

الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي والبحث العلمي
RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE
SCIENTIFIQUE
UNIVERSITÉ 20 AOUT 1955- SKIKDA



**Mémoire de Fin d'Etudes en vue de l'Obtention du
Diplôme de Master2 en Informatique**

Thème

Conception et Développement d'une
Application OCR-OMR : Correction
Automatique des Examens à base de
QCMs comme Etude de Cas

Présenté par :

- Lemchema Mounir
- Zouiten Abdelhak

Encadré par :

Dr. Mawloud MOSBAH

Année universitaire 2024-2025

Remerciement

Louange à Allah, Le Tout-Puissant, qui nous a accordé la force, la patience et la volonté d'achever ce travail.

Nous tenons à exprimer nos profonds remerciements et notre sincère gratitude au Dr. Mawloud MOSBAH, notre encadreur, pour la confiance qu'il nous a témoignée, ses conseils précieux, sa disponibilité constante et son encadrement rigoureux. Ses orientations avisées et ses remarques constructives ont été déterminantes dans la réalisation et l'aboutissement de ce projet.

Nos remerciements vont également à l'ensemble des enseignants et du personnel administratif de notre département pour la qualité de l'encadrement pédagogique et scientifique dont nous avons bénéficié tout au long de notre parcours universitaire.

Nous adressons nos vifs remerciements à nos familles respectives, en particulier nos mères, frères et sœurs, ainsi que nos enfants, pour leur affection, leurs encouragements et leurs prières permanentes. Leur soutien moral et leur compréhension ont été d'un grand réconfort et d'une motivation inestimable.

Nous n'oublions pas de remercier nos collègues et amis pour leur soutien, leur coopération et l'esprit de camaraderie qui a régné entre nous durant toutes ces années d'études.

À toutes les personnes qui, de près ou de loin, ont contribué à la réalisation de ce mémoire, nous adressons nos sincères remerciements et notre profonde reconnaissance.

Dédicace (1)

À la mémoire de mon cher père, qu'Allah lui accorde Sa miséricorde
et l'accueille dans Son vaste paradis.

Son souvenir, son courage et ses valeurs demeurent à jamais gravés
dans mon cœur et m'accompagnent à chaque étape de ma vie.

À ma mère, pour son amour inconditionnel, ses sacrifices, ses
prières et sa patience infinie. Qu'Allah la protège et la préserve.

À mes enfants bien-aimés, source de joie et de motivation. Que ce
travail soit pour eux un témoignage d'amour du savoir, un
encouragement à poursuivre leurs études avec ambition,
persévérance et foi en l'avenir.

À mes frères et sœurs, compagnons de vie et de cœur, pour leur
présence précieuse, leur affection et leur soutien constant.

À ma famille et à tous ceux qui ont cru en moi et m'ont
accompagné de près ou de loin, que ce soit par leurs paroles, leurs
conseils ou leur bienveillance.

Lemchema Mounir

Dédicace (2)

À la mémoire de deux femmes qui ont marqué ma vie à jamais :

Mon épouse, celle qui fut mon soutien, ma force et ma lumière,
Ma mère, celle qui m'a donné la vie, élevé avec amour et guidé par
ses valeurs.

Même absentes, votre amour continue de m'accompagner à chaque
étape.

Ce mémoire vous est dédié, avec tout mon respect, ma tendresse et
ma prière.

Que Dieu vous accorde Sa miséricorde et vous ouvre les portes de Son
paradis.

Et à mes enfants,

mes trésors, ma raison d'avancer malgré l'épreuve,
ce travail vous revient aussi.

Puisse-t-il vous inspirer, vous encourager, et vous rappeler que même
dans la douleur,

l'effort et la foi mènent toujours à l'accomplissement.

ZOUITEN abdelhak

Table des matières

REMERCIEMENT	
DEDICACE (1).....	
DEDICACE (2).....	
TABLE DES FIGURES.....	
RESUME	
ملخص.....	
ABSTRACT	
INTRODUCTION GENERALE	1
PARTIE 1 : ETAT DE L'ART.....	2
CHAPITRE 01 : RECONNAISSANCE AUTOMATIQUE DU MANUSCRIT.....	3
I. Introduction.....	4
II. Définitions, Principes de base, et étapes.....	4
III. Prétraitement.....	8
1. Dégradation et bruits des images de documents	8
2. Quelques filtres utilisés en prétraitement, traitement et post-traitement pour la binarisation	10
a) Filtre de Gabor	10
b) Distribution Gaussienne Généralisée (GGD).....	11
c) Filtre de Wiener	12
d) Filtrage Shrink et Swell	12
e) Filtre d'entropie locale.....	13
f) Filtre médian.....	13
g) Filtre bilatéral	14
h) Filtre de Kalman.....	15
i) Filtre de dilatation (Swell filter)	15
j) Transformée Curvelet	16
k) Filtre d'étirement de contraste.....	17
3. La binarisation.....	17
a) Binarisation d'Image Basée sur le Clustering	20
b) Binarisation Basée sur K-means.....	20
c) Binarisation Basée sur la Classification Floue	21
d) Binarisation d'Image Basée sur le Seuillage.....	22
e) Binarisation basée sur les réseaux de neurones	31
f) Combinaison des techniques de binarisation	32
4. La segmentation.....	33
a) Segmentation par seuillage	33
b) Segmentation par croissance de régions	34

c)	Segmentation basée sur les contours	35
IV.	<i>Techniques de reconnaissance</i>	37
1.	L'approche à base des modèles statistiques	38
2.	Reconnaissance par des réseaux de neurones.....	38
3.	Reconnaissance Optique de Caractère (ROC) contextuelle	38
V.	<i>Post-traitement</i>	38
1.	Utilisation de modèles de langage	39
2.	Dictionnaires et lexiques.....	39
3.	Normalisation du texte	39
4.	Re-segmentation ou alignement texte-image	39
VI.	<i>Evaluation</i>	39
1.	Les collections d'évaluation	40
2.	Les métriques.....	40
a)	F-measure (FM) :	41
b)	Pseudo-F-Measure.....	41
c)	PSNR	42
d)	La distance de distorsion réciproque (Distance Reciprocal Distortion Metric: DRD)	42
e)	La métrique du ratio négatif (Negative Rate Metric: NRM).....	43
f)	La métrique de pénalité à base de mal-classification (Misclassification Penalty Mteric: MPM)	43
VII.	<i>Les approches de reconnaissance optique</i>	44
VIII.	<i>Conclusion</i>	45
CHAPITRE 02 :	QUESTIONNAIRE A CHOIX MULTIPLE (QCM).....	46
I.	<i>Introduction</i>	47
II.	<i>Le Système Pédagogique</i>	48
III.	<i>Interrogation de l'Apprenant</i>	48
IV.	<i>Le QCM : Définition et enjeux</i>	49
1.	Différents types de QCM.....	50
a)	QCM à réponse unique	50
b)	QCM à réponses multiples.....	50
c)	QCM à pondération	50
d)	QCM à correction négative.....	50
2.	Avantages et limites du QCM.....	50
a)	Avantages	50
b)	Limites	51
3.	Système de Correction à basse de QCM	51
a)	Correction manuelle	51
b)	Correction semi-automatique.....	52
c)	Correction automatique par scanner OMR (Optical Mark Recognition).....	52
V.	<i>Conclusion</i>	53
PARTIE 2 :	CONTRIBUTION : ANALYSE, CONCEPTION, ET IMPLEMENTATION	55
CHAPITRE 03 :	ANALYSE ET CONCEPTION	56

I.	<i>Introduction</i>	57
II.	<i>Le langage de modélisation Unifié (UML)</i>	57
1.	Analyse des besoins :	57
2.	Conception :	57
3.	Implémentation(codage) :	58
4.	Tests et validation :	58
5.	Déploiement :	58
6.	Maintenance évolutive et corrective :	58
III.	<i>Analyse des Besoins</i>	58
1.	Motivation	58
2.	Notre Approche	59
IV.	<i>Conception</i>	77
1.	Le diagramme de classes	77
2.	Le diagramme de séquences.....	79
V.	<i>Conclusion</i>	80
CHAPITRE 04 :	IMPLEMENTATION.....	81
I.	<i>Introduction</i>	82
II.	<i>Environnement de développement</i>	82
1.	Java	83
2.	Eclipse	84
III.	<i>Quelques captures d'écran de notre application</i>	84
IV.	<i>Conclusion</i>	88
CONCLUSION GENERALE	89
I.	<i>Conclusion Générale</i>	90
REFERENCES	93
REFERENCES	94

Table des figures

Figure 1 : Les étapes du Système à base de OCR.-----	8
Figure 2 : Une image originale et quelques image filtrées par des filtres de Gabor avec différents paramètres. -	11
Figure 3 : Présentation d'une image de document numérisé et sa version affectée par un bruit Gaussien. -----	11
Figure 4 : Une image et sa version traitée par le filtre Wiener. -----	12
Figure 5 : Une image binarisée et sa version après un post-traitement utilisant les filtres Swell et Shrink.-----	13
Figure 6 : Une image de document ainsi que sa version après l'application du filtre d'entropie locale -----	13
Figure 7 : Une image originale et sa version obtenue en utilisant une méthode combinée avec le filtre médian.	14
Figure 8 : Une image de document et sa version binarisée intégrant le filtre bilatéral. -----	14
Figure 9 : Une image de la base DIBCO'2014 et sa version après l'application du filtre Kalman. -----	15
Figure 10 : Binarisée et sa version après application du filtre de dilatation (Swell filter) en post-traitement-----	16
Figure 11 : Image originale et sa version binarisée après ce traitement combiné-----	16
Figure 12 : Image originale et sa version contrastée (voir la version en ligne pour les couleurs) -----	17
Figure 13 : Image originale et son image binarisée utilisant K-Means -----	21
Figure 14 : Quelques images du jeu de données HDIBCO 2013 binarisées en utilisant la méthode Fuzzy C-Means (FCM). -----	22
Figure 15 : Quelques images de HDIBCO'16 binarisées par la méthode de seuillage basé sur le gradient. -----	24
Figure 16 : Quelques images issues de HDIBCO'16 binarisées en utilisant la méthode d'Otsu.-----	25
Figure 17 : Exemples d'images du jeu de données HDIBCO 2016 binarisées à l'aide de la méthode de Niblack. -----	26
Figure 18 : Exemple issu du jeu de données HDIBCO 2012, avec une image documentaire et son résultat de binarisation obtenu à l'aide de la méthode de Sauvola-----	27
Figure 19 : Présente quelques images du jeu de données HDIBCO 2013 binarisées à l'aide de la méthode Deghost. -----	27
Figure 20 : Présente quelques images issues du jeu de données HDIBCO 2016 binarisées à l'aide de la méthode de Niblack.-----	28
Figure 21 : Some images from HDIBCO'16 binarised using Bradley method -----	29
Figure 22 : Quelques images issues du jeu de données HDIBCO 2016 binarisées à l'aide de la méthode de Bernsen. -----	29
Figure 23 : Quelques images issues du jeu de données HDIBCO 2016 binarisées à l'aide d'une méthode adaptative locale.-----	29
Figure 24 : Illustre une image issue du jeu de données DIBCO 2009 ainsi que sa version binarisée obtenue à l'aide de la méthode de Gatos-----	30
Figure 25 : Montre une image originale ainsi que sa version binarisée obtenue à l'aide d'un réseau de neurones entièrement convolutionnel (FCNN). -----	31
Figure 26 : Présente des exemples d'images issues de HDIBCO 2013 binarisées à l'aide de techniques utilisant plusieurs seuils. -----	32
Figure 27 : Illustre une image originale et sa version binarisée obtenue à l'aide de la méthode de Howe. -----	33
Figure 28 : Examples of regional growth -----	35
Figure 29 : Examples of different operator image show -----	37

Figure 30 : Architecture Générale du Système SCAEQCM. -----	59
Figure 31 : Architecture fonctionnelle de notre futur système. -----	60
Figure 32 : Architecture fonctionnelle détaillée. -----	62
Figure 33 : Confection du sujet d'examen en se basant sur un formulaire. -----	66
Figure 34 : Confection du sujet d'examen en se basant sur un formulaire. -----	66
Figure 35 : Un exemple d'une copie d'examen scanné. -----	67
Figure 36 : Un exemple de la binarisation d'une copie d'examen. -----	68
Figure 37 : Un exemple de détection des cases et détection des cases à cocher. -----	72
Figure 38 : Un exemple d'attribution de points où les réponses de l'étudiants sont correctes. -----	73
Figure 39 : Un exemple sur le calcul de la note globale. -----	73
Figure 40 : Exemple de reconnaissance du nom, prénom, et groupe d'une copie d'un étudiant -----	75
Figure 41 : Un exemple d'un fichier EXCEL crée et rempli. -----	76
Figure 42 : Diagramme de classe -----	78
Figure 43 : Diagramme de séquences -----	79
Figure 44 : Les différentes bibliothèques Java utilisées. -----	82
Figure 45 : Notre langage de programmation utilisé. -----	84
Figure 46 : Le logo de l'éditeur dédié au Java, Eclipse. -----	84
Figure 47 : Interface d'accueil de notre prototype. -----	85
Figure 48 : Un formulaire affiché par notre application pour confectionner le sujet. -----	85
Figure 49 : Une interface graphique pour désigner le barème du sujet d'examen. -----	86
Figure 50 : Une interface graphique pour scanner et charger une copie d'examen. -----	87
Figure 51 : Une interface pour binariser une copie d'examen. -----	87
Figure 52 : Interface montrant la détection par le système des cases et des cases cochées. -----	88

Résumé

Au sein de l'université, les examens constituent le moyen standard pour évaluer les étudiants après un semestre d'étude. La correction des examens semble être une tâche dure et fastidieuse surtout pour les promotions de première année tronc-commun où il y a un nombre volumineux des étudiants. Pour cette raison, aller dans le sens des examens à base des QCM, où la correction devient seulement une vérification par rapport au corrigé type, est un raisonnement devant être adopté par les enseignants, de nos jours, pour simplifier la tâche de correction d'une part, et rendre la phase de réponse plus facile aux étudiants, surtout ceux qui sont faibles à la langue d'étude, d'autre part.

L'objectif de ce mémoire de fin d'études est de concevoir et de mettre en œuvre une application informatique permettant de corriger les copies d'examen automatiquement, reconnaître les noms des étudiants sur les copies, et remplir les notes sur la plateforme d'évaluation en générant un fichier en format Excel. Notre application s'inscrit alors dans le cadre de la reconnaissance automatique du manuscrit en reconnaissant les noms des étudiants, leurs groupes associés, ainsi que les réponses correctes.

Nous procédons alors à appliquer les différentes étapes du processus de reconnaissance de la phase de prétraitement qui englobe la binarisation et la segmentation automatique, jusqu'à l'étape de reconnaissance des noms des étudiants, leurs groupes associés, ainsi que les réponses correctes.

Dans le but de simplifier les tâches à effectuer par notre application, on exige un certain modèle de la copie d'examens afin de limiter l'espace où l'étudiant est invité à cocher la bonne réponse.

يُعدّ الامتحان وسيلةً معتمدةً بشكلٍ عام من قبل المجتمع التربوي لتقييم المتعلمين، سواء كانوا تلاميذ أو طلبة. ويُعتبر المتعلمون العنصر الأساسي داخل الوسط المدرسي أو الجامعي الذين يتوجب تقييمهم للتأكد من امتلاكهم لرصيد معرفي وفكري معين. وتمثّل عملية تصحيح الامتحانات مهمة دقيقة وشاقة، خصوصاً بالنسبة للأفواج التي تضم عدداً كبيراً من الطلبة. ولهذا السبب، فإن التفكير في تصحيح الامتحانات بشكل آلي بواسطة تطبيق معلوماتي يُعدّ مشروعاً مفيداً ومهماً، بل أصبح حلاً لا يمكن الاستغناء عنه في ظل التوفر الواسع لتقنيات المعلومات الحديثة في وقتنا الحالي.

في هذا المذكرة الخاصة بنيل شهادة الماستر، نولي اهتماماً بتصميم وإنجاز تطبيق معلوماتي قادر على تصحيح الامتحانات عوضاً عن الأستاذ. ويندرج هذا التطبيق بشكل عام ضمن مجال التعرف على الكتابة اليدوية، مع تضمّنه لبعض الأجزاء الأخرى المرتبطة بإعداد موضوع الامتحان كمرحلة تمهيدية، بالإضافة إلى حساب النقاط النهائية عبر تجميع العلامات الممنوحة لكل سؤال وفق سلم تنقيط محدد.

من الناحية الحاسوبية خارج إطار التعرف، يسمح التطبيق للأستاذ بإدخال موضوع الامتحان وسلم التنقيط عبر واجهة تفاعلية سهلة الاستعمال، ثم توليد موضوع الامتحان وفق نموذج تنسيق معين. وأثناء عملية التصحيح، يقوم التطبيق بمقارنة الإجابات التي اختارها المتعلم مع الإجابات الصحيحة المُحددة من قبل الأستاذ، ومنح نقطة لكل سؤال، ثم جمع النقاط لاحتساب العلامة النهائية.

أما من ناحية التعرف، يقوم التطبيق بعملية تحويل أوراق الإجابة الممسوحة ضوئياً إلى صورة ثنائية اللون (أسود وأبيض) من أجل تحديد المواضيع التي تم اختيارها من قبل المتعلم في كل سؤال.

وتنظيم هذه المذكرة يكون على النحو الآتي: في الجزء الأول الخاص بدراسة الحالة والأعمال السابقة، نستعرض بتفصيل مجال التعرف على الكتابة اليدوية. ثم في الفصل الثاني المتعلق بتحليل وتصميم التطبيق، نعرض المراحل المختلفة التي مكّنتنا من وضع تصور شامل للتطبيق. أما الفصل الأخير، المتعلق بالإنجاز، فنقدم فيه كيفية عمل التطبيق وواجهاته المختلفة. ونختم المذكرة ببعض الآفاق المستقبلية التي نراها جديرة بالاهتمام في إطار أعمالنا القادمة.

Abstract

Within university, exams constitute a standard mean to evaluate students after the end of any semester. The corrections of exams seem to be a hard and a tedious task especially with big promotions where there is a voluminous number of students. Therefore, adopting exams based on QCM, where the correction becomes a simple verification regarding to the correct model, is the alternative to be chosen by professors of nowadays to simplify firstly the correction task from professor view point and making a simple exam to be answered by students, especially who has some problems of the language of study in the other hand.

Our purpose within this Master dissertation is to design and develop an automatic application allowing to correct the exam copies, recognize the names of students, their associated groups, the correct answers given by the students, and setting the marks on the evaluation platform through generating an Excel document. Our application falls then on the purview of optical character recognition field trying to recognize the essential information like: student name, his/her group, as well as detecting the correct answers he/she gives.

We proceed then to apply the different stages of recognition processing ranging from binarization and segmentation to recognition of the essential information.

In the sake to simplify all the tasks of our application, we impose a certain template to be respected by the professor who establish the exam in order to limit the zone where the student is invited to select the correct suggestion.

Introduction Générale

L'examen constitue un moyen généralement adopté par la communauté des enseignants afin d'évaluer leurs apprenants, que ce soient des élèves ou des étudiants. Les apprenants sont alors les acteurs, au sein de l'environnement scolaire ou universitaire, qui nécessitent d'être évalués pour s'assurer d'avoir cumulé un certain bagage intellectuel. La correction des examens représente une tâche délicate surtout pour des promotions ayant un nombre important des étudiants. Pour cette raison, penser à faire la correction automatique des examens via une application informatique semble un projet utile et intéressant voire une alternative qu'on ne peut pas surmonter surtout de nos jours où il y a une disponibilité flagrante des nouvelles technologies de l'information.

Dans ce mémoire de Master, nous nous intéressons à la conception et à la réalisation d'une application informatique pouvant corriger des examens à la place des enseignants. Cette application s'inscrit, en grosso modo, dans le cadre de la reconnaissance du manuscrit avec quelques autres parties liées à la confection ou la génération du sujet, comme une phase de pré-traitement, et la production de la note finale via compter les différents scores obtenus tout au long des différentes questions composant l'examen suivant un certain barème.

Du point de vue calcul en dehors de la reconnaissance, l'application permet à l'enseignant d'introduire son sujet ainsi que son barème via une interface conviviale et générer le sujet respectant un certain modèle de mise en forme. Pendant la correction, l'application compare les réponses sélectionnées par l'apprenant aux réponses vraiment correctes selon l'enseignant, attribuer un score pour chaque question, et compter ces scores pour générer la note finale.

Dans l'optique de la reconnaissance, l'application binarise les copies scannées afin de maintenir seulement deux couleurs (noir et blanc) et localise les suggestions choisies par l'apprenant pour chaque question.

L'organisation du manuscrit est alors comme suit : dans la première partie d'état de l'art, nous présentons en détail le domaine de la reconnaissance du manuscrit. Dans le deuxième chapitre, liée à l'analyse et la conception de l'application, nous mettons en avant les différentes étapes qui nous permettent de concevoir notre application. Dans le dernier chapitre, qui touche à l'implémentation, nous présentons le mode de fonctionnement de notre application ainsi que les différentes interfaces du système. Nous terminons le mémoire avec quelques perspectives qui nous semblent utiles à compléter dans nos travaux futurs.

Partie 1 : Etat de l'art

Chapitre 01 : Reconnaissance Automatique du Manuscrit

I. Introduction

Dans le contexte de l'exploration de notre domaine d'étude ainsi que ses concepts essentiels de base, nous nous intéressons, dans ce premier chapitre d'état de l'art, à la reconnaissance optique du caractère. En effet, nous mettons en avant les différentes définitions des notions de base à l'instar des étapes de prétraitement pour bien préparer l'image de document et minimiser l'influence des problèmes liés à la mauvaise acquisition de ce type d'image que ce soit par un scanner ou une caméra. Par la suite, on aborde le traitement qui consiste à extraire les lignes de texte, les mots, voire les caractères pour entamer l'étape primordiale de reconnaissance et de construction du document en se basant cette fois-ci sur le code ASCII ou l'Unicode. Un post-traitement peut être utile pour bien améliorer les résultats envisagés de ne pas être satisfaisants or la plupart des étapes du processus de reconnaissance dès le prétraitement sont considérées comme des problématiques et des axes de recherche qui nous obligent d'aller à la littérature scientifique pour exploiter tout ce qui existe en termes des méthodes et des techniques.

Ce premier chapitre d'état de l'art intitulé « Reconnaissance automatique du manuscrit » est structuré alors comme suit : on commence, tout d'abords, par la définition de reconnaissance du manuscrit et du caractère optique, en abordant par la suite l'étape de prétraitement qui englobe les différents filtres permettant de corriger, plus ou moins, l'image de document. On parle après sur l'extraction, respectivement, de lignes du texte, de mots, et des caractères en tant qu'un ensemble de pixels en détaillant les différents travaux qui les aborde dans la littérature. Un post-traitement va être alors bien détaillé que ce soit basé sur le feedback de l'utilisateur et son interactivité ou directement sur les résultats en se basant sur d'autres ressources sémantiques externes liées au langage naturel sans aucune interaction avec l'utilisateur.

La reconnaissance optique de caractères (ROC), connue sous son acronyme anglais OCR (Optical Character Recognition), est une technologie permettant de convertir des documents imprimés ou manuscrits en fichiers numériques modifiables. Elle joue un rôle fondamental dans la numérisation de documents, l'archivage électronique, et l'automatisation de la saisie de données. Ce chapitre présente les principes, les techniques et les applications de la ROC, ainsi que ses limites et perspectives.

II. Définitions, Principes de base, et étapes

La reconnaissance optique du caractère (ROC) (En Anglais : Optical Character Recognition) est considérée comme l'un des sous-domaines de recherche les plus importantes dans le domaine large de la reconnaissance de formes (Pattern Recognition) depuis les dernières

décennies. Il est un sujet activement étudié que ce soit en industrie ou en académie à cause de son immense applicatif potentiel [1]. ROC a initialement été étudié aux débuts des années 1930 [2]. Il a eu ses origines en Allemagne comme un brevet d'invention produit par Gustav Tauschek. ROC est une technique de transformation de l'écrit ou du manuscrit que ce soit produit manuellement (Hand-Written) ou tapé par machine et capturé par une caméra (print-written or scanned paper documents), à l'instar des cartes de commerce, à un écrit codé par l'ordinateur sous forme de code ASCII ou UNICODE [3]. ROC, comme un sous-domaine de traitement d'images largement utilisé, est un processus qui consiste à lire les caractères écrits à la main pour les reconnaître. La reconnaissance du caractère est matérialisée comme un enchaînement d'étapes à savoir : la segmentation, l'extraction des caractéristiques et la classification.

La ROC est un procédé alors qui utilise des algorithmes de traitement d'image et d'intelligence artificielle pour identifier et transcrire des caractères présents sur une image. Elle se compose de plusieurs étapes :

- **Acquisition de l'image** : Généralement à l'aide d'un scanner ou d'un appareil photo.
- **Prétraitement** : Correction des défauts (bruit, inclinaison, contraste), et binarisation.
- **Segmentation** : Séparation des lignes, mots et caractères.
- **Reconnaissance des caractères** : Identification des lettres/chiffres à l'aide de modèles prédéfinis ou de réseaux de neurones.
- **Post-traitement** : Vérification orthographique, reconstruction logique du texte.

Comme illustré à la figure 1 [4], un système à base d'OCR est un processus constitué d'un enchaînement d'étapes. La première étape consiste à numériser un document en utilisant un scanner ou une caméra numérique. La deuxième phase qui se nomme « segmentation locale » détermine les constituants de l'image. Elle est nécessaire pour distinguer les régions d'écriture de l'image de tout ce qui est figure ou graphique avant d'entamer la tâche de reconnaissance. Cette métaphore vient de la nécessité de distinguer l'adresse d'une enveloppe, lors de la réception d'une lettre, de tout ce qui est timbre ou logo de la compagnie qui a envoyé la lettre. Dans le cas de l'image de document, « la segmentation locale » consiste à détecter les mots à reconnaître. La plupart des algorithmes OCR catégorise les mots en un ensemble de caractères séparés afin de lancer la reconnaissance individuellement. Généralement, la segmentation est effectuée par la séparation de chaque composant connecté. Cette technique de 'séparation des

composants connectés [5] est connue par sa simplicité d'implémentation mais malheureusement des problèmes peuvent se soulever si un caractère est constitué de plus d'un composant. Les agaçants problèmes qu'on peut affronter lors de la segmentation, dans le cas de l'image de document, sont : l'extraction des caractères fragmentés (qui peuvent s'étaler à plusieurs composants lors de l'adoption de la méthode 'des composants connectés'), l'élimination du bruit pouvant exister dans le texte, ainsi que la séparation entre le texte et le graphique.

La troisième étape dans le processus OCR est la phase de prétraitement. Les données brutes, liées aux données acquises, nécessitent un ensemble de tâches de prétraitement pour éliminer tout un bruit et permettant la bonne utilisabilité descriptive ainsi que l'analyse des caractères. En effet, tout dépend de la résolution du scanner, utilisé dans la phase d'acquisition et le seuillage qui peut être plus tard adopter, les caractères, en mode pixels, peuvent être vus mal-droites ou bruités. Ces problèmes liés aux données, générés pendant l'acquisition, doivent être surmontés et rectifiés, par des opérations de prétraitement à l'instar de l'opération de lissage (Smoothing), pour éviter toute influence négative sur l'opération postérieure de reconnaissance. Le lissage peut être dans les deux sens : la sous-opération de remplissage (Filling) éliminant les éventuelles fausses ruptures, fossés, creux, et coupures pouvant exister et la sous-opération d'amincissement (Thining) qui réduit la fausse largeur des traits. La technique qui est largement utilisée, dans le contexte de lissage, consiste à faire bouger une fenêtre via une image binarisée et appliquer quelques règles sur le contenu de la fenêtre. Le prétraitement inclut aussi l'opération de normalisation, tout au long de lissage, pour obtenir des caractères d'une taille uniforme et renforcer la non-sensibilité vis-à-vis la rotation.

En grosso modo, le prétraitement a pour but : (i) d'éliminer le bruit dans l'image de document brute en utilisant quelques filtres, (ii) la normalisation des données, et (iii) la compression de la quantité d'information à maintenir. En conséquence, le prétraitement donne alors une image de caractères claire contenant une information de forme suffisante, bien compressée, et avec moins de bruit.

L'étape du processus OCR, qui vient ensuite, est la segmentation qui consiste à segmenter et catégoriser les caractères de l'image ainsi leurs différents composants. A noter que toutes les étapes du processus OCR sont d'importance mais l'étape de segmentation avec sa particularité possède une importance spéciale [6]. La segmentation est utilisée comme une tâche de séparation des lignes et de séparation des courbes dans les caractères mal-écrits, qualifiée, unanimement, comme une tâche délicate voire tout un défi. Généralement, il y a trois catégories

de segmentation [7], d'après ce qui existe dans la littérature : (i) la segmentation explicite, (ii) la segmentation implicite, et (iii) la segmentation mixte qui combine les deux catégories précitées.

La cinquième étape, dans le processus OCR, est la représentation, préférablement en utilisant des caractéristiques plus compactes. Pour ce faire, un ensemble d'attributs et de caractéristiques sont extraites pour chaque classe qui aide à distinguer ses caractères des autres caractères des autres classes et réduire les variances à l'intérieur de la même classe [8]. Les méthodes, pour représentation de l'image de document, sont généralement catégorisées en trois groupes : (i) la transformation globale (transformation de Fourier, la transformation de Gabor, la transformation Wavelet, et les moments de Zernike), (ii) la représentation statistique, et (iii) la représentation géométrique et topologique.

La sixième étape, dans le processus OCR, est l'extraction de caractéristiques (Feature Extraction) qui consiste à capturer les caractéristiques essentielles des symboles. Elle est considérée comme l'une des problèmes les plus difficiles, non pas seulement pour la reconnaissance du manuscrit, mais dans le domaine de la reconnaissance des formes tout entier. Selon ce qui existe dans la littérature, il y a trois types de méthodes utilisées pour l'extraction des caractéristiques : (i) les points de distribution, (ii) les transformations, et (iii) l'analyse structurel.

La septième étape, dans le processus de reconnaissance, est l'apprentissage et la reconnaissance (Training and Recognition), largement utilisée dans le domaine de la reconnaissance de formes (Pattern Recognition) en général. Cette étape consiste alors à assigner des nouveaux éléments à des classes prédéfinies préalablement. Il y a généralement quatre approches pour l'apprentissage et la reconnaissance, à savoir: (i) la correspondance à base de modèles (Template Matching), (ii) les techniques statistiques (comme la reconnaissance non-paramétrique, la reconnaissance paramétrique, l'analyse de clustering, les chaînes de Markov cachées, et le raisonnement à base des ensembles flous), (iii) les techniques structurelles (les méthodes à base de grammaire et les méthodes graphiques), et (iiii) les réseaux de neurones artificiels.

La huitième étape, dans le processus de reconnaissance, est l'étape de post-traitement qui a pour but de détecter les erreurs de reconnaissance et de les corriger, voire de raffiner et d'améliorer les résultats.

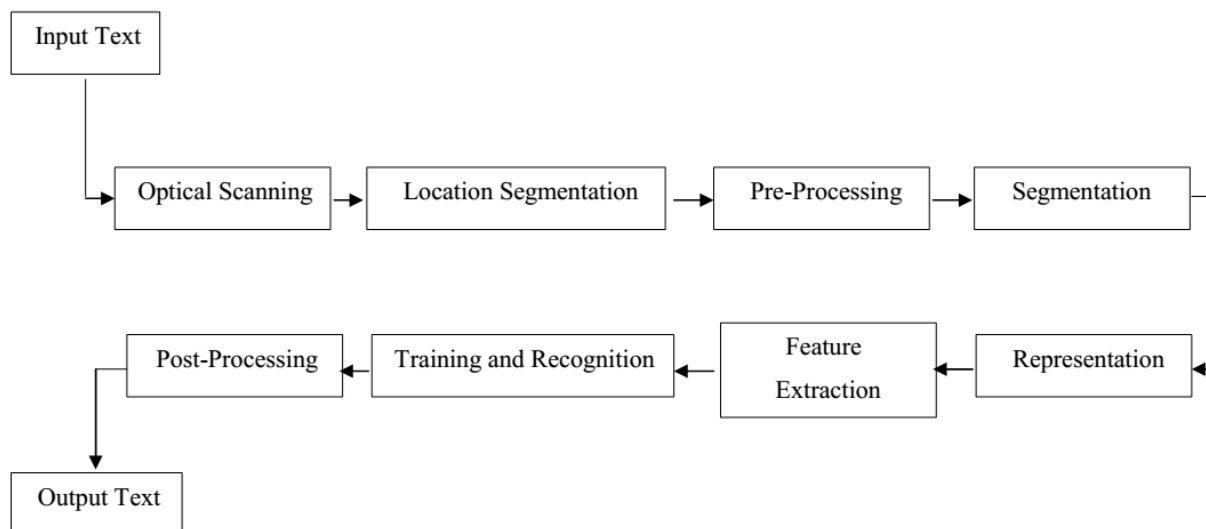


Figure 1 : Les étapes du Système à base de OCR.

III. Prétraitement

Dans section, nous mettons en avant les différentes tâches de prétraitement y compris la correction des défauts via des filtres, la binarisation, et la segmentation.

1. Dégradation et bruits des images de documents

Bhowmik [9] a classé la dégradation et les bruits des images de documents en deux catégories : Les bruits provenant de sources natives, qui apparaissent en raison du vieillissement, de la qualité du papier et de l'encre utilisée.

- Les bruits provenant de sources auxiliaires ou de facteurs externes, comme le mauvais fonctionnement des dispositifs de numérisation, les conditions d'éclairage lors de l'acquisition, ou le mauvais positionnement du document pendant la numérisation.

Les bruits issus des sources natives engendrent des problèmes tels que :

- L'encre qui bave ou traverse la page (ink-bleeding ou bleed through) : Cela se produit lorsque les deux faces de la page sont utilisées pour écrire et que l'encre transperce d'un côté à l'autre.
- Transparence ou bavure (show through ou smear) : Cela survient lorsque l'impression d'encre d'un côté devient visible de l'autre, créant souvent des taches sombres sur la page.
- Texte pâle : Cela apparaît dans les documents écrits ou imprimés avec une encre de mauvaise qualité. Avec le temps, l'encre rétrécit et s'affadit. La qualité du papier peut également être en cause.
- Détérioration du document : Elle résulte du vieillissement, du mauvais stockage, de la mauvaise manipulation, des catastrophes naturelles et d'autres conditions environnementales. Ce type de bruit engendre des taches sombres et d'autres artefacts sur les images de documents.

Les bruits issus des sources auxiliaires provoquent des problèmes tels que :

- Éclairage non uniforme : Qui survient dans les images obtenues par microscopie optique où l'intensité lumineuse diminue fortement selon le chemin parcouru.
- Variation de contraste : Elle apparaît en fonction des conditions environnementales dans lesquelles se déroule l'acquisition de l'image.
- Effet de flou : Observé dans les images de documents sous deux formes :
- Flou de mouvement : Dû au déplacement relatif entre la caméra et l'objet ou à un mouvement brusque de la caméra.
- Flou de mise au point : Lorsque certains points sont nets et d'autres non au moment de l'acquisition.
- Inclinaison (Skew) : Cela se produit soit en raison du mauvais positionnement du document pendant la numérisation, soit en raison du style d'écriture. Cela peut survenir au niveau de la page entière ou d'une région spécifique. En général, dans les documents imprimés, l'inclinaison est constatée au niveau de la page, tandis que dans les documents manuscrits, elle apparaît souvent au niveau de certaines régions.

Plus d'informations sur les problèmes de dégradation et leurs principaux modèles sont fournies par Zhou et al. [10].

2. Quelques filtres utilisés en prétraitement, traitement et post-traitement pour la binarisation

Le prétraitement consiste à préparer l'image du document qui présente de nombreux problèmes tels que : ombres, zones bruitées dues à la numérisation, vieillissement du document, taches d'encre, éclairage non uniforme, infiltrations d'encre, bavures, traînées, transparence, taches d'encre en arrière-plan et décoloration.

Le post-traitement, souvent considéré comme l'étape finale du processus de binarisation, permet d'éliminer les éventuels bruits restants, d'améliorer la qualité des régions de texte, de reconstruire les lignes brisées et de combler les éventuelles ruptures, écarts ou trous.

De nombreux filtres sont utilisés en tant qu'étapes de prétraitement ou de post-traitement pour la binarisation d'images de documents. Cette section présente les filtres essentiels les plus couramment utilisés.

a) Filtre de Gabor

Dans l'étude de Sehad et al. [11], les auteurs ont proposé une méthode simple adaptée à une implémentation logicielle et matérielle basée sur les filtres de Gabor pour la binarisation de documents anciens dégradés. La banque de filtres de Gabor est conçue en tenant compte du type de dégradation du document selon un protocole non-aveugle.

Tout d'abord, l'image du document est prétraitée à l'aide d'un filtre de Wiener pour atténuer la dégradation. Ensuite, le seuil de binarisation est estimé à partir des caractéristiques de texture telles que la moyenne et l'écart-type extraites de l'image originale et de l'image filtrée. De plus, un nouveau protocole, appelé protocole non-aveugle, est proposé pour estimer l'écart-type en fonction du type de dégradation, afin de définir les paramètres optimaux du filtre de Gabor, tels que la fréquence centrale et le nombre d'angles.

La Figure suivante illustre une image originale et quelques images filtrées par des filtres de Gabor avec différents paramètres.

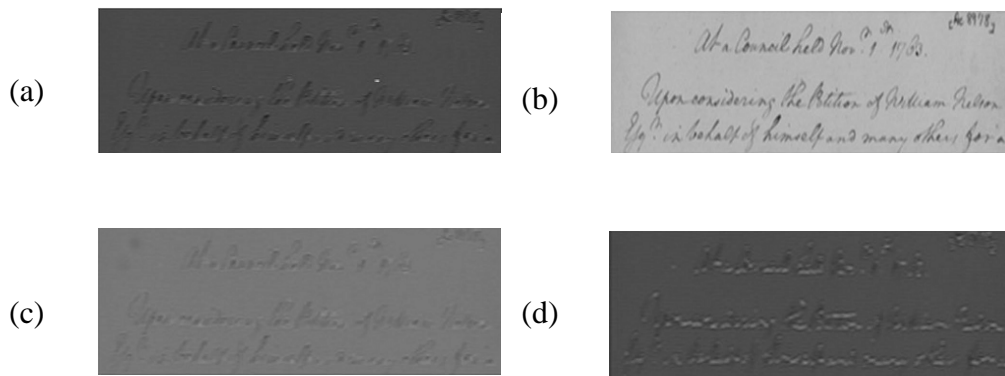


Figure 2 : Une image originale et quelques image filtrées par des filtres de Gabor avec différents paramètres.

b) Distribution Gaussienne Généralisée (GGD)

Dans l'étude de Krupinski et al [12], les auteurs ont confirmé l'utilisation de l'approximation d'histogramme par une distribution gaussienne généralisée (GGD) combinée à la méthode de Monte Carlo pour le prétraitement des images de documents dégradés avant la binarisation et les analyses ultérieures.

En effet, en extrayant les paramètres de la GGD, il est possible de modéliser et de supprimer les distorsions, ce qui améliore la qualité des données en entrée pour les étapes de seuillage et de reconnaissance de texte. Grâce à l'utilisation efficace de la méthode de Monte Carlo, le temps de traitement global a été réduit. L'approche proposée, basée sur la GGD, peut également être combinée avec d'autres algorithmes de binarisation qui n'ont pas été pris en compte dans l'étude de Krupinski et al [13].

La Figure suivante présente une image de document numérisé et sa version affectée par un bruit Gaussien.



Figure 3 : Présentation d'une image de document numérisé et sa version affectée par un bruit Gaussien.

c) Filtre de Wiener

Dans l'étude de Gatos et al. [14], les auteurs ont utilisé un filtre de Wiener adaptatif, basé sur des statistiques locales estimées à partir du voisinage immédiat de chaque pixel, comme étape de prétraitement pour réduire le bruit dans l'image du document.

L'utilisation d'un filtre de Wiener passe-bas (Jain) [15] s'est révélée efficace pour éliminer les zones bruitées, adoucir la texture de l'arrière-plan et améliorer le contraste entre les zones de fond et les zones de texte. Le filtre de Wiener est couramment utilisé en théorie du filtrage pour la restauration d'images.

La figure suivante montre une image et sa version traitée par le filtre de Wiener.

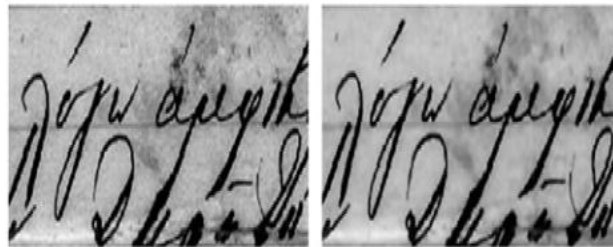


Figure 4 : Une image et sa version traitée par le filtre Wiener.

d) Filtrage Shrink et Swell

Dans les travaux de Gatos et al [16] et Biswas et al [17], les auteurs ont considéré le filtrage Shrink et Swell comme une étape de post-traitement.

Un filtre Shrink est utilisé pour éliminer le bruit présent dans l'arrière-plan en convertissant un pixel d'avant-plan entouré de pixels d'arrière-plan en pixel d'arrière-plan.

Inversement, un filtre Swell est appliqué pour combler les éventuelles ruptures, espaces ou trous dans l'avant-plan, en considérant qu'un pixel d'arrière-plan entouré de pixels d'avant-plan correspond à un trou dans l'objet, qui doit être rempli avec la valeur du pixel d'avant-plan.

La Figure suivante présente une image binarisée ainsi que sa version après un post-traitement utilisant les filtres Swell et Shrink.

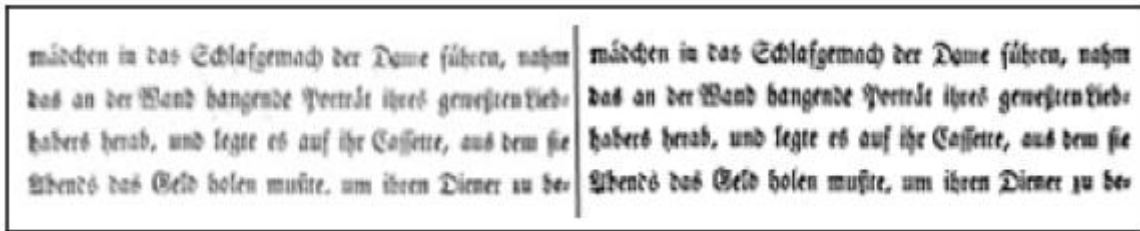


Figure 5 : Une image binarisée et sa version après un post-traitement utilisant les filtres Swell et Shrink.

e) Filtre d'entropie locale

Dans l'étude de Michalak et Okarma [18], les auteurs ont considéré le filtre d'entropie locale comme l'une des étapes de prétraitement pour la binarisation adaptative des images de documents présentant une illumination non uniforme, en vue d'une reconnaissance optique de texte ultérieure.

La figure suivante présente une image de document ainsi que sa version après l'application du filtre d'entropie locale.



Figure 6 : Une image de document ainsi que sa version après l'application du filtre d'entropie locale

f) Filtre médian

Dans le travail de Nafchi et al. [19], les auteurs ont utilisé le filtrage Gaussien et le filtrage médian afin d'améliorer le résultat final de la binarisation.

Le filtre gaussien est utilisé pour renforcer davantage la qualité de la sortie, tandis que le filtre médian est appliqué pour éliminer les bruits résiduels.

La figure suivante montre une image originale et sa version obtenue à l'aide d'une méthode combinée avec le filtre médian.

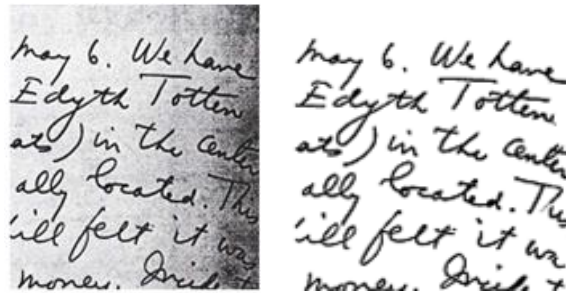


Figure 7 : Une image originale et sa version obtenue en utilisant une méthode combinée avec le filtre médian.

g) Filtre bilatéral

Dans l'étude d'Almeida et al. [20], les auteurs ont utilisé le filtre bilatéral, initialement introduit par Aurich et Weule [21] sous le nom de filtre gaussien non linéaire. Il a ensuite été redécouvert par Tomasi et Manduchi. [22], qui l'ont nommé filtre bilatéral.

Le filtre bilatéral est une technique qui permet de lisser les images tout en préservant les contours. Le poids attribué à chaque pixel voisin diminue à la fois en fonction de la distance spatiale entre les pixels dans le plan de l'image (domaine spatial S) et de la différence d'intensité (domaine de valeurs R).

La figure suivante montre une image de document et sa version binarisée obtenue à l'aide d'une méthode intégrant le filtre bilatéral.

Souhaites-tu que je te fasse un récapitulatif ou une visualisation comparative de tous ces filtres pour une intégration dans ton mémoire ?

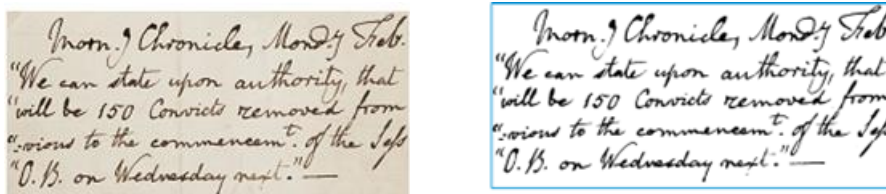


Figure 8 : Une image de document et sa version binarisée intégrant le filtre bilatéral.

h) Filtre de Kalman

Dans l'étude de Rahiche et Cheriet [23], les auteurs ont proposé une approche en deux étapes, appelée KFBin, pour la binarisation des images de documents basée sur la technique de filtrage de Kalman.

Selon Chang et Chang [24], le filtre de Kalman est probablement la technique la plus utilisée en traitement statistique du signal. Il permet de traiter les échantillons de données de manière récursive et très efficace.

Le filtre de Kalman est largement utilisé dans la littérature pour : (1) la réduction de bruit d'image, (2) la reconstruction d'image, (3) la réduction de bruit dans les vidéos, et le suivi d'objets.

Dans les travaux de Rahiche et Cheriet [25], le filtre de Kalman est utilisé pour convertir une image en entrée multi-canaux en deux images de sortie : l'une contenant la composante d'avant-plan améliorée, et l'autre contenant la composante d'arrière-plan.

La Figure suivante montre une image issue du jeu de données DIBCO'2014 et sa version associée après application du filtre de Kalman.

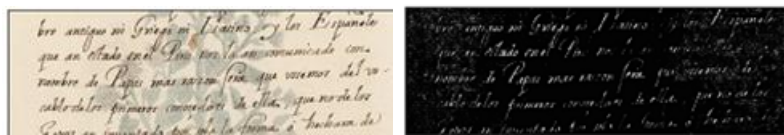


Figure 9 : Une image de la base DIBCO'2014 et sa version après l'application du filtre Kalman.

i) Filtre de dilatation (Swell filter)

Dans l'étude de Chen et Wang [26], les auteurs ont utilisé le filtre de dilatation (swell filter) comme étape de post-traitement, en parcourant l'ensemble de l'image binaire et en examinant chaque pixel blanc.

Malheureusement, et à notre connaissance, certains filtres intéressants tels que le guided filtering (He et al.) [27] et le filtre bayésien (Sarkka et Svensson.) [28] n'ont pas encore été suffisamment explorés pour la binarisation d'images de documents. Le guided filtering, à l'instar du filtre bilatéral, possède la propriété de lisser l'image tout en préservant les contours, avec un comportement encore meilleur au niveau des bords (Bera et al.) [29]. À noter que dans les travaux de Su et al. [30] et Pratikakis et al [31], où de nombreuses méthodes de la littérature

ont été comparées, plusieurs filtres tels que le filtre médian, le filtre gaussien, le guided filter, le filtre bilatéral, et le filtre de Wiener ont été intégrés aux approches mises en compétition. La Figure suivante présente une image binarisée et sa version après application du filtre de dilatation (swell filter) en post-traitement.

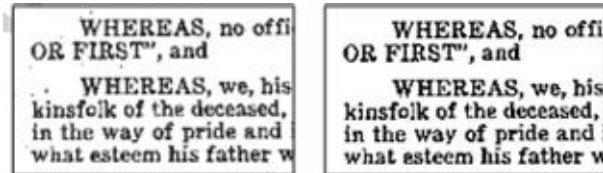


Figure 10 : Binarisée et sa version après application du filtre de dilatation (Swell filter) en post-traitement

j) Transformée Curvelet

Dans les travaux de Wen et al. [32], la transformée Curvelet a été combinée avec la méthode d'Otsu pour la binarisation d'images de documents présentant une illumination non uniforme. L'image affectée par cette illumination inégale est d'abord décomposée à l'aide de la transformée Curvelet, puis ses coefficients sont amplifiés par des fonctions non linéaires. L'image ainsi reconstruite est ensuite soumise à la méthode de binarisation d'Otsu, comme illustré dans la figure suivante, qui présente une image originale et sa version binarisée après ce traitement combiné.

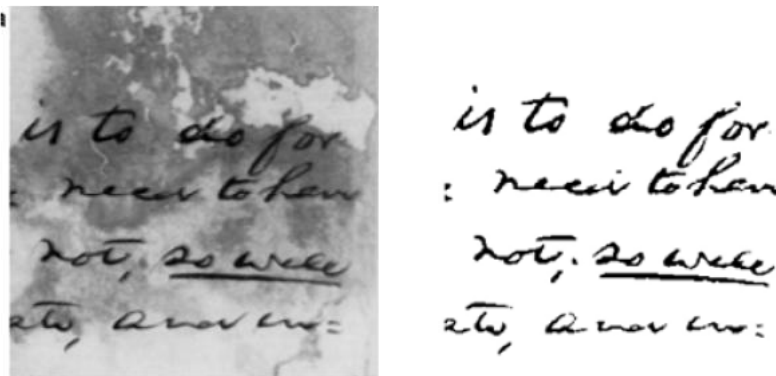


Figure 11 : Image originale et sa version binarisée après ce traitement combiné

k) Filtre d'étirement de contraste

L'étirement de contraste (souvent appelé normalisation), comme illustré dans la Figure suivante, est une technique simple d'amélioration d'image qui vise à améliorer le contraste d'une image en étendant la plage de valeurs d'intensité qu'elle contient afin de couvrir une plage de valeurs souhaitée. Dans Abd Elfattah et al. [33], les auteurs ont envisagé, comme première étape, d'étendre le niveau d'intensité de l'image à l'aide d'un filtre d'étirement de contraste, puis de supprimer le bruit à l'aide d'un algorithme de nettoyage d'image.

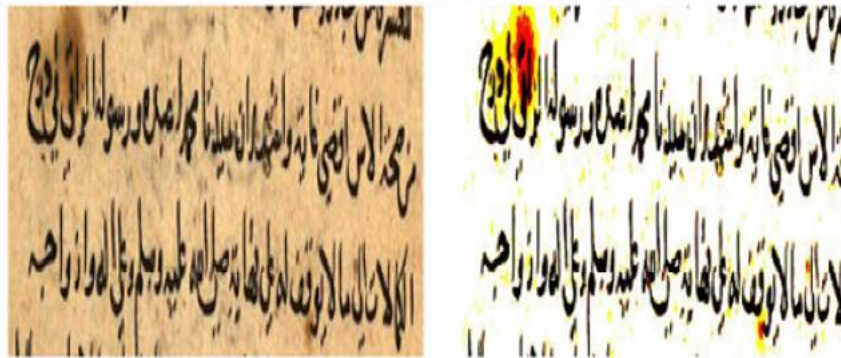


Figure 12 : Image originale et sa version contrastée (voir la version en ligne pour les couleurs)

3. La binarisation

L'une des étapes les plus cruciales du prétraitement des images de documents destinées à la reconnaissance de texte est leur binarisation, car elle influence de manière significative les résultats obtenus via la reconnaissance optique de caractères (OCR) (Krupinski et al.) [34].

Par définition, la binarisation (également appelée seuillage ou séparation avant-plan/arrière-plan) d'une image consiste à attribuer une étiquette à chaque pixel de l'image pour le classer en texte (avant-plan) ou arrière-plan. L'arrière-plan, quant à lui, est défini de façon très vague et peut contenir des informations parasites, voire même des données textuelles.

La motivation des recherches sur la binarisation et la reconnaissance des images de documents ne repose pas uniquement sur la conservation du patrimoine culturel et la mise au jour de faits historiques, mais aussi sur les applications potentielles des algorithmes développés dans d'autres secteurs industriels (Krupinski et al.) [35].

La binarisation d'images de documents est ainsi un domaine actif de la vision par ordinateur, où de nombreux algorithmes ont été proposés dans la littérature. Ces algorithmes peuvent être regroupés en deux grandes catégories :

- Méthodes basées sur le regroupement (clustering) (Moghaddam et Cheriet.) [36]
- Méthodes basées sur le seuillage (thresholding) (Moghaddam et Cheriet.) [37]

Il existe également des méthodes hybrides, telles que celle proposée par Kwon [38]. Les méthodes basées sur le clustering utilisent généralement des caractéristiques issues de modèles pour différencier le texte de l'arrière-plan et les classer (Moghaddam et Cheriet) [39]. La classification floue (Papamarkos) [40] et la segmentation récursive dans l'espace des composantes principales (PCA) (Drira) [41] en sont deux exemples.

Dans la catégorie des méthodes basées sur le seuillage, la plus célèbre et pionnière est la méthode d'Otsu (Chaki et al) [42], qui repose sur la maximisation de la séparation entre deux classes supposées.

Un travail antérieur (Abd Elfattah et al.) [43] a classé les méthodes en six catégories :

- Méthodes basées sur l'histogramme
- Méthodes basées sur le clustering
- Méthodes basées sur l'entropie
- Méthodes basées sur les attributs de l'avant-plan
- Méthodes de binarisation spatiale
- Méthodes adaptatives locales

Il convient également de signaler l'existence de méthodes issues de domaines récents comme l'optimisation. Par exemple, Abd Elfattah *et al.* [44] ont utilisé l'algorithme Artificial Bee Colony (ABC) comme algorithme de clustering pour segmenter le texte des images. Cet algorithme permet de déterminer la meilleure valeur de seuil pour obtenir une image binarisée de qualité selon des critères d'évaluation.

L'analyse de la littérature révèle également des travaux de second plan, sous forme de revues de littérature, études comparatives et surveys sur la binarisation d'images (Mustafa et Kader) [45] ; Chaki et al [46] ; Bhowmik [47]; Mustafa et al. [48]; Chauhan et al., [49]; Sulaiman et al. [50] ; Tensmeyer et Martinez) [51].

Par exemple :

Dans Mustafa et Kader [52], les auteurs ont proposé une revue approfondie, discuté et testé sept méthodes de binarisation sur le jeu de données H-DIBCO 2012. Ils ont formulé des critiques sur les techniques existantes, suggérant par exemple de considérer des méthodes traditionnelles d'amélioration du contraste en post-traitement.

Dans Bhowmik. [53], l'auteur a classé les approches de binarisation en trois grands groupes :

- Approche basée sur le seuillage
- Approche basée sur l'optimisation
- Approche basée sur la classification

Bien que Chaki et al [54] n'aient pas proposé de méthode d'évaluation, ils ont introduit l'idée importante de l'évaluation indirecte, suggérant que la qualité d'une méthode de binarisation peut se refléter ultérieurement dans l'efficacité des applications de reconnaissance utilisant l'image binarisée en entrée.

En s'écartant des catégorisations classiques, Chaki et al [55] ont divisé les méthodes de binarisation en fonction de leur apparition chronologique en deux catégories :

- Méthodes fondatrices
- Travaux récents

De leur côté, Mustafa et al. [56] ont comparé plusieurs techniques de binarisation d'images en s'appuyant sur le jeu de données H-DIBCO, et Chauhan et al. [57] ont mené une étude comparative sur des techniques de binarisation d'images de documents en utilisant les jeux de données DIBCO 2009 et DIBCO 2010.

Sulaiman et al [58] ont classé les techniques en quatre catégories :

- Méthodes de binarisation par seuillage global
- Méthodes de binarisation par seuillage local/adaptatif
- Méthodes de binarisation hybrides
- Méthodes basées sur l'apprentissage automatique

Dans Tensmeyer et Martinez [59], les auteurs ont proposé une revue exhaustive du domaine de la binarisation d'images historiques, présentant un tableau récapitulatif des techniques développées au cours de la dernière décennie, parmi lesquelles :

- Techniques basées sur le seuillage global
- Techniques basées sur le seuillage local
- Techniques basées sur les contours
- Techniques basées sur la transformation d'images
- Techniques basées sur des modèles mixtes
- Techniques basées sur les champs aléatoires conditionnels
- Techniques basées sur la théorie des jeux
- Techniques d'apprentissage automatique peu profond (shallow learning)
- Techniques d'apprentissage profond
- Techniques d'ajustement supervisé
- Techniques d'ajustement non supervisé

Approches principales pour la binarisation d'images de documents :

a) Binarisation d'Image Basée sur le Clustering

Le clustering et la classification sont deux tâches fondamentales de l'exploration de données. La classification est principalement utilisée dans l'apprentissage supervisé, tandis que le clustering relève de l'apprentissage non supervisé. Certains travaux de la littérature ont employé des algorithmes de clustering (classification non supervisée) pour la binarisation d'images de documents.

b) Binarisation Basée sur K-means

Certains travaux ont appliqué l'algorithme K-Means pour la binarisation d'images de documents (Jana et al.) [60]. En effet, dans Jana et al [61], les auteurs ont proposé une technique de clustering basée sur K-Means pour la binarisation adaptative d'images de documents dégradés.



Figure 13 : Image originale et son image binarisée utilisant K-Means

c) Binarisation Basée sur la Classification Floue

De nombreux travaux ont utilisé la méthode des Fuzzy C-means (FCM) dans la binarisation d'images de documents, comme dans les études de Papamarkos [62], Mustafa et al [63], Jana et al [64], Mondal et al [65] et Tong et al. [66].

La classification floue est généralement combinée aux méthodes de seuillage local afin d'en accélérer l'exécution.

Ainsi, dans Papamarkos [67], les auteurs ont proposé une méthode de binarisation floue des documents numériques, adaptée à la binarisation de documents flous ou mal éclairés, et pouvant être facilement modifiée pour s'adapter à tout type de caractéristiques spatiales.

Après apprentissage, les neurones de la couche de compétition de la carte autoorganisée (SOFM) définissent deux classes binaires. À partir du contenu de ces classes, des fonctions d'appartenance floue sont obtenues et ensuite utilisées avec l'algorithme FCM afin de réduire le problème de flou des caractères.

Dans Mustafa et al [68], la méthode proposée se compose de trois étapes :

(1) Une opération de prétraitement de l'image avant la binarisation, pour améliorer la qualité de l'image en appliquant à la fois un étirement du contraste et un filtre moyenneur. (2) Application de l'algorithme de binarisation sur l'image prétraitée. En appliquant l'algorithme FCM, les images sont converties en images binaires et divisées en deux composantes : Le texte et l'arrière-plan. (3) Enfin, une opération de suppression des artefacts fantômes (deghost) est réalisée pour éliminer les entités résiduelles pouvant apparaître sur l'image du document.

Dans Jana et al [69], les auteurs ont présenté une technique de binarisation rapide, performante et simple, reposant sur un seuillage global basé sur FCM et assistée par séparation de l'arrière-plan. La méthode proposée utilise un superset des régions de l'avant-plan pour évaluer correctement l'arrière-plan du document. L'arrière-plan est estimé à l'aide d'une fenêtre d'interpolation glissante de dimension variable, déterminée selon la nature du tracé des

caractères. Finalement, une approche globale est adoptée pour binariser l'image normalisée et améliorée, après séparation de l'arrière-plan, en regroupant les pixels via FCM.

Dans Mondal et al. [70], les auteurs ont proposé une méthode de binarisation locale par seuillage utilisant un clustering FCM rapide. Les images de documents historiques avec un arrière-plan non uniforme, des taches et de l'encre effacée sont d'abord traitées en supprimant l'arrière-plan à l'aide d'une méthode fondée sur l'inpainting. Ensuite, le clustering FCM est utilisé pour regrouper les pixels en trois clusters principaux : (1) Pixels de texte sûrs (2) Pixels d'arrière-plan sûrs (3) Pixels incertains, pouvant ou non être étiquetés comme texte.

Enfin, dans Tong et al [71], les auteurs ont proposé un algorithme de binarisation baptisé NFCM, combinant les algorithmes de Niblack et de FCM. Ils ont affirmé que leur méthode permettait non seulement de préserver les tracés des caractères, mais aussi de réduire les artefacts fantômes.

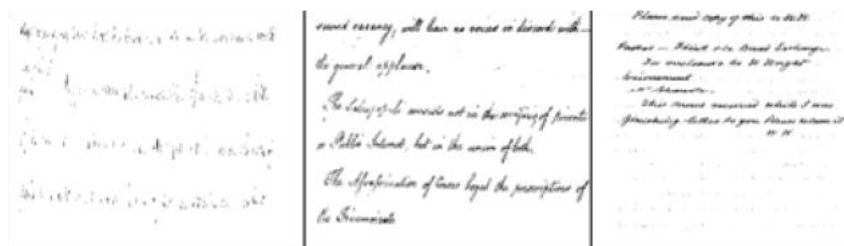


Figure 14 : Quelques images du jeu de données HDIBCO 2013 binarisées en utilisant la méthode Fuzzy C-Means (FCM).

d) Binarisation d'Image Basée sur le Seuillage

La binarisation d'image consiste généralement à convertir une image en couleur ou en niveaux de gris en une image binaire, où les pixels correspondant au texte et à l'arrière-plan sont respectivement marqués en noir et blanc (Jia et al) [72]. Un résultat binaire réussi doit préserver les informations significatives tout en éliminant les informations parasites (Wu et al) [73].

Le seuillage est une technique simple mais puissante pour séparer l'objet d'intérêt de l'arrière-plan (Yazid et Arof) [74]. Le seuillage d'image est largement utilisé dans de nombreuses applications de vision par ordinateur et de traitement graphique, notamment dans l'analyse des images de documents. Lorsqu'une image contient des objets nettement plus clairs (ou plus foncés) que l'arrière-plan, il est naturel de les séparer en définissant une valeur de seuil (Yazid et Arof) [75].

L'algorithme de binarisation basé sur le seuillage est simple : étant donné une valeur de seuil global fixe T (allant de 0 à 255), si la valeur d'intensité d'un pixel de l'image en entrée est supérieure à T , ce pixel est défini comme blanc, sinon il est défini comme noir. Si la source est une image en couleur, elle doit d'abord être convertie en niveaux de gris.

Malheureusement, une valeur de seuil global fixe peut fortement dégrader la qualité de l'image, notamment en raison de la variation de luminosité entre différentes régions de l'image, ce qui conduit à de nombreux pixels mal classés. Cette situation a conduit les chercheurs à envisager soit un seuil dynamique (Bernsen), soit un seuil adaptatif comme proposé dans Tensmeyer et Martinez [76], ou encore l'utilisation de plusieurs seuils locaux pour une même image (Jia et al [77] ; Kim [78]).

En raison de l'importance du seuillage (Sezgin et Sankur) [79], les méthodes de binarisation sont classées selon les critères principaux pris en compte pour calculer ce seuil (Chaki et al) [80]. La principale problématique des méthodes de binarisation par seuillage est alors de savoir comment désigner la valeur de seuil optimale.

L'analyse de la littérature révèle de nombreux travaux visant à sélectionner le seuil de binarisation optimal pour toute image en général, et en particulier pour les images de documents destinées à la reconnaissance optique de caractères (OCR). Parmi ces méthodes, on trouve :

- Les méthodes basées sur l'analyse de clustering (Otsu) [81]
- Les méthodes basées sur l'entropie (Johannsen ; Kapur et al) [82] [83]
- Les méthodes basées sur la variance de l'image (Sauvola et Pietikäinen; Niblack,) [84] [85]
- Les méthodes basées sur le contraste de l'image (Wu et al) [86]
- Les méthodes basées sur la symétrie structurelle des traits (Drira) [87]
- Les méthodes fondées sur les caractéristiques de texture (Liu et Srihari) [88]

Dans He et al. [89], les auteurs ont comparé certaines de ces méthodes de binarisation sur des images d'archives historiques.

À noter que pour les images présentant une illumination non uniforme ou de mauvaise qualité, le seuillage adaptatif est indispensable pour séparer efficacement les objets d'intérêt de l'arrière-plan (Yazid et Arof; Bloechle et al). [90] [91]

- Seuillage Global

Dans l'approche de seuillage global, une seule valeur de seuil est déterminée et appliquée à l'ensemble de l'image afin de séparer les pixels en avant-plan et arrière-plan. Dans les sous-sections suivantes, nous allons présenter les principales méthodes de binarisation utilisant le seuillage global.

- Seuillage Basé sur le Gradient

Dans Yazid et Arof [92], une nouvelle approche a été proposée pour créer une surface de seuil adaptatif afin de segmenter une image. Cette technique s'inspire de la méthode de Yanowitz et Bruckstein [93] et l'améliore en introduisant une surface de seuil plus simple et plus précise. La méthode repose sur un seuillage basé sur le gradient en construisant une surface de seuil. Elle comprend trois phases essentielles :

- Construire l'image inverse $T(i, j)$
- Obtenir la valeur k comprise entre -255 et 255
- Appliquer la binarisation pour séparer les objets de l'arrière-plan

$$result(i, j) = \begin{cases} 0, & (object) & \text{if } I(i, j) < T(i, j) + k_0 \\ 255, & (background) & \text{if } I(i, j) > T(i, j) + k_0 \end{cases} \quad (1)$$

Où $I(i, j)$ représente l'intensité du pixel à la position (i, j) dans l'image originale et k_0 est la somme minimale des différences absolues d'intensité.

La figure suivante présente quelques images issues du jeu de données HDIBCO 2016, binarisées à l'aide de la méthode de seuillage basée sur le gradient.

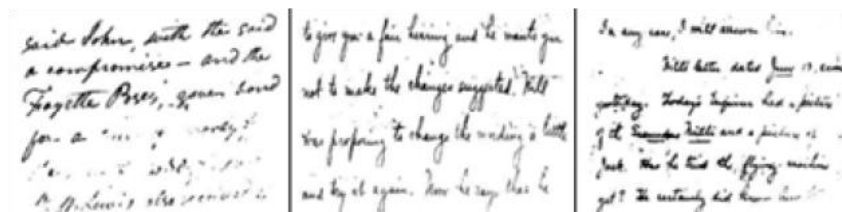


Figure 15 : Quelques images de HDIBCO'16 binarisées par la méthode de seuillage basé sur le gradient.

- Méthode d'Otsu

Dans l'étude de Otsu [94], une méthode très simple a été introduite, permettant de déterminer un seuil optimal et offrant une extension directe au problème de multi-seuillage.

Le seuil est obtenu automatiquement en se basant à la fois sur la variance globale et la variance inter-classes. Dans le cas d'une image à illumination non uniforme, Otsu part du principe que l'image contient deux zones distinctes :

- Une zone sombre
- Une zone claire

Afin de structurer son algorithme final. Le seuillage selon Otsu est alors défini par :

(La formule suit ensuite dans le texte original).

$$k = \frac{\sigma^2 B}{\sigma^2 G} \quad (2)$$

where K is a threshold value, $\sigma^2 B$ is a global variance of the entire image, and $\sigma^2 G$ is a between-class variance.

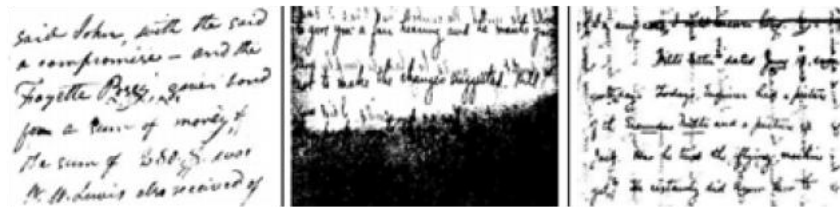


Figure 16 : Quelques images issues de HDIBCO'16 binarisées en utilisant la méthode d'Otsu.

- Seuillage Local

Malheureusement, le seuillage global n'est pas suffisant dans certaines situations (Sauvola et Pietikäinen) [95], notamment en raison de changements d'éclairage, d'erreurs de numérisation, de problèmes de résolution, de la mauvaise qualité du document source ou de la complexité de sa structure.

Dans le cas du seuillage local ou adaptatif, le même seuil n'est jamais utilisé pour toute l'image. Au contraire, ce sont les caractéristiques d'un pixel et de ses voisins dans une sous-région de l'image qui servent à déterminer le seuil local.

Dans les sous-sections suivantes, nous présentons quelques méthodes de binarisation basées sur le seuillage local.

- Méthode de Niblack

L'objectif principal de la méthode de Niblack [96] est de fixer la valeur de seuil en se basant sur l'écart-type local et la moyenne locale. Le seuil pour chaque pixel est déterminé selon l'équation suivante :

$$T(x, y) = m(x, y) + k\delta(x, y) \quad (3)$$

Où : $m(x,y)$ est la moyenne locale, $\delta(x,y)$ est l'écart-type local,

La taille de la fenêtre utilisée pour ce calcul est de 80×80 pixels, k est entre $(-0,2)$.

Il est important de noter que cette méthode ne fonctionne pas correctement si l'image présente une illumination non uniforme.

La figure suivante présente quelques images issues du jeu de données HDIBCO 2016 binarisées en utilisant la méthode de Niblack.

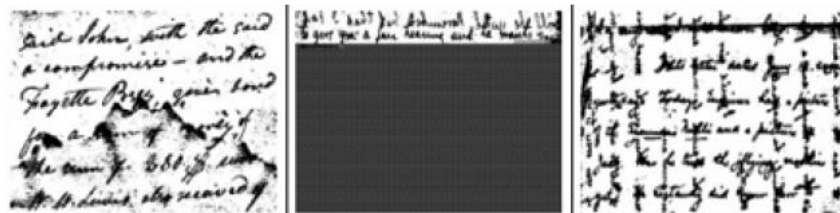


Figure 17 : Exemples d'images du jeu de données HDIBCO 2016 binarisées à l'aide de la méthode de Niblack.

- Méthode de Sauvola

Dans leur publication de 2000, Sauvola et Pietikäinen ont proposé une nouvelle méthode de binarisation d'images de documents reposant sur une approche adaptative, capable de gérer différentes situations au sein d'une même image. La technique de Sauvola réalise une analyse rapide de la surface de l'image afin de sélectionner et d'ajuster dynamiquement l'algorithme en fonction du contenu du document. Ce contenu est utilisé pour déterminer le type d'algorithme à appliquer, la nécessité d'ajuster certains paramètres et pour calculer une valeur de seuillage propre à chaque pixel ou à un intervalle régulier de pixels (selon une approche interpolative). Ainsi, le contenu du document pilote directement le processus de binarisation.

La figure suivante présente un exemple issu du jeu de données HDIBCO 2012, avec une image documentaire et son résultat de binarisation obtenu à l'aide de la méthode de Sauvola.

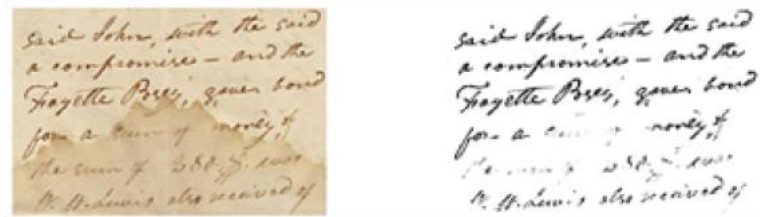


Figure 18 : Exemple issu du jeu de données HDIBCO 2012, avec une image documentaire et son résultat de binarisation obtenu à l'aide de la méthode de Sauvola

- Méthode Deghost

Dans leur étude de 2018, Li et al. ont proposé une règle de taux d'apprentissage combinant une détection adaptative des contours par seuillage et une porte temporelle. Grâce à un algorithme d'estimation du bruit, cette méthode détermine un seuil spatial adaptatif en fonction du bruit résiduel de non-uniformité présent dans l'image corrigée. Le taux d'apprentissage proposé permet de supprimer efficacement et de manière stable les artéfacts de type « ghosting » sans ralentir la vitesse de convergence du processus.

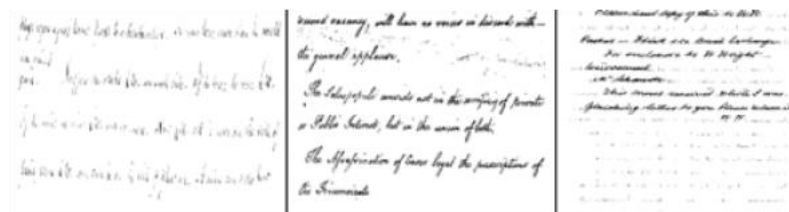


Figure 19 : Présente quelques images du jeu de données HDIBCO 2013 binarisées à l'aide de la méthode Deghost.

- Méthode de Nick

Dans le travail de Khurshid et al [97], une technique de seuillage local basée sur une fenêtre glissante a été proposée, dans le but d'améliorer la méthode de Niblack en abaissant légèrement le seuil de binarisation. La valeur du seuil est déterminée à l'aide de l'équation suivante :

$$T(x, y) = m + k\sqrt{\frac{(I^2 - m^2)}{N}} \quad (4)$$

Où le facteur k fonctionne de manière similaire à celui utilisé dans la méthode de Niblack, et où la taille de la fenêtre est fixée à 15×15 pixels. Dans cette équation, I représente l'intensité d'un pixel donné, m correspond à la moyenne locale calculée dans la fenêtre, et N désigne la taille totale de l'image.

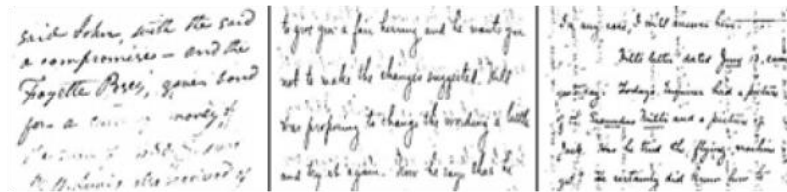


Figure 20 : Présente quelques images issues du jeu de données HDIBCO 2016 binarisées à l'aide de la méthode de Niblack.

- Méthode Bradley

Dans leur étude de 2007, Bradley et Roth ont proposé une amélioration et une extension de la méthode de Wellner (Pierre) [98], conçue pour être plus robuste face aux variations d'illumination au sein de l'image. Dans l'algorithme de Wellner, chaque pixel est comparé à la moyenne d'intensité de ses pixels voisins. La méthode de Bradley repose sur le principe suivant : un pixel est défini comme noir si sa luminance est inférieure d'au moins T pour cent à la moyenne des intensités des pixels dans une fenêtre locale de taille spécifiée ; sinon, il est défini comme blanc. Par défaut, la taille de la fenêtre est de 15×15 pixels et la valeur de T est fixée à 10. Le seuil T est calculé selon l'expression suivante :

$$T = m \left(1 - \frac{k}{100} \right) \quad (5)$$

La figure suivante présente plusieurs images issues du jeu de données HDIBCO 2016 binarisées à l'aide de la méthode de Bradley.

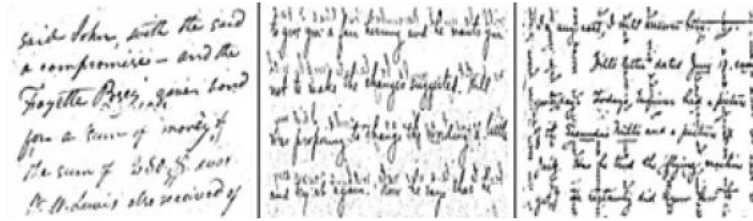


Figure 21 : Some images from HDIBCO'16 binarised using Bradley method

- Méthode de Bernsen

L'algorithme de Bernsen, proposé par Bernsen [99], est une technique de seuillage local reposant sur l'estimation d'une valeur de seuil locale pour chaque pixel. Cette valeur de seuil est appliquée uniquement si la différence entre les niveaux de gris maximum et minimum au sein d'une fenêtre locale dépasse un seuil prédéfini k . Dans le cas contraire, il est supposé que la région de la fenêtre appartient à une seule classe — soit le fond, soit l'avant-plan. Par défaut, la taille de la fenêtre (w) est de 3×3 pixels et la valeur de k est fixée à 15. L'équation finale de seuillage s'exprime comme suit :

$$T(x, y) = \frac{Z_{\max} + Z_{\min}}{2} \quad (6)$$

Où Z_{\min} et Z_{\max} désignent respectivement les valeurs minimale et maximale des niveaux de gris dans la fenêtre locale.

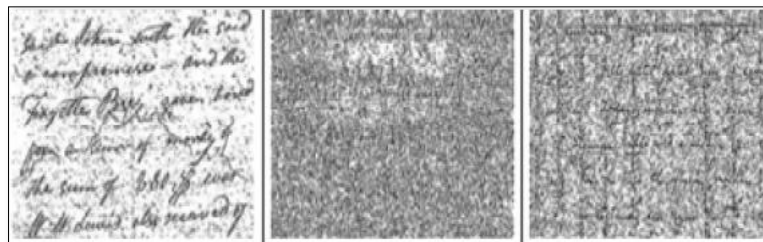


Figure 22 : Quelques images issues du jeu de données HDIBCO 2016 binarisées à l'aide de la méthode de Bernsen.

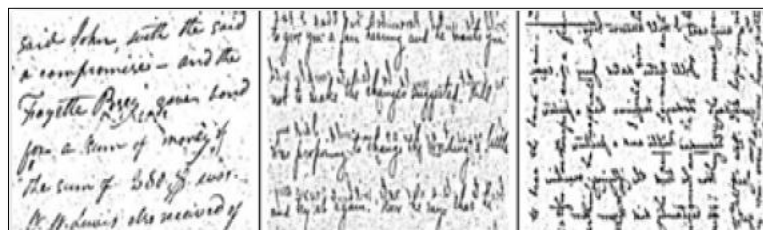


Figure 23 : Quelques images issues du jeu de données HDIBCO 2016 binarisées à l'aide d'une méthode adaptative locale.

- Seuillage adaptatif local

Le seuillage adaptatif local, tel que décrit par Woods et Gonzalez [100], est un algorithme simple permettant de séparer l'avant-plan du fond dans des images présentant des variations d'illumination. Pour chaque pixel de l'image, un seuil local est calculé en fonction des intensités des pixels voisins. Si la valeur d'un pixel est inférieure à ce seuil, il est classé comme appartenant au fond ; sinon, il est considéré comme faisant partie de l'avant-plan. Par défaut, la taille de la fenêtre (w) est de 15×15 pixels et le paramètre de seuil local (T) est fixé à 0,05. Le seuil est déterminé selon l'équation suivante :

$$T = \frac{\max + \min}{2} \quad (7)$$

- Méthode de Gatos

Dans leur étude de 2006, Gatos et al, ont proposé une approche adaptative pour la binarisation et l'amélioration des images de documents dégradés. Les auteurs affirment que cette méthode ne nécessite aucun réglage manuel des paramètres par l'utilisateur et qu'elle peut gérer différents types de dégradations documentaires. L'approche se compose de trois étapes essentielles : prétraitement, traitement et post-traitement.

Le prétraitement a pour objectif d'éliminer les zones bruitées, de lisser la texture de l'arrière-plan et d'améliorer le contraste entre les zones de texte et le fond. Pour cela, un filtre adaptatif de Wiener est utilisé. Ensuite, une segmentation initiale est réalisée pour distinguer les régions d'avant-plan et d'arrière-plan, accompagnée d'une estimation de la surface de l'arrière-plan.

Un seuillage final est ensuite appliqué en combinant la surface d'arrière-plan estimée avec l'image prétraitée. Enfin, une phase de post-traitement est mise en œuvre afin de supprimer les bruits résiduels, d'améliorer la qualité des régions de texte et de préserver la continuité des traits en éliminant les pixels isolés et en comblant les éventuelles ruptures, espaces ou lacunes dans les caractères.



Figure 24 : Illustre une image issue du jeu de données DIBCO 2009 ainsi que sa version binarisée obtenue à l'aide de la méthode de Gatos

e) Binarisation basée sur les réseaux de neurones

Plusieurs travaux, tels que ceux de Tensmeyer et Martinez [101], Kefali et al [102], Sari et al [103] et Akbari et al [104], ont exploré l'utilisation des réseaux de neurones en tant qu'outil d'intelligence artificielle pour la binarisation des images de documents.

Dans l'étude de Tensmeyer et Martinez [105], le problème de la binarisation est formulé comme une tâche d'apprentissage de classification pixel par pixel, prenant en compte différentes résolutions d'image, y compris l'image en pleine résolution. Pour cela, une architecture de réseau entièrement convolutionnel (FCN) a été mise en œuvre.

Kefali et al [106] ont proposé une méthode de séparation avant-plan/arrière-plan fondée sur des réseaux de neurones artificiels (ANN), appliquée à des documents anciens présentant diverses dégradations. L'approche consiste à entraîner un ANN sur des paires d'images dégradées et leurs versions binaires idéales, en exploitant à la fois des informations globales et locales de l'image.

Dans l'étude de Sari et al [107], les auteurs ont utilisé un réseau de neurones à rétropropagation pour classifier directement les pixels en fonction des caractéristiques de leur voisinage.

Enfin, Akbari et al [108] ont proposé un cadre basé sur des réseaux de neurones convolutifs (CNN) pour identifier les pixels de l'avant-plan à l'aide d'images multicanaux innovantes générées à partir de l'image source. Cette méthode décompose l'image d'origine en plusieurs sous-bandes issues de la transformation en ondelettes, puis reconstitue l'image à partir de chaque sous-bande. L'image multicanal est alors constituée en combinant l'image d'origine comme premier canal et les approximations par sous-bandes comme canaux supplémentaires. Deux configurations sont considérées : des images à deux canaux et à quatre canaux, traitées par deux types d'architectures CNN : à flux unique et à flux multiples. Les modèles CNN utilisés sont les réseaux populaires U-Net, SegNet et DeepLabv3+.

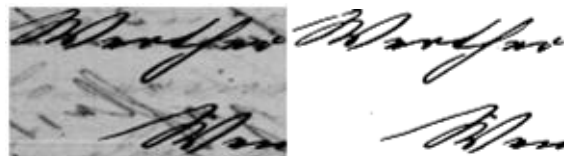


Figure 25 : Montre une image originale ainsi que sa version binarisée obtenue à l'aide d'un réseau de neurones entièrement convolutionnel (FCNN).

f) Combinaison des techniques de binarisation

Certains travaux de la littérature ont choisi de combiner des méthodes de binarisation existantes plutôt que de proposer de nouvelles techniques (Badekas et Papamarkos, Sari et al ; Yoon et al, Lelore et Bouchara) [109][110][111].

Dans Badekas et Papamarkos, les auteurs ont développé un système exploitant les avantages de plusieurs techniques de binarisation sélectionnées en combinant leurs résultats à l'aide d'un réseau de neurones autoorganisé de Kohonen. Ce système a été conçu pour classifier efficacement les pixels présentant une forte incertitude, notamment ceux situés au niveau des contours, dans les zones d'ombre ou d'autres régions ambiguës, difficiles à distinguer clairement entre avant-plan et arrière-plan.

Sari et al [112] ont proposé une technique de seuillage hybride combinant les avantages des méthodes globales et locales. Le processus s'effectue en deux étapes : dans un premier temps, un seuillage global est appliqué à l'image entière, permettant de classifier la majorité des pixels à l'aide de deux seuils déterminés. Dans un second temps, les pixels restants sont classifiés à l'aide d'une analyse locale. Plusieurs méthodes de seuillage local sont combinées et la valeur binaire finale de chaque pixel restant est déterminée en fonction de la classification la plus probable.

Enfin, Lelore et Bouchara [113] ont proposé l'algorithme FAIR, destiné à la restauration d'images de documents. Les auteurs affirment que cet algorithme est invariant à l'échelle et suffisamment rapide pour être utilisé dans des applications en temps réel. Il repose sur une technique de détection de contours à double seuil associée à un regroupement en clusters, permettant de détecter de petits détails tout en conservant une robustesse face au bruit.

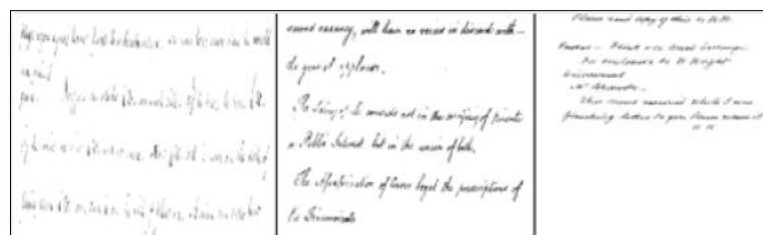


Figure 26 : Présente des exemples d'images issues de HDIBCO 2013 binarisées à l'aide de techniques utilisant plusieurs seuils.

Il convient de souligner que certaines approches dépassent le cadre de la classification précédemment présentée. C'est notamment le cas de la méthode proposée par Howe [114], qui repose sur la minimisation d'une fonction d'énergie globale inspirée d'un modèle de champ aléatoire de Markov (MRF). Cette technique vise à obtenir une binarisation optimale en prenant en compte à la fois les informations locales et globales de l'image via une stratégie de minimisation d'énergie.

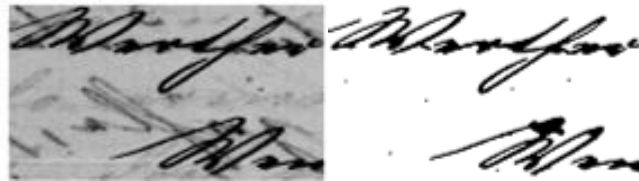


Figure 27 : Illustre une image originale et sa version binarisée obtenue à l'aide de la méthode de Howe.

4. La segmentation

La segmentation d'image constitue une tâche essentielle dans le processus de reconnaissance d'images. Dans les sous-sections suivantes, nous abordons différentes catégories de segmentation que l'on trouve dans la littérature scientifique.

a) Segmentation par seuillage

La segmentation par seuillage est la méthode la plus simple de segmentation d'image, et également l'une des plus courantes pour les applications parallèles. Elle consiste à segmenter directement une image en fonction de ses niveaux de gris.

On distingue deux approches principales :

- Seuillage global : l'image est divisée en deux zones (avant-plan et arrière-plan) à l'aide d'un seuil unique [115].
- Seuillage local (ou adaptatif) : plusieurs seuils sont appliqués localement afin de mieux traiter les variations d'éclairage ou de contraste [116].

L'algorithme le plus utilisé est la méthode d'Otsu : elle choisit automatiquement un seuil en maximisant la variance inter-classes (ou, de manière équivalente, en minimisant la variance

intra-classes) [117]. D'autres approches de seuillage comprennent des méthodes basées sur l'entropie, l'erreur minimale, la matrice de cooccurrence, la préservation des moments, des méthodes statistiques simples, la relaxation probabiliste, les ensembles flous ou des approches hybrides combinant plusieurs approches [118].

Avantages : Simplicité, rapidité de calcul, efficacité lorsque l'avant-plan et l'arrière-plan présentent un fort contraste. Inconvénients : résultats approximatifs lorsque les niveaux de gris se chevauchent ou en présence de bruit et d'irrégularités d'éclairage, car ce type de seuillage ne prend en compte que l'information d'intensité au niveau des pixels [119].

b) Segmentation par croissance de régions

La croissance de régions est un algorithme classique de segmentation d'image reposant sur les régions. Son principe de base consiste à regrouper des pixels présentant des caractéristiques similaires afin de constituer des régions cohérentes [120] :

- On commence par sélectionner un ou plusieurs pixels dits "graine" (seed).
- On ajoute ensuite, de manière itérative, les pixels voisins dont la différence d'intensité (valeur de gris) avec la graine est inférieure à un certain seuil TTT.
- Ce processus se poursuit jusqu'à ce qu'aucun pixel voisin ne satisfasse le critère, puis il est éventuellement relancé avec une nouvelle graine [121].
- Considérons un exemple schématique dans la figure 28 :
- Figure (28a) : Deux graines sont posées (surlignées par des carrés gris).
- Figure (28b) : Avec un seuil faible $T=3T = 3T=3$, deux régions distinctes et bien formées apparaissent.
- Figure (28c) : Avec un seuil plus élevé $T=6T = 6T=6$, la croissance fusionne les régions en une seule, montrant l'impact crucial du choix de TTT [122].

Ainsi, la croissance de régions offre une segmentation adaptée aux zones homogènes en niveaux de gris, mais sa performance dépend fortement du choix du seuil TTT et des graines sélectionnées. Veux-tu que je continue avec les explications sur la segmentation par contours (opérateurs Sobel, Laplacien) de la même façon ?

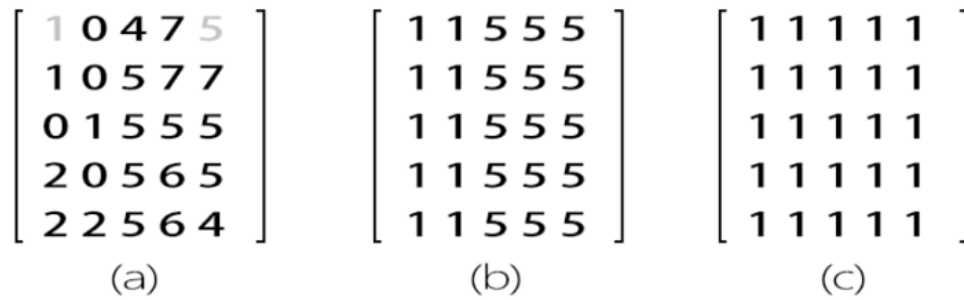


Figure 28 : Examples of regional growth

c) Segmentation basée sur les contours

Les contours des objets dans une image correspondent à des discontinuités locales (brusques variations de luminance, de couleur ou de texture) [123]. Ces discontinuités peuvent être détectées à l'aide d'opérations de dérivation, les dérivées étant calculées via des opérateurs différentiels [124]. Les opérateurs différentiels de premier ordre les plus utilisés sont : Prewitt, Roberts et Sobel [125]. Ceux de second ordre incluent Laplacien, Kirsch et Wallis.

- Segmentation forte
 - Opérateur de Sobel

L'opérateur de Sobel est principalement utilisé pour la détection des contours et consiste en un opérateur différentiel discret de premier ordre, qui estime le gradient de la fonction de luminance de l'image [126]. Contrairement à d'autres opérateurs (Prewitt, Roberts), il applique une pondération supérieure aux pixels centraux, ce qui lui permet de mieux atténuer le bruit tout en détectant efficacement les contours [127]. Il repose sur deux matrices 3×3 qui détectent respectivement les gradients horizontaux et verticaux. Le gradient global est ensuite calculé par :

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (1)$$

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad (2)$$

L'équation (1) permet de détecter les contours horizontaux (template transversal) et l'équation (2) permet de détecter les contours verticaux (template longitudinal). Les approximations du gradient horizontal et vertical de chaque pixel de l'image peuvent ensuite être combinées pour calculer l'amplitude du gradient à l'aide de la formule suivante :

$$G = \sqrt{G_x^2 + G_y^2} \quad (3)$$

Le gradient angulaire (direction du contour) peut ensuite être déterminé selon la formule :

$$\theta = \arctan\left(\frac{G_y}{G_x}\right) \quad (4)$$

Dans l'exemple ci-dessus, si l'angle θ est égal à zéro, cela signifie que l'image possède un contour vertical, avec une partie gauche plus sombre que la partie droite.

○ Opérateur de Laplace

L'opérateur de Laplace est un opérateur isotrope différentiel du second ordre. Il est particulièrement adapté lorsque seul l'emplacement du contour importe, indépendamment des différences de niveaux de gris des pixels environnants. La réponse de l'opérateur de Laplace est plus forte pour des pixels isolés que pour des contours ou des lignes, ce qui limite son utilisation aux images exemptes de bruit. En présence de bruit, il est nécessaire d'appliquer un filtrage passe-bas avant la détection de contours. C'est pourquoi, dans les algorithmes de segmentation, on associe souvent l'opérateur de Laplace à un opérateur de lissage pour former un nouveau modèle.

L'opérateur de Laplace est également le différentiel isotrope le plus simple, invariant par rotation. Sa forme discrète pour une image bidimensionnelle est exprimée comme suit :

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (5)$$

Cette forme est bien adaptée au traitement numérique d'images.

R. C. Gonzalez and R. E. Woods, Digital Image Processing, 4th ed., Pearson, 2018

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (6)$$

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (7)$$

L'équation (6) représente le masque discret standard de Laplace et l'équation (7) en est la version étendue.

L'opérateur de Laplace est utilisé pour améliorer l'effet de flou causé par le phénomène de diffusion, qui survient fréquemment lors du processus d'imagerie. En effet, cet opérateur agit en accentuant les zones de forte variation locale des niveaux de gris, ce qui permet de renforcer les contours et de corriger les effets d'adoucissement ou de diffusion. Cela s'explique par le fait que l'opérateur de Laplace suit un modèle de descente, en accentuant les transitions rapides dans l'image.



Figure 29 : Examples of different operator image show

- **Segmentation faible**

Contrairement à la segmentation forte, où la région segmentée correspond généralement à un objet entier (comme un caractère dans le cas de l'image de document), la segmentation faible peut inclure seulement une partie d'un caractère (si la région sélectionnée est trop petite) ou, à l'inverse, englober plusieurs caractères à la fois (si la région considérée est trop grande).

Bien que la segmentation faible ne soit généralement pas aussi précise ou performante que la segmentation forte, elle présente l'avantage d'être simple à implémenter et efficace en termes de calcul. De plus, dans certaines situations, notamment lorsque des séquences de caractères se répètent dans l'image, la segmentation faible peut se révéler compétitive par rapport à la segmentation forte.

IV. Techniques de reconnaissance

La reconnaissance de manuscrit [128] désigne un ensemble de techniques permettant de convertir une écriture manuscrite (sous forme d'image) en texte numérique. Il existe plusieurs approches, pour cela, qui peuvent être regroupées en deux grandes catégories selon le type d'écriture : En ligne (Online) et hors ligne (Offline).

1. L'approche à base des modèles statistiques

L'approche à base des modèles statistiques est l'une des plus anciennes et fondamentales pour la reconnaissance de l'écriture manuscrite, notamment avant l'émergence de deep learning. Elle repose principalement sur des techniques probabilistes pour modéliser les variations de l'écriture. Le modèle le plus emblématique dans cette catégorie est le modèle de Markov Caché (HMM-Hidden Markov Model) [Abirami]. [129]

2. Reconnaissance par des réseaux de neurones

La reconnaissance de manuscrit à base de réseaux de neurones [130] est une approche largement utilisée, notamment grâce aux avancées de deep learning. Ces modèles permettent de traiter des écritures très variées, sans avoir besoin d'une segmentation explicite des caractères, ce qui était souvent une limite des approches plus anciennes comme HMM. Les algorithmes d'apprentissage profond, notamment les CNN (Convolutional Neural Networks), ont amélioré considérablement la précision de la reconnaissance optique des caractères, notamment pour les écritures manuscrites et les documents dégradés.

3. Reconnaissance Optique de Caractère (ROC) contextuelle

Cette méthode [131] intègre un traitement linguistique (analyse de contexte, grammaire) pour améliorer la reconnaissance dans des cas ambigus (ex. : distinguer le chiffre "0" de la lettre "O"). Elle consiste à utiliser des informations contextuelles pour corriger ou affiner la reconnaissance des caractères ou des mots. Ces informations peuvent provenir de : la langue naturelle (modèles linguistiques), la structure du document (titres, colonnes, listes), la position spatiale des mots (gauche, droite, lignes), et les motifs typographiques ou lexicaux (noms propres, date). Pour le contexte, on utilise généralement plusieurs types de contexte : contexte linguistique, le contexte visuel ou spatial, et le contexte sémantique.

V. Post-traitement

En reconnaissance de manuscrits, le post-traitement joue un rôle essentiel pour améliorer la qualité du texte reconnu après la sortie brute du modèle. Voici quelques techniques couramment utilisées :

1. Utilisation de modèles de langage

Un modèle de langage est utilisé pour corriger les séquences de mots improbables ou incohérentes. Exemple : si le système OCR sort « Je sui allé à la maizon », un modèle de langage peut corriger en « Je suis allé à la maison ». Il y a deux types de modèles de langage : modèles statistiques (bigrammes, trigrammes) et les modèles neuronaux (BERT, GPT, etc.).

2. Dictionnaires et lexiques

Le principe consiste à comparer les mots reconnus avec un dictionnaire de mots valides pour corriger les erreurs de reconnaissance.

Parmi les méthodes utilisant les dictionnaires, on peut trouver : la correction orthographique basée sur la distance de Levenstein et le remplacement des mots hors vocabulaire (OOV) par des suggestions proches.

3. Normalisation du texte

L'objectif est d'uniformiser les formes du texte en procédant à : la conversion en minuscules, la suppression des accents (si nécessaire), et la correction des variantes orthographiques anciennes (utile pour les manuscrits historiques).

4. Re-segmentation ou alignement texte-image

La re-segmentation consiste à réévaluer et ajuster la segmentation du manuscrit (en lignes, mots, ou caractères) après une première reconnaissance automatique, afin d'améliorer la précision du système ou d'aligner le texte reconnu avec un texte de référence. Cela peut être utile lorsque : (1) le découpage initial en lignes ou mots est imparfait, (2) le modèle reconnaît mal les mots à cause d'une mauvaise segmentation, et lorsqu'on souhaite entraîner un modèle avec un alignement précis entre le texte et l'image (alignement mot à mot ou caractère par caractère).

VI. Evaluation

L'évaluation constitue une étape importante dans le processus de développement des systèmes, surtout du point de vue recherche scientifique où les algorithmes et les méthodes utilisés ne sont pas vraiment sûrs. Malheureusement, la plupart des étudiants ne donnent pas assez

d'importance à cette étape car elle vient après la concrétisation du système. Pour eux, une fois le système devient opérationnel leur mission est achevée. Du point de vue examination, il est intéressant d'évaluer le système en vue soit de le valider soit d'entamer une phase d'amélioration. L'évaluation est une sorte de comparaison entre ce qui est désiré et ce que le système nous génère. Pour ce faire, on a besoin d'une collection de donnée qui contient à la fois l'entrée et la sortie désirée ainsi que quelques métriques d'évaluation lors de l'évaluation automatique qui est purement objective. De temps en temps, on va dans le sens de l'évaluation humaine, nommée aussi subjective, qui se base sur l'avis d'un expert humain. Dans ce qui suit, nous présentons les différentes collections de données ainsi que les métriques automatiques pour l'évaluation dans le cadre des systèmes de reconnaissance du manuscrit.

1. Les collections d'évaluation

En consultant les travaux de recherche qui ont adressé la binarisation, on peut trouver les bases d'images à binariser suivantes associées à leurs images références :

La base d'images	Son site web
HDIBCO2009	http://www.iit.demokritos.gr/~bgat/DIBCO2009/benchmark
HDIBCO2010	http://www.iit.demokritos.gr/~bgat/H-DIBCO2010/benchmark
DIBCO2011	http://utopia.duth.gr/~ipratika/DIBCO2011/benchmark
H-DIBCO2012	http://utopia.duth.gr/~ipratika/HDIBCO2012/benchmark
DIBCO2013	http://utopia.duth.gr/~ipratika/DIBCO2013/benchmark
HDIBCO2014	http://users.iit.demokritos.gr/~bgat/HDIBCO2014/benchmark
DIBCO2016	http://vc.ee.duth.gr/h-dibco2016/benchmark
2017	http://vc.ee.duth.gr/dibco2017/benchmark/
2018	http://vc.ee.duth.gr/h-dibco2018/benchmark/
Synthesis Database	https://drive.google.com/drive/folders/15WrU2EnM06wj2AJ49EgOLHVb1yqhPW5a?usp=sharing

2. Les métriques

En consultant la littérature, on peut trouver plusieurs métriques permettant d'évaluer automatiquement et objectivement les différents algorithmes et les différentes techniques

proposées pour binariser une image numérique. Cette comparaison se fait en calculant soit le rapprochement ou l'écart pouvant être trouvé entre la sortie de l'algorithme à évaluer (à savoir l'image binarisée) et l'image attendue (Ground Truth) généralement venue dans un benchmark d'évaluation. Les métriques les plus fréquemment utilisées sont les suivantes :

a) F-measure (FM) :

C'est une métrique qui provient du domaine de la recherche d'information. Elle se base sur deux autres métriques à savoir : la précision et le rappel. A noter que dans la recherche d'information, la précision est le ratio entre le nombre des éléments retournés qui sont pertinents et le nombre d'éléments retournés alors que le rappel est le ratio entre le nombre d'éléments pertinents retournés et le nombre d'éléments pertinents de la base. Dans le contexte de la binarisation, la précision (precision) est le ratio entre le nombre des pixels vrais positifs (True Positif : TP) et le nombre de pixels positifs (les vrais et les faux « True Positif: TP et False Positif: FP) alors que le rappel (Recall) est le ratio entre le nombre de pixels vrais positifs (TP) et le nombre de pixels vrais positifs (TP) et les faux négatifs (False Négatif: FN).

La métrique F-measure (FM) est donnée comme suit :

$$FM = \frac{2 * Recall * Precision}{Recall + Precision}$$

Où :

$$Recall = \frac{TP}{TP+FN}$$
$$Precision = \frac{TP}{TP+FP}$$

b) Pseudo-F-Measure

La métrique Pseudo-F-Measure (PF) qui a été introduite dans [9] utilise la même formule de F-Measure mais au lieu de se baser sur la précision et le rappel, elle se base sur la pseudo-précision (et le pseudo-rappel. Les métriques pseudo-précision/rappel utilisent des poids de distance par rapport au contour des caractères dans l'image à comparer (Ground-Truth: GT). Dans le cas de la métrique pseudo-rappel, les poids du premier-plan (Foreground) sont normalisés selon la largeur de la police (Stroke width). Généralement, les poids sont désignés entre [0, 1]. Dans le cas de la métrique pseudo-précision, les poids sont contraints dans la zone d'arrière-plan de l'image à comparer (GT) en prenant en considération la largeur de la police du composant GT

le plus proche. A l'intérieur de la zone, les poids sont plus grands que 1 (généralement entre [1, 2]) alors qu'en d'hors de cette zone ils sont mis à 1.

c) PSNR

PSNR est une métrique permettant de mesurer à quel point une image est similaire à une autre image. Une valeur très grande de PSNR veut dire que les images sont très proches.

$$PSNR = 10 \log \left(\frac{C^2}{MSE} \right)$$

Où :

$$MSE = \frac{\sum_{x=1}^M \sum_{y=1}^N (I(x, y) - I'(x, y))^2}{MN}$$

La différence entre l'avant plan (Foreground) et l'arrière-plan (Background) égale à C.

d) La distance de distorsion réciproque (Distance Reciprocal Distortion Metric: DRD)

La distance de distorsion réciproque (DRD) a été initialement utilisée pour mesurer la distorsion visuelle dans les images de document binaires [132]. Elle est proprement liée à la perception visuelle humaine et elle mesure la distorsion des tous les pixels inversés comme suit :

$$DRD = \frac{\sum_{k=1}^S DRD_k}{NUBN}$$

Où : $NUBN$ est le nombre des blocks 8x8 non-uniforme (ni tous noir, ni tous blanc) dans l'image à comparer (GT) et DRD_k est la distorsion de $k^{\text{ème}}$ pixel inversé qui est calculé utilisant la matrice des poids normalisés 5x5 W_{Nm} comme définie dans [133]. DRD_k Égale donc à la somme pondérée de pixels dans les blocks 5x5 de l'image à comparer (GT) qui se diffère de $K^{\text{ème}}$ pixel inversé centré (x, y) dans l'image de sortie binarisée B. DRD_k est calculée alors comme suit :

$$DRD_k = \sum_{i=-2}^2 \sum_{j=-2}^2 |GT_k(i, j) - B_k(x, y)| * W_{Nm}(i, j)$$

e) La métrique du ratio négatif (Negative Rate Metric: NRM)

La métrique NRM est basée sur le chevauchement des pixels entre l'image binarisée et l'image référence (GT). Elle combine le ratio négatif faux NR_{FN} et le ratio positif vrai NR_{FP} . Elle se calcule comme suit :

$$NRM = \frac{NR_{FN} + NR_{FP}}{2}$$

Où :

$$NR_{FN} = \frac{N_{FN}}{N_{FN} + N_{TP}} \text{ et } NR_{FP} = \frac{N_{FP}}{N_{FP} + N_{TN}}$$

N_{TP} représente le nombre des positifs vrais, N_{FP} représente le nombre des positifs faux, N_{TN} représente le nombre des négatifs vrais, et N_{FN} représente le nombre des négatifs faux.

Au contraire de F-Mesure et PSNR, la binarisation qualifiée meilleure avec des valeurs petites de NRM.

f) La métrique de pénalité à base de mal-classification (Misclassification Penalty Mteric: MPM)

La métrique MPM évalue la prédiction (l'image binarisée) par rapport l'image référence (GT) à base d'objet par objet. La mal-classification des pixels est pénalisée par leur distance vis-à-vis les bordures d'objets de l'image de référence.

Cette métrique se calcule comme suit :

$$MPM = \frac{MP_{FN} + MP_{FP}}{2}$$

Où :

$$MP_{FN} = \frac{\sum_{i=1}^{N_{FN}} d_{FN}^i}{D}$$

$$MP_{FP} = \frac{\sum_{j=1}^{N_{FP}} d_{FP}^j}{D}$$

d_{FN}^i et d_{FP}^j représentent la distance du $i^{\text{ème}}$ pixel négatif faux et $j^{\text{ème}}$ pixel positif faux du contour de la segmentation de l'image de référence (GT). Le facteur de normalisation D est la somme de toutes les distances par rapport aux pixels contours d'un objet au sein de l'image de référence (GT).

Une valeur petite de MPM veut dire que l'algorithme de binarisation est très performant à l'identification des frontières d'un objet.

VII. Les approches de reconnaissance optique

La reconnaissance optique regroupe un ensemble de techniques visant à extraire automatiquement du texte ou des symboles à partir d'une image. Ces approches sont essentielles dans les systèmes de correction automatique de QCM, notamment pour reconnaître les zones cochées (OMR), les numéros d'identification ou le texte manuscrit (OCR/HTR).

Dans le contexte de la numérisation croissante des données et de l'automatisation des processus administratifs et éducatifs, la Reconnaissance Optique de Marques (OMR, pour Optical Mark Recognition) s'impose comme une technologie efficace et éprouvée. Contrairement à la reconnaissance optique de caractères (OCR), qui tente d'interpréter des lettres ou des mots, l'OMR se concentre sur la détection automatique de marques visuelles simples sur un document papier. Ces marques sont généralement situées dans des zones prédéfinies, telles que des cases, des ronds ou des rectangles.

Cette technologie est particulièrement prisée dans les domaines où de grands volumes de documents similaires doivent être analysés rapidement, comme les examens à choix multiples, les enquêtes papier, ou encore le dépouillement de bulletins de vote.

Contrairement à la reconnaissance de texte (OCR), l'OMR ne tente pas d'identifier des lettres ou des chiffres, mais plutôt de détecter la présence ou l'absence de marques dans des emplacements prédéfinis sur un formulaire.

Le processus typique d'OMR comprend :

- Scan du document : via un scanner classique ou spécialisé.
- Localisation des zones de réponse : grâce à un modèle de formulaire ou à des repères (cases ou cercles).
- Analyse de remplissage : le logiciel détecte si la marque est présente, en général selon une densité d'encre ou de pixels.
- Extraction des données : les marques sont converties en valeurs numériques ou textuelles exploitables par des logiciels.

L'OMR est appliqué dans plusieurs secteurs de notre vie contemporaine comme (1) dans l'éducation (Correction automatisée de QCM dans les examens, Évaluations nationales

standardisées comme les examens du bac dans certains pays), (2) dans les administrations et les entreprises (Sondages d'opinion et questionnaires de satisfaction et Enquêtes démographiques ou formulaires de recensement), et dans le secteur politique (Comptage automatisé de bulletins de vote, dans les systèmes de vote papier à choix multiples).

VIII. Conclusion

Dans ce premier chapitre d'état de l'art, nous avons abordé en détails le processus de reconnaissance du manuscrit. Nous avons parlé de prétraitement, de post-traitement, des techniques de reconnaissance, et d'évaluation.

Chapitre 02 : Questionnaire à choix multiple (QCM)

I. Introduction

Depuis plusieurs décennies, le Questionnaire à Choix Multiples (QCM) constitue une méthode d'évaluation largement adoptée dans divers secteurs, notamment l'enseignement supérieur, la formation professionnelle et les concours. Il repose sur la présentation d'une question suivie de plusieurs propositions de réponses, parmi lesquelles le candidat doit sélectionner la ou les bonnes réponses. Ce mode d'évaluation se distingue par sa rapidité d'administration et de correction ainsi que par sa capacité à traiter un grand volume de participants de manière standardisée et impartiale.

Comme le soulignent Bouvy et Warnier [134] (2016), le QCM permet d'interroger un large éventail de connaissances dans un temps limité, tout en limitant la subjectivité de la correction et de l'évaluation des réponses ». Lorsqu'il est rigoureusement élaboré, cet outil peut évaluer aussi bien des connaissances factuelles que des compétences cognitives de niveaux supérieurs. Avec le développement des technologies de traitement d'image et des systèmes de reconnaissance automatisée, de nouvelles opportunités se dessinent pour la gestion et la correction des QCM. Les techniques d'Optical Mark Recognition (OMR) et de Reconnaissance Optique de Caractères (OCR) se sont progressivement imposées comme des solutions efficaces pour le traitement de documents papiers numérisés. L'OMR permet de détecter et d'interpréter des marques manuscrites ou des cases cochées sur des formulaires standardisés, tandis que l'OCR vise à convertir des textes imprimés ou manuscrits en fichiers numériques exploitables. Ces technologies sont largement utilisées dans divers contextes. Par exemple, l'OMR est couramment déployée dans la correction automatisée des examens à grande échelle, des sondages et des enquêtes administratives [135]. L'OCR, quant à elle, trouve des applications dans la numérisation d'archives administratives, la lecture automatique de formulaires et la gestion documentaire, mais également dans des dispositifs pédagogiques tels que la correction de copies manuscrites et l'extraction de textes dans des QCM [136].

Ces avancées offrent des gains de temps significatifs et améliorent la fiabilité et l'équité des procédures d'évaluation en limitant les erreurs humaines et en assurant une correction standardisée. Dans le contexte actuel d'enseignement de masse et de digitalisation des procédures administratives et pédagogiques, ces systèmes automatisés s'inscrivent pleinement dans les exigences d'efficience et de neutralité du processus d'évaluation.

II. Le Système Pédagogique

L'organisation de l'enseignement et de l'apprentissage repose sur ce qu'on appelle un système pédagogique, c'est-à-dire un ensemble structuré de méthodes, de ressources et de dispositifs mis en œuvre pour transmettre des savoirs et évaluer les acquis des apprenants. Ce système englobe aussi bien les contenus pédagogiques que les stratégies d'enseignement, les modalités d'évaluation et les outils technologiques mobilisés pour soutenir ces processus.

Comme le souligne Lebrun (2005), un système pédagogique peut être défini comme « un ensemble cohérent d'éléments organisationnels, humains et matériels mis en interaction pour atteindre des objectifs éducatifs dans un cadre précis » [137]. Ce cadre n'est pas figé ; il évolue en fonction des contextes socio-éducatifs, des politiques institutionnelles et des avancées technologiques.

Ces dernières années, l'émergence et la généralisation des technologies numériques ont profondément transformé les systèmes pédagogiques traditionnels. L'intégration des plateformes d'apprentissage en ligne, des environnements numériques de travail et des dispositifs d'évaluation automatisée — tels que les systèmes de correction automatique de QCM — participent à cette évolution. Ces solutions permettent de diversifier les approches pédagogiques, de personnaliser les parcours d'apprentissage et de rendre l'enseignement plus accessible et plus efficient.

Dans ce contexte de mutation, l'automatisation de certaines tâches pédagogiques, notamment l'évaluation, représente un enjeu majeur pour les institutions éducatives. Elle permet de réduire la charge de travail des enseignants, d'améliorer la rapidité et l'objectivité des évaluations, et de renforcer ainsi la qualité des dispositifs pédagogiques.

III. Interrogation de l'Apprenant

L'évaluation des connaissances et des compétences de l'apprenant occupe une place centrale dans les dispositifs pédagogiques, qu'ils soient en présentiel ou à distance. L'interrogation de l'apprenant constitue un processus essentiel permettant de mesurer les acquis, de vérifier la compréhension des contenus enseignés et d'ajuster les démarches pédagogiques en conséquence. Parmi les nombreuses formes d'interrogation, le Questionnaire à Choix Multiples (QCM) s'est imposé comme un outil de référence en raison de sa simplicité de mise en œuvre, de sa rapidité de correction et de son objectivité dans le traitement des réponses [138].

Le recours au QCM permet de soumettre l'apprenant à un large éventail de questions couvrant différentes dimensions des savoirs, allant des connaissances factuelles aux compétences

d'analyse et de raisonnement [139]. Cet outil d'interrogation se prête aussi bien à des évaluations formatives, qui accompagnent le processus d'apprentissage, qu'à des évaluations sommatives destinées à valider les acquis à la fin d'un cycle d'enseignement.

Avec l'intégration des technologies numériques dans les systèmes éducatifs, l'interrogation de l'apprenant connaît aujourd'hui des évolutions significatives. Des travaux récents, tels que celui de Benhassine (2018) [140] dans son mémoire sur l'automatisation des corrections d'examens universitaires, démontrent que l'usage combiné du QCM et des systèmes de reconnaissance optique (OMR/OCR) permet non seulement de simplifier le processus d'interrogation, mais aussi de renforcer l'équité et la transparence de l'évaluation, notamment dans les dispositifs d'enseignement à grand effectif.

Par ailleurs, plusieurs auteurs soulignent l'importance d'élaborer des items de qualité pour garantir la validité et la fiabilité des QCM en tant qu'outils d'interrogation [141]. L'enjeu consiste à concevoir des questions qui ciblent réellement les compétences visées et à éviter les biais susceptibles d'altérer les résultats.

Enfin, l'interrogation de l'apprenant s'inscrit désormais dans un contexte où l'automatisation et la numérisation des processus pédagogiques sont devenues des exigences incontournables. Cette dynamique est illustrée par des projets tels que ceux menés par Boukhrissa (2020), qui a proposé un système de gestion et de correction automatisée des QCM dans un cadre universitaire, confirmant l'intérêt croissant pour des solutions associant ergonomie, efficacité et fiabilité dans l'interrogation et l'évaluation des apprenants.

IV. Le QCM : Définition et enjeux

Le Questionnaire à Choix Multiples (QCM) est une méthode d'évaluation normée, reposant sur des questions assorties de plusieurs propositions de réponse parmi lesquelles le candidat doit sélectionner celles qu'il estime correctes. Le QCM constitue un outil d'évaluation fermé qui facilite la standardisation des examens et la comparaison entre les candidats [142]. Sa principale caractéristique réside dans le fait que les réponses sont limitées à un ensemble prédéfini, réduisant ainsi l'interprétation subjective au moment de la correction.

Sur le plan pédagogique, le QCM joue un rôle stratégique dans l'évaluation des connaissances factuelles et de la compréhension. Il est particulièrement utilisé dans les examens à grand effectif et les concours en raison de sa capacité à garantir objectivité et gain de temps [143]. De plus, il offre des possibilités de correction automatique ou semi-automatique via des systèmes informatisés ou des technologies de lecture optique.

1. Différents types de QCM

Dans la littérature pédagogique et les travaux appliqués, on distingue plusieurs formes de QCM, chacune adaptée à des objectifs et contextes d'évaluation spécifiques [144].

a) QCM à réponse unique

Ce type de QCM propose plusieurs réponses parmi lesquelles une seule est correcte. Il est simple à concevoir et à corriger, et reste le plus utilisé dans les examens standardisés.

b) QCM à réponses multiples

Dans ce format, plusieurs propositions peuvent être correctes. Le candidat doit cocher toutes les bonnes réponses et uniquement celles-ci. Ce type nécessite souvent un barème particulier et rend la correction plus délicate, notamment lorsqu'elle est automatisée.

c) QCM à pondération

Chaque réponse se voit attribuer un poids ou une note partielle. Le total obtenu dépend des réponses sélectionnées et de leur poids relatif. Cette approche est utile pour valoriser des choix plus pertinents et affiner l'évaluation.

d) QCM à correction négative

Afin de dissuader les réponses au hasard, certaines épreuves appliquent une pénalité en cas de réponse erronée. Cette technique est fréquente dans les concours sélectifs où le taux de réussite doit être maîtrisé.

2. Avantages et limites du QCM

a) Avantages

Le QCM présente de nombreux atouts, qui justifient sa large adoption dans les évaluations universitaires et professionnelles : (1) Objectivité de la correction, en supprimant la part subjective présente dans les évaluations libres [145], (2) Gain de temps considérable, tant pour la passation que pour la correction, surtout lorsqu'elle est automatisée [146].

Standardisation de l'évaluation, facilitant les comparaisons interindividuelles et inter-promotions, et (3) Facilité d'analyse statistique des résultats, permettant d'identifier les questions discriminantes et de calibrer les examens.

b) Limites

En dépit de ses qualités, le QCM présente certaines faiblesses : (1) Favorise le hasard si les distracteurs sont mal conçus, (2) difficile d'évaluer des compétences de synthèse, d'analyse et de rédaction [147], (3) sensible aux erreurs de remplissage sur support papier, notamment en correction automatique, et (4) la qualité pédagogique du QCM dépend fortement de la pertinence des distracteurs et de la clarté des énoncés. [148]

3. Système de Correction à basse de QCM

Depuis l'apparition des QCM dans les dispositifs d'évaluation à grande échelle, différentes méthodes de correction ont été mises en place, évoluant progressivement d'une correction manuelle vers des systèmes semi-automatiques, puis automatiques. Ces évolutions ont été motivées par la nécessité d'améliorer la rapidité, la fiabilité et l'objectivité des corrections, notamment dans le contexte des examens de masse (Bouvy & Warnier, 2016 ; Guedjali, 2018) [149].

a) Correction manuelle

La correction manuelle des QCM consiste à vérifier visuellement les réponses fournies par les candidats en les comparant à un corrigé-type, et à attribuer la note correspondante. Cette méthode, encore utilisée dans certains établissements à effectifs réduits ou dans les formations professionnelles, présente toutefois plusieurs inconvénients majeurs.

La correction manuelle devient difficilement envisageable lorsque le nombre de copies dépasse quelques centaines. Elle est extrêmement chronophage et sujette à des erreurs humaines, dues à la fatigue, à l'inattention ou à une mauvaise interprétation des réponses [150]. De plus, cette méthode limite les possibilités d'analyse statistique automatisée des résultats et n'offre aucune traçabilité en cas de réclamation.

Dans une étude menée par Boussad [151], il a été observé que la correction manuelle mobilise un personnel important et peut retarder la publication des résultats, ce qui devient problématique dans les contextes universitaires où les calendriers pédagogiques sont souvent serrés.

b) Correction semi-automatique

La correction semi-automatique des QCM repose sur l'utilisation de grilles de correction ou de gabarits plastifiés superposés aux feuilles de réponse. Le correcteur vérifie visuellement les réponses cochées dans les cases à travers les ouvertures de la grille, ce qui facilite le repérage des bonnes réponses [152].

Bien que cette méthode accélère légèrement le processus et limite les erreurs de comparaison avec le corrigé, elle reste manuelle dans l'attribution des notes et sujette aux imprécisions. Cette approche, ne permet pas de supprimer totalement les erreurs humaines et reste peu adaptée à des volumes très importants de copies.

Par ailleurs, cette technique ne s'applique qu'aux QCM à réponse unique et à mise en page parfaitement standardisée, ce qui limite son utilisation dans les examens à structure variable.

c) Correction automatique par scanner OMR (Optical Mark Recognition)

L'OMR (Optical Mark Recognition) est une technologie de reconnaissance optique qui permet de détecter automatiquement les marques effectuées par les candidats sur des feuilles de QCM préformatées. Cette méthode repose sur la détection de zones noires ou de cases remplies à des positions précises, correspondant aux choix de réponse [153].

L'OMR permet d'accélérer considérablement le processus de correction, en traitant plusieurs centaines de copies en quelques minutes. Elle supprime les erreurs humaines et offre des possibilités d'exploitation statistique immédiate des résultats [154].

Le fonctionnement d'un système OMR repose sur : (1) la numérisation de la feuille de réponse via un scanner ou un appareil spécialisé, (2) la localisation des zones de réponse à l'aide de repères prévus sur la feuille, (3) la lecture automatique des cases cochées en détectant les zones noircies par analyse de contraste, (4) la comparaison des réponses lues avec le corrigé officiel, pour attribuer les notes, (5) l'édition des résultats et des statistiques.

Cette méthode nécessite des feuilles parfaitement standardisées, avec un positionnement rigoureux des cases et des repères de cadrage [155]. Malgré ses avantages, l'OMR présente plusieurs limites : (1) dépendance à la qualité d'impression et de numérisation. Une feuille floue, tachée, ou légèrement inclinée peut perturber la lecture optique [156], (2) erreurs de remplissage : cases mal cochées, raturées ou cochées partiellement, souvent indétectables ou mal interprétées par le système [157], (3) nécessité d'un format strict et uniforme des copies. Toute modification de mise en page nécessite un recalibrage du système [158], et (4) coût élevé

des scanners OMR professionnels, difficilement accessibles pour les petites structures universitaires ou établissements de formation. [159]

C'est pour répondre à ces limites que des approches hybrides et des solutions logicielles basées sur des outils open-source comme OpenCV et Tesseract-OCR ont été explorées dans plusieurs mémoires et thèses.

V. Conclusion

Le Questionnaire à Choix Multiples (QCM) s'est imposé, au fil des décennies, comme l'un des outils d'évaluation les plus répandus et les plus appréciés dans les systèmes éducatifs et professionnels. Par sa capacité à évaluer un large panel de connaissances et de compétences en un temps limité et de manière standardisée, il constitue un dispositif d'évaluation efficace et objectif, notamment dans les contextes de forte affluence.

L'étude approfondie des différentes formes de QCM — à réponse unique, à réponses multiples, à pondération ou à correction négative — a permis de mettre en lumière leur diversité d'applications et les précautions méthodologiques à respecter lors de leur élaboration. La qualité d'un QCM repose avant tout sur la clarté des énoncés, la pertinence des distracteurs et la cohérence du barème de correction, éléments déterminants pour garantir la validité et la fiabilité de l'évaluation.

Néanmoins, malgré ses nombreux avantages, le QCM présente certaines limites. Sur le plan pédagogique, il est parfois critiqué pour sa tendance à favoriser la mémorisation plutôt que la réflexion critique et la capacité d'analyse. De plus, sa capacité à évaluer des compétences rédactionnelles et argumentatives reste limitée, ce qui en restreint l'usage à certains types de disciplines et de savoirs.

Du point de vue logistique, la correction des QCM — notamment sur support papier — constitue un enjeu majeur dès lors que le nombre de candidats est important. La correction manuelle, bien que simple, s'avère chronophage, sujette aux erreurs et difficilement exploitable statistiquement. Les systèmes semi-automatiques apportent des améliorations mais restent limités dans leur capacité à supprimer totalement l'intervention humaine et à garantir une correction rapide et sécurisée à grande échelle.

Face à ces contraintes, les systèmes de correction automatique par reconnaissance optique (OMR) se sont progressivement imposés dans les établissements universitaires et les organismes de formation. Ils offrent des gains de temps considérables et une réduction

significative des erreurs de traitement. Toutefois, ces systèmes restent dépendants de la qualité du support papier et de la normalisation des grilles de réponses, limitant parfois leur flexibilité. Dans ce contexte, les récents travaux de recherche et les mémoires ont mis en évidence l'intérêt croissant pour des solutions basées sur les techniques de traitement d'image et de reconnaissance optique plus souples et adaptatives. Ces approches offrent la possibilité de traiter des copies numérisées, même en présence d'imperfections d'impression ou de remplissage, et permettent de développer des systèmes personnalisés adaptés aux contraintes pédagogiques et matérielles locales.

Ainsi, cette synthèse du concept du QCM, de ses formes et de ses méthodes de correction permet de mieux cerner les enjeux liés à l'évaluation de masse et d'identifier les pistes d'amélioration technologique à explorer. Elle justifie pleinement la nécessité de concevoir un système performant de correction automatique de QCM à base de cases à cocher sur support papier, intégrant des techniques avancées de traitement d'image et de reconnaissance optique, objectif principal de ce mémoire.

Partie 2 : Contribution : Analyse, Conception, et Implémentation

Chapitre 03 : Analyse et conception

I. Introduction

La conception et la réalisation constituent des étapes fondamentales dans le cycle de développement de tout système informatique. Après avoir défini le cadre théorique et exposé l'état de l'art relatif aux systèmes de correction automatique des QCM, il devient nécessaire de traduire ces connaissances en un modèle pratique, opérationnel et adapté aux besoins identifiés. Cette partie a pour objectif de décrire de manière détaillée la démarche adoptée pour concevoir et implémenter le système proposé.

Dans un contexte où les établissements éducatifs doivent gérer des volumes croissants d'examens et où la correction manuelle présente des contraintes en termes de temps, de fiabilité et de coûts, le recours à un système automatisé devient indispensable. Ce projet vise à développer un système de correction automatique des examens à base de QCM (SCAEQCM) papier en se basant sur la reconnaissance optique des cases cochées (OMR) et éventuellement des caractères (OCR), couplé à un algorithme de traitement pour l'analyse et la correction des réponses.

On commence ce chapitre alors par donner quelques informations concernant notre méthode de modélisation UML. Puis, on présente la motivation de notre travail pour entamer la phase d'analyse et de conception en présentant quelques diagrammes selon la notation UML.

II. Le langage de modélisation Unifié (UML)

Le développement d'un système logiciel, notamment un système de correction automatique de QCM, repose sur une méthodologie rigoureuse qui garantit la qualité, la maintenabilité et l'évolutivité du produit final. Cette méthodologie est étudiée et formalisée dans une discipline qu'on appelle le génie logiciel.

Le développement logiciel s'organise généralement en plusieurs phases clés, inspirées des modèles en cascade ou itératifs (modèle en V, RAD, Scrum, etc.). Les principales sont :

1. Analyse des besoins :

Recueillir et formaliser les exigences fonctionnelles et non fonctionnelles du système.

2. Conception :

Traduire les besoins en une architecture technique et logicielle à l'aide d'outils comme UML.

3. Implémentation(codage) :

Transformer les modèles de conception en code exécutable.

4. Tests et validation :

Vérifier la conformité du logiciel aux spécifications et corriger les anomalies.

5. Déploiement :

Installer le logiciel dans l'environnement de production.

6. Maintenance évolutive et corrective :

Adapter et améliorer le système en fonction des retours utilisateurs et des évolutions techniques [160].

Durant la phase de conception, le langage UML (Unified Modeling Language) intervient pour :

- Structurer les besoins via des diagrammes de cas d'utilisation.
- Modéliser les processus avec des diagrammes d'activité.
- Organiser les composants logiciels via des diagrammes de classes.
- Planifier les interactions système grâce aux diagrammes de séquence.

Ces diagrammes facilitent la communication entre les développeurs, les examinateurs et les utilisateurs finaux du système.

III. Analyse des Besoins

1. Motivation

La tendance actuelle d'évaluation dans le domaine de l'enseignement supérieur est vers l'adoption des examens à base des QCMS. Cette façon d'interroger les candidats semble contrôlable du point de vue correction et aussi du point de vue confection. Malgré que la correction des examens à base des QCMs est simple, le nombre important des copies ainsi que la durée limitée pour afficher les résultats rendent le basculement vers l'utilisation des systèmes automatiques une nécessité.

L'objectif de notre mémoire de Master alors est de matérialiser cette idée en concevant et implémentant un système purement automatique de correction des examens à base des QCMs.

2. Notre Approche

Au-delà de l'opération de correction automatique, l'originalité de ce système réside dans l'intégration d'un module **'d'assistance à la création des examens sous forme de QCM à cases à cocher'**. Ce module permettra à un confectionneur humain de concevoir ses sujets directement via une interface dédiée, générant des feuilles physiques standardisées prêtes à être distribuées aux étudiants et scannées par la suite. Le système assure ensuite la détection des réponses cochées, la correction automatique selon un barème défini et la génération des résultats et des statistiques.

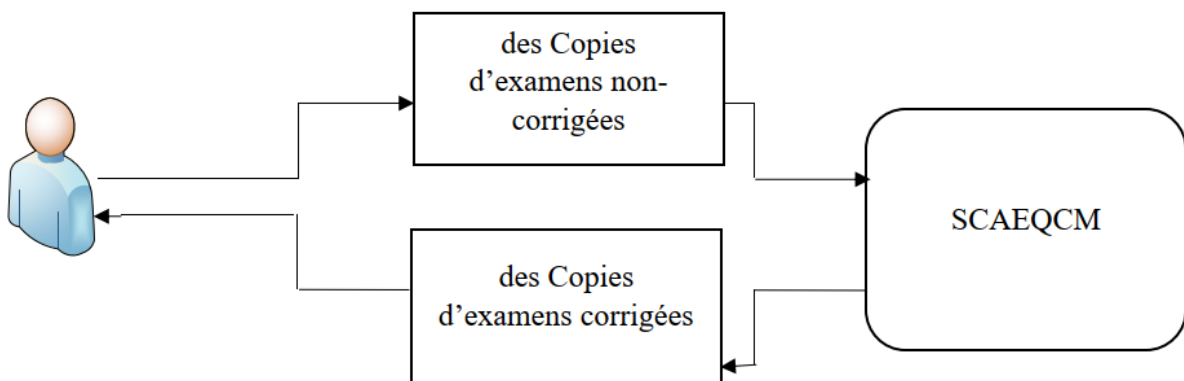


Figure 30 : Architecture Générale du Système SCAEQCM.

Ce chapitre d'analyse de besoins et de conception alors présentera successivement :

- Une architecture générale de notre futur système, incluant les différents acteurs et modules fonctionnels.
- Quelques diagrammes UML utilisés pour bien décrire le système en modélisant les cas d'utilisation, les séquences d'interaction et la structure des classes du système.

L'architecture générale de notre système SCAEQCM repose sur une organisation modulaire, structurée autour de plusieurs composants interconnectés et d'un seul acteur : Le confectionneur du sujet d'examen. Le Confectionneur est celui qui est responsable sur la création des sujets d'examen via un module assistant de conception à base de QCM, de la définition des barèmes de correction, et de la supervision du processus de correction. A noter que la confection des sujets d'examen est une tâche délicate qui nécessite elle-même tout un système automatique

dédié. Ce conseil peut se figurer plus tard à la section ‘*perspectives*’ comme un point qu’on encourage la communauté à prendre en considération dans le futur. Le candidat (*L’examiné*) qui reçoit le sujet généré par notre future application, qui coche les cases correspondant aux suggestions choisies sur la feuille d’examen imprimée, et qui remet la copie à la fin de l’examen pour être corrigée, ne figure pas parmi les acteurs de notre futur système.

Voilà l’architecture fonctionnelle de notre futur système qui montre l’ensemble de ressources entrées au système ainsi que l’ensemble de sorties générées.

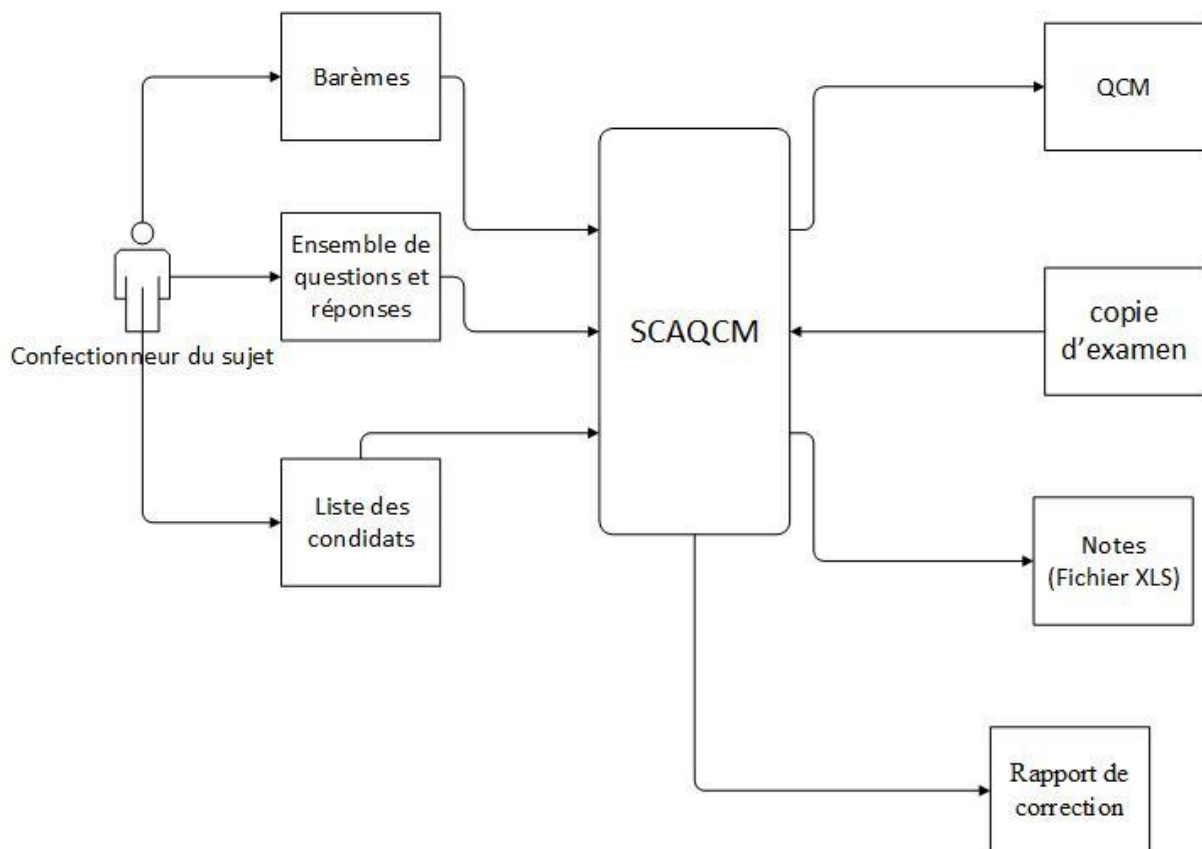


Figure 31 : Architecture fonctionnelle de notre futur système.

La figure ci-dessous présente l’architecture fonctionnelle détaillée de notre futur système.

Selon la figure, notre future système contient les modules suivants : (1) Confection du sujet (Formulaire), (2) Confection du sujet (Formulaire), (3) Scan des copies, (4) Vérification de la qualité d’image, (5) Binarisation, (6) Détection des cases, (8) Détection des cases à cocher, (9) Attribution des points(scores), (10) Calcule de la note globale, (11) Détection des noms(segmentation), (12) Détection des prénoms(segmentation), (13) Détection de groupe, (14) Reconnaissance des caractères du nom, (15) Reconnaissance du nom, (16) Reconnaissance

des caractères du prénom, (17) Reconnaissance du prénom, (18) Reconnaissance des chiffres du groupe, (19) Reconnaissance du groupe, (19) Addition de la note globale de l'étudiant dans le fichier Excel, (20) Validation des données, (21) Ajout dans le fichier Excel, (22) Enregistrement des erreurs éventuelles, (23) Génération d'un rapport de correction.

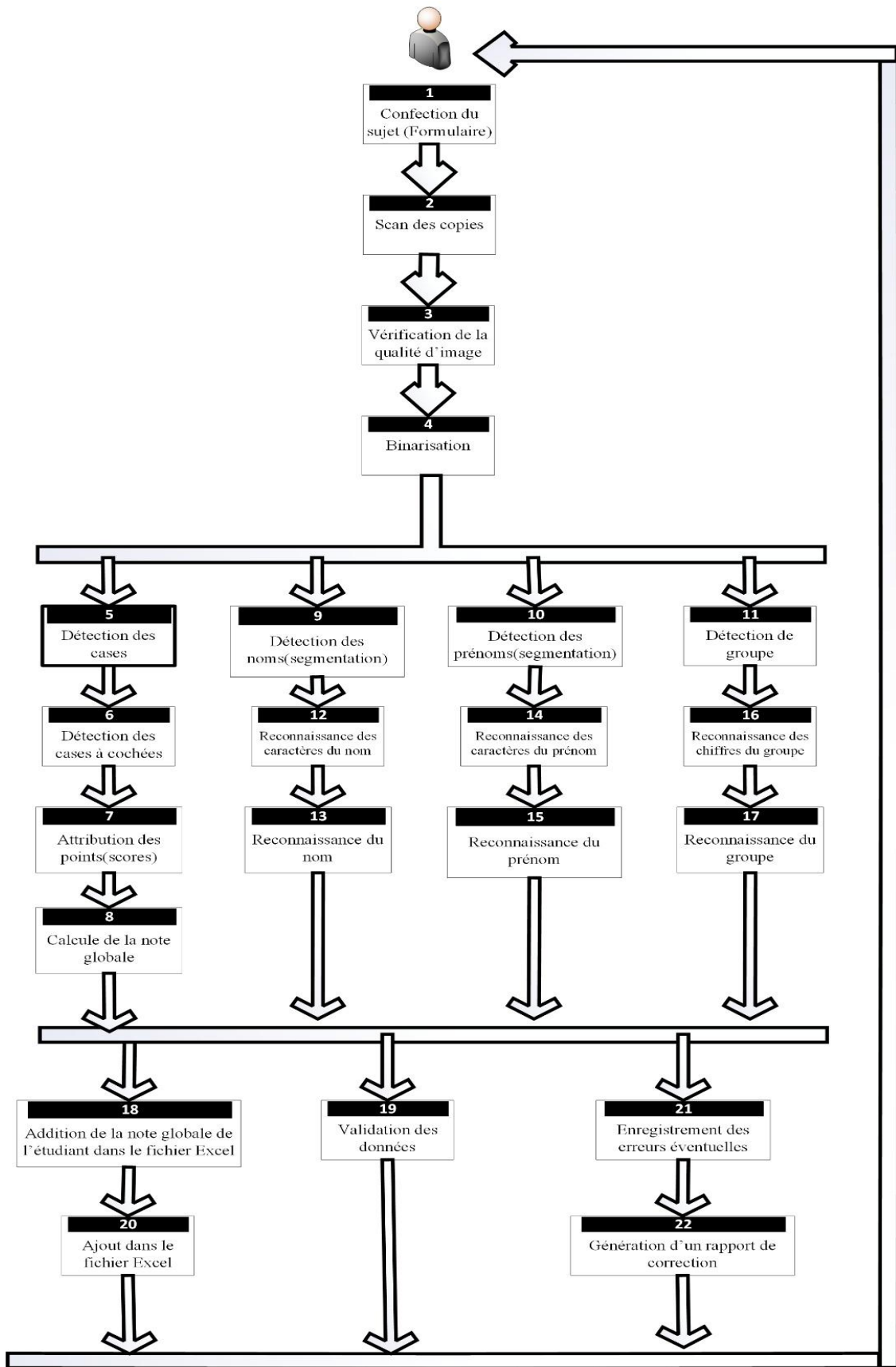


Figure 32 : Architecture fonctionnelle détaillée.

Voilà dans ce qui suit le pseudo code qui englobe toutes les étapes de notre futur système.

Pseudocode :

DEBUT

// Étape 1 : Confection du sujet (Formulaire)

Générer le formulaire QCM avec cases à cocher et champs pour nom, prénom, groupe
Imprimer le sujet et le distribuer

// Étape 2 : Scan des copies

POUR chaque copie d'examen scannée **FAIRE** Charger l'image de la copie

// Étape 3 : Vérification de la qualité d'image

SI image floue ou mal cadrée **ALORS** Consigner l'erreur et demander un nouveau scan
Passer à la copie suivante

SINON Continuer

// Étape 4 : Binarisation

Convertir l'image en noir et blanc
Appliquer un filtrage pour nettoyer l'image

// Étape 5 : Détection des cases

Localiser les positions des cases dans l'image

// Étape 6 : Détection des cases cochées

POUR chaque case détectée **FAIRE**

SI case cochée (selon densité de pixels) **ALORS** Enregistrer la réponse cochée

FIN SI

FIN POUR

// Étape 7 : Attribution des points (scores)

POUR chaque question **FAIRE**

SI réponse correcte **ALORS** Ajouter le score de la question

FIN SI

FIN POUR

// Étape 8 : Calcul de la note globale

Calculer la note totale de l'étudiant

// Étape 9 : Détection des noms (segmentation)

Localiser la zone du nom dans l'image

// Étape 10 : Détection des prénoms (segmentation)

Localiser la zone du prénom

// Étape 11 : Détection du groupe

Localiser la zone du groupe

// Étape 12 : Reconnaissance des caractères du nom (OCR)

Appliquer OCR sur la zone du nom

Nettoyer et valider le texte reconnu

// Étape 13 : Reconnaissance du nom

Enregistrer le nom reconnu

// Étape 14 : Reconnaissance des caractères du prénom (OCR)

Appliquer OCR sur la zone du prénom

Nettoyer et valider

// Étape 15 : Reconnaissance du prénom

Enregistrer le prénom reconnu

// Étape 16 : Reconnaissance des chiffres du groupe (OCR)

Appliquer OCR sur la zone du groupe

Nettoyer et valider

// Étape 17 : Reconnaissance du groupe

Enregistrer le groupe reconnu

// Étape 18 : Addition de la note globale de l'étudiant dans le fichier Excel

Préparer la ligne (Nom, Prénom, Groupe, Note)

// Étape 19 : Validation des données

SI Nom, Prénom, Groupe et Note valides **ALORS** Passer à l'étape suivante

SINON Consigner une erreur et passer à la copie suivante

// Étape 20 : Ajout dans le fichier Excel

Insérer la ligne dans le fichier Excel des résultats

// Étape 21 : Enregistrement des erreurs éventuelles

Enregistrer toute erreur ou anomalie détectée durant le traitement

FIN POUR

// Étape 22 : Génération d'un rapport de correction

Générer le rapport global des résultats :

- Nombre de copies corrigées
- Notes moyennes
- Statistiques de réponses
- Liste des erreurs

FIN

Dans ce qui suit, nous présentons en détails le mode de fonctionnement de chaque module.

- Confection du sujet (Formulaire)

Avant de pouvoir corriger les copies d'examen, le système SCAEQCM propose à l'enseignant ou au confectionneur de sujet une interface graphique conviviale qui permet de :

- Créer et structurer les questions de type QCM.
- Associer à chaque question les différentes réponses possibles (A, B, C, D...).
- Définir les bonnes réponses et leur barème (points attribués).
- Définir le nombre de questions et le nombre de réponses par question.
- Générer automatiquement le modèle d'examen à imprimer, comprenant : Les questions, les cases à cocher alignées et numérotées selon une séquence prédéfinie, les zones réservées au nom, prénom et groupe de l'étudiant.

Malgré que la tâche de confection de sujet ne soit pas la tâche primordiale dans notre futur système, elle reste quand même une tâche d'importance car elle permet d'écarter le confectionneur ou l'examineur de tout ce qui est mise en forme. C'est vrai que cette tâche n'est pas vraiment une confection au sens propre du mot car ce n'est pas le système qui propose les différentes questions ainsi que leurs suggestions associées mais il aide seulement à bien rédiger l'examen dans une mise en forme adéquate en se basant sur les questions et les suggestions de l'examineur. On peut accentuer encore une fois que cette tâche est importante du point de vue examinateur en soulignant que la confection du sujet qui se base sur le sens et la sémantique liée au contenu du sujet lui-même (le contenu de la matière en question) est encore plus importante et qui nécessite d'être citée dans la section 'perspectives' afin d'être réalisée dans les futurs travaux surtout avec l'apparition de quelques assistants d'intelligence artificiel comme ChatGPT qui peuvent être bien exploités dans ce sens.

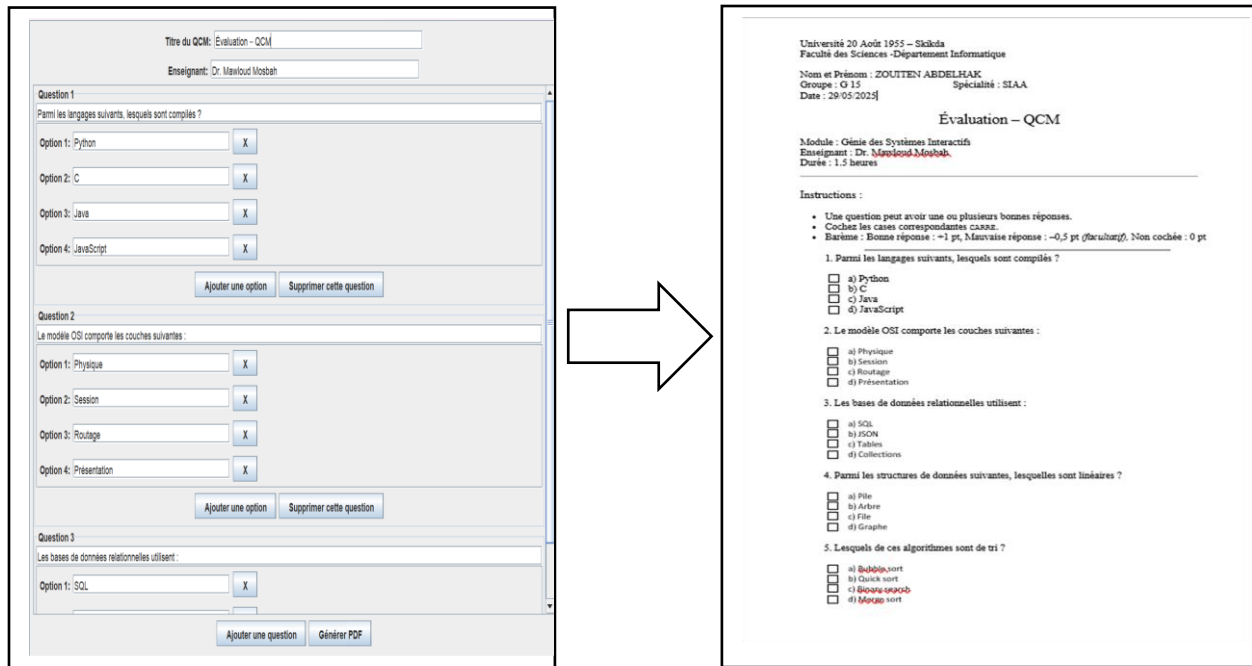


Figure 34 : Confection du sujet d'examen en se basant sur un formulaire.

- Scan des copies

Après que les candidats passent leurs examens, les copies papier des candidats sont regroupées pour être numérisées. Le système SCAEQCM prévoit un processus de numérisation par lot utilisant un chargeur automatique de documents (ADF) afin d'assurer un traitement rapide et organisé des copies.

Les copies sont insérées dans le chargeur et le processus de scan est lancé. Chaque copie est convertie en une image numérique selon un format et une résolution prédéfinie (ex : TIFF en 300 DPI en niveaux de gris). Les images générées sont automatiquement stockées dans un répertoire dédié en attente de traitement ultérieur.

Afin d'améliorer la fiabilité et la gestion du flux de numérisation, nous proposons d'intégrer plusieurs fonctionnalités complémentaires à cette étape : (1) enregistrement du nombre attendu de copies avant le scan, permettant au système de comparer dynamiquement le nombre de copies scannées au nombre prévu, (2) journalisation automatique du processus de scan en utilisant les attributs suivants : Nom du fichier, horodatage, état (valide / à vérifier), nom de l'opérateur ayant effectué le scan, et éventuelles les anomalies détectées, (3) contrôle qualité renforcé en temps réel par le biais des attributs suivants : vérification de la résolution de chaque image, *détection d'images* partiellement blanches (ex : page vide ou très peu remplie), et

détection de défauts fréquents comme : inclinaison importante, le flou, des zones noires, ou des marges tronquées, et finalement (4) organisation automatique des images.

Université 20 Août 1955 – Skikda
Faculté des Sciences -Département Informatique

Nom et Prénom : _____
Groupe : _____ Spécialité : _____
Date : _____

Évaluation – QCM

Module : _____
Enseignant : _____
Durée : _____ heures

Instructions :

- Une question peut avoir une ou plusieurs bonnes réponses.
- Cochez les cases correspondantes **CAREE**.
- Barème : Bonne réponse : +1 pt, Mauvaise réponse : -0.5 pt (*facultatif*), Non cochée : 0 pt

1. Parmi les langages suivants, lesquels sont compilés ?

- Python
- C
- Java
- JavaScript

2. Le modèle OSI comporte les couches suivantes :

- Physique
- Session
- Routage
- Présentation

3. Les bases de données relationnelles utilisent :

- SQL
- JSON
- Tables
- Collections

4. Parmi les structures de données suivantes, lesquelles sont linéaires ?

- Pile
- Arbre
- File
- Graphe

5. Lesquels de ces algorithmes sont de tri ?

- Bubble sort
- Quick sort
- Binary search
- Merge sort

Figure 35 : Un exemple d'une copie d'examen scanné.

- Vérification de la qualité d'image

Le système vérifie automatiquement : La résolution (elle doit généralement être supérieure ou égale à 300 dpi), le contraste, *l'absence* du flou, *l'inclinaison* (deskewing), et la présence des *repères d'alignement*. Les images non conformes sont écartées pour rescannés ou corrigés manuellement.

- Binarisation

Pour rendre le traitement simple et possible, l'image couleur ou en niveau de gris est convertie en noir et blanc (binarisation) en utilisant un algorithme de seuillage. En consultant la littérature, on peut trouver plusieurs méthodes de binarization. Pour notre cas d'étude, on a opté pour la méthode Canny (Otsu et adaptive thresholding de OpenCv). Cette méthode est considérée sans trop réfléchir autour de la meilleure méthode car on considère que chaque méthode de la

littérature peut nous bien nous servir dans notre cas non exigeant et on se contente par le fait de choisir une méthode sans justification malgré que la méthode adoptée qui se base sur un seuillage semble simple à implémenter. Les pixels au-dessus du seuil deviennent blancs. Donc, les pixels avec une valeur en dessous du seuil deviennent noirs et ceux avec une valeur supérieure du seuil deviennent blancs tout simplement. L'adoption d'une binarisation alors rendre facile le parcours de l'image de document pixel par pixels afin de détecter que ce soient les caractères, les cases, ou les cases à cocher.

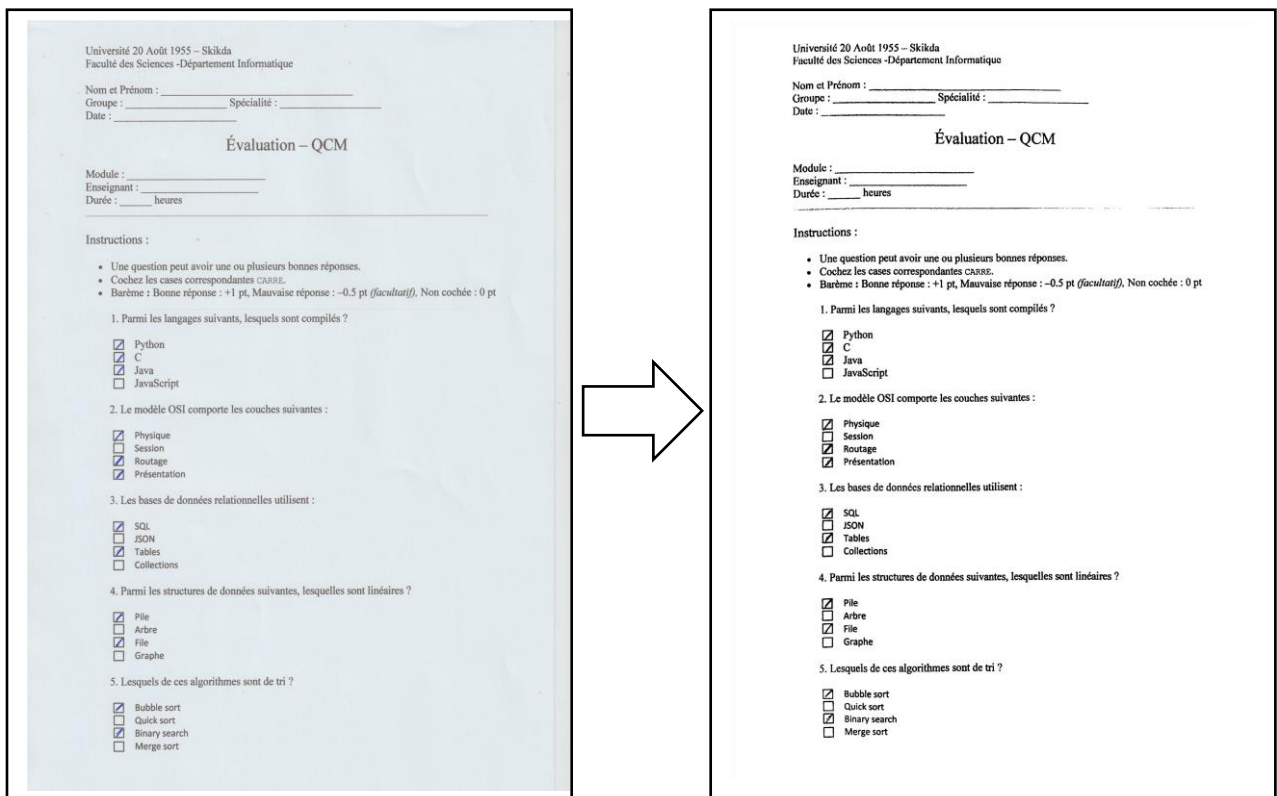


Figure 36 : Un exemple de la binarisation d'une copie d'examen.

- Détection des cases

Malgré qu'on puisse détecter les cases en se basant sur l'information de position des cases à cocher qui sont connues à l'avance, car elles sont définies lors de la confection du sujet, on a proposé une solution générique qui peut détecter les cases à cocher même pour les sujets d'examens non-confectionnés au niveau de notre système.

Dans le cas où le sujet est confectionné au niveau de notre application, notre futur système procède à effectuer les étapes suivantes :

- Chaque case est associée à des coordonnées précises (X, Y, largeur, hauteur) sur la copie scannée ainsi que l'ordre de séquençement par rapport aux autres cases.
- Ces coordonnées sont stockées dans un fichier de configuration ou en base de données lors de la génération du sujet.
- Lors du traitement d'une copie, le système extrait l'image binarisée et se réfère à ces coordonnées pour localiser chaque case à analyser.

Dans le cas échéant (le sujet d'examen n'est pas confectionné au niveau de notre application), notre futur système procède à appliquer le pseudo code suivant :

Entrée :

- Image binaire issue de l'étape précédente.
- Modèle de sujet généré par le système, qui définit :
 - Le nombre de questions.
 - Le nombre de propositions par question.
 - La séquence de passage des cases (par ex. de haut en bas et de gauche à droite).

Sortie :

- Une liste ordonnée des cases détectées et classées selon leur séquence dans le document.

Traitement :

Extraction de contours et détection des formes :

- Utilisation de techniques de détection de contours (OpenCV findContours).
- Filtrage des formes détectées pour ne conserver que celles correspondant aux dimensions approximatives des cases (ex : carrés ou rectangles d'une taille donnée).

Classement séquentiel :

- Les cases sont triées en fonction de leur **ordre de passage dans le document**, et non de leurs coordonnées.
- Par exemple :
 - Les cases sont analysées ligne par ligne ou colonne par colonne.
 - La première case détectée est affectée à **Question 1 / Proposition A**, la deuxième à **Question 1 / Proposition B**, etc.
- Ce classement séquentiel repose sur la structure de sujet mémorisée et la disposition fixe du modèle de copie.

Stockage des positions relatives :

- Pour chaque case, on mémorise uniquement son **rang dans la séquence globale** (ex : case n°5 = QCM question 2, proposition A), sans tenir compte des coordonnées X, Y.

- Détection des cases cochées

L'objectif de cette étape est de déterminer l'état de chaque case détectée (cochée ou non cochée) à partir de l'image binaire, en s'appuyant sur la densité de pixels noirs à l'intérieur de chaque zone réservée à une case.

Contrairement aux méthodes dynamiques qui recherchent des formes dans l'image à chaque exécution, ici le système repose sur une approche OMR (Optical Mark Recognition) basée sur l'analyse de zones bien définies et ordonnées selon leur séquence dans le modèle de sujet.

Le pseudo code adopté est comme suit :

Extraction des sous-images des cases :

- Le système récupère pour chaque case détectée une **sous-image** correspondant à la zone qu'elle occupe.
- Cette extraction s'appuie sur la séquence ordonnée des cases détectées à l'étape précédente.
- Une légère marge **ou padding** peut être appliqué autour de la sous-image pour compenser de petites imprécisions de numérisation.

Calcul de la densité de pixels noirs :

- Pour chaque sous-image, le système calcule le rapport entre le nombre de pixels noirs et le nombre total de pixels de la sous-image.
- Si ce **taux de noircissement** dépasse un seuil préconfiguré (ex : 40 %), la case est considérée comme cochée.

Association automatique avec la question et la réponse :

Chaque case, déjà positionnée dans une séquence précise, est liée directement à :

- Un numéro de question.
- Une proposition de réponse (A, B, C...).
- Un score éventuel attribué selon le barème spécifié dans le fichier de configuration.

Avantages de cette méthode :

- **Pas de recalcul de positions dynamiques** : Le système se base sur la séquence connue des cases issues de l'étape de détection.
- **Rapidité et fiabilité élevée** grâce à une simple analyse de densité binaire.

- **Tolérance aux décalages légers** car l'analyse se fait localement sur des sous-images bien définies.

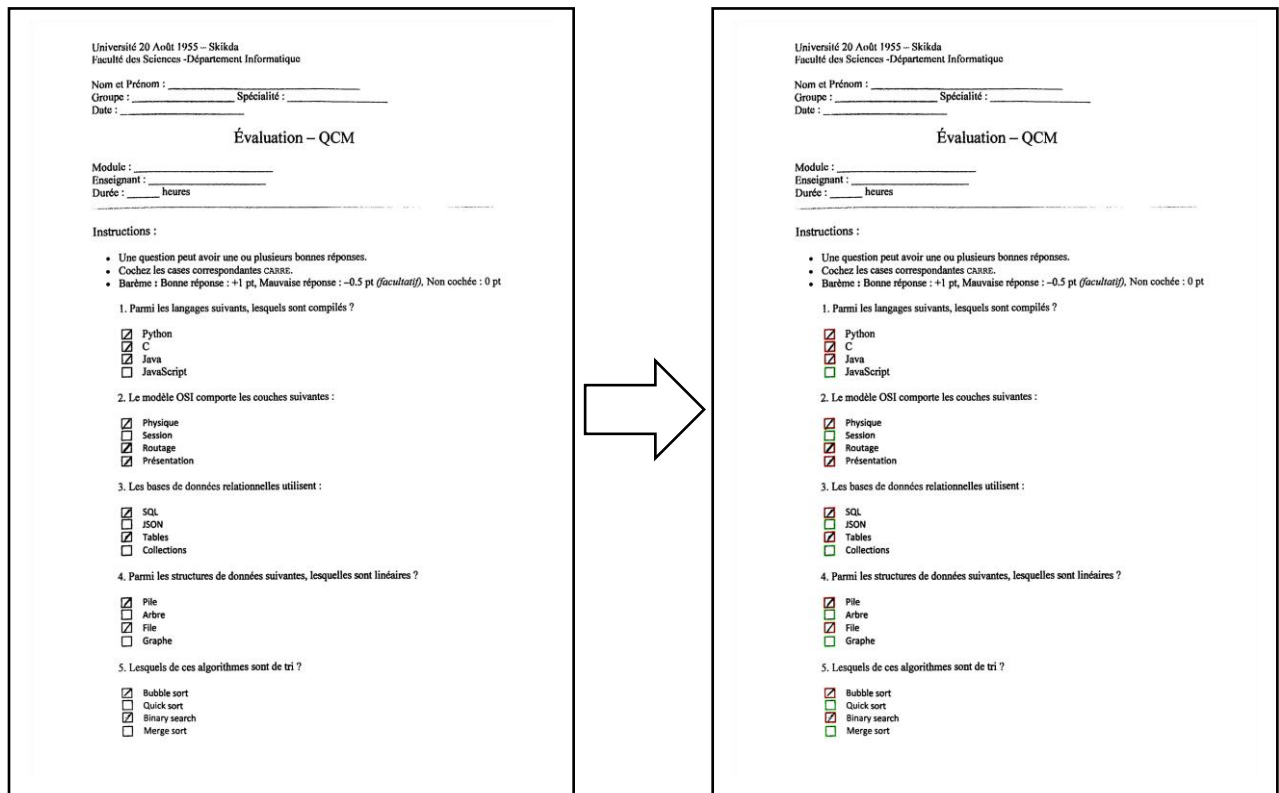


Figure 37 : Un exemple de détection des cases et détection des cases à cocher.

- Attribution des points (scores)

Pour chaque question, Si la case cochée correspond à la bonne réponse, un score est attribué. Des règles de notation spécifiques peuvent être appliquées (points négatifs pour mauvaise réponse, absence de réponse...).

Les réponses de l'exemple donné à la figure ci-dessous sont complètement conformes avec celles données en corrigé type ce qui permet à l'étudiant d'avoir la note toute entière.

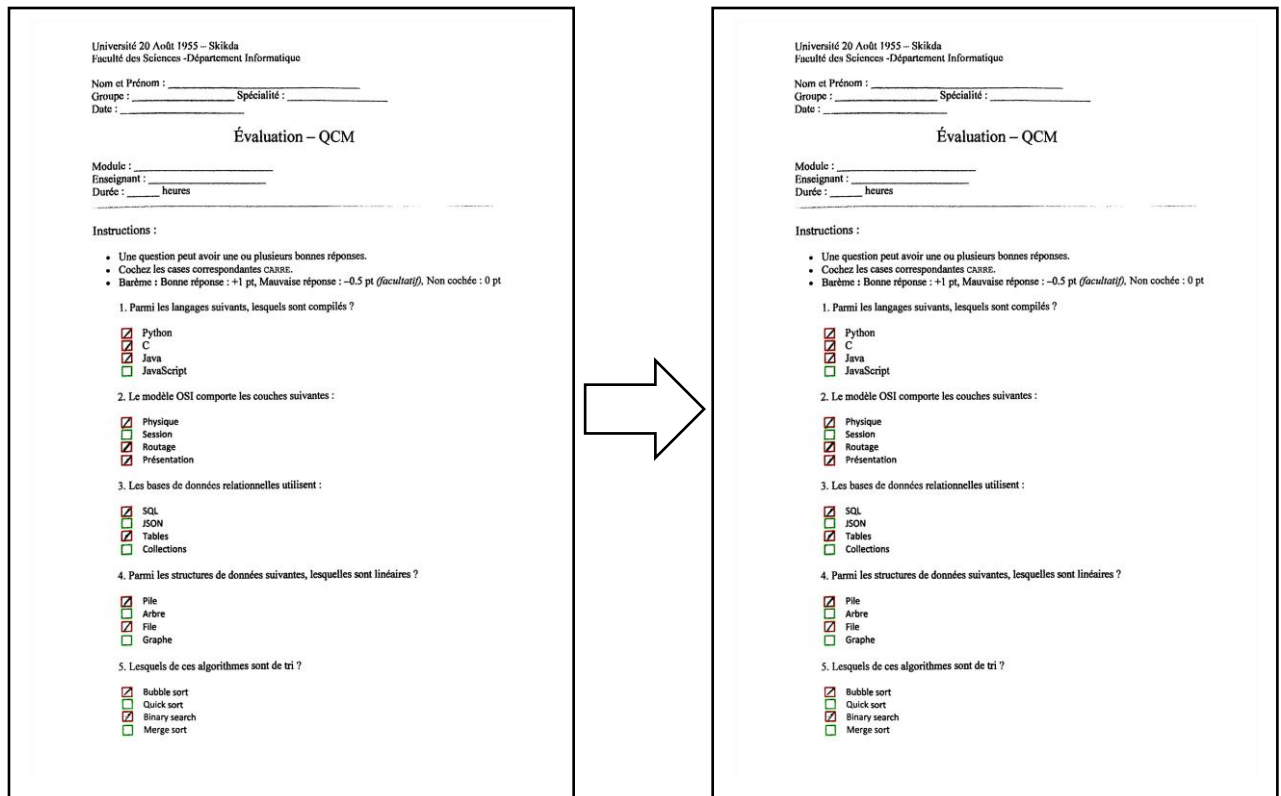


Figure 38 : Un exemple d'attribution de points où les réponses de l'étudiants sont correctes.

- Calcul de la note globale

La note finale d'une copie associée à un candidat est calculée en additionnant les scores de chaque question selon le barème défini.

Résultats de la simulation:

- Questions correctes: [1, 2, 5]
- Questions fausses: [3, 4]
- Note brute: 3,0/5,0 point(s)
- Note sur 20: 12,0/20

Détail des points:

- Q1: 1,0 point(s) - 4 option(s)
- Q2: 1,0 point(s) - 4 option(s)
- Q3: 0,0 point(s) - 4 option(s)
- Q4: 0,0 point(s) - 4 option(s)
- Q5: 1,0 point(s) - 4 option(s)

Figure 39 : Un exemple sur le calcul de la note globale.

- Détection des zones d'identification du candidat (segmentation)

Le système détecte les zones contenant : le nom, le prénom, et le groupe. Cette segmentation repose sur les repères du formulaire et des coordonnées prédéfinies. Chaque zone est extraite sous forme d'image.

- Reconnaissance des caractères du nom et reconstitution

La zone extraite est soumise à un système de OCR (Optical Character Recognition). Deux approches possibles :

- Tesseract OCR : reconnu pour la lecture de textes imprimés et manuscrits (modèles LSTM).
- Classifier basé sur SVM/KNN avec des descripteurs (HOG ou pixel features) pour les manuscrits structurés.

Chaque caractère est reconnu individuellement, puis les caractères sont assemblés pour reconstituer le nom complet.

- Reconnaissance des caractères du prénom et reconstitution

Même procédure que pour le nom :

- Segmentation de la zone du prénom.
- Application de l'OCR.
- Reconstruction du prénom complet à partir des caractères reconnus.

- Reconnaissance des chiffres du groupe et reconstitution

La zone contenant le numéro de groupe est extraite et soumise à l'OCR :

- Segmentation des chiffres
- Reconnaissance via un OCR optimisé pour les chiffres
- Reconstitution du numéro complet du groupe.

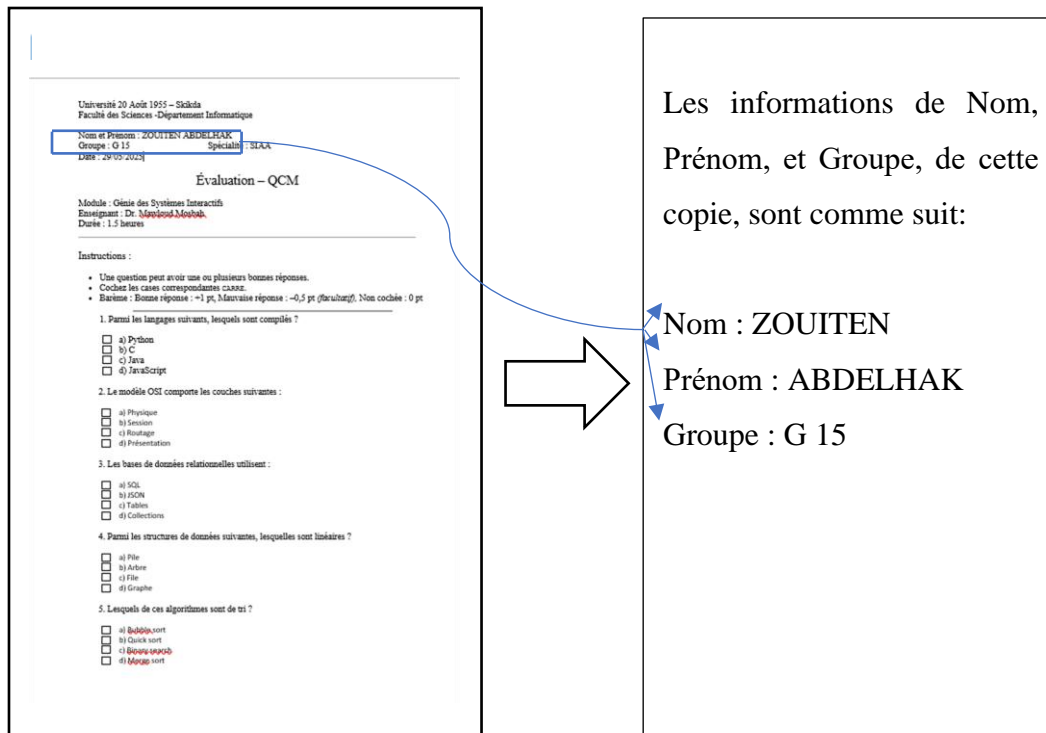


Figure 40 : Exemple de reconnaissance du nom, prénom, et groupe d'une copie d'un étudiant

- Addition de la note globale de l'étudiant dans le fichier Excel

Les données obtenues (nom, prénom, groupe et note globale) sont insérées automatiquement dans un fichier Excel structuré à l'aide de bibliothèques comme Apache POI (*Java*).

	A	B	C	D	E	F	G	H	I	J
1	MOSBAH Mawloud/informatique/Semestre 1/Terminologie scientifique et expression écrite/section 3									
2	Matricule	Nom	Prénom	Note	Absent	Absence Justifiée	Observation	Section	Groupe	
3	232336116103	BELOUHAB/بلوهاب	AHLEM/أحلام	15					section 3/groupe 15	
4	242436011801	BEN HARCHACHE/بن حرشاش	ABDE RRAHMANE/عبد الرحمان	17,25					section 3/groupe 15	
5	242436010520	KOUICEM/قويسم	RAYANE/ريان	17,25					section 3/groupe 15	
6	242436077418	LALLOUCHE/العلوش	CHAHRAZED/شهزاد	14,5					section 3/groupe 15	
7	242436035901	LALLOUI/العلوي	ROUMAÏSSA/رميسة	15					section 3/groupe 15	
8	232336008620	LAMAMRI/لعمامري	IYAD NAIM/إياد نعيم	14,75					section 3/groupe 15	
9	242436044509	LAOUAR/لعور	SELSEBILE/سلسبيل	18					section 3/groupe 15	
10	242436081311	LARKEM/لرقم	DIOUHAINA/جهينة	17,75					section 3/groupe 15	
11	242436011118	LARNEK/لرنق	SIHEM/سهام	13,5					section 3/groupe 15	
12	242436035401	LAZRI/العزري	ADEM/آدم	15,25					section 3/groupe 15	
13	232436121517	LEKHCHINE/لخشين	NOUR ELYAKINE/نور البقين	16,5					section 3/groupe 15	
14	232436122812	LEKHLIFI/لخايفي	Wissem/وسام	17,5					section 3/groupe 15	
15	242436008616	LEMMOUCHE/لموشي	ISLAM/إسلام	16,25					section 3/groupe 15	
16	232436107720	LESSAK/لساق	AMANI/أمانى	13,75					section 3/groupe 15	
17	242436074612	LOKCHIRI/لقشيري	RAMA/راما	16,5					section 3/groupe 15	
18	212436121616	LOKRAI/لقريبي	NOR HANE/نورهان	15,25					section 3/groupe 15	
19	242436034516	LOURACI/لوراسي	NOUARA/نواراة	13,5					section 3/groupe 15	
20	242436081502	LYZIDI/ليزدي	FATEN/فاتن	14,5					section 3/groupe 15	
21	242436027318	MAATALLAH/معطل الله	MOHAMED ELHACHMI/محمد الهاشمي	16,25					section 3/groupe 15	
22	242436025308	MABROUK/مابروك	AYA/آية	14					section 3/groupe 15	

Figure 41 : Un exemple d'un fichier EXCEL crée et rempli.

- Validation des données

Le système vérifie :

- La cohérence des noms et prénoms (absence de caractères non valides).
- La validité des notes (dans l'intervalle possible).
- L'absence de doublons ou de champs vides.

- Ajout dans le fichier Excel

Les données validées sont définitivement enregistrées dans le fichier Excel.

- Enregistrement des erreurs éventuelles

Toutes les erreurs rencontrées (image non lisible, problème de détection, OCR incorrect, données incohérentes) sont consignées dans un journal d'erreurs sous forme de fichier texte ou table Excel.

- Génération d'un rapport de correction

Le système compile un rapport global qui comprend :

- La liste des étudiants avec leurs notes.

- Les statistiques de réussite (taux de réussite, moyenne, écart-type...).
- La liste des erreurs rencontrées.
- Les copies rejetées et leurs motifs.

Ce rapport est généré en Excel ou en PDF.

IV. Conception

Dans le cadre de la conception de ce système, il a été nécessaire de modéliser à la fois la structure statique des entités qui le composent, ainsi que les interactions dynamiques entre ces différentes entités au fil du traitement d'une copie.

Pour cela, deux types de diagrammes ont été utilisés :

(1) Le diagramme de classes : Qui permet de représenter de manière statique les différentes classes et objets manipulés dans le système, ainsi que leurs relations. Ce diagramme met en évidence les entités telles que *QCM*, Copie, Étudiant, Question, Correction et Note, et précise les associations et dépendances qui les relie (2) Le diagramme de séquence : qui illustre de façon dynamique et temporelle le déroulement des interactions entre les acteurs (enseignant, étudiant, système) et les objets du système, depuis la création du QCM jusqu'à la correction automatique et l'attribution de la note. Ce diagramme permet de visualiser l'enchaînement des messages échangés et des traitements réalisés au cours du processus.

Ces diagrammes contribuent à clarifier l'architecture et le fonctionnement du système, en facilitant sa compréhension, son développement et sa maintenance.

Dans cette section, nous présentons deux diagrammes, un est statique qui est le diagramme de classes et l'autre est dynamique qui est le diagramme de séquences.

1. Le diagramme de classes

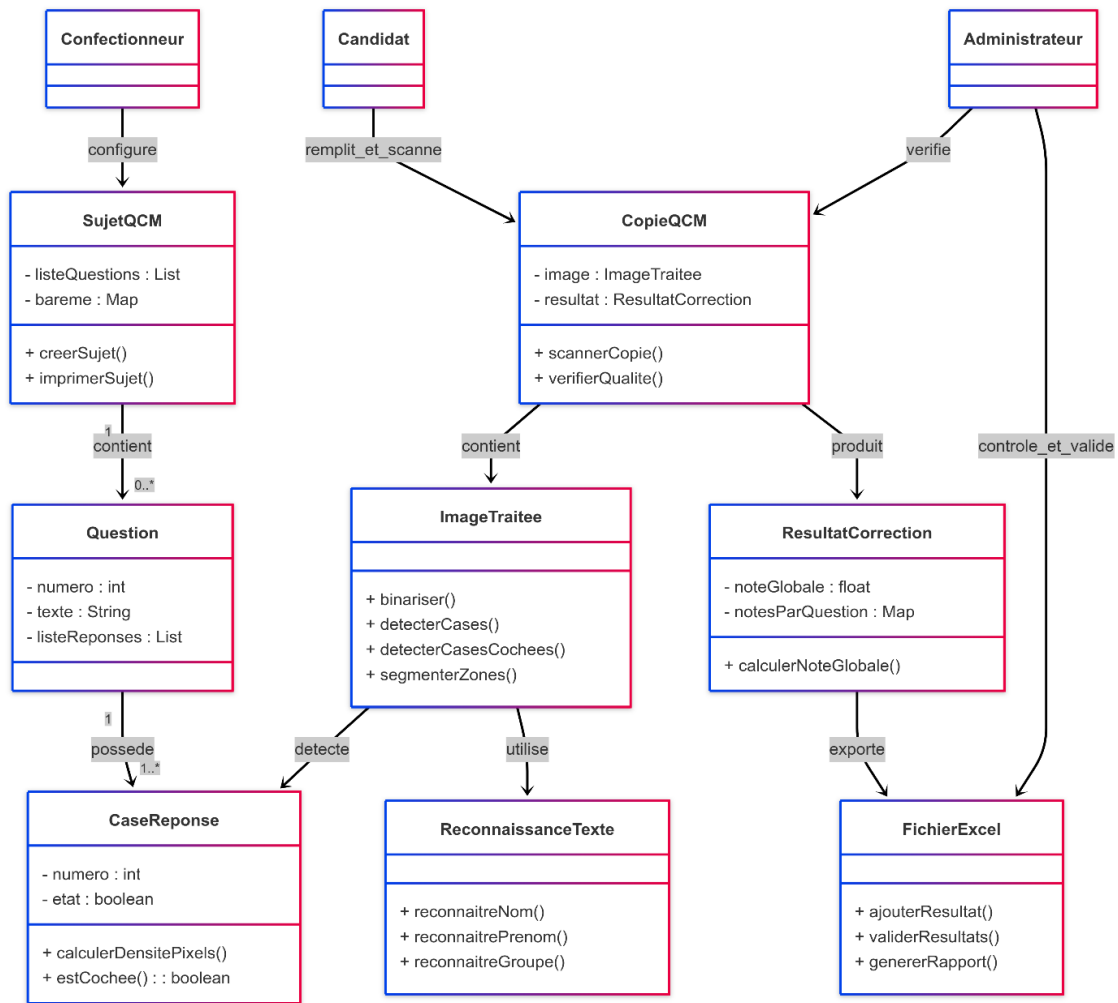


Figure 42 : Diagramme de classe

2. Le diagramme de séquences

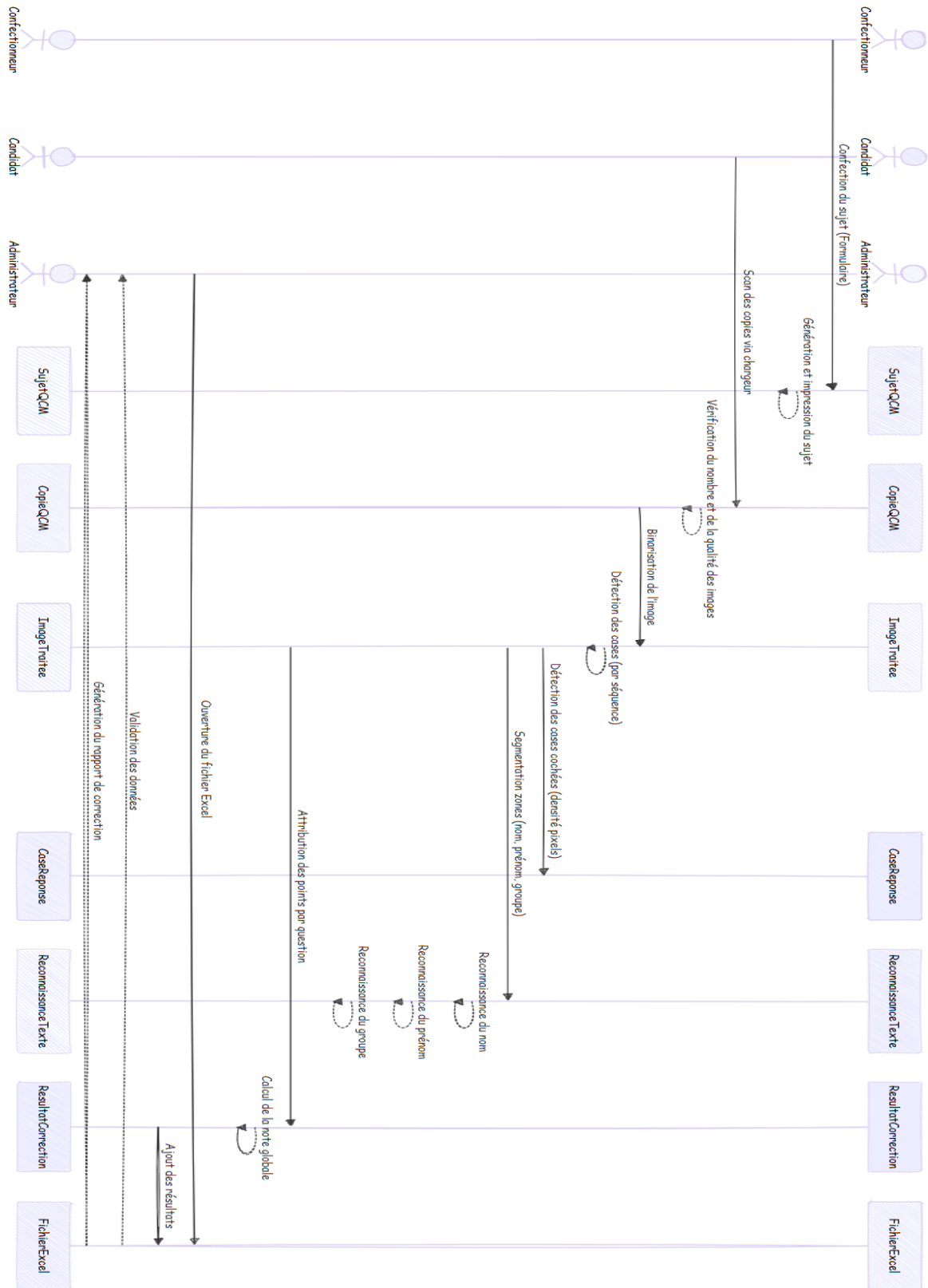


Figure 43 : Diagramme de séquences

V. Conclusion

Dans ce chapitre d'analyse et de conception, nous avons procédé à la définition et à la modélisation de l'architecture générale de notre système SCAEQCM. L'objectif principal était de concevoir un environnement intégré permettant à la fois la création assistée des sujets d'examen, la correction automatique des QCMs pour savoir quelles sont les cases bien cochées et quelles sont qui ne sont pas, collecte automatisée des notes pour avoir la note globale de chaque copie ainsi que l'identification automatique des coordonnées de chaque copie comme le nom, le prénom, et le groupe, pour atteindre la phase finale celle de la création et le remplissage d'un fichier EXCEL importable à la plateforme Progress.

Chapitre 04 : Implémentation

I. Introduction

Dans ce chapitre, nous présentons la mise en œuvre de notre système. En commençant par présenter les outils utilisés ainsi que quelques captures d'écrans.

II. Environnement de développement

Tout au long de l'implémentation de notre application, nous avons utilisé Java comme langage de programmation ainsi que quelques bibliothèques comme :

- `itext7-core-7.2.1.jar` (pour la création et la manipulation de fichiers PDF).
- `Tess4j-3.4.8.jar` (pour exploiter les fonctions OCR de Tesseract OCR).
- `Opencv-4.9.0.jar` (pour exploiter les fonctions OMR).
- `Poi-bin-5.2.3.jar` (pour créer et manipuler un fichier EXCEL).
- `JTwain.jar` (pour des fonctions de scan).

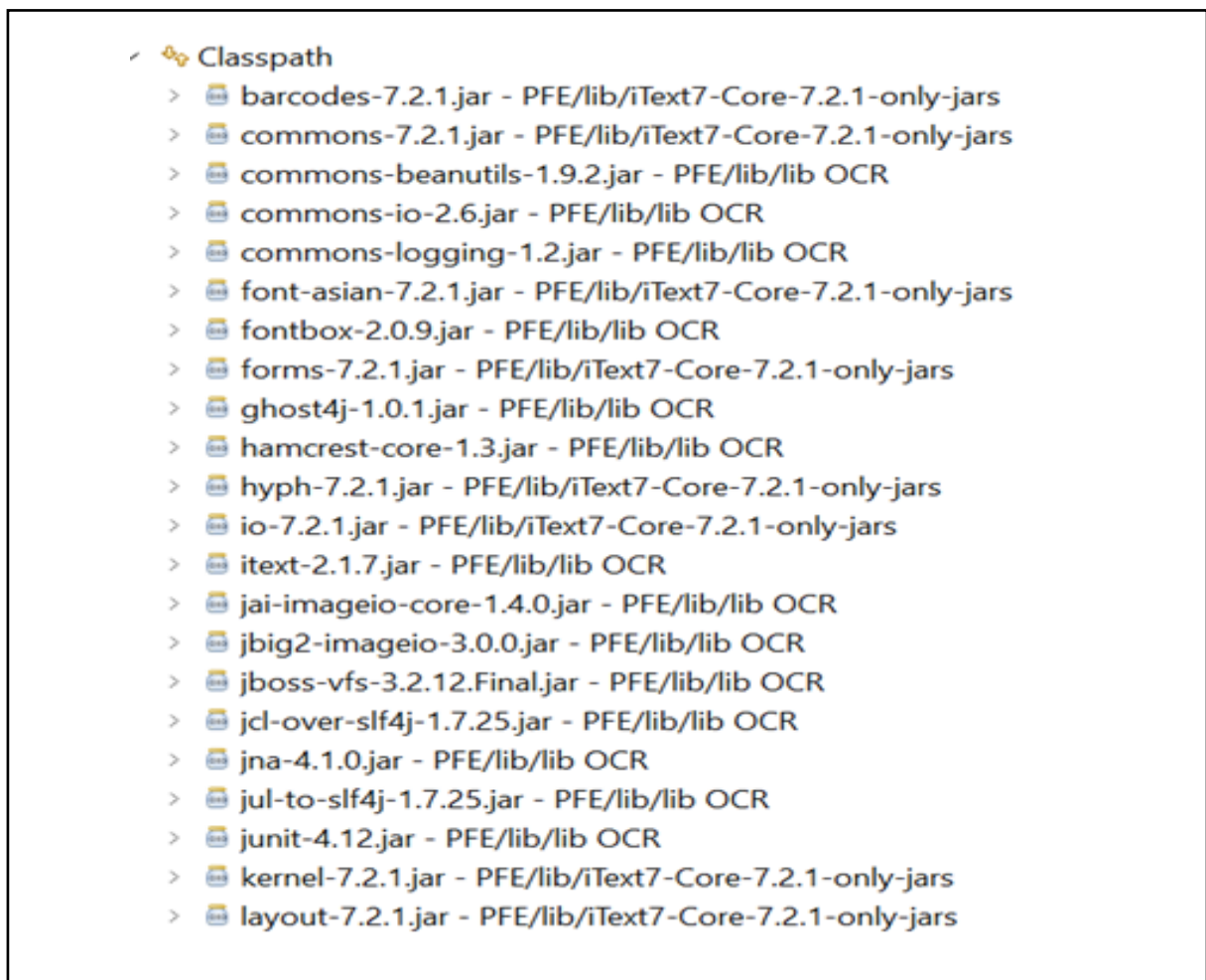


Figure 44 : Les différentes bibliothèques Java utilisées.

Pour l'éditeur dédié, on a opté pour Eclipse.

1. Java

Java est un langage de programmation polyvalent et orienté objet, ainsi qu'une plateforme de calcul, initialement publié par Sun Microsystems en 1995 et aujourd'hui détenu par Oracle Corporation. Il est réputé pour sa portabilité, ses performances et son adoption massive dans divers types d'applications, allant des serveurs d'entreprise aux appareils mobiles. Parmi les principales caractéristiques de Java, on peut citer :

- **Orienté objet** : Java repose sur les principes de la programmation orientée objet, ce qui facilite l'organisation et la structuration de systèmes logiciels complexes à l'aide d'objets et de classes.
- **Indépendance vis-à-vis de la plateforme** : Java assure l'indépendance plateforme grâce à l'utilisation de bytecode. Les programmes Java sont compilés en bytecode, qui peut être exécuté sur n'importe quel appareil équipé d'une machine virtuelle Java (JVM), permettant ainsi la philosophie "écrire une fois, exécuter partout".
- **Robuste et sécurisé** : Java intègre des fonctionnalités garantissant des applications fiables et sécurisées, telles qu'une gestion rigoureuse de la mémoire, un système de gestion des exceptions et un gestionnaire de sécurité qui définit des règles d'accès pour les classes.
- **Syntaxe simple et familière** : La syntaxe de Java est conçue pour être facile à comprendre et similaire à celle du C et du C++, ce qui permet aux développeurs d'apprendre et d'adopter rapidement le langage.
- **Multithreading** : Java prend en charge le multithreading, permettant aux développeurs de créer des applications capables d'exécuter plusieurs tâches simultanément, améliorant ainsi les performances des applications complexes.
- **Gestion automatique de la mémoire** : Java utilise un mécanisme de garbage collection (ramasse-miettes) automatique pour gérer la mémoire, réduisant ainsi les risques de fuites mémoire et d'autres problèmes associés.
- **Bibliothèque standard étendue** : Java dispose d'une vaste bibliothèque standard comprenant un large éventail de classes et d'interfaces pour diverses tâches telles que les structures de données, le réseau, les entrées/sorties de fichiers, le développement d'interfaces graphiques (GUI), et bien plus encore.
- **Dynamique et extensible** : Java prend en charge le chargement dynamique de classes, permettant l'ajout de nouveau code à l'exécution. Sa nature modulaire et la disponibilité

de nombreuses bibliothèques et frameworks tiers en font un langage hautement extensible.

- **Haute performance** : Le compilateur Just-In-Time (JIT) de Java optimise le bytecode en code machine natif lors de l'exécution, améliorant ainsi les performances des applications Java.



Figure 45 : Notre langage de programmation utilisé.

2. Eclipse

Eclipse est un **environnement de développement intégré (IDE)** open source, principalement utilisé pour le développement en **Java**, mais extensible à d'autres langages via des plugins. Créé à l'origine par IBM en 2001, il est aujourd'hui maintenu par l'**Eclipse Foundation**. Son architecture modulaire et sa grande communauté en font un outil incontournable pour les développeurs.



Figure 46 : Le logo de l'éditeur dédié au Java, Eclipse.

III. Quelques captures d'écran de notre application

Dans cette section, nous présentons les différentes captures d'écran de notre prototype.

La **Figure** illustre l'interface d'accueil de notre prototype. Comme illustré à la figure, les différents boutons, permettant d'accéder aux différentes fonctionnalités du système, sont trouvés à gauche. A la droite de la fenêtre, le système affiche les différentes interfaces graphiques liées aux différentes fonctionnalités une fois un bouton est cliqué.

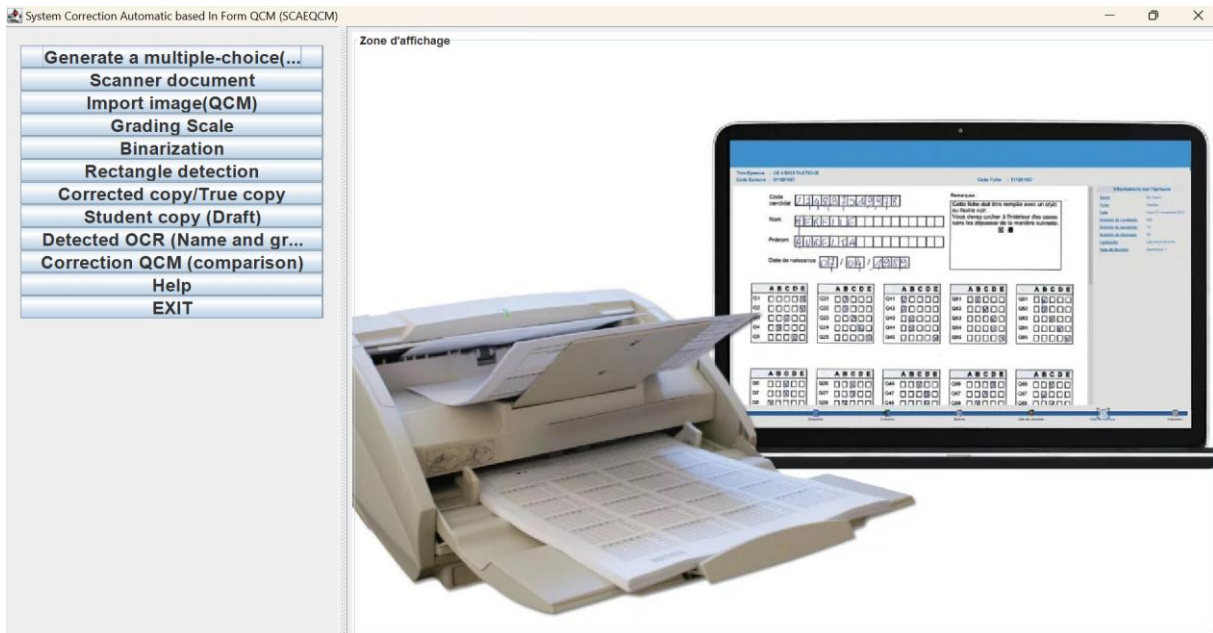


Figure 47 : Interface d'accueil de notre prototype.

La figure ci-dessous présente un formulaire visualisé par notre application pour être rempli par l'utilisateur comme une façon pour confectionner le sujet d'examen en version PDFs et sous forme de QCMs.

Titre du QCM:

Enseignant:

Question 1
Parmi les langages suivants, lesquels sont compilés ?

Option 1:

Option 2:

Option 3:

Option 4:

Question 2
Le modèle OSI comporte les couches suivantes :

Option 1:

Option 2:

Option 3:

Option 4:

Question 3
Les bases de données relationnelles utilisent :

Option 1:

Figure 48 : Un formulaire affiché par notre application pour confectionner le sujet.

La figure ci-dessous présente une interface que le système visualise à l'utilisateur pour désigner le barème.

Gestion de QCM - Questions Individuelles

Configuration Simulation

Paramètres généraux

Appliquer une pénalité:

Malus par réponse fausse:

Seuil de réussite (%):

Modifier Ajouter Question Supprimer Question

Question	Points	Options
Question 1	1.0	4
Question 2	1.0	4
Question 3	1.0	4
Question 4	1.0	4
Question 5	1.0	4

Sauvegarder Charger

Figure 49 : Une interface graphique pour désigner le barème du sujet d'examen.

La figure ci-dessous présente une interface permettant à l'utilisateur de scanner et de charger le fichier scanné pour lancer plus tard la phase de correction automatique.

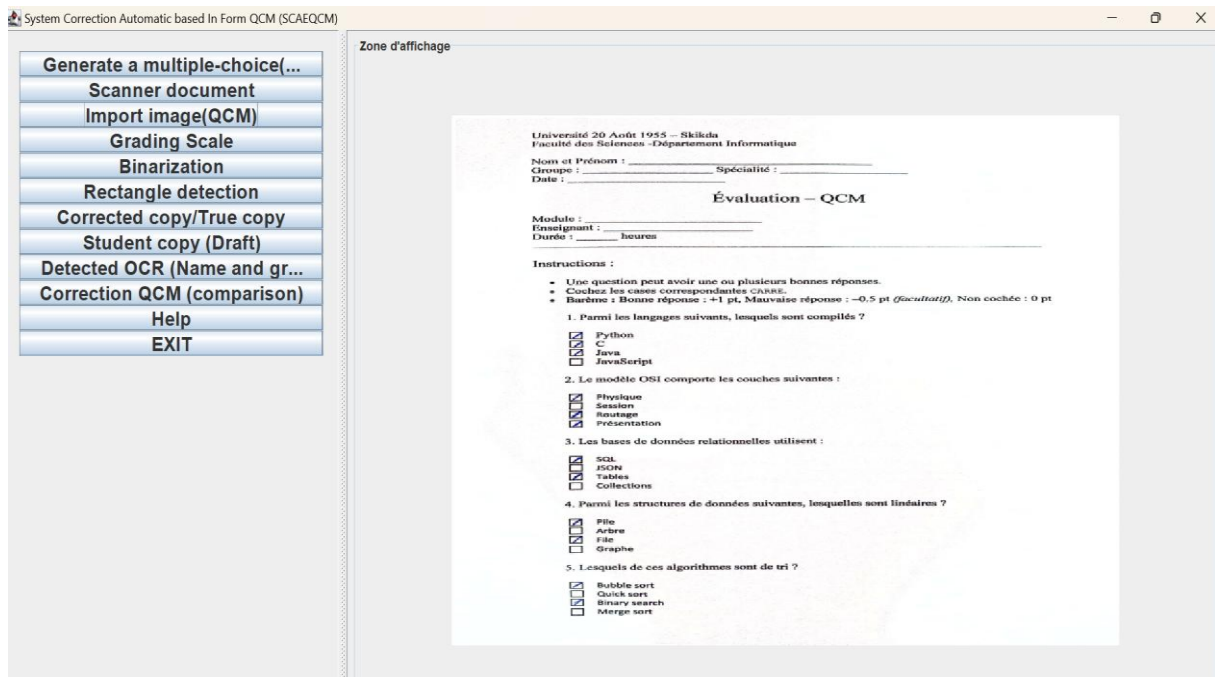


Figure 50 : Une interface graphique pour scanner et charger une copie d'examen.

La figure ci-dessous illustre une interface permettant de binariser une copie d'examen.

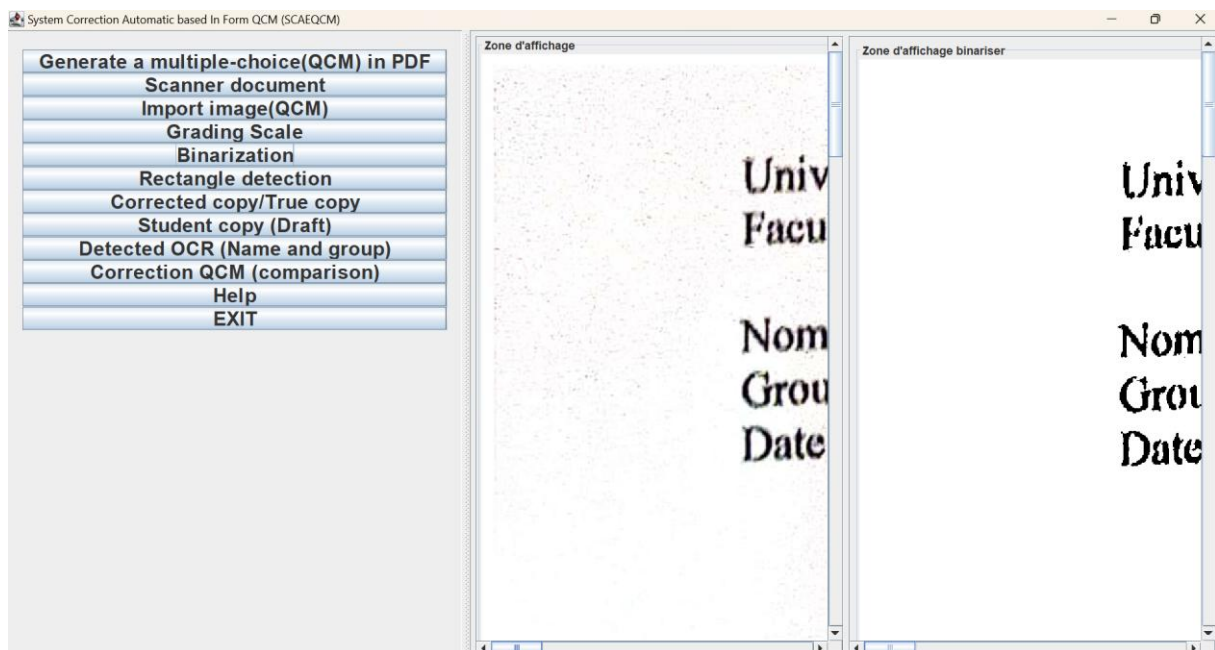


Figure 51 : Une interface pour binariser une copie d'examen.

La figure ci-dessous présente une interface qui montre la fonctionnalité de détection par le système des cases et des cases à cocher (les cases non cochées sont colorées en vert alors que les cases cochées sont colorées en rouge).

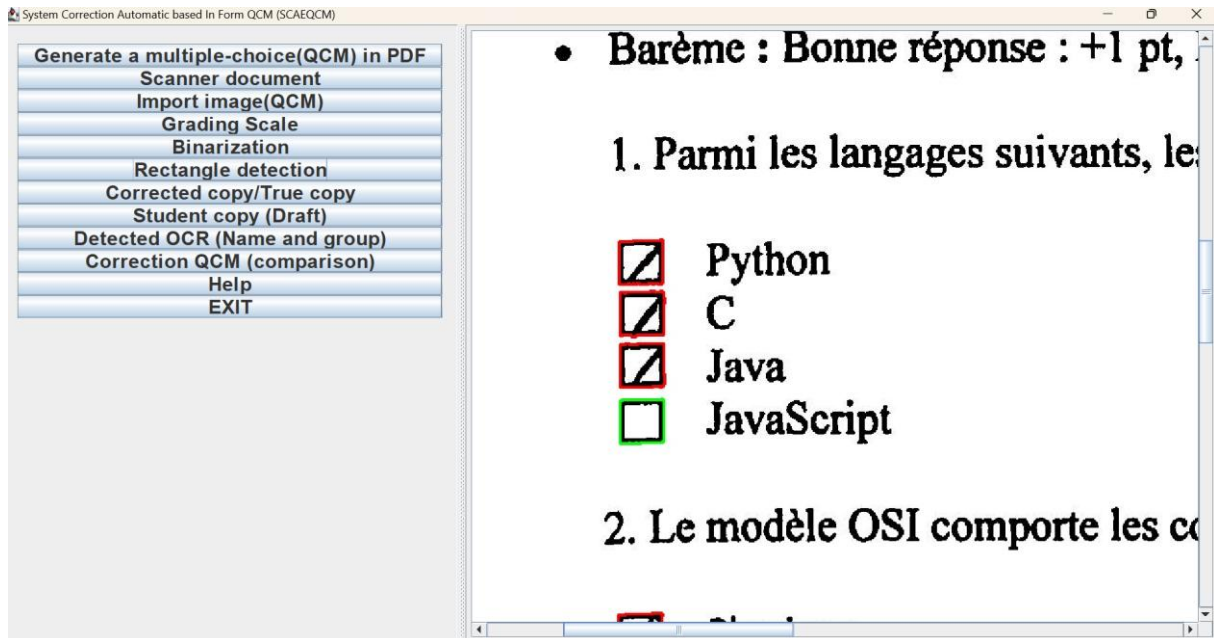


Figure 52 : Interface montrant la détection par le système des cases et des cases cochées.

IV. Conclusion

Dans ce chapitre d'implémentation, nous avons bien présenté les outils que nous avons utilisé pour réaliser notre prototype. De plus, quelques captures d'écran ont été bien présentées.

Conclusion Générale

I. Conclusion Générale

Dans le cadre de ce mémoire, nous avons entrepris le développement d'un système de correction automatique de questionnaires à choix multiples (QCM) à base de cases à cocher, répondant aux exigences croissantes des établissements d'enseignement en matière de gestion efficace et fiable des évaluations. L'idée de ce projet est née face à la problématique récurrente rencontrée par les enseignants et les établissements, à savoir la lourdeur de la correction manuelle des examens, la perte de temps et les erreurs humaines fréquentes dans les grandes sessions d'évaluation.

Dans un premier temps, un état de l'art approfondi a été réalisé, permettant de situer ce projet dans son contexte académique et technologique. Nous avons ainsi défini le concept du QCM, ses types variés (réponse unique, réponses multiples, QCM à pondération, à correction négative), ses avantages pédagogiques (simplicité, rapidité de correction, objectivité) et ses limites, notamment en matière de formulation des questions et de gestion des cas ambigus. Nous avons ensuite étudié les méthodes classiques de correction de QCM : manuelle, semi-automatique via des grilles et automatique grâce aux techniques de reconnaissance optique de marques (OMR). Ces systèmes, bien que performants dans certains cas, restent souvent coûteux, rigides et sensibles à la qualité du support papier et du remplissage par les candidats. La deuxième partie du mémoire a été consacrée à la conception et à la réalisation du système proposé. Une architecture générale a été définie, intégrant plusieurs modules :

Un assistant de création de QCM, permettant à l'examineur de concevoir et de générer des feuilles d'examen structurées avec des cases à cocher adaptées.

Un module de numérisation, chargé de la récupération des copies scannées.

Un module de traitement d'images, utilisant la bibliothèque OpenCV pour la détection et la segmentation des cases cochées.

Un module de reconnaissance optique via Tesseract OCR pour le traitement des éventuelles annotations textuelles.

Enfin, un module de correction et de calcul des résultats, basé sur un algorithme de comparaison entre les cases cochées et le corrigé type fourni à l'avance.

La modélisation du système a été formalisée grâce aux diagrammes UML, comprenant :

Le diagramme de cas d'utilisation qui décrit les interactions entre les utilisateurs (examineur, étudiant, système).

Les diagrammes de classes et de séquences qui définissent l'organisation des objets et l'enchaînement des opérations.

Les diagrammes d'activités pour illustrer le déroulement des processus et un diagramme de déploiement pour situer les composants matériels et logiciels dans l'environnement opérationnel.

Pour la réalisation pratique, nous avons opté pour le langage Java sous l'environnement de développement Eclipse, en intégrant la bibliothèque OpenCV pour la partie traitement d'image et Tesseract OCR pour la reconnaissance des textes et des marques. Le système a été déployé sur un environnement Windows 11, utilisant un matériel standard composé d'un PC portable et d'une imprimante pour la génération et l'impression des feuilles d'examen.

La phase de test et validation a permis de mesurer l'efficacité du système sur des cas réels et simulés. Les résultats ont démontré une reconnaissance correcte des réponses, un gain de temps considérable et une diminution significative des erreurs de correction. Toutefois, des difficultés ont été observées en cas de mauvaise qualité d'impression ou de scannage, ou lorsque les étudiants remplissent mal les cases prévues.

Ce projet apporte donc plusieurs contributions :

Un outil semi-automatique accessible et économique pour les établissements n'ayant pas les moyens d'investir dans des scanners OMR industriels.

La possibilité pour les enseignants de concevoir facilement leurs examens et d'automatiser leur correction.

Un système adaptable, pouvant évoluer vers des fonctionnalités supplémentaires.

Quelques axes de perfectionnement peuvent être envisagés :

L'intégration d'un module d'analyse statistique pour générer automatiquement des indicateurs de performance.

Le développement d'une interface web ou mobile pour gérer à distance la création des QCM et le suivi des corrections.

L'ajout de techniques d'apprentissage automatique (machine learning) pour la reconnaissance intelligente des manuscrits et des cas ambigus.

L'intégration du support multilingue pour répondre aux besoins des établissements internationaux.

En somme, ce mémoire a permis de mettre en œuvre un système fonctionnel répondant à un besoin concret dans le milieu éducatif. La démarche adoptée, alliant étude théorique, conception méthodologique et implémentation technique, constitue une base solide pour de futures extensions. Ce travail ouvre des perspectives intéressantes pour l'optimisation des systèmes

d'évaluation dans les établissements d'enseignement et représente un pas supplémentaire vers la digitalisation intelligente du processus éducatif.

Références

Références

- [1] Jia, F., Shi, C., He, K., Wang, C., & Xiao, B. (2018). Degraded document image binarization using structural symmetry of strokes. *Pattern Recognition*, 74, 225-240. Wu, Y., Natarajan, P., Rawls, S., & AbdAlmageed, W. (2016, September). Learning document image inarization from data. In 2006 IEEE International Conference on Image Processing (ICIP) (pp. 3763-3767). IEEE.
- [2] Bernsen, J. (1986). Dynamic Thresholding of Grey-Level Images, [in:] Proc. of the 8th Int. In Conf. on attern Recognition.
- [3] Sauvola, J., & Pietikäinen, M. (2000). Adaptive document image binarization. *Pattern recognition*, 33(2), 225-236.
- [4] Kim, I. J. (2004, October). Multi-window binarization of camera image for document recognition. In Ninth International Workshop on Frontiers in Handwriting Recognition (pp. 323-327). IEEE.
- [5] Jia, F., Shi, C., He, K., Wang, C., & Xiao, B. (2018). Degraded document image binarization using structural symmetry of strokes. *Pattern Recognition*, 74, 225-240.
- [6] Ntirogiannis, K., Gatos, B., & Pratikakis, I. (2012). Performance evaluation methodology for historical document image binarization. *IEEE Transactions on Image Processing*, 22(2), 595-609.
- [7] Ntirogiannis, K., Gatos, B., & Pratikakis, I. (2012). Performance evaluation methodology fo historical document image binarization. *IEEE Transactions on Image Processing*, 22(2), 595-609.
- [8] Lu, H., Kot, A. C., & Shi, Y. Q. (2004). Distance-reciprocal distortion measure for binary document images. *IEEE Signal Processing Letters*, 11(2), 228-231.
- [9] Bhowmik, S. (2023) 'Document image binarization', *Document Layout Analysis*, pp.11–30, Springer Nature Singapore, Singapore.
- [10] Zhou, Y., Zuo, S., Yang, Z., He, Z., He, J., Shi, J. and Zhang, R. (2023) 'A review of document image enhancement based on document degradation problem', *Applied Sciences*, Vol. 13, No. 13, pp.1–23.
- [11] Sehadi, A., Chibani, Y., Hedjam, R. and Cheriet, M. (2019) 'Gabor filter-based texture for ancient degraded document image binarization', *Pattern Analysis and Applications*, Vol. 22, No. 1, pp.1–22.
- [12] Krupinski, R., Lech, P., Teclaw, M. and Okarma, K. (2019) 'Binarization of degraded document images with generalized Gaussian distribution', *Computational Science-ICCS 2019: 19th International Conference, Proceedings, Part V, 12–14 June, Springer International Publishing, Faro, Portugal*, pp.177–190.
- [13] Krupinski, R., Lech, P., Teclaw, M. and Okarma, K. (2019) 'Binarization of degraded document images with generalized Gaussian distribution', *Computational Science-ICCS 2019: 19th International Conference, Proceedings, Part V, 12–14 June, Springer International Publishing, Faro, Portugal*, pp.177–190.
- [14] Gatos, B., Pratikakis, I. and Perantonis, S.J. (2006) 'Adaptive degraded document image binarization', *Patter Recognition*, Vol. 39, No. 3, pp.317–327.
- [15] Jain, A.K. (1989) *Fundamentals of Digital Image Processing*, Prentice-Hall, Inc.
- [16] Gatos, B., Pratikakis, I. and Perantonis, S.J. (2006) 'Adaptive degraded document image binarization', *Patter Recognition*, Vol. 39, No. 3, pp.317–327.
- [17] Biswas, B., Bhattacharya, U. and Chaudhuri, B.B. (2014) 'A global-to-local approach to binarization of degraded document images', *2014 22nd International Conference on Pattern Recognition, August, IEEE*, pp.3008–3013.
- [18] Michalak, H. and Okarma, K. (2019) 'Improvement of image binarization methods using image preprocessing with local entropy filtering for alphanumeric character recognition purposes', *Entropy*, Vol. 21, No. 6, pp.1–18.
- [19] Nafchi, H.Z., Moghaddam, R.F. and Cheriet, M. (2013) 'Historical document binarization based on phase information of images', *Computer Vision-ACCV 2012 Workshops: ACCV 2012 International Workshops, Revised Selected Papers, Part II, 5–6 November, Springer Berlin Heidelberg, Daejeon, Korea*, pp.1–12.
- [20] Almeida, M., Lines, R.D., Bernadino, R., Jesus, D. and Lima, B. (2018) 'A new binarization algorithm for historical documents', *Journal of Imaging*, Vol. 4, No. 2, pp.1–27.
- [21] Aurich, V. and Weule, J. (1995) 'Non-linear Gaussian filters performing edge preserving diffusion', *Mustereerkennung 1995: Verstehen akustischer und visueller Informationen, Springer Berlin Heidelberg*, pp.538–545.
- [22] Tomasi, C. and Manduchi, R. (1998) 'Bilateral filtering for gray and color images', *Sixth International Conference on Computer Vision, January, IEEE, IEEE Cat. No. 98CH36271*, pp.839–846.
- [23] Rahiche, A. and Cheriet, M. (2019) 'KFBin alman filter-based approach for document image binarization', *Image Analysis and Recognition: 16th International Conference, ICIAR 2019, Proceedings, Part I, 27–29 August, Springer International Publishing, Waterloo, ON, Canada*, pp.150–161.
- [24] Chang, C.I. and Chang, C.I. (2017) 'Discrete-time Kalman filtering for hyperspectral processing', *Real-Time Recursive Hyperspectral Sample and Band Processing: Algorithm Architecture and Implementation*, pp.49–71.

-
- [25] Rahiche, A. and Cheriet, M. (2019) 'KFBin alman filter-based approach for document image binarization', *Image Analysis and Recognition: 16th International Conference, ICIAR 2019, Proceedings, Part I*, 27–29 August, Springer International Publishing, Waterloo, ON, Canada, pp.150–161.
- [26] Chen, Y. and Wang, L. (2017) 'Broken and degraded document images binarization', *Neurocomputing*, Vol. 237, No. 1, pp.272–280.
- [27] He, K., Sun, J. and Tang, X. (2012) 'Guided image filtering', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 35, No. 6, pp.1397–1409.
- [28] Sarkka, S. and Svensson, L. (2023) *Bayesian Filtering and Smoothing*, Vol. 17, Cambridge University Press, England.
- [29] Bera, S.K., Ghosh, S., Bhowmik, S., Sarkar, R. and Nasipuri, M. (2021) 'A non-parametric binarization method based on ensemble of clustering algorithms', *Multimedia Tools and Applications*, Vol. 80, No. 1, pp.7653–7673.
- [30] Su, B., Lu, S. and Tan, C.L. (2010) 'Binarization of historical document images using the local maximum and minimum', *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*, June, pp.159–166.
- [31] Pratikakis, I., Zagoris, K., Kaddas, P. and Gatos, B. (2018) 'ICFHR2018 Competition on handwritten document image binarization contest (H-DIBCO 2018)', *International Conference on Frontiers in Handwriting Recognition*, p.1.
- [32] Wen, J., Li, S. and Sun, J. (2013) 'A new binarization method for non-uniform illuminated document images', *Pattern Recognition*, Vol. 46, No. 6, pp.1670–1690.
- [33] Abd Elfattah, M., Hassanien, A.E., Mostafa, A., Ali, A.F., Amin, K.M. and Mohamed, S. (2015) 'Artificial bee colony optimizer for historical Arabic manuscript images binarization', *2015 11th International Computer Engineering Conference (ICENCO)*, December, IEEE, pp.251–255.
- [34] Krupinski, R., Lech, P., Teclaw, M. and Okarma, K. (2019) 'Binarization of degraded document images with generalized Gaussian distribution', *Computational Science-ICCS 2019: 19th International Conference, Proceedings, Part V*, 12–14 June, Springer International Publishing, Faro, Portugal, pp.177–190.
- [35] Krupinski, R., Lech, P., Teclaw, M. and Okarma, K. (2019) 'Binarization of degraded document images with generalized Gaussian distribution', *Computational Science-ICCS 2019: 19th International Conference, Proceedings, Part V*, 12–14 June, Springer International Publishing, Faro, Portugal, pp.177–190.
- [36] Moghaddam, R.F. and Cheriet, M. (2010) 'A multi-scale framework for adaptive binarization of degraded document images', *Pattern Recognition*, Vol. 43, No. 6, pp.2186–2198.
- [37] Moghaddam, R.F. and Cheriet, M. (2010) 'A multi-scale framework for adaptive binarization of degraded document images', *Pattern Recognition*, Vol. 43, No. 6, pp.2186–2198.
- [38] Kwon, S.H. (2004) 'Threshold selection based on cluster analysis', *Pattern Recognition Letters*, Vol. 25, No. 9, pp.1045–1050.
- [39] Moghaddam, R.F. and Cheriet, M. (2010) 'A multi-scale framework for adaptive binarization of degraded document images', *Pattern Recognition*, Vol. 43, No. 6, pp.2186–2198.
- [40] Papamarkos, N. (2001) 'A technique for fuzzy document binarization', *Proceedings of the 2001 ACM Symposium on Document Engineering*, November, pp.152–156.
- [41] Drira, F. (2006) 'Towards restoring historic documents degraded over time', *Second International Conference on Document Image Analysis for Libraries (DIAL'06)*, April, IEEE, pp.8–357.
- [42] Chaki, N., Shaikh, S.H., Saeed, K., Chaki, N., Shaikh, S.H. and Saeed, K. (2014) *A Comprehensive Survey on Image Binarization Techniques*, pp.5–15, Springer, India.
- [43] Abd Elfattah, M., Hassanien, A.E., Mostafa, A., Ali, A.F., Amin, K.M. and Mohamed, S. (2015) 'Artificial bee colony optimizer for historical Arabic manuscript images binarization', *2015 11th International Computer Engineering Conference (ICENCO)*, December, IEEE, pp.251–255.
- [44] Abd Elfattah, M., Hassanien, A.E., Mostafa, A., Ali, A.F., Amin, K.M. and Mohamed, S. (2015) 'Artificial bee colony optimizer for historical Arabic manuscript images binarization', *2015 11th International Computer Engineering Conference (ICENCO)*, December, IEEE, pp.251–255.
- [45] Mustafa, W.A. and Kader, M.M.M.A. (2018a) 'Binarization of document images: a comprehensive review', *Journal of Physics: Conference Series*, June, IOP Publishing, Vol. 1019, No. 1, pp.1–10.
- [46] Chaki, N., Shaikh, S.H., Saeed, K., Chaki, N., Shaikh, S.H. and Saeed, K. (2014) *A Comprehensive Survey on Image Binarization Techniques*, pp.5–15, Springer, India.
- [47] Bhowmik, S. (2023) 'Document image binarization', *Document Layout Analysis*, pp.11–30, Springer Nature Singapore, Singapore.
- [48] Mustafa, W.A., Aziz, H., Khairunizam, W., Ibrahim, Z., Shahrman, A.B. and Razlan, Z.M. (2018) 'Review of different binarization approaches on degraded document images', *2018 International Conference on Computational Approach in Smart Systems Design and Applications (ICASSDA)*, August, IEEE, pp.1–8.
-

-
- [49] Chauhan, S., Sharma, E. and Doegar, A. (2016) 'Binarization techniques for degraded document images – a review', 2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), September, IEEE, pp.163–166.
- [50] Sulaiman, A., Omar, K. and Nasrudin, M.F. (2019) 'Degraded historical document binarization: a review on issues, challenges, techniques, and future directions', *Journal of Imaging*, Vol. 5, No. 4, pp.1–25.
- [51] Tensmeyer, C. and Martinez, T. (2020) 'Historical document image binarization: a review', *SN Computer Science*, Vol. 1, No. 3, p.173.
- [52] Mustafa, W.A. and Kader, M.M.M.A. (2018a) 'Binarization of document images: a comprehensive review', *Journal of Physics: Conference Series*, June, IOP Publishing, Vol. 1019, No. 1, pp.1–10.
- [53] Bhowmik, S. (2023) 'Document image binarization', *Document Layout Analysis*, pp.11–30, Springer Nature Singapore, Singapore.
- [54] Chaki, N., Shaikh, S.H., Saeed, K., Chaki, N., Shaikh, S.H. and Saeed, K. (2014) *A Comprehensive Survey on Image Binarization Techniques*, pp.5–15, Springer, India.
- [55] Chaki, N., Shaikh, S.H., Saeed, K., Chaki, N., Shaikh, S.H. and Saeed, K. (2014) *A Comprehensive Survey on Image Binarization Techniques*, pp.5–15, Springer, India.
- [56] Mustafa, W.A., Aziz, H., Khairunizam, W., Ibrahim, Z., Shahrman, A.B. and Razlan, Z.M. (2018) 'Review of different binarization approaches on degraded document images', 2018 International Conference on Computational Approach in Smart Systems Design and Applications (ICASSDA), August, IEEE, pp.1–8.
- [57] Chauhan, S., Sharma, E. and Doegar, A. (2016) 'Binarization techniques for degraded document images – a review', 2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), September, IEEE, pp.163–166.
- [58] Sulaiman, A., Omar, K. and Nasrudin, M.F. (2019) 'Degraded historical document binarization: a review on issues, challenges, techniques, and future directions', *Journal of Imaging*, Vol. 5, No. 4, pp.1–25.
- [59] Tensmeyer, C. and Martinez, T. (2020) 'Historical document image binarization: a review', *SN Computer Science*, Vol. 1, No. 3, p.173.
- [60] Jana, P., Ghosh, S., Bera, S.K. and Sarkar, R. (2017a) 'Handwritten document image binarization: an adaptive k-means based approach', 2017 IEEE Cactus Conference (CALCON), December, IEEE, pp.226–230.
- [61] Jana, P., Ghosh, S., Bera, S.K. and Sarkar, R. (2017a) 'Handwritten document image binarization: an adaptive k-means based approach', 2017 IEEE Cactus Conference (CALCON), December, IEEE, pp.226–230.
- [62] Papamarkos, N. (2001) 'A technique for fuzzy document binarization', *Proceedings of the 2001 ACM Symposium on Document Engineering*, November, pp.152–156.
- [63] Mustafa, W.A. and Kader, M.M.M.A. (2018a) 'Binarization of document images: a comprehensive review', *Journal of Physics: Conference Series*, June, IOP Publishing, Vol. 1019, No. 1, pp.1–10.
- [64] Jana, P., Ghosh, S., Sarkar, R. and Nasipuri, M. (2017b) 'A fuzzy C-means based approach towards efficient document image binarization', 2017 Ninth International Conference on Advances in Pattern Recognition (ICAPR), December, IEEE, pp.1–6.
- [65] Mondal, T., Coustaty, M., Gomez-Kramer, P. and Ogier, J.M. (2019) 'Learning free document image binarization based on fast fuzzy c-means clustering', 2019 International Conference on Document Analysis and Recognition (ICDAR), September, IEEE, pp.1384–1389.
- [66] Tong, L.J., Chen, K., Zhang, Y., Fu, X.L. and Duan, J.Y. (2009) 'Document image binarization based on NFCM', 2009 2nd International Congress on Image and Signal Processing, October, IEEE, pp.1–5.
- [67] Papamarkos, N. (2001) 'A technique for fuzzy document binarization', *Proceedings of the 2001 ACM Symposium on Document Engineering*, November, pp.152–156.
- [68] Mustafa, W.A. and Kader, M.M.M.A. (2018a) 'Binarization of document images: a comprehensive review', *Journal of Physics: Conference Series*, June, IOP Publishing, Vol. 1019, No. 1, pp.1–10.
- [69] Jana, P., Ghosh, S., Sarkar, R. and Nasipuri, M. (2017b) 'A fuzzy C-means based approach towards efficient document image binarization', 2017 Ninth International Conference on Advances in Pattern Recognition (ICAPR), December, IEEE, pp.1–6.
- [70] Mondal, T., Coustaty, M., Gomez-Kramer, P. and Ogier, J.M. (2019) 'Learning free document image binarization based on fast fuzzy c-means clustering', 2019 International Conference on Document Analysis and Recognition (ICDAR), September, IEEE, pp.1384–1389.
- [71] Tong, L.J., Chen, K., Zhang, Y., Fu, X.L. and Duan, J.Y. (2009) 'Document image binarization based on NFCM', 2009 2nd International Congress on Image and Signal Processing, October, IEEE, pp.1–5.
- [72] Jia, F., Shi, C., He, K., Wang, C. and Xiao, B. (2018) 'Degraded document image binarization using structural symmetry of strokes', *Pattern Recognition*, Vol. 74, No. 1, pp.225–240.
- [73] Wu, Y., Natarajan, P., Rawls, S. and AbdAlmageed, W. (2016) 'Learning document image binarization from data', 2006 IEEE International Conference on Image Processing (ICIP), September, IEEE, pp.3763–3767.
- [74] Yazid, H. and Arof, H. (2013) 'Gradient based adaptive thresholding', *Journal of Visual Communication and Image Representation*, Vol. 24, No. 7, pp.926–936.
-

-
- [75] Yazid, H. and Arof, H. (2013) 'Gradient based adaptive thresholding', *Journal of Visual Communication and Image Representation*, Vol. 24, No. 7, pp.926–936.
- [76] Tensmeyer, C. and Martinez, T. (2020) 'Historical document image binarization: a review', *SN Computer Science*, Vol. 1, No. 3, p.173.
- [77] Jia, F., Shi, C., He, K., Wang, C. and Xiao, B. (2018) 'Degraded document image binarization using structural symmetry of strokes', *Pattern Recognition*, Vol. 74, No. 1, pp.225–240.
- [78] Kim, I. J. (2004, October). Multi-window binarization of camera image for document recognition. In *Ninth International Workshop on Frontiers in Handwriting Recognition* (pp. 323-327). IEEE.
- [79] Sezgin, M. and Sankur, B.L. (2004) 'Survey over image thresholding techniques and quantitative performance evaluation', *Journal of Electronic Imaging*, Vol. 13, No. 1, pp.146–168.
- [80] Chaki, N., Shaikh, S.H., Saeed, K., Chaki, N., Shaikh, S.H. and Saeed, K. (2014) *A Comprehensive Survey on Image Binarization Techniques*, pp.5–15, Springer, India.
- [81] Otsu, N. (1979) 'A threshold selection method from gray-level histograms', *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 9, No. 1, pp.62–66.
- [82] Johannsen, G. (1982) 'A threshold selection method using information measures', *International Conference on Pattern Recognition (ICPR'82)*, pp.140–143.
- [83] Kapur, J.N., Sahoo, P.K. and Wong, A.K. (1985) 'A new method for gray-level picture thresholding using the entropy of the histogram', *Computer Vision, Graphics, and Image Processing*, Vol. 29, No. 3, pp.273–285.
- [84] Sauvola, J. and Pietikainen, M. (2000) 'Adaptive document image binarization', *Pattern recognition*, Vol. 33, No. 2, pp.225–236.
- [85] Niblack, W. (1985) *An Introduction to Digital Image Processing*, Strandberg Publishing Company, Denmark.
- [86] Wu, Y., Natarajan, P., Rawls, S. and AbdAlmageed, W. (2016) 'Learning document image binarization from data', *2016 IEEE International Conference on Image Processing (ICIP)*, September, IEEE, pp.3763–3767.
- [87] Drira, F. (2006) 'Towards restoring historic documents degraded over time', *Second International Conference on Document Image Analysis for Libraries (DIAL'06)*, April, IEEE, pp.8–357.
- [88] Liu, Y. and Srihari, S.N. (1997) 'Document image binarization based on texture features', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, No. 5, pp.540–544.
- [89] He, J., Do, Q.D.M., Downton, A.C. and Kim, J. (2005) 'A comparison of binarization methods for historical archive documents', *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)*, August, IEEE, pp.538–542.
- [90] Yazid, H. and Arof, H. (2013) 'Gradient based adaptive thresholding', *Journal of Visual Communication and Image Representation*, Vol. 24, No. 7, pp.926–936.
- [91] Yazid, H. and Arof, H. (2013) 'Gradient based adaptive thresholding', *Journal of Visual Communication and Image Representation*, Vol. 24, No. 7, pp.926–936.
- [92] Yazid, H. and Arof, H. (2013) 'Gradient based adaptive thresholding', *Journal of Visual Communication and Image Representation*, Vol. 24, No. 7, pp.926–936.
- [93] Yanowitz, S.D. and Bruckstein, A.M. (1989) 'A new method for image segmentation', *Computer vision, Graphics, and Image Processing*, Vol. 46, No. 1, pp.82–95.
- [94] Otsu, N. (1979) 'A threshold selection method from gray-level histograms', *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 9, No. 1, pp.62–66.
- [95] Sauvola, J. and Pietikainen, M. (2000) 'Adaptive document image binarization', *Pattern recognition*, Vol. 33, No. 2, pp.225–236.
- [96] Niblack, W. (1985) *An Introduction to Digital Image Processing*, Strandberg Publishing Company, Denmark.
- [97] Khurshid, K., Siddiqi, I., Faure, C. and Vincent, N. (2009) 'Comparison of Niblack inspired binarization methods for ancient documents', *Document Recognition and Retrieval XVI*, January, Vol. 7247, No. 1, pp.267–275, SPIE.
- [98] Pierre, D. (1993) *WELLNER Adaptive Thresholding for the Digital Desk*, Technical Report EPC-1993-110.
- [99] Bernsen, J. (1986) 'Dynamic thresholding of gray-level images', *Proceedings of 8th International Conference on Pattern Recognition*.
- [100] Woods, R.E. and Gonzalez, R.C. (2008) *Digital Image Processing*, Pearson Education, India.
- [101] Tensmeyer, C. and Martinez, T. (2017) 'Document image binarization with fully convolutional neural networks', *2017 14th IAPR International Conference on document analysis and recognition (ICDAR)*, November, IEEE, pp.99–104.
- [102] Kefali, A., Sari, T. and Bahi, H. (2014) 'Foreground-background separation by feed-forward neural networks in old manuscripts', *Informatica*, Vol. 38, No. 4, pp.329–338.
- [103] Sari, T., Kefali, A. and Bahi, H. (2012) 'An MLP for binarizing images of old manuscripts', *2012 International Conference on Frontiers in Handwriting Recognition*, September, IEEE, pp.247–251.
- [104] Akbari, Y., AI-Maadeed, S. and Adam, K. (2020) 'Binarization of degraded document images using convolutional neural networks and wavelet-based multichannel images', *IEEE Access*, Vole. 8, No. 1, pp.153517–
-

153534.

[105] Tensmeyer, C. and Martinez, T. (2017) 'Document image binarization with fully convolutional neural networks', 2017 14th IAPR International Conference on document analysis and recognition (ICDAR), November, IEEE, pp.99–104.

[106] Kefali, A., Sari, T. and Bahi, H. (2014) 'Foreground-background separation by feed-forward neural networks in old manuscripts', *Informatica*, Vol. 38, No. 4, pp.329–338.

[107] Sari, T., Kefali, A. and Bahi, H. (2012) 'An MLP for binarizing images of old manuscripts', 2012 International Conference on Frontiers in Handwriting Recognition, September, IEEE, pp.247–251.

[108] Akbari, Y., AI-Maadeed, S. and Adam, K. (2020) 'Binarization of degraded document images using convolutional neural networks and wavelet-based multichannel images', *IEEE Access*, Vole. 8, No. 1, pp.153517–153534.

[109] Badekas, E. and Papamarkos, N. (2007) 'Optimal combination of document binarization techniques using a self-organizing map neural network', *Engineering Applications of Artificial Intelligence*, Vol. 20, No. 1, pp.11–24.

[110] Sari, T., Kefali, A. and Bahi, H. (2014) 'Text extraction from historical document images by the combination of several thresholding techniques', *Advances in Multimedia*, Vol. 2014, No. 1, pp.1–10.

[111] Lelore, T. and Bouchara, F. (2013) 'FAIR: a fast algorithm for document image restoration', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 35, No. 8, pp.2039–2048.

[112] Sari, T., Kefali, A. and Bahi, H. (2014) 'Text extraction from historical document images by the combination of several thresholding techniques', *Advances in Multimedia*, Vol. 2014, No. 1, pp.1–10.

[113] Lelore, T. and Bouchara, F. (2013) 'FAIR: a fast algorithm for document image restoration', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 35, No. 8, pp.2039–2048.

[114] Howe, N.R. (2013) 'Document binarization with automatic parameter tuning', *International Journal on Document Analysis and Recognition (IJ DAR)*, Vol. 16, No. 3, pp.247–258.

[115] Nobuyuki Otsu, "A threshold selection method from gray-level histograms", *IEEE Transactions on Systems, Man, and Cybernetics*, 1979 – Méthode de seuillage global par maximisation de la variance inter-classes math.ucdavis.edu+7researchgate.net+7scientific.net+7ijaem.netfr.wikipedia.orgencord.com+2en.wikipedia.org+2ripublication.com+2

[116] Encord, "Image Thresholding in Image Processing", 2023 – Explications sur les méthodes de seuillage, Otsu, seuillage adaptatif encord.com.

[117] Nobuyuki Otsu, "A threshold selection method from gray-level histograms", *IEEE Transactions on Systems, Man, and Cybernetics*, 1979 – Méthode de seuillage global par maximisation de la variance inter-classes math.ucdavis.edu+7researchgate.net+7scientific.net+7ijaem.netfr.wikipedia.orgencord.com+2en.wikipedia.org+2ripublication.com+2. Encord, "Image Thresholding in Image Processing", 2023 – Explications sur les méthodes de seuillage, Otsu, seuillage adaptatif encord.com.

[118] Encord, "Image Thresholding in Image Processing", 2023 – Explications sur les méthodes de seuillage, Otsu, seuillage adaptatif encord.com.

[119] Wikipédia (article "Thresholding (image processing)") – Limites du seuillage global et avantages du seuillage local en.wikipedia.org

[120] Vision Systems Design – "Understanding region-based segmentation" :« Region-growing techniques cluster the pixels that represent homogeneous areas in an image ... by grouping adjacent pixels whose properties differ by less than some specified amount. » github.com+15vision-systems.com+15bioinfopublication.org+15stackoverflow.com

[121] Vision Systems Design – "Understanding region-based segmentation" :« Region-growing techniques cluster the pixels that represent homogeneous areas in an image ... by grouping adjacent pixels whose properties differ by less than some specified amount. » github.com+15vision-systems.com+15bioinfopublication.org+15stackoverflow.com

[122] Wikipédia – entrée "Region growing"

[123] D. Marr and E. Hildreth, "Theory of Edge Detection," *Proceedings of the Royal Society B: Biological Sciences*, vol. 207, no. 1167, pp. 187–217, 1980.

[124] J. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[125] R. M. Haralick, "Digital Step Edges from Zero Crossing of Second Directional Derivatives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 1, pp. 58–68, 1984.

[126] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed., Prentice Hall, 2008.

[127] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed., Prentice Hall, 2008.

[128] [Memon, 2020] : Memon, J., Sami, M., Khan, R. A., & Uddin, M. (2020). Handwritten optical character ecognition (OCR): A comprehensive systematic literature review (SLR). *IEEE access*, 8, 142642-142668.

-
- [129] [Abirami, 2015] : Abirami, S., Essakiammal, V., & Baskaran, R. (2016). Statistical features based character recognition for offline handwritten Tamil document images using HMM. *International Journal of Computational Vision and Robotics*, 5(4), 422-440.
- [130] [Subramani, 2020]: Subramani, N., Matton, A., Greaves, M., & Lam, A (2020). A survey of deep learning approaches for ocr and document understanding. *arXiv preprint arXiv:2011.13534*.
- [131] [Wong, 2001] : Wong, W. S., Sherkat, N., & Allen, T. (2001, September). Contextual focus for improved recognition of hand-filled forms. In *Proceedings of Sixth International Conference on Document Analysis and Recognition* (pp. 748-752). IEEE.
- [132] Lu, H., Kot, A. C., & Shi, Y. Q. (2004). Distance-reciprocal distortion measure for binary document images. *IEEE Signal Processing Letters*, 11(2), 228-231.
- [133] Lu, H., Kot, A. C., & Shi, Y. Q. (2004). Distance-reciprocal distortion measure for binary document images. *IEEE Signal Processing Letters*, 11(2), 228-231.
- [134] Bouvy, J., & Warnier, S. (2016). *Le QCM : Vade-mecum pour l'enseignant et l'étudiant*.
- [135] Rajan, R., Mishra, P., & Sharma, D. (2019). Survey on Optical Mark Recognition and its applications. *International Journal of Computer Applications*.
- [136] Smith, R. (2007). An Overview of the Tesseract OCR Engine. *Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR)*
- [137] Lebrun, M. (2005). *Des technologies pour enseigner et apprendre : Une introduction aux technologies de l'information et de la communication pour l'enseignement (TICE)*. De Boeck Université.
- [138] Bouvy, J., & Warnier, S. (2016). *Le QCM : Vade-mecum pour l'enseignant et l'étudiant*.
- [139] Roegiers, X. (2010). *Une pédagogie de l'intégration : Compétences et intégration des acquis dans l'enseignement*. De Boeck Supérieur.
- [140] Benhassine, F. (2018). *Conception et réalisation d'un système de correction automatisée des examens universitaires à base de QCM*. Mémoire de Master, Université de Bejaia.
- [141] De Ketele, J.-M., & Gérard, F.-M. (2005). *Mesurer les compétences des élèves : Vers de nouveaux outils*. De Boeck Supérieur.
- [142] Bouvy, C., & Warnier, S. (2016). *Le QCM : vade-mecum de conception et de correction*.
- [143] Boussad, K. (2020). *Conception et développement d'une application web de correction automatique des QCM* (Mémoire de Master, Université de Béjaïa).
- [144] Guedjali, F. (2018). *Réalisation d'une application mobile de correction automatique de QCM sur papier* (Mémoire de Master, Université de Constantine 2).
- [145] Bouvy, J., & Warnier, S. (2016). *Le QCM : Vade-mecum pour l'enseignant et l'étudiant*.
- [146] Boussad, K. (2020). *Conception et développement d'une application web de correction automatique des QCM* (Mémoire de Master, Université de Béjaïa).
- [147] Guedjali, F. (2018). *Réalisation d'une application mobile de correction automatique de QCM sur papier* (Mémoire de Master, Université de Constantine 2).
- [148] Boussad, K. (2020). *Conception et développement d'une application web de correction automatique des QCM* (Mémoire de Master, Université de Béjaïa).
- [149] Bouvy, C., & Warnier, S. (2016). *Le QCM : vade-mecum de conception et de correction*.
- [150] Benamara, A. (2015). *Étude et réalisation d'un système intelligent de correction automatique des QCM par reconnaissance optique de marques*. Mémoire de Master, Université Abderrahmane Mira de Béjaïa.
- [151] Boussad, K. (2020). *Conception et développement d'une application web de correction automatique des QCM*. Mémoire de Master, Université de Béjaïa.
- [152] Guedjali, F. (2018). *Réalisation d'une application mobile de correction automatique de QCM sur papier*. Mémoire de Master, Université de Constantine 2.
- [153] González, R. C., & Woods, R. E. (2018). *Digital Image Processing*. Pearson.
- [154] Rajput, A., & Kulkarni, R. (2012). Optical Mark Recognition based Automatic Test Grading System. *International Journal of Computer Applications*.
- [155] Benamara, A. (2015). *Étude et réalisation d'un système intelligent de correction automatique des QCM par reconnaissance optique de marques*. Mémoire de Master, Université Abderrahmane Mira de Béjaïa.
- [156] González, R. C., & Woods, R. E. (2018). *Digital Image Processing*. Pearson.
- [157] Rajput, A., & Kulkarni, R. (2012). Optical Mark Recognition based Automatic Test Grading System. *International Journal of Computer Applications*.
- [158] Houda, D., et al. (2021). Automated MCQ Correction System Based on Image Processing Techniques. In *Proceedings of the 2021 International Conference on Information Systems and Advanced Technologies (ICISAT 2021)*.
- [159] Boussad, K. (2020). *Conception et développement d'une application web de correction automatique des QCM*. Mémoire de Master, Université de Béjaïa.
- [160] Sommerville, I. (2016). *Software Engineering* (10th ed.). Pearson.
-