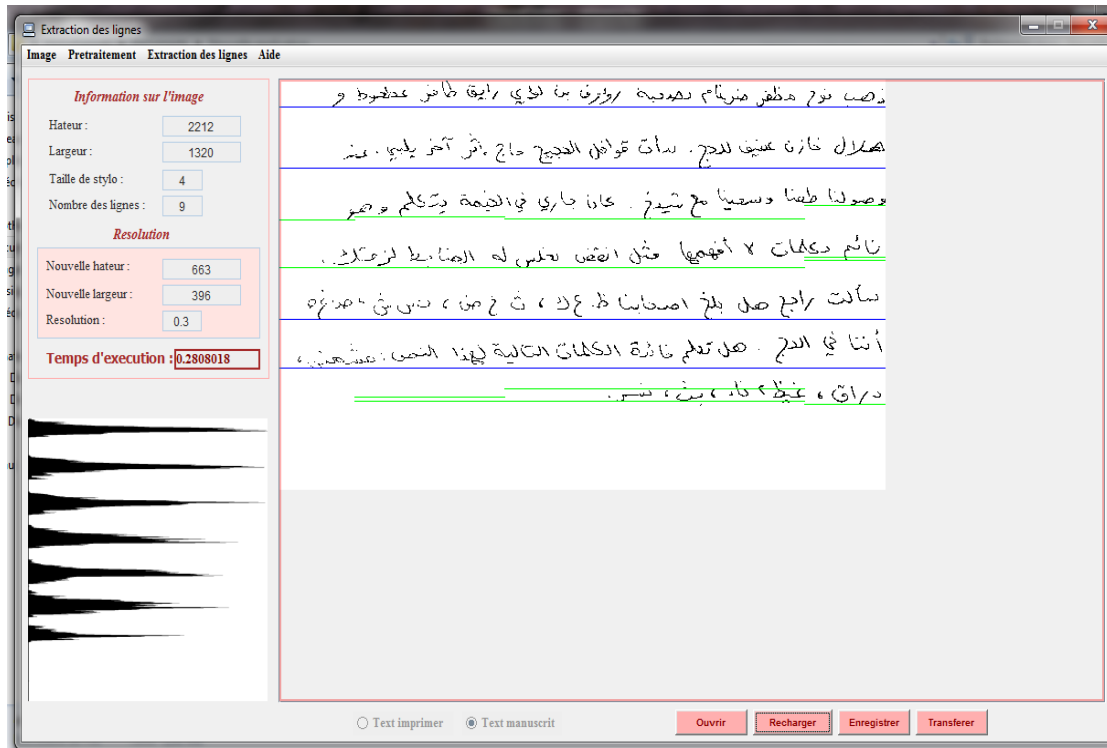


Figure 4-5 Extraction des lignes par méthode hybride

c. Transformation en image originale

Ce traitement es réalisée en cliquant sur le bouton **transférer** qui est activée après le choix de la résolution d’image, il permet de calculer les coordonnées originale des lignes extrait à partir d’image résolu sur l’image originale.

4.3 Test

Dans cette étude pour tester l’efficacité de ce modèle nous avons utilisé des images de basses résolutions (modèle XRM). De texte arabe manuscrit de la base KHATT réalisée par l’université de Roi Fahd du pétrole et des minéraux (KFUPM).

Nous avons appliqué les trois méthodes sur les images de test selon deux cas, le premier avec résolution normale, et le second avec résolution basse (modèles XRM), et à la fin nous avons calculé le temps d’exécution pour chaque cas.

Le tableau suivant illustre des exemples des résultats obtenus.

Image	Nombre de lignes réel	Résolution normale						Résolution basse					
		Projection horizontale		Projection par fenêtre		Projection hybride		Projection horizontale		Projection par fenêtre		Projection hybride	
		Nbl / Nbl Total	Temps (s)	Nbl / Nbl Total	Temps (s)	Nbl / Nbl Total	Temps (s)	Nbl / Nbl Total	Temps (s)	Nbl / Nbl Total	Temps (s)	Nbl / Nbl Total	Temps (s)
1	5	4/6	0,14	5/7	1,75	4/7	0,27	4/4	0,07	5/5	0,45	4/4	0,07
2	7	7/7	0,22	7/11	4,19	6/7	0,54	7/7	0,12	7/7	0,28	6/7	0,15
3	5	5/5	0,16	5/6	2,67	4/4	0,39	5/5	0,06	5/5	0,44	3/3	0,16
4	7	7/9	0,10	7/10	3,16	6/8	0,11	5/5	0,07	7/7	0,22	6/7	0,07
5	6	6/6	0,17	6/10	2,91	5/8	0,49	6/6	0,06	6/6	0,37	5/7	0,10
6	6	6/6	0,12	6/9	2,18	6/7	0,41	6/6	0,12	6/6	0,59	6/6	0,12
7	7	6/7	0,14	7/11	2,32	6/8	0,54	6/7	0,12	7/7	0,53	5/7	0,18
8	5	5/6	0,17	5/8	3,46	5/10	0,47	5/5	0,08	5/5	0,54	5/6	0,14
9	6	4/5	0,34	6/11	6,13	6/7	0,59	2/2	0,19	6/6	1,15	6/6	0,14
10	7	6/10	0,10	7/10	3,08	5/9	0,32	6/6	0,06	6/6	0,35	4/5	0,11
11	7	7/7	0,31	7/11	5,34	7/11	0,33	6/6	0,12	7/7	0,43	6/7	0,12
12	7	7/7	0,16	7/10	2,82	6/9	0,36	6/6	0,06	7/7	0,15	5/7	0,11
13	6	6/6	0,13	6/8	2,77	5/5	0,33	6/6	0,05	6/6	0,11	4/5	0,10
14	5	5/5	0,10	5/8	4,10	5/8	0,50	5/5	0,09	5/5	0,23	4/7	0,19
15	6	6/8	0,39	6/10	5,13	6/10	0,32	6/6	0,16	6/7	0,25	6/9	0,20
16	5	5/5	0,05	5/6	2,13	4/4	0,19	5/5	0,04	5/5	0,08	4/4	0,07
17	5	5/6	0,27	5/7	5,94	5/6	0,35	4/5	0,15	5/6	0,21	5/6	0,18
18	6	6/6	0,43	6/10	5,57	6/12	0,31	6/6	0,14	6/6	0,50	6/9	0,18
19	4	4/5	0,15	4/6	8,59	4/6	0,56	3/4	0,10	4/5	0,25	4/6	0,17
20	5	5/5	0,14	5/8	7,07	5/8	0,45	5/5	0,07	5/5	0,41	5/7	0,10

Nbl : nombre des lignes .

Nbl Total : nombre total des lignes détectées.

Tableau 4-1 Résultats de test

4.3.1. Comparaison entre les résultats :

Pour chaque méthodes en comparer entre le temps d'exécution de méthode résolution normale et avec résolution basse, ce comparaison illustré dans les graphe suivants :

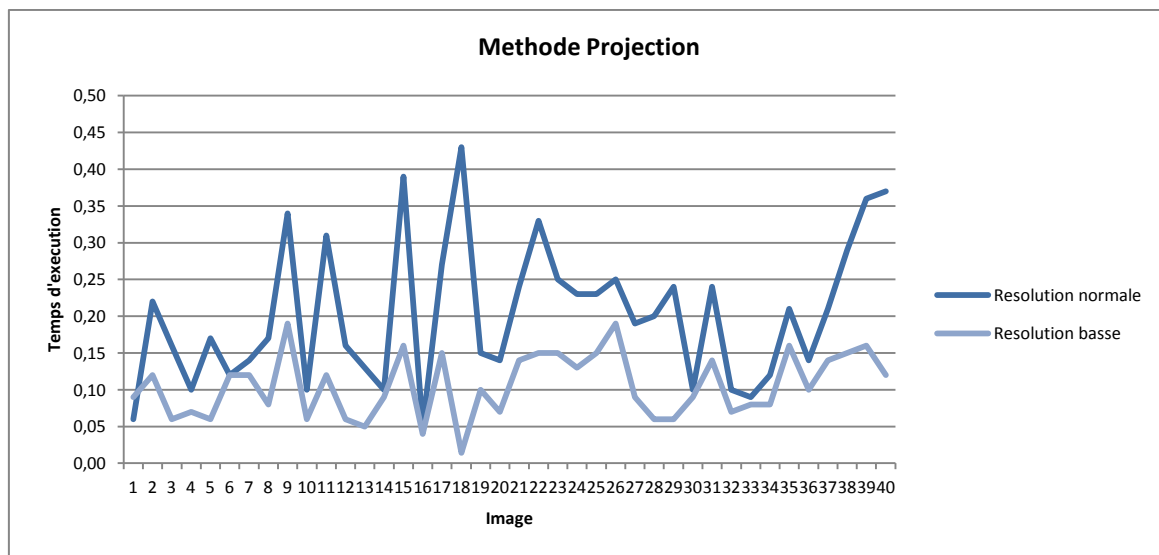


Figure 4-6 Illustration de temps d'exécution sans résolution

La première méthode est la projection horizontale, dans cette méthodes pas de grande déférence est 1.8 fois. Dans le cas de la résolution basse on trouve une moyenne de 0.11 s, et avec la résolution basse une moyenne de 0.20 s.

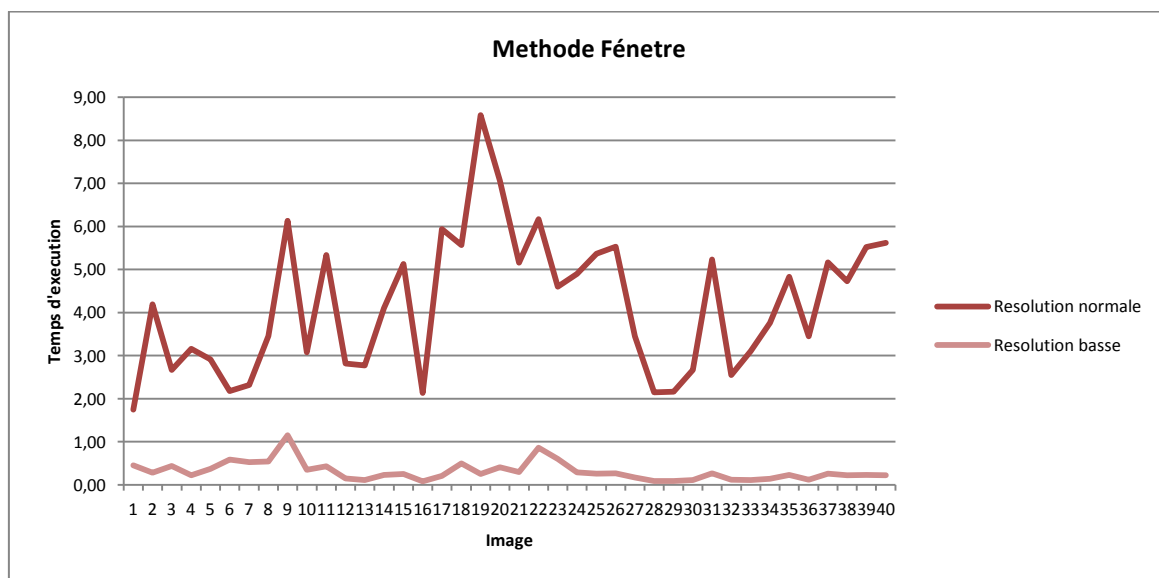


Figure 4-7 Illustration de temps d'exécution de méthode par fenêtre

La deuxième méthode est la projection partielle, dans cette méthode le temps d'exécution avec le modèle de résolution est plus rapide par rapport à la méthode sans résolution la différence est 13 fois, la différence est justifiée par la taille de l'image petite.

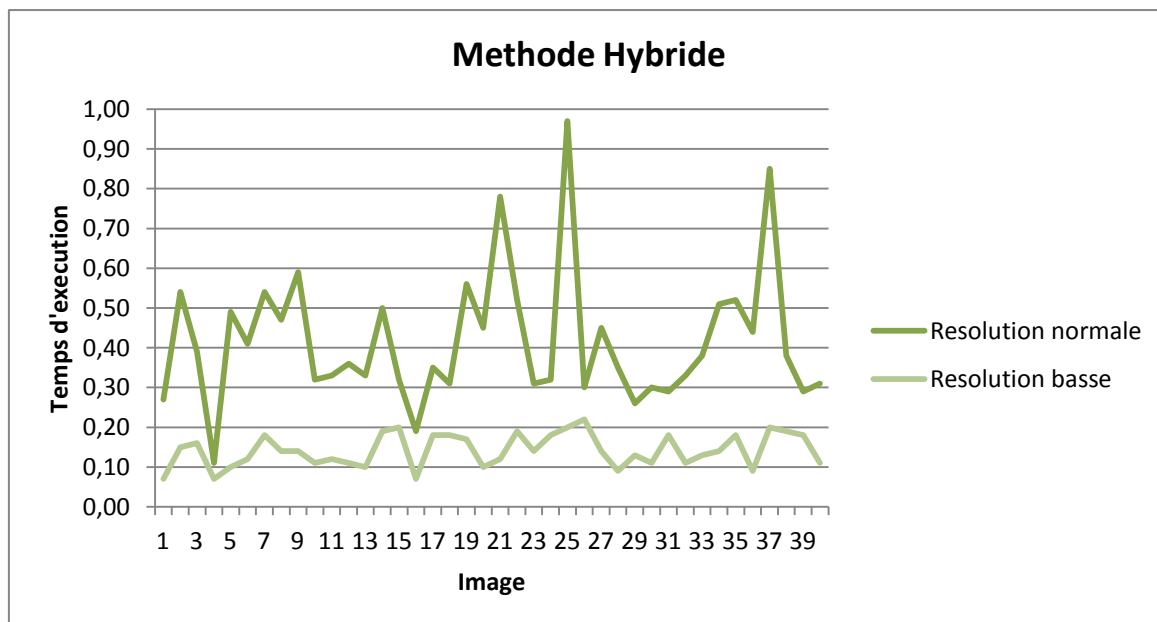


Figure 4-8 Illustration de temps d'exécution de méthode hybride

Le dernier méthode est le méthode hybride, qui appliqué les deux méthodes projection et partiel, dans ce méthode la différence 03 fois est moyenne parce que le temps est modifier sauf dans les parties qui appliqué le deuxième méthode.

4.3.2. Comparaison entre les méthodes :

En comparer les rapidité entre ces méthodes el le meilleur méthode en deux cas, par les moyenne des résultats présenter dans les tableaux au-dessus :

Méthode \ Taux	Taux de détection (%)	Temps d'exécution (s)
Projection	97	0,20
Fenêtre	100	4,19
Hybride	91	0,42

Tableau 4-1 Les moyenne de résultats sans résolution

Dans ce tableau le temps d'exécution meilleur est le temps de la méthode de projection horizontale, mais, la meilleure détection est celle de la méthode de fenêtre glissante.

Taux Méthode	Taux de détection (%)	Temps d'exécution (s)
Projection	92	0,11
Fenêtre	99	0,31
Hybride	88	0,14

Tableau 4-2 Les moyenne des résultats avec résolution basse

Comme dans le tableaux précédent, la méthode projection est le meilleur temps, et pour le détection le méthode fenêtre.

Conclusion

Nous avons présenté dans ce chapitre les résultats de test obtenus des trois méthodes de segmentation : projection horizontale, fenêtre glissante et méthode hybride des deux précédentes. Ces sont réalisés sur la base de texte manuscrit KHAT. Les résultats obtenus démontrent que l'utilisation des modèles de cross résolution ou de résolution basse permet d'accélérer de manière remarquable l'extraction des lignes par les méthodes de segmentation descendantes.

Conclusion générale

L'objectif de notre travail est l'amélioration de segmentation de texte en lignes dans le cas des méthodes ascendantes par l'utilisation des modèles de cross résolution ou de résolution basse ce qui permet d'accélérer remarquablement le temps d'extraction.

Durant les dernières années plusieurs méthodes d'extraction de lignes de texte ont été développées en concentrant sur la qualité et la précision des techniques réalisées. D'une part, ces techniques sont généralement basées sur l'utilisation d'un ensemble de traitements qui inclut des différents balayages ou scans sur l'image du texte à traiter. D'autre part, ces techniques utilisent souvent des images avec une grande résolution de 150 à 300 dpi afin de garder les informations essentielles du texte. Alors, la minimisation de la taille de l'image va minimiser directement le temps de calcul des différentes méthodes.

Nous avons dégagé de cette étude une contribution qui peut profiter de l'idée de création des modèles de cross résolution pour surmonter le problème du temps de calcul important perdu durant l'extraction des lignes du texte.

Le choix de la bonne résolution basse est lié par la taille ou plus pratiquement les tailles des stylos présentent dans l'image du texte à traiter. Il est évident, que la petite taille sera prise comme référence pour choisir le bon modèle de cross résolution. Ces modèles permettent d'accélérer significativement l'extraction des lignes et donnent les mêmes résultats qu'avec l'image en résolution de 150 ou de 300 dpi.

Le travail que nous avons réalisé durant ce mémoire, constitue une étape et un apport pour la segmentation en lignes de l'écriture, cependant nous pensons qu'il peut être amélioré, et étendu par les points suivants:

- Tester à appliquer le principe des modèles XRM sur des niveaux plus détaillés que les lignes tels que mots et caractères.
- Le principe des modèles XRM peut être utilisé pour faire une squelettisation d'un texte donné..

Bibliographie

- [1] **S. Haitaamar** «Segmentation de textes en caractères pour la reconnaissance optique de l'écriture Arabe» Université El-hadj Lakhdar, BATNA, 2007.
- [2] **M. Amrouch** «Reconnaissance de caractères imprimés et manuscrits, textes et documents basée sur les modèles de Markov cachés» Université Ibn Zohr, Agadir, 00.
- [3] **E. Mamouni** «Sélection expérimentale de modèles SVM multi-classes application à la reconnaissance des caractères arabes» Université des Sciences et des Technologies d'Oran Mohamed BOUDIAF, Oran, 2012.
- [4] **R. Azizi** «Une approche hybride pour la reconnaissance d'écriture arabe manuscrite» Université de Constantine, Constantine, 2007.
- [5] **F. Menasri** «Contributions à la reconnaissance de l'écriture arabe manuscrite» Université PARIS Descartes, PARIS, 2008.
- [6] **Y. Essaady** «Contribution au développement d'approches de reconnaissance automatique de caractères imprimés et manuscrits, de textes et de documents Amazighes» Université Ibn Zohr, Agadir, 2012.
- [7] **A. Boukharouba** «Contribution à la segmentation et à la reconnaissance de l'écriture arabe manuscrite» Université Mentouri, Constantine, 2011.
- [8] **F. Zaiz** «Les Supports Vecteurs Machines (SVM) pour la reconnaissance des caractères manuscrits arabes» Université Mohamed Khider, Biskra, 2010.
- [9] **E. Poisson** «Architecture et apprentissage d'un système hybride neuro-markovien pour la reconnaissance de l'écriture manuscrite en ligne» Université de Nantes, Nantes, 2005.
- [10] **I. Bengussoum** «Proposition d'un modèle de classificateur logique, application à la reconnaissance du texte arabe imprimé.» Université d'El-oued, El-oued, 2014.
- [11] **Z. Mehennaoui** «Reconnaissance de l'écriture Arabe manuscrite à base des machines à vecteurs de supports» Université Badji-Mokhtar, Annaba, 2006.
- [12] **M. Kadri** «Méthode de reconnaissance de l'écriture arabe manuscrite en utilisant les réseaux neuronaux» Université des Sciences et des Technologies Mohamed BOUDIAF d'Oran, Oran, 2010.
- [13] **S. Nemouchi** «Reconnaissance de l'Écriture Arabe par Systèmes Flous» Université Badji Moukhtar, Annaba, 2010.
- [14] **M. Kamel** «Reconnaissance des formes appliquée à l'écriture arabe manuscrite par

des multiclassifieurs» Université Mohamed Khider Biskra, Biskra, 2010.

- [15] **S. Nebti** «Reconnaissance de Caractères Manuscrits par Intelligence Collective» Université Ferhat Abass-Sétif, Sétif, 2013.
- [16] **Z. Mehennaoui** «Reconnaissance de l'écriture Arabe manuscrite à base des machines à vecteurs de supports» Université Badji-Mokhtar, Annaba, 2006.
- [17] **D. Xavier** «Contributions à la reconnaissance de l'écriture cursive à l'aide de modèles de Markov cachés» Université René Descartes - Paris V, Paris, 2004.
- [18] **S. Likforman** «Apport du traitement des images à la numérisation des documents manuscrits anciens» Paris, 2003.
- [19] **A. Bennisri, A. Zahour et B. Taconet** «Extraction des lignes d'un texte manuscrit arabe» chez *Vision Interface*, Canada, 1999.
- [20] **A. Zahour, S. Likorman, W. Boussalaa et B. Taconet** «Text Line segmentation of historical Arabic documents» France, 2007.
- [21] **S. Likforman** «Une méthode de résolution des conflits d'alignements pour la segmentation des documents manuscrits» chez *Traitement et Communication de l'Information*, Paris, 1995.
- [22] **Y. Mokhtari et A. Yousfi** «Segmentation of Arabic handwritten text to lines» chez *The International Conference on Advanced Wireless, Information, and Communication Technologies*, Maroc, 2015.