

الجمهورية الجزائرية الديمقراطية الشعبية  
وزارة التعليم العالي و البحث العلمي

République Algérienne Démocratique et Populaire  
Ministère de L'Enseignement Supérieur et de la Recherche Scientifique

**Université 8 Mai 1945 Guelma**



Faculté des Mathématiques, de l'Informatique et des Sciences de la Matière  
Département d'Informatique

**Thèse de doctorat troisième cycle**

**Spécialité : Informatique**

Présentée par

**MERABTI Hocine**

**THÈME**

---

*Approches bio-inspirées pour la reconnaissance de formes*

---

**Directeur de thèse :**

Hamid SERIDI

Prof. Université 8 Mai 1945, Guelma

**DEVANT LE JURY**

**Président :**

Yacine LAFIFI

Prof. Université 8 Mai 1945, Guelma

**Examineur :**

Mohammed NEMISSI

MCA. Université 8 Mai 1945, Guelma

Taher BOUHADADA

Prof. Université Badji Mokhtar, Annaba

Nadir FARAH

Prof. Université Badji Mokhtar, Annaba

**Année 2017**

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

**J**e dédie ce modeste travail à :

*Mes très chers parents,*

*Mes chères sœurs,*

*Toute ma famille,*

*Tous mes amis.*

## Remerciements

**P**AR ces quelques lignes, je tiens à remercier toutes les personnes qui ont participé de près ou de loin au bon déroulement de cette thèse.

**E**N premier lieu, je tiens à remercier spécialement mon directeur de thèse Prof. Hamid SEDIDI pour la confiance qu'il m'a témoignée en acceptant d'encadrer ce travail. Je lui remercie également pour le temps et la patience qu'il m'a accordés tout au long de ces années.

**M**ES remerciements vont également aux messieurs Brahim FAROU et Mohamed Nadjib KOUAHLA pour toutes nos discussions, leurs suggestions toujours avisées et leur relecture scrupuleuse du manuscrit.

**J**E ne manquerais pas non plus de dire un grand merci aux membres du jury : Prof. Yacine LAFIFI le président, Dr. Mohamed NEMISSI , Prof. Taher BOUHADADA et Prof. Nadir FARAH les examinateurs, qui ont accepté d'évaluer cette thèse à sa juste valeur, et de me faire part de leurs remarques surement pertinentes qui contribueront au perfectionnement du présent travail.

**U**N très grand merci à tous mes collègues membres du laboratoire LabSTIC, pour leur soutien durant ce parcours doctoral.

## ملخص

اهتم العديد من الباحثين في مجال الاعلام الآلي لعقود من الزمن بتطوير أنظمة للتعرف على الكتابة اليدوية، ولا تزال حتى الآن تُعد حقل أبحاث جد نشط نظرا لكثرة مجالات تطبيقاتها المعاصرة مثل : القراءة الأتوماتيكية للعاوين البريدية، القراءة الأتوماتيكية للشيكات المصرفية، القراءة الأتوماتيكية للاستمارات، ... إلخ. تعتبر عملية التعرف على الكتابة مهمة صعبة، و ذلك بسبب التنوع في أشكال الحروف و الاختلاف في أنماط الكتابة. يوفر علم البيولوجيا العديد من مصادر الالهام و الاستيحاء في مجال الذكاء الاصطناعي و التعرف على الأشكال. مجال الاعلام الآلي المستوحى من البيولوجيا يقوم باستغلال هذه المصادر من أجل تطوير خوارزميات و أنظمة الكمبيوتر القادرة على حل المشاكل المتعلقة بالتعرف على الأشكال، و تحديدا التعرف على المخطوطات المكتوبة يدويا. باستيحائنا من خصائص الجهاز المناعي، نقترح هنا نموذجا يسمح بالكشف و التعرف على أشكال معقدة مثل الأرقام و الحروف المكتوبة يدويا. الهدف من هذا البحث هو اقتراح مقارنة جديدة غير مباشرة من أجل التعرف على المخطوطات المعزولة (حروف و أرقام). تهدف هذه المقاربة إلى تحسين نوعية قاعدة البيانات و التعامل مع حالة عدم اليقين المتعلقة بالمخطوطات من أجل زيادة أداء التصنيف. تستند هذه المقاربة على نظام المناعة الاصطناعي (AIS) و نظام مستند على قواعد النوعية (QRBS). يعتبر AIS كمحسن فهو يختار أفضل المرشحين من أجل مرحلة التعلم، أما QRBS فيعمل كنظام التعرف النوعي فهو يستخدم قواعد النوعية لكي يتعرف على المخطوطات و يعالج أيضا كلاً من حالة الغموض و عدم اليقين في الكتابة خلال مراحل التعلم و التصنيف التي تجعل بعض الحروف غير قابلة للقراءة و تقلل كذلك من اتخاذ القرار النهائي. التجارب أجريت على قواعد البيانات MNIST و الحروف الانجليزية. المقارنات مع النهج الحديثة الاخرى التي استخدمت نفس قاعدة البيانات تدل على أنّ المقاربة المقترحة فعّالة.

**الكلمات المفتاحية :** التعرف على الأنماط، التعرف على المخطوطات اليدوية، المقاربة المستوحاة من البيولوجيا، أنظمة المناعة الاصطناعية، المنطق الضبابي، المقاربة النوعية.

# Abstract

For several decades, the handwriting recognition has attracted the attention of many researchers and remains a very active research field due to its numerous applications such as: the automatic reading of addresses and forms, bank checks processing, etc. The handwriting recognition is a difficult task due to the great variability of writing styles. Biology allows offering many inspiration sources in the field of artificial intelligence and pattern recognition. The bio-inspired computing exploits these inspiration sources in order to develop algorithms and computer systems that are able solving the pattern recognition problems, and more precisely the handwriting recognition. Inspired by the immune system properties, we propose a model that allows detecting and recognizing complex shapes such as handwritten digits and letters. The aim of this study is to provide a new off-line approach for isolated handwritten characters recognition (digits and letters). This approach aims to improve the database quality and to deal with characters uncertainty in order to increase the classification performance. It is based on a combination of two main subsystems: an artificial immune system (AIS) and qualitative rules-based system (QRBS). The AIS acts as an optimizer. It selects the best candidates for training. QRBS works as a qualitative recognizer system. It uses qualitative rules to recognize characters. During the training and classification phases, it handles both vagueness and uncertainty in handwriting that make some characters unreadable and may decrease the accuracy of the overall process. Experiments are conducted on the MNIST and English letter databases. Comparisons with other recent approaches using the same database indicate that our approach is effective.

**Key-words:** Pattern recognition, Handwritten characters recognition, Bio inspired approach, Artificial immune system, Fuzzy logic, Qualitative approach.

## Résumé

Depuis plusieurs décennies, la reconnaissance de l'écriture manuscrite a attiré l'attention d'un grand nombre de chercheurs et reste un champ de recherche très actif dû à ses nombreuses applications comme la lecture automatique des formulaires, des adresses postales, des chèques bancaires, etc. La reconnaissance de l'écriture manuscrite est une tâche difficile en raison de la grande variabilité des styles d'écritures. La biologie permet d'offrir de nombreuses sources d'inspiration dans le domaine de l'intelligence artificielle et de la reconnaissance de formes. L'informatique bio inspirée exploite ces sources d'inspiration afin de développer des algorithmes et des systèmes informatiques qui sont capables de résoudre les problèmes de la reconnaissance de formes, et plus précisément la reconnaissance d'écritures manuscrite. Inspirés par les propriétés du système immunitaire, nous proposons un modèle qui permet de détecter et de reconnaître des formes complexes comme les chiffres et lettres manuscrits. L'objectif de ce travail de recherche est de proposer une nouvelle approche pour la reconnaissance de caractères manuscrits isolé (chiffres et lettres). Cette approche a pour but d'améliorer la qualité de la base de données et de traiter l'incertitude des caractères afin d'augmenter la performance de classification. Elle est basée sur une combinaison de deux principaux sous-systèmes : un système immunitaire artificiel (AIS) et un système à base des règles qualitatives (QRBS). L'AIS se considère comme un optimiseur. Il sélectionne les meilleurs candidats pour l'apprentissage. Le QRBS travail en tant qu'un système de reconnaissance qualitative. Il utilise des règles qualitatives pour reconnaître les caractères. Lors de la phase de décision, il gère à la fois l'imprécision et l'incertitude dans l'écriture qui rendent certains caractères illisibles et peuvent diminuer la prise de décision final. Les expériences sont menées sur les bases de données MNIST et lettres anglaises. Les comparaisons avec d'autres approches récentes utilisant la même base de données indiquent que notre approche est efficace.

**Mots-Clés :**Reconnaissance de formes, Reconnaissance de caractères manuscrits, Approches bio inspirées, Systèmes immunitaires artificiels, Logique flou, Approche qualitative.

# Table des matières

ملخص	i
<b>Abstract</b>	<b>ii</b>
<b>Résumé</b>	<b>iii</b>
<b>Table des figures</b>	<b>viii</b>
<b>Liste des tableaux</b>	<b>x</b>
<b>Liste des algorithmes</b>	<b>xi</b>
<b>Liste des abréviations</b>	<b>xii</b>
<b>Introduction Générale</b>	<b>1</b>
<b>1 La reconnaissance de formes</b>	<b>5</b>
1.1 Introduction . . . . .	5
1.2 Présentation de la reconnaissance de formes . . . . .	5
1.3 Les domaines d'application . . . . .	6
1.3.1 Reconnaissance des formes sur signaux temporels . . . . .	7
1.3.2 Reconnaissance des formes dans les images numériques . . . . .	7
1.4 Processus de la reconnaissance de formes . . . . .	8
1.4.1 Schéma fondamental . . . . .	8
1.4.2 Vers l'implémentation . . . . .	9
1.4.3 Les étapes d'un processus de RdF . . . . .	10
1.4.4 Combinaison des classifieurs . . . . .	16
1.5 Conclusion . . . . .	17
<b>2 La reconnaissance de l'écriture manuscrite</b>	<b>18</b>
2.1 Introduction . . . . .	18
2.2 Les applications pratiques . . . . .	19
2.3 Les systèmes de reconnaissance de l'écriture . . . . .	19
2.3.1 Reconnaissance selon le mode d'acquisition . . . . .	19
2.3.1.1 La reconnaissance hors-ligne (off-line) . . . . .	19
2.3.1.2 La reconnaissance en-ligne (on-line) . . . . .	20
2.3.2 Reconnaissance selon le type d'écriture . . . . .	22



2.3.2.1	Ecriture imprimée . . . . .	22
2.3.2.2	Ecriture manuscrite . . . . .	22
2.4	Les formes du texte manuscrit . . . . .	22
2.5	Les types de reconnaissance d'écriture manuscrite . . . . .	24
2.5.1	La Reconnaissance de caractères . . . . .	25
2.5.2	La Reconnaissance des mots . . . . .	25
2.5.3	La reconnaissance de phrases et textes . . . . .	26
2.6	Le fonctionnement d'un système de reconnaissance de caractères manus- crits . . . . .	27
2.6.1	L'étape d'acquisition . . . . .	28
2.6.2	L'étape de prétraitement . . . . .	28
2.6.2.1	Binarisation . . . . .	29
2.6.2.2	Réduction de bruit . . . . .	32
2.6.2.3	Localisation et normalisation . . . . .	33
2.6.2.4	Squelettisation . . . . .	35
2.6.3	La segmentation . . . . .	35
2.6.3.1	Segmentation implicite . . . . .	36
2.6.3.2	Segmentation explicite . . . . .	37
2.6.4	L'extraction des caractéristiques . . . . .	37
2.6.4.1	Caractéristiques statistiques . . . . .	38
2.6.4.2	Caractéristiques structurales . . . . .	38
2.6.4.3	Transformations globales . . . . .	39
2.6.5	La classification . . . . .	39
2.6.5.1	Les approche basée sur les prototypes . . . . .	39
2.6.5.2	Les approches statistiques . . . . .	39
2.6.5.3	Les approches structurales . . . . .	40
2.6.5.4	Les approches bios inspirées . . . . .	40
2.6.5.4.1	Les réseaux de neurones artificiels (RNAs) . . . . .	41
2.6.5.4.2	Les approches évolutionnaires (AEs) . . . . .	43
2.6.5.4.3	L'intelligence en essaim (IE) . . . . .	45
2.6.5.4.4	Les systèmes immunitaires artificiels (AISs) . . . . .	46
2.6.6	Le post-traitement . . . . .	47
2.7	Problèmes liés à la reconnaissance d'écriture manuscrite . . . . .	47
2.8	Les difficultés de la reconnaissance de l'écriture manuscrite . . . . .	48
2.8.1	Nombre de scripteurs . . . . .	48
2.8.2	Taille du vocabulaire . . . . .	48
2.8.3	Les contraintes morphologiques . . . . .	49
2.9	Conclusion . . . . .	49
<b>3</b>	<b>Les systèmes immunitaires artificiels</b>	<b>51</b>
3.1	Introduction . . . . .	51
3.2	Le système immunitaire naturel (biologique) . . . . .	51
3.2.1	Les différents types d'immunité . . . . .	52
3.2.1.1	L'immunité innée . . . . .	53
3.2.1.2	L'immunité adaptative . . . . .	53
3.2.2	La physiologie du système immunitaire naturel . . . . .	54
3.2.3	Le mécanisme de défense du système immunitaire naturel . . . . .	55
3.2.4	Propriétés du système immunitaire naturel . . . . .	56

3.3	Le système immunitaire artificiel (AIS)	57
3.3.1	Définitions	57
3.3.2	Le processus de conception d'un AIS	57
3.3.2.1	La représentation	58
3.3.2.2	Les mesures d'affinité	59
3.3.2.3	Les algorithmes immunitaires	59
3.3.3	Les algorithmes immunitaires de base	59
3.3.3.1	L'algorithme de la sélection négative NSA	60
3.3.3.2	L'algorithme de la sélection clonale CSA	61
3.3.3.3	Le réseau immunitaire artificiel AIN	63
3.3.3.4	Le système immunitaire artificiel de reconnaissance AIRS	66
3.3.3.5	La théorie de danger DT	67
3.4	Pourquoi utiliser les systèmes immunitaires artificiels pour la reconnaissance de formes	69
3.5	Comparaison entre les différents systèmes bio inspirés	70
3.6	Conclusion	72
<b>4</b>	<b>L'approche qualitative</b>	<b>73</b>
4.1	Introduction	73
4.2	Architecture	74
4.2.1	Génération automatique des règles	74
4.2.1.1	Construction des prémisses	75
4.2.1.2	Interface homme-machine	78
4.2.2	Système d'inférence	79
4.2.3	Prise de décision optimale	80
4.3	Conclusion	83
<b>5</b>	<b>Proposition d'un système pour la reconnaissance de caractères manuscrits</b>	<b>85</b>
5.1	Introduction	85
5.2	Formulation du problème	85
5.2.1	Prétraitement	87
5.2.2	Extraction de caractéristiques	87
5.2.2.1	Caractéristiques statistiques	88
5.2.2.2	Caractéristiques structurelles	91
5.2.3	Classification	92
5.3	Les approches proposées pour la reconnaissance de caractères manuscrits	92
5.3.1	L'algorithme AIRS	92
5.3.1.1	Description de l'algorithme	92
5.3.1.1.1	Initialisation	93
5.3.1.1.2	Apprentissage de l'antigène	93
5.3.1.1.3	Classification	96
5.3.1.2	Paramètres de l'algorithme	97
5.3.2	L'approche bio-qualitative	99
5.3.2.1	L'algorithme AIS pour la réduction de données	100
5.3.2.2	L'algorithme QRBS pour la classification	101
5.4	Résultats expérimentaux	102
5.4.1	Bases de données	102
5.4.2	Évaluation de l'AIS	103

5.4.3	Évaluation de l'AIS_QRBS . . . . .	106
5.4.4	Comparaisons . . . . .	111
5.5	Conclusion . . . . .	113
<b>Conclusion Générale</b>		<b>114</b>
<b>Annexe A Système Immunitaire Artificiel (AIS)</b>		<b>117</b>
<b>Annexe Système Immunitaire Artificiel (AIS)</b>		<b>117</b>
A.1	Étape d'initialisation . . . . .	118
A.2	Étape d'identification de cellule mémoire et génération des ARBs . . . . .	119
A.3	Étape d'allocation des ressources et compétition . . . . .	121
A.4	Étape de mise à jours de l'ensemble de cellules mémoires MC . . . . .	123
A.5	Aspects computationnels de l'algorithme AIS . . . . .	123
<b>Bibliographie</b>		<b>125</b>
<b>Publications de l'auteur</b>		<b>140</b>

# Table des figures

1.1	Schéma élémentaire de RdF . . . . .	8
1.2	Analogie entre un système réel et un système de RdF . . . . .	11
1.3	Schéma général d'un système de RdF . . . . .	12
2.1	Exemple d'écriture en ligne et hors ligne [LOR 98]. . . . .	20
2.2	Exemple des plates-formes disposant d'un système en-ligne de reconnaissance de l'écriture manuscrite [SIT 16]. . . . .	21
2.3	Les différentes formes du texte manuscrit [LAA 97] . . . . .	23
2.4	Les différentes formes d'écriture manuscrite . . . . .	24
2.5	Les différentes étapes pour la reconnaissance d'un texte . . . . .	28
2.6	Histogramme des niveaux de gris . . . . .	29
2.7	Problème de seuillage global . . . . .	30
2.8	Résultats de seuillage globales sur les manuscrits arabes [KEF 10] . . . . .	31
2.9	Résultats de seuillage locales sur les manuscrits arabes [KEF 10] . . . . .	32
2.10	Histogramme de la segmentation lignes [FAR 09] . . . . .	34
2.11	Exemples de mots manuscrits avec des tailles différentes [AMR 12]. . . . .	34
2.12	La technique des histogrammes de projection pour la détection d'inclinaison [HAJ 05] . . . . .	35
2.13	(a) image original avec l'angle d'inclinaison, (b) la correction d'inclinaison par les histogrammes projection [LOU 09]. . . . .	35
2.14	Topologie d'un mot pour l'estimation de ligne de base [BEN 07] . . . . .	35
2.15	Différentes segmentation pour le mot quatorze [AUG 01] . . . . .	37
2.16	Les approches bios inspirées . . . . .	41
2.17	Un réseau de neurones. (a) modèle biologique, (b) modèle artificiel . . . . .	42
2.18	Schéma globale d'un système évolutionnaire . . . . .	44
3.1	Différentes lignes de défense dans un système immunitaire naturel [TIM 00].	52
3.2	Architecture du système immunitaire. . . . .	53
3.3	Présentation d'une cellule B et cellule T . . . . .	54
3.4	Mécanisme de défense du système immunitaire naturel . . . . .	55
3.5	Schéma général pour résoudre un problème par un AIS . . . . .	58
3.6	La représentation du modèle Shape-Space . . . . .	59
3.7	Structure générale de l'algorithme de la sélection négative. (a) la phase Examen (Censoring), (b) la phase contrôle (Monitoring) . . . . .	61
3.8	Structure générale de l'algorithme de la sélection clonale . . . . .	63
3.9	La théorie du réseau idiologique de Jerne [JER 74] . . . . .	64

3.10	Principe de théorie du danger [NAN 09] . . . . .	69
4.1	Architecture simplifiée du QRBS [SER 01]. . . . .	74
4.2	Les principales étapes d'un QRBS . . . . .	75
4.3	La construction des règles [SER 01] . . . . .	76
4.4	Système d'inférence [SER 01] . . . . .	80
5.1	Un aperçu du système proposé . . . . .	86
5.2	Prétraitement d'un chiffre 8 de la base de données MNIST . . . . .	87
5.3	Exemple d'une transformation : (a) Chiffre "9" original, (b) Rotation de chiffre "9" . . . . .	89
5.4	Histogrammes de projections. (a) Chiffre squeletté, (b) Projection hori- zontale, (c) Projection verticale, (d) Projection Diagonale 45°, (e) Projec- tion Diagonale 135° . . . . .	91
5.5	Caractéristiques structurelles. (a) Chiffre " 2 ", (b) Points extrêmes, point de croisement et occlusion extraites à partir du squelette de chiffre " 2 " .	91
5.6	Un aperçu du cycle de vie de l'algorithme AIRS . . . . .	93
5.7	Processus d'allocation des ressources et compétition . . . . .	94
5.8	Une illustration de différentes étapes de l'algorithme AIS pour la réduc- tion des données . . . . .	101
5.9	Un échantillon de chiffres de la base de données MNIST [LEC 98] . . . .	102
5.10	Un échantillon de lettres anglaises manuscrites [GAU 12] . . . . .	103
5.11	Taux de reconnaissance en fonction de l'initialisation des ARB . . . . .	104
5.12	Taux de reconnaissance en fonction du nombre de clones . . . . .	104
5.13	Les affections du paramètre de taux clonale dans les performances du système . . . . .	105
5.14	La variation du nombre de règles optimisé sur la base de données MNIST. (a) Seuil de corrélation = 0,9, (b) Seuil de corrélation = 0,95, (c) Seuil de corrélation = 0,99 . . . . .	109
5.15	Taux de reconnaissance sur la base de données MNIST. (a) Seuil de cor- rélation = 0,9, (b) Seuil de corrélation = 0,95, (c) Seuil de corrélation = 0,99 . . . . .	110
A.1	Étape d'initialisation . . . . .	118
A.2	Étape d'identification de cellule mémoire et génération des ARBs . . . .	119
A.3	Étape d'allocation des ressources et compétition . . . . .	121
A.4	Étape de mis à jours de l'ensemble de cellules mémoires MC . . . . .	123

# Liste des tableaux

1.1	Les différents domaines d'application de la RdF. . . . .	7
2.1	Une brève comparaison entre la reconnaissance en-ligne et hors-ligne [ALR 06] . . . . .	21
3.1	Les différentes distances et leurs formules mathématiques correspondantes.	59
3.2	Correspondance entre le NIS et l'AIRS. . . . .	67
3.3	comparaison entre les AISs, les AGs et les RNAs. . . . .	71
4.1	Correspondance entre le degré symbolique, l'échelle des valeurs linguistiques, l'espace des probabilités et le degré de probabilité symbolique . .	79
5.1	Paramètres de sélection pour AIRS. . . . .	103
5.2	Performances du AIS avec les approches KNN, Fuzzy-KNN et la Distance Euclidienne. . . . .	106
5.3	Résultats de la classification de la base MNIST avec l'approche qualitative	107
5.4	Résultats de la classification de la base MNIST avec la méthode AIS_QRBS	107
5.5	Une étude comparative sur l'approche qualitative avec et sans réduction de données et avec et sans prise de décision optimale . . . . .	111
5.6	Performances de reconnaissance de caractères manuscrits avec AIS_KNN, AIS_Fuzzy_KNN, et AIS_QRBS. . . . .	111
5.7	Comparaison de l'AIS_QRBS avec d'autres méthodes sur la base de données MNIST. . . . .	112
5.8	Comparaison de l'AIS_QRBS avec d'autres méthodes sur la base de données de lettres anglaise manuscrites . . . . .	112
A.1	Complexité computationnelle de l'algorithme AIS . . . . .	124

# Liste des Algorithmes

3.1	Algorithme de génération des détecteurs de la Sélection Négative . . . . .	61
3.2	Pseudo code de l'algorithme de la sélection clonale. . . . .	63
3.3	Pseudo code de l'algorithme du réseau immunitaire. . . . .	65
3.4	Pseudo code de l'algorithme du AIRS. . . . .	67
4.1	Algorithme d'optimisation de prise de décision . . . . .	81
5.1	Mise à jour de l'ensemble de cellules mémoire . . . . .	95
5.2	Pseudo code du Fuzzy-KNN . . . . .	97
A.1	Génération des ARBs . . . . .	120
A.2	Fonction de mutation . . . . .	120
A.3	Stimulation, Allocation de Recourse, Suppression d'ARB . . . . .	122

# Liste des abréviations

Ab	Anticorps
ACO	L'optimisation par colonie de fourmi (Ant Colony Optimization)
ACP	Analyse en composantes principales
AD	Arbres de décisions
AEs	Approches évolutionnaires
Ag	Antigène
AGs	Algorithmes génétiques
AIN	Réseau immunitaire artificiel (Artificial Immune Network)
AIRS	Système immunitaire artificielle de reconnaissance (Artificial Immune System for Recognition)
AIS	Système immunitaire artificiel (Artificial Immune System)
AM	Apprentissage machine
APC	Cellules présentatrices d'antigène (Antigen Presentation cell)
ARB	La boule artificielle de reconnaissance (Artificial recognition Ball)
AT	Seuil d'affinité (Threshold Affinity)
ATS	Seuil d'affinité scalaire (Threshold Affinity Scalar)
BCR	Récepteur de cellule B (B Cell receptor)
BDD	Base de données
BPMLP	Une méthode nommée : back propagation (BP) multilayer perceptron (MLP)
CSA	Algorithme de sélection clonale (Clonal Selection Algorithm)
DT	Théorie de danger (Danger Theory)
ECG	Electrocardiogramme
ED	L'évolution différentielle
EEG	Electroencéphalogramme
GA-MLP	Une méthode nommée : genetic algorithm (GA) multilayer perceptron MLP
GARP	Générateur automatique de règles de production
HCA	Une méthode de classification nommée : Hierarchical Covering Algorithm
HMMs	Modèles de markov cachés
IA	Intelligence artificielle
IE	L'intelligence en essaim



KFDA	Une méthode de classification nommée : Kernel fisher discriminant analysis
KNN	K-plus proches voisins
MC	Cellule mémoire (memory cell)
MHC	Des molécules de classification nommée : complexe majeur d'histocompatibilité
MNIST	Une base de données : contient des caractères manuscrits
NBR	Nombre de règles générées
NBRO	Nombre de règles optimisées
NIS	Système immunitaire naturel (Natural Immune System)
NIST	Une base de données : contient des caractères isolés et du texte écrits
NIT	Théorie des réseaux immunitaires (network Immune theory)
NMC	Classifieur du plus proche moyen
NN	Neurone naturel
NNC	Classifieur du plus proche voisin
NNs	Réseaux de neurones (Neurals Networks)
NSA	Algorithme de sélection négative (negative selection Algorithm)
OCR	Reconnaissance optique de caractères (Optical Character Recognition)
PE	Programmation évolutionnaire
PG	Programmation génétique
PSO	L'optimisation par essaim de particule (Practical Swarm optimization)
QRBS	Système à base des règles qualitative (Qualitative Rules-Based System)
RdF	Reconnaissance de formes
RNA	Réseaux de neurone artificiel
RNN	Réseaux de neurone naturel
SE	Les stratégies d'évolution
SI	Système immunitaire
SUCRAGE	Un système nommé : Supervised Classification by Rules Automatic Generation
SVM	Machine à vecteur support (Support Vector Machine)
TAL	Traitement Automatique des Langues
TCR	Récepteur de cellule T (T Cell Receptor)
TDB	Théorie de la décision bayésienne

# Introduction Générale

**L**A recherche des formes dans un ensemble de données est une pratique très ancienne. Par exemple, c'est grâce aux observations de Tycho Brahe au XVI<sup>ème</sup> siècle que Jhann Kepler a pu élaborer sa théorie sur le mouvement des planètes. De même, l'observation de la régularité des spectres atomiques a joué un rôle clé dans le développement de la physique quantique au début du XX<sup>ème</sup> siècle. On peut multiplier de tels exemples et remonter très loin dans notre passé. Le domaine de la reconnaissance de formes RdF s'occupe de l'extraction automatique des régularités d'un ensemble de données en utilisant des algorithmes informatiques et aussi de l'utilisation de ces régularités pour prendre des mesures telles que la classification des données en différentes catégories [BIS 95].

Les recherches dans le domaine de RdF se sont concentrées à trouver des algorithmes qui peuvent résoudre sur ordinateur les problèmes qui sont intuitivement résolus par les êtres humains. L'utilisation de l'ordinateur a pour objectif principal d'automatiser les tâches effectuées par les êtres humains, pour accélérer leurs traitements, éviter de faire des erreurs comme les erreurs de calcul, éviter les tâches répétitives et fastidieuses portant sur de gros volumes, etc. L'homme a tenté de modéliser le savoir humain par un ordinateur, c'est-à-dire visionner par un ordinateur, mais cette vision demeure confrontée à la réalité, le doute, le vague, l'imprécis ou l'incertain qui font partie de sa pensée. L'introduction des techniques de représentation des connaissances approximatives tente d'automatiser cette démarche soit avec des approches numériques ou bien symboliques. À cet effet, l'approche que nous proposons consiste à élaborer de nouveaux outils qui permettent de représenter des connaissances imprécises et incertaines afin de mieux simuler le raisonnement humain.

Résoudre un problème de RdF revient à construire un système permettant de donner les meilleures performances pour une application donnée, avec une optimisation de chacune des étapes du système pour donner les meilleures solutions possible. Généralement, un système de RdF peut être divisé en quatre étapes principales [WOZ 03] : l'étape de préparation des données (prétraitement suivi d'une extraction des primitives), l'étape d'apprentissage, l'étape de classification et l'étape de post traitement. Comme ces étapes sont successives et les sorties de chacun d'eux sont utilisées par l'étape qui suit, le choix et l'optimisation d'une telle étape influencent sur les autres étapes. Par exemple, le choix du classificateur influence sur le choix d'étape d'apprentissage utilisée.

La RdF peut être vue comme un problème de classification où il s'agit de reconnaître des formes par l'attribution de la classe la plus pertinente à chaque forme d'entrée. Elle est aussi vue comme un problème d'apprentissage, le système doit généralement apprendre tout seul comment distinguer entre les formes de différentes classes. En plus des deux

premiers, l'extraction des caractéristiques constitue également un axe de la RdF où le bon choix des primitives permet d'améliorer le rendement du système et cela quel que soit le type de classifieur utilisé ou la méthode d'apprentissage appliquée. Durant les dernières décennies, la plupart des études élaborées dans ce domaine sont basées sur les techniques d'extraction de caractéristiques ou sur les méthodes de classification [LIU 03 ; LIU 04].

Plusieurs méthodes et algorithmes informatiques qui ont été développés pour la résolution des problèmes de la RdF trouvent également leur équivalent dans la nature. Le développement rapide de la technologie et l'augmentation de la puissance des ordinateurs ont permis d'élaborer de nouvelles méthodes s'adaptant à ce progrès. La complexité de certains problèmes est telle que les méthodes algorithmiques standards ne sont pas applicables. De nouvelles voies ont alors été explorées par les chercheurs, les plus notables émergent du calcul bio-inspiré. Ce concept a permis de générer plusieurs approches appelées bio-inspirés utilisés comme l'une des notions clés de notre thèse.

Pour rendre la communication homme-machine de plus en plus facile et souple, des techniques numériques permettant la communication avec l'ordinateur par l'écriture manuscrite sont améliorées constamment, dues à leurs grandes importances dans le développement actuel des technologies des ordinateurs, surtout, dans les domaines d'applications qui nécessitent une gestion automatique de l'écriture humaine. Il existe plusieurs domaines dans lesquels la reconnaissance de l'écriture manuscrite est attendue avec impatience, soit pour simuler la lecture humaine ou pour faciliter les tâches humaines. Parmi ces domaines d'applications, on trouve : les systèmes de tri postal, les services des chèques bancaires, et le traitement automatique des documents administratifs. Grâce à la grande importance de ces domaines dans la vie quotidienne, le problème de la reconnaissance de l'écriture, et plus précisément les caractères manuscrits, reste un champ de recherche très actif.

## Problématique

Les principaux problèmes de la reconnaissance de caractères manuscrits sont liés essentiellement à plusieurs facteurs : la pente des images et le bruit introduits par le scanner dans la phase d'acquisition, la variabilité des styles d'écriture et les défauts d'encre causés par les scripteurs et la variabilité et la complexité des formes elles-mêmes. En plus des problèmes standards, l'écriture imprécise et irrégulière peut rendre certains caractères illisibles et peut donc augmenter la confusion entre les caractères similaires.

Pour surmonter ces problèmes et optimiser les performances de classification, de nombreux chercheurs ont tenté d'intégrer le concept flou et les techniques d'optimisation dans le processus de reconnaissance, soit au niveau d'extraction de caractéristiques ou dans la classification. Les études établies dans cette perspective nous guident vers l'une des stratégies suivantes : des méthodes inspirées de l'être humain ou de la nature pour l'optimisation, des théories floues pour traiter les informations imprécises, des méthodes combinées ou des améliorations de méthodes classiques pour la classification.

Le but de la reconnaissance de l'écriture est de transformer un texte écrit en une représentation compréhensible par la machine et facilement reproductible par un logiciel de traitement de texte. Cette tâche n'est pas triviale, car les mots possèdent une infinité de représentations. Chaque personne produit une écriture qui lui est propre. Suivant le type d'écriture qu'un système doit reconnaître (manuscrit, cursif, non contraint ou imprimé) et les opérations à effectuer, les résultats peuvent varier notablement.

Jusqu'à maintenant, il n'y a pas de système de reconnaissance de caractères manuscrits qui assure à 100% la classification et la reconnaissance de leurs entrées. Un système donné peut adapter aux différents caractères en entrées, mais pas à toutes les variations d'écriture. L'adaptation aux changements des styles d'écriture reste une question clé et une problématique majeure dans la reconnaissance de caractères manuscrits. Dans ce manuscrit, nous avons essayé de traiter et corriger cette problématique par la proposition d'un nouveau système afin d'augmenter les performances.

## Objectif

L'objectif de notre travail est de concevoir un système de reconnaissance de caractères manuscrits s'appuyant sur des techniques partant des phénomènes naturels tels que les systèmes immunitaires naturels, et du raisonnement humain tel que les techniques de pondération basées sur des règles purement qualitatives, afin d'améliorer le taux de reconnaissance sur différents types de caractères.

Le problème principal traité dans notre travail consiste à améliorer la performance de classification des caractères manuscrits (chiffres et lettres), en améliorant la qualité des données d'apprentissage et réduisant l'incertitude dans la prise de décision.

## Contributions principales

Notre étude tourne autour du problème de classification et vise à améliorer la reconnaissance des caractères manuscrits (chiffres et lettres). Nous avons proposé dans cette thèse un système de reconnaissance de caractères manuscrits hors ligne basé sur une approche bio-qualitative qui figure parmi les propositions rares de la littérature. Cette approche est une combinaison de deux principaux sous-systèmes, à savoir, le système immunitaire artificiel (AIS) et le système à base des règles qualitatives (QRBS), pour faire face à la variabilité du style d'écriture. Le premier sous-système est appliqué pour réduire la dimensionnalité de l'ensemble d'apprentissages (optimisation) tandis que le deuxième est utilisé pour diminuer l'incertitude et l'imprécision des caractères dans la prise de décision.

Dans le contexte de la réduction des données, nous exploitons la capacité de l'AIS pour sélectionner les meilleurs candidats à partir de l'ensemble d'apprentissages. Cette étape est cruciale, elle peut réduire de manière significative le nombre de règles du QRBS, la complexité de calcul et le stockage mémoire.

Dans le contexte de la prise de décision, nous exploitons les avantages de QRBS dans la modélisation des informations imprécises pour traiter le problème de chevauchement et aussi pour éviter ou corriger les affectations arbitraires lorsque les caractéristiques des différents caractères sont presque similaires.

L'utilisation du principe de généralisation (de l'AIS) et de modélisation qualitative du raisonnement (du QRBS) au sein d'un système de reconnaissance de caractères manuscrits est l'une des particularités de notre travail.

## Organisation de la thèse

Cette thèse est organisée en deux grandes parties. La première partie comportant les chapitres 1,2, 3 et 4, est dédiée aux éléments d'état de l'art. Elle a pour objectif de

présenter le fruit de notre étude de synthèse concernant les domaines aussi vastes et hétéroclites que la reconnaissance de formes, la reconnaissance d'écriture manuscrite, les approches bio inspirées, les systèmes immunitaires artificiels et les approches qualitatives. Donc, nous proposons un tour d'horizon de ces domaines qui ne prétend pas l'exhaustivité.

La deuxième partie concerne directement la contribution de cette thèse. Elle a pour objectif de mettre en exergue comment certaines caractéristiques de systèmes bio-inspirés et systèmes flous, en particulier les systèmes immunitaires artificiels et les systèmes à base des règles qualitative, ont été exploitées et mises à profit de la reconnaissance de caractères manuscrits par l'apprentissage et la classification. Certains éléments de la théorie ont été introduits au fur et à mesure des nécessités pour faciliter la compréhension des modèles de transposition développés.

De manière plus détaillée, le contenu de cette thèse est organisé en cinq chapitres de la façon suivante :

**Le chapitre 1** présente un aperçu sur la reconnaissance de formes et les étapes suivies pour la réalisation d'un système de reconnaissance en général.

**Le chapitre 2** est consacré à l'état de l'art dans le domaine de la reconnaissance d'écriture manuscrite, et plus précisément les caractères manuscrits. Dans ce chapitre, nous décrivons en détail les étapes nécessaires au développement d'un système de reconnaissance d'écriture manuscrite, les méthodes utilisées pour la classification des caractères en précisant les méthodes bio inspirées, et les difficultés et les problèmes liés à cette reconnaissance.

**Le chapitre 3** décrit les systèmes immunitaires artificiels (AIS) avec : une présentation générale du système immunitaire naturel (NIS) et les différents mécanismes utilisés dans l'identification et la détection des intrus, les différents algorithmes immunitaires artificiels disponibles ainsi que les différents domaines d'application.

**Le chapitre 4** expose l'approche qualitative et met en œuvre d'une part, la méthode de construction de règles, le moteur d'inférence, l'interface homme-machine et la probabilité qualitative pour le traitement de l'incertitude et l'imprécision, et d'autre part, la prise de décision optimale pour éviter le problème de saturation de la base de connaissances qui affecte la précision du taux de classification.

**Le chapitre 5** montre en détail les approches proposées qui se résument dans : une approche bio inspirée, une approche bio inspirée floue et une approche bio qualitative ; pour la reconnaissance des caractères manuscrits. Nous avons également présenté et discuté les résultats obtenus.

Nous terminons cette thèse par une conclusion générale et quelques perspectives de nos recherches dans ce domaine.

# La reconnaissance de formes

## 1.1 Introduction

Ce chapitre est consacré aux concepts généraux de la reconnaissance des formes et la classification. Dans cette partie de la thèse, nous décrirons tout d'abord une présentation du domaine de la reconnaissance de formes. Ensuite, nous portons sur les différents domaines d'application. Nous présenterons après le processus général de la reconnaissance de formes et quelques aspects fondamentaux de la classification. Finalement, nous présenterons quelques classifieurs de base, et évoquerons les méthodes de combinaison des classifieurs.

## 1.2 Présentation de la reconnaissance de formes

La reconnaissance de formes (RdF) est l'une des rares disciplines qui ne nécessitent pas une définition formelle dans le cadre général. Les deux termes « reconnaissance » et « forme » sont très utilisés dans la vie quotidienne. Dans l'ingénierie, elle concerne de l'étude des systèmes automatiques ou semi-automatiques capables de reconnaître les formes qu'on leur présente.

Watanabe a défini une forme comme [WAT 85] : « *l'opposé du chaos ; c'est une entité vaguement définie, à laquelle on peut associer un nom* ». Selon bishop [BIS 95], le terme reconnaissance des formes englobe une grande gamme de problèmes de traitement d'information ayant de grandes importances pratiques. Les formes à identifier dépendent de l'application : ils peuvent être des images, des lettres, des paroles, des cibles militaires, des formes d'onde d'un signal, etc. En des termes informatiques, une forme est un ensemble de valeurs, appelé *attributs*, auquel est associé un nom (ou étiquette), qui est leur *classe*. Plusieurs formes peuvent avoir la même classe, on dit alors que ce sont les *exemples* ou *réalisations* de la classe. D'une façon générale, les formes se définissent comme un ensemble de caractéristiques ou de comportements qui peut être différencié et classé à partir d'une collecte d'informations.

Malgré la diversité des systèmes de RdF qui existent, l'être humain reste le plus parfait parmi eux. La capacité de reconnaissance que nous pouvons accomplir sur des formes à grande variabilité est et restera toujours impressionnante. Nous distinguons sans effort, par exemple le rire du cri, le miaulement de l'aboïement, le bruit du bateau de celui de l'avion, le visage d'un bébé de celui d'un vieux, etc. Nous pouvons rappeler des centaines de ces exemples.

La RdF (parfois la reconnaissance de motifs) est un domaine pluridisciplinaire qui se situe au croisement de différentes disciplines. On cite parmi ces derniers : les mathématiques (probabilités et statistiques), l'électronique, les sciences de l'ingénieur, la biologie, l'informatique et l'intelligence artificielle [GAG 05 ; THE 03]. C'est à partir des années soixante que la RdF est devenue une discipline spécifique. L'objectif principal de cette discipline est de réaliser des systèmes informatisés qui simulent les activités humaines de perception, de reconnaissance et de compréhension comme : la reconnaissance de l'écrit, de la parole, l'interprétation de scènes, la robotique, la reconnaissance de signaux médicaux, etc. La RdF ne se limite pas seulement à la perception visuelle, mais inclut également tous les types de perception artificielle exigeant de reconnaître des formes de données [CAR 99]. Cela implique aussi une certaine pluridisciplinarité pour comprendre l'aspect physique des capteurs (acquisition), les aspects mathématiques de la classification (décision), ceux relatifs à l'informatique (apprentissage).

Aujourd'hui, les progrès scientifiques et techniques nous permettent, à l'aide des machines, d'essayer de simuler les activités humaines. Il y'a beaucoup de cas particuliers où les tâches de reconnaissance simples effectuées par l'homme peuvent être confiées par la machine. De ce fait, l'ordinateur a un objectif d'automatiser les tâches habituellement effectuées par l'homme. Comme nous avons vu dans le paragraphe précédent, la plupart des activités humaines reposent sur une faculté importante de notre cerveau : pouvoir distinguer entre les formes. Pour pouvoir automatiser cette tâche, l'ordinateur doit obligatoirement acquérir la faculté de reconnaître les formes [DEN 06].

Le problème que la RdF cherche à résoudre est d'associer une classe à une forme inconnue. On considère souvent la RdF comme un problème de classification, c'est-à-dire qu'on essaye de trouver la fonction qui affecte à toute forme inconnue sa classe la plus pertinente. Elle s'intéresse donc au développement des systèmes intelligents destinés à la perception et la prise de décision [THE 03].

Les systèmes de RdF intègrent toute la chaîne perception-reconnaissance depuis l'acquisition des données brutes jusqu'à la compréhension élaborée de ces données. Ces dernières ont entre-temps subi de nombreuses transformations. Prenons l'exemple de l'OCR (Optical Character Recognition) qui utilise pour la lecture des caractères à partir des images de textes imprimés. Dans cet exemple et afin de reconnaître un caractère d'image, il faut d'abord segmenter l'image en zones. Puis, sélectionner les zones de texte et découper celles-ci en lignes. Enfin, découper les lignes en mots et les mots en caractères. Au cours de ces transformations, le système ou bien la machine font des traitements d'image et de signal. De ce fait, la RdF fait appel à d'autres disciplines connexes telles que le traitement d'image et de signal, l'intelligence artificielle ou le traitement automatique des langues (TAL).

### 1.3 Les domaines d'application

Pratiquement, les objets à reconnaître sont de type signal ou image, ce qui implique deux grandes familles d'applications : la reconnaissance des signaux et la reconnaissance des formes dans les images numériques. Dans la première famille, on essaye d'automatiser l'interprétation des signaux, et dans la deuxième, on essaye d'automatiser le processus de la vision humaine. Le tableau suivant montre les différents domaines d'applications de RdF.

Tableau 1.1 – Les différents domaines d’application de la RdF.

Famille	Domaines d’études	Applications
<b>Traitement Du Signal</b>	Reconnaissance de la parole	Bureautique, commande vocale, ordinateur sans clavier ni souris
	Reconnaissance du locuteur	Banques, commerces
	Électrocardiogramme, électro-encéphalogramme	Médical
<b>Traitement Des Images</b>	Reconnaissance des caractères, du manuscrit	saisie de texte, bureautique, tri postal, compression télécopie, chèque
	Reconnaissance des signatures	Banques, commerces
	Reconnaissance des empreintes digitales, des visages	Banques, commerces, police
	Analyse de radiographie, échographie, reconnaissance chromosomes, comptage globules	Contrôles systématiques de santé
	Détection de défauts circuits intégrés, pièces métalliques, manufacturés	Contrôle de qualité industrielle
	Identification d’objets	Tri d’objets industriels, surveillance militaire
	Localisation d’objets	Guidage de robots industriels, guidage de missiles
	Analyse d’image de satellite	Météorologie, agriculture, ressources terrestres, surveillance militaire
	Analyse de photos aériennes	Agriculture, surveillance militaire

### 1.3.1 Reconnaissance des formes sur signaux temporels

- Signal de parole : reconnaissance de la parole (Qu’est ce qu’il a dit ?),
- Reconnaissance du locuteur (qui parle ?),
- Signaux biomédicaux : EEG (électroencéphalogramme), ECG (électrocardiogramme)...
- Surveillance d’instruments, diagnostic de panne,
- ...

### 1.3.2 Reconnaissance des formes dans les images numériques

- Lecture automatique de caractères
- Reconnaissance d’empreintes digitales
- Radiographies
- Analyse de scènes, interprétation d’images,
- ...



## 1.4 Processus de la reconnaissance de formes

Dans le problème de reconnaissance, il y a un petit nombre de notions qui apparaissent constamment. Ces notions sont appelées les constantes fondamentales de la RdF. Nous allons les expliquer dans le schéma de la figure 1.1 qui peut être considéré comme une composante de base de tout problème de reconnaissance.

### 1.4.1 Schéma fondamental

D'une façon schématique, on peut dire que toute action de reconnaissance de formes met en jeu trois ensembles  $(O, R, \Omega)$  et deux applications  $(\psi, \xi)$  expliquées comme suit [BOU 98] :

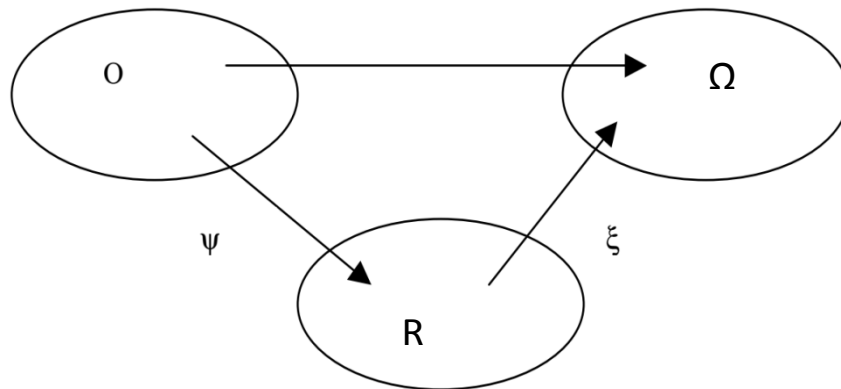


FIGURE 1.1 – Schéma élémentaire de RdF

#### A. L'ensemble $O$

$O$  désigne l'ensemble des objets à reconnaître. C'est une partie du monde qui nous entoure, focalisé sur le problème à traité. Les données d'un problème particulier de RdF, nous permettent de spécifier les éléments de  $O$ .

#### B. L'espace de représentation $R$

$R$  est l'ensemble ou bien l'espace des représentations des objets de  $O$ . Il permet de représenter ou de décrire ces objets. Cette représentation doit être sous forme numérisée afin que l'ordinateur puisse la prendre en compte et la traiter. Grâce à des capteurs physiques ou des descriptions simples, des prises de mesures peuvent être réalisées et permettront de représenter chacun de ces objets par une suite des éléments organisationnels (caractéristiques ou descripteurs) qui peuvent être :

- des mesures, des valeurs numériques, des niveaux de gris pour les pixels, ...
- des formes élémentaires (des critères de formes, des éléments d'un alphabet, etc) qui sont elles-mêmes facilement représentables par une liste d'entités numériques.

Chaque élément de  $R$  est donc représenté par un ensemble de caractéristiques pris dans un certain ordre, et on l'appellera vecteur de caractéristiques. Cette représentation notamment s'utilise pour la construction de  $\xi$ , et d'une façon générale, permet de faciliter la résolution du problème de reconnaissance. En général, la relation existant entre les éléments de  $R$  est une relation de voisinage. Elle peut être associée à une distance, à une similarité, à une ressemblance ou encore à une proximité.

L'ensemble  $R$  dépend nécessairement de  $\xi$  et de  $\psi$  :

- Avant de mettre n'importe quelle caractéristique dans  $R$ , il faut d'abord que  $\psi$  nous permette de la saisir et de la mémoriser.
- Le choix de  $R$  est associé strictement au choix de  $\xi$  ; on parle ici de la partie algorithmique du problème de RdF. Le choix pourrait être la recherche du meilleur rapport entre la pertinence de l'information et la limitation de la complexité.

### C. L'espace d'interprétation $\Omega$

$\Omega$  désigne l'ensemble des classes, il souvent être réduit à un ensemble de noms ou d'étiquettes. En général, on le présente par un ensemble fini de classes  $c : \Omega = \{c_1, \dots, c_n\}$ . Par exemple en OCR (Reconnaissance Optique de Caractères), chaque élément  $c$  considère le nom d'un chiffre, d'une lettre majuscule ou minuscule, d'un des signes (de ponctuation ou non), d'un caractère, qu'on peut voir sur les claviers des ordinateurs ou d'autres machines à écrire.

$\Omega$ , aussi bien et  $R$ , peut être aussi considéré comme un espace, c'est-à-dire muni d'une structure qui se résume souvent par des relations de proximité ou de ressemblance entre leurs éléments.

L'objectif de la RdF est d'associer une interprétation (nom/étiquette) de  $\Omega$  à chaque entité de  $O$ . Dans la plupart des cas, on peut considérer que les deux ensembles  $O$  et  $\Omega$  sont connus. Cette considération n'est pas vraie pour  $\Omega$  en classification non supervisée par un être humain, et dans certains problèmes de diagnostic [DUB 90].

En reconnaissance non supervisée (automatique), nous devons passer par l'espace de représentation (qui permet de fournir des données convenables par les machines utilisées) et, pour réaliser l'association mentionnée ci-dessus, nous avons besoin de construire, en plus de  $R$ , les deux applications suivantes :

$$\psi : O \rightarrow R$$

$$\xi : R \rightarrow \Omega$$

### D. L'application $\Psi$

Nous parlons ici d'application pour la facilité du langage. Les activités des capteurs (caméras, scanners, appareils de mesures) peuvent saisir des informations qui sont accessibles ou non par l'homme (photo, infrarouge). Les informations issues de ces capteurs sont des signaux bruts, et ne sont pas toujours exploitables par la machine. Pour cela, des procédures de prétraitement et/ou de paramétrisation sont alors mises en œuvre.  $\Psi$  peut recouvrir ces activités.

### E. La fonction d'identification $\xi$

On entend par une "identification" une application  $\xi$  de l'espace  $R$  dans l'espace  $\Omega$ . « *Le but de la reconnaissance de formes est d'effectuer cette application de manière constructive, en d'autres termes de trouver un algorithme qui l'effectue pour toute valeur de  $R$*  » [SIM 84].

Il convient de noter que l'application  $\xi$  définie sur  $R$  donne un résultat qui n'est pas toujours une classe  $c$  unique. Ce résultat peut être aussi un couple (ou une famille de couples) de type  $(c, f)$ , par exemple  $f$  peut être une valeur d'appartenance floue ou une probabilité.

## 1.4.2 Vers l'implémentation

### A. Présentation de l'ensemble d'apprentissages

Généralement, dans un processus de RdF, un échantillonnage est effectué de l'espace de représentation  $R$  sous forme d'un ensemble  $P$ , dit *ensemble d'apprentissages*. Ce dernier est souvent considéré soit comme un sous-ensemble de  $R$ , soit comme un sous-ensemble du produit cartésien  $R \times \Omega$ . Mais rien ne garantit a priori que  $P$  soit représentatif de  $R$ . Un bon fonctionnement de  $\xi$ , sur les éléments de  $P$ , ne signifie pas qu'il est bon pour tous les éléments de  $R$ .

### **B. Évolution de l'ensemble d'apprentissage**

L'ensemble d'apprentissages ne considère pas comme une donnée fixée, il s'évolue au cours du traitement de problème posé. Les raisons qui prouvent cette hypothèse sont :

- Le temps peut être un paramètre du problème, car de nouvelles formes peuvent apparaître (des polices nouvellement créées en OCR par exemple). Il y a alors une évolution des formes, donc de leur représentation,
- $P$  est non représentatif ou incomplet, car  $\Omega$  n'était pas parfaitement appréhendé,
- Une possibilité d'agrandir  $P$  est apparue, car des informations et des connaissances externes peuvent valider ou infirmer de nouveaux couples de type  $(c, x)$ .
- L'évolution de la technologie et les capacités de nos machines actuelles nous permettent ce changement alors que dans des périodes antiques c'était impossible.

Cette évolution de  $P$  doit pouvoir s'accompagner d'une évolution de  $\xi$ .

### **1.4.3 Les étapes d'un processus de RdF**

Comme les humains utilisent leurs sensations pour acquérir les connaissances relatives à leur environnement afin de réagir avec lui, de même un système de RdF commence par l'acquisition des données à travers des capteurs. Une représentation des données brutes est ensuite produite et présentée au classifieur qui produit une décision. Un système de RdF de base se compose principalement d'un *capteur*, d'un dispositif d'*extraction des caractéristiques* et d'un *classificateur* (Figure 1.2). Un système plus complet inclura également des étapes de *prétraitement*, d'*évaluation* et de *post traitement*.

## Analogie

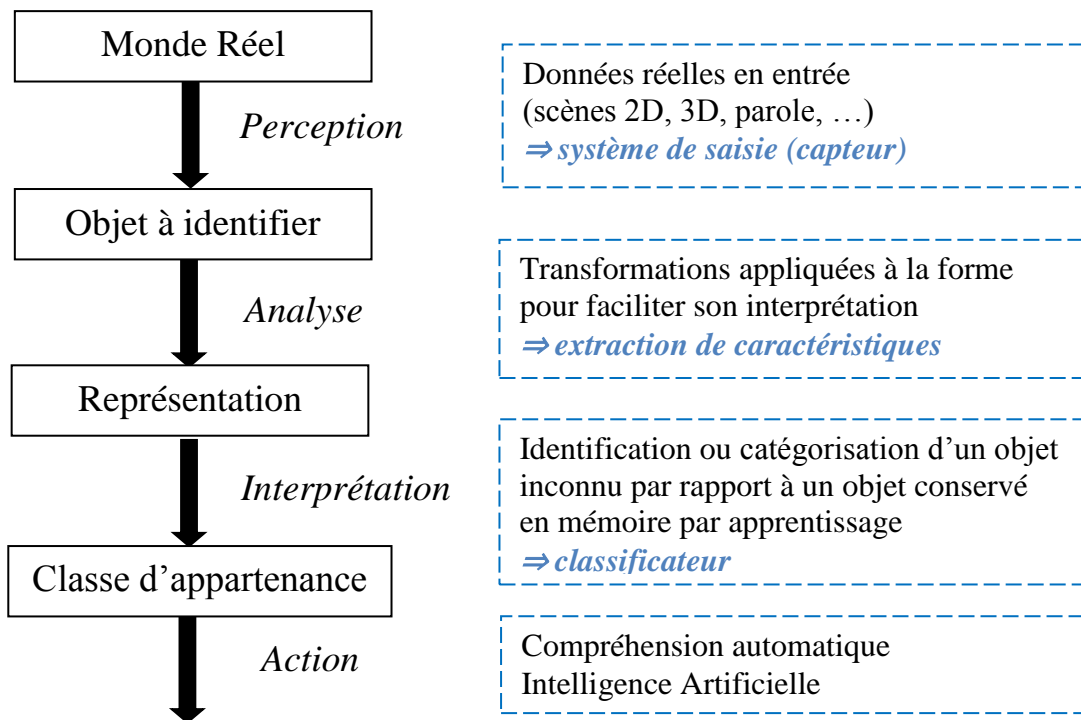


FIGURE 1.2 – Analogie entre un système réel et un système de RdF

À partir de la figure 1.2, on déduit que dans le cas général, il existe trois étapes principales pour la reconnaissance qui sont :

- *L'étape d'acquisition* : l'espace d'observation permet l'acquisition des données à travers des capteurs.
- *L'étape d'extraction des caractéristiques* : le passage de l'espace d'observation vers l'espace de représentation.
- *L'étape de classification* : le passage de l'espace de représentation vers l'espace d'interprétation.

La majorité des systèmes de RdF ont le schéma de fonctionnement suivant [WOZ 03] (figure 1.3).

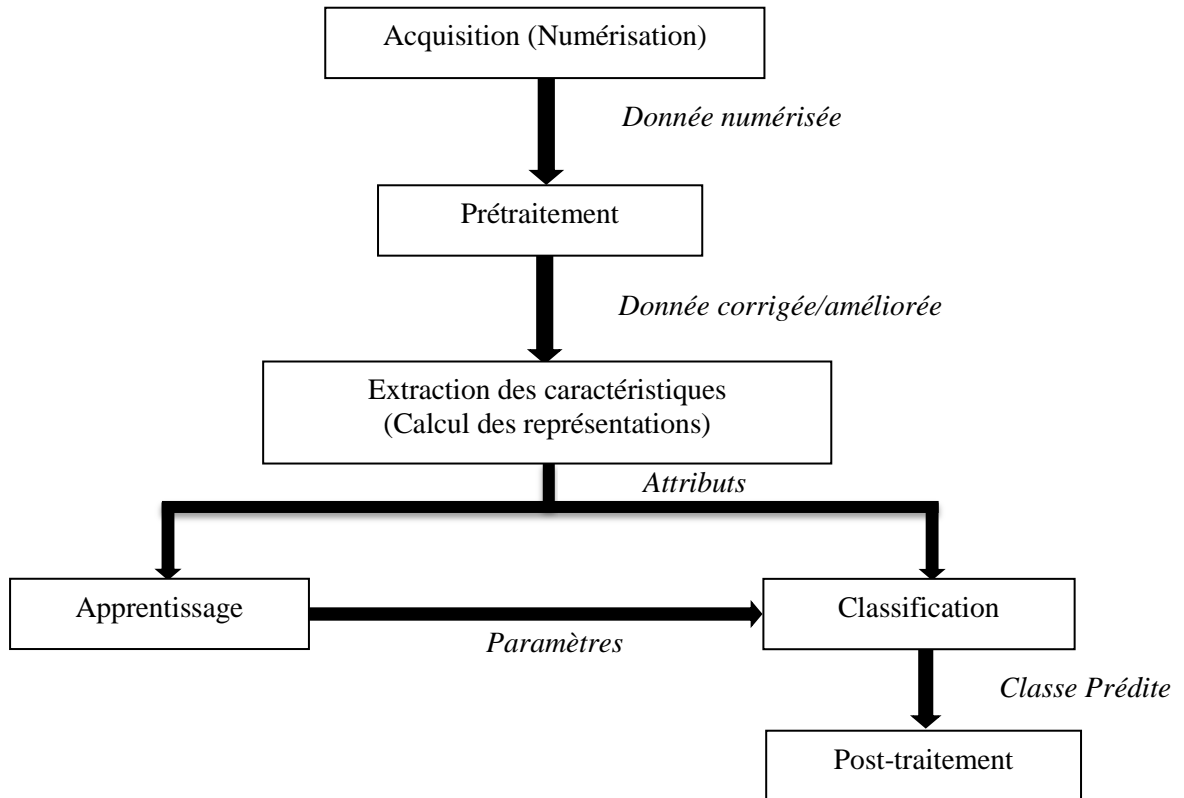


FIGURE 1.3 – Schéma général d'un système de RdF

La figure 1.3 présente la majorité des étapes comportant un processus de RdF. Une description bien détaillée de ces étapes est la suivante [NEM 10] :

#### A. Acquisition des données

Il s'agit de construire une représentation des données, à partir des informations du monde réel, directement manipulable par la machine. Des capteurs (microphone, caméra, instruments de mesure) convertissent les signaux reçus du monde réel en une représentation numérique discrète adaptée au système de traitement utilisé. L'opération se résume par la transformation d'un signal analogique en un signal numérique. L'espace résultant, appelé *espace de représentation*, a une dimension  $\mathbf{R}'$  très grande lui permettant de disposer du maximum d'informations sur les formes numérisées.

La qualité du matériel d'acquisition peut influencer directement sur le pouvoir décisionnel du classifieur, car le bruit dans une image transmise par ce dernier peut fausser les résultats. De plus, les opérations de prétraitement les plus évoluées ne permettent pas une élimination totale du bruit.

Il existe deux types d'acquisition selon la manière d'utiliser l'image :

- *Acquisition physique*. Le premier niveau d'acquisition, le scanneur en l'occurrence, est relié à l'ordinateur et est piloté par un logiciel. Ce dernier permet de digitaliser l'information analogique et stocke les données numériques dans un fichier graphique.
- *Acquisition Logique*. Le deuxième niveau d'acquisition, dit logique, consiste à exploiter le fichier graphique pour en extraire l'image. Elle se présente alors sous forme d'un tableau  $M$  à  $n$  lignes et  $m$  colonnes.

### **B. Prétraitement des données**

Cette étape consiste à sélectionner dans l'espace de représentation l'information nécessaire au domaine d'application. Il s'agit généralement : de réduire les bruits introduit par les outils d'acquisition, la qualité du papier, l'encre ou encore l'effet du temps dans les documents anciens, et de chercher à ne garder que l'information significative, de l'objet à classer, par la suppression de la redondance. Le nouvel espace de représentation a une dimension  $\mathbf{R}$  très inférieure à  $\mathbf{R}'$ , mais demeure un espace de grande dimension et contient des informations encore assez primitives.

### **C. Extraction des caractéristiques**

Cette étape permet de fournir un certain nombre de caractéristiques ou paramètres (les célèbres attributs) en utilisant des opérations ou des algorithmes de sélection et/ou d'extraction d'attributs. L'objectif de l'extraction et de la sélection des caractéristiques est d'identifier les caractéristiques qui sont importantes dans la discrimination de classes de formes. Cela signifie de trouver des primitives les plus robustes possible, les plus pertinentes (afin de diminuer la taille du vecteur de primitives) et les moins coûteuses en calcul. Selon le cas, les valeurs de ces primitives peuvent être réelles, entières ou binaires [Ben 02]. Les informations superflues ou inutiles dégradent les performances du classifieur et devront être enlevées, alors que le contenu valable devra être préservé. Les attributs étant limités en nombre, l'espace des paramètres ainsi obtenu est de dimension  $m$  très petite par rapport à  $\mathbf{R}$ .

Les caractéristiques peuvent être quantitatives ou qualitatives [KUN 04]. Les caractéristiques discrètes avec un grand nombre de valeurs possibles sont traitées comme quantitatives, tandis que les caractéristiques qualitatives (catégoriques) sont celles avec un petit nombre de valeurs possibles, avec ou sans graduations. Bien que la relation entre le taux d'erreurs, le nombre d'exemples d'entraînement et le nombre d'attributs soit très difficile à établir, il est généralement recommandé d'utiliser un ensemble d'entraînement d'une taille d'au moins 10 fois le nombre d'attributs. Plus le classifieur devient complexe, plus ce rapport devient large [JAI 00]. D'un autre côté, la réduction du nombre d'attributs peut conduire à une perte dans la discrimination du système résultant et ainsi une réduction de la précision du classifieur. Watanabe indique que les attributs doivent être choisis prudemment, puisqu'il est possible de rendre deux formes arbitraires similaires en les encodant avec suffisamment d'attributs redondants [WAT 85].

Il est important de faire la différence entre *sélection d'attributs* et *extraction d'attributs*. Le terme sélection d'attributs fait référence aux algorithmes qui sélectionnent le meilleur sous ensemble d'attributs, à partir de l'ensemble d'attributs de départ. Les méthodes qui créent de nouveaux attributs à partir de transformations ou combinaisons des attributs d'origine sont appelées algorithmes d'extraction d'attributs. En général, l'extraction d'attributs précède la sélection d'attributs : d'abord les attributs sont extraits des données initiales, puis une partie de ces attributs est sélectionnée. Le choix entre extraction d'attributs et sélection d'attributs dépend du domaine de l'application ainsi que de l'ensemble d'entraînement disponible. La sélection d'attributs permet de faire des économies dans les coûts des mesures puisque certains attributs sont éliminés et les attributs restants gardent leur interprétation physique originale. D'un autre côté, les attributs transformés générés par l'extraction d'attributs peuvent fournir de meilleures qualités discriminantes, mais ces nouveaux attributs peuvent ne pas avoir de signification physique claire.

### **D. Apprentissage**

L'apprentissage consiste à apprendre les caractéristiques communes aux classes de

formes de manière à bien distinguer les familles homogènes de formes. La mise en œuvre des classifieurs implique donc la proposition à l'avance d'une forme générale du modèle et l'emploi d'une phase d'apprentissage pour estimer les paramètres inconnus de ce modèle. L'apprentissage va permettre d'optimiser ces paramètres pour le problème à résoudre, en utilisant des données d'entraînement. Lorsque les données d'entraînement sont préalablement classées, l'apprentissage est dit *supervisé*, sinon il est *non supervisé* [BIS 95]. Alors, l'apprentissage s'effectue selon plusieurs formes [NEM 10] :

- *Apprentissage supervisé*. Dans l'apprentissage supervisé, un échantillon représentatif de l'ensemble des formes à reconnaître est fourni au module d'apprentissage. Chaque forme est étiquetée par un opérateur appelé superviseur ou professeur. Cette étiquette permet d'indiquer au module d'apprentissage la classe dans laquelle le professeur souhaite que la forme soit rangée. Cette phase d'apprentissage consiste à analyser les ressemblances entre les éléments d'une même classe et les dissimilarités entre les éléments de classes différentes pour en déduire la meilleure partition de l'espace des représentations. Les paramètres décrivant cette partition sont stockés dans une table d'apprentissage à laquelle le module de décision se référera ensuite pour classer les formes qui lui sont présentées.
- *Apprentissage non-supervisé*. Dans l'apprentissage non-supervisé (ou clustering), il n'y a aucun superviseur explicite et le système forme des clusters (ou des groupements) des exemples d'entrée (formes non étiquetées). L'étape de la classification va se charger d'identifier automatiquement les formes appartenant à une même classe.
- *Apprentissage par renforcement*. La façon la plus typique pour apprendre un classifieur est de lui présenter un exemple, de calculer sa tentative et d'utiliser la sortie désirée pour l'améliorer. Dans l'apprentissage par renforcement, l'adaptation du classifieur s'effectue seulement en lui indiquant si sa réponse est vraie ou fautive. Il s'agit donc de lui dire c'est juste ou c'est faux sans dire pourquoi.

### E. Test

L'étape de test permet d'évaluer la performance du classifieur pour un apprentissage donné. C'est une étape importante, car elle peut mettre en cause le choix des primitives ou le choix de la méthode d'apprentissage. En effet, il est difficile de trouver a priori les primitives pertinentes et la méthode d'apprentissage la plus adaptée au problème posé, d'où l'utilité de procéder par itérations successives. Ces itérations consistent à extraire des primitives jugées utiles au problème de reconnaissance à résoudre et à tester la performance du système avec cet ensemble de primitives. Au fur et à mesure que les performances du système souhaitées ne sont pas atteintes, alors il suffit de trouver à nouveau une nouvelle famille de primitives ou de combiner les primitives extraites avec de nouvelles primitives [BEN 02].

Le test d'une donnée se déroule en trois étapes [SAI 01] :

- Acquisition des observations de l'objet à tester ,
- Extraction des mêmes caractéristiques utilisées lors de l'apprentissage ,
- Classement parmi les classes  $C_1, \dots, C_c$  par la règle de décision.

Les phases d'apprentissage et de test sont réalisées préalablement à l'intégration du module de décision dans le système de reconnaissance. Dans tous les cas, on peut permettre au système de reconnaissance de répéter les phases d'apprentissage et de test tant qu'on n'a pas atteint les performances désirées. Le calcul de cette performance est le résultat du classifieur utilisé.

### F. Classification

La classification est l'élaboration d'une règle de décision qui transforme les attributs caractérisant les formes en appartenance à une classe (passage de l'espace de codage, vers l'espace de décision). En utilisant les modèles (paramètres) obtenus lors de l'apprentissage, le classifieur assigne à chaque forme inconnue sa ou ses formes les plus probables. Pour construire un classifieur, il existe quatre approches : *basée sur les prototypes, structurale, statistique et bio inspirée*. Une description détaillée de ces approches est présentée dans le deuxième chapitre.

*Mesures de performances d'un classifieur* : Il est important de savoir à quel degré notre modèle s'en tire après, et même durant, la phase de conception. Pour une entrée donnée, un classifieur peut générer les réponses suivantes [GAS 05] :

- **Une reconnaissance** : dans ce cas, le classifieur identifie bien l'entrée, et il lui attribue sa classe appropriée.
- **Un rejet** : le classifieur n'a pas pu identifier cette entrée.
- **Une substitution (erreur)** : le classifieur attribue à l'entrée une autre classe.

En général, quand les échantillons étiquetés à disposition sont suffisamment nombreux ils sont séparés en deux parties disjointes et en respectant les proportions par classes de la base initiale. Une partie sert pour former la base d'apprentissage et l'autre pour former la base de test. Le découpage le plus courant est de 2/3 pour l'apprentissage et 1/3 pour la base de test. La performance d'un classifieur peut être mesurée en calculant les trois taux suivant :

$$\text{Taux de reconnaissance} = \frac{\text{Nombre d'individus bien classés}}{\text{Nombre total d'individus}} \quad (1.1)$$

$$\text{Taux de rejet} = \frac{\text{Nombre d'individus rejetés}}{\text{Nombre total d'individus}} \quad (1.2)$$

$$\text{Taux de substitution} = \frac{\text{Nombre d'individus mal classés}}{\text{Nombre total d'individus}} \quad (1.3)$$

La répartition des exemples bien et mal classés est illustrée dans l'équation suivante :

$$\text{Base de test (Nombre total d'exemples)} = \quad (1.4)$$

nombre d'individus mal classés + nombre d'individus bien classés

### G. Post traitement

Le rôle de la phase de post traitement consiste à valider et à corriger les erreurs de l'étape de reconnaissance, en tenant compte par exemple des informations lexicales, syntaxiques, et sémantiques propres aux formes [OLL 99]. Pour un système de reconnaissance de textes manuscrits, le classifieur se charge de classer chaque caractère séparément, alors que le post traitement applique un correcteur orthographique sur tout le texte pour valider et éventuellement corriger le résultat de la classification. Bien que facultative, cette phase permet d'améliorer considérablement la qualité de la reconnaissance [FAR 09].

- *Utilisation de l'information lexicale*. Le contrôle lexical vérifie la présence d'un mot dans un lexique de référence. Si la reconnaissance s'est effectuée au niveau caractère, le post traitement consistera, par exemple à vérifier à l'aide d'un dictionnaire l'exactitude des mots obtenus en combinant les différentes hypothèses fournies par l'étape de reconnaissance. On attribue ensuite à chaque mot présent dans le lexique une valeur de confiance qui dépend de celles attribuées par l'étape de reconnaissance aux différents caractères composant ce mot.



- *Utilisation de l'information syntaxique.* L'utilisation d'un ensemble de règles grammaticales permet de prendre en compte le contexte syntaxique d'un document afin de confirmer ou infirmer la séquence de mots proposés. Cette approche est souvent appliquée dans les systèmes de lecture des montants de chèques bancaires, pour lesquelles les règles syntaxiques ne sont pas trop nombreuses [OLL 99].
- *Utilisation de l'information sémantique.* Lorsque la nature du problème le permet, il est également possible de mettre en corrélation des résultats obtenus sur des champs différents de l'image, mais codant la même information (cas des chèques bancaires, des bons de commande ou des enveloppes postaux). Il existe, à l'heure actuelle, un certain nombre de travaux visant à exploiter au mieux la redondance d'information entre les champs numériques et littéraux des chèques bancaires ou entre le code postal et le nom de ville dans les adresses postales.

### 1.4.4 Combinaison des classifieurs

Bien qu'il y ait beaucoup de questions sans réponse au sujet de l'adaptation des classifieurs aux problèmes réels, la combinaison des classifieurs est devenue rapidement répandant et appréciant beaucoup d'attention des communautés de la RdF et de l'apprentissage automatique. Pour résoudre les problèmes de reconnaissance, on doit parfois combiner plusieurs classifieurs. L'intérêt de ces combinaisons consiste essentiellement pour plusieurs raisons [KUN 04 ; JAI 00] :

- Le concepteur du système peut avoir accès à plusieurs classifieurs, développés dans des contextes différents pour des représentations/descriptions totalement différentes d'un même problème.
- On peut avoir plusieurs ensembles d'entraînement. Ces ensembles ont été collectés/extraits à différents moments dans différents environnements, et peuvent être représentés par des paramètres différents.
- Des classifieurs différents, entraînés sur les mêmes données, peuvent présenter de grandes différences dans la qualité de la classification. Chaque classifieur peut avoir son propre sous-espace de paramètres où il est le plus efficace.
- Certains classifieurs comme les systèmes immunitaires montrent des comportements différents selon la phase d'initialisation, ceci est dû à la part aléatoire du processus d'initialisation. Au lieu de ne garder qu'un seul classifieur, on peut combiner les systèmes immunitaires obtenus pour bénéficier des résultats de tous les entraînements.

Sommairement, on peut avoir des ensembles de paramètres différents, des ensembles d'apprentissage différents, des méthodes de classification différentes ou des sessions d'entraînement différentes, tout ceci résulte en un ensemble de classifieurs dont les sorties peuvent être combinées [FAR 06], dans l'espoir d'améliorer la qualité globale de la classification.

Un schéma ou plan de combinaison classique consiste en un ensemble de classifieurs individuels et d'une fonction de combinaison qui combine les sorties de classifieurs individuels pour prendre une décision. Quand les classifieurs doivent-ils être invoqués et comment doivent-ils interagir entre- eux, c'est l'architecture du plan de combinaison qui le détermine. On a trois architectures principales [WEB 02] :

- **L'architecture parallèle.** Tous les classifieurs individuels sont invoqués de façon indépendante, et leurs résultats sont combinés par une fonction de combinaison. Les sorties des classifieurs peuvent être pondérées avant la combinaison.

- **L'architecture en cascade.** Les classifieurs individuels sont invoqués de manière linéaire. Le nombre de classes possibles pour une forme donnée est réduit graduellement. Pour des raisons d'efficacité, les classifieurs imprécis mais peu gourmands en calcul sont invoqués en premier, suivis par des classifieurs plus précis mais aussi plus lents.
- **L'architecture hiérarchique.** Les classifieurs individuels sont combinés dans une structure, qui est similaire aux arbres de décision. Les nœuds de l'arbre peuvent être associés avec des classifieurs complexes qui nécessitent un grand nombre de caractéristiques. L'avantage de cette architecture est la grande efficacité et la flexibilité dans l'exploitation du pouvoir discriminant des attributs.

En combinant ces trois architectures basiques, on peut concevoir des systèmes de classification encore plus complexes.

## 1.5 Conclusion

Nous avons présenté dans ce chapitre une brève description du domaine de la RdF avec l'inclusion de leurs concepts de base, en particulier les domaines d'application et centres d'intérêts, leur processus, ainsi les approches utilisées dans la phase de classification. Ces concepts sont nécessaires et permettent aux lecteurs de suivre les démarches faites dans les chapitres suivants.

Dans le chapitre qui suit, nous aborderons la reconnaissance automatique de l'écriture manuscrite qui est un sous domaine de la RdF où nous contribuons notre travail.

# La reconnaissance de l'écriture manuscrite

## 2.1 Introduction

La reconnaissance de l'écriture est l'un des principaux domaines de la Rdf. Grâce à son grand intérêt en tant que système de gestion automatique des documents utilisant de plus en plus des dispositifs numérisés et des méthodes d'interprétation des documents, ce domaine, en particulier la reconnaissance d'écriture manuscrite, est considéré comme le rêve insaisissable de toute personne qui veut entrer des données dans un ordinateur, soit pour simuler la perception des êtres humains ou bien faciliter leurs tâches.

La reconnaissance de l'écriture manuscrite a attiré l'attention d'un grand nombre de chercheurs depuis plusieurs décennies et reste un champ de recherche très active due à ses nombreuses applications [VIN 02], comme : les systèmes de tri postal [LU 02 ; BRA 03], les services offerts par les systèmes bancaires [GOR 99 ; XU 01], le traitement automatique des formulaires et documents administratifs [HIR 01], ainsi que les applications de reconnaissance d'écriture dite « en ligne » à travers les PDA, les Tablet-PC ou les stylos caméra [CHA 06b].

Un système de reconnaissance dépend principalement de plusieurs facteurs, à savoir : le style d'écriture, la taille de vocabulaire et le nombre de scripteurs. Ces facteurs constituent des paramètres fondamentaux aussi bien pour leur mise en pratique que pour leur complexité de conception, ils se différencient aussi selon différents critères, par exemple leur mode de reconnaissance, c'est-à-dire par la stratégie suivie pour la localisation des unités de base de l'écriture dans les mots et de la nature des informations disponibles pour la reconnaissance. Suivant le type d'écriture qu'un système doit reconnaître (manuscrit, cursif ou imprimé), les opérations à effectuer et les résultats peuvent varier notablement.

L'objectif de ce chapitre consiste d'une part à introduire et présenter un état de l'art du domaine de la reconnaissance de l'écriture, ce qui nous permettra de situer le problème de reconnaissance de caractères manuscrits, et d'autre part d'exposer les différentes approches, méthodes et techniques réalisées, dans ce domaine, depuis plusieurs années. Nous présentons dans ce qui suit une vue globale des principaux concepts, facteurs et critères pilotant un système d'écriture manuscrite.

## 2.2 Les applications pratiques

Plusieurs domaines d'activité comportent des applications pratiques de la reconnaissance de l'écriture, on peut citer :

- *Les banques* : l'authentification de chèques bancaires comme la correspondance entre l'identité de la signature et son signataire, et la correspondance entre les montants et libellé [GOR 99 ; XU 01].
- *Les assurances* : la vérification de clauses de contrats pour les assurances [AUG 00].
- *La poste* : la lecture et le tri automatiques des adresses et du courrier [GRA 03].
- *Les télécommunications* : l'échange à distance de fichiers informatiques .
- *La police et la sécurité* : la reconnaissance de numéros d'immatriculation des véhicules pour le contrôle routier, l'identification du scripteur et l'authentification et l'identification de manuscrits.
- *L'industrie et les affaires* : la reconnaissance de documents techniques et la gestion des stocks.
- *La bureautique* : l'indexation et l'archivage automatique de documents, et pour la publication électronique [LEB 07].
- *L'administration* : la lecture automatique de documents administratifs et la reconnaissance de plans cartographiques [HER 98].

## 2.3 Les systèmes de reconnaissance de l'écriture

Les systèmes de reconnaissance de l'écriture peuvent être classifiés selon deux critères : le mode d'acquisition et le type d'écriture.

### 2.3.1 Reconnaissance selon le mode d'acquisition

Un système de reconnaissance de l'écriture peut être divisé, selon la nature de l'entrée en deux catégories : hors-ligne et en-ligne [GAY 14]. La reconnaissance en ligne est exécutée en même temps avec l'apparition d'écriture. Tandis que, dans le cas hors-ligne, la reconnaissance est faite après l'écriture. Un système de reconnaissance optique de caractères (OCR) est un processus hors-ligne [LAC 13].

#### 2.3.1.1 La reconnaissance hors-ligne (off-line)

La reconnaissance hors-ligne (ou statique, ou encore en différé) est obtenue par la saisie d'un texte, déjà existant, parvenue par un capteur (scanneur ou caméra). Ce dernier dispose d'une image binaire ou en niveaux de gris. Ce mode d'acquisition introduit une difficulté supplémentaire relative à la variabilité du tracé en épaisseur et en connectivité, nécessitant l'application de techniques de prétraitement.

Les systèmes de reconnaissance hors-ligne sont largement utilisés dans des domaines pratiques nécessitant un traitement rapide avec un taux de reconnaissance élevé. Ils sont très utilisés notamment dans la lecture automatique des montants de chèques bancaires [AUG 00 ; GOR 99 ; KAU 00], la lecture d'adresses et le tri automatique du courrier [KIM 98], la reconnaissance automatique de contenu de formulaires précaisés [HER 98], la reconnaissance et l'indexation des collections patrimoniales, des archives et des documents

anciens [LEB 07 ; NIC 06], la lecture des images issues des tableaux blancs [LIW 07], et la recherche d'information dans une base de documents manuscrits [BEN 04]. D'ailleurs, notre étude portera uniquement et plus particulièrement sur ces systèmes, c'est le cadre applicatif de cette thèse.

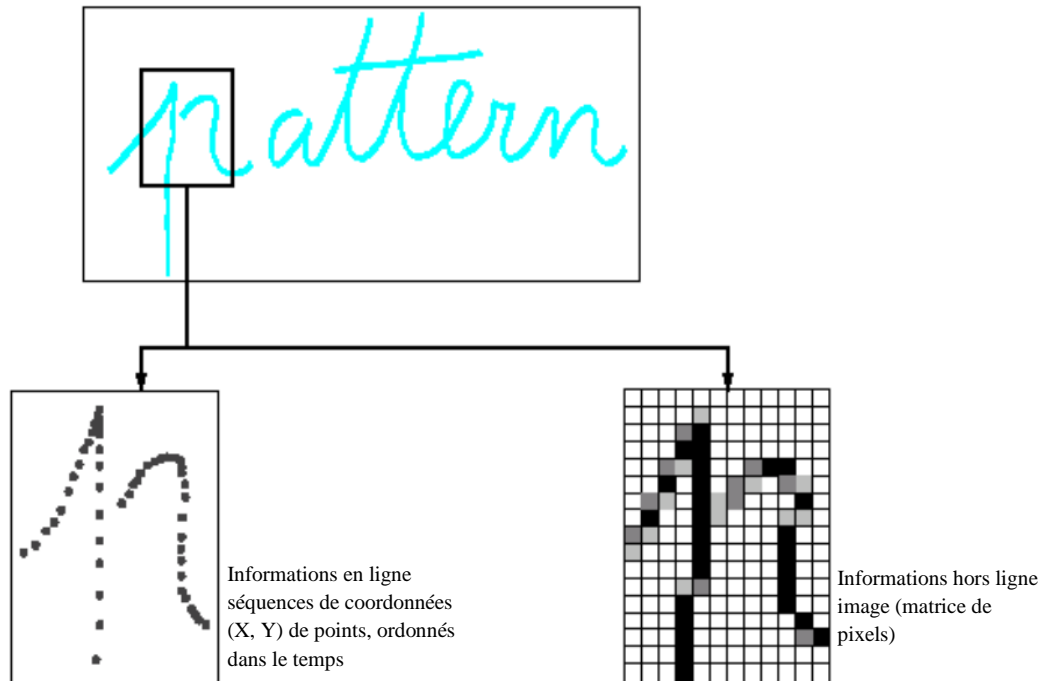
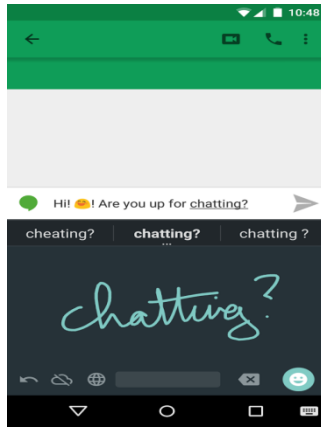


FIGURE 2.1 – Exemple d'écriture en ligne et hors ligne [LOR 98].

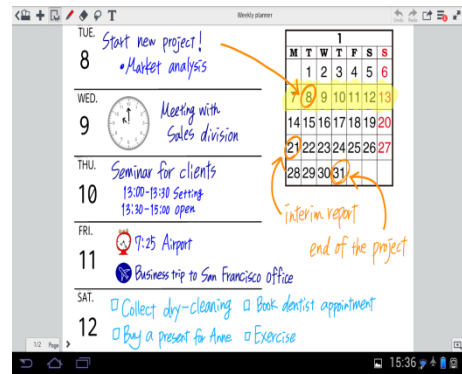
### 2.3.1.2 La reconnaissance en-ligne (on-line)

Dans le cas en-ligne, ou bien dynamique [BEL 01], le système reçoit les images d'entrée de données en temps réel. Cela permet d'une part d'intégrer les mouvements du stylo électronique [PAQ 01] et pression de l'information [SMR 98], et d'autre part de calculer la relation entre les points pour extraire les caractéristiques afin de reconnaître les symboles au fur et à mesure qu'ils sont écrits à la main. D'une manière générale, il exploite des informations temporelles dans la construction des caractères. La reconnaissance en-ligne présente un avantage majeur qui est la possibilité de correction et de modification de l'écriture de manière interactive vu la réponse en continu du système [SHU 08].

Parmi les plates-formes disposant d'un système de reconnaissance de l'écriture, nous trouvons le Palm (Figure 2.2 (a)) et l'agenda électronique (Figure 2.2 (b)). Ces deux appareils regroupent une tablette à numériser et un programme procédant à la reconnaissance de l'écriture. Elles assurent des fonctions de communication, de prise de notes, de gestion d'agenda et de rendez-vous. De ce fait, leurs utilisations sont plus attrayantes puisqu'elle évite le besoin de scanner, au préalable, l'écriture [AYA 04]. Aujourd'hui, d'autres applications se sont développées, à partir de ces assistants personnels, pour le domaine de la médecine, du marketing, du management, de l'éducation, de la gestion et de l'industrie.



(a) Communication en ligne



(b) Agenda électronique

FIGURE 2.2 – Exemple des plates-formes disposant d'un système en-ligne de reconnaissance de l'écriture manuscrite [SIT 16]

La plupart des systèmes en-ligne sont analytiques et se basent sur des techniques tels que : l'appariement élastique (elastic matching) [BAH 01], les k-plus proches voisins (KNN) [SCH 96], les Support Vecteur Machines (SVM) et les réseaux de neurones (NNs) pour la classification des lettres isolées [BEN 94; BEN 00; MEZ 03; MEZ 02]. Pour la reconnaissance de mots ou phrases cursives, des approches hybrides combinant les modèles de markov cachés (HMMs) avec des NNs sont utilisées [FAR 09]. Un bon état de l'art des principaux systèmes de reconnaissance de l'écriture en ligne suivie d'une analyse des différentes techniques est présenté dans [PLA 00].

La reconnaissance de l'écriture hors-ligne est plus complexe que celle qui est en-ligne due à la présence du bruit dans le procédé d'acquisition des images et la perte d'information temporelle telle que l'ordre d'écriture et la vitesse [MOR 03]. La difficulté d'un système de reconnaissance d'écriture hors-ligne s'accroît quand le système traite des scripts d'écriture manuscrite cursive telle qu'un script d'écriture manuscrite arabe [SHU 08]. Le tableau suivant présente une brève comparaison de ces deux approches

Tableau 2.1 – Une brève comparaison entre la reconnaissance en-ligne et hors-ligne [ALR 06]

Critère de comparaison	En-ligne	Hors-ligne
Outils d'acquisition	-Stylo électronique plus tablette graphique.	- Scanner ou caméra.
Bruit d'image	- faible	- existence d'un bruit important.
Informations disponibles	- la position, - la direction du mouvement, - les points de fin, - les points de début, - ordre des traits.	-absence d'informations contextuelles.

## 2.3.2 Reconnaissance selon le type d'écriture

### 2.3.2.1 Écriture imprimée

Dans ce type d'écriture, les caractères sont souvent bien alignés et bien séparés verticalement ce qui permet de faciliter la phase de lecture, malgré que certaines fontes présentent parfois des accollements qu'il faut défaire. De plus, le graphisme des caractères est semblable à une fonte existante qui constitue un modèle pour l'identification. L'écriture se caractérise généralement par le nombre de fontes (*mono*, *multi* ou *omni-fonte*). Le système de reconnaissance, dans ce cas, est appelé OCR.

Un système OCR est dit *mono-fonte* s'il ne traite qu'une seule fonte à la fois, c'est-à-dire qu'il ne connaît que le graphisme d'une fonte unique. Ce cas est simple notamment au niveau de l'apprentissage, car l'alphabet représenté est réduit. Un système OCR est dit *multi-fonte* s'il est capable de reconnaître un mélange de quelques fontes parmi un ensemble de fontes préalablement apprises. Dans ce cas, le traitement doit réduire les écarts existant entre un même caractère (taille, épaisseur et inclinaison). En cas d'éventuelles ressemblances de caractères des différentes fontes, l'apprentissage doit gérer les ambiguïtés entre eux. La reconnaissance dans ce cas doit bien identifier les différences entre ces caractères. Un système OCR *omni-fonte* est capable de reconnaître toute fonte, généralement sans apprentissage, en se basant sur les règles topologiques et morphologiques de l'écriture [PLA 00].

### 2.3.2.2 Écriture manuscrite

Dans ce type d'écriture, les caractères sont souvent reliés et leur graphisme est inégalement proportionné. Une des spécificités de cette reconnaissance est la grande variabilité intra-classe associée aux données. Cette dernière est à imputer principalement aux scribes puisque chacun a un style propre [GRA 04].

Dans le cadre de notre travail, nous nous intéressons à la reconnaissance de l'écriture manuscrite, qui est souvent considérée plus difficile et plus complexe que celle de l'écriture imprimée. La suite de ce chapitre sera donc consacrée à ce type d'écriture.

## 2.4 Les formes du texte manuscrit

Il existe plusieurs formes du texte manuscrit : texte contraint et *non contraint*, *texte continu*, *texte cursif* et *non cursif* et *caractères isolés* (Figure 2. 3).

La reconnaissance de texte *continu* nécessite une segmentation de l'image d'entrée avant la phase de la classification, alors que la reconnaissance des caractères *isolés* procède par la classification des caractères isolés sans avoir besoin d'une étape de segmentation. Dans un système de reconnaissance de texte *imprimé*, l'entrée est toujours sous forme de caractères isolées. Un cas particulier du texte *continu* est la reconnaissance de l'écriture *cursive*. La segmentation dans ce cas est beaucoup plus difficile et des méthodes comme le modèle de Markov caché (HMM) sont souvent employées. L'écriture *non cursive* est également connue en tant que caractères isolés [LAA 97].

Un texte imprimé ou manuscrit peut être également *contraint* ou *non contraint*.

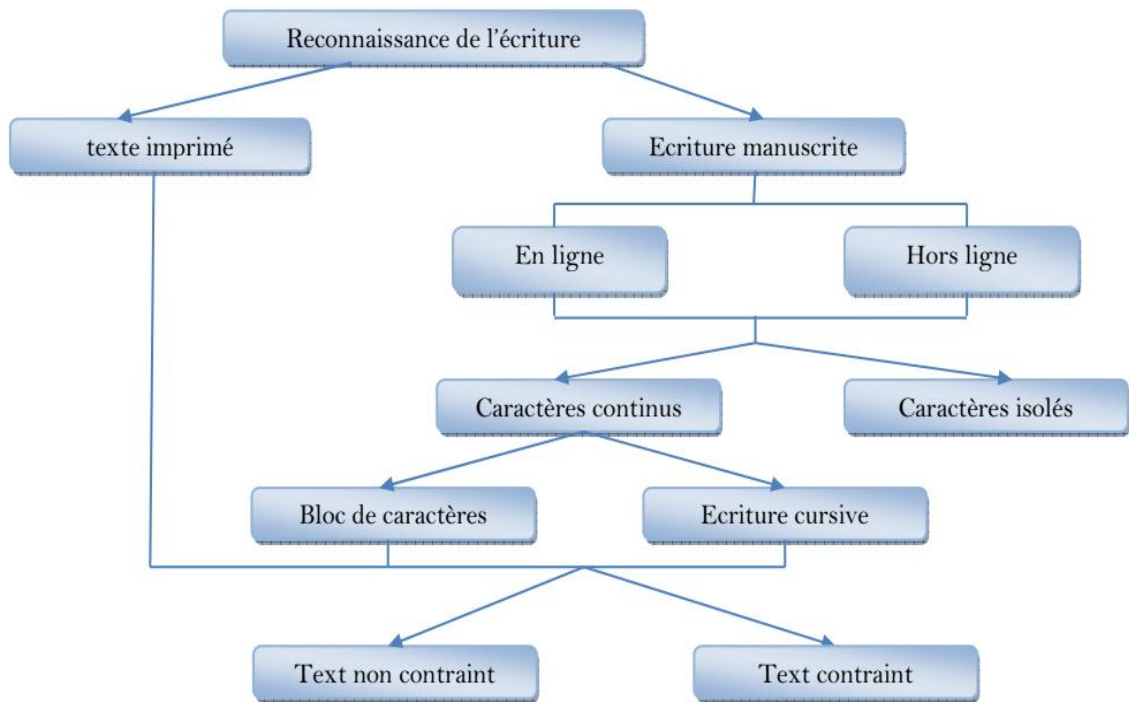


FIGURE 2.3 – Les différentes formes du texte manuscrit [LAA 97]

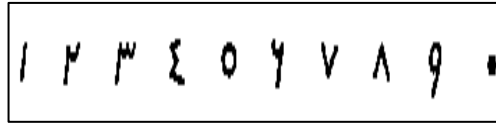
Le texte *contraint* inclut des positions où l'utilisateur doit écrire dans des emplacements prédéfinis. Ainsi, le système a des informations à priori sur le texte. Dans le cas du texte *non contraint*, l'utilisateur a un espace vide, comme une page blanche, qu'il peut la compléter par n'importe quelle information, avec n'importe quel style et dans n'importe quelle direction. Dans la pratique, un texte d'entrée peut être considéré *non contraint* s'il y a plus d'une ligne d'entrée et si le nombre de mots dans chaque ligne n'est pas connu à l'avance [LAA 97].

De plus, chaque langue possède ses propres caractéristiques telles que le nombre de lettres et l'écriture de ces lettres dans un mot. Prenant l'exemple de la langue arabe cursive. L'arabe est écrit de droite à gauche et elle est toujours *cursive*. Elle a 29 lettres de base et huit diacritiques. Chaque lettre a plusieurs formes selon sa position dans le mot. Par exemple, la lettre *Kha* a quatre formes :

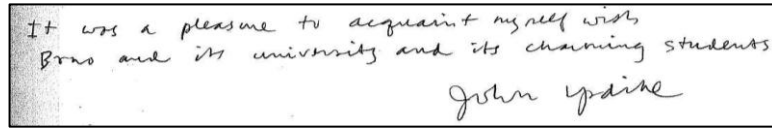
*isolée* (خ) *initiale*, *médiale*, et *finale*, de droite à gauche respectivement (خخخ)  
[ABA 08]

Des exemples de formes du texte manuscrit sont montrés dans la figure suivante :

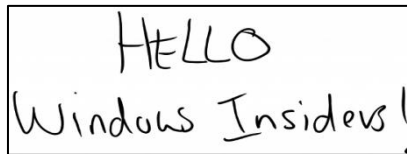




(a) texte contraint caractères isolés



(b) texte cursif



(c) texte continu



(d) texte contraint



(e) texte non contraint

FIGURE 2.4 – Les différentes formes d'écriture manuscrite

## 2.5 Les types de reconnaissance d'écriture manuscrite

Nous décrivons ici une terminologie importante sur les types d'entrée et de sortie du système de reconnaissance. Les unités de l'écriture significative peuvent être classées en quatre types : *caractère*, *mot*, *phrase* et *partie d'un texte* [PAR 99]. Ces unités sont définies comme ceux qui sont écrits par la main humaine. Les définitions du dictionnaire de ces quatre unités sont [PAR 99] :

- *Caractère* : un symbole graphique comme une lettre d'hiéroglyphe ou d'alphabet ou bien un chiffre utilisé dans l'écriture ou l'impression.
- *Mot* : un caractère écrit ou imprimé ou combinaison de caractères représentant un mot parlé.
- *Phrase* : un mot ou groupe de mots formant un constituant syntaxique avec une fonction grammaticale unique.
- *Partie d'un texte* : un mot, une clause, une phrase ou un groupe de clauses ou de phrases formant une unité syntaxique.

### 2.5.1 La Reconnaissance de caractères

La reconnaissance de caractères est le problème de classification des caractères pré isolé dans un ensemble d'alphabets. La plupart des chercheurs ont adopté l'approche classique de la Rdf dans laquelle l'étape du prétraitement est suivie par l'extraction de caractéristiques et la classification [PAR 99]. Il y a eu beaucoup de sous-thèmes, mais ceux-ci peuvent être simplement divisés dans les catégories suivantes : l'extraction et la sélection de caractéristiques, la méthode de classification, et la combinaison de classificateurs.

### 2.5.2 La Reconnaissance des mots

Les algorithmes de la reconnaissance hors-ligne des mots manuscrits prennent généralement deux entrées : l'image du mot isolé et un lexique représentant les hypothèses possibles pour cette image [PAR 99]. La taille de ce lexique est limitée par l'application utilisée, soit par l'extraction des interprétations permises ou incompatibles. Prenons l'exemple de chèques bancaires, pour identifier la somme en lettres, les seules transcriptions permises sont des nombres écrits en lettres. En outre, si un nombre écrit en mot manuscrit ne présente pas des ascendants ni des descendants, alors en extrayant à partir du lexique toutes les interprétations incompatibles, c'est-à-dire les transcriptions permises composées de lettres sans ascendants ou descendants [VIN 02]. Le but de cette reconnaissance est d'attribuer un score de correspondance à chaque entrée du lexique ou de sélectionner la meilleure entrée parmi l'ensemble de lexiques. Différentes approches pour la reconnaissance de mots manuscrits ont été rapportées, mais ils peuvent être classés en deux catégories principales : les approches analytiques et les approches holistiques [PAR 99]. Une approche holistique reconnaît un mot entier comme une unité, alors que l'approche analytique considère un mot comme une séquence de petits unités ou composants [MON 10].

#### A. Les approches analytiques

Dans une approche analytique, la reconnaissance d'un mot est effectuée en le considérant comme une séquence d'unités (caractères), qui font partie d'un alphabet prédéfini. L'étape de la segmentation des mots en segments ou en caractères considère une partie du processus de reconnaissance. C'est une tâche difficile due aux différents problèmes tels que la fragmentation ou le chevauchement de caractères. Cette opération est en fait la plus difficile en raison de l'ambiguïté produite des mots manuscrits. Dans la littérature, la plupart des méthodes analytiques réussies utilisent des stratégies appelées segmentation-basée-reconnaissance [MOR 03].

#### B. Les approches holistiques

L'idée de base dans une approche holistique est de reconnaître l'image de mot en considérant le mot tout entier. Dans ce cas, il n'est pas nécessaire de fractionner le mot en segments. À l'inverse des méthodes analytiques, les méthodes holistiques sont contraintes aux applications ayant une petite taille de lexique comme dans les systèmes de traitement des chèques bancaires [MOR 03]. Alors que les méthodes analytiques exigent la segmentation d'un mot avant sa reconnaissance, elles sont avantageuses par rapport aux approches holistiques dans le sens où il est plus fiable de reconnaître un ensemble de caractères que des mots entiers. En effet, la fréquence d'apparition d'un mot est bien inférieure à celle de ses lettres, qui sont partagées par tous les mots du lexique. En plus, les méthodes holistiques ne satisfont pas le critère de portabilité, pour chaque nouvelle application, on doit former l'ensemble des mots du lexique [MOR 03].

La plupart des travaux de la reconnaissance des mots manuscrits présupposent que le mot est segmenté par un algorithme approprié au domaine d'application avant d'être traité par le système de reconnaissance [KIM 00 ; MOR 03 ; SCA 00 ; AZI 10].

Il existe un autre type de reconnaissance contenant un ensemble des composants et il nécessite aussi une étape de segmentation avant leur processus de décision. Ce type s'appelle la reconnaissance des chaînes de chiffres.

Dans la reconnaissance des chaînes de chiffres, un procédé de segmentation des chiffres chevauchés est nécessaire. Le problème de reconnaissance des chaînes numériques est également différent du problème de la reconnaissance des mots manuscrits, en effet, presque aucune information contextuelle n'est disponible préalablement, c'est-à-dire n'importe quel chiffre peut suivre n'importe quel autre [NEB 13a]. La segmentation des chaînes numériques est généralement une tâche difficile parce que les différents numéros dans une chaîne peuvent être chevauchés, disjoints (un chiffre peut être constitué de plusieurs segments) ou connectés [NEB 13a]. Dans la plupart des cas, la connexion et le chevauchement des chiffres sont les situations les plus fréquentes qui sont observées dans l'écriture manuscrite.

Le processus de segmentation peut être divisé en deux classes : segmentation-puis-reconnaissance [OUC 07] et segmentation-basée-reconnaissance [PAL 03]. Dans la première classe, le module de la segmentation fournit une séquence d'hypothèses où chaque sous-séquence devrait contenir un caractère isolé soumis au système pour la reconnaissance. Dans la seconde classe, le processus fournit un ensemble d'hypothèses de segmentation et définit le chiffre segmenté en appliquant une reconnaissance à chaque hypothèse de segmentation fournie. Cette approche donne une meilleure fiabilité que la précédente, car elle fournit plusieurs hypothèses qui augmentent le choix du classificateur pour trouver la reconnaissance correcte. Leur inconvénient majeur se situe dans l'effort requis pour comparer toutes les hypothèses produites.

### 2.5.3 La reconnaissance de phrases et textes

Quelques travaux sur la reconnaissance de caractères et mots ont été étendus aux reconnaissances de phrases et de texte manuscrit pour des applications pratiques comme l'interprétation des adresses postales (noms des rues), le traitement automatique des chèques bancaires (montant littéral et dates sur les chèques) et des formulaires (la lecture de texte), ... etc. Les précédents efforts de recherche sont liés aux problèmes de la reconnaissance de phrases manuscrites assument que les mots sont souvent isolés ou peuvent être parfaitement isolés. Généralement le schéma des systèmes de reconnaissance de phrases suit les étapes de traitement suivantes : la segmentation en mots, la reconnaissance des mots et un post-traitement. Les images de mots segmentées sont envoyées à un système de reconnaissance de mots et des contraintes linguistiques et des informations contextuelles sont ensuite utilisées dans l'étape du post-traitement pour compléter le procédé de la sélection de phrases, en supposant que les images de mot sont parfaitement isolées [PAR 99]. En d'autres approches, pour compenser les erreurs dues à la segmentation en mots, un ou plusieurs segments groupés se sont envoyés au système et le meilleur chemin est choisi comme la phrase reconnue.

Les méthodes de la reconnaissance de phrases peuvent être divisées en deux catégories principales : reconnaissance liée à la segmentation et reconnaissance indépendante de la segmentation [MOR 03].

La première catégorie consiste à découper la phrase à un ensemble de composants en

fonction de l'espacement entre les composants adjacents [FAV 98]. La difficulté de cette méthode réside dans le fait que la phrase n'a pas d'espacement uniforme entre ses composants. En plus, ces espaces ne peuvent pas être estimés facilement par une métrique à une seule dimension [MOR 03]. La deuxième catégorie consiste à traiter la phrase manuscrite tout entière comme une unité séparée [MAR 01 ; MOR 03].

Comme notre travail se concentre sur la reconnaissance hors-ligne de caractères manuscrits, nous devons examiner la manière dont ces systèmes de reconnaissance fonctionnent pour atteindre le but fixé.

### **2.6 Le fonctionnement d'un système de reconnaissance de caractères manuscrits**

Dans les systèmes hors-lignes, le schéma général de reconnaissance de caractères passe par les étapes classiques de traitement, à savoir [VAM 09 ; VAM 10] : *l'acquisition, le prétraitement, la segmentation, l'extraction des caractéristiques, l'apprentissage et reconnaissance (classification), et le post-traitement.*

L'étape d'acquisition permet de numériser l'entrée par un scanner. L'étape de prétraitement a pour but de réduire le bruit et les variabilités de l'écriture pour n'en garder que les informations utiles. L'étape de segmentation permet de décomposer l'image d'un texte en entités (mots, caractères ou graphèmes) afin de réduire la complexité des processus de traitements ultérieurs. L'étape d'extraction des caractéristiques permet de représenter l'image d'entrée comme un vecteur de primitives de dimension fixe. Le résultat est une base de données d'apprentissage qui constitue la base de référence de système. L'étape de classification consiste à attribuer une classe de correspondance au caractère à reconnaître en comparant la description de ce dernier (issue de la base de test) avec celles de la base d'apprentissage. L'étape de post-traitement est une opération facultative et consiste à vérifier et éventuellement à corriger l'hypothèse de reconnaissance générée par le module de classification en utilisant des informations contextuelles telles que : lexicales, syntaxiques, sémantiques, pragmatiques, etc.

Dans les sections qui suivent, nous expliquerons en revue les techniques de mise en œuvre de chaque étape du système de reconnaissance.

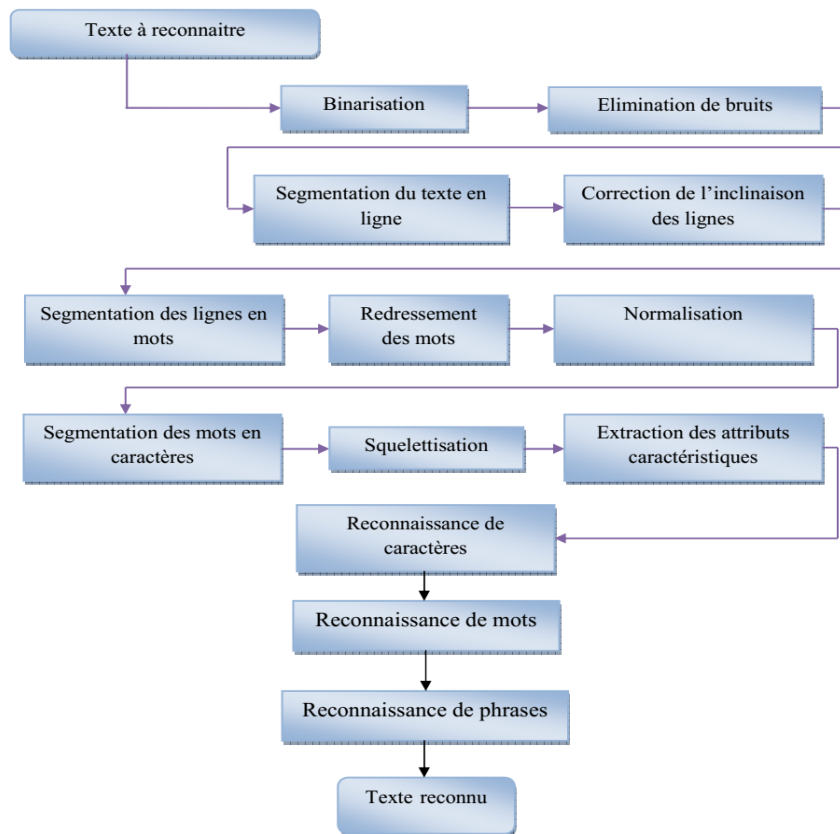


FIGURE 2.5 – Les différentes étapes pour la reconnaissance d'un texte

### 2.6.1 L'étape d'acquisition

L'acquisition consiste à numériser l'écriture à l'aide d'un capteur. Le résultat de cette phase représente l'entrée de système. Il dépend de deux paramètres importants : résolution et niveau d'éclairage. Bien que l'acquisition soit assez simple, elle est très importante, car elle influence sérieusement sur les étapes suivantes.

### 2.6.2 L'étape de prétraitement

L'étape de prétraitement est nécessaire pour réduire la dimension de représentation de caractère (matrice de grande taille), et plus particulièrement, pour rendre plus facile la conception du système de reconnaissance. L'objectif principal de cette étape est de faciliter la caractérisation de la forme ou de l'entité à reconnaître (caractère, chiffre, mot), soit en nettoyant la forme par élimination du bruit (introduit par les outils d'acquisition, la qualité du papier, l'encre ou encore l'effet du temps dans les documents anciens), ou en réduisant la quantité d'information à traiter pour ne garder que les informations les plus significatives.

L'image d'entrée peut subir à des manquements (des absences de points ou trous) ou des empâtements (une surcharge de points) à cause de divers problèmes d'éclairage et de saisie. Il convient de corriger, si possible, ces problèmes avant toute étape d'analyse. Par ailleurs, il n'est pas toujours utile de travailler sur tous les points de l'image d'entrée pour l'extraction de caractéristiques, il s'agit d'éliminer les points redondants. Parmi les techniques de prétraitement les plus utilisés, on trouve : la binarisation, la réduction de bruit, la localisation et la normalisation, et la squelettisation.

### 2.6.2.1 Binarisation

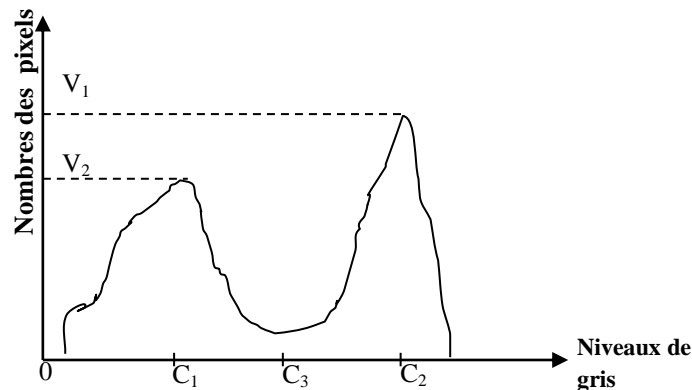
La binarisation est une opération qui permet de distinguer entre le fond de l'image et l'information manuscrite contenue dans l'image. Elle consiste à attribuer à chaque pixel de l'image une valeur de 0 ou de 1 : '0' représente le noir (l'objet/le texte), et le '1' représente le blanc (l'arrière-plan). La plus simple façon pour obtenir une image binaire est de choisir une valeur seuil, puis de classer tous les pixels dont les valeurs sont au-dessus de ce seuil comme étant des pixels d'arrière-plan, et tous les autres pixels comme étant des pixels de fond. Cette opération s'appelle le *seuillage* [THE 12].

La valeur du seuil est déterminée selon deux manières : soit par une analyse globale, c'est-à-dire à partir de l'histogramme de niveau de gris de tous les pixels de l'image si le contraste texte/arrière-plan est suffisamment marqué, alors elle sera égale à la valeur du niveau de gris se trouvant dans la vallée entre les deux pics de l'histogramme (Figure 2.6). Soit par une analyse des contrastes locaux si les documents présentent des variations locales de contraste.

Soient  $I(M \times N)$  l'image à traiter,  $s$  le seuil choisi et  $f(x,y)$  l'application qui représente le niveau de gris du pixel aux coordonnées  $(x,y)$ ,  $0 \leq x \leq M$ ,  $0 \leq y \leq N$ . Les pixels de l'objet sont ceux ayant le niveau de gris inférieur à  $s$ , et les pixels de l'arrière-plan sont ceux ayant le niveau de gris supérieur. Alors, l'image binarisée  $g$  est déterminée par les pixels  $(x,y)$  dont la valeur est donnée par la formule :

$$g(x,y) = \begin{cases} 1, & \text{si } f(x,y) > s \\ 0, & \text{si } f(x,y) \leq s \end{cases} \quad (2.1)$$

Il s'agit de calculer le seuil d'une manière adéquate, car les composantes ou les ségments d'objet liés des traits fins peuvent se déconnecter, ce qui modifie la forme originale de l'objet. En effet, le mauvais choix du seuil peut détruire une grande part d'information contenue dans l'image de l'enveloppe. Alors, une bonne binarisation doit être capable de conserver à la fois tous les caractères et les gravures sans ravoir trop de bruit.



$C_1$  et  $C_2$  : sont les niveaux de gris des pics,  $C_3$  : le niveau de gris de seuil

FIGURE 2.6 – Histogramme des niveaux de gris

La plupart des méthodes de binarisation des images de texte se basent sur des méthodes de seuillages globales ou locales qui seront abordées dans la section suivante [AMR 12].

#### A. Seuillage global

Cette méthode globale consiste à calculer un seul seuil à partir d'une mesure globale sur toute l'image. Elle permet de déterminer l'appartenance d'un pixel à l'objet ou au fond sur toute l'image. Elle n'est plus applicable lorsque le fond est bruité ou non homogène, ou bien si la qualité d'impression du texte n'est pas constante dans toute la page. Dans ces cas, des taches parasites peuvent apparaître. Elle est surtout convenable pour les documents simples et de bonne qualité. Les méthodes d'Otsu [OTS 75], de Kapur [KAP 85], de Pun [PUN 80], ou de Cheng et Chen [CHE 98a] peuvent être tenues comme des représentants de cette méthode, mais chacun a de différentes stratégies pour atteindre leur objectif. Cependant, la solution fournie par ces techniques n'est pas toujours efficace, et la figure 2.7 ci-dessous montre le résultat imparfait retourné par un seuil global, dans le cas d'une mauvaise illumination du document. Pour surmonter ces problèmes, il a fallu trouver des techniques permettant d'adapter localement le niveau du seuil.

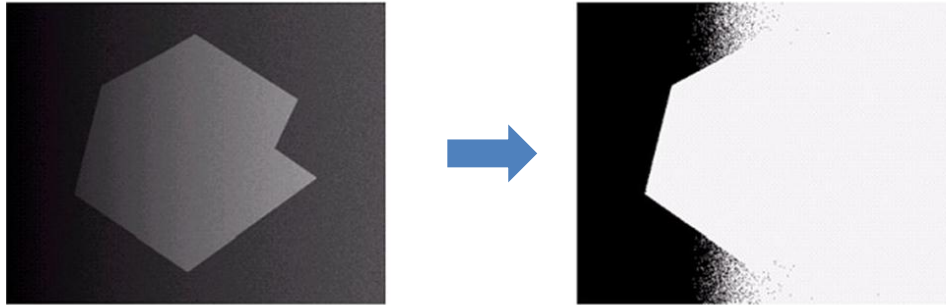


FIGURE 2.7 – Problème de seuillage global

### B. Seuillage local

Le seuillage local a pour principe d'adopter une étude localisée autour du pixel pour déterminer quel seuil utiliser. Il existe des techniques qui utilisent une fenêtre centrée sur le pixel à traiter (fenêtre de voisinage de pixel) pour réaliser cette étude locale. Cette fenêtre peut être de différentes tailles, généralement en fonction de la taille moyenne du texte dans le document. En 1986 et pour la première fois, Bernsen a proposé une technique donnant de bons résultats [BER 86]. Dans sa méthode, Le seuil est donné par la formule suivante :

$$S(i, j) = \frac{(\min(i, j) + \max(i, j))}{2} \quad (2.2)$$

Avec  $S(i, j)$  seuil à appliquer pour le point  $(i, j)$ ,  $\max(i, j)$ ,  $\min(i, j)$  valeur du niveau de gris maximal/minimal dans une fenêtre de taille  $N \times M$  centrée en  $(i, j)$ .

Niblack proposa, dans la même année, une méthode similaire à ce principe, mais prenant en compte d'autres paramètres [NIB 85]. Le seuil est calculé par la relation ci-dessous

$$S(i, j) = \mu(i, j) + k \times \sigma(i, j) \quad (2.3)$$

Avec  $S(i, j)$  seuil à appliquer pour le point  $(i, j)$ .

$\sigma(i, j)$  valeur de l'écart type dans une fenêtre de taille  $N \times M$  centrée en  $(i, j)$ .

$\mu(i, j)$  : valeur moyenne des niveaux de gris dans la même fenêtre.

$k$  : constante fixée le plus généralement à 0,2.

Une amélioration de la méthode de Niblack a été proposée par Sauvola en 2000, afin de réduire sa sensibilité au bruit sur le fond [SAU 00]. Le seuil est défini ainsi : (2.4) :

$$S(i, j) = \mu(i, j) + k \times \left( \frac{\sigma(i, j)}{R} - 2 \right) \quad (2.4)$$

## 2. La reconnaissance de l'écriture manuscrite

Avec  $R$  étant une constante permettant d'ajuster la dynamique de l'écart type (fixé en général à 128). Les Figures 2.8 et 2.9 ci-dessous présentent les résultats obtenus, par quelques méthodes de seuillage, sur des images de manuscrits arabes.

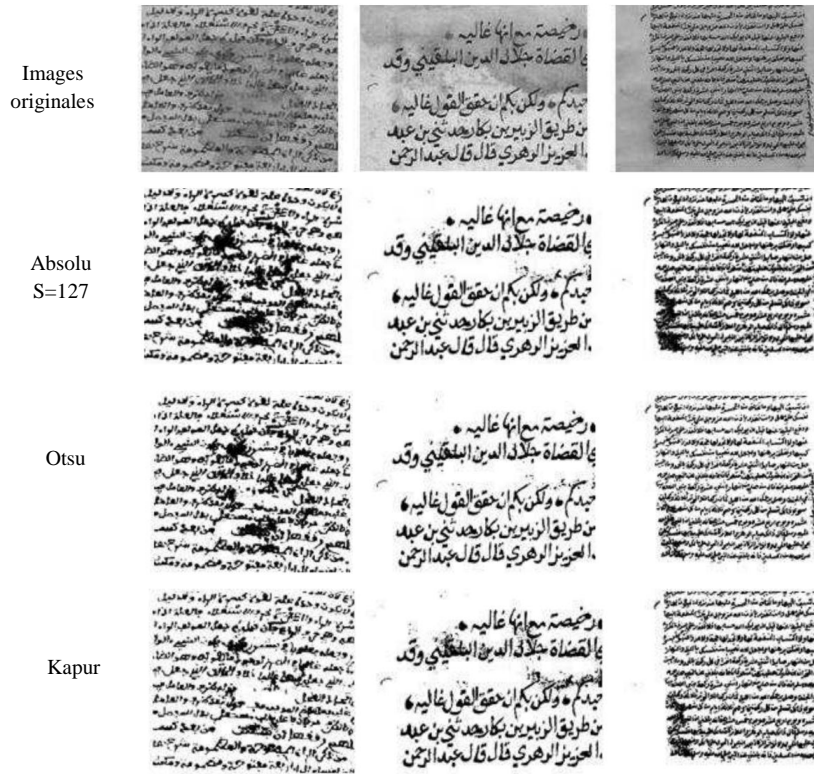


FIGURE 2.8 – Résultats de seuillage globales sur les manuscrits arabes [KEF 10]





FIGURE 2.9 – Résultats de seuillage locales sur les manuscrits arabes [KEF 10]

### 2.6.2.2 Réduction de bruit

Cette étape a pour but de résoudre les problèmes de manquement et d'empâtement. Elle permet donc d'améliorer la qualité de l'image. Pour cela, il s'agit d'effectuer une opération de réévaluation de la valeur d'un pixel en fonction des pixels voisins.

La réduction de bruit est généralement basée sur deux approches principales qui sont : *le filtrage* (les masques) et *les opérations morphologiques* (érosion, dilatation, ... etc.)

#### A. Filtrage

Le filtrage permet d'améliorer la qualité de l'image en améliorant la perception de certains détails, réduisant le bruit, compensant certains défauts du capteur, etc.

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

- Par élimination des points de la forme isolés ou situés à l'extrémité des contours, en appliquant sur l'image, dans les huit directions, le masque suivant :

0	0	0
X	1	0
0	0	0

- Par élimination des points formant des angles droits ou des croisements du contour, en appliquant le masque suivant dans les huit directions :

X	0	0
1	1	0
0	0	0

- Pour le bouchage, il s'agit d'égaliser les contours et de boucher les trous internes à la forme des caractères en lui ajoutant des points noirs. Par exemple, nous pouvons utiliser le bouchage de trous isolés. Dans ce cas, si le voisinage d'un point du fond correspond au masque suivant, alors ce point est mis à 1.

1	1	1
1	0	1
1	1	1

### B. Morphologie mathématique

On peut définir un opérateur de morphologie mathématique par l'intermédiaire d'un élément structurant, qui est une forme. On peut le voir aussi comme un masque. Il y a deux opérations morphologiques de base : l'érosion et la dilatation [BUR 01]. À partir de ces deux opérations, on peut définir d'autres opérations plus complexes, telles que l'ouverture et la fermeture [TEA 80].

L'utilisation de ces opérations morphologique peut être pour l'extraction du contour (interne et externe).

### 2.6.2.3 Localisation et normalisation

#### A. Localisation

Pour pouvoir reconnaître une écriture manuscrite, il faut d'abord la localiser. Lorsque l'image d'entrée n'est pas trop complexe, on peut réaliser une localisation en comptant les pixels blancs dans le sens des lignes, puis, pour chaque ligne de caractères, en comptant les pixels blancs dans le sens des colonnes. La comparaison du résultat du comptage avec un seuil permet de détecter les débuts et fins de caractères. Cette étape se divise souvent en trois niveaux différents de segmentation, correspondant à la localisation des zones d'intérêt sur une image, la localisation des lignes et enfin la localisation des mots des caractères ou même des graphèmes [OLL 99 ; FAR 09].

Dans le premier niveau, il s'agit de déterminer les zones d'intérêt existant sur un document en utilisant des connaissances a priori sur la nature du document traité (chèques, adresses, formulaires...), et en s'appuyant sur les règles classiques de disposition spatiale de l'écriture. Ce processus peut être relativement simple si la structure du document est fixe et connue préalablement. Prenons l'exemple des chèques bancaires, la localisation se base sur les positions et les dimensions estimées des différents champs. La localisation de ces champs peut s'avérer beaucoup plus difficile lorsque la structure de document est plus complexe.

Le deuxième niveau consiste à localiser les lignes d'écriture au sein des zones ou blocs de texte extraits préalablement. De ce fait, on s'appuie le plus souvent sur un modèle physique de disposition de l'écriture. Lorsque ce modèle est respecté, c'est-à-dire que les lignes d'écriture sont plus ou moins parallèles et horizontales, le processus d'extraction est relativement simple et des méthodes de détection, basées généralement sur l'analyse d'histogrammes de projection déterminés sur l'ensemble du bloc de texte, donnent des résultats très satisfaisants (Fig 2.10). Néanmoins, lorsque le document manuscrit présente des dispositions variables, des lignes de bases fluctuantes ou des inclinaisons différentes, des techniques spécifiques doivent être mises en œuvre [HEU 03].

Le troisième niveau a pour but de localiser les mots qui se trouvent à l'intérieur des lignes extraites préalablement. Cette opération se base généralement sur la détection d'espaces à l'intérieur d'une ligne par une recherche de composantes connexes, ou par des his-

togrammes de projection. Comme il existe deux types d'espaces : les espaces intra-mot et les espaces inter-mots, il s'agit souvent d'utiliser une hypothèse de départ qui consiste à considérer que les espaces entre les mots sont plus grands que les espaces entre les lettres. La comparaison à un seuil de décision intra-inter mots est une technique simple permettant de différencier entre ces deux types d'espaces.

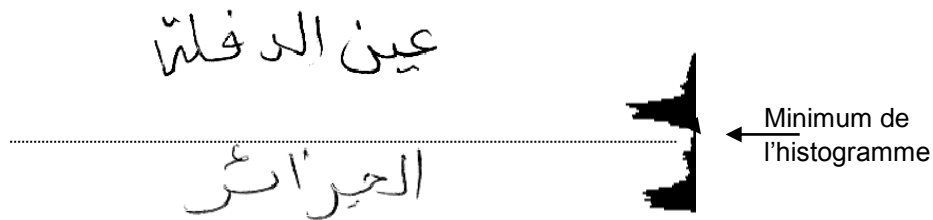


FIGURE 2.10 – Histogramme de la segmentation lignes [FAR 09]

### B. Normalisation

La normalisation de la taille des images est une technique naturelle de prétraitement qui permette de ramener les caractères à la même taille. En effet, à cause de la variation des tailles de caractères (d'une écriture à l'autre) ou des opérations d'agrandissement ou de réductions sur ces tailles, les caractères peuvent subir une légère déformation dans la taille, ce qui complique les tâches de segmentation et peut causer une instabilité des paramètres.

La normalisation tend aussi à réduire ou à éliminer autant que possible les variabilités liées aux styles et orientations d'écriture pour rendre celle-ci la plus indépendante possible du scripteur. Donc, les traitements de normalisation peuvent être :

- Normalisation de la taille.
- Correction de l'inclinaison des lignes : (Skew correction).
- Correction de l'inclinaison des caractères : (Slant correction).
- Estimation de la ligne de base.

Les travaux dans [SAO 97 ; FUJ 07] présentent une normalisation de différentes variations rencontrées avec différentes techniques.

L'algorithme de normalisation que nous avons utilisé est très simple et il sera présenté dans le chapitre 5 de cette thèse.

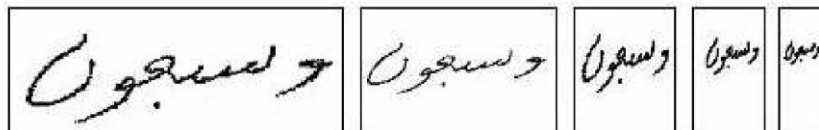


FIGURE 2.11 – Exemples de mots manuscrits avec des tailles différentes [AMR 12].

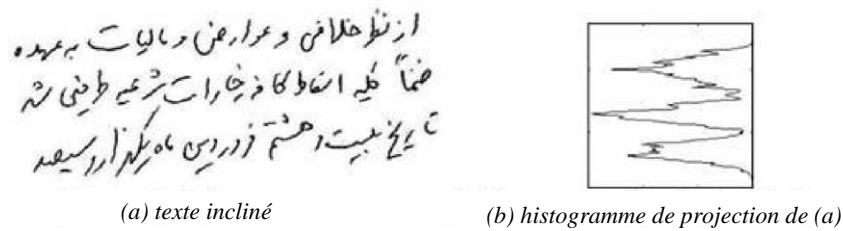


FIGURE 2.12 – La technique des histogrammes de projection pour la détection d'inclinaison [HAJ 05]

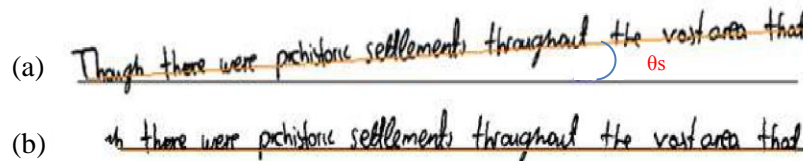


FIGURE 2.13 – (a) image original avec l'angle d'inclinaison, (b) la correction d'inclinaison par les histogrammes projection [LOU 09].

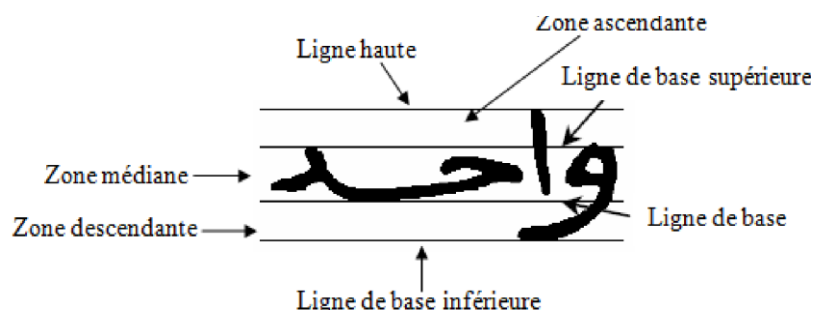


FIGURE 2.14 – Topologie d'un mot pour l'estimation de ligne de base [BEN 07]

### 2.6.2.4 Squelettisation

La squelettisation permet de ramener l'image prétraitée de caractère à une forme (écriture) linéaire d'une épaisseur égale à un pixel, en préservant la forme, la connexité, la topologie du tracé (nombre de parties, nombre de trous) et ses propriétés métriques (longueur totale, distance entre parties de la forme). Elle permet d'extraire des caractéristiques importantes comme les intersections, le nombre de tracés et leurs positions relatives. D'une façon générale, elle permet de réduire considérablement le nombre de données et d'optimiser les étapes suivantes de traitement.

### 2.6.3 La segmentation

La segmentation est une étape qui consiste à diviser le document en lignes puis chaque ligne en mots pour finir au niveau le plus bas : caractère ou graphème. Elle permet de réduire la complexité des modules de traitements qui suit [CAS 96 ; LU 96]. En effet, la séparation des lignes, des mots, des pseudo-mots, des caractères et des graphèmes constitue des opérations difficiles et coûteuses, car les lignes sont parfois enchevêtrées les unes

dans les autres et les caractères sont généralement liés dus à la variation d'écriture. Le problème le plus ardu, selon la littérature, réside dans la segmentation de l'écriture cursive, où la communauté de la reconnaissance de manuscrits admit le paradoxe de Sayre suivant [SAY 73] : « *une lettre ne peut être segmentée avant d'avoir été reconnue et ne peut être reconnue avant d'avoir été segmentée* ».

Les trois premiers types de segmentation étaient présentés dans la partie Localisation. Cette partie consiste à localiser les caractères ou les graphèmes à l'intérieur des mots extraits dans les niveaux précédents. Le problème majeur de cette partie est de chercher à déterminer avec exactitude la frontière de séparation qui existe entre les caractères. Malheureusement, la segmentation du mot en caractères ou graphèmes est une étape cruciale lorsque la taille du vocabulaire est importante.

Les techniques de segmentation existant dans la littérature se divisent en deux types à savoir : la segmentation implicite et la segmentation explicite. Nous allons les détailler comme suit :

### 2.6.3.1 Segmentation implicite

La segmentation est dite implicite lorsque celle-ci est basée sur un moteur de reconnaissance pour valider et classer les hypothèses de segmentation (Figure 2.5). Elle est donc soutenue conjointement par le processus de reconnaissance des caractères ou des graphèmes, en évitant la pré segmentation d'un mot en lettres ou entités plus fines [MIL 97; PEC 03]. Dans ce type, les processus de segmentation et de reconnaissance sont réalisés conjointement, d'où le nom parfois employé de "*segmentation-basée-reconnaissance*".

Ce type de segmentation se base principalement sur la division du signal d'entrée (l'écriture manuscrite) à un ensemble d'intervalles dont le pas et la dimension sont fixés préalablement(en un ou plusieurs pixels).

Dans la littérature, on trouve deux manières d'utiliser ce type. La première s'appelle méthode par fenêtrage ou fenêtres glissantes. Cette méthode utilise une fenêtre mobile de largeur variable pour trouver des séquences de points de segmentations potentiels qui seront confirmés ou non par la reconnaissance de caractères [DUP 04; PEC 03; PEC 06; MÄR06; BIA 06]. La variation de la taille et l'inclinaison de la fenêtre donnent plusieurs séquences qui seront traitées par la suite dans le module de reconnaissance. En outre, cette méthode est efficace, car l'information est localisée par des modèles des lettres et la validation se fait par ces modèles, elle s'avère très difficile dans le paramétrage de la fenêtre mobile (hauteur, largeur, inclinaison), ce qui nécessite une expérimentation au préalable de toutes les hypothèses existantes.

La deuxième méthode est appelée méthode par recherche de primitives. Cette dernière utilise la détection des combinaisons de caractéristiques qui permettent de donner par la suite une meilleure reconnaissance. Dans ce cas, il n'y a pas de module de segmentation parce qu'en réalité c'est le classificateur qui segmente d'une manière indirecte les mots.

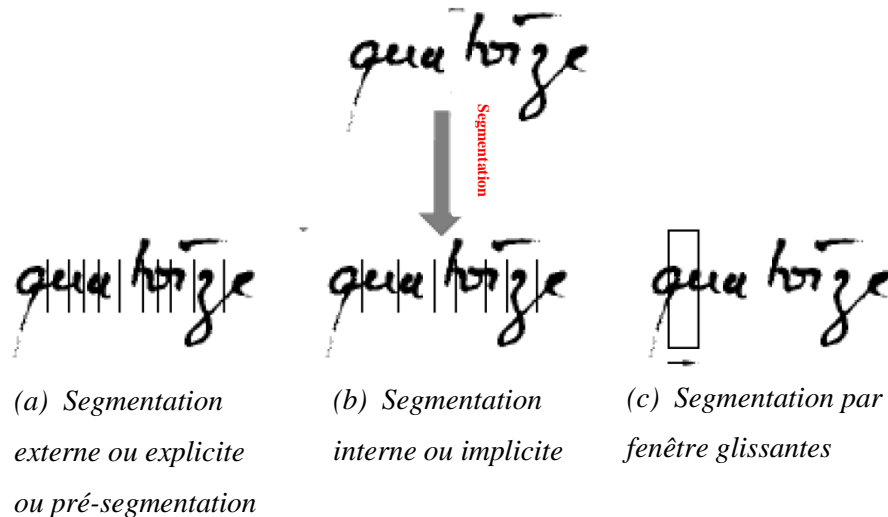


FIGURE 2.15 – Différentes segmentation pour le mot quatorze [AUG 01]

### 2.6.3.2 Segmentation explicite

Le module de segmentation, dans cette méthode, est exécuté avant le module de reconnaissance, et il est nettement visible dans l'architecture des systèmes de reconnaissance (Figure 2.15). La communication avec le classificateur est unidimensionnelle c'est-à-dire dès que la phase de segmentation est terminée, le système sort totalement de ce module.

Cette segmentation a un avantage majeur, c'est que l'information est localisée explicitement, puisqu'on sépare les lettres non pas d'après leur reconnaissance, mais d'après des critères topologiques ou morphologiques. Malheureusement, il n'existe pas de méthode de segmentation fiable à 100 %, toute erreur de segmentation pénalise par la suite les performances de système. Pour la mise en œuvre d'une segmentation explicite, il existe quatre approches [AMR 12] :

- Les approches basées sur des analyses par morphologie mathématique : permettent la sélection des points de segmentation en utilisant le principe de régularité et singularité [MOT 97].
- Les approches basées sur l'analyse des contours : déterminent les candidats de coupure en s'appuyant sur les extremas locaux du contour [BAL 06 ; DUP 04 ; MAD 01].
- Les approches basées sur l'analyse du profil d'histogramme de projection verticale [YAN 98].
- Les approches basées sur l'analyse du squelette : consistent à repérer les points de coupure sur le squelette en se basant sur des seuils ajustés [DUP 04 ; ZER 07].

Pour plus d'informations sur les techniques de segmentation, le lecteur peut se reporter aux références suivantes [TAY 01 ; AKR 98].

## 2.6.4 L'extraction des caractéristiques

L'extraction des caractéristiques, souvent appelées l'extraction de primitives, consiste à représenter les données d'entrée (mots, caractères, graphèmes) en un vecteur de primitives de dimension fixe. Le but de cette phase est d'identifier les informations pertinentes,

discriminante et de dimension limitée pour l'étape de classification, tout en évitant le risque de perte des informations importantes et significatives. En effet, un mauvais choix des primitives influence négativement et nettement sur les résultats même si on utilise un classificateur très performant.

Le résultat de cette phase est une séquence de valeurs numériques ou de symboles qui représente l'image binaire et qui traduit, soit la présence ou l'absence (cas booléen) ou la valeur associée (cas réel) à la caractéristique concernée. Selon [BEN 07] l'extraction de caractéristiques est un « *problème d'extraction à partir de l'image, de l'information la plus pertinente, pour un problème de classification donné, c'est-à-dire celle qui minimise la variabilité intra-classe et qui maximise la variabilité interclasse* ».

Dans la littérature, les caractéristiques sont généralement classées en trois catégories principales [AMR 12] : caractéristiques statistiques, caractéristiques structurelles et topologiques, et transformations globales.

### 2.6.4.1 Caractéristiques statistiques

Les caractéristiques statistiques tentent de représenter l'information distribuée sur l'image entière [DEH 01 ; ELH 05]. Elles sont souvent générées par l'estimation de la densité de pixels appartenant à toute la totalité de l'image (caractères ou mots), ou uniquement dans certaines parties. Pour cette raison, des mesures statistiques (entropie, moyenne, variance, etc.) peuvent être associées à des caractéristiques locales [ALM 02a ; ALM 02b]. Les caractéristiques plus utilisées pour la reconnaissance d'écriture manuscrite sont :

- L'histogramme directionnel qui représente le nombre de pixels sur une ligne dans une direction quelconque de l'image.
- Les moyennes locales de pixels d'une fenêtre dans l'image : principe du "zoning".
- Les projections verticales et horizontales des densités de pixels noir/blanc : représentent le nombre de pixels sur chaque ligne ou chaque colonne de l'image.
- L'histogramme des transitions noir/blanc et vice versa : permet de retenir le nombre des transitions 0-1 et 1-0 entre pixels.
- Les directions des contours dans une fenêtre locale.
- Les profils de projection.

### 2.6.4.2 Caractéristiques structurelles

Les caractéristiques structurelles représentent les propriétés topologiques et géométriques de l'objet. Elles ne sont pas extraites à partir de l'image brute de l'écriture, mais à partir de la représentation de la forme à savoir le squelette et le contour. Il s'agit particulièrement de : convexités, concavités, occlusions, ascendants, descendants, composantes connexes, points extrêmes et points terminaux, segments de droites et leurs attributs (position, orientation...), mesures de pentes, boucles, arcs, croisements, paramètres de courbures, angularités, jonctions des traits, longueur et épaisseur des traits, surfaces et les périmètres [PAL 01 ; KAP 02 ; ALH 07].

Contrairement au type statistique, les caractéristiques structurelles ont une grande capacité discriminative très forte. Ainsi, cette prétention influe sur la rapidité de la décision dans la phase de classification. Une mauvaise gestion de ces caractéristiques, comme la mauvaise détection de leur présence, conduit automatiquement à des résultats insatisfaisants pendant le processus de classification [AMR 12].

### 2.6.4.3 Transformations globales

L'objectif de transformations globales est de changer l'espace de représentation de l'image d'entrée pour faciliter l'extraction des caractéristiques utiles et pertinentes. Dans ce type, il n'y a pas une extraction réelle des caractéristiques puisque le vecteur est construit à partir de tous les pixels de l'image. Plusieurs techniques sont utilisées dans la littérature, on peut citer : la transformation de Hough [GIN 04], la transformée de Fourier [ZER 07], la transformée en ondelettes [RED 15] les moments invariants [WOO 96] : les moments de Hu [KLA 02], Zernike [MEZ 05]... etc.

### 2.6.5 La classification

Classer un ensemble d'objets, veut dire attribuer à chacun une classe ou (catégorie ou étiquette) parmi plusieurs classes définies préalablement. Cette tâche est appelée « classification » ou « discrimination », et l'algorithme qui réalise automatiquement cette tâche est appelé classificateur (classifieur) [ARR 06].

Pour construire un tel classificateur, il existe plusieurs approches qui peuvent être séparées en quatre catégories distinctes : les approches basées sur les prototypes, les approches statistiques, les approches structurelles et les approches bio inspirées.

#### 2.6.5.1 Les approche basée sur les prototypes

Le principe de ces approches est simple, les formes similaires sont assignées à la même classe. Ces formes peuvent être classées en utilisant quelques prototypes par classe dès qu'une bonne mesure de similarité a été choisie. Le choix de la mesure de similarité et des prototypes est crucial pour le succès de la classification. Un exemple de ces approches est le classifieur du plus proche moyen (*Nearest Mean Classifier NMC*) [SKU 02]. Le choix des prototypes dans ce classificateur est simple et robuste : chaque classe de formes est représentée par la moyenne de toutes les formes d'apprentissage de la classe. Un autre exemple est le classifieur du plus proche voisin (*Nearest Neighbour Classifier NNC*) [LIU 01].

#### 2.6.5.2 Les approches statistiques

Cette approche repose essentiellement sur des fondements mathématiques (probabilité et statistique). Des concepts bien connus de la théorie des statistiques sont utilisés pour établir les frontières de décision entre les classes des formes pour les caractériser d'une manière statistique. Autrement dit, cette approche bénéficie des méthodes d'apprentissage automatique s'appuyant sur des bases théoriques connues telles que la théorie de la décision bayésienne, l'analyse en composantes principales et les méthodes de classification non supervisées. Les approches statistiques peuvent être divisées en deux principales familles : les méthodes *paramétriques* et les méthodes *non paramétriques*. Pour plus d'informations, Kunt décrit un ensemble de ces méthodes statistiques en reconnaissance de formes dans [KUN 00].

- *Les méthodes paramétriques*, ces méthodes travaillent sous l'hypothèse que les classes étudiées suivent une distribution de probabilité d'une certaine forme connue a priori. La classification ici consiste à déterminer la classe pour laquelle la forme inconnue présente la probabilité d'appartenance maximale. Ces méthodes nécessitent des bases d'apprentissage assez importantes pour une estimation correcte



des paramètres de la distribution supposée. Elles englobent : la règle de Bayes [BEL 92], les méthodes neuronales et les chaînes de Markov [ELY 96].

- *Les méthodes non paramétriques*, dans ces méthodes, les lois de probabilité sont inconnues pour une de ces classes. Le problème dans ce cas revient à établir des frontières de décision entre les classes. Les techniques les plus utilisées sont : la méthode du plus proche voisin [BEL 92], et la méthode d'appariement de graphes [BEL 92].

### 2.6.5.3 Les approches structurelles

Les approches structurelles (syntaxiques) consistent à mettre en relation la structure des formes analysées et la syntaxe d'un langage formel. Dans ces approches, une classe est décrite par une grammaire. Chaque grammaire est composée d'un ensemble de règles syntaxiques qui déterminent l'ensemble des formes valables dans cette classe et présentant en principe des caractéristiques structurelles communes. Cet ensemble est appelé langage engendré par la grammaire [BOU 09]. La description des formes dans ce cas, est réalisée par l'intermédiaire de phrases, et le problème de classification est ramené à un problème d'analyse de grammaire.

La différence principale entre ces approches et les approches statistiques est que ces caractéristiques sont des formes élémentaires et non pas des mesures. Une autre différence est qu'elles introduisent la notion d'ordre dans la description d'une forme. Parmi toutes les méthodes structurelles disponibles, les arbres de décisions sont sans aucun doute les plus populaires [THE 03].

### 2.6.5.4 Les approches bios inspirées

Les approches bios inspirées sont les plus anciennes et les plus populaires, créées pour deux objectifs principaux. Le premier consiste à la modélisation des systèmes naturels et leur simulation sur ordinateur. Le deuxième implique l'étude des phénomènes naturels pour développer des systèmes informatiques et des algorithmes aptes à résoudre des problèmes complexes.

Ces systèmes ou approches sont aussi connus sous le nom de métaphores biologiques [DEC 07]. On peut citer parmi les approches les plus connues : les réseaux de neurones artificiels, les algorithmes évolutionnaires, l'intelligence en essaim et les systèmes immunitaires artificiels (Figure 2.16).

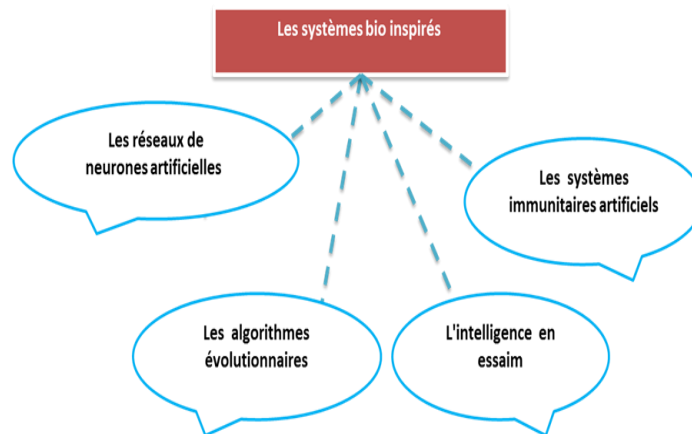


FIGURE 2.16 – Les approches bios inspirées

### 2.6.5.4.1 Les réseaux de neurones artificiels (RNAs)

#### A. Principes de conception

Un RNA est considéré comme un système de traitement de l'information conçu avec une inspiration qui provient du système nerveux. Les neurones sont les unités de base utilisées pour le calcul dans le cerveau, et leurs modèles simplifiés sont les unités de traitement de base de RNAs. Un neurone naturel (NN) est une cellule composée, indépendamment de leur taille et de leur forme, d'un corps cellulaire (soma) qui contient le noyau, des dendrites qui forment une arborisation autour du corps, d'un axone qui est un prolongement de corps jusqu'à leur extrémité, et des synapses qui forment une arborisation de contact avec les dendrites des autres neurones (Figure 2.17 (a)). Une caractéristique importante des RNAs est la représentation distribuée et le traitement parallèle de l'information (connaissance). Le système nerveux est modifié et mis à jour d'une manière continue. Notamment, ses fonctions sont modifiables par expérience.

Un RNA peut être défini comme ensemble de neurones distribués dans une architecture parallèle et fortement connectés ou chaque neurone calcule une sortie en fonction de l'information qu'il a reçue en entrée [TOU 92]. Il présente un certain nombre de caractéristiques et de fonctionnalités en commun avec le RNN comme :

- Le traitement de l'information de base se produit dans de nombreux éléments simples appelés des *neurones* (artificiels), des *nœuds* ou des *unités*.
- Ces neurones peuvent *recevoir* et *envoyer* des *stimuli* entre eux à travers un environnement.
- Les neurones peuvent être reliés les uns aux autres formant un ou des réseaux de neurones.
- Les informations (signaux) sont transmises entre les neurones via des liens de connexion appelés *synapses*.
- L'efficacité d'une synapse, représentée par une résistance associée ou valeur de poids, correspond à l'information stockée dans le neurone, donc dans le réseau.
- La connaissance est acquise à partir de l'environnement par un processus connu sous le nom d'apprentissage, qui est essentiellement chargé d'adapter les forces de connexion (les valeurs de poids) aux stimuli environnementaux.

Du point de vue conceptuel, un RNA peut être caractérisé par trois caractéristiques principales : (1) un ensemble de neurones artificiels, appelés également *nœuds*, *unités*,

ou tout simplement *neurones*, (2) La forme de la connectivité entre les neurones, appelée *l'architecture* ou la *structures* du réseau, et (3) une méthode pour déterminer les valeurs des poids, appelée leurs algorithmes d'*entraînement* ou d'*apprentissage*.

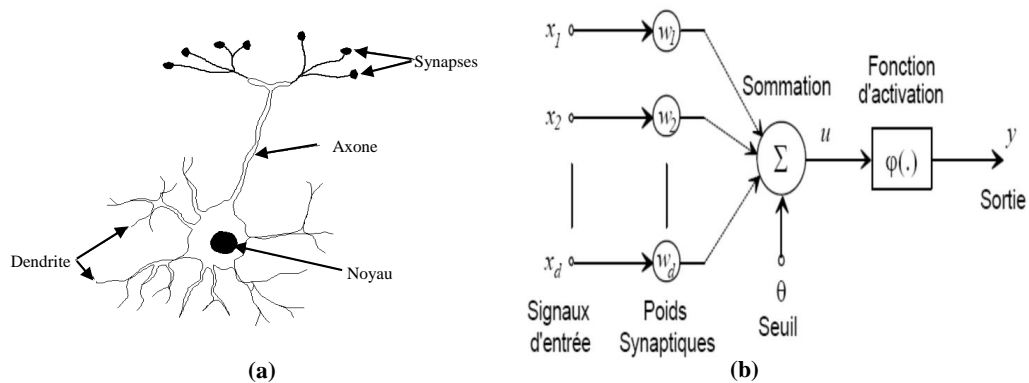


FIGURE 2.17 – Un réseau de neurones. (a) modèle biologique, (b) modèle artificiel

Selon la figure 2.17 (b), un neurone artificiel fait une somme pondérée des potentiels d'actions qui lui parviennent (chacun de ces potentiels est une valeur numérique qui représente l'état du neurone qu'il a émis), puis s'active suivant la valeur de cette sommation pondérée. Si cette somme dépasse un certain seuil, le neurone est activé et transmet une réponse (sous forme de potentiel d'action) dont la valeur est celle de son activation. Si le neurone n'est pas activé, il ne transmet rien.

Donc, l'adaptation de réseau à un problème donné va passer par un choix de cette topologie et les poids de liaison entre les neurones correspondent au processus d'apprentissage (supervisé, non supervisé ou par renforcement) qui est défini par la façon dont les poids sont ajustés.

### B. Avantages et inconvénients des réseaux de neurones

- Les RNAs sont des outils statiques de classification, en raison de leur propriété d'approximation parcimonieuse, ils se sont révélés d'être des classificateurs très performants.
- Leur grand avantage réside dans leur capacité d'apprentissage automatique, ce qui permet de résoudre des problèmes sans avoir recours à l'écriture des règles complexes, tout en étant tolérant aux erreurs.
- La faculté d'apprentissage à partir d'exemples représentatifs, par la technique de rétro propagation des erreurs.
- En segmentation, le réseau de neurones détermine de lui-même combien de groupes (clusters) distincts renferment chaque classe.
- Comportement moins mauvais en cas de faible quantité de données.
- Aptitude à modéliser des structures complexes et des données irrégulières.
- La connaissance acquise par un réseau de neurones est codée par les valeurs des poids synaptiques, les réseaux de neurones sont donc des boites noires où les connaissances sont inintelligibles pour l'utilisateur et ils ne permettent pas d'interpréter les modèles construits. En cas d'erreurs système, il est quasiment impossible d'en déterminer la cause.

- Manque des techniques d'initialisation, le choix des valeurs initiales des poids du réseau et le réglage du pas d'apprentissage jouent un rôle important dans la vitesse de convergence.
- Difficile à paramétrer, par exemple dans le cas de la perception multicouche, il est difficile de définir le nombre de neurones dans les couches cachées (l'absence de méthode systématique permettant de définir la meilleure topologie du réseau et le nombre de neurones à placer dans la/les couche(s) cachée(s)).
- La mise en œuvre, qui passe par un apprentissage, peut être longue.
- Le problème du sur-apprentissage.

### **C. Domaines d'Application**

Parmi toutes les techniques bios inspirées, le RNA constitue le domaine le plus ancien et le plus vaste. Il permet, comme leur principe indique, de calculer toute fonction calculable. Il est particulièrement utilisé pour : l'optimisation et l'apprentissage, le clustering, la classification, la RdF, les problèmes d'approximation des fonctions (mapping)...etc.

#### **2.6.5.4.2 Les approches évolutionnaires (AEs)**

##### **A. Principe de conception**

Les approches évolutionnaires (AEs) sont basées sur la biologie évolutive (phénomène de la descendance; génération après l'autre), qui stipule essentiellement qu'une population d'individus capable de se reproduire et soumise à des variations génétiques suivit par une sélection de résultats dans des nouvelles populations d'individus de plus en plus adaptés à leur environnement. Un algorithme évolutionnaire standard peut ainsi être caractérisé par :

- Une population d'individus capables de se reproduire : chaque individu représente ou encode un point dans l'espace de recherche des solutions potentielles à un problème.
- La programmation génétique est basée sur les algorithmes génétiques et les individus (ou chromosomes) sont des programmes informatiques représentés en une structure d'arbre.
- Ces individus peuvent se reproduire, et génèrent ainsi une descendance qui héritera certains traits de leurs parents.
- Les stratégies d'évolution, initialement conçues pour résoudre des problèmes à variables continues, servent à modéliser les paramètres stratégiques qui contrôlent la variation dans l'évolution, autrement dit "l'évolution de l'évolution".
- La variation génétique : la progéniture est sujette à une variation génétique par mutation, qui modifie sa composition génétique. La mutation permet l'apparition de nouveaux caractères dans la descendance et, par conséquent, l'exploration de nouvelles régions de l'espace de recherche.
- La sélection naturelle : une mesure de fitness pour chaque individu est calculée afin d'évaluer leur adaptation à l'environnement. Une comparaison des fitness individuelles mène à une compétition pour la survie et la reproduction dans l'environnement. Cela donne un avantage sélectif pour les individus ayant une meilleure fitness.

Le schéma global d'un système évolutionnaire est montré dans la figure 2.18.

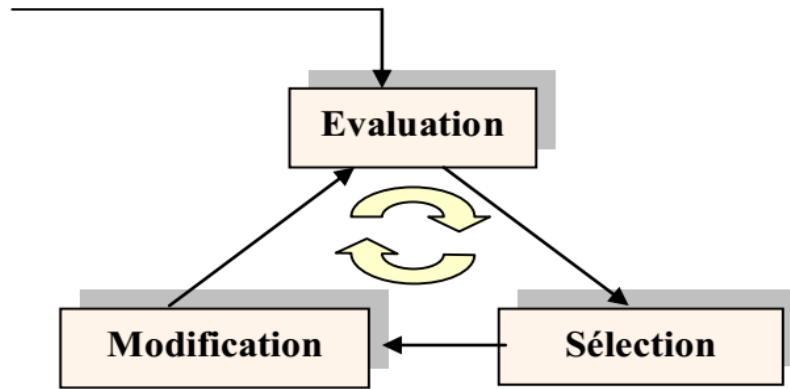


FIGURE 2.18 – Schéma globale d'un système évolutionnaire [BEN 11]

Selon la figure 2.18, après une connaissance du problème, le principe général d'un algorithme évolutionnel est comme ci : **manipulation** des **entrées**, **lecture** des **sorties**, et à **nouveau** manipulation des entrées afin d'**améliorer** les sorties.

L'abstraction des procédures du AE a donné lieu à des algorithmes dits évolutionnaires. Ces derniers sont scindés en cinq grandes familles d'algorithmes, et les différences entre elles ont laissé des traces dans le paysage évolutionnaire actuel, en dépit d'une unification de nombreux concepts [LEP 12]

- Les *algorithmes génétiques* modélisant l'évolution génétique sont initialement conçus pour résoudre des problèmes d'optimisation à variables discrètes.
- La *programmation génétique* est basée sur les algorithmes génétiques et les individus (ou chromosomes) sont des programmes informatiques représentés en une structure d'arbre.
- La *programmation évolutionnaire* historiquement conçue pour des problèmes d'apprentissage à partir d'automates à états finis, œuvre directement sur le phénotype en utilisant une succession de sélections et de mutations.
- Les *stratégies d'évolution*, initialement conçues pour résoudre des problèmes à variables continues, servent à modéliser les paramètres stratégiques qui contrôlent la variation dans l'évolution, autrement dit « l'évolution de l'évolution ».
- L'*évolution différentielle*, initialement conçue pour résoudre des problèmes à variables continues, consiste à biaiser un opérateur de mutation appliqué à un individu en fonction des différences calculées entre d'autres individus sélectionnés aléatoirement.

#### **B. Avantages et inconvénients des approches évolutionnaires :**

- Ils sont adaptés aux espaces de recherche complexes contrairement aux algorithmes évolutionnaires, la définition des autres méthodes heuristiques à des problèmes complexes est une tâche très difficile.
- Ils opèrent au niveau du codage des paramètres sans se soucier de leur nature, donc ils s'appliquent à de nombreuses classes de problèmes, qui dépendent éventuellement de plusieurs paramètres de natures différentes (booléens, entiers, réels, fonctions,...).
- Ce sont des méthodes robustes à l'initialisation (leurs convergences ne dépendent pas de la valeur initiale comme les RNs) qui permettent de déterminer l'optimum global d'une fonctionnelle ou de s'en approcher, et ils sont parallélisables.

- La difficulté d'ajustement de paramètres constitue le principal problème des algorithmes évolutionnaires, certains algorithmes sont conçus pour résoudre des problèmes spécifiques et leur adaptation à d'autres situations sera très difficile.
- Ils sont très sensibles à la fonction d'évaluation et un seul changement dans le problème peut mener à un comportement tout à fait différent.
- La nature heuristique des algorithmes évolutionnaires implique, par définition, qu'ils constituent des méthodes d'approximation qui ne donnent aucune garantie d'exactitude ou d'optimalité. En outre, il n'existe aucune façon de prédire leurs temps d'exécution. Leur inconvénient majeur réside dans le nombre important d'évaluations nécessaires et leur temps de convergence.

### **C. Domaines d'Application**

Ces méthodes s'emploient pour l'optimisation de fonctions numériques difficiles, le traitement d'image, le contrôle de systèmes industriels, la planification (routage, ordonnancement, conditionnement,...). Elles sont aussi utilisées pour contrôler un système évoluant dans le temps (l'adaptation de la population à des conditions changeantes).

#### **2.6.5.4.3 L'intelligence en essaim (IE)**

Plusieurs définitions de l'intelligence en essaim peuvent être trouvées dans la littérature [DEC 07].

« L'intelligence en essaim comporte toute tentative de concevoir des algorithmes ou des outils distribués de résolution de problèmes, inspirée par le comportement collectif des insectes sociaux et d'autres sociétés animales » [BON 99].

On peut accorder à cette intelligence collective une autre définition comme : « C'est l'auto organisation d'un groupe qui fait émerger un comportement inattendu ».

##### **A. Principe**

L'intelligence en essaim regroupe principalement deux axes de recherche : le premier appuie sur les travaux basés sur les insectes sociaux (fourmis, abeilles, sauterelles, termites, moustiques et insectes migrants), et le deuxième porte sur les travaux basés sur la capacité des sociétés humaines à traiter des connaissances (les bactéries, les oiseaux, et les animaux aquatiques/poissons). Bien que les deux approches soient tout à fait différentes dans leur enchaînement d'étapes et leurs sources d'inspiration, ils présentent certains points communs. En général, les deux approches s'appuient sur une population (colonie ou essaim) d'individus (insectes sociaux ou particules) capables d'interagir (directement ou indirectement) entre eux à travers un environnement. On peut citer parmi ces approches : l'algorithme d'optimisation par essaim de particule (PSO) et l'algorithme d'optimisation par les colonies de fourmis (ACO).

##### **B. Avantages et inconvénients de l'Intelligence en essaim**

- Ce sont des méthodes flexibles, elles s'adaptent aux modifications de l'environnement, et robustes, même lorsqu'un des composants n'est plus en état de remplir son rôle, le groupe peut continuer à exécuter sa tâche. On parle ici de l'auto-organisation.
- Elles n'ont pas des comportements très exploratoires ce qui a conduit à la nécessiter d'une hybridation avec d'autres méthodes de recherche locales pour améliorer la précision des solutions trouvées.
- Les algorithmes multi-essaims sont plus complexes et comportent plus de paramètres à régler.

### C. Domaines d'Application

Elles sont conçues pour résoudre les problèmes d'optimisation discrète, d'ordonnement séquentiel, d'affectation, du voyageur de commerce, du routage réseau, de déterminer la stratégie de contrôle, etc.

#### 2.6.5.4.4 Les systèmes immunitaires artificiels (AISs)

##### A. Principe

Le système immunitaire (SI) est le responsable de nous protéger contre l'attaque des micro-organismes externes avec plusieurs mécanismes de défense, plusieurs niveaux, et une certaine redondance. Il inclut l'*apprentissage* et la *mémoire*.

Les systèmes immunitaires artificiels (AISs) incluent tout système ou outil de calcul qui extrait les idées et les métaphores du système biologique immunitaire. Les éléments de base d'un AIS sont :

- La *maturation d'affinité* : promouvoir l'apprentissage (l'adaptation) par le biais somatique d'hyper mutation et de sélection.
- La *sélection clonale* : décrit comment les cellules immunitaires et des molécules interagissent avec les antigènes.
- La *sélection négative* : génère un ensemble(s) des détecteurs du non-soi pour la détection d'anomalie.
- Le *Réseau immunitaire* : effectue la dynamique et la métadynamique du système structuré d'une manière en forme de réseau.
- La *théorie du danger* : des cellules présentatrices d'antigène (cellules dendritiques) du système immunitaire sont eux-mêmes activées via une alarme : les signaux de danger.

Le chapitre 3 traite en détail les systèmes immunitaires artificiels.

##### B. Avantages et inconvénients des systèmes immunitaires artificiels

- Ce sont des algorithmes simples avec des propriétés très intéressantes qui les rendent naturellement adaptés à la reconnaissance des formes et des tâches de classification.
- En outre, les chercheurs ont réalisé que cette heuristique peut également être adoptée avec succès pour des tâches d'optimisation. Cependant, pas beaucoup de chercheurs ont exploré ce domaine, puisque la plupart des travaux en cours sur les systèmes immunitaires artificiels se concentrent sur les architectures, les modèles et les applications ne sont pas liés à l'optimisation [COE 07].
- Ce sont des modèles intelligents, inspirés du système immunitaire naturel, pour la solution des problèmes du monde réel.
- Leur principale faiblesse est normalement les paramètres supplémentaires dont ils ont besoin (dont certains peuvent être difficiles à ajuster pour un problème quelconque). En fait, certains de ces paramètres peuvent ne pas être évidents à première vue (par exemple, la proportion des antigènes et des anticorps dans la population, le nombre de clones à produire, etc.), mais peuvent devenir fastidieux lorsque vous essayez de les utiliser.

##### C. Domaines d'Application

Les SIAs sont construits pour l'apprentissage machine et la RDF, la détection des anomalies et sécurité des systèmes informatiques, l'analyse des données (découvertes des connaissances des BDD, clustering..), les systèmes à base d'agent, la programmation,

la planification, la navigation autonome, le contrôle, la recherche, l'optimisation, la vie artificielle (durée de vie artificielle), etc.

Nous nous intéressons, dans cette thèse, à la technique de reconnaissance basée sur les systèmes immunitaires artificiels (AIS). Cette technique diffère des techniques déjà citées précédemment par le fait que les AIS ont la capacité de mémorisation et d'adaptation, et qui sont très utiles dans le domaine de reconnaissance d'écriture manuscrite. Le chapitre qui suit présente une description plus détaillée de ces systèmes.

### 2.6.6 Le post-traitement

Dans le but de vérifier et améliorer le taux de reconnaissance, le post-traitement est effectué pendant ou après l'étape de reconnaissance en faisant des corrections orthographiques ou morphologiques à l'aide de dictionnaires de digrammes, tri-grammes ou n-grammes [BEL 01]. Cette étape se charge également de vérifier si le résultat de reconnaissance est correct (même si elle est unique) en se basant sur d'autres informations non disponibles au classificateur [MEN 08].

## 2.7 Problèmes liés à la reconnaissance d'écriture manuscrite

La tâche de la reconnaissance automatique n'est pas toujours facile, divers problèmes compliquent le processus de reconnaissance, parmi lesquels nous citons les plus importants [AKR 98 ; FAR 09] :

- **La qualité du document** : un document télécopié ou photocopie plusieurs fois est plus difficile à traiter que la copie originale. L'écriture devient plus mince ou au contraire plus épaisse, dégradée avec les parties de texte qui manquent ou des tâches qui apparaissent, des ouvertures ou des bouchages de boucles.
- **L'impression** : un document composé est de meilleure qualité qu'un document dactylographié qui, à son tour, est plus clair qu'un texte issu d'une imprimante matricielle. Une imprimante à jet d'encre peut introduire des taches d'encre et un étalement des caractères. Un laser peut générer des lignes ou des fonds.
- **Le support de l'information** : tel que le papier, joue également sur les performances de la reconnaissance par sa qualité : son grammage, sa granulation et sa couleur.
- **L'acquisition** : la numérisation en temps réel introduit souvent des distorsions dans l'image. Dans le cas hors ligne, la qualité du texte numérisé est un compromis entre les variations de la position du papier, la propriété de la vitre du dispositif de numérisation et sa résolution.

Les problèmes posés par la reconnaissance de l'écriture manuscrite, l'objet de notre travail, sans aucune contrainte sont plus complexes que ceux liés à l'écriture imprimée. Les erreurs de lecture dans le cas du manuscrit sont dues aux variations infinies de l'écriture de nature aléatoire, qui dépendent de plusieurs facteurs.



## 2.8 Les difficultés de la reconnaissance de l'écriture manuscrite

La disposition spatiale peut être influé par des contraintes externes, qui peuvent être imposées au scripteur, et des contraintes internes à l'écriture proviennent des habitudes propres à chaque scripteur.

Les contraintes externes conduisant à une écriture :

- Prés casés : l'utilisation doit s'efforcer d'écrire à l'intérieur des cases prédéfinies.
- Zonée : le scripteur écrit dans des zones bien limitées.
- Guidée : dans ce cas, l'écriture est guidée par l'existence d'une ligne support.
- Générale : correspondant à une écriture à emplacement totalement libre.

Parmi les contraintes internes, on trouve :

- Lettres séparées.
- Groupes de lettres liées à l'intérieur d'un mot.
- Mots entièrement liés.
- Plusieurs de ces cas à l'intérieur d'un même texte manuscrit.

### 2.8.1 Nombre de scripteurs

La difficulté de mise en œuvre d'un système de reconnaissance d'écriture croit également avec sa portée d'utilisation, c'est-à-dire avec le nombre d'utilisateurs ou scripteurs. Donc, nous pouvons citer les systèmes suivants :

- **Système mono-scripteur** : C'est le système le plus simple avec apprentissage de l'écriture propre à l'utilisateur considéré.
- **Système multi-scripteur** : présente un degré de complexité supérieur, dans ce cas le système doit s'adapter à l'écriture de plusieurs scripteurs.
- **Système multi-mono-scripteurs** : se considère comme une simple extension du cas mono-scripteurs, le système s'adapte à l'écriture de chaque utilisateur après que celui-ci se soit fait connaître par son nom.
- **Système omni-scripteurs** : C'est le système le plus général, car il s'agit de reconnaître l'écriture de n'importe quel utilisateur sans que la notion d'identité du scripteur n'intervienne.

### 2.8.2 Taille du vocabulaire

Le problème à résoudre dans ce cas dépend de la taille du vocabulaire à reconnaître. Prenons l'exemple de la reconnaissance des mots manuscrites, on peut distinguer deux catégories d'applications :

- **Les applications à vocabulaire limité** : le nombre de mots à reconnaître constitue un lexique de taille réduite (<1000 mots). La complexité est moindre, car la réduction du nombre limite l'encombrement mémoire et favorise l'utilisation de méthodes de reconnaissance directes et donc rapides, par balayage systématique de l'ensemble des mots du lexique.

- **Les applications à vocabulaire très étendu** : le nombre de mots à reconnaître constitue un lexique étendu (>1000 mots) formant un dictionnaire. Cela va provoquer des problèmes d'encombrement de la mémoire et de temps d'accès à chaque mot. Dans ce cas, les méthodes les plus efficaces sont des approches de type recherches arborescentes avec raffinements successifs.

D'autres types de critères peuvent influencer sur la complexité des systèmes de reconnaissances. Ils sont relatifs aux variations intrinsèques de l'écriture, dans un contexte cursif. Parmi ces variations, on peut noter celles :

- **Propres au scripteur** : traduisant le style personnel en termes de rapidité, de continuité et de régularité. Tous ces éléments influent sur la forme des lettres (écriture penchée).
- **Propres à l'écriture manuscrite** : la forme d'une lettre dépend de sa position dans le mot (début, milieu, fin) ainsi que des lettres voisines. Toutes ces variations vont conduire à des formes morphologiques différentes d'une même lettre, appelées allographes.

### 2.8.3 Les contraintes morphologiques

Les difficultés de mise en œuvre d'un système de reconnaissance d'écriture augmentent en fonction de sa capacité plus ou moins grande, à prendre en considération plusieurs types de modèles distincts, et à tolérer des variantes par rapport à ces modèles afin d'imposer de moins en moins de contraintes à l'utilisateur. On peut distinguer trois niveaux successifs :

- Un premier niveau correspondant à la reconnaissance d'un nombre limité d'entités isolées.
- Un second niveau correspondant à la reconnaissance d'entités plus globales.
- Le troisième niveau est celui de la reconnaissance globale des textes manuscrits qui peuvent comporter plusieurs types d'écritures différentes comme des retouches.

## 2.9 Conclusion

Ce chapitre dévoile les principaux aspects et notions liés au domaine de la reconnaissance d'écriture manuscrite. Nous avons en premier lieu exposé les différents types de reconnaissance, ensuite nous avons cité d'une manière générale les différentes distinctions faites entre les formes à reconnaître, et les étapes suivies pour la réalisation d'un système de reconnaissance. Nous avons également détaillé chaque étape de ce processus de reconnaissance avec analyse et spécification des différentes méthodes et techniques qui y sont développées et utilisées.

Nous avons en premier lieu exposé les différents types de reconnaissance, ensuite nous avons cité d'une manière générale les différentes distinctions faites entre les formes à reconnaître, et les étapes suivies pour la réalisation d'un système de reconnaissance. Nous avons également détaillé chaque étape avec analyse et spécification des différentes méthodes et techniques qui y sont développées et utilisées.

Nous avons aussi donné une description des approches de classification utilisées dans ce domaine, en se basant sur les approches bio inspirées. Enfin, nous avons énuméré les

principaux problèmes et difficultés rencontrés par la reconnaissance de l'écriture manuscrite.

Dans le chapitre suivant, nous allons introduire la théorie des systèmes immunitaires artificiels, et nous mettrons l'accent sur les différents modèles conçus de ces outils en soulignant leurs particularités. En effet, les systèmes de reconnaissances développés et utilisés au cours de cette thèse sont basés sur ces outils.

# Les systèmes immunitaires artificiels

## 3.1 Introduction

La défense de notre corps se fait par le système immunitaire (SI) qui détecte le dysfonctionnement de ses propres cellules et empêche les cellules étrangères de pénétrer dans le corps. Ce système biologique a des capacités intelligentes de détection et de reconnaissance de l'antigène (soi/non-soi) au sein du corps. Inspiré par ce mécanisme biologique, l'informatique rencontre l'immunologie, et plus particulièrement, un champ émergent de celle-ci connu sous le nom de : l'immuno-informatique, le calcul immunologique, ou tout simplement le système immunitaire artificiel (AIS).

Le système immunitaire artificiel (AIS) est l'une des approches bio inspirées, qui couvre un ensemble de techniques de l'intelligence artificielle (IA) et l'apprentissage machine (AM) qui possèdent des propriétés non linéaires de classification et adoptent les concepts de système immunitaire naturel (NIS). Il a été prouvé comme l'un des mécanismes qui possèdent une bonne capacité dans la résolution de problèmes complexes tels que la RdF.

Nous exposons dans ce chapitre les notions de base relatives aux AISs et leurs applications. Ce chapitre est composé de deux parties principales dont la première partie est consacrée à la présentation du système immunitaire biologique (NIS), en montrant les différents types d'immunité, les divers composants immunitaires et les différents mécanismes utilisés par ce système. Ensuite, nous synthétisons les propriétés intéressantes de ce système, qui constituent d'un point de vue informatique une source d'inspiration très riche. La deuxième partie est consacrée à la définition de l'AIS ainsi que leur processus de conception. Ensuite, nous présentons les différents algorithmes et les modèles immunitaires existants.

## 3.2 Le système immunitaire naturel (biologique)

Le système immunitaire naturel NIS est un système de défense complexe capable de protéger le corps humain de n'importe quelle entité étrangère et dangereuse appelée : antigène. Il distingue les éléments du corps le « *soi* ou *self* » et les éléments étrangers au corps « *non-soi* ou *non-self* » comme : les virus, les bactéries, les champignons et autres parasites. Ces derniers vont être identifiés par les cellules et les molécules du système immunitaire en provoquant une réponse immunitaire [TIM 00]. On appelle réponse immunitaire, d'une manière générale, le déclenchement du système immunitaire face à une

maladie.

Il est bien connu que le système immunitaire utilise plusieurs lignes de défense qui travaillent d'une manière parfois parallèle, d'autres fois séquentielle (Figure 3.1) [TIM 00]. Selon le type de l'agent pathogène et la façon dont il pénètre dans le corps, le système immunitaire utilise des mécanismes de réaction différents, soit pour neutraliser l'effet pathogène ou de détruire les cellules infectées [BEN 11]. Un aperçu détaillé du système immunitaire peut être trouvé dans de nombreux manuels, par exemple dans [Gol06].

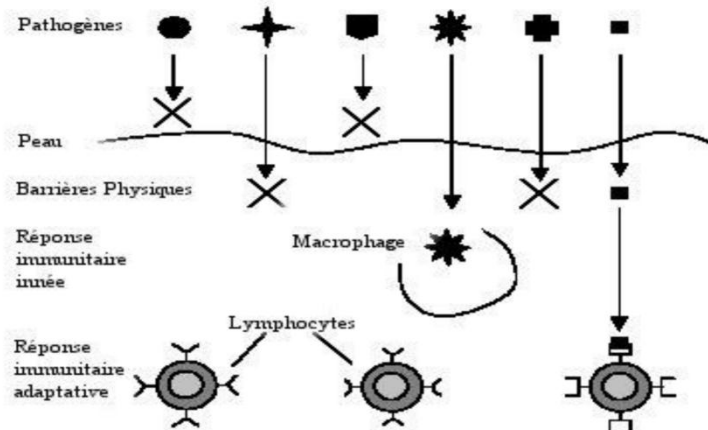


FIGURE 3.1 – Différentes lignes de défense dans un système immunitaire naturel [TIM 00]

Il existe plusieurs types de cellules et molécules immunitaires, les plus importantes sont les globules blancs ou lymphocytes, appelées les cellules T et les cellules B. Les cellules T sont essentielles pour l'activation des cellules B qui peuvent produire un type spécifique d'anticorps (paratope). Les anticorps produits se raccordent à des parties spécifiques d'antigènes, appelées épitopes, pour faciliter leurs éliminations. Le degré de cette liaison ou affinité sera très fort lorsque les deux formes sont complémentaires. Une fois l'élimination accomplie, les cellules B et T se transforment en cellules mémoires pour une éventuelle attaque de même intrusion [VAM 09 ; NEB 13a].

#### 3.2.1 Les différents types d'immunité

Selon la figure 3.1, on peut dire qu'il existe trois lignes de défense pour une réponse immunitaire [DAS 99]. Les deux premières sont dites *innées* ou *non spécifiques* car elles agissent, quelle que soit la nature de l'intrusion. La réponse ou bien l'immunité innée peut comporter des barrières physiques telles que la peau et les voies respiratoires, et des obstacles physiologiques tels que les acides de l'estomac et les enzymes destructrices dans les phagocytoses. La troisième ligne est dite *adaptative* (spécifique ou acquise), car elle est élaborée en fonction du type d'intrusion. L'immunité adaptative est aussi composée de deux parties : l'immunité humorale et l'immunité à médiation cellulaire.

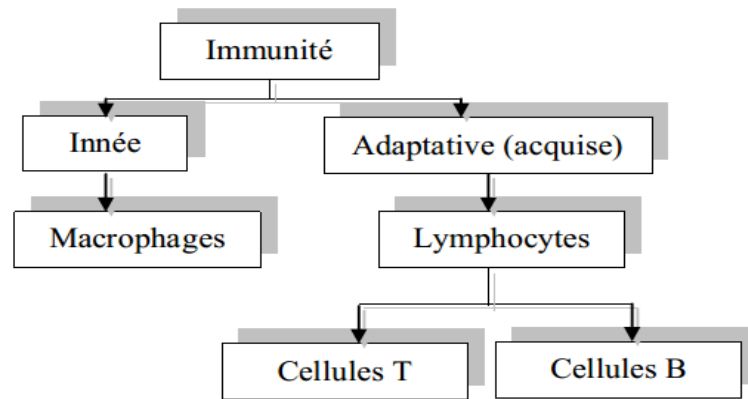


FIGURE 3.2 – Architecture du système immunitaire.

#### 3.2.1.1 L'immunité innée

L'immunité innée constitue la première ligne de défense face à une infection. Elle agit en ne tenant pas compte (non spécifique) du type d'intrusion (maladie) qu'elle combat. Elle est présente à la naissance, mise en jeu immédiatement et fonctionnelle 96 heures. Elle constitue de défenses cellulaires (phagocytes) et chimiques (enzymes, acides gras, acide lactique, flore intestinale, ph, etc.) qui fournissent des conditions de vie inappropriées pour que le nombre d'antigènes ne puissent survivre. On outre, elle n'a pas la possibilité de développer de mémoire à l'égard des intrus. Cependant, ce type d'immunité ne peut pas assurer la protection complète du corps, il est caractérisé par [TIM 04 ; KIM 00] :

- Les mécanismes de détection des organismes étrangers sont constants, aussi bien pour les infections répétées.
- La réponse du système immunitaire inné est non spécifique à un type particulier d'intrus, mais elle est identique contre tous les pathogènes qui envahissent le corps.
- Il joue un rôle vital pour l'initialisation et la régularisation de la réponse immunitaire adaptative.

#### 3.2.1.2 L'immunité adaptative

L'immunité adaptative comporte différents types de cellules dont chacune joue un rôle important. Dans ce type d'immunité, les propriétés d'apprentissage, d'adaptabilité et de mémorisation, caractérisent le résultat de contact du système immunitaire avec les antigènes. La réponse ici peut durer plusieurs jours avant d'être efficace. Si l'antigène a déjà essayé d'investir le corps auparavant et est reconnu, elle est bien plus rapide, grâce à la faculté de mémorisation. Elle se compose principalement de deux types d'immunités : humorale (par les anticorps contenus dans les liquides corporels) et à médiation cellulaire (se base sur une médiation cellulaire). La réponse de l'immunité adaptative est lancée après la réponse de l'immunité innée, et les deux sont liées et se complètent. Ce type d'immunité a aussi des caractéristiques comme [TIM 04 ; KIM 00] :

- Il s'occupe des intrus qui ne sont pas détectés par le système immunitaire inné.
- Il est généré dynamiquement contre les organismes étrangers pendant sa durée de vie. Il fournit des mécanismes plus efficaces qui seront adaptés aux changements antigéniques.

- Il est adressé à des intrus spécifiques.
- La présence d'une mémoire immunologique qui permet aux cellules de se souvenir des intrus déjà rencontrés lors des prochaines rencontres.

#### 3.2.2 La physiologie du système immunitaire naturel

Le processus de génération et de développement de cellules immunitaires est assuré principalement par deux organes : la *moelle osseuse* et le *thymus*. La moelle osseuse est un tissu graisseux présent dans les os, dont la fonction est de fabriquer les cellules du sang, et est aussi le lieu de développement de certaines catégories de ces cellules. Le thymus est un organe lymphoïde qui participe à la maturation des lymphocytes T sous l'influence des hormones thymiques [DEC 99].

Le NIS possède plusieurs types de cellules immunitaires, mais cette étude sera focalisée principalement sur les *lymphocytes* : les *cellules B* et *T* [DEC 03 ; DEC 99]. Les lymphocytes qui se développent dans la moelle osseuse sont nommés des cellules B et ceux qui migrent et se développent dans le thymus sont nommés des cellules T. Ces cellules possèdent des récepteurs qui sont situés sur leur surface pour reconnaître des modèles antigéniques.

##### A. Les cellules B

Les lymphocytes B sont responsables de l'immunité humorale, qui vise à produire et à sécréter des anticorps spécifiques comme une réponse aux corps étrangers [DEC 99]. Chaque cellule B ne peut produire qu'un type spécifique d'anticorps. Le récepteur de cellule B est appelé BCR ou bien *anticorps* (Figure 3.3 (a)).

**B. Les cellules T** Les cellules T (T est l'abréviation de thymus) sont responsables de l'immunité cellulaire, qui vise à détruire les cellules pathogènes, que ça soit des bactéries ou des cellules cancéreuses.

Il existe plusieurs types de cellules T : les cellules T *aideuses* (T helper) et les cellules T *cytotoxiques* ou *tueuses* (T killer). Les cellules T aideuses assurent des fonctions nécessaires pour la régularisation de la réponse immunitaire, par exemple l'activation ou la suppression du développement de certains types de réponses immunitaire (l'activation des cellules B). Les cellules T *cytotoxiques* assurent des fonctions de suppression des envahisseurs microbiens, des virus ou des cellules cancéreuses en injectant des produits toxiques pour les tuer. Une autre variante de cellules T, les *suppresseurs*, servent à éviter les réactions immunitaires non appropriées (maladies auto-immune). Ainsi, les cellules T présentent des récepteurs TCR sur leur surface comme le montre la figure 3.3 (b).

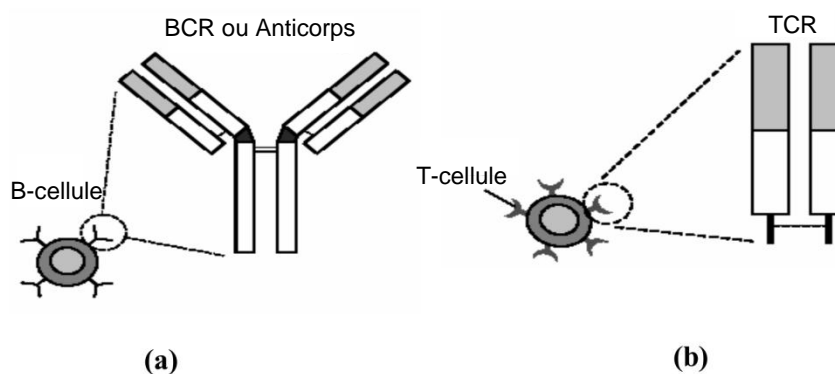


FIGURE 3.3 – Présentation d'une cellule B et cellule T

### C. Les anticorps

Les anticorps sont des protéines complexes. Leur rôle est de détecter et neutraliser les agents pathogènes de manière spécifique. Ils constituent l'immunoglobuline principale du sang. Ils sont sécrétés par des cellules dérivées des lymphocytes B, appelées les plasmocytes. Il est important d'indiquer qu'il y a plus de dix millions d'anticorps différents dans un organisme, ce qui explique leur spécificité [ALL 14].

### D. Les antigènes

On appelle antigène toute substance étrangère à l'organisme capable de déclencher une réponse immunitaire. Il s'agit le plus souvent de protéines ou des polysaccharides et leurs dérivés lipidiques, qui sont reconnus de manière spécifique par des anticorps et également par les lymphocytes T.

## 3.2.3 Le mécanisme de défense du système immunitaire naturel

Le processus d'élimination de l'antigène est assuré par une armée de cellules et de molécules immunitaires (figure 3.4) [BEN 03a] :

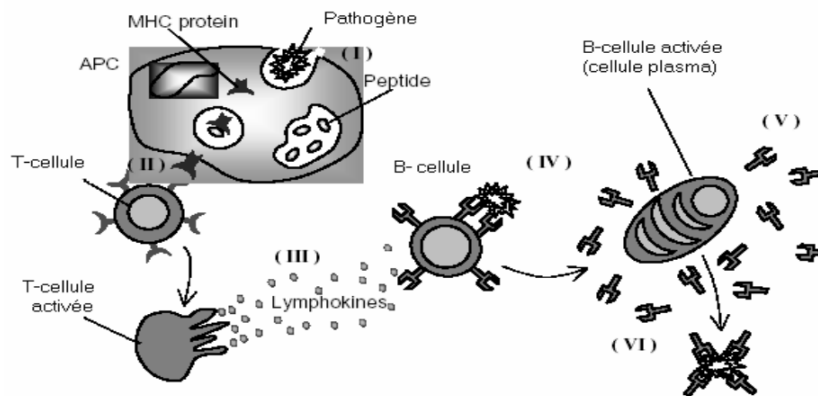


FIGURE 3.4 – Mécanisme de défense du système immunitaire naturel

1. Quand un intrus envahit le corps, les cellules présentatrices d'antigène APC constituant la deuxième ligne de défense comme les macrophages, procèdent à l'ingestion et la digestion de l'antigène rencontré pour le présenter comme des fragments de peptides antigéniques.
2. Ces peptides seront liés avec les molécules MHC (complexe majeur d'histocompatibilité). Le macrophage présente la partie résultante sur sa surface pour permettre leur liaison avec les cellules T qui ont la capacité de reconnaître la combinaison de peptide-MHC.
3. Les cellules T activées par cette identification produisent et sécrètent des signaux chimiques (lymphokines) tout en alertant les autres cellules blanches pour les mobiliser.
4. Les cellules B répondent à ces signaux. À la différence des récepteurs de cellules T, ceux de cellules B peuvent reconnaître les parties d'antigènes libres sans les molécules MHC.
5. Une fois stimulées par les cellules T, les cellules B produisent et se différencient et sécrètent des récepteurs spécifiques appelés anticorps.



6. Les anticorps produisant se raccordent aux antigènes pour faciliter leurs destructions et leurs suppressions, et cela par différents mécanismes.
7. Un nombre de cellules B et T se transforment en cellules mémoires qui ont une durée de vie illimitée, pour une éventuelle attaque de la même intrusion.

#### 3.2.4 Propriétés du système immunitaire naturel

Le NIS fournit des propriétés très intéressantes, et devient une référence précieuse et une source d'inspiration pour la recherche en informatique. Voici quelques-unes de ces propriétés les plus importantes [DEC 02a; TIM 08c] :

- **Multicouche** . Le NIS possède une architecture multicouche qui consiste en deux sous-systèmes interliés : inné et adaptatif. Chaque sous-système a un mécanisme différent et la combinaison entre eux permet d'assurer la plus haute sécurité globale.
- **Unicité**. Chaque élément dans le NIS possède son propre rôle et assume des responsabilités particulières.
- **Adaptabilité** . Le NIS capable de produire des cellules spécifiques pour l'identification des antigènes. Cela est garanti par la théorie de la sélection clonale suivie par le mécanisme de l'hyper-mutation somatique.
- *Autonomie* . Le NIS ne possède aucune gestion ou contrôle extérieur. Il détecte et élimine, d'une façon autonome globale, les intrus, et il se répare et remplace les cellules endommagées.
- **Distribution** . Les cellules, les molécules et les organes immunitaires sont distribués partout dans le corps pour assurer sa protection, et ils ne cèdent à aucun contrôle centralisé.
- **Parallélisme** . Le NIS est capable de produire plusieurs réponses immunitaires en même temps à des endroits répartis.
- **Tolérance au soi** . Le NIS peut différencier entre les cellules de soi et les cellules de non-soi.
- **Apprentissage**. Le NIS peut « apprendre » continuellement des structures nouvelles. Il augmente la capacité d'identification des anticorps à un antigène sélectif (les réponses primaire et secondaire).
- **Dynamique** . Le NIS change continuellement par la production et la reproduction de nouvelles cellules et molécules, et l'élimination des cellules vieilles ou endommagées.
- **Mémorisation** . Un ensemble de cellules constituent l'ensemble des cellules mémoires seront dotées, après une réponse immunitaire à un antigène donné, pour une durée de vie longue, de telles sortes que les futures réponses au même antigène soient plus rapides et plus fortes.
- **Coopération**. Les cellules immunitaires coopèrent leurs capacités pour assurer une meilleure détection et identification et également une réponse plus puissante, par exemple la coopération des cellules T aideuses avec les molécules MHC.
- **Diversité** . Le NIS comporte une variété d'éléments (cellules, molécules, protéines, etc.) qui accomplissent tous le même rôle qui consiste à identifier n'importe quels intrus envahissant le corps. Cette particularité est due à plusieurs mécanismes qui

sont par exemple : l'hypermutation somatique, la reproduction de récepteur, l'identification approximative des intrus, etc. De plus, il existe diverses lignes de défense, telle que la réponse innée et adaptative.

- **Détection** . Le NIS est capable de détecter, d'identifier et de réagir aux intrus que le corps n'a jamais rencontrés auparavant.
- **Robustesse** . Aucun composant du NIS n'est indispensable pour son fonctionnement. La mort des cellules est habituellement équilibrée par la production des autres nouvelles.

## 3.3 Le système immunitaire artificiel (AIS)

Le système immunitaire biologique capable de protéger le corps humain contre une variété énorme de pathogènes étrangers. Dans les dernières années, un nombre de chercheurs ont étudié le succès et la compétence de ce système naturel et ont proposé le *modèle immunitaire artificiel* AIS pour la résolution de divers problèmes du monde réel. Plusieurs approches ont été proposées pour mettre en œuvre les mécanismes de base du NIS [DAS 03]. Cette section sera consacrée à introduire l'AIS avec une présentation de ses modèles de base qui ont été mis en œuvre.

### 3.3.1 Définitions

**Définition 1** . Selon Timmis [TIM 00] : « Un système immunitaire artificiel est un système informatique basé sur les métaphores du système immunitaire naturel ».

**Définition 2** : Dasgupta a défini l'AIS comme suit [DAS 03] : « Le système immunitaire artificiel est la composition de méthodologies intelligentes inspirées par le système immunitaire naturel afin de résoudre des problèmes du monde réel ».

**Définition 3** : Timmis et De Castro [DEC 02a] ont donné la définition suivante : « Les systèmes immunitaires artificiels sont des systèmes adaptatifs inspirés par des théories immunologiques et des observations de fonctions immunitaires, des principes et des modèles, qui seront appliqués à la résolution des problèmes ».

### 3.3.2 Le processus de conception d'un AIS

Le schéma pour concevoir un algorithme, de point de vue quantitatif, exige au moins les éléments de base suivants [DEC 03 ; TIM 04] :

- Une représentation pour les composants du système.
- Un ensemble de mécanismes pour évaluer l'interaction des individus avec l'environnement. Les environnements sont simulés par un ensemble de stimulus d'entrée, une ou plusieurs fonctions d'évaluation.
- La procédure d'adaptation qui dirige la dynamique du système, c'est-à-dire comment son comportement varie dans le temps.

Ce schéma est adopté par Timmis et De Castro [Tim 02] qui ont proposé un processus de conception d'un AIS (voir figure 3.5), qui exige :

- Une représentation des éléments pour créer les modèles abstraits des cellules et d'organes concernés.

- Un ensemble de fonction nommée fonctions d'affinité (*affinité antigénique*) pour évaluer les interactions entre ces éléments artificiels d'une manière quantitative.
- Un ensemble d'algorithmes ou modèles immunitaires adéquats pour diriger la dynamique du système immunitaire artificiel.

La figure 3.5 récapitule les éléments de base pour la conception d'un AIS.

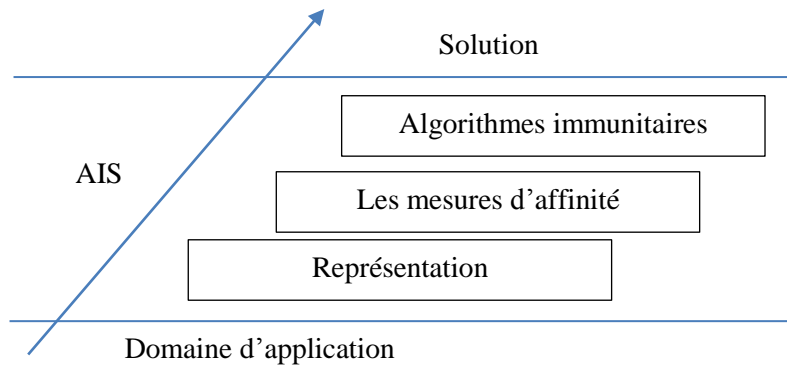


FIGURE 3.5 – Schéma général pour résoudre un problème par un AIS

#### 3.3.2.1 La représentation

D'une manière générale, il est important de noter que la conception d'un AIS pour un problème donné, ce problème est étudié comme étant l'antigène et les solutions de ce problème sont modélisées par les anticorps.

Dans les sections précédentes, les cellules B et T ont été décrites comme les cellules les plus importantes dans le système immunitaire. Elles présentent des récepteurs superficiels utiles pour la reconnaissance des pathogènes étrangers dont les formes de ces récepteurs sont complémentaires à la forme d'antigène. Les cellules et les molécules immunitaires sont alors les éléments qui doivent être modélisés et utilisés dans les modèles proposés par l'AIS [DEC 03]. Il est connu que chaque antigène agit spécifiquement avec tous les anticorps. Les représentations antigène-anticorps (Ag-Ab) déterminent une distance utilisée pour calculer ou bien évaluer le degré de complémentarité ou d'interaction entre ces molécules. Plusieurs méthodes de représentation ont été proposées. La représentation *Shape-Space* est la plus couramment utilisée.

Le modèle *Shape-Space* (Forme-Espace) a été introduit par Perelson et Oster [PER 79] en 1979. Ce modèle décrit les interactions entre les antigènes et les anticorps (Ag-Ab) d'une façon quantitative. L'ensemble des primitives qui caractérise une molécule est appelé la *forme généralisée* (generalized shape). Dans le NIS, le degré de liaison (le degré de correspondance ou l'affinité) entre le récepteur d'anticorps (Ab ou TCR) et un antigène Ag est désigné par « S ». Ce degré de liaison est mesuré via les *régions de complémentarité* entre les deux éléments, tel qu'il est illustré dans la figure 3.6.

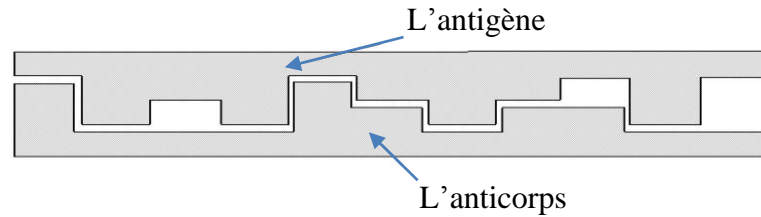


FIGURE 3.6 – La représentation du modèle Shape-Space

Les valeurs du modèle Shape-Space peuvent être réelles, entières, binaires ou symboliques.

### 3.3.2.2 Les mesures d'affinité

Les antigènes et les anticorps ont des représentations vectorielles de la forme :

$$Ab = \langle Ab1, Ab2, \dots, AbL \rangle, \quad Ag = \langle Ag1, Ag2, \dots, AgL \rangle .$$

L'affinité entre un antigène et un anticorps est relative à leur distance, qui peut être estimée par n'importe quelle mesure de distance entre deux chaînes ou vecteurs. Différentes distances existent dont les plus utilisées sont : la distance *euclidienne*, la distance de *Manhattan* ou encore la distance de *Hamming*. Les trois distances peuvent être données respectivement par les équations présentées au tableau suivant [DEC 02b] :

Tableau 3.1 – Les différentes distances et leurs formules mathématiques correspondantes.

Distance	Formule mathématique correspondante
Euclidienne	$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$
Manhattan	$d(x, y) = \sum_{i=1}^n  x_i - y_i $
Hamming	$d(x, y) =  xxory $

Les équations présentées sur le tableau 3.1 montrent comment calculer les affinités entre des molécules par le modèle Shape-Space selon les mesures de distance. Plus la distance Ag-Ab est petite, plus l'affinité entre ces derniers est grande.

### 3.3.2.3 Les algorithmes immunitaires

L'existence et l'identification de certains algorithmes immunitaires a pour objectif d'une part, de donner une représentation appropriée pour les cellules et les molécules immunitaires, et la façon d'évaluer leurs interactions, et d'autre part, de régir la dynamique et la métadynamique du système, c'est-à-dire comment son comportement varie dans le temps. La section suivante détaille les algorithmes immunitaires de base avec leurs applications.

## 3.3.3 Les algorithmes immunitaires de base

Un système immunitaire artificiel est une classe d'algorithmes inspirés par les principes et le fonctionnement du système immunitaire naturel. Cet algorithme exploite géné-

ralement les propriétés du système immunitaire celles de l'apprentissage et la mémorisation [FAR 86].

Parmi les algorithmes des systèmes immunitaires artificiels, on trouve selon le contexte et le problème à résoudre : l'algorithme de sélection négative (NSA) [FOR 98], l'algorithme de sélection clonale (CSA) [DEC 02b], le système immunitaire artificielle de reconnaissance (AIRS) [WAT 04], la théorie des réseaux immunitaires (NIT) [JER 74] et la théorie de danger (DT) [AIC 10]. Une représentation des différentes approches du AIS est donnée dans [DAS 06 ; TIM 08a].

Cette section montre les différents algorithmes et modèles de l'AIS définis dans la littérature, où on présente pour chaque algorithme : l'inspiration biologique, le principe de l'algorithme et ses domaines d'application.

#### 3.3.3.1 L'algorithme de la sélection négative NSA

##### A. L'inspiration biologique

Le processus de suppression des cellules auto-réactives est réalisé par un mécanisme appelé sélection négative. Le but de la sélection négative est de fournir une tolérance aux cellules de soi. Elle porte principalement sur la capacité du système immunitaire à détecter les antigènes inconnus alors que ne réagissant pas aux cellules de soi. L'idée sur laquelle se base cet algorithme est que seules les cellules T qui ne s'attaquent pas aux cellules du soi sont autorisées à quitter le thymus et auront pour tâche de reconnaître les cellules du non soi. Lors de la génération de cellules T, les récepteurs sont construits par un processus de réarrangement génétique pseudo-aléatoire. Ensuite, ils subissent un processus de censure dans le thymus, appelé sélection négative. Dans ce cas, les cellules T qui réagissent contre les protéines du soi sont détruits, tandis que, seules celles qui ne se lient pas aux protéines du soi sont autorisées à quitter le thymus et circulent continuellement dans le tout le corps.

L'algorithme de la sélection négative repose sur les mêmes principes de la génération aléatoire des récepteurs et d'élimination de cellules à récepteurs auto-réactifs.

##### B. L'algorithme

L'algorithme de la sélection négative est inspiré par le mécanisme produisant un ensemble de cellules T matures capables de se lier seulement aux éléments du non-soi. Forrest et al [FOR 98] ont proposé le premier algorithme de la sélection négative qui reflète le principe de la discrimination dans le NIS. Dans ce sens, il est question de distinguer entre les cellules du corps humain (soi) et les agents étrangers (non-soi).

L'algorithme de la sélection négative est considéré comme un processus de détection d'anomalies composé de trois phases principales : la définition du soi, la génération des détecteurs, et le contrôle d'occurrence des anomalies. Il déroulera comme suit (Figure 3.7) :

Étant donné l'ensemble des chaînes de soi ( $P$ ) à être protégé et qui définissent l'état normal du système, générer un ensemble de *détecteurs* ( $M$ ) qui n'identifie aucun élément appartenant à l'ensemble  $P$ . La génération des *détecteurs*  $M$  se passe par les étapes itératives décrites ci-dessous (Figure 3.7 (a)) [DAS 06] :

- Générer des éléments candidats ( $C$ ) d'une façon aléatoire de l'ensemble  $P$ .
- Déterminer l'affinité entre chaque cellule  $C$  et tous les éléments de l'ensemble  $P$ .
- **Si** l'affinité d'un élément  $C$  avec au moins un élément  $P$  est supérieure ou égale à un seuil d'affinité prédéfini, **alors** cet élément  $C$  est considéré comme un élément de soi et sera supprimé.

**Sinon** il sera considéré comme un détecteur de non-soi et sera ajouté à l'ensemble de détecteurs M.

Après avoir obtenu l'ensemble de détecteurs, l'étape suivante de l'algorithme consiste à contrôler le système contre la présence des éléments de non-soi. Chaque élément détecté par les détecteurs générés est considéré comme un élément de non-soi (Figure 3.7 (b)).

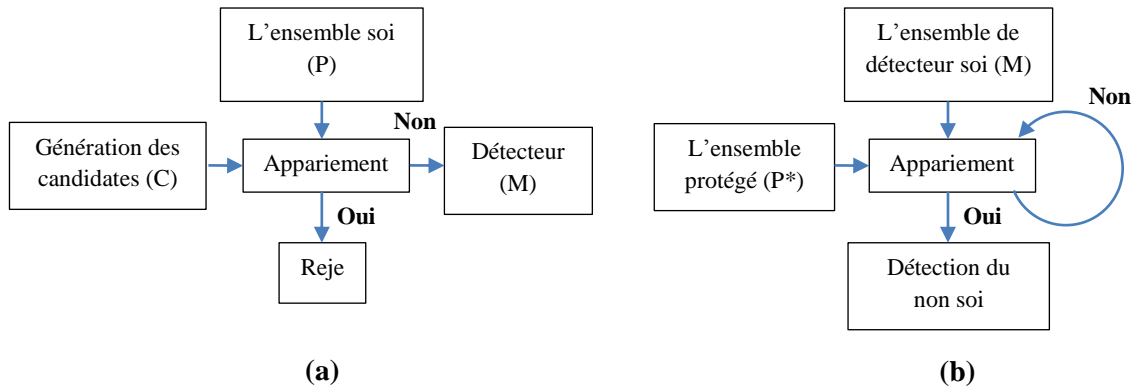


FIGURE 3.7 – Structure générale de l'algorithme de la sélection négative. (a) la phase Examen (Censoring), (b) la phase contrôle (Monitoring)

---

**Algorithme 3.1 :** Algorithme de génération des détecteurs de la Sélection Négative

---

**Données :**  $S$  = ensemble d'éléments du soi déjà identifiés

**Résultat :**  $D$  = ensemble de détecteurs générés

**répéter**

    Générer aléatoirement des détecteurs potentiels et les placer dans un ensemble  $P$

    Déterminer l'affinité de chaque élément de  $P$  avec chaque élément de l'ensemble du soi  $S$

**si** au moins un élément de  $S$  identifie un détecteur de  $P$  selon un seuil d'identification **alors**

        | le détecteur est rejeté,

**sinon**

        | il est ajouté à l'ensemble de détecteurs disponibles  $D$

**fin**

**jusqu'à** critère d'arrêt atteint;

---

### 3.3.3.2 L'algorithme de la sélection clonale CSA

#### A. L'inspiration biologique

La sélection clonale est la théorie expliquant comment le système immunitaire interagit avec les antigènes. Elle présente l'idée que seules les cellules qui reconnaissent l'antigène seront clonées (proliférées) et mutées pour être ainsi sélectionnées. Vu que seules les cellules B sont capables de muter pour optimiser la réponse immunitaire, seules ces cellules nous intéressent. Les AISs s'inspirent de cette théorie. Les caractéristiques de base de cette théorie sont les suivantes :

- Une fois que les cellules B sont en contact avec l'antigène, elles se multiplient et donnent plusieurs clones et chaque clone subit à un mécanisme de mutation avec

- des taux élevés (hypermutation somatique).
- Cette mutation, dite hypermutation, sert à trouver des clones de la cellule mère possédant une plus grande affinité avec l'antigène.
  - Une fois activés, les clones des lymphocytes sont produits, exprimant des récepteurs identiques à la cellule mère ayant rencontré l'antigène. Ainsi, une expansion clonale des lymphocytes d'origine survient. Cela garantit que seule l'activation des lymphocytes spécifiques d'un antigène sont produits en grand nombre.
  - Le principe a également affirmé que toutes cellules ayant des récepteurs spécifiques aux molécules du soi doivent être supprimées au cours du développement du lymphocyte. Ceci garantit que seul l'antigène d'un agent pathogène peut provoquer l'expansion clonale d'un lymphocyte, et donc susciter une réponse immunitaire adaptative destructrice.

Au cours de l'expansion clonale des cellules B, l'affinité des anticorps augmente proportionnellement à l'antigène qui l'a déclenchée à cause de l'hypermutation somatique. Ce phénomène s'appelle la maturation de l'affinité. Elle est responsable pour que lors d'une exposition ultérieure à l'antigène, la réponse immunitaire soit plus efficace grâce à des anticorps ayant une affinité antigénique plus élevée.

#### **B. L'algorithme**

L'algorithme de la sélection clonale a été utilisé comme source d'inspiration pour effectuer des tâches d'optimisation et de RdF [DeC 00]. Cette inspiration appuie sur le principe de maturation d'affinité de l'antigène conduit par les cellules B, avec son mécanisme d'hypermutation associé. Elle exploite souvent l'idée de cellules mémoires pour conserver les bonnes solutions (candidats) au problème à résoudre.

Les étapes nécessaires pour le déroulement d'un algorithme de la sélection clonale sont (Figure 3.8) :

1. Initialisation : commencer tout d'abord par générer aléatoirement un nombre  $r$  d'anticorps (cellules immunitaires) défini qui présente la taille de population initiale.
2. Sélection : sélectionner parmi ces anticorps les  $n$  meilleurs qui ont la plus grande affinité avec l'antigène introduit.
3. Prolifération : cloner cette nouvelle population  $P_n$ .
4. Mutation : muter cette nouvelle population pour augmenter leur degré de correspondance avec l'antigène. La mutation des cellules est effectuée en modifiant un ou plusieurs bits du vecteur représentant la cellule par une autre valeur.
5. Re-sélection : exposer ces clones mutés de nouveau à l'antigène et recalculer leurs affinités. Placer les meilleurs éléments dans l'ensemble des cellules mémoires  $M$ .
6. Remplacement : remplacer les cellules de faible affinité de  $M$  par des nouvelles cellules aléatoirement régénérées. L'introduction de ces nouveaux éléments a pour objectif d'éviter le problème d'optimums locaux.

Le cycle est répété ensuite tant que certains critères sont vérifiés selon le problème à résoudre.

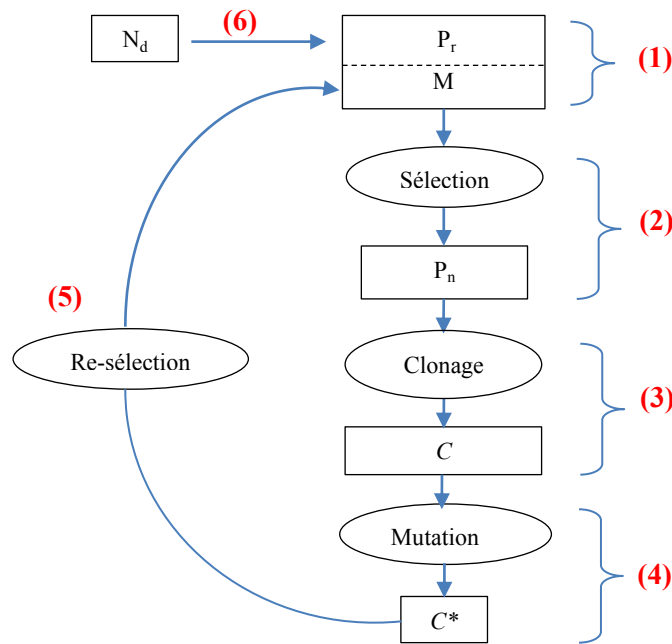


FIGURE 3.8 – Structure générale de l’algorithme de la sélection clonale

L’algorithme de la sélection clonale a plusieurs caractéristiques intéressantes telles que : l’ajustement dynamique de la taille de la population, l’exploration de l’espace de recherche, la localisation des optima multiples, la capacité de maintenir des solutions aux optima locaux et définir des critères d’arrêt.

**Algorithme 3.2 :** Pseudo code de l’algorithme de la sélection clonale.

**Données :**  $AG$  = ensemble de motifs à identifier,

$n$  le nombre de mauvais éléments à sélectionner pour être remplacés.

**Résultat :**  $MC$  = ensemble de détecteurs mémoire capable de classer les modèles inaperçus

Créer aléatoirement un ensemble initial d’anticorps  $P$ .

**pour** tous les modèles en  $AG$  **faire**

- Déterminer l’affinité de chaque anticorps dans  $P$
- Générer des clones d’un sous-ensemble d’anticorps  $P$  selon la plus grande affinité : le nombre de clones d’un anticorps est proportionnel à son affinité.
- Muter les clones de l’ensemble  $P$ , et placer une copie de la plus haute affinité anticorps dans  $P$  dans un ensemble mémoire  $MC$ .
- Remplacer  $n$  anticorps ayant une basse affinité par de nouveaux anticorps générés aléatoirement.

**fin**

### 3.3.3.3 Le réseau immunitaire artificiel AIN

#### A. L’inspiration biologique

Le premier modèle de réseaux immunitaire a été proposé par le biologiste Jerne en 1974 [JER 74]. Le système immunitaire, dans cette théorie proposée, possède un réseau idiotypique d’un ensemble de cellules interconnectées pour la reconnaissance de l’antigène [TIM 08b]. Ces cellules à la fois stimulent et inhibent les uns des autres d’une certaine manière qui mènent à la stabilisation du réseau. La présence d’un paratope et un



idiotype sur chaque cellule d'anticorps rend la constitution d'un tel réseau possible. Le paratope présent sur l'une des cellules B est reconnu par des idiotypes des autres cellules B de sorte que chaque cellule reconnaît ainsi que reconnu. La connexion entre deux cellules de ce réseau est assurée si les affinités qu'ils partagent dépassent un certain seuil, et la force de cette connexion est directement proportionnelle à l'affinité partagée entre eux.

Au point de vue de la construction du réseau, deux choses sont très importantes : la liaison antigène-anticorps et la liaison anticorps-anticorps. Ce réseau idiotypique peut également être considéré comme ayant des capacités cognitives qui le rendent similaire à un réseau de neurones [NAN 09].

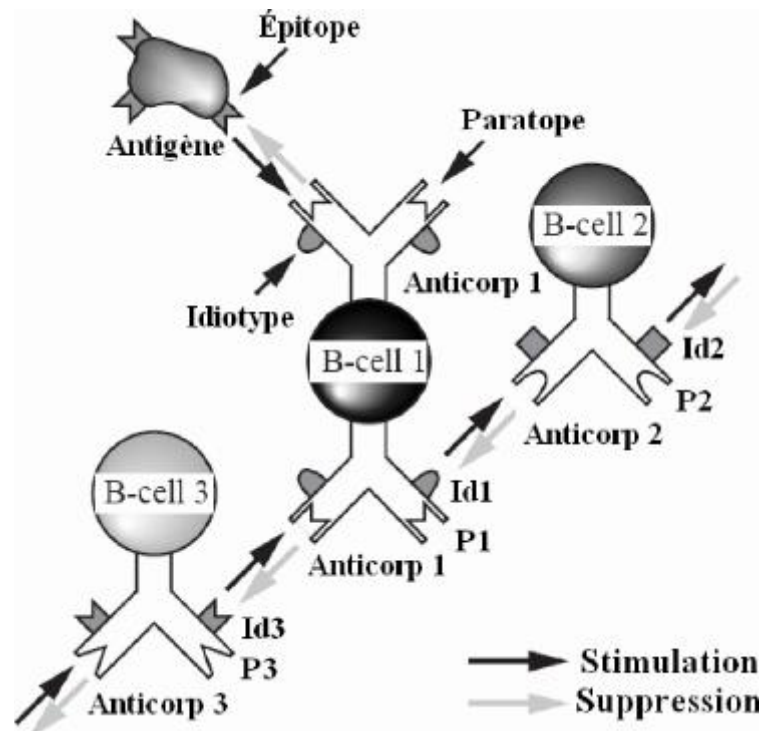


FIGURE 3.9 – La théorie du réseau idiotypique de Jerne [JER 74]

L'idée de Jerne est exposée schématiquement dans la figure 3.9. L'idiotype Id1 de l'anticorps 1 stimule les cellules B-cell 2, par le paratope P2. Du point de vue de l'anticorps 2, l'idiotype Id1 est considéré comme un antigène. En conséquence, l'anticorps 2 supprime les cellules B-cell 1. D'autre part, l'anticorps 3 stimule l'anticorps 1 puisque l'idiotype Id3 fonctionne comme antigène par rapport à l'anticorps 1. De cette façon, les chaînes de stimulation et d'élimination des anticorps forment un réseau à grande échelle et maintiennent la stabilisation de ce réseau [CHA 06a].

### B. L'algorithme

La théorie du réseau immunitaire a suggéré un système immunitaire avec un comportement dynamique (un réseau auto-régulé de molécules et de cellules qui se reconnaissent entre elles) même en absence d'un antigène de non-soi. Le déroulement de cet algorithme peut être récapitulé dans les trois étapes suivantes

1. Initialisation : créer aléatoirement une population initiale d'anticorps.
2. Présentation antigénique : pour chaque modèle antigénique faire :
  - Pour chaque élément de réseau, déterminer son affinité avec l'antigène présenté.

- Sélectionner les anticorps de plus haute affinité et les cloner proportionnellement à leurs degrés d'affinité.
- *Interactions clonales* : déterminer l'interaction réseau (affinité) de tous les éléments de l'ensemble mémoire.
- Chaque clone produit est muté (hypermutation somatique) inversement proportionnel à son affinité antigénique.
- *Méta dynamique* : éliminer tous les clones mémoires dont l'affinité avec l'antigène est moins d'un seuil prédéterminé.
- Sélectionner quelques clones de plus haute affinité pour constituer l'ensemble mémoire dans le réseau.
- *Interactions de réseau* : déterminer la similitude entre chaque paire d'anticorps du réseau.
- Éliminer les anticorps restants qui ont une affinité inférieure à un seuil donné.
- Déterminer l'interaction réseau (affinité) de tous les éléments de l'ensemble mémoire. Éliminer tous ces clones mémoires dont l'affinité avec l'antigène est inférieure à un seuil donné.
- Incorporer les clones restants de l'ensemble mémoire avec les anticorps du réseau.
- Déterminer la similitude entre chaque paire d'anticorps du réseau.
- Éliminer tous les anticorps du réseau dont l'affinité est moins d'un seuil prédéfini.

3. Cycle : répéter ces étapes tant que certains critères sont vérifiés.

---

**Algorithme 3.3** : Pseudo code de l'algorithme du réseau immunitaire.

---

**Données** :  $AG$  = ensemble de motifs à identifier,  $NT$  seuil d'affinité du réseau,  $ct$  seuil du pool clonal,  $h$  le nombre de clones ayant la plus grande affinité,  $a$  le nombre de nouveaux anticorps à introduire.

**Résultat** :  $MC$  = ensemble des détecteurs mémoire capables de classer les modèles inconnus

Créer aléatoirement une population initiale de  $MC$  anticorps du réseau

**répéter**

**pour** chaque modèle antigénique de  $AG$  faire

- Déterminer l'affinité de chaque anticorps en  $MC$
- Générer des clones d'un sous-ensemble d'anticorps à partir de  $MC$  avec la plus grande affinité : Le nombre de clones d'un anticorps est proportionnel à son affinité.
- Muter les clones de l'ensemble  $A$ ,  $a$  et placer  $h$  clones ayant la plus grande affinité dans l'ensemble de mémoire clonale,  $C$ .
- Éliminez tous les éléments de  $C$  dont l'affinité antigénique est inférieure à un seuil prédéfini  $ct$ .
- Calculer l'affinité entre tous les anticorps de  $C$  et éliminer les anticorps dont l'affinité anticorps-anticorps est inférieure au seuil  $CT$ .
- Incorporer le reste des clones de  $C$  dans  $MC$ .

**fin**

- Déterminer l'affinité entre chaque paire d'anticorps en  $MC$  et éliminer tous les anticorps dont l'affinité est inférieure à un seuil  $NT$ .
- Introduire un nombre aléatoire d'anticorps générés aléatoirement dans le réseau.

**jusqu'à** ce qu'un critère d'arrêt soit atteint;

---

### 3.3.3.4 Le système immunitaire artificiel de reconnaissance AIRS

#### A. L'inspiration biologique

La présentation antigénique antigène-anticorps considère l'idée de base pour la construction d'un AIRS. La modélisation des cellules B est la plus rencontrée dans ce type de représentation où elle est utilisée pour mettre en œuvre l'idée de correspondance entre les données d'apprentissage (antigènes) et les solutions possibles (anticorps ou cellules B). L'idée de « *artificial recognition ball* » (ARB) est introduite par Timmis et al [TIM 01], qui a été inspirée par le travail de Farmer [FAR 86] décrivant l'interaction antigénique au sein d'un réseau immunitaire.

Autrement dit, un ARB peut être considéré comme représentant d'un certain nombre de cellules B identiques ; c'est un mécanisme utilisé pour réduire les redondances et de dicter la survie dans la population. Une fois l'affinité entre une cellule B et un antigène a été déterminée, la cellule B concernée se transforme en une cellule plasma et subit à l'expansion clonale. Pendant le processus d'expansion clonale, les cellules B subissent une prolifération rapide (clonage) en proportion de la façon dont elles correspondent à l'antigène. Ces clones puis passent par la maturation d'affinité, où certains engagent une hypermutation somatique (inversement proportionnelle à l'affinité antigénique) et, finalement, vont passer par un processus de sélection par lequel une cellule donnée peut devenir une cellule mémoire. Ces cellules mémoire sont conservées pour permettre une réponse plus rapide au même, ou similaire antigène qui doit devenir l'hôte réinfecté. Ce taux rapide de réponse est connu comme étant la réponse immunitaire secondaire. Dans l'algorithme AIRS, l'idée de l'expansion clonale et la maturation d'affinité, sont utilisés pour favoriser la production de cellules mémoire potentielles. Ces cellules mémoires sont ensuite utilisées pour la classification.

Selon les travaux de Timmis [TIM 01], l'AIRS utilise l'idée de niveau de stimulation pour un ARB, qui, encore une fois, a été dérivé à partir des équations d'un réseau immunitaire décrit dans [FAR 86]. Bien qu'AIRS a été inspiré par ce travail sur les réseaux immunitaires, le développement du classifieur conduit à l'abandon des principes du réseau en faveur d'un modèle simple à base de population.

Cependant, AIRS ne prend pas en compte le principe de la mutation d'affinité proportionnelle. Lorsque de nouveaux ARBs ont été créés, ils ont été soumis à un processus de mutation aléatoire avec une certaine probabilité, et ont ensuite été incorporés dans l'ensemble des cellules mémoire lorsque leur affinité satisfait à certains critères.

Dans AIRS, les ARBs en compétition pour la survie reposent sur l'idée d'un système de ressources limitées. Un nombre prédéfini de ressources existait, pour lequel les ARBs ont rivalisé en fonction de leur niveau de stimulation : plus la valeur de stimulation d'un ARB est haute, plus le nombre de ressources allouées est grand. Les ARBs qui ne peuvent pas rivaliser avec succès sur les ressources, seront retirées du système. Le terme métadynamique du système immunitaire se réfère à l'évolution constante des cellules B à travers la prolifération cellulaire et la mort. Cela a été présenté dans AIRS avec la production continue et l'élimination des ARBs à partir de la population.

Le tableau si-dessous résume la correspondance entre un NIS et un AIRS.

#### B. L'algorithme

L'algorithme AIRS prend un antigène comme entrée et produit un ensemble de cellules mémoires comme sortie. Cet ensemble de cellules mémoires MC est considéré comme une base représentative qui est utilisée durant la phase de classification. Dans cet algorithme, les antigènes (Ag) présentent les données d'apprentissage et de test, et les ARBs (Artificial Recognition Balls) présentent l'ensemble des anticorps. Les étapes

Tableau 3.2 – Correspondance entre le NIS et l’AIRS.

NIS	AIRS
Anticorps	Vecteur de caractéristique
Antigènes	Donnée d’apprentissage
Expansion clonale	Reproduction des ARBs qui stimulent mieux avec des anti-gènes
Maturation d’affinité	Mutation aléatoire de l’ARB et le retrait des ARBs les moins stimulé
Mémoire immunitaire	L’ensemble mémoire des ARBs mutés
Métadynamiques	La production continue et l’enlèvement des ARBs et des cellules mémoire

principales impliquées dans cet algorithme sont [WAT 04] :

- La normalisation des données et l’initialisation.
- L’identification des cellules mémoires et la génération des ARBs.
- La compétition pour les ressources dans le développement d’une cellule mémoire candidate.
- La mise à jour de l’ensemble de cellules mémoires par l’introduction potentielle de la cellule mémoire candidate dans cet ensemble.

---

**Algorithme 3.4 :** Pseudo code de l’algorithme du AIRS.

---

**Données :**  $AG$  = ensemble d’antigènes, représentant les éléments de données à reconnaître.

**Résultat :**  $MC$  = ensemble de cellules mémoires capable de classifier des nouveaux éléments de données.

Générer aléatoirement un ensemble de cellules mémoires  $ARB$ .

**pour** tout antigène  $Ag \in AG$  **faire**

    -Calculer l’affinité de toute cellule mémoire  $Ab \in ARB$  avec  $Ag$ .

    -Sélectionner  $Ab$  qui a la plus grande affinité puis la cloner, placer les clones dans  $C$ .

**pour** tout clone d’anticorps  $c \in C$  **faire**

        | Muter  $c$  à un taux proportionnel de mutation.

**fin**

    -Copier tous les  $c \in C$  dans  $ARB$ .

    -Déterminer l’affinité de  $c$  avec  $Ag$ .

**si** critère  $AT$  **alors**

        | remplacer  $Ab$  (qui a la plus grande affinité) par  $c$  (qui a la plus grande affinité) dans  $M$ .

**sinon**

        | Copier  $c$  (qui a la plus grande affinité) dans  $MC$ .

**fin**

**fin**

---

### 3.3.3.5 La théorie de danger DT

La théorie du danger est une alternative à la sélection négative. Cette théorie stipule que les cellules présentatrices d’antigène APC du système immunitaire inné sont elles-mêmes activées via une alarme : les signaux de danger. Cette théorie est suggérée par

Matzinger en 1992 [MAT 02]. Dans cette approche, le système immunitaire, au lieu de réagir directement aux entités non-soi, il réagirait plutôt à des cellules du soi attaquées, ces cellules enverraient alors des signaux d'alarme ou de danger. Matzinger donc a caractérisé la théorie du danger comme un moyen de distinguer entre certaines cellules du soi et d'autres du non-soi, ce qui pourrait expliquer pourquoi le système ne réagit pas aux entités non-soi non dangereuses, et réagirait aux cellules de soi dangereuses. Ceci est la base de la théorie du danger.

L'immunologie classique stipule qu'une réponse immunitaire est déclenchée lorsque le corps rencontre quelque chose de non-soi ou d'étranger. Concernant la théorie du danger, et de plus la distinction soi/non-soi, il doit y avoir une autre discrimination qui se produisait et qui ne concordait plus avec les observations expérimentales. Par exemple :

- Le système naturel ne réagit pas à certaines bactéries étrangères dans nos intestins ou à la nourriture que nous mangeons, bien que tous les deux sont des entités étrangères (agents non-soi).
- À l'inverse, certains procédés auto réactifs sont utiles, par exemple contre des molécules de soi exprimées par des cellules stressées.
- La définition de soi est une problématique réaliste, le soi est confiné au sous-ensemble réellement vu par les lymphocytes lors de la maturation.
- Le corps humain change au cours de sa durée de vie, et ainsi le soi change aussi. Par conséquent, la question se pose de savoir si les défenses apprises tôt dans la vie contre le non-soi pourraient être auto-réactives plus tard.

La théorie de danger prend soin de « non-soi, mais non dangereux » et de « soi, mais dangereux » envahisseurs dans le système naturel. L'idée centrale est que le système immunitaire ne répond pas aux non-soi, mais au danger : d'une part, il réagit contre le non-soi, sauf si ce dernier n'est pas dangereux, et d'autre part, il ne doit pas réagir contre le soi, sauf si ce dernier est dangereux. Pratiquement, il n'y a pas besoin d'attaquer tout ce qui est étranger, ce qui semble être soutenu par les contre-exemples ci-dessus.

Comme le montre la figure 3.10, lorsqu'une cellule est en détresse, elle envoie un signal d'alarme qui se propage dans les alentours de la cellule. Ce signal établit une « zone de danger » autour de la cellule attaquée. Les antigènes dans le voisinage sont capturés par les APCs comme les macrophages (réponse innée) puis présentés aux lymphocytes. Ainsi, les cellules B, qui se lient avec les antigènes à l'intérieur de la zone de danger, sont activées, stimulées et subissent le processus d'expansion clonale. Celles qui ne se sont pas liées avec les antigènes ou bien qui sont trop loin de la zone de danger ne sont pas stimulées.

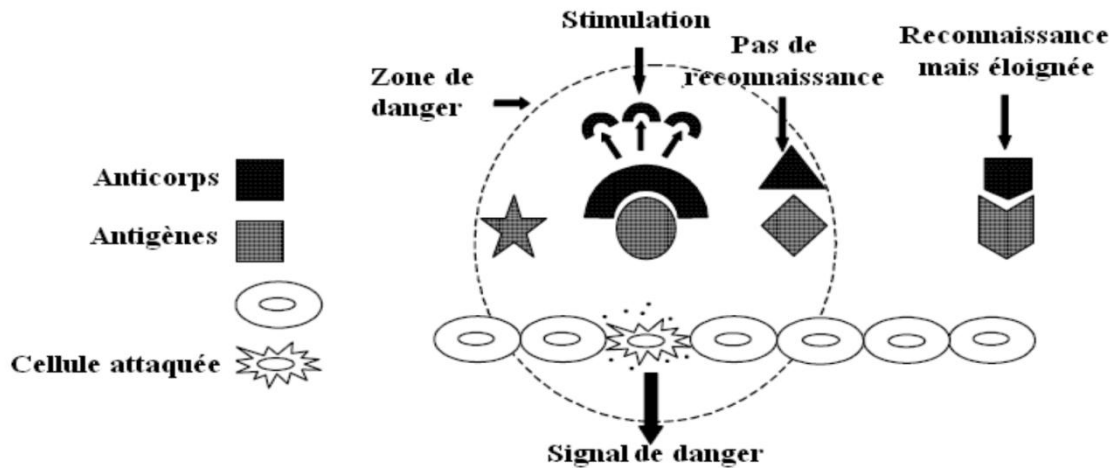


FIGURE 3.10 – Principe de théorie du danger [NAN 09]

Dans cette théorie, le danger est mesuré en fonction de la gravité des dégâts sur les cellules indiquées par l’envoi de signaux de détresse lors d’une mort inhabituelle. La nature même de ces signaux n’est pas encore claire. Ils peuvent être positifs, indiquant la présence d’un événement ou d’un état, comme pour la sécrétion de protéines lors d’un choc thermique. Ils peuvent être aussi négatifs, comme lors d’une présence de microbes dans l’organisme.

L’inconvénient majeur de cette théorie est que le système doit attendre que le soi soit endommagé avant de pouvoir activer la protection, parce qu’il nécessite des exemples d’états dangereux. Ceci n’est pas le cas de la détection négative qui se contente des exemples d’états sains du système. C’est à ce niveau-là que cette théorie va à l’encontre de certaines notions de la sélection du soi et non-soi traditionnelle.

### 3.4 Pourquoi utiliser les systèmes immunitaires artificiels pour la reconnaissance de formes

Le système immunitaire artificiel fournit un ensemble de modèles qui, en totalité, réussiraient à reproduire le comportement du système naturel. Dit que le système naturel a des propriétés très intéressantes pour la recherche en informatique, illustrées dans la section 3.2.4 de ce chapitre, l’AIS exploite ces propriétés afin de résoudre le problème traité. Dans la problématique de la RdF, l’AIS aurait les avantages suivants [HAB 12] :

- **L’adaptabilité** . Le système adapte ses ressources selon la complexité de la forme à reconnaître. si la forme rencontrée est connue (ressemble aux formes précédemment rencontrées), elle est traitée rapidement par la cellule mémoire correspondante. Si la forme rencontrée est moins comparable aux formes précédemment rencontrées, leur traitement augmente, c’est-à-dire une augmentation de la concentration de l’antigène correspondant, de nombre de cellules activées, et de temps de traitement.
- **L’unicité** . Le système utilise un mécanisme unique pour la reconnaissance rapide des formes déjà rencontrées (classification) et pour la mémorisation des formes inconnues (apprentissage). En outre, le système est en apprentissage continu, chaque forme inconnue lance directement le processus de mémorisation.

- Le système est en constante adaptation avec son environnement . Les formes qui sont rencontrées dans l'environnement sont mémorisées, et celles qui ne le sont pas (ou ne le sont plus) sont tout simplement oubliées. De cette manière, même si le système est placé dans un environnement avec une mémoire initiale très générale, avec le temps sa mémoire va devenir spécifique à son environnement de travail et donc, plus performante.
- Les singularités . Les formes mémorisées peuvent être « oubliées » si elles ne sont pas rencontrées pendant longtemps.
- La vaccination du système permet de lui faire apprendre rapidement de nouvelles formes, ou bien de restaurer une partie de sa mémoire.
- Le système fait de la sélection d'attributs de façon automatique, un anticorps n'a pas besoin de se lier avec toute la surface de l'agent infectieux, mais seulement avec la partie qui le rend spécifique.

Pour que ne se limite pas à cette liste, on peut certainement trouver d'autres propriétés très intéressantes avec une recherche plus poussée dans les mécanismes de fonctionnement du systèmes immunitaires naturels.

## 3.5 Comparaison entre les différents systèmes bio inspirés

Le tableau suivant récapitule une comparaison entre les différents systèmes inspirés de la biologie dont le but est de mettre en évidence leurs similitudes ainsi que leurs différences [DAS 99]. Ces systèmes sont : les systèmes immunitaires artificiels qui sont inspirés du système immunitaire humain, les réseaux de neurones qui sont inspirés du fonctionnement du cerveau et les algorithmes évolutionnaires inspirés par la théorie de l'évolution darwinienne [DEC 03 ; DEC 00].

Les approches évolutionnaires AEs partagent de nombreux concepts avec les AISs. La population, l'organisation du phénotype et du génotype ainsi que la prolifération des éléments les plus adaptés, sont présentes dans plusieurs approches des AISs. D'autre part, les modèles des systèmes immunitaires artificiels basés sur les réseaux immunitaires ressemblent par leurs structures ainsi que par les différentes interactions aux modèles connexionnistes. Dans [DAS 99 ; DEC 02a ; GRE 10], une comparaison entre les AISs et les RNAs a été soulignée.

Le domaine des systèmes immunitaires artificiels est un sujet de recherche très riche. Une comparaison est valable néanmoins pour montrer exactement les points de correspondance avec d'autres approches naturelles. Les AISs se distinguent des autres modèles inspirés de la biologie, comme les réseaux de neurones RNAs et les algorithmes génétiques AGs par :

- **La structure.** Les AISs peuvent être utilisés sous forme d'un réseau d'éléments interconnectés ou sous forme d'un ensemble d'éléments travaillant ensemble sans notion de communication, contrairement au RNAs qui ne fonctionnent qu'en étant interconnectés et aux AGs fonctionnant par population d'individu.
- **L'adaptation.** Les AISs réagissent à des situations non vécues et ceci grâce à une mutation de ses éléments, ce qui n'est pas réalisable avec les RNAs.

Tableau 3.3 – comparaison entre les AISs, les AGs et les RNAs.

Caractéristiques \ Systèmes	AIS	RNA	AG
<b>Composants</b>	Chaîne d'attribut	Neurones artificiels	Chaînes de chromosomes
<b>Endroits des composants</b>	Endroits dynamiques	Endroits prédéfinis/dynamiques	Endroits dynamiques
<b>Structure</b>	Ensemble d'éléments discrets ou gérés en réseau	Neurones gérés en réseau	Élément discret
<b>Stockage de la connaissance</b>	Chaînes d'attributs/connexion réseau	Poids de connexion	Chaînes chromosomiques
<b>Dynamique</b>	Apprentissage / Évolution	Apprentissage	Évolution
<b>Méta dynamique</b>	Élimination / Recrutement des composants	Algorithme constructif	Élimination / Recrutement des composants
<b>Interactions avec d'autres composants</b>	Par l'identification des chaînes d'attribut ou des connexions réseau	Par des connexions réseau	Par des opérateurs de recombinaison et/ou la fonction d'évaluation
<b>Interaction avec l'environnement</b>	Identification d'un modèle en entrée ou d'une évaluation d'une fonction objective	Les unités d'entrée reçoivent les stimuli environnementaux	Évaluation d'une fonction objective
<b>Seuil</b>	Influence l'affinité des éléments	Influence l'activation de neurones	Influence les variations génétiques
<b>Robustesse</b>	Population / Réseaux d'individus	Réseau d'individu	Population d'individu
<b>État</b>	Concentration et affinité	Niveau d'activation des neurones de sortie	L'information génétique dans les chromosomes
<b>Contrôle</b>	Principe, théorie ou processus immunitaire	Algorithme d'apprentissage	Algorithme évolutionnaire
<b>Possibilités de généralisation</b>	Réaction croisée	Extrapolation du réseau	Détection des schémas communs
<b>Non-linéarité</b>	Fonction d'activation par attachement	Fonction d'activation neuronale	Non explicite

- **L'interaction avec l'environnement.** Les AISs sont capables de reconnaître des modèles fournis comme pour les RNAs ou évaluer des fonctions objectives dont il faut atteindre, de la même manière que pour les AGs.
- **Un Modèle.** Les AISs, comme le système immunitaire biologique, évoluent dans



le temps et ils apprennent de leurs expériences antérieures, contrairement aux AGs qui ne font qu'évoluer et aux RNAs qui n'ont que la faculté d'apprentissage.

Le domaine des systèmes immunitaires artificiels regroupe donc quelques parts des réseaux de neurones et d'autres des algorithmes génétiques, ce qui le rend plus flexible à l'utilisation et applicable à différents domaines technologiques.

## 3.6 Conclusion

Les systèmes immunitaires sont un exemple de systèmes naturels qui ont servi de source d'inspiration pour résoudre une grande variété de problèmes informatiques. Dans le cadre de la RdF, et plus précisément la reconnaissance d'écriture manuscrite, les NISs présentent plusieurs propriétés très intéressantes comme la distinction entre le soi et le non-soi, la détection de changement, la mémorisation, l'adaptabilité ainsi que la gestion des ressources.

Les AISs se divisent en cinq grandes familles : la sélection négative, qui s'inspire des mécanismes naturels de distinction entre soi et non-soi, pour résoudre des problèmes de surveillance et de détection de changement. La sélection clonale artificielle, qui s'inspire des mécanismes naturels de mémorisation pour résoudre des problèmes d'optimisation. Les réseaux immunitaires artificiels, qui s'inspirent de la théorie des réseaux immunitaires pour construire des systèmes permettant la distinction et la mémorisation. Ces derniers ont surtout été utilisés pour l'analyse de données. La théorie de danger qui s'inspire de la théorie de sélection négative pour la détection et l'élimination de composante de soi qui fait de danger. Cette méthode peut être utilisée dans la détection des pannes de détecteur de système lui-même. L'AIRS qui s'inspire de la théorie des réseaux immunitaires et de sélection clonale pour l'adaptation, la distinction et la mémorisation. Cet algorithme a été utilisé pour résoudre des problèmes d'optimisation, de reconnaissance de formes et de classification.

Pour notre étude sur la reconnaissance d'écriture manuscrite on s'intéresse à la reconnaissance de caractères, alors on a choisi le modèle AIRS pour l'apprentissage et l'optimisation de décision dans le système proposé.

# L'approche qualitative

## 4.1 Introduction

La représentation des connaissances est généralement associée à un raisonnement, qui est formalisé par une certaine logique. Depuis plusieurs décennies, la logique classique « logique du vrai ou du faux » est utilisée dans une variété d'applications en intelligence artificielle et reconnaissance des formes. Cependant, elle reste inadéquate à la représentation des concepts linguistiques tels que : « elle parle beaucoup, mais agit peu », « il habite très loin de chez moi ». Elle est donc incapable, dans la plupart des cas, à représenter exactement ce qui est approximativement vrai ou faux. Pour surmonter cette imperfection, les techniques de représentation des connaissances approximatives tentent d'automatiser cette démarche, soit par des approches numériques ou bien symboliques.

L'approche que nous présenterons dans ce chapitre consiste à élaborer de nouveaux outils qui permettent de représenter des connaissances incertaines afin de mieux simuler le raisonnement humain. On s'intéresse plus particulièrement à la formulation des données sous forme d'expressions linguistiques pondérées par des mots, plutôt que par des nombres. De ce fait, une approche purement qualitative, formalisée à l'aide d'un ensemble d'outils « logico-symboliques » pour le traitement du raisonnement plausible (incertain, approximatif, inexact...), sera présentée par la suite de chapitre.

Comme l'objectif de notre travail est lié aux systèmes de reconnaissance de caractères manuscrits et, plus précisément, à la phase de classification de ces systèmes, l'approche qualitative que nous allons présenter dans ce chapitre, et que nous avons appliquée dans la suite de notre travail se situe dans le même cadre d'approche de Séridi et al [SER 06]. Elle peut être vue comme étant une extension du système SUCRAGE [BOR 99]. Pour cela, nous avons essayé d'exploiter leur capacité de décision dans des situations imprécises et incertaines afin d'obtenir un meilleur taux de reconnaissance.

Contrairement aux systèmes classiques basés sur une appréciation des valeurs numériques, l'approche qualitative fournit plus de compréhension pour l'expert et pour les utilisateurs. Dans [SER 06 ; BEL 10], la comparaison entre cette approche et les systèmes numériques a montré que leurs performances étaient meilleures que d'autres systèmes.

Dans la suite de ce chapitre, nous donnerons une description simplifiée de l'approche « qualitative » ou bien « symbolique », notée comme « système à base des règles qualitative QRBS », en mettant l'accent sur le processus de décision.

## 4.2 Architecture

Tandis que l'approche qualitative QRBS est basée sur la construction et l'exploitation des règles, elle est construite sur la base d'un système informatique qui met en œuvre la méthode de construction de règles GARP (Générateur Automatique de Règles de Production) et qui intègre aussi bien le système d'inférence. L'architecture du QRBS est principalement composée de deux sous-systèmes (figure 4.1) : un générateur automatique de règles et un système d'inférence. Le premier sous-système génère des règles de classification de l'ensemble de traitement (apprentissage), de la forme : « SI *antécédent* ALORS *conséquence* ». La prémisse d'une règle est représentée par les attributs des objets à classer, et la conséquence fournit un degré de croyance qualitative, d'appartenance à une classe, associée à une valeur linguistique. Le deuxième sous-système consiste à la classification des nouveaux objets de l'ensemble de tests. Une architecture simplifiée du QRBS est représentée par le schéma de la figure 4.1.

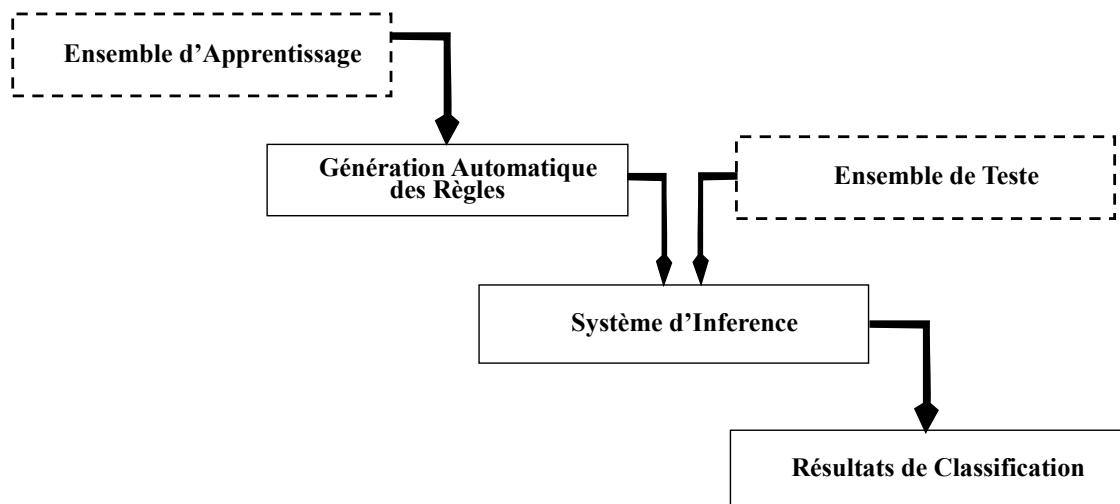


FIGURE 4.1 – Architecture simplifiée du QRBS [SER 01].

Les principales caractéristiques du QRBS sont les suivantes :

- La construction des règles est fondée sur la recherche de corrélation toutes classes confondues.
- Le mode discrétisation est régulier.
- L'inférence exacte est utilisée pour l'exploitation de ces règles.
- Le degré de croyance affecté aux règles est une probabilité numérique.
- La conorme est probabiliste.

Les processus de génération de règles et de classification des nouveaux objets seront décrits brièvement comme suit :

### 4.2.1 Génération automatique des règles

La phase d'apprentissage consiste en une génération automatique de règles à prémisses multivalentes de la forme : « SI *antécédent/prémisse* ALORS *conséquence/ conclusion* ». Une telle règle est notée : prémisse → conclusion.

La morphologie des règles générées par ce sous-système est :

$$A_1 \text{ et } A_2 \text{ et } \dots \text{ et } A_n \rightarrow y, \alpha$$

où :  $A_i$  est une condition du type : «  $X_j$  est dans  $[a, b]$  ».  $X_j$  est la  $j^{\text{ème}}$  composante du vecteur représentant un objet.  $[a, b]$  est un intervalle issu de la discrétisation des domaines de variation des attributs (ici de l'attribut  $X_j$ ).  $y$  est une hypothèse sur l'appartenance à une classe et  $\alpha$  est un degré de croyance représentant l'incertitude de la conclusion.

Les différentes phases qui consistent à générer la base de règles sont représentées par le schéma de la figure 4.2, et le principe de fonctionnement du générateur automatique de règles de production sera présenté dans les sections suivantes.

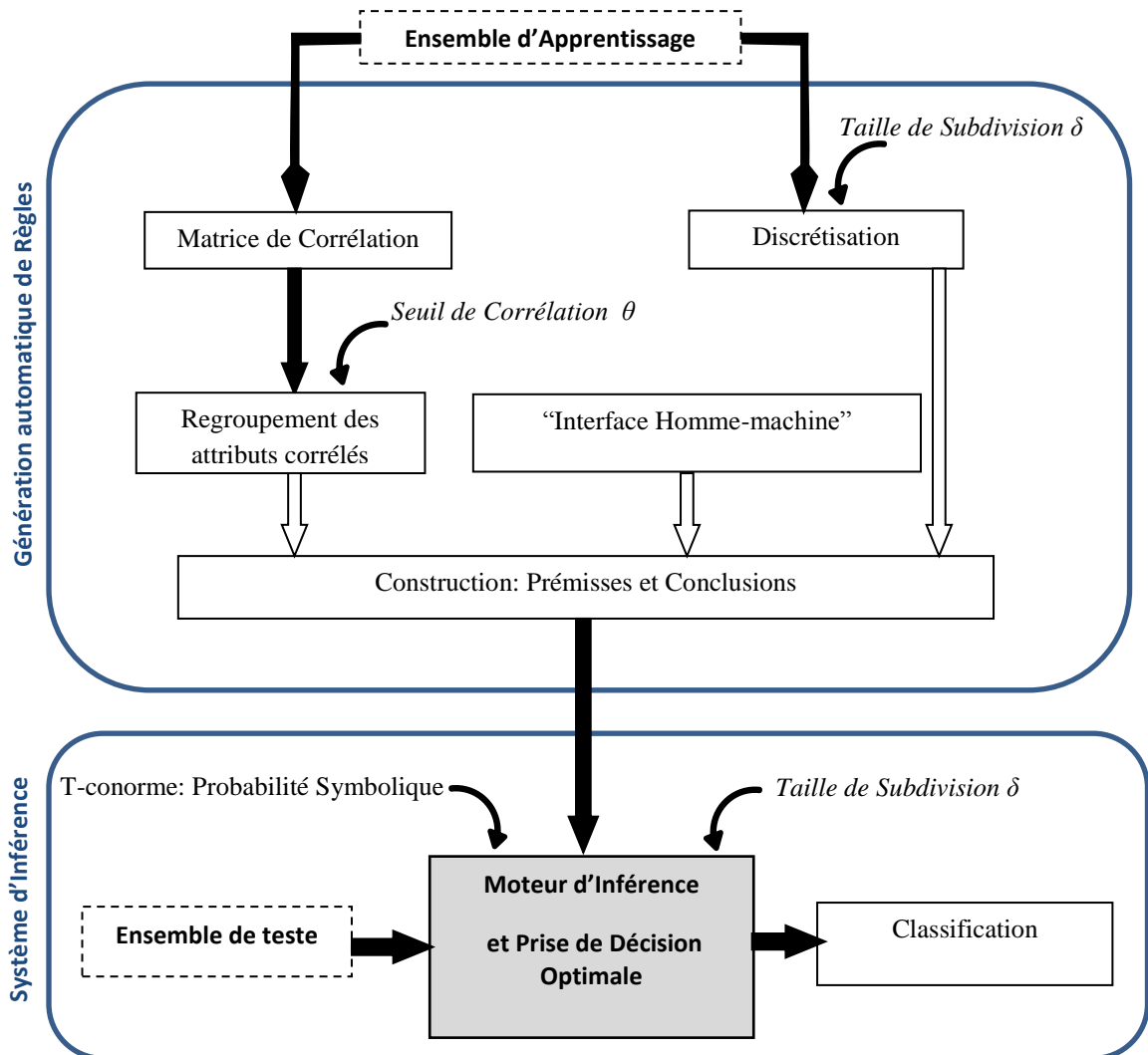


FIGURE 4.2 – Les principales étapes d'un QRBS

#### 4.2.1.1 Construction des prémisses

Dans un premier temps, un traitement préalable de la base de données s'impose pour construire l'ensemble (la base) d'apprentissage. Ce dernier est constitué d'un ensemble d'exemples. Chaque exemple est représenté par un vecteur numérique  $V_i$  de dimension  $p$  et étiqueté par une classe  $y_i$  à laquelle il appartient. La représentation d'un exemple de l'ensemble d'apprentissage est la suivante :

$$V_i = \{X_1^i, X_2^i, \dots, X_p^i\} \text{ est de classe } y_i$$

Une fois que l'ensemble d'apprentissage est mis sous ce format, il subit des traitements pour construire la base de règles afin de classer ultérieurement de nouveaux objets non étiquetés par le nom de la classe à laquelle ils appartiennent. La construction des prémisses à partir de cet ensemble est basée sur deux procédés principaux : la corrélation et la discrétisation (figure 4.3).

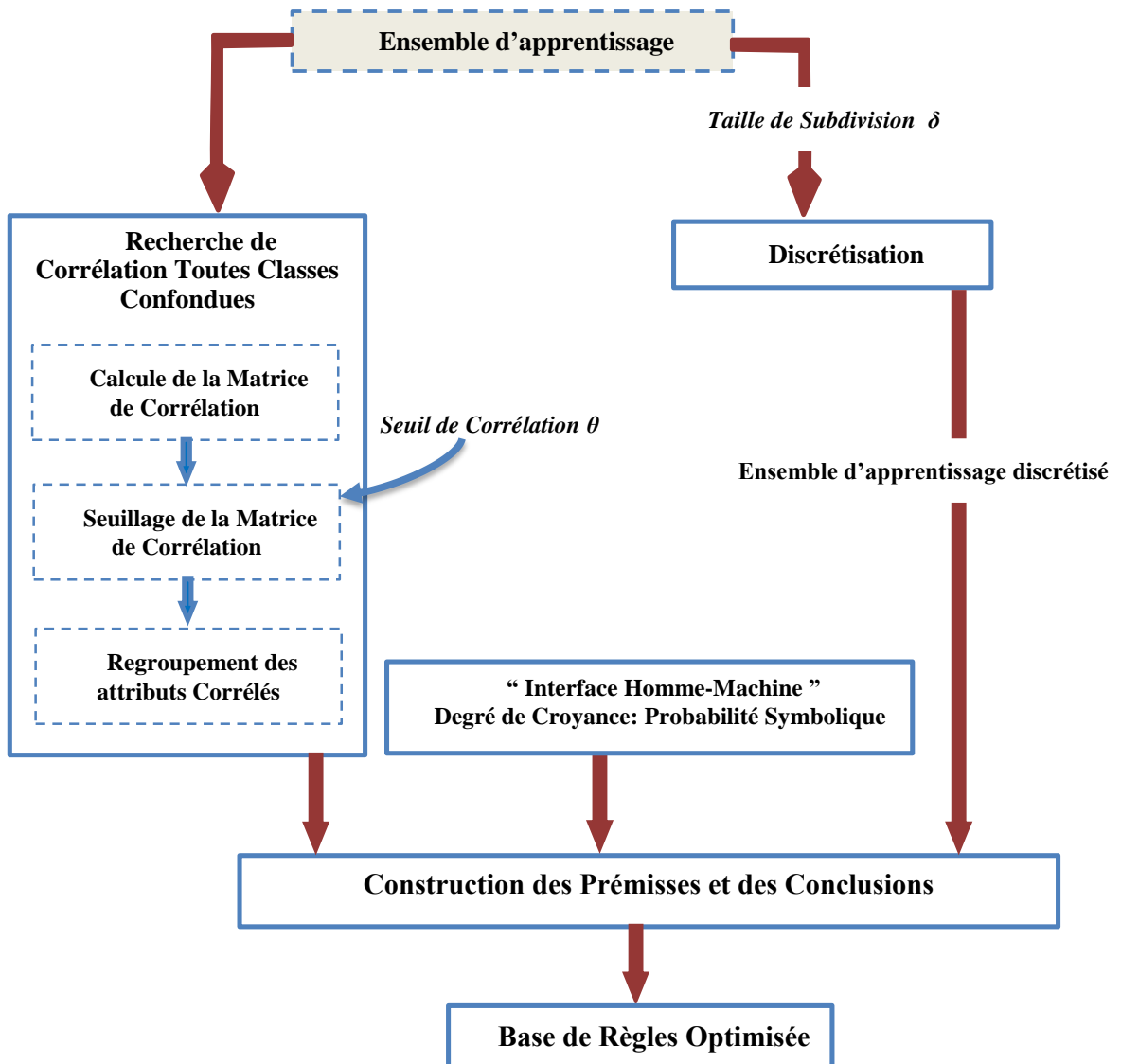


FIGURE 4.3 – La construction des règles [SER 01]

### A. Corrélation

Le processus de corrélation consiste à rechercher la corrélation linéaire entre les composantes de vecteur de tous de l'ensemble d'apprentissage, ceci est réalisé en trois étapes :

- La première étape consiste à calculer la matrice de corrélation  $R_{p \times p}$  entre les composantes des vecteurs de l'ensemble d'apprentissage.
- La deuxième étape consiste à déduire la matrice de corrélation  $R_{\theta}$  une matrice seuillée, en fixant un seuil ( $\theta$ ) à partir duquel deux attributs sont corrélés. Chaque élément de la matrice  $R_{p \times p}$  est comparé à  $\theta$ , si sa valeur est inférieure à  $\theta$ , alors

l'élément qui lui correspond dans  $R_\theta$  est pris égal à zéro. Sinon s'il est supérieur ou égal à  $\theta$ , alors il est pris égal à 1.

- La troisième étape consiste à extraire les composantes connexes maximales pour le regroupement des attributs corrélés.

### B. Discrétisation

Le processus de discrétisation consiste à construire une partition régulière de l'ensemble d'apprentissage. La discrétisation consiste à diviser le domaine de variation de chaque attribut de l'ensemble d'apprentissage en  $\delta$  parties égales de taille "L" chacune. Si le domaine de variation de l'attribut est  $[a_{min}, a_{max}]$ , la taille sera  $L = (a_{i,max} - a_{i,min})/\delta$ , et les sous-intervalles obtenus après subdivision de l'attribut sont totalement ordonnés et désignés par  $rk_0, rk_1, \dots, rk_{(\delta-1)}$ , tels que :  $rk_0 = [a_{i,min}, a_{i,1}[$ ,  $rk_1 = [a_{i,1}, a_{i,2}[$ ,  $\dots$ ,  $rk_{(\delta-1)} = [a_{i,\delta-1}, a_{i,max}]$ .

Chaque attribut prend sa valeur à partir de l'ensemble totalement ordonné  $\{rk_0, rk_1, \dots, rk_{(\delta-1)}\}$  [BOR 98], avec  $\delta$  comme la taille de la subdivision du domaine de variation  $R_i = [a_{i,max}, a_{i,min}]$  de l'attribut  $X_i$ , avec  $i = 1, \dots, p$ , et avec la taille de l'amplitude de chaque sous-intervalle  $rk_i$  égal à  $L = (a_{i,max} - a_{i,min})/\delta$ .

Ces opérations sont réalisées sur l'ensemble de la formation sans tenir compte des classes.

Les résultats émanant des deux traitements de l'ensemble d'apprentissage serviront à la construction des règles, c'est à dire la construction des prémisses associées aux conclusions. En effet, dès qu'on dispose des composantes connexes, on regroupe les attributs corrélés pour chaque composante connexe dans les mêmes prémisses. Chaque prémisses construite génère "C" règles, où "C" est le nombre total de classes. Pour chaque prémisses, toutes les conclusions possibles d'appartenance à une classe sont générées. Donc, une règle est l'association d'une prémisses à une classe  $y_j$  de  $y_1, y_2, \dots, y_C$  avec un certain degré de croyance, c'est-à-dire une certaine incertitude. Chaque prémisses construite génère un nombre maximal de règles :

$$NR_{max} = C * \sum_{i=1}^{n_{cc}} \delta^{k_i} \quad (4.1)$$

Où  $n_{cc}$  est le nombre des composantes connexes.  $C$  est le nombre de classes, et  $k_i$  est le cardinal de la  $i^{me}$  composante connexe.

Une règle est une association d'une prémisses à une classe  $y_j$  de  $y_1, y_2, \dots, y_C$  avec un certain degré de croyance  $\alpha$ . Les degrés de croyance sont des probabilités numériques estimés à partir d'une appréciation objective fondée sur des hypothèses d'appartenance des prémisses à une classe. La probabilité d'appartenance d'un objet donné  $X$  à une classe  $y_j$  est :

$$P\left(\frac{y_j}{X}\right) \approx \frac{n_i}{\sum_{j=1}^c n_j} \quad (4.2)$$

Où  $n_i$  est le nombre d'exemples de la classe  $y_j$  dans l'ensemble d'apprentissage.  $X$  est le vecteur représentatif de l'exemple.

Dans cette approche, le problème d'indépendance ne se pose pas et cette probabilité est estimée directement lors de la phase d'apprentissage. Si une prémisses ne contient aucun exemple d'apprentissage, alors la règle correspondante ne sera pas générée. La méthode utilisée pour optimiser le nombre de règles générées est celle qui consiste à éliminer les règles ayant un degré de croyance nul ou indéterminé. Ces règles n'ont aucune

influence pendant la phase de la classification.

Enfin, le générateur fournit des règles de la forme suivante :

Si  $X_1$  et  $X_2, \dots$  et  $X_p$ , alors  $X_i = \{X_1, X_2, \dots, X_p\}$  appartient à  $y_j$  avec  $\alpha_i$ ,  $j = 1, \dots, c$

Où :  $X_i$  est une valeur discrète.  $y_j$  est une hypothèse sur la classe d'appartenance.  $\alpha_i$  est le degré de croyance représentant l'incertitude de la conclusion.

**Exemple :** Soit un ensemble d'apprentissage qui est représenté par 4 attributs  $\{X_1, X_2, X_3, X_4\}$ . Le seuil de corrélation  $\theta$  est égal à 0.95. La matrice seuillée  $R_\theta$  est de la forme suivante :

$$R_\theta = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

En regroupant les composantes corrélées, on obtient les composantes connexes maximales suivantes  $\{X_1\}$ ,  $\{X_2\}$  et  $\{X_3, X_4\}$ .

Soit une composante connexe de l'exemple précédent  $\{X_3, X_4\}$ . Leur cardinal est égal à 2. Si la taille de subdivision est  $\delta = 3$ , alors le nombre de prémisses générées est de  $3^2$ .

Si nous supposons qu'il y a trois classes  $\{y_1, y_2, y_3\}$ . Une prémisses possible générée pour la composante  $\{X_1\}$  est :  $X_1$  est dans  $rk_0$ . Cette prémisses conduit à 3 règles qui sont :

- **Règle 1** : Si  $X_1$  est dans  $rk_0$  Alors  $y_1$  avec le degré  $\alpha_1$ .
- **Règle 2** : Si  $X_1$  est dans  $rk_0$  Alors  $y_2$  avec le degré  $\alpha_2$ .
- **Règle 3** : Si  $X_1$  est dans  $rk_0$  Alors  $y_3$  avec le degré  $\alpha_3$ .

Le nombre maximal de règles générées par les composantes connexes  $\{X_1\}$ ,  $\{X_2\}$  et  $\{X_3, X_4\}$  vaut :  $3 * (3^1 + 3^1 + 3^2) = 45$ .

Si après discrétisation, le nombre d'exemples  $n_i$  ( $i = 1, 2, 3$ ) correspondant à la composante connexe  $X_1$  de l'ensemble d'apprentissage de la classe  $y_j$  vaut respectivement :  $n_1 = 42$ ,  $n_2 = 9$  et  $n_3 = 0$ ; dans ce cas, la probabilité ( $y_j/prmisse$ ) est égale à :

$$\alpha_1 = P(y_1/X_1 \text{ est dans } rk_0) = 42/(42 + 9) = 0,823529.$$

$$\alpha_2 = P(y_2/X_1 \text{ est dans } rk_0) = 9/(42 + 9) = 0,176471.$$

$$\alpha_3 = P(y_3/X_1 \text{ est dans } rk_0) = 0/(42 + 9) = 0.$$

Par conséquent, les seules règles générées par le générateur seront les règles : 1 et 2.

#### 4.2.1.2 Interface homme-machine

Cette étape consiste à représenter et exploiter l'incertitude numérique par des degrés symboliques, qui peuvent être interprétés comme des probabilités qualitatives et comparatives.

Pour cette raison, une interface homme-machine est nécessaire pour traduire les probabilités numériques en probabilités symboliques. A cet effet, une échelle de gradualité symbolique  $L_M$ , composée de  $M$  valeurs,  $L_M = \{\tau_\alpha, \alpha = 0, \dots, M - 1\}$ , qui est totalement ordonné, est introduit dans le processus :

**Premièrement**, chaque degré symbolique  $\tau_\alpha$  est associé à une valeur linguistique. La taille de l'échelle de gradualité symbolique est définie par  $M$ .

Prenons  $M = 9$ , ce choix a pour fournir aux utilisateurs plus d'expressivité et de flexibilité et, par conséquent, une plus grande précision.

**Deuxièmement**, chaque degré  $\tau_\alpha$  est associée à une valeur de l'échelle des degrés de

probabilités symboliques  $dg(\tau_\alpha)$  de  $D_M = \{dg(\tau_\alpha) = \alpha, \alpha = 0, \dots, M - 1\}$ , avec  $dg(\tau_\alpha)$  est le degré de probabilité symbolique.

*La dernière expertise* consiste de partitionner l'espace des probabilités en sous-espace selon la croyance d'un expert.

Le tableau 4.1 montre la correspondance entre le degré symbolique, l'échelle des valeurs linguistiques, l'espace des probabilités et le degré de probabilité symbolique.

Tableau 4.1 – Correspondance entre le degré symbolique, l'échelle des valeurs linguistiques, l'espace des probabilités et le degré de probabilité symbolique

Degré symbolique $\Leftrightarrow$ Valeur linguistique	Degré symbolique $\Leftrightarrow$ Degré de probabilité symbolique	Espace des probabilités $\Leftrightarrow$ Degré de probabilité symbolique
$\tau_0 \Leftrightarrow$ Très Faible	$\tau_0 \Leftrightarrow dg(\tau_0) = 0$	Si $P = 0$ alors $dg(\tau_0) = 0$
$\tau_1 \Leftrightarrow$ Faible	$\tau_1 \Leftrightarrow dg(\tau_1) = 1$	Si $P \in ]0, 0.02]$ alors $dg(\tau_1) = 1$
$\tau_2 \Leftrightarrow$ Au-dessus du Faible	$\tau_2 \Leftrightarrow dg(\tau_2) = 2$	Si $P \in ]0.02, 0.05]$ alors $dg(\tau_2) = 2$
$\tau_3 \Leftrightarrow$ Au-dessous du Moyen	$\tau_3 \Leftrightarrow dg(\tau_3) = 3$	Si $P \in ]0.05, 0.10]$ alors $dg(\tau_3) = 3$
$\tau_4 \Leftrightarrow$ Moyen	$\tau_4 \Leftrightarrow dg(\tau_4) = 4$	Si $P \in ]0.10, 0.20]$ alors $dg(\tau_4) = 4$
$\tau_5 \Leftrightarrow$ Au-dessus du Moyen	$\tau_5 \Leftrightarrow dg(\tau_5) = 5$	Si $P \in ]0.20, 0.50]$ alors $dg(\tau_5) = 5$
$\tau_6 \Leftrightarrow$ Au-dessous de l'Élevé	$\tau_6 \Leftrightarrow dg(\tau_6) = 6$	Si $P \in ]0.50, 0.80]$ alors $dg(\tau_6) = 6$
$\tau_7 \Leftrightarrow$ Élevé	$\tau_7 \Leftrightarrow dg(\tau_7) = 7$	Si $P \in ]0.80, 0.95]$ alors $dg(\tau_7) = 7$
$\tau_8 \Leftrightarrow$ Très Élevé	$\tau_8 \Leftrightarrow dg(\tau_8) = 8$	Si $P \in ]0.95, 1]$ alors $dg(\tau_8) = 8$

## 4.2.2 Système d'inférence

Le système d'inférence consiste à classer de nouveaux objets n'appartient pas à l'ensemble d'apprentissage (figure 4.4). Ce système reçoit à l'entrée la base de règles générées précédemment, ainsi qu'un vecteur de l'ensemble de test (non étiquetée par le nom de la classe à laquelle il appartient, et discrétisé), pour être classé. Une fois que ce vecteur est discrétisé, le système d'inférence associe à ce nouvel objet un degré de croyance symbolique définissant sa classe d'appartenance.

La méthode de raisonnement utilisée par le système d'inférence est exacte. Les règles déclenchées sont celles dont les prémisses sont vérifiées exactement par le nouveau vecteur à classer. L'inférence exacte mise en œuvre par le moteur d'inférence est le Modus Ponens classique (l'inférence par déduction). Si l'on connaît la relation d'implication entre une prémisse et une conclusion ( $P \rightarrow C$ ) et si la prémisse  $P$  est vraie, alors le Modus Ponens permet de déduire la conclusion  $C$ . Une fois que les règles sont déclenchées et regroupées par classe, on associe à chaque classe un degré de croyance symbolique final qui est évaluée par la  $T$ -conorme probabiliste symbolique [SER 01] :  $T(p, q) = \min(p + q, M)$ , et  $p$  et  $q$  sont les degrés de probabilités symboliques. La  $T$ -conorme probabiliste symbolique généralisée est définie par [SER 01] :

$$S_j = \min \left( \sum_{i=1}^p P_{i,j} M \right) \quad (4.3)$$

Où :  $P_{i,j}$  est le degré de probabilité symbolique d'appartenance de l'attribut  $X_i$  à la classe  $y_j$ .



Enfin, la classe attribuée au nouvel objet est celle-ci qui a le degré de probabilité symbolique le plus haut.

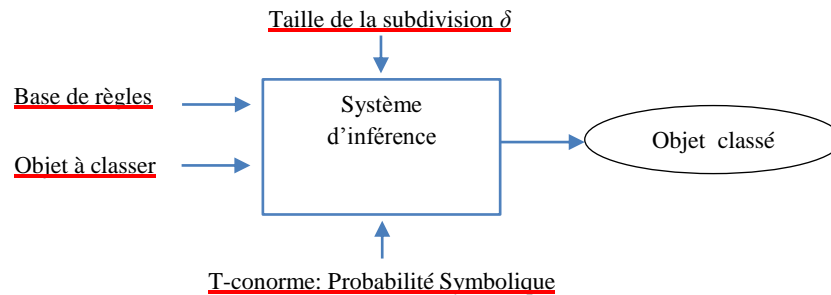


FIGURE 4.4 – Système d'inférence [SER 01]

### 4.2.3 Prise de décision optimale

Afin d'éviter le problème de saturation de la base de connaissances qui affecte la précision du taux de classification (par exemple, la plus grande valeur que peut prendre la  $T$ -conorme probabiliste symbolique sous différentes hypothèses ne peut dépasser la valeur maximale de l'échelle de croyance proposé, dans notre cas il s'agit de la valeur  $M = 9$ ), une fonction de pénalité est introduit pour assouplir ou bien corriger cette contrainte. Son rôle principal est d'étendre l'échelle de croyance en pondérant le degré de probabilité symbolique résultant de la  $T$ -conorme probabiliste symbolique à chaque fois, lorsque le résultat dépasse la valeur maximale de l'échelle de croyance, qui est égale à neuf. Le choix d'appartenance de ce degré à une classe se fait par la fonction de pénalité. Cette dernière est définie par :

$$F_j = \max \left( \sum_{i=1}^p P_{i,j} - (M - 1), 0 \right) \quad (4.4)$$

où :  $P_{i,j}$  est le degré de probabilité symbolique d'appartenance de l'attribut  $X_i$  à la classe  $y_j$ .

La fonction de pénalité normalisée est définie comme suit :

$$Fn_j = (F_j * 100) / (M - 1) * (Card(X) - 1) \quad (4.5)$$

où  $Card(X)$  est le cardinal du vecteur  $X$ .

La principale règle de décision qui attribue une classe à un vecteur est la suivante :

**Si**  $S_j = S_{\max}$  **et**  $Fn_j = Fn_{\max}$ , **alors**  $X \in y_j$   
 tel que  $S_{\max} = \max\{S_j, j = 1, \dots, C\}$ ,  $Fn_{\max} = \max\{Fn_j, j = 1, \dots, C\}$  et  $y = \{y_j, j = 1, \dots, C\}$ .

On peut résumer les règles de décision optimale dans les cas suivants :

- Si le maximum de l'échelle des degrés de probabilités symboliques est atteint par plusieurs classes, le choix d'une classe parmi celles-ci est celle qui a un degré de probabilité symbolique et une fonction de pénalité maximale.
- Dans le cas où le maximum de l'échelle des degrés de probabilités symboliques est atteint au moins par deux classes et leurs fonctions de pénalité correspondantes sont égales, alors le choix se fait d'une manière aléatoire.

- Si le maximum de l'échelle des degrés de probabilités symboliques n'est atteint par aucune classe, alors on associe au vecteur la classe qui a le plus grand degré de probabilité symbolique.
- Si le maximum de l'échelle des degrés de probabilités symboliques n'est atteint par aucune classe et si au moins deux classes ont des degrés de probabilités symboliques égaux, alors le choix se fait d'une manière aléatoire.
- Si le maximum de l'échelle des degrés de probabilités symboliques est atteint par une seule classe, alors on associe au vecteur cette classe.

L'algorithme de prise de décision optimale qui illustre ces cas est décrit comme suit :

---

**Algorithme 4.1 :** Algorithme d'optimisation de prise de décision

---

```

 $S = \{S_j, j = 1, \dots, C\},$ 
 $S_{\max} = \max\{S_j, j = 1, \dots, C\},$ 
 $F_n = \{F_{n_j}, j = 1, \dots, C\},$ 
 $F_{n_{\max}} = \max\{F_{n_j}, j = 1, \dots, C\},$ 
 $Y = \{y_j, j = 1, \dots, C\},$ 
 $J = \{j \in \{1, \dots, C\} / S_j = S_{\max}\},$ 
 $K = \{k \in J / F_{n_k} = F_{n_{\max}}\},$ 
M est la valeur maximale de l'échelle des degrés,
X est un exemple de la base de test.
si ( $S_{\max} = M$ ) alors
    | si ( $Card(J) = 1$ ) et ( $Card(K) = 1$ ) alors
    | |  $X \in y_j, j \in J$ 
    | fin
    | sinon si ( $Card(J) > 1$ ) and ( $Card(K) = 1$ ) alors
    | |  $X \in y_k, k \in K$ 
    | fin
    | sinon si ( $Card(J) > 1$ ) et ( $Card(K) > 1$ ) alors
    | |  $X \in rand(\{y_k, k \in K\})$ ; // rand(R) retourne une valeur aléatoire de R
    | fin
sinon
    | si ( $Card(J)=1$ ) alors
    | |  $X \in y_j, j \in J$ 
    | fin
    | sinon si ( $Card(J)>1$ ) alors
    | |  $X \in rand(y_j, j \in J)$ 
    | fin
fin

```

---

**Exemple :** Soit un ensemble d'apprentissage avec un nombre d'attributs égal à  $p = 4$ , le nombre de classes maximal vaut 3 et les domaines de variation de chacune des composantes  $X_1, X_2, X_3, X_4$  sont respectivement :

[4.3, 7.9]

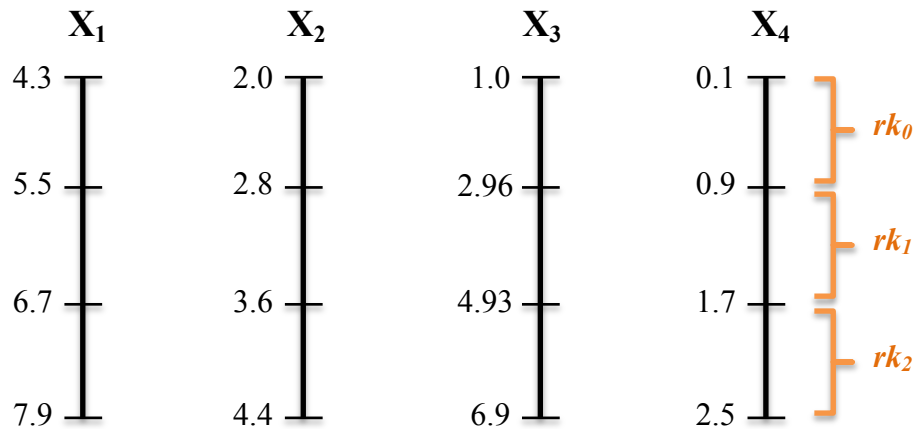
[2.0, 4.4]

[1.0, 6.9]

[0.1, 2.5]

#### 4. L'approche qualitative

Pour une taille de la subdivision  $\delta = 3$ , la discrétisation des attributs conduit aux intervalles suivants :



Soit le vecteur à classer  $V = \{5.4, 3.9, 1.7, 0.4\}$ . Son vecteur discrétisé conduit par la discrétisation est  $V_{discrtisé} = \{rk_0, rk_2, rk_0, rk_0\}$ .

Pour un seuil de corrélation de 0.95, la matrice seuillée est :

$$\begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{matrix}$$

Pour une taille de subdivision égale à 3, les règles déclenchées sont les suivantes :

- Règle 2 : Si  $X_1$  est dans  $rk_0$  Alors  $y_1$  avec le degré 0.832529
- Règle 1 : Si  $X_1$  est dans  $rk_0$  Alors  $y_2$  avec le degré 0.176471
- Règle 17 : Si  $X_2$  est dans  $rk_2$  Alors  $y_1$  avec le degré 0.857143
- Règle 15 : Si  $X_2$  est dans  $rk_2$  Alors  $y_3$  avec le degré 0.142857
- Règle 20 : Si  $X_3$  est dans  $rk_0$  et Si  $X_4$  est dans  $rk_0$  Alors  $y_1$  avec le degré 1.0.

Les règles déclenchées concluant à l'appartenance à la classe  $y_1, y_2, y_3$  conduisent aux ensembles suivants :

- A={règle 2, règle 17, règle 20}
- B={règle 1}
- C={règle 15}

Prenons la taille de l'échelle de gradualité symbolique  $M = 9$ . L'introduction de l'interface homme-machine traduit chaque mesure de probabilité numérique  $P$  (définie sur l'ensemble  $P(X)$ , ensemble des parties de  $X$  à valeurs dans  $[0, 1]$ ), à une valeur de l'échelle  $D_M$  des degrés de probabilités symboliques :

- Si  $P = 0$  alors  $dg(\tau_0) = 0$
- Si  $P \in ]0, 0.02]$  alors  $dg(\tau_1) = 1$
- Si  $P \in ]0.02, 0.05]$  alors  $dg(\tau_2) = 2$
- Si  $P \in ]0.05, 0.1]$  alors  $dg(\tau_3) = 3$
- Si  $P \in ]0.1, 0.2]$  alors  $dg(\tau_4) = 4$

Si  $P \in ]0.2, 0.5]$  alors  $dg(\tau_5) = 5$   
Si  $P \in ]0.5, 0.8]$  alors  $dg(\tau_6) = 6$   
Si  $P \in ]0.8, 0.95]$  alors  $dg(\tau_7) = 7$   
Si  $P \in ]0.95, 1]$  alors  $dg(\tau_8) = 8$

Après l'interfaçage homme-machine, les règles de l'exemple précédent prennent les formes suivantes :

Règle 2 : Si  $X_1$  est dans  $rk_0$  Alors  $y_1$  avec  $dg(\tau_7) = 7$   
Règle 1 : Si  $X_1$  est dans  $rk_0$  Alors  $y_2$  avec  $dg(\tau_4) = 4$   
Règle 17 : Si  $X_2$  est dans  $rk_2$  Alors  $y_1$  avec  $dg(\tau_7) = 7$   
Règle 15 : Si  $X_2$  est dans  $rk_2$  Alors  $y_3$  avec  $dg(\tau_4) = 4$   
Règle 20 : Si  $X_3$  est dans  $rk_0$  et Si  $X_4$  est dans  $rk_0$  Alors  $y_1$  avec  $dg(\tau_8) = 8$

En appliquant la  $T$ -conorme probabiliste symbolique généralisée et la fonction de pénalité normalisée on obtient :

$$S_1 = \min(7 + 7 + 8, 8) = 8$$
$$F_1 = \max(7 + 7 + 8 - (9 - 1), 0) = 14$$
$$Fn_1 = (14 * 100) / (9 - 1) * (4 - 1) = 58.33\%$$

D'où  $V$  appartient à  $y_1$  avec  $S_1 = 8$  et  $Fn_1 = 58.33\%$ .

$$S_2 = \min(4, 8) = 4$$
$$F_2 = \max(4 - (9 - 1), 0) = 0$$
$$Fn_2 = (0 * 100) / (9 - 1) * (4 - 1) = 0\%$$

D'où  $V$  appartient à  $y_2$  avec  $S_2 = 4$  et  $Fn_2 = 0\%$ .

$$S_3 = \min(4, 8) = 4$$
$$F_3 = \max(4 - (9 - 1), 0) = 0$$
$$Fn_3 = (0 * 100) / (9 - 1) * (4 - 1) = 0\%$$

D'où  $V$  appartient à  $y_3$  avec  $S_3 = 4$  et  $Fn_3 = 0\%$ .

En appliquant les règles de décisions, on déduit que :  $V$  appartient à la première classe ( $y_1$ ) avec un degré de probabilité symbolique égale à 8 et une fonction de pénalité égale à 58.33%.

### 4.3 Conclusion

Nous avons vu dans ce chapitre un modèle (QRBS) de traitement symbolique des connaissances. Dans les systèmes de reconnaissance de caractères manuscrits, ce modèle a un double objectif. Premièrement, il est utilisé pour augmenter le taux de reconnaissance des caractères manuscrits. Deuxièmement, il est appliqué pour traiter les problèmes d'imprécisions et d'incertitudes dans la phase de classification.

Le QRBS simule les connaissances qui lui sont fournies par des évaluations purement linguistiques définies dans un ensemble fini et totalement ordonné de valeurs symboliques. L'interface homme-machine qui a introduit permet de représenter les incertitudes en degrés de probabilités symboliques. Les opérateurs symboliques servent à substituer les formules classiques (numériques) par des formules symboliques. Ces dernières sont

simples à manipuler et opèrent sur des nombres entiers, ce qui établit un lien entre le modèle qualitatif et le monde numérique et se prêtent bien à une interprétation auprès de l'être humain.

Une fois que nous avons présenté les deux systèmes, AIS et QRBS, le chapitre suivant sera consacré à étudier le lien étroit entre ces deux systèmes par expérimentation.

# Proposition d'un système pour la reconnaissance de caractères manuscrits

## 5.1 Introduction

Après avoir détaillé les bases théoriques de chaque méthode de classification (AIS et QRBS) dans les chapitres 3 et 4 respectivement, nous allons essayer de les adapter à la tâche de la reconnaissance de caractères manuscrits.

Dans ce chapitre, nous allons présenter le travail réalisé, qui est basé sur la combinaison de deux approches, notamment le système immunitaire artificiel AIS et le système à base des règles qualitatives QRBS. Nous allons démontrer premièrement l'importance de chaque approche dans le cadre de l'apprentissage et de la décision en illustrant le potentiel de chacune d'eux par expérimentation sur plusieurs exemples de la base de chiffres manuscrits MNIST et des échantillons des lettres anglaises. Nous allons expliquer par la suite de manière détaillée les différentes techniques utilisées pour la réalisation de l'approche que nous avons proposé. Cette dernière s'attache à traiter la qualité de la base de données en minimisant la variabilité intra-classes et en maximisant la variabilité inter-classes. Ceci est effectué par la sélection des meilleurs candidats dans la phase d'apprentissage à travers l'AIS, et le contournement de situations imprécises et incertaines dans la phase de décision à travers le QRBS. Nous allons également montrer, discuter et évaluer les résultats obtenus pour les réglages de paramètres, ainsi que pour la classification. Les bases de données utilisées nous ont permis d'étudier les performances de ces deux approches de manière indépendante ainsi que leur combinaison. Ces performances sont aussi évaluées et comparées à d'autres méthodes bien connues.

## 5.2 Formulation du problème

Comme nous l'avons évoqué dans les deux premiers chapitres, un système de la reconnaissance de caractères se compose de trois sous-étapes principales : le prétraitement, l'extraction de caractéristiques et la classification.

Les étapes de prétraitement sont nécessaires avant l'étape d'extraction de primitives pour ne garder que la structure de base caractérisant un caractère. L'étape d'extraction de primitives consiste à rassembler dans un vecteur plusieurs attributs qui caractérisent de manière unique chaque classe des caractères de la base d'apprentissage. Le but de cette étape est d'extraire les attributs qui permettent de maximiser le taux de reconnaissance

avec la moindre quantité d'information. Après la détermination des vecteurs d'attributs, il est possible de les affecter aux classes possibles des caractères. Cette affectation est faite dans notre travail à travers différents classifieurs (simple et hybride) dont le but est d'augmenter le taux de reconnaissance.

La figure 5.1 décrit une présentation générale des différentes étapes de notre système. Une description détaillée de chaque étape est faite par la suite.

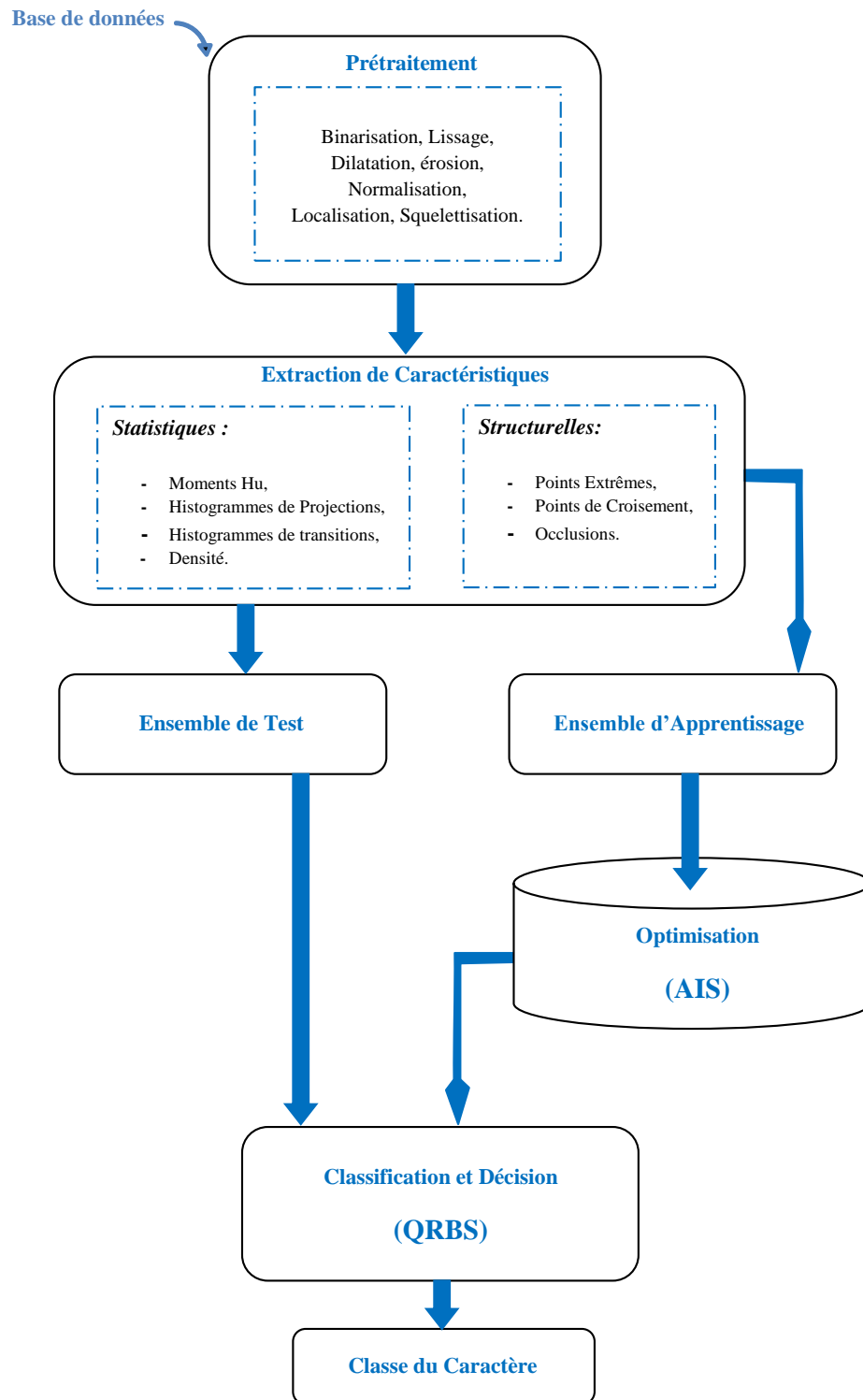


FIGURE 5.1 – Un aperçu du système proposé

### 5.2.1 Prétraitement

Les prétraitements appliqués sur l'image des caractères (chiffres et lettres) permettent, d'une part, d'éliminer ou de réduire le bruit dans l'image, et d'autre part, de simplifier les traitements ultérieurs. Nos opérations de prétraitements comprennent : la binarisation, le lissage, la dilatation, l'érosion, la normalisation, le cadrage (localisation), et la squelettisation.

- **Binarisation.** Le but de la binarisation est de faire surgir l'information utile par rapport à l'arrière-plan. Dans notre cas, binariser veut dire convertir l'image d'entrée en niveaux de gris en une image binaire. Nous avons utilisé la méthode d'Otsu.
- **Lissage.** Nous avons appliqué cette opération afin d'éliminer les bruits introduits dans l'image à cause des systèmes d'acquisition, ou à cause de la qualité du papier et du stylo utilisés.
- **Dilatation et érosion (Fermeture).** Pour but de fermer les bords disjoints et de lisser les bords globaux de l'image. La dilatation élargit l'image originale en utilisant les éléments structurants pour combler les lacunes dans l'image de caractère. L'érosion réduit l'image originale en utilisant les éléments structurants pour éliminer les éléments indésirables.
- **Normalisation.** Il faut impérativement ramener les caractères à la même taille parce que les caractères peuvent subir une légère déformation dans la taille ce qui complique les tâches qui suivent et influence sur la stabilité des paramètres. Nous avons redimensionné l'image originale en matrice de 30x30 pixels dont chaque élément est binaire.
- **Cadrage et squelettisation.** Nous avons utilisé ces deux opérations pour minimiser la surface utilisable et diminuer l'information utile en ne gardant que le squelette de la forme. Cadrer veut dire localiser et garder la zone d'intérêt. Le squelette est un pixel épais produit par la ligne médiane du caractère sans changer sa structure essentielle.

La figure 5.2 montre un exemple d'une image de chiffre 8, avant et après l'étape de prétraitement. L'image d'entrée représente le chiffre original, et l'image de sortie montre le squelette du chiffre. Par conséquent, l'image est propre et prête pour l'extraction de caractéristiques.

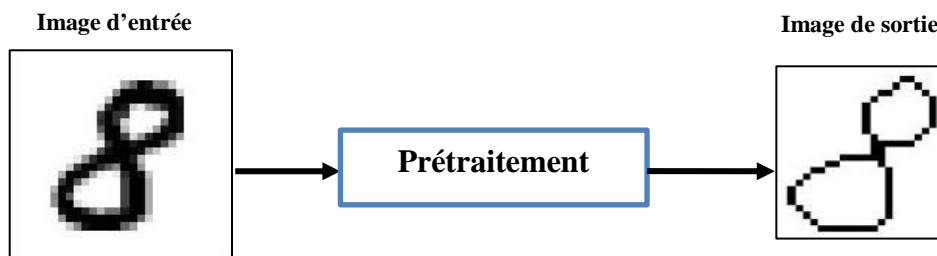


FIGURE 5.2 – Prétraitement d'un chiffre 8 de la base de données MNIST

### 5.2.2 Extraction de caractéristiques

L'extraction de caractéristiques est une phase importante de tout système de reconnaissance. C'est un processus qui permet d'extraire les caractéristiques pertinentes, de



supprimer les données redondantes, et de produire un ensemble de primitives numériques représentant le caractère (chiffre/lettre). Le problème de l'extraction de caractéristiques est défini dans [DEV 82] comme « le problème de l'extraction, à partir des vecteurs de données, les informations qui sont les plus pertinentes pour des raisons de classification, au sens de minimiser la variabilité intra-classe des caractères, tout en augmentant la variabilité inter-classe de ces caractères ». Le choix d'une méthode d'extraction de caractéristiques appropriée dépend principalement de la nature du problème de la reconnaissance, et des données disponibles [TRI 96], mais le succès d'une méthode dans une application particulière ne signifie pas que cette méthode est également efficace dans d'autres applications.

Comme nous l'avons mentionnés dans le chapitre 3, les caractéristiques se divisent en deux catégories : structurelle et statistique [BRI 04 ; RAJ 11]. Les caractéristiques statistiques sont tirées de distributions statistiques des points, et les caractéristiques structurelles sont basées sur les propriétés géométriques et topologiques du caractère. La sélection des meilleures caractéristiques couvrant différents styles d'écriture n'est pas une tâche simple, et il n'y a pas de théorie généralisée qui conduit à une sélection optimale pour réduire au minimum la variabilité intra-classe et maximiser la variabilité inter-classe. Les caractéristiques structurelles et statistiques semblent complémentaires, et l'utilisation d'un mélange de ces deux types peut donner de meilleures performances [HAN 03 ; CAI 99]s. Dans notre travail, 32 caractéristiques sont extraites expérimentalement du caractère : 29 caractéristiques statistiques et 3 structurelles.

### 5.2.2.1 Caractéristiques statistiques

- *Caractéristiques à base des moments.* Les moments de Hu (1962) sont bien connus comme étant invariant à la taille, la translation et la rotation de caractère, et ils fonctionnent plus rapidement (moins de temps consommé) et sont moins sensibles au bruit. Ce sont des mesures statistiques pures de la distribution des pixels autour du centre de gravité du caractère. Leurs efficacité et leurs robustesse les rendent très utile pour décrire les formes dans la reconnaissance de caractères [CAS 87 ; CHI 99]. Ils sont donnés par :

$$\phi_1 = \eta_{20} + \eta_{02} \quad (5.1)$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \quad (5.2)$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \quad (5.3)$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} - \eta_{03})^2 \quad (5.4)$$

$$\begin{aligned} \phi_5 = & (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + \\ & + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \end{aligned} \quad (5.5)$$

$$\eta_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \quad (5.6)$$

$$\begin{aligned} \phi_7 = & (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + \\ & (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \end{aligned} \quad (5.7)$$

Afin de rendre les caractéristiques invariants par translation et changement d'échelle, nous remplaçons le moment géométrique par le moment central normalisé ( $\eta_{pq}$ ). Les notations  $\eta_{pq}$  utilisés ci-dessus sont appelées les moments géométriques centraux normalisés. Comme leur nom l'indique, pour obtenir ces moments, nous calculons le moment nor-

malisé  $\eta_{pq}$  à partir du moment central  $\mu_{pq}$  qui est déterminé par le moment géométrique. Une description de ce procédé est montrée comme suit :

Les moments géométriques classiques  $m_{pq}$  d'une image  $f(x,y)$  sont calculées selon la formule suivante :

$$m_{pq} = \sum_{x=1}^M \sum_{y=1}^N x^p y^q f(x,y) \quad (5.8)$$

Ceci permet de calculer le centre de masse de l'image. Les moments centralisés  $\mu_{pq}$  sont des moments géométriques de cette image par rapport au centre de masse  $(\bar{x}, \bar{y})$  :

$$\mu_{pq} = \sum_{x=1}^M \sum_{y=1}^N (x - \bar{x})^p (y - \bar{y})^q f(x,y) \quad (5.9)$$

Où

$$(\bar{x}, \bar{y}) = \left( \frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right) \quad (5.10)$$

Les moments centraux sont invariants au translation. Pour les rendre invariants par changement d'échelle, nous utilisons des moments normalisés  $\eta_{pq}$  définis par :

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^\gamma} \quad (5.11)$$

Avec

$$\gamma = \frac{p+q}{2} + 1, \quad \forall p+q \geq 2 \quad (5.12)$$

Un exemple de l'utilisation de ces caractéristiques est illustré dans la figure 5.3.



<b>Hu moments</b>	$\emptyset_1$	$\emptyset_2$	$\emptyset_3$	$\emptyset_4$	$\emptyset_5$	$\emptyset_6$	$\emptyset_7$
<b>(a)</b>	0.5380	0.1874	0.0589	0.0326	0.0014	0.0141	-1.054-04
<b>(b)</b>	0.5389	0.1882	0.0540	0.0294	0.0012	0.0128	-8.926-05

FIGURE 5.3 – Exemple d'une transformation : (a) Chiffre "9" original, (b) Rotation de chiffre "9"

- *Caractéristiques à base d'histogrammes de projections et de transitions* . Les histogrammes de projections et de transitions calculent, respectivement, le nombre de pixels de premier plan et le nombre de transitions de premier plan au arrière-plan dans une direction spécifiée (horizontale, verticale et les deux diagonale (45°, 135°)). Pour une image

$f(x,y)$ , nous considérons les quatre directions :

1. Projection horizontale :

$$H_{ph}(i) = \sum_{j=1}^m f(i, j), \quad 1 \leq i \leq n \quad (5.13)$$

2. Projection verticale :

$$H_{pv}(j) = \sum_{i=1}^n f(i, j), \quad 1 \leq j \leq m \quad (5.14)$$

3. Projection diagonale  $45^\circ$  :

$$H_{pd+}(k) = \sum_{\substack{j=\min(k,m), \\ i=k-j+1, i \leq n}} f(i, j), \quad 1 \leq k \leq m+n-1 \quad (5.15)$$

4. Projection diagonale  $135^\circ$  :

$$H_{pd+}(k) = \sum_{\substack{j=\min(k,m), \\ i=k-j+1, i \leq n}} f(i, j), \quad 1 \leq k \leq m+n-1 \quad (5.16)$$

5. Transition horizontale :

$$HT_{NBH}(i) = NBTNB(f, i, H), \quad 1 \leq i \leq n \quad (5.17)$$

6. Transition verticale :

$$HT_{NBV}(i) = NBTNB(f, j, V), \quad 1 \leq j \leq m \quad (5.18)$$

7. Transition diagonale  $45^\circ$  :

$$HT_{NBD+}(k) = NBTNB(f, i, D+), \quad 1 \leq k \leq n+m-1 \quad (5.19)$$

8. Transition diagonale  $135^\circ$  :

$$HT_{NBD-}(k) = NBTNB(f, i, D-), \quad 1 \leq k \leq n+m-1 \quad (5.20)$$

Où :  $NBTNB(f, L, D)$  est une fonction qui permet d'évaluer le nombre de transition du blanc au noir de la matrice image  $f$ , sur une ligne  $L$ , dans la direction  $D$ .

Vingt primitives sont extraites à partir d'histogrammes de projections et de transitions :

- Nous avons calculé, pour les histogrammes de projection horizontale et pour tout les histogrammes de transitions (horizontale, verticale et les deux diagonales ( $45^\circ/135^\circ$ )), la moyenne, la variance et le maximum.
- Pour les histogrammes de projection verticale et diagonale ( $45^\circ/135^\circ$ ), nous avons calculé seulement la moyenne et la variance due à la non-influence de la valeur maximum.

La figure 5.4 présente les histogrammes de projections pour un chiffre (8).

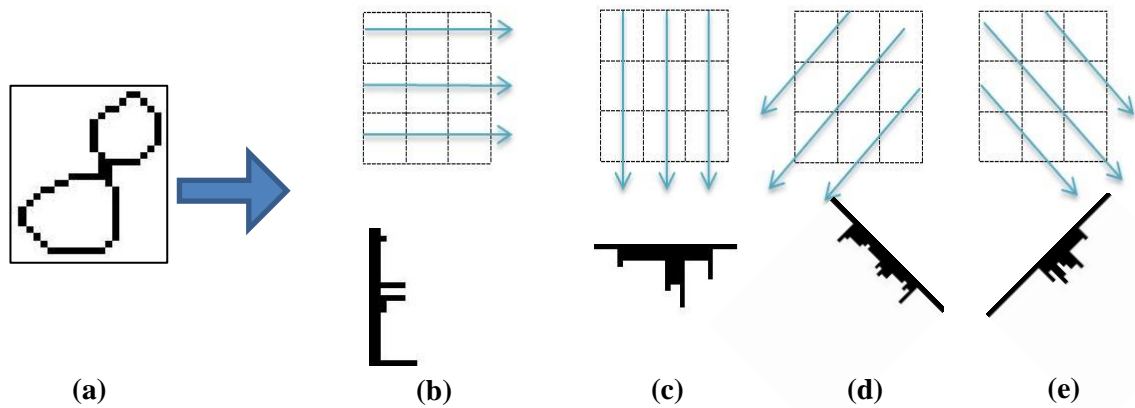


FIGURE 5.4 – Histogrammes de projections. (a) Chiffre squeletté, (b) Projrcion horizontale, (c) Projection verticale, (d) Projection Diagonale 45°, (e) Projection Diagonale 135°

• *Caractéristiques de densités.* La densité de chaque caractère squeletté est le rapport entre de pixels de premier plan et l'aire de l'image, calculée par la formule suivante :

$$D = \frac{\sum p(i, j)}{n * m} \quad (5.21)$$

Où  $p(i, j)$  est la valeur de pixels constituant l'objet.  $m$  et  $n$  sont la longueur et la largeur de l'image de caractère respectivement.

### 5.2.2.2 Caractéristiques structurelles

- *Caractéristiques de point extrême* . Correspond au début/fin d'un segment de ligne dans le squelette de caractère avec seul voisin (figure 5.5).
- *Caractéristiques de points de croisement* . Un point de croisement relie trois branches ou plus (trois voisin) dans l'image (figure 5.5).
- *Caractéristiques d'occlusions*. Correspondent aux contours intérieurs du caractère squeletté (figure 5.5). Ils se distinguent par leur degré d'inclusion (intégrale ou partielle) dans l'image.

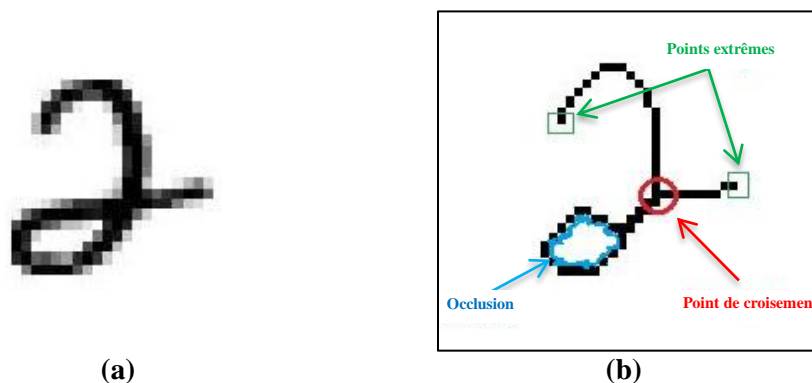


FIGURE 5.5 – Caractéristiques structurelles. (a) Chiffre " 2 ", (b) Points extrêmes, point de croisement et occlusion extraites à partir du squelette de chiffre " 2 "

### 5.2.3 Classification

Cette étape expose l'approche proposée pour la classification automatique des caractères manuscrits en utilisant une combinaison entre le système immunitaire artificiel AIRS et la méthode qualitative QRBS. Nous allons appliquer le système résultant AIS-QRBS [MER 16], sur des caractères isolés, tout en proposant les traitements adéquats pour la construction des meilleures images d'entrée de ce système. Nous présentons par la suite deux projections de ce système : la première est une méthode de classification par l'algorithme AIRS, dans laquelle nous proposons l'utilisation de règles floues dans sa phase de classification. La deuxième est une méthode de classification par l'approche bio-qualitative, où on fait un appel à deux processus principaux : un processus de réduction de la base de données et un autre de décision par des règles de pondération qualitatives.

## 5.3 Les approches proposées pour la reconnaissance de caractères manuscrits

### 5.3.1 L'algorithme AIRS

Dans cette section, nous testons la capacité du l'AIRS pour reconnaître des formes complexes (caractères manuscrits) tout en explorons leur mécanisme d'adaptation à la variabilité de ces formes.

#### 5.3.1.1 Description de l'algorithme

L'AIRS est un algorithme d'apprentissage supervisé inspiré du métaphore de système immunitaire. Cet algorithme prend un antigène comme entrée et produit un ensemble de cellules mémoires comme sortie. Cet ensemble de cellules mémoires (MC) est considéré comme une base représentative qui est utilisée durant la phase de classification. Dans l'algorithme AIRS, les antigènes (*Ag*) représentent les données d'apprentissage et de test, et les ARBs (Artificial Recognition Balls) représentent l'ensemble des anticorps.

L'algorithme AIRS est composé essentiellement de quatre étapes successives [JOU 09 ; ZHU 11] : l'initialisation, l'identification de cellule mémoire et génération des ARBs, l'allocation des ressources et compétition ; et l'introduction de cellule mémoire candidate.

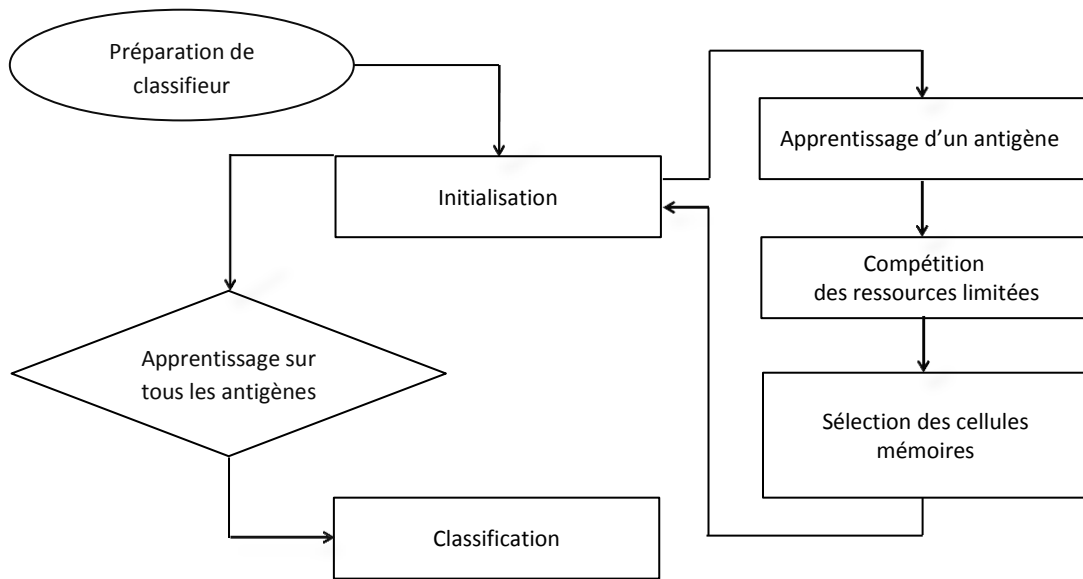


FIGURE 5.6 – Un aperçu du cycle de vie de l'algorithme AIRS

### 5.3.1.1.1 Initialisation

Lors de l'initialisation, toutes les données d'apprentissage (antigènes) sont normalisées de sorte que la valeur de la distance euclidienne entre chaque couple de vecteurs de l'ensemble d'apprentissages est dans l'intervalle  $[0, 1]$ . Un seuil  $AT$  est la valeur d'affinité moyenne de tous ces antigènes. Il est calculé en utilisant l'expression suivante :

$$\text{seuil d'affinité} = \frac{\sum_{i=1}^n \sum_{j=i+1}^n \text{affinité}(Ag_i, Ag_j)}{\frac{n(n-1)}{2}} \quad (5.22)$$

Où  $n$  est le nombre de données d'apprentissage (antigènes),  $Ag_i$  et  $Ag_j$  sont deux antigènes, et  $\text{affinité}(Ag_i, Ag_j)$  représente la distance euclidienne entre les vecteurs caractéristiques de ces deux antigènes.

Cette valeur étant calculée, s'utilise ensuite pendant la phase d'apprentissage afin de déterminer si des cellules de mémoire candidates, évoluées dans cette phase, peuvent remplacer les cellules mémoire existants dans le classifieur (ce point sera expliqué ultérieurement).

Après la normalisation, l'étape suivante consiste à initialiser le pool de cellules mémoires (memory cell pool MC). L'initialisation du MC est une étape facultative, elle consiste à sélectionner aléatoirement un certain nombre d'antigènes pour les transformer en cellules mémoires. De même manière on a initialisé l'ensemble des ARBs.

Pour chaque échantillon d'apprentissage  $Ag$ , nous effectuons les étapes d'apprentissage suivantes :

### 5.3.1.1.2 Apprentissage de l'antigène

#### A. Identification de cellule mémoire et génération des ARBs

Cette étape a pour but de sélectionner, à partir de l'ensemble MC, une cellule mémoire  $mc_{match}$  qui a la plus haute valeur de stimulation par rapport à l'antigène en cours de traitement.

$$mc_{match} = \operatorname{argmax}_{mc \in MCAg.c} stimulation(Ag, mc) \quad (5.23)$$

La stimulation est la fonction inverse de l'affinité, c'est-à-dire une affinité à faible valeur implique une stimulation antigénique plus importante. Elle est définie par l'expression  $1 - \operatorname{dist}(Ag, mc)$  où  $\operatorname{dist}$  désigne la distance euclidienne entre les deux vecteurs sélectionnés.

Une fois  $mc_{match}$  identifiée, elle sera utilisée pour générer de nouvelles ARBs (clonage). Ensuite, ces nouveaux clones sont ajoutés à la population des ARBs préexistants. Le nombre de clones est calculé comme suit :

$$\text{nombre\_clones} = \text{hyper\_colonale\_rate} * \text{clonale\_rate} * \text{stimulation}(Ag, mc_{match}) \quad (5.24)$$

Chaque caractéristique du nouveau clone peut être mutée à un taux de mutation défini, selon la fonction de mutation que nous avons proposé (décrite dans l'annexe), afin de maintenir la diversité entre les candidats des cellules mémoires.

### B. Allocation des ressources et compétition

Le but de cette partie de l'algorithme est de développer une bonne cellule mémoire candidate afin de mieux classer un antigène donné. Ceci est fait en ajoutant un processus d'allocation de ressources qui permet de réduire le coût de calcul dans la génération des clones. Ce processus est lié à la stimulation de chaque élément d'ARB (Ab) avec l'antigène courant et un taux de clonage prédéfini. Ce processus peut être décrit dans la figure suivante :

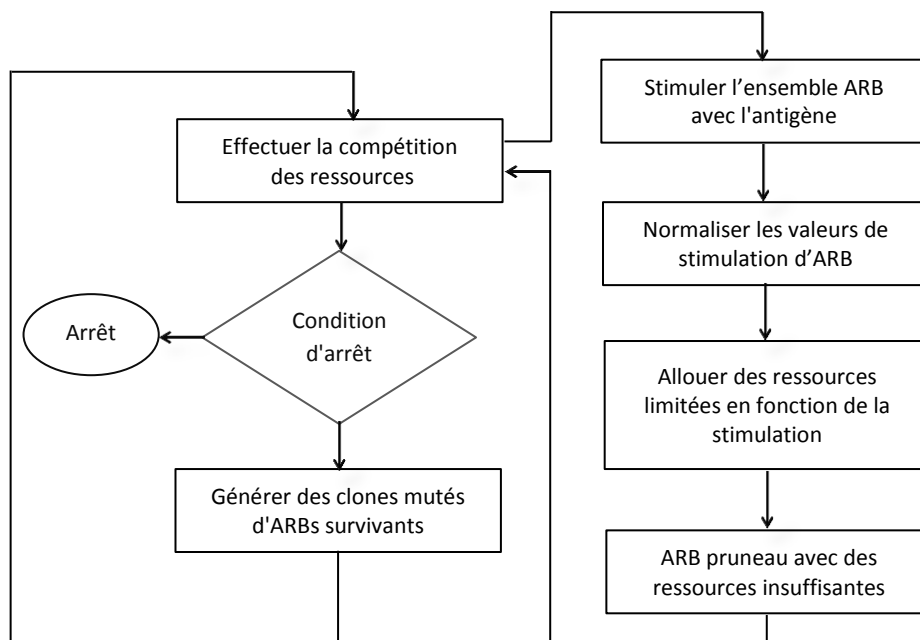


FIGURE 5.7 – Processus d'allocation des ressources et compétition

Un membre d'ARB avec une stimulation antigénique plus élevée sera fourni avec un grand nombre de ressources. Puis, les membres d'ARB avec des stimulations antigénique plus petits et ceux qui n'ont pas de ressources sont éliminés du système.

$$ressources = stimulation(Ag, Ab) * clonale\_rate \quad (5.25)$$

Ce processus d'allocation des ressources est répété jusqu'à ce que la stimulation moyenne  $S_i$ , pour chaque sous-ensemble de la classe d' $Ab$ , soit supérieur au seuil d'affinité  $AT$ .

$$S_i = \frac{\sum_{j=1}^{|ARB_i|} Ab_j.stim}{|ARB_i|}, \quad Ab_j \in ARB_i \quad (5.26)$$

Où  $i = 1, 2, \dots, nc$ ,  $nc$  représente le nombre de classes,  $|ARB_i|$  représente le nombre d'ARBs appartenant à la  $i^{me}$  classe, et  $Ab_j.stim$  représente le niveau de stimulation de la  $j^{me}$  ARB de la  $i^{me}$  classe.

À ce point, la population ne se compose que des ARBs ayant la stimulation la plus élevée avec l'antigène donné, et sont enfin prêts pour la phase finale de l'apprentissage.

### C. Introduction de cellule mémoire candidate

Après que la valeur totale de stimulation dans toutes les classes atteint le seuil d'affinité, on choisit parmi les ARBs restants, une cellule mémoire candidate,  $mc_{candidate}$  qui a la stimulation la plus élevée avec l'antigène d'apprentissage. Cette cellule mémoire candidate est ajoutée à l'ensemble de cellules mémoires en utilisant l'algorithme expliqué dans [JOU 09] comme suit :

---

#### Algorithme 5.1 : Mise à jour de l'ensemble de cellules mémoire

---

**Données :**  $Ag$ =antigène.  
**Résultat :**  $mc$ =cellule mémoire apparié.  
 Calculer  
 $CondStim \leftarrow stimulation(Ag, mc_{candidate})$   
 $MatchStim \leftarrow stimulation(Ag, mc_{match})$   
 $CellAff \leftarrow affinité(mc_{candidate}, mc_{match})$   
**si** ( $CondStim > MatchStim$ ) **alors**  
     **si** ( $CellAff < AT * ATS$ ) **alors**  
          $MC \leftarrow MC - mc_{match}$   
     **fin**  
      $MC \leftarrow MC + mc_{candidate}$   
**fin**

---

La cellule mémoire candidate est ajoutée au pool de cellules mémoire MC seulement si elle est plus stimulée par  $Ag$  que  $mc_{match}$ . De plus, si l'affinité entre  $mc_{candidate}$  et  $mc_{match}$  est inférieure à  $AT * ATS$  (où  $AT$  est le seuil d'affinité calculé lors de l'initialisation, et  $ATS$  est un paramètre défini par l'utilisateur nommé : affinity threshold scalar), alors  $mc_{candidate}$  remplace  $mc_{match}$  dans MC.

Une fois que la cellule mémoire candidate a été évalué pour l'addition dans le MC établie, l'entraînement sur cet antigène se termine. L'antigène suivant est ensuite sélectionné à partir de l'ensemble d'apprentissage et les sous-étapes A., B. et C. sont répétées jusqu'à ce que tous les antigènes soient présentés au système. Un pool de cellules mémoires final est généré et prêt pour la classification. Cet ensemble résultant représente et couvre théoriquement l'espace de distribution total de l'ensemble d'exemples d'apprentissage.

Une vue plus simple de ces étapes est illustrée par l'algorithme 3.4 du chapitre 3. Les détails de ces étapes d'apprentissage seront représentés dans l'annexe A sous-forme



algorithmique.

### 5.3.1.1.3 Classification

Lorsque le processus d'apprentissage est terminé, le pool de cellules mémoire résultant devient le cœur de l'AIRS. La classification des exemples de test est effectuée par deux manières. Dans la première, nous utilisons le principe de K-plus-proche voisin (KNN) [SER 13] où se trouvent les  $k$  meilleures correspondances aux exemples de données, et la classe est déterminée par la majorité de votes.

Dans la deuxième, le concept de flou est introduit, car il existe quelques problèmes avec l'utilisation de l'approche KNN. L'un de ces problèmes est que, généralement, chacun des voisins trouvés a une importance similaire aux autres voisins dans l'affectation de l'étiquette de classe du vecteur d'entrée. Un autre problème est que, quand un vecteur d'entrée est attribué à une classe, il ne détermine pas la force d'adhésion (d'appartenance) à cette classe [KEL 85]. Le K-plus proche voisin flou (Fuzzy-KNN) [CHI 12] est utilisé dans ce cas pour surmonter les limites de la KNN en attribuant une appartenance de classe à chaque caractère de test [MER 14]. Le concept des ensembles flous dans cette approche est utilisé, dont les éléments ont un degré d'appartenance. Dans la théorie des ensembles classiques, un élément appartient ou non à un ensemble. Dans les ensembles flous, les éléments de l'ensemble ont une fonction d'appartenance attachée à eux dans l'intervalle réel  $[0, 1]$ .

L'approche Fuzzy-KNN trouve les  $k$  les plus proches voisins de l'exemple de test, et attribue par la suite une valeur d'appartenance pour chaque exemple à toutes les classes. Cette valeur d'appartenance dépend de la classe de ses  $k$  plus proches voisins. Un nouvel exemple  $X$  a une appartenance  $u_i(X)$ , pour une classe  $i$ , donnée par :

$$u_i(X) = \frac{\sum_{j=1}^k u_{ij} \left( \frac{1}{d(X, x_j)^{\frac{2}{m-1}}} \right)}{\sum_{j=1}^k \left( \frac{1}{d(X, x_j)^{\frac{2}{m-1}}} \right)} \quad (5.27)$$

Où :  $u_{ij}$  donne le degré d'appartenance du  $j^{\text{ème}}$  vecteur de l'ensemble d'apprentissage à la  $i^{\text{ème}}$  classe.  $d(X, x_j)$  est la distance entre  $X$  et son  $j^{\text{ème}}$  voisin le plus proche  $x_j$ . Le paramètre  $m$  détermine comment la distance est fortement pondérée lors du calcul de l'appartenance à une classe.

La classe affectée au vecteur  $X$  est donnée par :

$$\operatorname{argmax}_{i=1}^{nc} (u_i(X)) \quad (5.28)$$

Le pseudo code de l'algorithme Fuzzy-KNN est présenté comme suit [KEL 85] :

---

**Algorithme 5.2 :** Pseudo code du Fuzzy-KNN

---

**Données :**  $x$  de classification inconnue  
 Soit  $k$ ,  $1 \leq k \leq n$ .  
 $i = 1$   
**tant que**  $k - nn$  à  $x$  trouvé **faire**  
 | Calculer la distance  $dist(x, x_i)$   
 | **si**  $i \leq k$  **alors**  
 | | Inclure  $x_i$  dans l'ensemble de  $k - nn$ .  
 | **fin**  
 | **sinon si**  $x_i$  plus proche de  $x$  que tout plus proche voisin précédent **alors**  
 | | Supprimer le plus éloigné des  $k - nn$   
 | | Inclure  $x_i$  dans l'ensemble de  $k - nn$ .  
 | **fin**  
 |  $i = i + 1$   
**fin**  
 $i = 1$   
**tant que**  $x$  est assigné une appartenance à toutes les classes **faire**  
 | Calculer l'équation 5.27  
 |  $i = i + 1$ .  
**fin**

---

Pour la décision de Fuzzy-KNN, chaque caractère a une appartenance de 1 aux classes qui figurent dans les  $k$  plus proches voisins, et une appartenance de 0 aux autres classes, et est attribué à la classe qui a la valeur d'appartenance la plus élevée.

### 5.3.1.2 Paramètres de l'algorithme

Comme mentionné précédemment, l'algorithme AIRS a un certain nombre de paramètres configurables par l'utilisateur pour affiner le processus d'apprentissage à des problèmes spécifiques des domaines. Il a été montré par Watkins [WAT 01] que cette technique reste relativement stable (en termes de précision de classification) dans une gamme de paramètres sur différentes bases de données standard. Cette section donne un bref aperçu de la nature et l'effet spécifique de chacun des paramètres d'algorithme.

- **Affinité (Affinity).** Une mesure de la "proximité" ou de la "similarité" entre deux anticorps ou antigènes. Dans l'implémentation courante, elle n'est autre que la distance euclidienne entre deux vecteurs de caractéristiques, ses valeurs appartiennent à l'intervalle  $[0, 1]$ . En outre, une petite valeur d'affinité indique une forte similarité entre les vecteurs.
- **Boule de reconnaissance artificielle (Artificial Recognition Ball ARB).** L'ARB est également connu sous le nom *B-Cell*. Un ARB se compose d'un anticorps, d'un comptage du nombre de ressources détenues par la cellule, et d'une valeur courante de stimulation de cette cellule.
- **Cellule mémoire candidate (Candidate memory cell).** Un anticorps de l'ARB, de la même classe comme l'antigène d'apprentissage, qui a été le plus stimulé après l'exposition à l'antigène donné.
- **Ensemble de tests (testing set).** Une collection des antigènes utilisés pour évaluer la performance de classification du AIRS.

- **Ensemble d'apprentissage (training set)**. Une collection des antigènes utilisés pour la formation de l'AIRS.
- **Nombre d'exemples d'initialisation (Number of Initialisation Instances)**. Le nombre d'exemples d'initialisation est le nombre d'instances de l'ensemble d'apprentissage choisis au hasard et utilisés pour initialiser le pool de cellules mémoire. Ce paramètre peut être dans l'intervalle  $[0, \text{nombre total de l'ensemble d'apprentissage}]$ , et est généralement fixé à des faibles valeurs. La souplesse de ce paramètre, en particulier lorsqu'il est réglé sur des faibles valeurs, permet à l'algorithme de déterminer automatiquement le nombre et la nature des éléments de cellules mémoire qui composent le classifieur produit.
- **Ressources (resources)**. Les ressources placent une limite directe sur le nombre d'ARBs qui peut coexister dans la population ARBs. Chaque ARB alloue un nombre de ressources basé sur sa valeur de stimulation et son taux clonale. Le nombre total de ressources du système est réglé sur une certaine limite. Si le nombre de ressources consommées est supérieur au nombre de ressources autorisés à exister dans le système, alors les ressources sont retirées de l'ensemble des ARBs les moins stimulés jusqu'à ce que le nombre de ressources dans le système retourne au nombre autorisé (alloué). Si toutes les ressources d'un ARB donné sont retirées, alors cet ARB est retiré de la population de cellules.
- **Seuil d'affinité (Affinity threshold AT)**. Le seuil d'affinité est la valeur d'affinité moyenne parmi tous les antigènes de l'ensemble d'apprentissage ou parmi un sous-ensemble sélectionné de ces antigènes d'apprentissage.
- **Seuil d'affinité scalaire (Affinity threshold scalar ATS)**. Une valeur comprise entre 0 et 1 qui, lorsqu'il est multiplié par le seuil d'affinité, fournit une valeur de coupure pour le remplacement des cellules mémoire dans le cycle d'apprentissage d'AIRS.

L'*ATS* a l'effet le plus direct sur le remplacement de cellules mémoires. Autrement dit, une cellule mémoire nouvellement évoluée (candidate) remplace une cellule mémoire déjà établie (match) si et seulement si elle est plus stimulée par l'antigène d'apprentissage et l'affinité entre les deux cellules mémoires (candidate et match) est inférieure au produit de l'*AT* et l'*ATS*.

En outre, l'*ATS* fournit un moyen de réglage automatique de l'*AT* en le rendant plus souple (inférieure à la moyenne) ou plus dur (supérieure à la moyenne). L'effet de ramollissement de l'*AT* provoque moins de remplacement des cellules mémoires. L'inverse est vrai lorsque le seuil est plus dur. Les valeurs courantes pour ce paramètre utilisateur sont dans l'intervalle  $[0,1, 0,3]$ , qui est une atténuation considérable de la moyenne, provoquant moins de remplacement, et donc un plus grand pool de cellules mémoires (diversité des exemples), et avec une classification plus efficace.

- **Seuil de stimulation (stimulation threshold)**. Un paramètre entre 0 et 1 utilisé comme un critère d'arrêt pour l'apprentissage sur un antigène spécifique. Pour l'implémentation courante, uniquement lorsque la valeur de stimulation moyenne des ARBs de chaque classe est supérieure au seuil de stimulation qui fait l'arrêt d'apprentissage en réaction à un antigène particulier. Ce paramètre contrôle la quantité de raffinement effectué sur les ARBs pour un antigène donné, et donc à quel point les ARBs seront en réaction à l'antigène.

- **Taux clonale (*Clonale\_rate*)**. Une valeur entière utilisée pour déterminer le nombre de clones à produire par un ARB après avoir répondu à un antigène donné.

Le taux clonale est utilisé dans trois endroits de l'algorithme. En premier lieu, il est utilisé en conjonction avec le taux d'hypermuation pour déterminer le nombre de clones d'une meilleure cellule mémoire correspondante (match) qui peut créer à la population d'ARB. En second lieu, il est utilisé pour déterminer le nombre de clones créés par chaque ARB pendant la phase de clonage d'ARB. Enfin, il est multiplié par la stimulation normalisée d'un ARB pour déterminer leurs ressources allouées. Cela signifie que le nombre de clones ARB créés sera dans l'intervalle de  $[0, \text{Taux clonale}]$ . Cela implique aussi que les ressources allouées pour un ARB seront également dans cet intervalle, ce qui a un effet sur le nombre total de ressources qui devraient être allouées. Par conséquent, le taux clonale a un double rôle en tant que facteur d'allocation des ressources et facteur de mutation clonale pour la population de cellules. Les valeurs typiques pour le taux clonale sont 10.

- **Taux d'hypermuation (*hypermuation\_rate*)**. Une valeur entière utilisée avec le taux clonale et la stimulation des cellules pour déterminer le nombre de clones mutés qu'une cellule mémoire match peut créer. Comme mentionné précédemment, le nombre de clones sera dans l'intervalle  $[0, \text{Taux clonale}]$  qui est alors augmenté par un facteur du taux d'hypermuation. Dans l'implémentation actuelle, la cellule mémoire sélectionnée injecte au moins « *taux d'hyper mutation \* taux clonale \* valeur de stimulation* » de clones mutés dans la population cellulaire au moment de l'introduction de l'antigène.
- **Taux de mutation (*mutation\_rate*)**. Un paramètre compris entre 0 et 1 qui indique la probabilité de mutation de chaque caractéristique donnée d'un ARB.

### 5.3.2 L'approche bio-qualitative

Le but de tout algorithme d'apprentissage consiste à former une généralisation, à partir d'un ensemble d'entraînement d'objets étiquetés, afin de maximiser la précision de la classification de nouveaux objets. La précision maximale dépend principalement de la qualité des données d'entrée et de l'adéquation de l'algorithme d'apprentissage choisi pour ces données. La présence des cas d'échantillons d'entraînement chevauchés, mal étiquetés ou bien bruités peut fortement dégrader la précision de la classification correspondante, et par conséquent, les limites de décision qui en découlent seront difficiles.

La plupart des travaux, dans le domaine de RdF, sont basés sur des décisions nettes ou floues pondérées par des probabilités numériques ou degrés de croyance. En fait, les deux approches n'ont pas une signification parfaite et doivent encore être mises au point pour donner plus de sens aux problèmes de classification. L'utilisation de l'approche qualitative comme un moyen naturel, qui est proche de la perception humaine, peut augmenter la compréhension aux problèmes de classification, et plus précisément, dans la reconnaissance de caractères d'écriture.

Au regard au schéma de la figure 5.1, le déroulement de l'approche proposée se passe par deux phases principales : la phase de réduction de données (optimisation) et la phase de classification.

### 5.3.2.1 L'algorithme AIS pour la réduction de données

Le processus de réduction de donnée n'est pas comme l'extraction de caractéristiques ou la sélection de caractéristiques. Le processus d'extraction de caractéristiques représente le caractère entier comme un petit ensemble de caractéristiques pertinentes sous forme d'un vecteur, et le processus de sélection de caractéristiques choisit un sous-ensemble, à partir de ce premier ensemble de sorte que l'espace de caractéristiques est réduit de façon optimale selon un certain critère [MOT 02]. Ces deux processus visent à réduire la quantité de données en représentant les caractères dans des petits ensembles de caractéristiques et en se concentrant sur les données pertinentes. La représentation de données dans ces deux cas a changé, mais l'ensemble total de leurs exemples n'a pas changé. En fonction de ces caractéristiques, le module de réduction de données permet la réalisation de ces deux modifications.

Le module de réduction des données est le processus où un plus petit ensemble de prototypes est sélectionné et utilisé pour la phase de classification. Les candidats résultants peuvent contenir soit des membres du prototype de l'ensemble original de données ou de nouveaux objets formés en utilisant les modèles originaux. Ce module permet de fournir des améliorations significatives dans la précision de la classification ainsi que la réduction de stockage et de temps de calcul nécessaire.

Par exemple : soit un ensemble de données  $D$  de  $n$  éléments,  $D = \{d_1, d_2, \dots, d_n\}$  où  $d_i$  représente un élément de cet ensemble. Chaque  $d_i$  a deux attributs : vecteur de caractéristiques ( $d_i.f$ ) et étiquette de classe ( $d_i.c \in C = \{1, 2, \dots, k\}$ ). L'objectif du processus de réduction de données est de trouver à partir de l'ensemble original  $D$ , un ensemble condensé  $D'$  tel que  $|D'| \ll |D|$ .

Le méta heuristique proposé permet de traiter le problème d'apprentissage en termes d'optimisation et de complexité. Les systèmes immunitaires artificiels étant auto-organisés alors adaptatifs, apprenants et évolutifs, ils présentent certaine valeur heuristique et des capacités d'optimisation potentielles.

Les cellules mémoire du système immunitaire ont la capacité de regrouper les envahisseurs étrangers par rapport à leur origine, et elles produisent des réponses immunitaires en fonction de ce groupement [SAH 07]. Dans notre travail, cette capacité de regroupement du système immunitaire a été utilisée comme source d'inspiration pour concevoir l'algorithme de réduction de données.

L'algorithme AIS prend un antigène comme entrée et produit un ensemble de cellules mémoire en tant que sortie. Cette ensemble de cellules mémoire est considéré comme l'ensemble condensé de prototypes qui est utilisé par le QRBS lors de la classification. Dans l'algorithme AIS, les antigènes  $Ag$  se réfèrent aux données d'apprentissage et de test, les cellules mémoire MC sont les points de données représentatives (prototypes), et les boules de reconnaissance artificielles ARB désignent un groupe d'anticorps  $Ab$ .

En projetant l'exemple précédent :  $D$  est considéré comme l'ensemble des antigènes  $AG = \{Ag_1, Ag_2, \dots, Ag_n\}$ . L'algorithme AIS est utilisé pour obtenir l'ensemble  $D'$  qui est synonyme au pool de cellule mémoire,  $MC = \{mc_1, mc_2, \dots, mc_m\}$ , où  $m \leq n$  et  $mc_i$  est une cellule mémoire ayant deux caractéristiques similaires à celles d'un antigène particulier. Pour tout  $mc_i$ ,  $mc_i.c \in C = \{1, 2, \dots, k\}$  est la classe correspondante et  $mc_i.f$  est le vecteur caractéristique.

Une simple vue d'étapes de l'AIS utilisé pour la réduction des données est illustré dans la figure suivante :

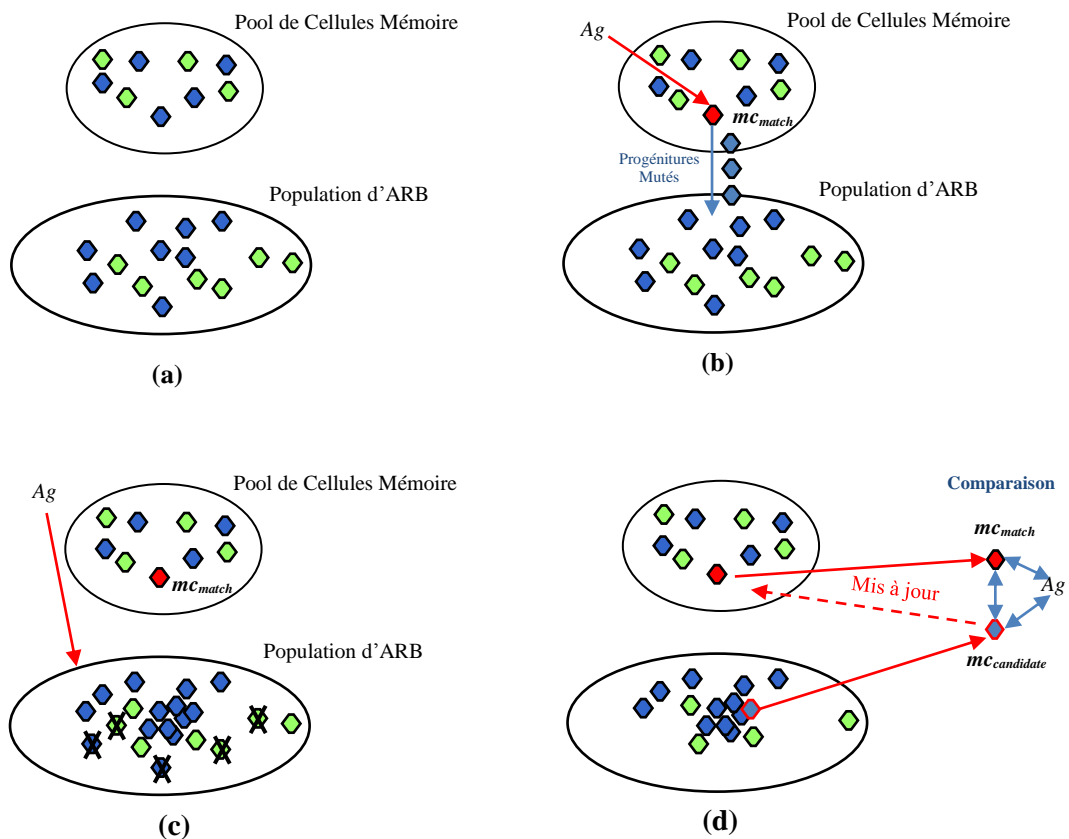


FIGURE 5.8 – Une illustration de différentes étapes de l'algorithme AIS pour la réduction des données

### 5.3.2.2 L'algorithme QRBS pour la classification

La phase de réduction des données consiste à trouver ou extraire le nombre optimal de cellules mémoire MC représentant les caractéristiques pertinentes de l'ensemble d'apprentissage qui donnera une précision de classification la plus élevée. L'algorithme AIS présenté dans la section précédente est appliqué pour réduire la dimensionnalité de l'ensemble de d'apprentissage tout en gardant les meilleurs candidats. Les valeurs optimales des paramètres de l'algorithme AIS conduisant à l'ensemble optimal d'apprentissage sont présentées au tableau 5.1. Ces valeurs sont définies expérimentalement. Les résultats de l'AIS sont soumis au QRBS pour extraire des règles pondérées par des degrés qualitatifs qui servent à classer des objets inconnus.

Le but de l'introduction de l'approche qualitative QRBS dans la phase de décision est également double : d'une part, il permet de traiter à la fois l'imprécision et l'incertitude dans les caractères manuscrits en utilisant des règles qualitatives pour les reconnaître. Deuxièmement, il permet d'optimiser la prise de décision finale en introduisant le processus de prise de décision optimal.

L'algorithme QRBS est composé principalement de deux sous-systèmes : un générateur automatique de règles et un système d'inférence. Le premier sous-système génère des règles de classification à partir de l'ensemble d'apprentissage optimisé. Le deuxième sous-système consiste à classer des nouveaux objets à partir de l'ensemble de tests.

Cet algorithme montre sa supériorité par rapport aux autres approches numériques

[SER 06; BEL 10]. Cependant, le nombre de règles augmente lorsque l'ensemble d'apprentissage est grand et le nombre de classes est important. Cela peut augmenter le temps d'exécution et peut diminuer la performance de l'algorithme. Pour minimiser cette contrainte, l'ensemble d'apprentissage est optimisé en réduisant sa taille et en préservant les meilleurs candidats représentatifs.

Les résultats expérimentaux de notre système sur différentes bases de données montrent le potentiel de l'introduction de cette approche qualitative dans la phase de décision.

## 5.4 Résultats expérimentaux

### 5.4.1 Bases de données

Pour évaluer les performances de notre système, deux types de caractères manuscrits ont été employés : les chiffres manuscrits obtenus à partir de la base de caractères manuscrits MNIST [LEC 98], et des échantillons de lettres anglaises manuscrites de [GAU 09]. La base de données MNIST pour « *Modified ou Mixed National Institute of Standards and Technology* » est une base de données de chiffres arabes manuscrits. Elle est considérée comme un jeu de données très utilisé en apprentissage automatique [DEN 12].

Vu que la problématique de la reconnaissance d'écriture manuscrite est difficile à résoudre, la base MNIST reste un bon test pour les algorithmes d'apprentissage. elle est également devenue un test standard. Elle regroupe 60000 images d'apprentissage et 10000 images de test, issues d'une base de données antérieure, appelée simplement NIST. Les chiffres sont en noir et blanc dans une image de niveau de gris de taille 28x28 pixels, ou 784 pixels au total comme des caractéristiques [DEN 12]. La base notée MNIST de chiffres manuscrits présente l'avantage d'être une base de référence dans la communauté scientifique [LeC 98].

La reconnaissance de chiffres manuscrits est presque au même niveau de difficulté que la reconnaissance de caractères manuscrits. La seule distinction entre eux réside dans la taille du vocabulaire, les chiffres manuscrits ont un vocabulaire très restreint (10 chiffres).

La deuxième base de données proposée par Dinesh Dileep Gaurav [GAU 12] contient 650 échantillons de lettres anglaises de A à Z, avec 25 images pour chaque lettre. Les images montrées ci-dessous tirées de ces deux bases données indiquent une idée sur les bases d'apprentissage employées.

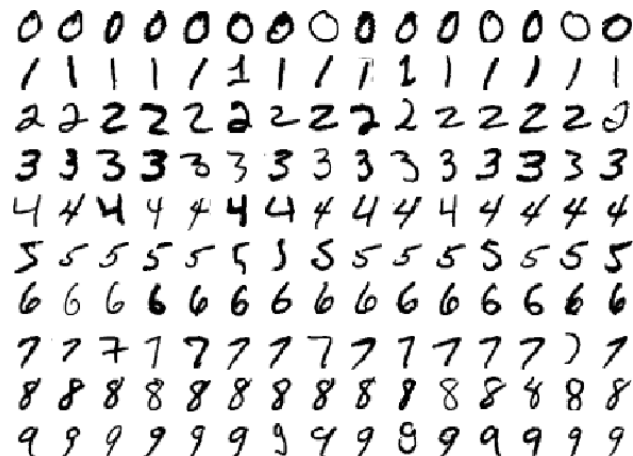


FIGURE 5.9 – Un échantillon de chiffres de la base de données MNIST [LEC 98]

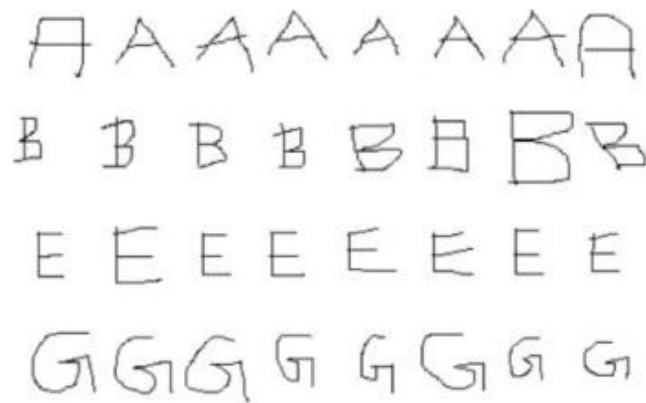


FIGURE 5.10 – Un échantillon de lettres anglaises manuscrites [GAU 12]

Afin d'évaluer la performance de notre approche sur les chiffres et les lettres manuscrites isolés, la base de données est divisée en deux parties. La première partie est composée d'un ensemble d'images d'apprentissage (8000 images de chiffres et 500 images de lettres), et la deuxième partie contient un ensemble d'images de test (2000 images de chiffres et 150 images de lettres).

Après la préparation de la base de données, un programme informatique, mettant en œuvre l'approche proposée, a été développé avec et sans l'utilisation de l'AIS. Le programme a été testé sur ces bases de données. L'évaluation de performances de l'approche proposée est présentée, et les résultats expérimentaux sont donnés, discutés et comparés avec d'autres approches.

### 5.4.2 Évaluation de l'AIS

La première étape de l'apprentissage consiste à effectuer plusieurs tests pour initialiser les paramètres (*Taux clonale*, *Taux hyper\_clonale*, *Taux d'hypermutation*, *Taux de mutation* et *Seuil d'affinité scalaire*) qui sont nécessaires pour le calcul du nombre de clones, les ressources d'ARBs et la fonction de mutation. Les valeurs optimales des paramètres de l'algorithme AIRS conduisant à un ensemble optimal d'apprentissage sont présentées dans le tableau 5.1.

Tableau 5.1 – Paramètres de sélection pour AIRS.

Paramètres	Valeurs
Taux clonale	10
Taux hyper_clonale	4
Taux de mutation	0.1
Taux d'hypermutation	15
Seuil d'affinité scalaire	0.01

L'utilisation de l'AIRS avec l'approche KNN dans la phase de classification donne un résultat de performance de 95.75% pour les chiffres et 84.66% pour les lettres, avec  $k=3$ . Comme premier essai, ce résultat est prometteur pour une reconnaissance des caractères manuscrits.

Pour voir l'effet des paramètres initiaux sur la précision de classification, nous avons effectué des expériences sur des paramètres qui influent sur la variation de nombre d'ARBs initial (ARB population) et le nombre de clones (number of clones). Le but principale de



cette étude est de déterminer quel effet et quelle modification doit retourner ces paramètres afin d'éviter les influences négatives sur la précision de classification.

La figure 5.11 montre l'effet de la variation du nombre d'anticorps initial (ARB) dans l'intervalle [10, 800], sur le taux de reconnaissance au cours de la phase d'apprentissage.

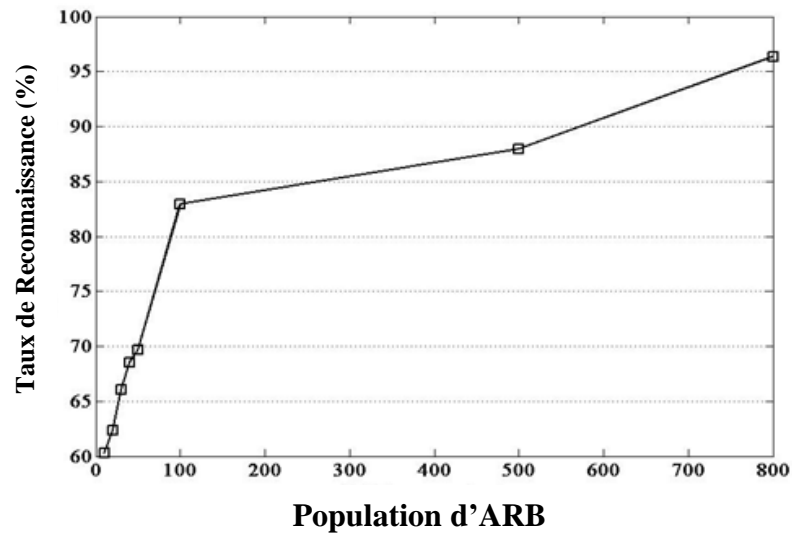


FIGURE 5.11 – Taux de reconnaissance en fonction de l'initialisation des ARB

La figure 5.12 montre l'effet de la variation du nombre de clones dans l'intervalle [40, 1000] sur le taux de reconnaissance au cours de la phase d'apprentissage

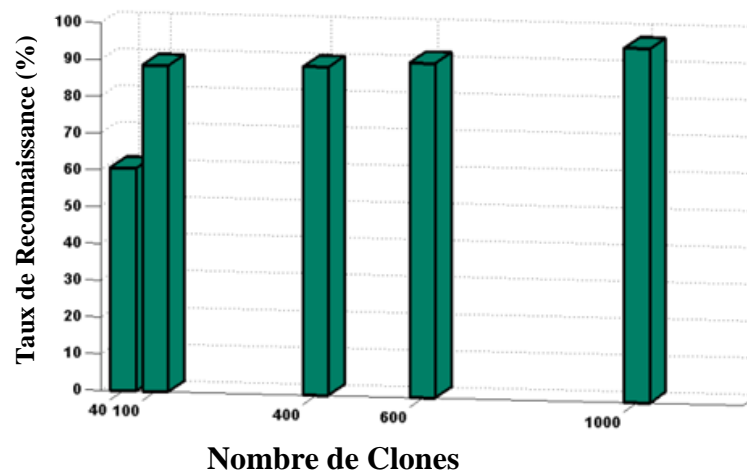


FIGURE 5.12 – Taux de reconnaissance en fonction du nombre de clones

Les figures 5.11 et 5.12 montrent qu'avec l'augmentation de l'ensemble des ARBs initial et le nombre de clones, le taux de reconnaissance augmentent en même temps. Cette amélioration du taux de reconnaissance est due principalement à la diversité de la population qui change leurs caractéristiques selon l'antigène présenté à son entrée. Cela montre l'adaptation de l'algorithme AIRS au changement des données d'entrée.

Dans le but d'examiner la manière dont les valeurs du paramètre taux clonale (clonal\_rate) affecte à la précision de la reconnaissance, des expériences prennent différentes

valeurs de ce paramètre dans l'intervalle [100, 1200]. Les résultats sont présentés dans la figure 5.13 suivante :

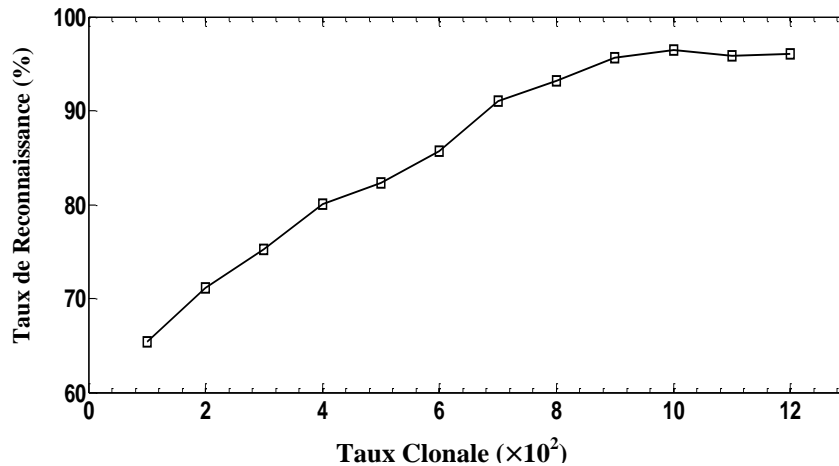


FIGURE 5.13 – Les affections du paramètre de taux clonale dans les performances du système

D'après la figure 5.13, les valeurs du paramètre *clonal\_rate* ont une influence importante sur la précision de la reconnaissance de l'algorithme. Cette influence est réduite de manière significative dans l'intervalle entier [900, 1200]. Ces résultats montrent que l'algorithme a une diversité caractéristique, qui est montée par la variation du paramètre *clonal\_rate*. Par conséquent, le principe de la fixation de paramètre *clonal\_rate* est d'assurer un nombre suffisant de nouveaux anticorps qui sont ajoutés à la population d'anticorps initial, ce qui garantira la diversité de la population d'anticorps.

En plus de cette caractéristique de diversité, nous pouvons caractériser l'AIRS aussi par :

- *La mémoire* : la mémoire de l'algorithme AIRS est dans le pool de cellules de mémoire MC développé par l'exposition à des données d'apprentissage (fait l'expérience).
- *L'adaptation* : l'adaptation se produit principalement dans l'ensemble d'ARB. Avec chaque nouvelle expérience, AIRS évolue une cellule mémoire candidate en réaction à cette expérience. Si cette cellule mémoire est d'une qualité suffisante, alors la structure de mémoire est adaptée pour l'inclure.
- *La prise de décision* : la décision initiale est lorsqu'on trouve la cellule mémoire la plus similaire à l'antigène d'apprentissage entrant. Cette cellule est utilisée en tant que progéniture d'un pool de cellules en évolution. Au cours de la classification, la décision primaire est basée sur les k cellules mémoire les plus semblables à l'élément de données en cours de classement.

Les résultats de performance de l'AIRS avec les approches KNN, Fuzzy\_KNN et la Distance Euclidienne sont présentés dans le tableau 5.2.

Le tableau 5.2 montre qu'avec l'introduction de règles floues et fonctions d'appartenance dans la phase de décision du système, il a une amélioration importante dans les performances de classification. La classification de l'AIS\_Fuzzy\_KNN donne des meilleurs résultats avec un taux de reconnaissance 96,50 % pour les chiffres et 86,66 % pour les lettres.

En outre, il est important de noter que nous avons trouvé dans la base MNIST un


Tableau 5.2 – Performances du AIS avec les approches KNN, Fuzzy-KNN et la Distance Euclidienne.


Classificateurs	Taux de Reconnaissance (%)	
	Chiffres	Lettres
AIRS_ Distance Euclidienne	91.05	80.66
AIS_KNN (k=3)	95.75	84.66
AIS_Fuzzy_KNN (k=3)	96.50	86.66


certain nombre de chiffres très similaires appartenant aux différentes classes c'est-à-dire, ayant le même vecteur d'attributs mais étiqueté de manière différente, ce qui pose le problème de la confusion entre les différentes classes et ainsi diminuer le taux de reconnaissance.


Nous avons trouvé plusieurs chiffres mal classés à partir de la base de données MNIST, citons quelques exemples :


 sont similaires au chiffre 7 mais étiquetés en tant que chiffre 1.

 est similaire au chiffre 3 mais étiquetés en tant que chiffre 2.

 est similaire au chiffre 5 mais étiquetés en tant que chiffre 3.

 est similaire au chiffre 9 mais étiquetés en tant que chiffre 4.

 est similaire au chiffre 6 mais étiquetés en tant que chiffre 5.

 est similaire au chiffre 9 mais étiquetés en tant que chiffre 5.

La plupart des classifieurs consacrés à résoudre le problème de chevauchement, basés sur le concept de répartition, sont incapables de classer correctement un objet appartenant à différentes classes avec le même degré d'appartenance. Ce dernier sera classé de façon aléatoire. Ce cas est très fréquent dans la reconnaissance de caractères manuscrits, et peut survenir en particulier lorsque les caractéristiques des objets sont quasi similaires et par conséquent conduisent à la saturation de la base de connaissances. Pour traiter ces cas, le concept qualitatif est introduit, avec la prise de décision optimale.

### 5.4.3 Évaluation de l'AIS\_QRBS

La phase de classification consiste à la sélection des valeurs optimales pour le réglage des paramètres du classifieur qui va générer le meilleur taux de reconnaissance. Pour

déterminer ces valeurs et par conséquent, obtenir des résultats expérimentaux optimaux, nous testons l'influence et l'efficacité de l'AIS dans le processus en changeant différents paramètres.

La meilleure performance de classification par AIS\_QRBS : 98,00 % a été obtenue avec une réduction des données de 17,65 % sur les chiffres, et 96,66 % a été obtenue avec une réduction des données de 19,42 % sur les lettres.

Les résultats obtenus par l'approche qualitative sans et avec l'introduction de l'AIS sur l'ensemble d'apprentissage, de la base MNIST, sont illustrés dans les deux tableaux suivants :

Tableau 5.3 – Résultats de la classification de la base MNIST avec l'approche qualitative

SEUIL	PAS				
	10	20	30	40	50
0.99	NBR. 3520 NBRO. 2653 TAUX. 86%	NBR. 6720 NBRO. 4771 TAUX. 94%	NBR. 9920 NBRO. 6599 TAUX. 94.5%	NBR. 13120 NBRO. 8132 TAUX. 97%	NBR. 16320 NBRO. 9601 TAUX. 97 %
0.95	NBR. 4510 NBRO. 2574 TAUX. 87%	NBR. 10710 NBRO. 4636 TAUX. 93.5%	NBR. 18910 NBRO. 6413 TAUX. 94.5%	NBR. 29110 NBRO. 7901 TAUX. 97 %	NBR. 41310 NBRO. 9554 TAUX. 97.5%
0.9	NBR. 16500 NBRO. 2503 TAUX. 87.5%	NBR. 29110 NBRO. 4528 TAUX. 94.5%	NBR. 98700 NBRO. 6226 TAUX. 95%	NBR. 306900 NBRO. 7732 TAUX. 97.5%	NBR. 1341300 NBRO. 9341 TAUX. 97.5%
0.85	NBR. 1613480 NBRO. 2476 TAUX. 87.5%	NBR. 40846680 NBRO. 4406 TAUX. 93.5%	NBR. 2.8629+8 NBRO. 6093 TAUX. 96.5%	NBR. 1.1586+9 NBRO. 7511 TAUX. 97.5%	NBR. 3.4503+9 NBRO. 93489 TAUX. 97%

**NB.**

**NBR** : Nombre de règles générées.

**NBRO** : Nombre de règles optimisées.

**TAUX** : Taux de classification.

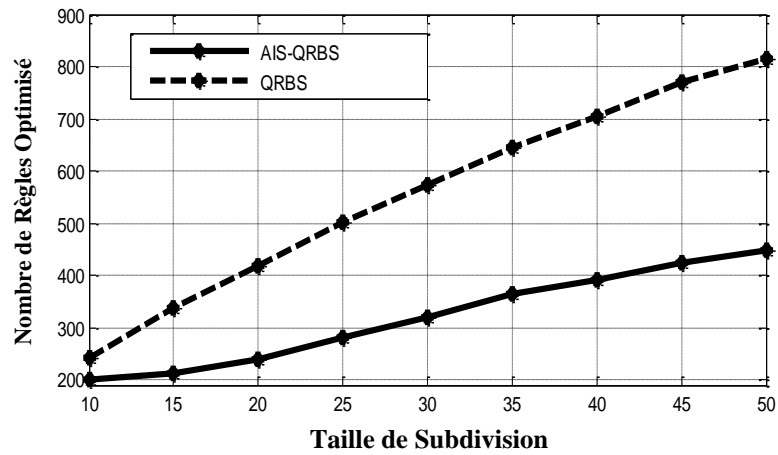
**PAS** : Cardinal de la subdivision.

Tableau 5.4 – Résultats de la classification de la base MNIST avec la méthode AIS\_QRBS

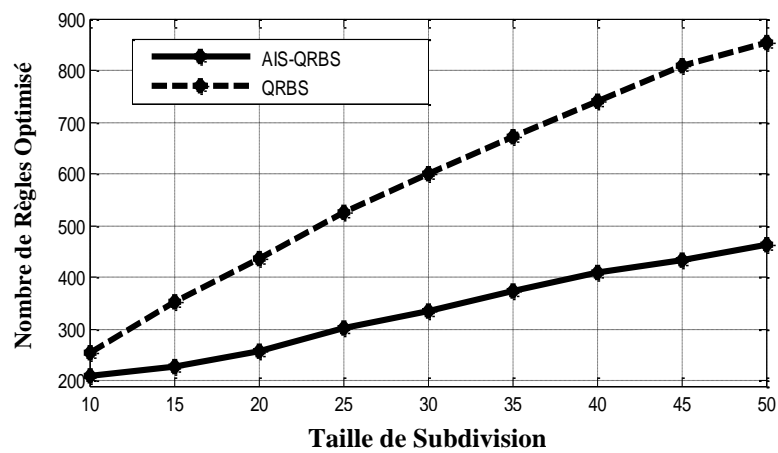
SEUIL	PAS				
	10	20	30	40	50
0.99	NBR. 3200 NBRO. 281 TAUX. 89%	NBR. 6410 NBRO. 465 TAUX. 96%	NBR. 9600 NBRO. 677 TAUX. 96.5%	NBR. 12800 NBRO. 840 TAUX. 97.5%	NBR. 16000 NBRO. 966 TAUX. 97.5%
0.95	NBR. 4000 NBRO. 263 TAUX. 89.5%	NBR. 7100 NBRO. 453 TAUX. 95.5%	NBR. 18000 NBRO. 627 TAUX. 96%	NBR. 28000 NBRO. 776 TAUX. 97.5 %	NBR. 40000 NBRO. 927 TAUX. 98 %
0.9	NBR. 12900 NBRO. 253 TAUX. 90%	NBR. 76600 NBRO. 437 TAUX. 96.5%	NBR. 278700 NBRO. 600 TAUX. 96.5%	NBR.651600 NBRO. 742 TAUX. 98%	NBR.1264500 NBRO. 922 TAUX. 97.5%
0.85	NBR.102800 NBRO. 243 TAUX. 90%	NBR.1025500 NBRO. 410 TAUX. 95.5%	NBR.8108400 NBRO. 574 TAUX. 96.5%	NBR.25611200 NBRO. 708 TAUX . 97.5%	NBR.62514000 NBRO. 814 TAUX. 97%

L'analyse des valeurs figurant dans les tableaux 5.3 et 5.4 montre que le taux de classification est d'autant meilleur lorsque la valeur de corrélation est élevée, quelque soit la valeur du pas de discrétisation. Le nombre de règles optimisé, pour un seuil de corrélation donné, augmente au fur et à mesure avec l'augmentation du pas de discrétisation. Le meilleur taux de classification (98%) est obtenu pour deux valeurs différentes du seuil de corrélation (0.9 et 0.95) dans le tableau 5.4, et le nombre de règles optimisé correspondant est égal à 742. Les temps de génération de règles et de classification ne seront pas pris en considération parce que le temps de traitement dépend non seulement du nombre de règles générées mais aussi de la vitesse de la machine.

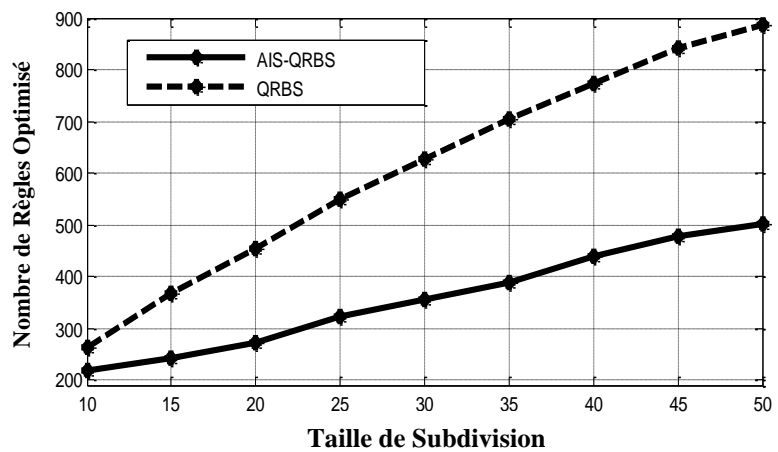
Les figures 5.14 et 5.15 représentent le nombre de règles générées et le taux de reconnaissance obtenus concernant les deux paramètres : seuil de corrélation (*correlation threshold*) et taille de subdivision (*subdivision size*).



(a)

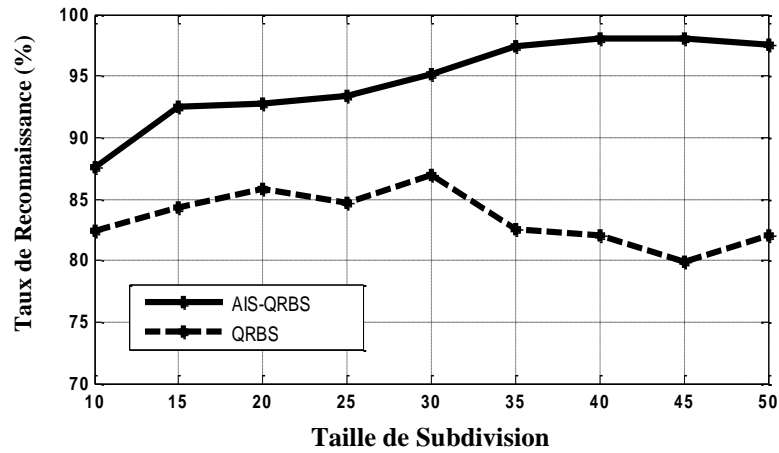


(b)

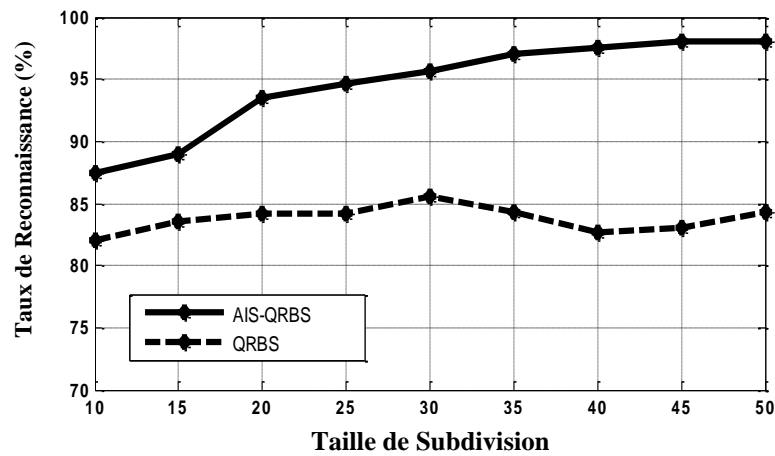


(c)

FIGURE 5.14 – La variation du nombre de règles optimisé sur la base de données MNIST. (a) Seuil de corrélation = 0,9, (b) Seuil de corrélation = 0,95, (c) Seuil de corrélation = 0,99



(a)



(b)

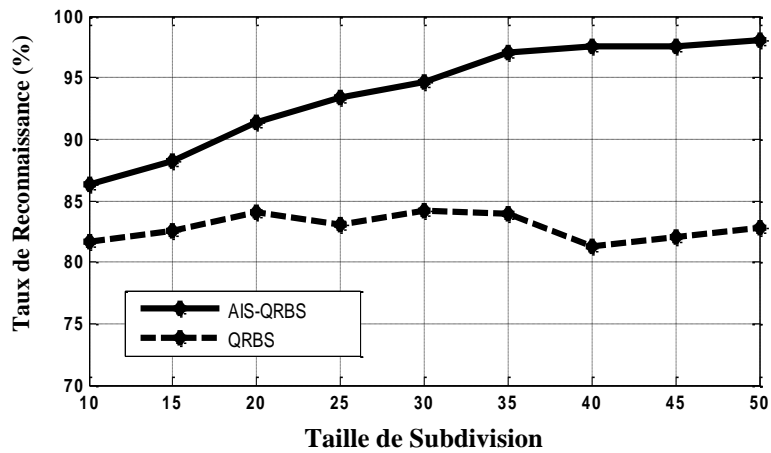


FIGURE 5.15 – Taux de reconnaissance sur la base de données MNIST. (a) Seuil de corrélation = 0,9, (b) Seuil de corrélation = 0,95, (c) Seuil de corrélation = 0,99

Au regard de ces deux figures, nous observons que le nombre de règles optimisés augmente lorsque la taille de subdivision augmente pour un seuil de corrélation donné. En outre, les taux de reconnaissance obtenus par l'approche proposée AIS-QRBS sont toujours meilleurs que celle obtenue par l'approche QRBS seule, et avec l'utilisation d'un nombre minimum de règles. Le taux de reconnaissance le plus élevé réalisé est 98,00 %

pour les chiffres lorsque la valeur du seuil de corrélation est égale à 0,9, et la taille de la subdivision est égale à 40 avec un nombre optimal de règles.

Le tableau 5.5 résume un ensemble de tests effectués sur les bases de données MNIST et lettres anglaises avec et sans l'introduction de la phase de réduction de données, et aussi avec et sans prise de décision optimale.

Tableau 5.5 – Une étude comparative sur l'approche qualitative avec et sans réduction de données et avec et sans prise de décision optimale .

Paramètres	QRBS		AIS_QRBS	
	Chiffres	Lettres	Chiffres	Lettres
Seuil de corrélation	0.85	0.9	0.9	0.9
Taille de subdivision	40	10	40	40
Nombre de Règles Optimisés	7511	10246	742	958
Taux de reconnaissance (%) sans prise de décision optimale	86.15	82.66	97.10	95.33
Taux de reconnaissance (%) avec prise de décision optimale	97.50	86.33	98.00	96.66

On peut observer à partir du tableau 5.5 que l'approche qualitative associée à AIS avec une prise de décision optimale s'avère plus performant que l'approche qualitative seule. Les meilleurs taux de reconnaissance réalisés sont 98,00% pour les chiffres et 96,66% pour les lettres, lorsque la valeur du seuil de corrélation est égale à 0,9 et la taille de subdivision est égale à 40. Le taux de reconnaissance est augmenté de 0,5% pour les chiffres et 13,33% pour les lettres. On remarque aussi que le nombre de règles a diminué avec l'introduction de l'AIS ; ce nombre a diminué environ 90% pour les deux chiffres et lettres.

Notre approche bénéficie de deux optimisations importantes conduisant à une amélioration significative. La première optimisation est réalisée pendant la phase de construction de l'ensemble d'apprentissage et elle permet de générer les règles appropriées minimales. La seconde optimisation est réalisée dans la prise de décision et elle permet de résoudre le problème de saturation de la base de connaissances et, par conséquent, fournir un taux de reconnaissance élevé.

#### 5.4.4 Comparaisons

Afin d'évaluer l'influence de l'AIS lorsqu'il est couplé avec d'autres classifieurs, nous allons présenter, dans ce qui suit, une comparaison entre les résultats des approches AIS\_KNN, AIS\_Fuzzy\_KNN, et AIS\_QRBS. Le tableau 5.6 résume les taux de reconnaissance obtenus par ces approches.

Tableau 5.6 – Performances de reconnaissance de caractères manuscrits avec AIS\_KNN, AIS\_Fuzzy\_KNN, et AIS\_QRBS.

Approches	Taux de Reconnaissance (%)	
	Chiffres	Lettres
AIS_KNN (k=3)	95.75	84.66
AIS_Fuzzy_KNN (k=3)	96.50	86.66
AIS_QRBS	98.00	96.66



Le tableau 5.6 montre que l'AIS associée à l'approche Fuzzy\_KNN donne des meilleurs résultats par rapport à la première configuration AIS\_KNN avec un taux de reconnaissance de 96,50 % pour les chiffres et 86,66 % pour les lettres quand  $k = 3$  (nombre optimal de voisins les plus proches constaté expérimentalement). Cependant, AIS\_QRBS donne de meilleurs scores par rapport à tous les classifieurs avec un taux de reconnaissance de 98,00 % pour les chiffres et 96,66 % pour les lettres.

Pour situer notre approche, une comparaison est faite avec des méthodes plus récentes sur les bases de données utilisées.

Les tableaux 5.7 et 5.8 sont des tableaux comparatifs pour chaque base de données.

Tableau 5.7 – Comparaison de l'AIS\_QRBS avec d'autres méthodes sur la base de données MNIST.

Approches	Ensemble de données Apprentissage-Test	Taux de Reconnaissance (%)
Dendritic Neurons with Morphological Learning [Hus 14]	10,000- 5000	90.26
GA-MLP [CHE 07]	60000-10000	96.15
KNN [BAB 14]	50000-5000	96.94
KFDA [JIA 14]	300-300	97.08
BPMLP [JAG 14]	60000-10000	98.00
<b>AIS_QRBS</b>	8000-2000	<b>98.00</b>
HCA [CHE 15]	60000-10000	98.50

Tableau 5.8 – Comparaison de l'AIS\_QRBS avec d'autres méthodes sur la base de données de lettres anglaise manuscrites

Approches	Taux de Reconnaissance (%)
MLPs_BA [NEB 13b]	92.20
NN [GAU 12]	95.38
<b>AIS_QRBS</b>	<b>96.66</b>

Comme on le voit dans le tableau 5.7, la performance de classification obtenue par AIS\_QRBS sur la base MNIST était parmi les meilleurs classifieurs rapportés dans la littérature avec un taux de reconnaissance satisfaisant (98,00 %). Cependant, il est un peu moins précis que l'algorithme HCA [CHE 15] avec un taux d'erreur de 1,5 %, parce que le HCA a utilisé un grand ensemble d'apprentissage. De plus, notre système ne traite pas les exemples mal classés.

D'après le tableau 5.8, nous remarquons que l'AIS\_QRBS atteint un meilleur score par rapport aux autres classifieurs avec un taux de reconnaissance égale à 96,66, et présente de bons résultats sur la base de données des lettres anglaises.

L'avantage de notre système par rapport aux autres consiste dans l'optimisation de la base de données dans la phase d'apprentissage en utilisant l'algorithme AIS, et la réduction de l'incertitude dans la phase de décision en utilisant l'approche QRBS. En effet, les études que nous avons effectuées sur les problèmes de classification ont conduits que la nature et la taille de la base de données ont une grande influence sur l'apprentissage du classifieur, et par conséquent, sur la qualité des résultats. La plupart des travaux qui existent dans la littérature ont négligé ce point crucial, et ils ont focalisé leurs intérêts sur

le changement et l'amélioration des classifieurs et des caractéristiques afin d'augmenter le taux de reconnaissance. Les tests effectués sur diverses bases de données ont montré que l'amélioration de la qualité de la base de données augmente de manière significative le taux de reconnaissance par rapport à une base de données non traitée.

### 5.5 Conclusion

Nous avons vu dans ce chapitre l'élaboration d'une nouvelle approche AIS\_QRBS fondée sur des règles bio-qualitatives. Cette approche est basée sur une combinaison de deux sous-systèmes principaux : le système immunitaire artificiel AIS et le système à base des règles qualitatives QRBS, pour faire face à la variabilité du style d'écriture et prendre en considération l'incertitude et l'imprécision des caractères.

L'AIS est utilisé comme un optimiseur. Il est appliqué pour réduire la dimensionnalité de l'ensemble d'apprentissage tout en gardant les meilleurs candidats. Les résultats de l'AIS sont soumis aux QRBS pour extraire des règles pondérées par des degrés qualitatifs qui permettent de classifier des objets inconnus. L'objectif de l'introduction de l'AIS dans la phase d'apprentissage est double : d'une part, il permet de réduire la taille des données d'apprentissage en conservant les meilleurs candidats. Comme on a vu précédemment, le meilleur taux est obtenu avec une réduction de données d'apprentissage de 17.65% pour les chiffres, et de 19,42 % pour les lettres. D'autre part, il permet de réduire le nombre de règles de QRBS en générant le minimum des règles appropriées. Le nombre de règles a diminué avec l'introduction de l'AIS à 6769 pour les chiffres, et à 9288 pour les lettres.

Le QRBS est utilisé comme un classifieurs. Le but de l'introduction de QRBS dans la phase de décision est également double : d'une part, il permet de traiter les cas d'imprécision et d'incertitude dans l'écriture en utilisant des règles qualitatives pour reconnaître les caractères. D'autre part, il permet d'optimiser la prise de décision finale à travers le processus de prise de décision optimale. En effet, lorsque l'AIS est couplé avec QRBS, la capacité de reconnaissance a été augmentée de façon significative de 0,5 % pour les chiffres et 13,33 % pour les lettres.

Nos expériences sur les bases de données utilisées montrent que notre approche donne de bons résultats en termes de précision et se compare favorablement avec d'autres travaux sur les mêmes bases de données.

# Conclusion Générale

## Conclusion

Le travail présenté dans cette thèse s'inscrit dans le cadre de la reconnaissance hors ligne de l'écriture manuscrite. La diversité et la richesse dans ce domaine qualifié de vaste, nous ont obligés à restreindre nos recherches aux phases d'apprentissage et classification pour un système de reconnaissance de caractères manuscrits (chiffres et lettres). Afin de résoudre cette problématique de manière efficace, nous avons proposé une nouvelle approche dite bio-qualitative consacrée aux améliorations apportées par une méthode bio-inspirée à une approche qualitative.

L'approche proposée a été étudiée et les résultats ont été discutés dans le but d'améliorer les performances du système de reconnaissance de caractères manuscrits en termes de précision et de décision. Pour ce faire, nous avons examiné les résultats obtenus selon différents types de caractères manuscrits et différents types de paramètres.

Dans cette thèse, nous avons proposé trois contributions :

La première contribution expose le potentiel et la capacité de l'algorithme AIS dans la reconnaissance de caractères manuscrits, dans l'adaptation aux variations de formes de caractères, et aussi dans la généralisation de l'ensemble d'apprentissage. Les caractères sont présentés sous forme d'un vecteur de caractéristiques structurelles et statistiques, notamment : les moments de Hu, les histogrammes de projection et de transition, la densité, les points d'extrémité, les points de croisement et les occlusions. Nous avons décrit comment la méthode bio-inspirée AIS peut être employée pour résoudre le problème de la reconnaissance de caractères manuscrits tout en examinant leurs performances. Cette méthode a montré son efficacité dans la phase d'apprentissage par son pouvoir de généraliser l'ensemble de données et afin d'extraire les meilleurs candidats. Bien que cette méthode soit une réussite en termes de généralisation, leur inconvénient majeur réside dans la phase de décision. La décision dans la plupart des AISs est achevée par l'approche de K plus proche-voisin. Cependant cette approche n'a pas la possibilité d'assigner correctement un objet à une classe spécifique lorsque ce dernier a la même valeur de similarité pour plus d'une classe. Dans le cas où l'ensemble de données chauveche ou les caractéristiques sont presque similaires, la décision est prise de façon aléatoire. À cet effet et pour surmonter ces limitations, nous avons introduit le concept flou dans la phase de décision intégrée dans l'approche Fuzzy-KNN. L'approche Fuzzy-KNN assure que les affectations arbitraires ne sont pas prises. La combinaison AIS\_Fuzzy\_KNN a apporté des améliorations des performances en termes du taux de reconnaissance relativement aux AIS de base.

La deuxième contribution permet d'exposer cette amélioration. Due à la nature des bases de données utilisées, la présence des cas où les exemples d'apprentissage restent

chevauchés ou bien mal étiquetés permet fortement de dégrader la précision de notre approche, et par conséquent, augmenter la difficulté des décisions qui en découlent du système. L'approche qualitative a pour but de surmonter cette difficulté afin de traiter l'incertitude dans la phase de décision.

La troisième contribution est consacrée à une combinaison de l'AIS avec une approche à base des règles qualitative QRBS. Cette nouvelle approche AIS-QRBS permet d'abord la réduction de données en optimisant et en améliorant la qualité des données d'apprentissage par la méthode AIS. Le résultat est considéré comme une base de référence contenant les meilleurs exemples représentatifs. Cette base de référence est soumise par la suite à l'approche qualitative QRBS. Cette dernière permet d'extraire les règles pondérées par des degrés qualitatifs pour classifier des objets inconnus. Ces règles permettent à la fois de traiter et corriger l'imprécision et l'incertitude dans les caractères donnés, et d'optimiser la prise de décision finale en introduisant le processus de prise de décision optimale. En effet, lorsque l'AIS est couplé avec le QRBS, la capacité de reconnaissance a été augmentée de manière significative pour les deux cas : chiffres et lettres. En outre, l'introduction de l'AIS dans cette approche qualitative permet d'optimiser les règles générées par QRBS.

La reconnaissance de caractères manuscrits à base d'une approche bio-qualitative offre une solution plus avancée par rapport aux autres méthodes que nous avons exposées, car elle procure une fiable exploration et une vraie exploitation de l'espace des caractères d'apprentissage. Elle offre aussi des améliorations en termes de précision, voire de qualité de classification, quand elle est acquise à partir de méthode d'apprentissage et de généralisation efficace.

L'approche proposée est vérifiée sur les chiffres arabes manuscrits, et sur les lettres anglaises manuscrites. À travers l'étude présentée qui a mis en valeur les capacités intéressantes de notre approche, nous concluons que le travail présenté offre une solution très satisfaisante au problème de la reconnaissance des caractères manuscrits. La comparaison avec d'autres méthodes récentes montre que nos résultats sont meilleurs et suffisants pour confirmer le potentiel de notre approche.

## Perspectives

Ce travail de recherche dégage quelques perspectives, notamment :

- Examiner le potentiel de l'AIS\_QRBS pour reconnaître des chaînes de caractères manuscrits non contraints.
- Proposer une nouvelle et bonne méthode de segmentation dans le cas de reconnaissances des chaînes de caractères manuscrits. Nous avons déjà fait les premières expériences de cette suggestion.
- Examiner le potentiel de l'AIS\_QRBS avec l'utilisation des autres bases de données déséquilibrées.
- La méthode proposée nécessite une initialisation de certains paramètres donnés par l'utilisateur : la sélection des paramètres de l'AIS et QRBS, respectivement. Dans ce contexte, il sera utile de développer une conception expérimentale ou de proposer des techniques afin de découvrir la meilleure combinaison de paramètres et de les fixer automatiquement.

- Étudier l'efficacité de l'AIS\_QRBS en utilisant un grand ensemble de caractéristiques, afin d'améliorer le taux de reconnaissance sans une augmentation significative du temps de classification.

## Systeme Immunitaire Artificiel (AIS)

Dans cette annexe, nous présentons en détail les procédures importantes du processus d'apprentissage de l'AIS [WAT 01 ; WAT 85]. Pour cette raison, nous mettons les conventions de notation suivantes :

- $MC$  : représente l'ensemble des cellules mémoire et  $mc$  représentent un membre individuel de cet ensemble.
- $Ag.c$  : représentent la classe d'un antigène donné  $Ag$ , où  $Ag.c \in C = \{1, 2, \dots, nc\}$ , et  $nc$  est le nombre de classes dans l'ensemble de données.
- $Ag.f$  et  $mc.f$  : représentent, respectivement, le vecteur de caractéristiques d'un antigène et d'une cellule mémoire donnés,  $Ag$  et  $mc$ .
- $Ag.f_i$  : représente la valeur de la  $i^{ème}$  caractéristique dans  $Ag.f$ .
- $AB$  : représente l'ensemble des ARBs, ou la population des cellules existantes. En outre, un  $Ac$  représente un seul ARB où  $Ac \in AB$ .
- $MU$  : représente un ensemble de clones mutés d'ARBs.
- $Ac.c$  : représente la classe d'un ARB donné  $Ac$ , où  $Ac.c \in C = \{1, 2, \dots, nc\}$ ,
- $Ac.stim$  : représente le niveau de stimulation de l'ARB  $Ac$ .
- $Ac.resources$  : représente le nombre de ressources détenues par l'ARB  $Ac$ .
- $TotalNumRes$  : représente le nombre total de ressources autorisées par le système.

## A.1 Étape d'initialisation

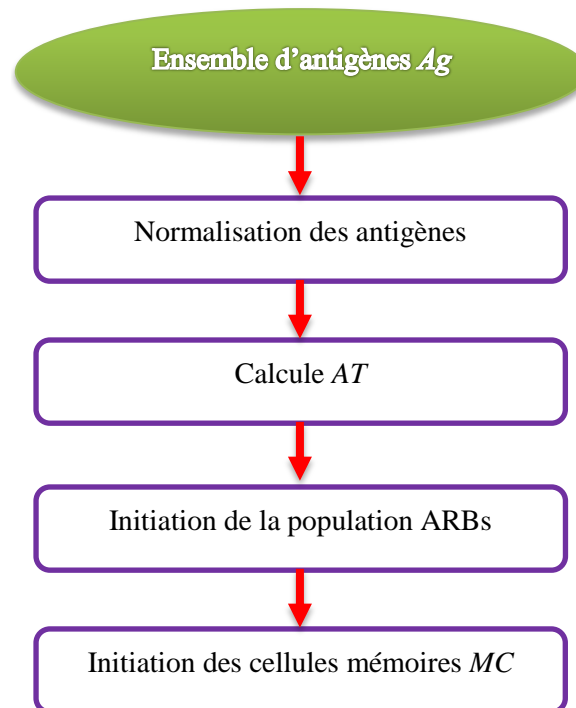


FIGURE A.1 – Étape d'initialisation

## A.2 Étape d'identification de cellule mémoire et génération des ARBs

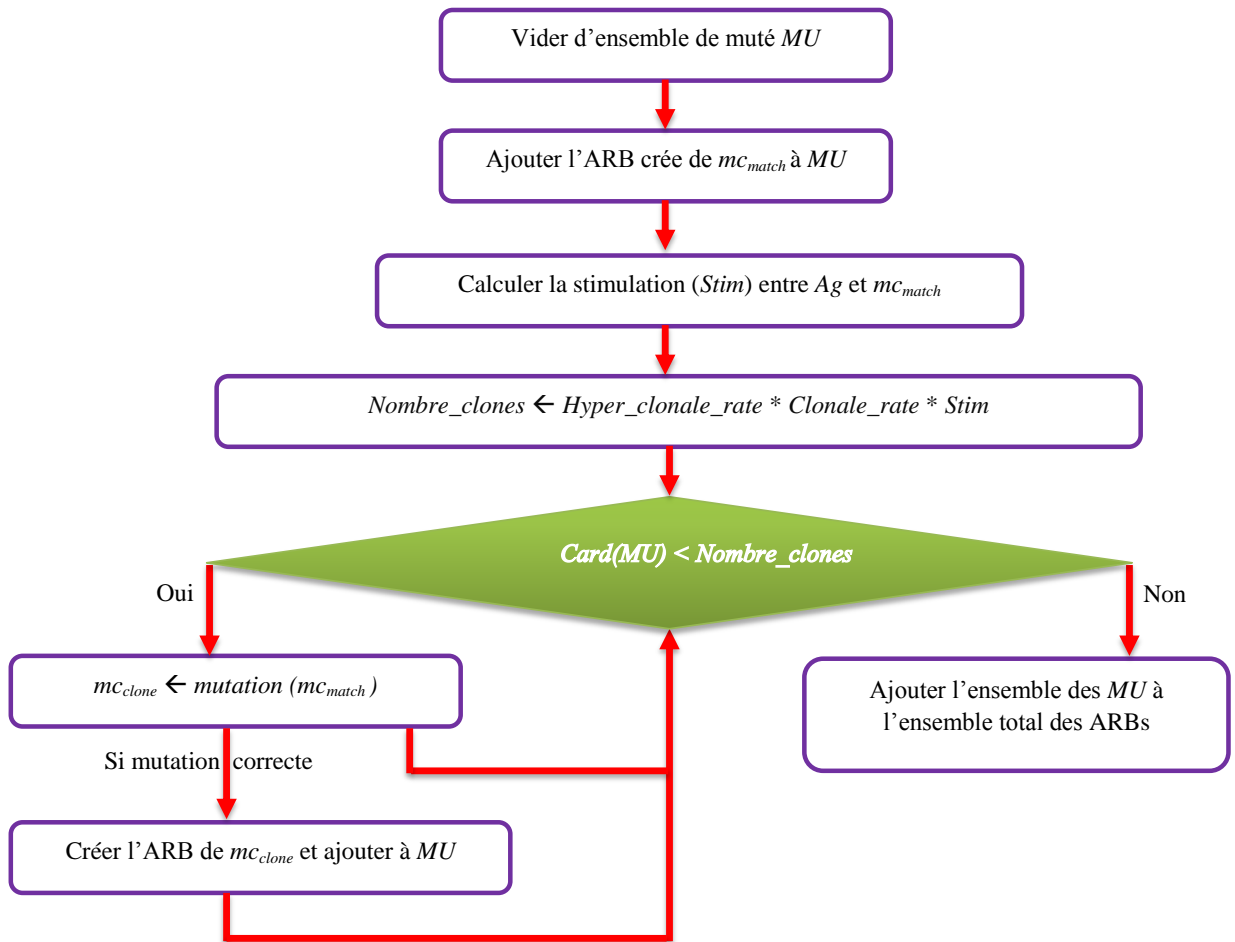


FIGURE A.2 – Étape d'identification de cellule mémoire et génération des ARBs

Dans l'algorithme A.1, la fonction  $prendARB(x)$  renvoie un ARB avec  $x$  comme anticorps de cet ARB.



**Algorithme A.1 : Génération des ARBs**

---

```

MU ← ∅
MU ← MU ∪ prendARB(mcmatch)
Stim ← Stimulation(Ag, mcmatch)
NumClones ← hyper_clonal_rate * clonal_rate * Stim
tant que |MU| < NumClones faire
    | mut ← faux
    | mcclones ← mcmatch
    | mcclones ← mutation(mcclone, mut)
    | si mut ≡ vrai alors
    | | MU ← MU ∪ prendARB(mcmatch)
    | fin
fin
AB ← AB ∪ MU

```

---

Dans la fonction de mutation, la fonction *drandom*() renvoie une valeur aléatoire dans l'intervalle [0, 1], et (*lrandom*() *mod* *nc*) renvoie une valeur aléatoire dans l'intervalle {0, *nc*}

**Algorithme A.2 : Fonction de mutation**

---

```

Mutation (x, b, seuil_change)
pour chaque x.fi dans x.f faire
    | Change ← drandom()
    | Change_to ← drandom()
    | si (change < mutation_rate) alors
    | | x.fi ← Change_to * valeur de normalisation
    | | b ← vrai
    | | nbi ← nbi + 1
    | fin
fin
si (b ≡ vrai) alors
    | Change ← drandom()
    | Change_to ← (lrandom() mod nc)
    | si (change < mutation_rate) et nbi ≥ seuil_change alors
    | | x.c ← Change_to
    | fin
fin
Retourner x

```

---

### A.3 Étape d'allocation des ressources et compétition

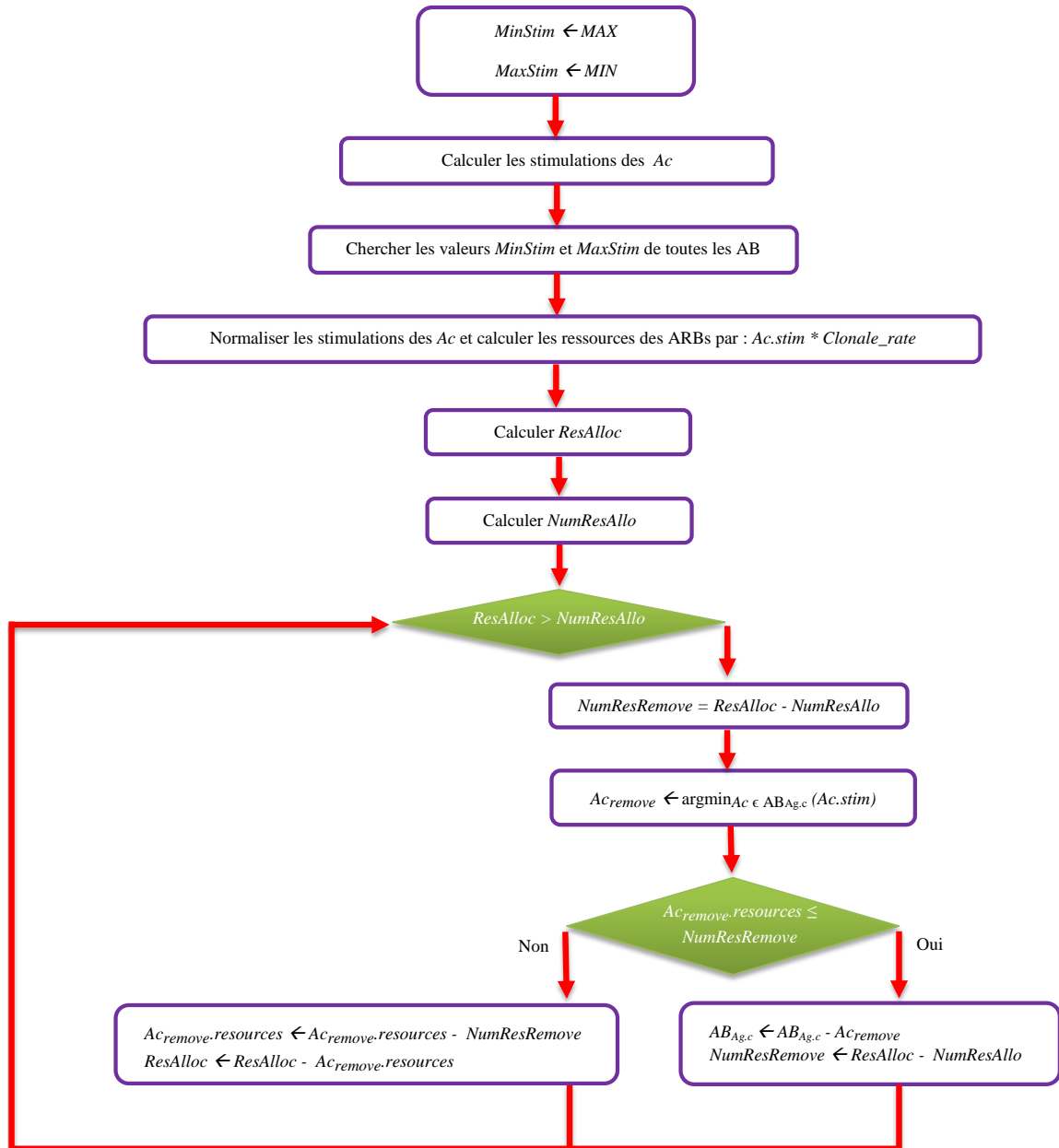


FIGURE A.3 – Étape d'allocation des ressources et compétition

---

**Algorithme A.3 :** Stimulation, Allocation de Recourse, Suppression d'ARB

---

```

MinStim ← MAX
MaxStim ← MIN
pour chaque ( $Ac \in AB$ ) faire
    Stim ← Stimulation(Ag, Ac)
    si Stim < MinStim alors
        | MinStim ← Stim
    fin
    si (Stim > MaxStim) alors
        | MaxStim ← stim
    fin
    Ac.Stim ← Stim
fin
pour chaque ( $Ac \in AB$ ) faire
    si ( $Ac.c \equiv Ag.c$ ) alors
        |  $Ac.stim \leftarrow \frac{Ac.stim - MinStim}{MaxStim - MinStim}$ 
    sinon
        |  $Ac.stim \leftarrow 1 - \frac{Ac.stim - MinStim}{MaxStim - MinStim}$ 
    fin
    Ac.recources ← Ac.stim * Clonale_rate
fin
i ← 1
tant que ( $i \leq nc$ ) faire
    ResAlloc ←  $\sum_{j=1}^{AB_i} Ac_j.ressources, Ac_j \in AB_i$ 
    si ( $I \equiv Ag.c$ ) alors
        | NumResAllo ←  $\frac{TotalNumRes}{2}$ 
    sinon
        | NumResAllo ←  $\frac{TotalNumRes}{2*(nc-1)}$ 
    fin
    tant que ( $ResAlloc < NumResAllo$ ) faire
        NumResRemove ← ResAlloc - NumResAllo
        AcRemove ← Argmin $_{Ac \in AB_i}(Ac.stim)$ 
        si AcRemove.ressources ≤ NumResRemove alors
            |  $AB_i \leftarrow AB_i - Ac_{remove}$ 
            | ResAlloc ← ResAlloc - AcRemove.ressources
        sinon
            | AcRemove.ressources ← AcRemove.ressources - NumResRemove
            | ResAlloc ← ResAlloc - AcRemove.ressources
        fin
    fin
    i ← i + 1
fin

```

---

## A.4 Étape de mise à jours de l'ensemble de cellules mémoires MC

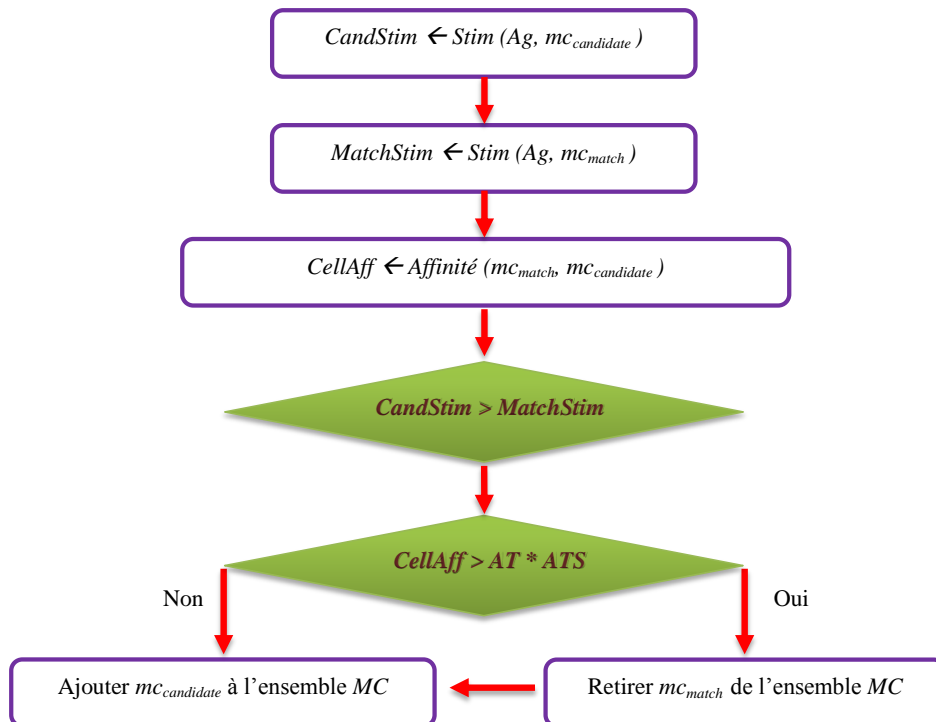


FIGURE A.4 – Étape de mis à jours de l'ensemble de cellules mémoires MC

## A.5 Aspects computationnels de l'algorithme AIS

Le calcul de la complexité de notre proposition est basé sur l'analyse formelle de ces ressources (temps ou espace) nécessaires pour son exécution.

Rappelons-nous que le calcul de temps d'exécution seul ne peut être pas un facteur d'efficacité pour n'importe quelle proposition, car les machines et les langages de programmation sont évolués au fil du temps. Le nombre d'instructions exécutées dans un algorithme reste le seul facteur crédible, ce qui revient au calcul de la complexité de l'algorithme.

Le temps requis par un algorithme peut mesurer en comptant le nombre maximum d'instructions exécutées, qui est proportionnel au nombre maximum de fois que chaque boucle est exécutée. Le temps nécessaire pour exécuter un algorithme est également en fonction de donnée d'entrée. Au lieu de nous traiter directement avec la donnée d'entrée, nous utilisons des paramètres qui caractérisent sa taille.

Dans ce qui suit, nous proposons un calcul approximatif de l'algorithme AIS dans le pire des cas, en utilisant des paramètres qui caractérisent les calculs effectués, comme :

- la dimension de répertoire  $Ag$  en disposition  $n \times l$ , où  $n$  représente le nombre de données d'apprentissage (antigènes) et  $l$  représente la longueur de chaque élément,
- le nombre  $N_1$  du pool MC initiale,
- le nombre  $N_2$  de la population ARB initiale,
- le nombre  $N_3$  de clones générés à partir de  $mc_{match}$ ,
- le nombre total de clones  $N_c$ ,

- la valeur d'affinité  $f$ ,
- le nombre  $N_r$  de clones survivants.

La complexité de calcul de l'algorithme AIS est résumée dans le tableau A.1.

Tableau A.1 – Complexité computationnelle de l'algorithme AIS

<b>Opérations</b>	<b>Pire cas d'exécution</b>
(1) Initialisation	$O(N_1 \times l) + O(N_2 \times l)$
(1) Génération des ARBs	$O(N_1 \times f) + O(N_3 \times l)$
(1) Allocation des ressources et compétition	$O(N_c \times l) + O(N_r \times l)$
(1) Mise à jours de MC	$O(N_r \times l)$
Tout	$O(n(N_c \times l))$

# Bibliographie

- [ABA 08] Gheith A. ABANDAH, Khaled S. YOUNI et Mohammed Z. KHEDHER. « Handwritten Arabic character recognition using multiple classifiers based on letter form ». In : *Proceedings of the 5th IASTED International Conference on Signal Processing, Pattern Recognition, and Applications (SPPRA'08)*. 2008, p. 128–133.
- [ABÈ09] Riad ABÈS, Charles-Antoine DUTERTRE et Jean-Luc TEILLAUD. « Les anticorps : mieux les connaître pour mieux s' en servir ». In : *MS. Médecine sciences* 25.12 (2009), p. 1011–1019.
- [AIC 10] Uwe AICKELIN et al. *The Danger Project*. 2010.
- [AKR 98] Hamza AKROUM et Latifa HAMAMI. « Prétraitement et segmentation d'images de documents ». In : *Signal and image processing, NWSIP'98, Sidi Bel Abbes-Algérie. Proceedings. first national Workshop on*. 1998, p. 24–27.
- [ALH 07] Rami AL-HAJJ MOHAMAD. « Reconnaissance hors ligne de mots manuscrits cursifs par l'utilisation de systèmes hybrides et de techniques d'apprentissage automatique ». Thèse de doct. Paris, ENST, 2007.
- [ALL 14] Pierre ALLAIN. *Les médicaments : leurs cibles, leurs effets bénéfiques et indésirables : pharmacologie*. CDM éd., 2014.
- [ALM 02a] Somaya AL-MA'ADEED, Dave ELLIMAN et Colin HIGGINS. « A data base for Arabic handwritten text recognition research ». In : *Frontiers in Handwriting Recognition, 2002. Proceedings. Eighth International Workshop on*. IEEE. 2002, p. 485–489.
- [ALM 02b] Somaya AL-MA'ADEED, Colin HIGGINS et Dave ELLIMAN. « Recognition of off-line handwritten Arabic words using hidden Markov model approach ». In : *Pattern Recognition, 2002. Proceedings. 16th International Conference on*. T. 3. IEEE. 2002, p. 481–484.
- [ALR 06] Hasan AL-RASHAIDEH. « Preprocessing phase for Arabic word handwritten recognition ». In : *Institut d'informatique et autimatisme* 6.1 (2006), p. 11–19.
- [AMR 12] Mustapha AMROUCH. « Reconnaissance de caractères imprimés et manuscrits, textes et documents basée sur les modèles de Markov cachés ». Thèse de doct. Université Ibn Zohr, Agadir, Morocco, 2012.

- [ARR 06] Denis ARRIVAULT. « Apport des graphes dans la reconnaissance non-contrainte de caractères manuscrits anciens ». Thèse de doct. Université de Poitiers, 2006.
- [AUG 00] Emmanuel AUGUSTIN et al. « Le système de reconnaissance de documents A2IA check reader ». In : *CIFED'2000. Colloque International Francophone sur l'Écrit et le Document*. PPUR presses polytechniques. 2000, p. 241.
- [AUG 01] Emmanuel AUGUSTIN. « Reconnaissance de mots manuscrits par systèmes hybrides : Réseaux de neurones et modèles de Markov cachés ». Thèse de doct. Paris 5, 2001.
- [AYA 04] Nedjem Eddine AYAT. « Sélection automatique de modèle dans les machines à vecteurs de support : application à la reconnaissance d'images de chiffres manuscrits ». Thèse de doct. École de technologie supérieure, Université du Québec, 2004.
- [AZI 10] Nabiha AZIZI, Nadir FARAH et Mokhtar SELLAMI. « Off-line handwritten word recognition using ensemble of classifier selection and features fusion ». In : *Journal of Theoretical and Applied Information Technology* 14 (2010).
- [BAB 14] U. Ravi BABU, Y. VENKATESWARLU et Aneel Kumar CHINTHA. « Handwritten digit recognition using K-nearest neighbour classifier ». In : *Computing and Communication Technologies (WCCCT), 2014 World Congress on*. IEEE. 2014, p. 60–65.
- [BAH 01] Claus BAHLMANN et Hans BURKHARDT. « Measuring HMM similarity with the Bayes probability of error and its application to online handwriting recognition ». In : *Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on*. IEEE. 2001, p. 406–411.
- [BAL 06] Gregory R. BALL, Sargur N. SRIHARI et Harish SRINIVASAN. « Segmentation-Based And Segmentation-Free Methods for Spotting Handwritten Arabic Words ». In : *Tenth International Workshop on Frontiers in Handwriting Recognition*. Université de Rennes 1. Suvisoft, 2006.
- [BEL 01] Abdel BELAÏD. « Reconnaissance automatique de l'écriture et du document ». In : *Pour la science* (2001), 22 p.
- [BEL 10] Saoussen BEL HADJ KACEM, Amel BORGHI et Moncef TAGINA. « Approximate Reasoning based on Linguistic Modifiers in a Learning System ». In : *Proceedings of the Fifth International Conference on Software and Data Technologies*. 2010, p. 431–437.
- [BEL 92] Abdel BELAÏD et Yolande BELAÏD. *Reconnaissance des Formes : Méthodes et applications*. InterÉditions, 1992, p. 429.
- [BEN 00] Najoua BEN AMARA, Abdel BELAÏD et Nouredine ELLOUZE. « Utilisation des modèles markoviens en reconnaissance de l'écriture arabe : Etat de l'art ». In : *Colloque international francophone sur l'écrit et le document*. 2000, p. 181–191.
- [BEN 02] Nadia BENAHMED. *Optimisation de réseaux de neurones pour la reconnaissance de chiffres manuscrits isolés : Sélection et pondération des primitives par algorithmes génétiques*. 2002.

- [BEN 03a] Esma BENDIAB. *Recalage d'images par systèmes immunitaires artificiels*. 2003.
- [BEN 03b] Ameer BENSEFIA, Thierry PAQUET et Laurent HEUTTE. « Documents manuscrits et recherche d'information ». In : *Document numérique 7.3* (2003), p. 47–60.
- [BEN 04] Ameer BENSEFIA. « Analyse de documents manuscrits : identification et vérification du scripteur ». Thèse de doct. Rouen, 2004.
- [BEN 07] Abdallah BENOURETH. « Reconnaissance de Mots Arabes Manuscrits par Modèles de Markov Cachés à Durée d'at Explicite ». Thèse de doct. Université Badji Mokhtar, Annaba, 2007.
- [BEN 11] Esma BENDIAB. « Vie artificielle, analyse, traitement et fouille de données ». Thèse de doct. Université Mentouri, Constantine, 2011.
- [BEN 94] Yoshua BENGIO et Yann LE CUN. « Word normalization for online handwritten word recognition ». In : *Pattern Recognition, 1994. Vol. 2-Conference B : Computer Vision & Image Processing., Proceedings of the 12th IAPR International. Conference on*. T. 2. IEEE. 1994, p. 409–413.
- [BER 86] John BERNSEN. « Dynamic thresholding of gray-level images ». In : *International Conference on Pattern Recognition*. 1986.
- [BIA 06] Fadi BIADSY, Jihad EL-SANA et Nizar HABASH. « Online Arabic Handwriting Recognition Using Hidden Markov Models ». In : *Tenth International Workshop on Frontiers in Handwriting Recognition*. Université de Rennes 1. Suvisoft, 2006.
- [BIS 06] Christopher M. BISHOP. *Pattern recognition and Machine Learning*. Springer, 2006.
- [BIS 95] Christopher M. BISHOP. *Neural Networks for Pattern Recognition*. New York, USA : Oxford University Press, Inc., 1995.
- [BON 99] Eric BONABEAU, Marco DORGIO et Guy THÉRAULAZ. *Swarm intelligence : from natural to artificial systems*. Oxford University Press., 1999.
- [BOR 98] Amel BORGHI. « Supervised classification by automatic rules generation ». In : *the Fourth World Congress on Expert Systems, Mexico City, Mexico, March 16, 1998*. Instituto Tecnológico y de Estudios Superiores de Monterrey, ITESM. 1998, p. 561–568.
- [BOR 99] Amel BORGHI BEN-BOUZID. « Apprentissage supervisé par génération de règles : le système SUCRAGE ». Thèse de doct. Université Paris 6, France, 1999.
- [BOU 09] Fateh BOUGAMOUZA. « Contribution à la reconnaissance automatique de l'écriture manuscrite arabe, application sur les montants littéraux des chèques ». Thèse de doct. Université Mentouri -Constantine, 2009.
- [BOU 98] Mohammed BOUAYAD. « Prétopologie et reconnaissance des formes ». Thèse de doct. Institut national des sciences appliquées, Lyon, 1998.
- [BRA 03] Anja BRAKENSIEK, Jorg ROTTLAND et Gerhard RIGOLL. « Confidence measures for an address reading system ». In : *Seventh international conference on document analysis and recognition, ICDAR2003*. IEEE. 2003, p. 294–298.



- [BRI 04] Alceu de S. BRITTO et al. « Foreground and background information in an HMM-based method for recognition of isolated characters and numeral strings ». In : *Frontiers in Handwriting Recognition, 2004. IWFHR-9 2004. Ninth International Workshop on*. IEEE. 2004, p. 371–376.
- [BUR 01] Gilles BUREL. *Introduction au traitement d'images - Simulations sous Matlab*. HERMES SCIENCE PUBLICATIONS, 2001, p. 224.
- [CAI 99] Jinhai CAI et Zhi-Qiang LIU. « Integration of structural and statistical information for unconstrained handwritten numeral recognition ». In : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 21.3 (1999), p. 263–270.
- [CAR 99] Claude CARIOU et al. « Reconnaissance de formes en multi-orientation et multi-échelle sur documents techniques ». In : *17eme Colloque sur le traitement du signal et des images, FRA, 1999*. GRETSI, Groupe d'Etudes du Traitement du Signal et des Images. 1999.
- [CAS 87] Glenn L. CASH et Mehdi HATAMIAN. « Optical character recognition by the method of moments ». In : *Computer Vision, Graphics, and Image Processing* 39.3 (1987), p. 291–310.
- [CAS 96] Richard G. CASEY et Eric LECOLINET. « A survey of methods and strategies in character segmentation ». In : *IEEE transactions on pattern analysis and machine intelligence* 18.7 (1996), p. 690–706.
- [CHA 06a] Yahia CHABANE. « Elaboration d'un contrôleur adaptatif pour un robot mobile basé sur les réseaux immunitaires artificiels ». Mém.de mast. Université Saad Dahlab de Blida, 2006.
- [CHA 06b] Clément CHATELAIN. « Extraction de séquences numériques dans des documents manuscrits quelconques ». Thèse de doct. Université de Rouen, 2006.
- [CHE 07] Mohamed CHERIET. « Strategies for visual arabic handwriting recognition : Issues and case study ». In : *Signal Processing and Its Applications, 2007. ISSPA 2007. 9th International Symposium on*. IEEE. 2007, p. 1–6.
- [CHE 15] Jie CHEN, Zhao SHU et Zhang YANPING. « Hierarchical Covering Algorithm ». In : *Tsinghua Science and Technology* 19.1 (2015), p. 76–81.
- [CHE 98a] H. D. CHENG, Jim-Rong CHEN et Jiguang LI. « Threshold selection based on fuzzy c-partition entropy approach ». In : *Pattern recognition* 31.7 (1998), p. 857–870.
- [CHE 98b] Kwok-Wai CHEUNG, Dit-Yan YEUNG et Roland T CHIN. « A Bayesian framework for deformable pattern recognition with application to handwritten character recognition ». In : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20.12 (1998), p. 1382–1388.
- [CHI 12] Mohamed Amine CHIKH, Meryem SAIDI et Nesma SETTOUTI. « Diagnosis of diabetes diseases using an artificial immune recognition system2 (AIRS2) with fuzzy k-nearest neighbor ». In : *Journal of medical systems* 36.5 (2012), p. 2721–2729.
- [CHI 99] Y. C . CHIM, Ashraf A. KASSIM et Y. IBRAHIM. « Character recognition using statistical moments ». In : *Image and vision computing* 17.3 (1999), p. 299–307.

- [COE 07] Carlos A. COELLO COELLO, Gary B. LAMONT et David A. VAN VELDHUIZEN. *Evolutionary algorithms for solving multi-objective problems*. Springer Science & Business Media, 2007.
- [COR 12] David W. CORNE, Alan P. REYNOLDS et Eric BONABEAU. « Swarm intelligence ». In : *Handbook of Natural Computing*. Springer, 2012, p. 1599–1622.
- [DAS 03] Dipankar DASGUPTA, Zhou JI, Fabio A. GONZÁLEZ et al. « Artificial immune system (AIS) research in the last five years. » In : *IEEE Congress on Evolutionary Computation (1)*. 2003, p. 123–130.
- [DAS 06] Dipankar DASGUPTA. « Advances in artificial immune systems ». In : *IEEE computational intelligence magazine* 1.4 (2006), p. 40–49.
- [DAS 99] Dipankar DASGUPTA. *Artificial Immune Systems and Their Applications*. Springer Publishing Company, Incorporated, 1999.
- [DEC 00] Leandro Nunes DE-CASTRO et Fernando José VON ZUBEN. « Artificial immune systems : Part II—A survey of applications ». In : *FEEC/Univ. Campinas, Campinas, Brazil* (2000).
- [DEC 01] Leandro Nunes DE-CASTRO et Fernando José VON ZUBEN. « aiNet : an artificial immune network for data analysis ». In : *Data mining : a heuristic approach* 1 (2001), p. 231–259.
- [DEC 02a] Leandro Nunes DE-CASTRO et Jonathan TIMMIS. *Artificial immune systems : a new computational intelligence approach*. Springer Science & Business Media, 2002.
- [DEC 02b] Leandro Nunes DE-CASTRO et Fernando J. VON ZUBEN. « Learning and optimization using the clonal selection principle ». In : *IEEE transactions on evolutionary computation* 6.3 (2002), p. 239–251.
- [DEC 03] Leandro Nunes DE-CASTRO et Jonathan TIMMIS. « Artificial immune systems as a novel soft computing paradigm ». In : *Soft computing* 7.8 (2003), p. 526–544.
- [DEC 07] Leandro Nunes DE-CASTRO. « Fundamentals of natural computing : an overview ». In : *Physics of Life Reviews* 4.1 (2007), p. 1–36.
- [DEC 99] Leandro Nunes DE-CASTRO et Fernando José VON ZUBEN. « Artificial immune systems : Part I—basic theory and applications ». In : *Universidade Estadual de Campinas, Dezembro de, Tech. Rep* 210 (1999).
- [DEH 01] Mehdi DEGHAN et al. « Handwritten Farsi (Arabic) word recognition : a holistic approach using discrete HMM ». In : *Pattern Recognition* 34.5 (2001), p. 1057–1065.
- [DEN 06] Abdelhakim DENECHÉ. « Approches bio-inspirées pour la reconnaissance de formes ». Thèse de doct. Université mentouri , Constantine, 2006.
- [DEN 12] Li DENG. « The MNIST database of handwritten digit images for machine learning research ». In : *IEEE Signal Processing Magazine* 29.6 (2012), p. 141–142.
- [DEV 82] Pierre A. DEVIJVER et Josef KITTLER. *Pattern Recognition : A statistical approach*. London : Prentice-Hall, 1982.

- [DUB 90] Bernard DUBUISSON. *Diagnostic et reconnaissance des formes (Traité des nouvelles Technologies, Série diagnostic et Maintenance)*. Hermes Science Publications, 1990.
- [DUP 04] Xavier DUPRÉ. « Contributions à la reconnaissance de l'écriture cursive à l'aide des modèles de Markov cachés ». Thèse de doctorat dirigé par Stamon, Georges Informatique Paris 5 2004. Thèse de doct. Paris 5, 2004, 1 vol. (482 p.)
- [ELH 05] Ramy EL-HAJJ, Likforman-Sulem LAURENCE et Mokbel CHAFIC. « HMM-based Arabic Cursive Handwritten Recognition System ». In : *The RTST conference (Int'l. Conference on Research Trends in science and Technology)*. LAU University Beirut Lebanon. 2005.
- [ELY 96] Abdenaim EL-YACOUBI. « Modélisation Markovienne de l'écriture manuscrite Application à la reconnaissance des adresses postales ». Thèse de doct. 1996.
- [FAR 06] Nadir FARAH, Labiba SOUCI et Mokhtar SELLAMI. « Classifiers combination and syntax analysis for Arabic literal amount recognition ». In : *Engineering Applications of Artificial Intelligence* 19.1 (2006), p. 29–39.
- [FAR 09] Brahim FAROU. « Système hybride utilisant les réseaux de neurones et les modèles de Markov cachés pour la reconnaissance de mots manuscrits Arabe ». Thèse de doct. Université 8 mai 1945 de Guelma, 2009.
- [FAR 86] J. Doyne FARMER, Norman H. PACKARD et Alan S. PERELSON. « The immune system, adaptation, and machine learning ». In : *Physica D : Non-linear Phenomena* 22.1 (1986), p. 187–204.
- [FAV 98] John T. FAVATA, Sargur N. SRIHARI et Venu GOVINDARAJU. « Off-line handwritten sentence Recognition ». In : *In Proc. 6th International Workshop on Frontiers of Handwriting Recognition (IWFHR), Taegon-Korea*. 1998, p. 171–176.
- [FOR 98] Stephanie FORREST et al. « Self-nonsel discrimination in a computer ». In : *IEEE Symposium on Security and Privacy*. Oakland. 1998, p. 202–212.
- [FUJ 07] Hiromichi FUJISAWA. « Robustness design of industrial strength recognition systems ». In : *Digital Document Processing*. Springer, 2007, p. 185–212.
- [GAG 05] Christian GAGNE. « Algorithmes évolutionnaires appliqués à la reconnaissance des formes et à la conception optique ». In : *Université Laval* (2005).
- [GAS 05] Ibtissem GASMI, Hayet MEROUANI et Labiba SOUICI-MESLATI. « Combinaison de classifieurs ». In : *3rd International Conference : Sciences of Electronic, Technologies of Information and telecommunications SETIT2005, Tunisie*. 2005. ISBN : 9973-51-546-3.
- [GAU 09] Dinesh Dileep GAURAV. *Feature extraction for character recognition*. <http://www.mathworks.com/matlabcentral/fileexchange/24624-feature-extraction-for-character-recognition>. 2009.
- [GAU 12] Dinesh Dileep GAURAV et Renu RAMESH. « A feature extraction technique based on character geometry for character recognition ». In : *arXiv preprint arXiv :1202.3884* (2012).

- [GAY 14] P. GAYATHRI et Sonal AYYAPPAN. « Off-line handwritten character recognition using Hidden Markov Model ». In : *Advances in Computing, Communications and Informatics (ICACCI, 2014 International Conference on)*. IEEE. 2014, p. 518–523.
- [GIN 04] Michael van GINKEL, C. L. Luengo HENDRIKS et Lucas J. van VLIET. « A short introduction to the Radon and Hough transforms and how they relate to each other ». In : *Delft University of Technology* (2004).
- [GOR 99] Nikolai GORSKI et al. « A2ia check reader : A family of bank check recognition systems ». In : *Document Analysis and Recognition, 1999. IC-DAR'99. Proceedings of the Fifth International Conference on*. IEEE. 1999, p. 523–526.
- [GRA 03] Frédéric GRANDIDIER. « Un nouvel algorithme de sélection de caractéristiques : application à la lecture automatique de l'écriture manuscrite ». Thèse de doct. École de technologie supérieure, 2003.
- [GRA 04] Frédéric GRANDIDIER, Robert SABOURIN et Ching Y. SUEN. « Quelques techniques pour l'amélioration du pouvoir discriminant de primitives discrètes ». In : *Conférence Internationale Francophone sur l'Écrit et le Document (CIFED 04)*. 2004.
- [GRE 10] Julie GREENSMITH, Amanda WHITBROOK et Uwe AICKELIN. « Artificial immune systems ». In : *Handbook of Metaheuristics*. Springer, 2010, p. 421–448.
- [GRO 95] Patrick J. GROTHOR. « NIST special database 19 handprinted forms and characters database ». In : *National Institute of Standards and Technology* (1995).
- [HAB 12] Amar HABOUSSI. « Systèmes Immunitaires Artificiels pour le diagnostic des systèmes complexes ». Mémoire de magister. Université Hadj Lakhdar ,Batna, Algérie, 2012.
- [HAJ 05] Mohammad Mehdi HAJI. « Farsi handwritten word recognition using continuous hidden Markov models and structural features ». In : *Iran : MSC, Computer Engineering Shiraz University Shiraz* (2005).
- [HAN 03] Madasu HANMANDLU et al. « Unconstrained handwritten character recognition based on fuzzy logic ». In : *Pattern Recognition 36.3* (2003), p. 603–623.
- [HER 98] Pierre HEROUX et al. « Classification method study for automatic form class identification ». In : *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on*. T. 1. IEEE. 1998, p. 926–928.
- [HEU 03] Laurent HEUTTE. « Analyse de reconnaissance de l'écriture : de nouvelles perspectives en traitement automatique de Documents manuscrits ». Thèse de doct. Université de Rouen, 2003.
- [HIR 01] Takashi HIRANO, Okada YASUHIRO et Yoda FUMIO. « Field Extraction Method from Existing Forms Transmitted by Facsimile ». In : *ICDAR*. 2001.
- [HUS 14] Shaista HUSSAIN, Shih-Chii LIU et Arindam BASU. « Improved margin multi-class classification using dendritic neurons with morphological learning ». In : *2014 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE. 2014, p. 2640–2643.

- [JAG 14] Viragkumar N. JAGTAP et Shailendra K. MISHRA. « Fast efficient artificial neural network for handwritten digit recognition ». In : *International Journal of Computer Science and Information Technologies* 5.2 (2014), p. 2302–2306.
- [JAI 00] Anil K. JAIN, Robert P. W. DUIN et Jianchang MAO. « Statistical pattern recognition : A review ». In : *IEEE Transactions on pattern analysis and machine intelligence* 22.1 (2000), p. 4–37.
- [JER 74] Neilo Kaj JERNE. « Towards a Network Theory of the Immune System ». In : *Annual Immunology* 125 (1974).
- [JIA 14] Wenhao JIANG et Fu-lai CHUNG. « A trace ratio maximization approach to multiple kernel-based dimensionality reduction ». In : *Neural Networks* 49 (2014), p. 96–106.
- [JOU 09] Chichang JOU et Hung-Chang LEE. « Handwritten numeral recognition based on simplified structural classification and fuzzy memberships ». In : *Expert Systems with Applications* 36.9 (2009), p. 11858–11863.
- [KAP 02] Rajiv KAPOOR, Deepak BAGAI et Tara Singh KAMAL. « Representation, Extraction of Nodal Features of DevNagri Letters. » In : *ICVGIP*. Citeseer. 2002.
- [KAP 85] Jagat Narain KAPUR, Prasanna K SAHOO et Andrew KC WONG. « A new method for gray-level picture thresholding using the entropy of the histogram ». In : *Computer vision, graphics, and image processing* 29.3 (1985), p. 273–285.
- [KAU 00] Guido KAUFMANN et Horst BUNKE. « Detection and correction of recognition errors in check reading ». In : *International Journal on Document Analysis and Recognition* 2.4 (2000), p. 211–221.
- [KEF 10] Abderrahmane KEFALI, Toufik SARI et Mokhtar SELLAMI. « Evaluation of several binarization techniques for old Arabic documents images ». In : 2010.
- [KEL 85] James M. KELLER, Michael R. GRAY et James A. GIVENS. « A fuzzy k-nearest neighbor algorithm ». In : *IEEE transactions on systems, man, and cybernetics* 4 (1985), p. 580–585.
- [KIM 00] Jin Ho KIM, Kye Kyung KIM et Ching Y SUEN. « An HMM-MLP hybrid model for cursive script recognition ». In : *Pattern Analysis & Applications* 3.4 (2000), p. 314–324.
- [KIM 02] Jung Won KIM. « Integrating artificial immune algorithms for intrusion detection ». Thèse de doct. University College London (University of London), 2002.
- [KIM 98] Gyeonghwan KIM et Venu GOVINDARAJU. « Handwritten phrase recognition as applied to street name images ». In : *Pattern Recognition* 31.1 (1998), p. 41–51.
- [KIN 07] Thomas J. KINDT et Barbara A. GOLDSBY Richard A. and Osborne. *Ku-byImmunology : international edition*. WH Freeman San Fransisco, 2007.

- [KLA 02] Tim J. KLASSEN et Malcolm I. HEYWOOD. « Towards the on-line recognition of arabic characters ». In : *Proceedings of the 2002 International Joint Conference on Neural Networks IJCNN*. T. 2. 2002, p. 1900–1905.
- [KUN 00] Murat KUNT. *Reconnaissance des formes et analyse de scènes*. T. 3. PPUR presses polytechniques, 2000.
- [KUN 04] Ludmila I. KUNCHEVA. *Combining pattern classifiers : methods and algorithms*. John Wiley & Sons, 2004.
- [LAA 97] Jorma LAAKSONEN et al. *Subspace classifiers in recognition of handwritten digits*. Helsinki University of Technology, 1997.
- [LAC 13] Everton B. LACERDA et Carlos A. B. MELLO. « Segmentation of connected handwritten digits using Self-Organizing Maps ». In : *Expert systems with applications* 40.15 (2013), p. 5867–5877.
- [LEB 07] Frank LE-BOURGEOIS et Hubert EMPTOZ. « Debora : Digital access to books of the renaissance ». In : *International Journal of Document Analysis and Recognition (IJ DAR)* 9.2-4 (2007), p. 193–221.
- [LEC 98] Yann LECUN et al. « Gradient-based learning applied to document recognition ». In : *Proceedings of the IEEE* 86 (1998), p. 2278–2324.
- [LEP 12] Julien LEPAGNOT. « Conception de métaheuristiques pour l’optimisation dynamique : application à l’analyse de séquences d’images IRM ». Thèse de doct. Université Paris-Est, 2012.
- [LIU 01] Cheng-Lin LIU et Masaki NAKAGAWA. « Evaluation of prototype learning algorithms for nearest-neighbor classifier in application to handwritten character recognition ». In : *Pattern Recognition* 34.3 (2001), p. 601–615.
- [LIU 03] Cheng-Lin LIU et al. « Handwritten digit recognition : benchmarking of state-of-the-art techniques ». In : *Pattern Recognition* 36.10 (2003), p. 2271–2285.
- [LIU 04] Cheng-Lin LIU et al. « Handwritten digit recognition : investigation of normalization and feature extraction techniques ». In : *Pattern Recognition* 37.2 (2004), p. 265–279.
- [LIW 07] Marcus LIWICKI et Horst BUNKE. « Handwriting recognition of whiteboard notes studying the influence of training set size and type ». In : *International Journal of Pattern Recognition and Artificial Intelligence* 21.01 (2007), p. 83–98.
- [LOR 98] Guy LORETTE et Jean-Pierre CRETTEZ. « Reconnaissance de l’écriture manuscrite ». In : *Traité informatique. Techniques de l’Ingénieur*, 1998, p. 1–15.
- [LOU 09] Georgios LOULLOUDIS et al. « Text line and word segmentation of handwritten documents ». In : *Pattern Recognition* 42.12 (2009), p. 3169–3183.
- [LU 02] Yue LU et Chew Lim TAN. « Combination of multiple classifiers using probabilistic dictionary and its application to postcode recognition ». In : *Pattern Recognition* 35 (2002), p. 2823–2832.
- [LU 96] Yi LU et Malayappan SHRIDHAR. « Character segmentation in handwritten words an overview ». In : *Pattern recognition* 29.1 (1996), p. 77–96.

- [MAD 01] Sriganesh MADHVANATH, Venu KRPASUNDAR et Venu GOVINDARAJU. « Syntactic methodology of pruning large lexicons in cursive script recognition ». In : *Pattern Recognition* 34.1 (2001), p. 37–46.
- [MAR 01] Urs-Victor MARTI et Horst BUNKE. « Using a statistical language model to improve the performance of an HMM-based cursive handwriting recognition system ». In : *International journal of Pattern Recognition and Artificial intelligence* 15.01 (2001), p. 65–90.
- [MÄR06] Volker MÄRGNER, Haikal EL ABED et Mario PECHWITZ. « Offline Handwritten Arabic Word Recognition Using HMM - a Character Based Approach without Explicit Segmentation ». In : *The 9th colloque International Francophone sur l'écrit et le Document, CIFED, Fribourg Swiss*. 2006, p. 259–264.
- [MAT 02] Polly MATZINGER. « The Danger Model : a Renewed Sense of Self ». In : *Science* 296 (2002), p. 301–304.
- [MEN 08] Fares MENASRI. « Contributions à la reconnaissance de l'écriture arabe manuscrite ». Thèse de doct. Université Paris Descartes, 2008.
- [MER 14] Hocine MERABTI, M. Nadjib KOUAHLA et Hamid SERIDI. « Reconnaissance des Chiffres Manuscrits par Un Système Immunitaire Artificiel Flou ». In : *4èmes Colloque International ISKO-Maghreb'2014, Concepts et Outils pour le Management de la Connaissance (KM)*. Alger, Novembre, 2014.
- [MER 16] Hocine MERABTI, M. Nadjib KOUAHLA et Hamid SERIDI. « Bio-qualitative rules-based system for handwritten characters recognition ». In : *KES Journal* 20.1 (2016), p. 21–36.
- [MEZ 02] Neila MEZGHANI, Amar MITICHE et Mohamed CHERIET. « Reconnaissance en-ligne de caractères arabes manuscrits par un réseau de Kohonen ». In : *Proc. of Vision Interface 2002* (2002), p. 186–191.
- [MEZ 03] Neila MEZGHANI, Mohamed CHERIET et Amar MITICHE. « Combination of Pruned Kohonen Maps for On-line Arabic Characters Recognition. » In : *ICDAR*. T. 3. Citeseer. 2003, p. 900.
- [MEZ 05] Neila MEZGHANI, Amar MITICHE et Mohamed CHERIET. « A new representation of shape and its use for high performance in online Arabic character recognition by an associative memory ». In : *International Journal of Document Analysis and Recognition (IJ DAR)* 7.4 (2005), p. 201–210.
- [MIL 97] Housem MILED et al. « Coupling observation/letter for a Markovian modelisation applied to the recognition of Arabic handwriting ». In : *Document Analysis and Recognition, 1997., Proceedings of the Fourth International Conference on*. T. 2. IEEE. 1997, p. 580–583.
- [MON 10] Gholam Ali MONTAZER, Hamed Qahri SAREMI et Vahid KHATIBI. « A neuro-fuzzy inference engine for Farsi numeral characters recognition ». In : *Expert Systems with Applications* 37 (2010), p. 6327–6337.
- [MOR 03] Marisa Emika MORITA. *Automatic recognition of handwritten dates on Brazilian bank cheques*. Ecole de Technologie Supérieure (Canada), 2003.
- [MOT 02] Hiroshi MOTODA et Huan LIU. « Feature selection, extraction and construction ». In : *Communication of IICM (Institute of Information and Computing Machinery, Taiwan) Vol 5* (2002), p. 67–72.

- [MOT 97] Deya MOTAWA, Adnan AMIN et Robert SABOURIN. « Segmentation of Arabic Cursive Script. » In : *icdar*. T. 97. 1997, p. 625–628.
- [NAN 09] Satyasai Jagannath NANDA. « Artificial Immune Systems : Principle, Algorithms And Applications ». Thèse de doct. Institut national de technologie, électronique et ingénierie de communication, Rourkela, India, 2009.
- [NEB 13a] Salima NEBTI. « La reconnaissance de caractères manuscrits par l’intelligence collective ». Thèse de doct. Université Ferhat Abass, Sétif, 2013.
- [NEB 13b] Salima NEBTI et Abdellah BOUKERRAM. « Handwritten characters recognition based on nature-inspired computing and neuro-evolution ». In : *Applied intelligence* 38.2 (2013), p. 146–159.
- [NEM 10] Mohammed NEMISSI. « Classification et reconnaissance des formes par algorithmes hybrides ». Thèse de doct. Université 08 mai 1945, Guelma, 2010.
- [NIB 85] Wayne NIBLACK. *An introduction to digital image processing*. Strandberg Publishing Company, 1985.
- [NIC 06] Stéphane NICOLAS. « Segmentation par champs aléatoires pour l’indexation d’images de documents ». Thèse de doct. Rouen, 2006.
- [OLL 99] Daniel OLLIVIER. « Une approche économisant les traitements pour reconnaître l’écriture manuscrite : application à la reconnaissance des montants littéraux de chèques bancaires ». Thèse de doct. Université Paris XI (Orsay), 1999.
- [OTS 75] Nobuyuki OTSU. « A threshold selection method from gray-level histograms ». In : *Automatica* 11.285-296 (1975), p. 23–27.
- [OUC 07] Salim OUCHTATI, Mouldi BEDDA et Abderrazak LACHOURI. « Segmentation and recognition of handwritten numeric chains ». In : *Journal of Computer Science* 3.4 (2007), p. 242–248.
- [OWE 13] Judith A. OWEN, Jenni PUNT et Sharon A. STRANFORD. *Kuby immunology*. WH Freeman New York, 2013.
- [PAL 01] Umapada PAL, Abdel BELAÏD et Christophe CHOISY. « Water reservoir based approach for touching numeral segmentation ». In : *Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on*. IEEE. 2001, p. 892–896.
- [PAL 03] Rafael PALACIOS, Amar GUPTA et Patrick SP WANG. « Feedback-based architecture for reading courtesy amounts on checks ». In : *Journal of Electronic Imaging* 12.1 (2003), p. 194–202.
- [PAQ 01] Thierry PAQUET, Heutte LAURENT et Lecourtier YVES. « Problématique de la Reconnaissance de l’Écriture ». In : *ASTI’2001 des Sons, des Images et des Documents à leur Interprétation, France*. 2001.
- [PAR 99] Jaehwa PARK. « Hierarchical character recognition and its use in handwritten word/phrase recognition ». Thèse de doct. State University of New York at Buffalo, 1999.
- [PEC 03] Mario PECHWITZ et Volker MAERGNER. « HMM Based Approach for Handwritten Arabic Word Recognition Using the IFN/ENIT-Database. » In : *ICDAR*. T. 3. Citeseer. 2003, p. 890.



- [PEC 06] Mario PECHWITZ, Volker MAERGNER et Haikal EL ABED. « Comparison of two different feature sets for offline recognition of handwritten Arabic words ». In : *Tenth International Workshop on Frontiers in Handwriting Recognition*. Suvisoft. 2006.
- [PER 79] Alan S. PERELSON et George F. OSTER. « Theoretical studies of clonal selection : minimal antibody repertoire size and reliability of self-non-self discrimination ». In : *Journal of theoretical biology* 81.4 (1979), p. 645–670.
- [PLA 00] Réjean PLAMONDON et Sargur N SRIHARI. « Online and off-line handwriting recognition : a comprehensive survey ». In : *IEEE Transactions on pattern analysis and machine intelligence* 22.1 (2000), p. 63–84.
- [PUN 80] Thierry PUN. « A new method for gray\_level picture threshold using the entropy of the histogram ». In : *Signal processing* 2.3 (1980), p. 223–237.
- [RAJ 11] Rani RAJNEESH, Siddharth KARTAR SINGH et Renu DHIR. *Handwritten Gurmukhi Numeral Recognition using Different Feature Sets*. 2011.
- [RED 15] P. Kiran Kumar REDDY. « Discriminative wavelet shape descriptors for recognition of 2d patterns and efficient usage of morphological 2d transforms ». Thèse de doct. FACULTY OF COMPUTER SCIENCE et ENGINEERING, Jawaharlal Nehru Technological University Anantapur, 2015.
- [SAH 07] Seral SAHAN et al. « A new hybrid method based on fuzzy-artificial immune system and k-nn algorithm for breast cancer diagnosis ». In : *Computers in Biology and Medicine* 37.3 (2007), p. 415–423.
- [SAI 01] Christophe SAINT-JEAN. « Classification paramétrique robuste partiellement supervisée en reconnaissance des formes ». In : *Thèse de Doctorat, université de La Rochelle-UFR Sciences Laboratoire d'Informatique et d'Imagerie Industrielle* (2001).
- [SAO 97] George SAON. « Modèles markoviens uni- et bidimensionnels pour la reconnaissance de l'écriture manuscrite hors-ligne ». Thèse de doct. Université Henri Poincaré, Nancy 1, 1997.
- [SAU 00] Jaakko SAUVOLA et Matti PIETIKÄINEN. « Adaptive document image binarization ». In : *Pattern recognition* 33.2 (2000), p. 225–236.
- [SAY 73] Kenneth M SAYRE. « Machine recognition of handwritten words : A project report ». In : *Pattern recognition* 5.3 (1973), p. 213–228.
- [SCA 00] Carlo SCAGLIOLA, Gianluca NICCHIOTTI et Francesco CAMASTRA. « Enhancing cursive word recognition performance by the integration of all the available information ». In : *7th International Workshop on Frontiers in Handwriting Recognition*. Amsterdam, Netherlands. 2000.
- [SCH 96] Holger SCHWENK et Maurice MILGRAM. « Constraint tangent distance for on-line character recognition ». In : *Pattern Recognition, 1996., Proceedings of the 13th International Conference on*. T. 4. IEEE. 1996, p. 520–524.
- [SEG 06] Marc SEGOND. « Algorithmes bio-mimétiques pour la reconnaissance de formes et l'apprentissage ». Thèse de doct. Université de du Littoral - Côte d'Opale, 2006.

- [SER 01] Hamid SERIDI. « Une nouvelle approche qualitative du traitement de l'incertain ». Thèse de doct. Université de Reims Champagne-Ardenne, 2001.
- [SER 06] Hamid SERIDI et al. « Approximate reasoning in supervised classification systems ». In : *Journal of Advanced Computational Intelligence and Intelligent Informatics* 10.4 (2006), p. 586–593.
- [SER 13] Hamid SERIDI, Hocine MERABTI et M. Nadjib KOUAHLA. « AIRS Classifiers in Handwriting Recognition ». In : *Control Engineering Conference CONTECH '13, Istanbul, Turkey*. December, 2013.
- [SHU 08] Abdulla SHUBAIR, Amer AL-NASSIRI et Rosalina Abdul SALAM. « Off-Line Arabic Handwritten Word Segmentation Using Rotational Invariant Segments Features. » In : *Int. Arab J. Inf. Technol.* 5.2 (2008), p. 200–208.
- [SIM 84] Jean-Claude SIMON. *La reconnaissance des formes par algorithmes*. Rapp. tech. 1984.
- [SIT 16] SITE. *Android, MT*. URL : <http://www.android-mt.com/application/note-anytime-lite-prise-notes-au-stylet-24101> (visité le 10/06/2016).
- [SKU 02] Marina SKURICHINA, Liudmila I KUNCHEVA et Robert PW DUIN. « Bagging and boosting for the nearest mean classifier : Effects of sample size on diversity and accuracy ». In : *International Workshop on Multiple Classifier Systems*. Springer. 2002, p. 62–71.
- [SMR 98] Pavel SMRŽ, Michal MARTINÁSEK et al. « Offline Recognition of Cursive Handwritten Czech Text ». In : *International Conference on Current Trends in Theory and Practice of Computer Science*. Springer. 1998, p. 437–442.
- [TAY 01] Yong Haur TAY et al. « An offline cursive handwritten word recognition system ». In : *TENCON 2001. Proceedings of IEEE Region 10 International Conference on Electrical and Electronic Technology*. T. 2. IEEE. 2001, p. 519–524.
- [TEA 80] Michael Reed TEAGUE. « Image analysis via the general theory of moments ». In : *JOSA* 70.8 (1980), p. 920–930.
- [THE 03] Sergios THEODORIDIS et Konstantinos KOUTROUMBAS. *Pattern Recognition, Second Edition*. Academic Press San Diego, Calif, USA, 2003.
- [THE 12] Pavlidis THEODOSIOS. *Algorithms for graphics and image processing*. Springer Science & Business Media, 2012.
- [TIM 00] Jonathan TIMMIS. « Artificial immune systems : a novel data analysis technique inspired by the immune network theory ». Thèse de doct. Department of Computer Science, 2000.
- [TIM 01] Jonathan TIMMIS et Mark NEAL. « A resource limited artificial immune system for data analysis ». In : *Knowledge-Based Systems* 14.3 (2001), p. 121–130.
- [TIM 04] Jonathan TIMMIS et al. « An overview of artificial immune systems ». In : *Computation in Cells and Tissues*. Springer, 2004, p. 51–91.
- [TIM 08a] Jonathan TIMMIS et al. « An interdisciplinary perspective on artificial immune systems ». In : *Evolutionary Intelligence* 1.1 (2008), p. 5–26.

- [TIM 08b] Jonathan TIMMIS et al. « Immuno engineering ». In : *Biologically-Inspired Collaborative Computing*. Springer, 2008, p. 3–17.
- [TIM 08c] Jonathan TIMMIS et al. « Theoretical advances in artificial immune systems ». In : *Theoretical Computer Science* 403.1 (2008), p. 11–32.
- [TOU 92] Claude TOUZET. *Les réseaux de neurones artificiels : introduction au connexionnisme*. EC2, 1992.
- [TRE 04] Guillaume TREMBLAY. *Optimisation d'ensembles de classifieurs non paramétriques avec apprentissage par représentation partielle de l'information*. 2004.
- [TRI 96] Oivind Due TRIER, Anil K JAIN et Torfinn TAXT. « Feature extraction methods for character recognition-a survey ». In : *Pattern recognition* 29.4 (1996), p. 641–662.
- [VAM 09] Georgios VAMVAKAS, Basilis GATOS et Stavros J PERANTONI. « A novel feature extraction and classification methodology for the recognition of historical documents ». In : *2009 10th International Conference on Document Analysis and Recognition*. IEEE. 2009, p. 491–495.
- [VAM 10] Georgios VAMVAKAS, Basilis GATOS et Stavros J PERANTONIS. « Handwritten character recognition through two-stage foreground sub-sampling ». In : *Pattern Recognition* 43.8 (2010), p. 2807–2816.
- [VIN 02] Alessandro VINCIARELLI. « A survey on off-line cursive word recognition ». In : *Pattern recognition* 35.7 (2002), p. 1433–1446.
- [WAT 01] Andrew WATKINS. « AIRS : A resource limited artificial immune classifier ». Thèse de doct. Mississippi Sate University, 2001.
- [WAT 04] Andrew WATKINS, Jonathan TIMMIS et Lois BOGGESE. « Artificial immune recognition system (AIRS) : An immune-inspired supervised learning algorithm ». In : *Genetic Programming and Evolvable Machines* 5.3 (2004), p. 291–317.
- [WAT 85] Satoshi WATANABE. *Pattern recognition : human and mechanical*. John Wiley & Sons, Inc., 1985.
- [WEB 02] Andrew R WEBB. *Statistical Pattern Recognition, Second edition*. John Wiley & Sons, 2002.
- [WOO 96] Jeffrey WOOD. « Invariant pattern recognition : a review ». In : *Pattern recognition* 29.1 (1996), p. 1–17.
- [WOZ 03] P. WOZNICA et M. MENNAL. *Reconnaissance des formes*. 2003.
- [XU 01] Qizhi XU, Louisa LAM et Ching Y SUEN. « A knowledge-based segmentation system for handwritten dates on bank cheques ». In : *Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on*. IEEE. 2001, p. 384–388.
- [YAN 98] Berrin YANIKOGLU et Peter A SANDON. « Segmentation of off-line cursive handwriting using linear programming ». In : *Pattern Recognition* 31.12 (1998), p. 1825–1833.
- [ZER 07] Narima ZERMI, Messaoud RAMDANI et Mouldi BEDDA. « Arabic Handwriting Word Recognition Based on a Hybrid HMM/ANN Approach ». In : *International Journal of Soft Computing* 2.1 (2007), p. 5–10.

- [ZHU 11] Cheng-hui ZHU et al. « Study of off-line handwritten Chinese character recognition based on dynamic pruned FSVMs ». In : *Electrical and Control Engineering (ICECE), 2011 International Conference on*. IEEE. 2011, p. 395–398.

# Publications de l'auteur

## PUBLICATION INTERNATIONALE

Hocine Merabti, M. Nadjib Kouahla, Hamid Seridi : «Bio-qualitative rules-based system for handwritten characters recognition ». KES Journal20(1) : 21-36 (2016).

## COMMUNICATIONS INTERNATIONALES

Hocine Merabti, Brahim Farou, M. Nadjib Kouahla, Hamid Seridi, «Extraction des Caractéristiques Structurelles pour la Segmentation des Chiffres Manuscrits Connectés ». 13èmes Colloque International sur l'Optimisation et les Systèmes d'Information CO-SI'2016, Sétif, Algérie, 30 mai au 1 Juin 2016.

Hocine Merabti, Brahim Farou, M. Nadjib Kouahla, Hamid Seridi, « Une nouvelle méthode pour la segmentation des chiffres manuscrits connectés ». 5èmes Colloque International ISKO-Maghreb'2015, Knowledge Organization in the perspective of Digital Humanities. Hammamet (Tunisia), Novembre 2015.

Hocine Merabti, M. Nadjib Kouahla, Hamid Seridi, « Reconnaissance des Chiffres Manuscrits par Un Système Immunitaire Artificiel Flou ». 4èmes Colloque International ISKO-Maghreb'2014, Concepts et Outils pour le Management de la Connaissance (KM). Alger, Novembre 2014.

Hamid Seridi, Hocine Merabti, M. Nadjib Kouahla, « AIRS Classifiers in Handwriting Recognition ». Control Engineering Conference CONTECH '13, Istanbul, Turkey, December 2013.

Benhamza Karima, Hocine Merabti, Hamid Seridi, « Adaptive Edge Detection Using Ant Colony». The 8th International Workshop on Systems, Signal Processing and their Applications 2013 (WOSSPA2013), Alger, Mai 2013.

COMMUNICATIONS NATIONALES

Hocine Merabti, Brahim Farou, M. Nadjib Kouahla, Hamid Seridi, « An Artificial Immune Simulation for a Complex Problem Resolution ». Le 1er séminaire national Simulation Numérique sur les Sciences Appliquées SNSA'2016, Guelma, Décembre 2016.

Hocine Merabti, Brahim Farou, M. Nadjib Kouahla, Hamid Seridi, « New method for Segmentation of connected handwritten digits ». La 4ème Journée des Sciences et Technologies de l'Information et de la Communication JSTIC'2015, Guelma, Mai 2015.

Hocine Merabti, Hamid Seridi, « Image Segmentation by Using Ant Colony ». La 1ère Journée d'études Simulation et NanoMatériaux, Guelma, Décembre 2014.

Hocine Merabti, M. Nadjib Kouahla, Hamid Seridi, « Handwritten Digits Recognition Using a Fuzzy Artificial Immune System ». Les 3èmes Journées Doctorales en Informatique JDI'2014, Guelma, Décembre 2014.

Hocine Merabti, M. Nadjib Kouahla, Hamid Seridi, « Handwritten Digits Recognition Using Bio-inspired Approach and Fuzzy Rules ». La 3ème Journée des Sciences et Technologies de l'Information et de la Communication JSTIC'2014, Guelma, Juin 2014.

Hocine Merabti, M. Nadjib Kouahla, Hamid Seridi, « Un système immunitaire artificiel pour la reconnaissance des caractères manuscrits isolés ». La 2ème Journée des Sciences et Technologies de l'Information et de la Communication JSTIC'2013, Guelma, Juillet 2013.

Hocine Merabti, Karima Benhamza, Hamid Seridi, « Etude et analyse les méthodes du Natural Computing ». Les 2èmes Journées Doctorales en Informatique JDI'2012, Guelma, Novembre 2012.

Hocine Merabti, Karima Benhamza, Hamid Seridi, « Etat de l'Art sur le Natural Computing ». La 1ère Journée des Sciences et Technologies de l'Information et de la Communication JSTIC'2012, Guelma, Juillet 2012.