MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE MOULOUD MAMMERI, TIZI-OUZOU

FACULTE DE GENIE ELECTRIQUE ET D'INFORMATIQUE
DEPARTEMENT AUTOMATIQUE

THESE DE DOCTORAT

en **Automatique**

Présentée par

AIT AIDER Malika

Ingénieur UMMTO
Magister UMMTO

CONTRIBUTION A LA RECONNAISSANCE AUTOMATIQUE DE CARACTERES NUMERIQUES

Thèse soutenu le 08/07/2019 ddeevant le jury d'examen composé de

M. DIAF Moussa	Professeur	UMMTO	**Président**
M. HAMMOUCHE Kamal	Professeur	UMMTO	**Rapporteur**
M. MEDJAHER Kamal	Professeur	Université de Toulouse, France	**Examinateur**
Mme. HAMAMI Latifa	Professeur	ENP Alger	**Examinateur**
Mme. ACHELI Dalila	Professeur	UMB Boumerdès	**Examinateur**
M. LAHDIR Mourad	Professeur	UMMTO	**Examinateur**
M. GACEB Djamel	MCA	UMB Boumerdès	**Invité**

2019

Les machines sont déjà interactives avec le cerveau : l'ordinateur et le genre humain travaillent ensemble. Il se pourrait d'ailleurs qu'un jour les historiens se rendent compte que l'événement le plus important du XXe siècle, ce n'était pas la guerre, ni le krach financier, mais le soir où Kasparov, le joueur d'échecs, a perdu sa partie contre une petite boîte en métal. Et noté : La machine n'a pas calculé, elle a pensé.

Entretien Télérama n◦ 3230 décembre 2011

George Steiner

Remerciements

En premier lieu, je souhaite témoigner de toute ma reconnaissance à Monsieur HAMMOUCHE Kamal, mon directeur de thèse, pour m'avoir fait confiance pour conduire ces travaux. Je suis particulièrement touchée par la disponibilité qu'il m'a accordée au cours de la réalisation de cette thèse. Je le remercie infiniment pour les nouvelles et nombreuses idées suggérées pour enrichir ce travail. Merci d'avoir orienté ce sujet vers une thématique récente et d'actualité. Merci pour toute l'aide que vous m'avez apportée durant la réalisation de cette thèse.

Mes remerciements s'adressent également à Monsieur GACEB Djamel, maitre de conférence classe A (HDR), à l'université M'hammed BOUGARA de Boumerdès, d'avoir apporté sa contribution dans ce travail de thèse. Je lui exprime ici ma gratitude pour toute l'aide qu'il m'a apportée. Merci infiniment d'avoir répondu à mon invitation.

Je tiens à remercier Monsieur DIAF Moussa, professeur à l'université Mouloud MAMMERI de Tizi-Ouzou, qui m'a fait le grand honneur d'accepter la présidence du jury et pour l'intérêt qu'il a porté à ce travail.

Mes remerciements s'adressent aux examinateurs Madame HAMAMI Latifa, professeur à ENP d'Alger, Madame ACHELI Dalila, professeur à UMB Boumerdès, Monsieur MEDJAHER Kamal, professeur à l'université de Toulouse et Monsieur LAHDIR Mourad, professeur à l'université Mouloud MAMMERI de Tizi-Ouzou, pour leur déplacement et pour avoir pris le temps

d’examiner ce modeste manuscrit avec une grande attention. Je tiens à les remercier vivement d’avoir accepté la charge d’évaluer ce travail et de prendre part à mon jury de thèse. Un grand merci pour leur travail critique vis à vis de ce manuscrit, et également pour leur attention lors de la soutenance.

Un immense merci à Monsieur RAVI pour sa relecture extrêmement rigoureuse et ses très nombreuses corrections. Je souhaite également remercier Monsieur GUERMAH Said d’avoir apporté sa contribution dans ce travail de thèse. Je n’y manquerai pas de remercier également mes collègues pour leur soutien moral.

Pour finir, je suis infiniment reconnaissante à ma famille pour leur affection, leur patience et leur soutien.

Merci à tous !

Table des matières

Liste des figures

Liste des tableaux

Liste des abréviations

OCR	Optical Character Recognition.	1
HMMs	Hidden Markov Models.	6
CNN	Convolutional Neural Network.	20
Kppv	K plus proches voisins.	25
SVM	Support Vector Machines.	25
MLP	Multi Layer Perceptron.	25
k-means	Algorithme des k moyennes.	25
Fuzzy c-means	Algorithme des c moyennes ou k moyennes floue.	26
SIFT	Scale Invariant Feature Transform.	33
DoG	Difference of Gaussian.	34
LoG	Laplacian of Gaussian.	34
PCA-SIFT	Principal Component Analysis-Scale Invariant Feature Transform.	35
SURF	Speeded Up Robust Features.	35
BOW	Bag of Visual Words.	37
HOG	Histogram of Oriented Gradients.	37
LBP	Local Binary Patterns.	39
AMR	Analyse Multirésolution.	45
TOD	Transformée en Ondelettes Discrète.	46
USPS	United States Postal Service database	47
MNIST	Mixed (Modified) National Institute of Standards and Technology.	48
NIST	National Institute of Standards and Technology	48
CVLSD	Computer Vision Lab Single Digit dataset.	48
SVHN	Street View House Numbers.	49
CDF	Cohen-Daubechies-Feauveau wavelet.	54
RBM	Restricted Boltzmann Machine.	76
DBM	Deep Boltzmann Machine	80
DAE	Deep Auto-Encodeurs.	81

Introduction

Depuis bien longtemps, l'écriture est considérée comme une modalité de communication privilégiée entre les individus. C'est aussi un moyen de diffusion et de conservation des connaissances. Aujourd'hui, bien que l'imprimerie puis l'informatique aient permis d'automatiser le processus d'écriture, l'écriture manuscrite reste extrêmement présente dans notre monde. En effet, une masse de documents manuscrits continue de croitre chaque jour, et de plus en plus d'industries et de services ont un besoin de faire appel à des techniques de traitement rapides, tout en assurant la sécurité de ces documents.

Pour satisfaire en partie les exigences des ces industriels, des machines dotées d'un système de reconnaissance des caractères ont été mises en place. Un système de reconnaissance des caractères se base sur les principes de l'intelligence artificielle et plus précisément celui de l'apprentissage automatique (Machine Learning) ou de l'apprentissage profond (Deep Learning) dont l'objectif est de doter les ordinateurs de la capacité d'apprendre à reconnaitre un caractère, un mot ou une phrase. Les applications potentielles visées par cette approche sont le traitement automatique des chèques, du courrier postal, de formulaires, de documents d'archive. Malgré que le domaine de la reconnaissance des caractères est très ancien et que d'énormes progrès ont été enregistrés, il demeure un domaine actif de recherche pour la science informatique d'autant plus que, jusqu'à ce jour, aucun système n'est fiable à 100%. En effet, un système de reconnaissance de l'écriture manuscrite générique est loin d'être réalisé à cause de la variabilité de l'écriture manuscrite. En outre, c'est un domaine très vaste tant par ses applications spécifiques que par ses techniques.

On distingue traditionnellement la reconnaissance en ligne et la reconnaissance hors ligne. La reconnaissance en ligne traite des signaux temporels qui sont saisis au moyen d'un stylet sur une surface sensible, comme les tablettes électroniques. Ce type de système s'avère plus au moins difficile vu que la reconnaissance s'effectue pendant la saisie du texte. Quant à la

reconnaissance hors ligne, elle a pour objectif de convertir les documents papiers scannés vers des formats modifiables et exploitables.
Généralement, une stratégie utilisée dans la reconnaissance hors ligne de l'écriture manuscrite consiste à segmenter un texte en lignes, mots, et caractères puis de procéder à la reconnaissance de caractères isolés.

La reconnaissance de caractères isolés implique une étape d'extraction des caractéristiques pouvant décrire les caractères et une méthode de classification dont le but est de construire un modèle de référence pour chaque classe (caractère). Les performances d'un tel système dépendent fortement de la qualité des caractéristiques à discerner les caractères. Celles-ci doivent être pertinentes et non redondantes. On conçoit qu'un mauvais choix de ces caractéristiques entraine automatiquement la chute des performances du système de reconnaissance, même en présence d'un classifieur très performant.

L'objectif de cette thèse est porté principalement sur l'extraction des caractéristiques pour la reconnaissance des chiffres manuscrits isolés. Plusieurs techniques d'extraction et de sélection des caractéristiques discriminantes sont étudiées et analysées. Cependant, nos contributions s'articulent essentiellement autour de la transformée en ondelettes discrète (TOD), les réseaux de neurones convolutifs (CNN) et le classifieur machine à vecteurs de support (SVM).

La TOD permet de décomposer l'image du caractère en sous-bandes images, plus au moins décorrélées entre elles, afin de ressortir des caractéristiques locales et globes. Son avantage réside dans l'usage des fenêtres d'analyses de tailles variables pouvant détecter dans une image du caractère des caractéristiques pertinentes, difficilement décelables voire invisibles sur l'image originale. Tandis que les réseaux de neurones convolutifs (CNN), qui font partie des techniques d'apprentissage profond ou *Deep Learning*, sont exploitées pour générer automatiquement des caractéristiques par apprentissage à partir d'un ensemble de données originales ou représentées dans le domaine de la TOD. A partir de ces caractéristiques, le classifieur SVM est adopté pour discriminer les chiffres manuscrits.

Cette thèse est organisée en 4 chapitres:
Le chapitre 1 présente quelques notions générales sur l'écriture manuscrite. Il décrit les principales étapes nécessaires pour la réalisation d'un système de reconnaissance de l'écriture, tout en détaillant chacun des modules. On y trouvera également les principales contraintes qui

compliquent la reconnaissance de l'écriture manuscrite ainsi que quelques applications concernant ce domaine.

Le chapitre 2 présente un état de l'art sur les principales méthodes d'extraction de caractéristiques utilisées en reconnaissance des caractères manuscrits et imprimés. Certaines techniques telles que la transformation de caractéristiques invariantes à l'échelle (SIFT), les caractéristiques robustes accélérées (SURF), les techniques de sacs de caractéristiques visuels (BOF), les histogrammes de gradients orientés (HOG), les motifs binaires locaux (LBP), les moments de Zernike et les filtres de Gabor sont exposées d'une manière détaillée.

Le chapitre 3 est consacré aux techniques développées pour la reconnaissance des chiffres manuscrits isolés en s'appuyant sur la transformée en ondelettes discrète (TOD) et le classifieur SVM.
Nous présenterons en premier lieu les bases de données des chiffres manuscrits utilisées pour évaluer les techniques proposées, ainsi qu'une brève description de l'algorithme SVM, suivie d'une introduction de quelques concepts liés à la transformée en ondelettes 2D (TOD). Par la suite, nous exposerons trois de nos contributions.
La première porte sur le choix du type de l'ondelette et des sous-bandes images qui conviennent le mieux à la discrimination des chiffres manuscrits.
La seconde présente une étude comparative entre plusieurs techniques de caractérisation des chiffres manuscrits. Cette étude sera suivie de la description d'une méthode de caractérisation hybride associant la TOD à la technique basée sur les histogrammes de gradients orientés (HOG) pour la discrimination des chiffres manuscrits isolés.
Nous proposons en dernier, une technique de réduction et de sélection des caractéristiques qui combine l'Analyse en Composantes Principales (ACP) et la méthode de Sélection Séquentielle Ascendante (SFS) pour la sélection des caractéristiques pertinentes permettant la discrimination des chiffres manuscrits isolés.

Le chapitre 4 est dédié à la reconnaissance des chiffres manuscrits isolés à base des réseaux de neurones convolutifs. Nous présenterons d'abord quelques exemples de réseaux de neurones sur lesquels se sont fondés les modèles d'apprentissage profond. Ensuite, nous exposons brièvement les principaux modèles d'apprentissage profond, et d'une manière plus détaillée, les modèles neuronaux convolutifs. Nous proposons finalement trois réseaux de neurones convolutifs que nous avons développés pour la reconnaissance des chiffres

manuscrits. Ces réseaux sont conçus autour d'une même architecture. Le premier est un réseau CNN standard, constitué de couches de convolution destinées à extraire des caractéristiques à partir des images des caractères, et de couches complètement connectées (MLP) dédiées à la classification supervisée des caractères. Le second, semblable au premier, effectue la classification des caractères par l'intermédiaire du classifieur SVM à la place du MLP. Quant au dernier, il combine la transformée en ondelettes (TOD), le réseau de neurones convolutif (CNN) ayant une architecture identique à celles des deux réseaux précédents et le classifieur SVM.

Chapitre 1

Reconnaissance de l'écriture manuscrite

1.1. Introduction

L'écriture est une modalité de communication utilisée par l'homme depuis bien longtemps pour transmettre les informations à travers l'espace et le temps. Elle est aussi devenue un complément indispensable à la modalité orale, à la fois pour assurer la conservation des informations mais aussi parce qu'elle offre des possibilités de description mieux adaptées à certaines tâches. Ce domaine de l'écriture reste toujours à explorer, étant donnée sa complexité et sa diversité. L'écriture peut être sous différents styles; imprimée ou manuscrite.
Si le problème de l'écriture imprimée est résolu en grande partie, celui du manuscrit reste encore un sujet de recherche très actif. Cela est dû au caractère de variabilité qui le caractérise. En effet, l'écriture manuscrite est à la fois personnelle et universelle, particulière et générale, elle possède deux caractéristiques fondamentales: un aspect qui caractérise l'auteur de l'écrit, et un contenu sémantique qui caractérise le sens de ce qui est écrit.
Dans ce chapitre, nous donnons un aperçu des modalités de reconnaissance de l'écriture manuscrite, tout en expliquant les différentes étapes d'un système de reconnaissance de caractères. Nous présentons également les différents systèmes de reconnaissance existants, ainsi que les applications auxquelles ils sont dédiés.

1.2. De la reconnaissance de formes aux systèmes de lecture automatique

La reconnaissance de formes est un domaine majeur de l'informatique, dans lequel les recherches sont particulièrement actives. Il existe un très grand nombre d'applications qui peuvent nécessiter un module de reconnaissance, notamment dans les systèmes de traitement visant à automatiser certaines tâches de l'homme, comme par exemple, la reconnaissance de l'écriture. Dans ce domaine, il s'agit d'identifier la forme des caractères en utilisant un système de lecture automatique connu sous le nom d'OCR (Optical Character Recognition).

La reconnaissance optique de caractères (OCR) est un procédé informatique qui permet de reconnaître dans une image, les lettres composant un texte. Il consiste à transformer un fichier image en un fichier texte sous forme numérique. Cette tâche s'avère nécessaire pour offrir une manipulation plus aisée des données: archivage, indexation, recherche, etc...

Avec l'apparition des premiers ordinateurs, le premier système OCR développé est dédié à la reconnaissance des caractères typographiés latins et numériques. Par la suite, ces systèmes ont été étendus au cas de l'écriture manuscrite. Cependant, l'OCR s'est avéré être une tâche très complexe, à cause de la diversité de l'écriture manuscrite. De plus, l'identification de chaque caractère est un problème difficile à résoudre du fait qu'il est lié au problème de segmentation. C'est ainsi que les premiers travaux développés concernent essentiellement la reconnaissance de caractères manuscrits isolés. Parallèlement, d'autres systèmes de reconnaissance de caractères en ligne via des tablettes ont vu le jour.

Grâce aux avancées technologiques, les applications de l'OCR se sont multipliées et des systèmes de plus en plus complexes ont fait leur apparition. En effet, de nouvelles techniques d'intelligence artificielle telles que les réseaux de neurones, les modèles de Markov cachés HMMs (Hidden Markov Models) et la logique floue ont été utilisées dans ces systèmes [1,2]. Toutefois, on est encore loin de réaliser un système générique, même si les systèmes qui sont commercialisés trouvent leurs applications dans certains domaines bien particuliers.

1.3. Les différents systèmes de reconnaissance de caractères

La reconnaissance automatique de l'écriture manuscrite est un domaine de recherche très actif et qui a manifesté un intérêt remarquable dans l'accomplissement de beaucoup de tâches fastidieuses, répétitives et même très couteuses en temps comme celles que l'on rencontre dans les applications réelles de tri du courrier, la lecture du montant de chèques, la lecture des bordereaux, des bons de commande ou des feuilles d'impôts. Son objectif est de remplacer l'opérateur humain par un système capable d'effectuer le même travail mais en un temps beaucoup plus réduit.

L'automatisation de tout ou partie de ces tâches nécessite de doter la machine de la capacité à lire l'écriture manuscrite. Or la nature de l'écriture peut varier de façon considérable selon les scripteurs. A cet effet, aucun système n'est actuellement capable de reconnaître l'écriture manuscrite de façon universelle comme peut le faire tout être humain lettré. C'est ainsi que les chercheurs se sont tournées vers des systèmes dédiés, où l'on connaît à priori le style d'écriture que le système doit traiter. La figure 1.1 montre un panorama de systèmes de

reconnaissance de caractères existant. Ces systèmes diffèrent selon le type d'écriture et de l'application visée.

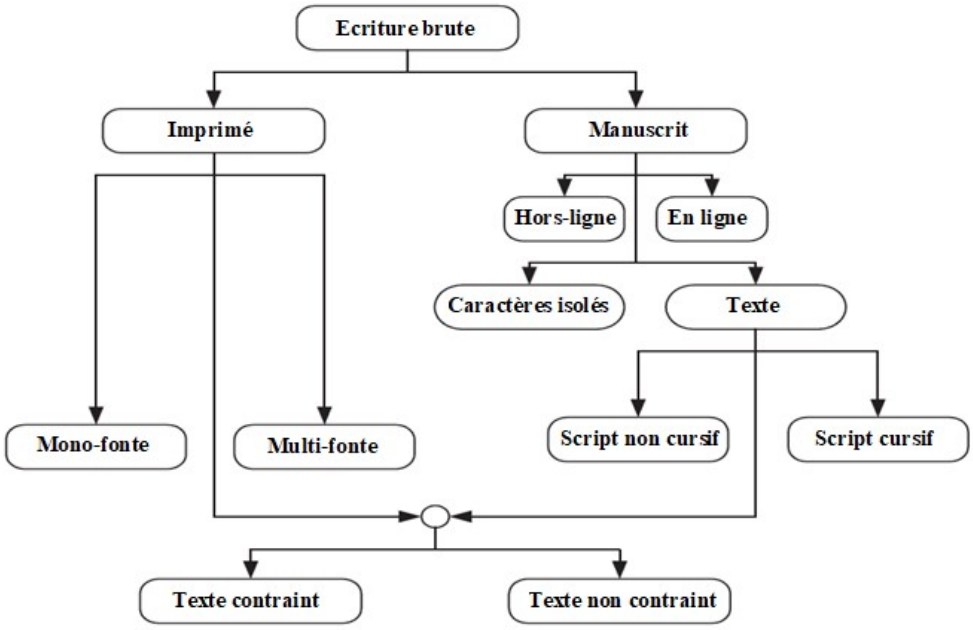


Figure 1.1. Panorama des types d'écriture

1.3.1. Reconnaissance de l'imprimé

Dans le cas de la reconnaissance de caractères imprimés, les systèmes OCR sont basés sur la notion de fontes. L'évolution de la typographie (police, taille, etc …) au cours des années, a permis l'introduction d'un très grand nombre de polices de caractères. Il existe deux types de systèmes de reconnaissance de l'imprimé, qui diffèrent selon le nombre de fontes utilisé: les systèmes mono-fontes, qui utilisent un seul type de fonte et les systèmes multi-fontes qui traitent du texte avec certaines fontes. Plus tard, d'autres systèmes traitant une variété de fontes ont fait leur apparition.

Pour un certain type de fontes, les caractères sont bien alignés et souvent bien séparés verticalement, ce qui simplifie la phase de reconnaissance, bien que certaines fontes qui se touchent parfois doivent être séparées à l'avance.

Les différents systèmes développés dans la littérature dans le cas de l'imprimé ont montré des performances assez bonnes avec des taux de reconnaissance obtenus s'approchant assez souvent des 100% [3]. Cela est dû à la forme des allographes (caractères) qui est constante pour une fonte donnée. Cependant, le problème de reconnaissance de l'imprimé n'est pas encore entièrement résolu; certaines difficultés sont liées à la qualité de l'acquisition. En effet, dans le cas d'images de mauvaise qualité (bruit, luminosité), dès que les conditions

d'acquisition ne sont plus optimales, le taux de reconnaissance diminue considérablement et les systèmes de reconnaissance sont mis à défaut. L'autre difficulté des systèmes de reconnaissance est liée à la dégradation des caractères. Dès que la qualité du document chute, le système présente un taux d'erreur important.

1.3.2. Reconnaissance du manuscrit

Contrairement à la reconnaissance de l'imprimé, la reconnaissance de l'écriture manuscrite soulève encore de nombreux problèmes, même si des avancées technologiques ont été enregistrées ces dernières années. Une des sources de ces problèmes est liée à l'extrême variabilité qui caractérise l'écriture. En effet, un texte manuscrit peut se présenter sous plusieurs formes: caractères isolés, texte cursif, texte non cursif, texte contraint ou non contraint. Cette diversité de l'écriture a été étudiée dans de nombreuses approches de reconnaissance proposées dans la littérature. Généralement, ces méthodes se distinguent par les données à traiter et par le style d'écriture utilisé.

Principalement, il existe deux approches de reconnaissance de caractères manuscrits: la reconnaissance en ligne et la reconnaissance hors ligne. Ces deux types de reconnaissance se distinguent par le mode d'acquisition (Figure 1.2) qui est lié au type du signal traité et les applications concernées.

Les systèmes de reconnaissance d'écriture en ligne ou dynamiques consistent à reconnaitre une forme d'une écriture représentée par la trajectoire du stylo, tandis que les systèmes de reconnaissance d'écriture hors ligne ou statiques consistent à reconnaitre ce qui a été écrit dans un document scanné.

1.3.2.1. L'écriture en ligne

L'écriture en ligne est obtenue lors de sa réalisation par une saisie en continu du tracé. Elle concerne les nombreux objets électroniques de poche, tels qu'une tablette électronique munie de stylo spécial permettant de saisir du texte sans clavier. Les données se présentent alors sous la forme d'une séquence de points ordonnés dans le temps correspondant à la position du stylo. Dans ce cas, le signal est de type monodimensionnel et le système de reconnaissance peut ainsi bénéficier de l'ensemble des techniques qui sont développées auparavant pour la parole [4]. La reconnaissance en ligne est plus avantagée que celle hors ligne à cause de la possibilité de correction et de modification de l'écriture de manière interactive; c'est à dire au moment même où le scripteur écrit. Les systèmes de reconnaissance en ligne sont

principalement employés dans le domaine de sécurité, tels que la certification d'auteur ou la vérification de signature [5].

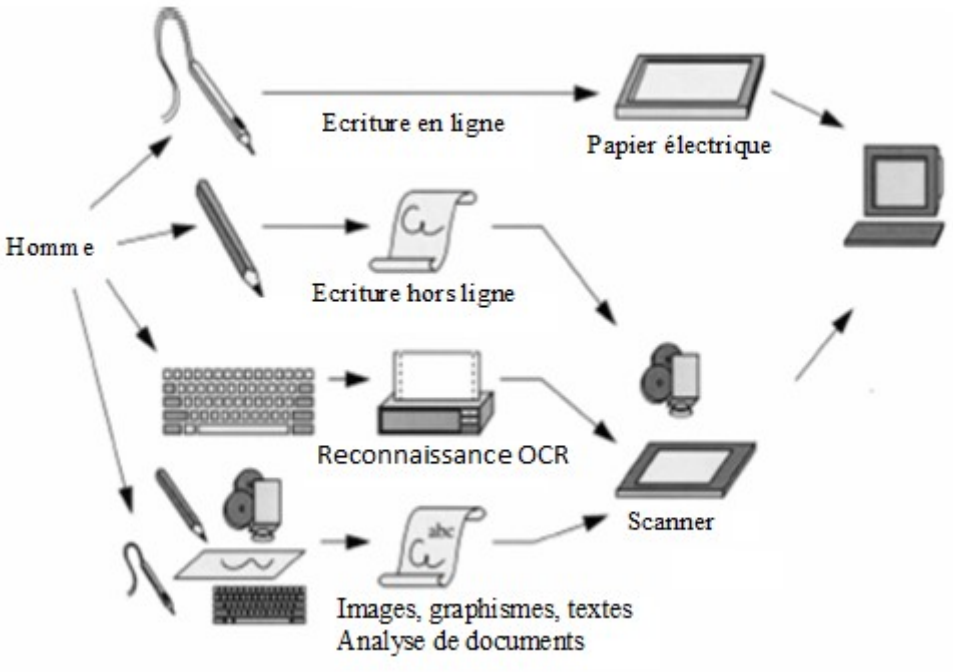


Figure 1.2. Systèmes de reconnaissance de l'écriture

1.3.2.2. L'écriture hors ligne

Dans le cas de systèmes hors ligne, la reconnaissance est effectuée une fois que l'écriture est présente sur un support en papier. L'acquisition des données se fait généralement à l'aide d'un scanner ou d'une caméra. Ainsi, l'information obtenue est bidimensionnelle. Contrairement aux systèmes en ligne, les systèmes hors ligne ne disposent pas de l'information temporelle et dynamique du tracé.

Au cours de ces dernières années, ce domaine a connu beaucoup de progrès et ce, grâce à l'évolution des techniques de traitement d'images et de reconnaissance de formes. L'application des techniques statistiques, telles que les réseaux de neurones et les modèles de Markov cachés ont permis d'obtenir des résultats satisfaisants dans ce domaine [6].

1.4. Complexité des systèmes de reconnaissance de l'écriture manuscrite

Dans les deux cas des systèmes en ligne et hors ligne, la reconnaissance de l'écriture manuscrite n'a pu progresser que grâce à une particularisation des problèmes à résoudre. Dans ce cas, le but recherché est de diminuer l'influence de la variabilité sur la reconnaissance. Ainsi, les chercheurs du domaine ont été amenés à s'intéresser à des applications particulières.

Pour un type d'application donnée, il est possible d'imposer à l'écriture à reconnaître, un certain nombre de restrictions et de contraintes telles que le nombre de scripteurs potentiels, le style d'écriture et la taille du vocabulaire utilisé.

- Le nombre de scripteurs: la réduction du nombre de scripteurs potentiels permet éventuellement de réduire la variabilité et d'apprendre les différents styles d'écriture [7]. Par contre, dans le contexte omni-scripteur, la difficulté s'accroit du fait que le système doit être capable de généraliser son apprentissage à n'importe quel type d'écriture (Figure 1.3).

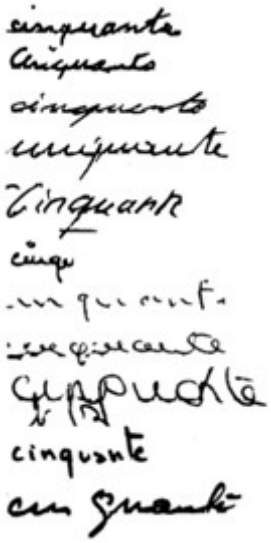


Figure 1.3. Exemples de variabilité de l'écriture manuscrite

- Le style d'écriture: la difficulté à reconnaître l'écriture augmente avec les différents styles d'écriture. Tappert [8] a établi une classification de l'écriture par ordre de difficulté croissante de reconnaissance: écriture scripte pré-casée, écriture scripte avec caractères espacés, écriture scripte libre, écriture cursive, écriture mixte cursive et scripte (Figure 1.4).

Le script caractérise l'écriture d'un mot en lettres séparées. Sa reconnaissance est ainsi largement simplifiée. Le cursif correspond à un mot où toutes les lettres sont attachées. L'écriture cursive est dite naturelle et présente l'avantage d'une liaison favorisant la fluidité et la rapidité du geste par opposition à l'écriture calligraphiée. Cependant, son apprentissage et sa reconnaissance est beaucoup plus complexe. Le type mixte est une combinaison des deux types précédents.

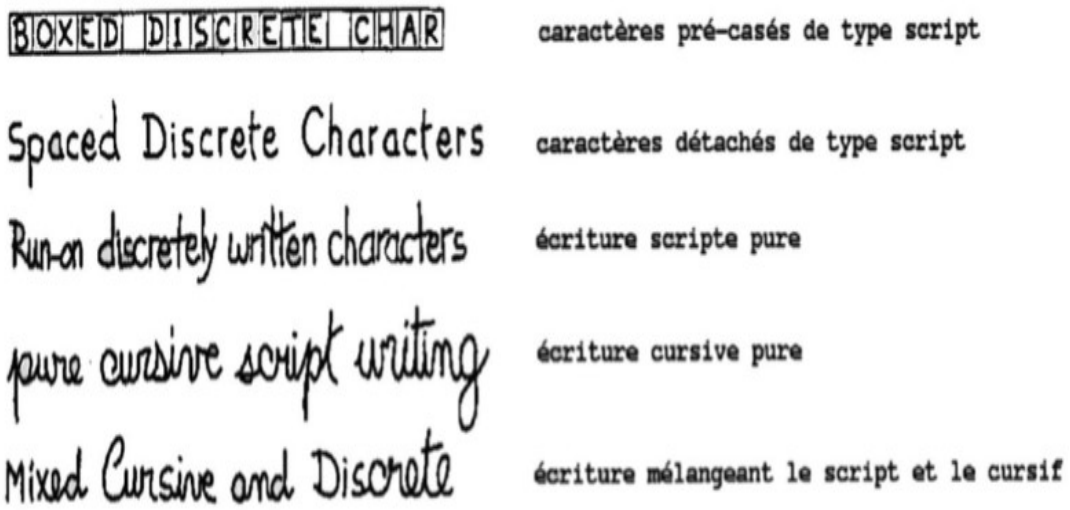


Figure 1.4. Différents types d'écriture manuscrite [8]

- Disposition spatiale du texte: la présentation d'un texte varie globalement entre deux formats: l'écriture contrainte correspondant à une écriture guidée par des cadres (les formulaires par exemple) et l'écriture non-contrainte correspondant à une écriture guidée exclusivement par le scripteur, donc extrêmement variable. Les écritures externes ou internes détachées (écriture en bâtons) sont, bien entendu, les plus aisées à traiter du fait de la séparation plus ou moins des lettres.

- Taille du vocabulaire: les systèmes de reconnaissance de textes sont souvent basés sur un lexique qui facilite grandement la lecture [9], surtout si celui ci possède un faible nombre de mots (cas des montants littéraux de chèques qui contiennent une trentaine de mots). La reconnaissance de mots est d'autant plus aisée que le nombre de mots dans le lexique est faible. Notons que dans le cas de la reconnaissance de séquences numériques qui s'élève à plusieurs milliers de mots (la présence d'un lexique est plus rare), la confrontation systématique n'est plus possible (cas de la reconnaissance de codes postaux).

- Langue: On en dénombre tout au plus 200 langues écrites sur plus 7100 existantes [10]. Si de nombreux systèmes de reconnaissance existent dans la langue latine, il n'en est pas de moins pour certaines langues comme le Japonais, le Chinois, le Perse, le Tifinagh, l'Arabe, les langues Indou, etc... Par exemple, dans le cas de l'écriture Arabe, la présence massive des points diacritiques, des hampes, des jambages et des ligatures entre les caractères complique beaucoup plus la tâche de l'OCR. En particulier, en inde, des recherches intensives ont été menées sur les problèmes de reconnaissance qui sont liés aux différentes écritures existantes.

En effet, plusieurs centaines de langues sont parlées et écrites en Inde, comme le Hindi, le Tamoul, le Télougou, le Bengali, le Kannada [11].Certains langages partagent des scripts communs, tandis que d'autres ont des scripts uniques. Ces divers scripts compliquent davantage la tâche du système de reconnaissance par la présence d'une variété de problèmes auxquels il faut apporter des solutions.

1.5. Organisation générale d'un système de reconnaissance hors ligne

L'architecture d'un système de reconnaissance de l'écriture manuscrite varie d'un système à un autre et en fonction de l'application envisagée. Dans ce qui suit, on présentera dans le cadre général, un système dédié à la reconnaissance optique de caractères (OCR). Ce système suit plusieurs étapes successives: acquisition, prétraitements, segmentation, extraction de caractéristiques, classification, décision, suivis éventuellement d'une phase de post-traitement. La figure 1.5 schématise le diagramme de ce système.

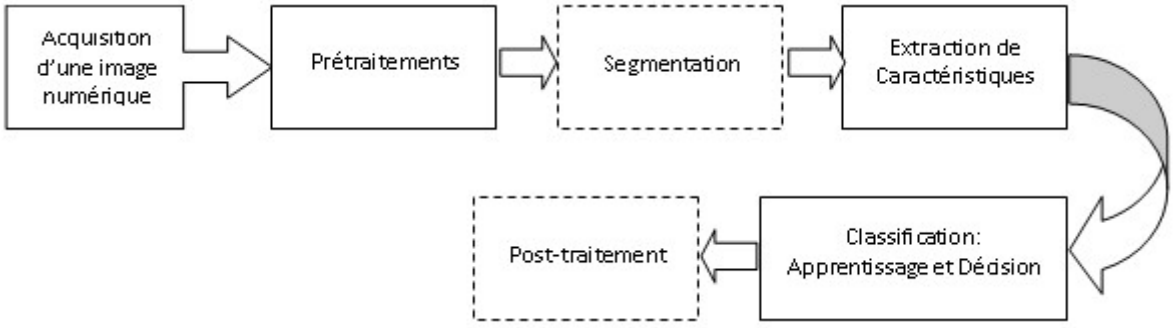


Figure 1.5. Schéma général de reconnaissance optique de caractères

1.5.1. Acquisition

La première étape dans tout système de reconnaissance est la phase d'acquisition. Elle consiste à récupérer l'image d'un texte au moyen des capteurs physiques tels que le scanner, la caméra ou tout autre dispositif approprié. L'image peut être binaire (noir et blanc), en niveaux de gris ou en couleur. Il est à noter que cette étape est très importante car la décision finale du système de reconnaissance dépend non seulement de la qualité du document utilisé ou des conditions d'acquisition mais également de la qualité du dispositif considéré pour effectuer cette tâche.

1.5.2. Prétraitements

Les prétraitements forment une série d'opérations effectuées sur l'image d'entrée ayant pour but de faciliter les traitements ultérieurs au niveau de chaque module du système de reconnaissance (segmentation, extraction de caractéristiques et reconnaissance). Cette étape est nécessaire lorsque les techniques utilisées pour extraire les caractéristiques ne sont pas robustes aux bruits ou aux déformations géométriques telles que la rotation. Elle n'est pas spécifique à un type d'application mais fait partie de tout système de reconnaissance de forme. Elle consiste à améliorer la qualité des images en éliminant les défauts dus à l'éclairage et au processus d'acquisition. Ces défauts sont liés soit à l'état de conservation du document physique original (tâches, trous, parties manquantes, qualité du stylo, couleur de l'encre, tracés ou écriture partiellement effacés), soit à l'étape de numérisation physique en elle-même (mauvaises conditions d'′eclairage, mauvaise orientation du document ou de la caméra, qualité du capteur …) [12]. Des traitements numériques permettent de corriger en partie ou totalement ces défauts de l'image, de manière à en améliorer la visualisation et l'analyse par des traitements ultérieurs.
Dans cette section, nous donnerons quelques techniques de prétraitement, les plus utilisées en reconnaissance de l'écriture manuscrite.

1.5.2.1. Suppression du bruit

Le bruit est très souvent présent sur les images de documents, car il peut apparaitre à différents endroits de la chaîne de numérisation: à l'impression, pendant la vie du document et à la numérisation. Ainsi, plusieurs traitements classiques sont mis en évidence, tels que la modification d'histogramme pour rehausser les contrastes; corriger la luminosité et les techniques de filtrage pour atténuer le bruit.

Les filtres les plus connus sont les filtres passe-haut, passe-bas, Sobel, Prewitt et Laplacien. Par exemple, le filtre passe-haut consiste à détecter la présence de traits d'écriture en recherchant les pixels correspondant au passage écriture/fond, donc à des transitions rapides. Cela permet de repérer les traits de l'écriture et de la différencier ainsi des pixels indésirables.
Le filtre médian est un autre type de filtre utilisé pour éliminer des pixels isolés sur l'image (bruit impulsionnel) [13].
Les opérateurs morphologiques en tant que techniques de lissage sont souvent rencontrés dans le traitement des documents manuscrits. Ils utilisent les opérateurs de base de dilatation et

d'érosion ainsi que leurs compositions (ouverture et fermeture) pour corriger les défauts liés soit à l'absence de points (trous) ou à une surcharge de points au niveau du tracé d'écriture manuscrite [14].

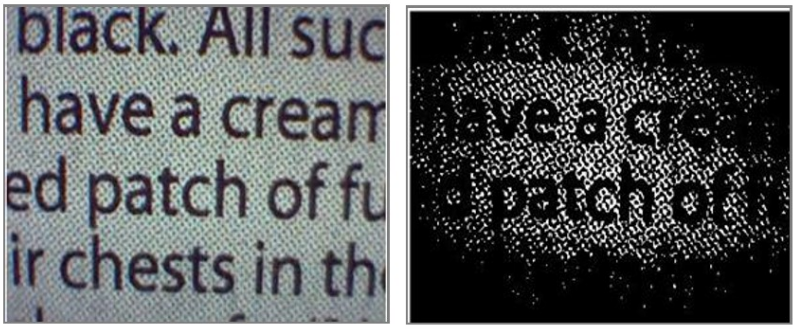
Toutefois, il est important de mentionner que ces traitements destinés à corriger certains défauts peuvent avoir des effets négatifs sur d'autres éléments de l'image. Par exemple, un filtre passe-bas permet d'éliminer certains bruits mais rend les tracés d'écriture flous. D'où la nécessité de manier ces traitements avec prudence.

1.5.2.2. Binarisation

Cette étape préliminaire à la segmentation a pour but de réduire la quantité d'information présente dans l'image et de garder uniquement l'information recherchée, telle que l'écriture par exemple. En reconnaissance de l'écriture manuscrite, la binarisation par seuillage est une technique très souvent utilisée. On peut distinguer deux types de seuillage d'images, global et local.

Le seuillage global trouve un seuil valable pour toute l'image, tel que les pixels dont la valeur est au-dessus de ce seuil sont considérés comme l'arrière-plan (blanc) et les autres comme l'information utile (noir). En général, on utilise un seuil de binarisation approprié qui traduit la limite entre les contrastes fort et faible dans l'image. Plusieurs techniques de seuillage global ont été proposées dans [15]. Parmi elles, l'algorithme d'Otsu [16] est souvent cité comme le plus performant.

Les techniques de binarisation globales ne conviennent pas à des documents complexes ou dégradés. De plus, dans le cas d'une mauvaise illumination du document, il sera difficile de trouver un seuil de binarisation global (Figure 1.6).



(a) (b)

Figure 1.6. Problème de seuillage global
(a) Image originale, (b) binarisation par Otsu

Pour pallier à cet inconvénient, les algorithmes de seuillage local ont été proposés. Ils procèdent par utilisation de différentes valeurs de seuil pour chaque pixel et ce, en exploitant l'information spatiale de ses voisins. Les techniques de seuillage local reportées dans la littérature, permettent de traiter des images extrêmement bruitées et de régler le problème de contraste de luminosité [17,18]. Une synthèse des méthodes de binarisation est proposée dans [19,20].

1.5.2.3. Redressement de l'inclinaison de la ligne de base de l'écriture

La correction de l'écriture est primordiale pour faciliter les traitements ultérieurs car un défaut d'orientation du document pendant l'acquisition, ou une écriture imprécise, peut conduire à une inclinaison de la ligne de base par rapport à l'axe horizontal. Rappelons que la ligne de base ou d'appui est une ligne sur laquelle reposent les caractères ne possédant pas des dépassements bas. L'idée de ce prétraitement est de redresser horizontalement les lignes d'écriture obliques à l'aide d'une transformation géométrique de type rotation isométrique des points de l'image (Figure1.7(a)).

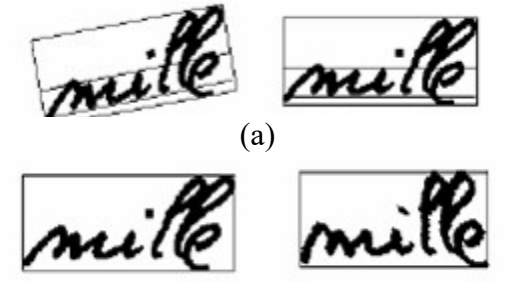
La ligne d'appui de l'écriture offre également des informations importantes pour les différentes étapes de la chaîne de reconnaissance. Pour la segmentation, elle guide le processus de détection de points de liaison entre caractères. En outre, elle permet de préciser les positions des diacritiques (les points, les accents) et de localiser les ascendants (tels que les caractères d, l, b) et les descendants (tels que j, g, p) et donc d'aider le processus de normalisation et d'extraction de caractéristiques.

Le redressement de l'écriture se fait généralement en deux étapes [21]. La première permet l'estimation de l'angle θ d'inclinaison globale de la ligne de base et la deuxième étape sert à corriger l'inclinaison par l'application d'une rotation de l'image d'un angle θ. A cet effet, des recherches intensives ont été consacrées à ce sujet. L'état de l'art présenté dans [22, 23], donne un aperçu sur les principales techniques utilisées pour la détermination de cette inclinaison. Parmi elles, on trouve la transformée de Hough et les histogrammes de projection.

1.5.2.4. Redressement des caractères inclinées ou penchées

Il existe des écritures droites, d'autres inclinées à gauche ou à droite par rapport à l'axe vertical ou encore un mélange des deux formes d'écriture. Le redressement de la pente de l'écriture permet son uniformisation, la rendant ainsi la plus indépendante possible des spécificités du scripteur. Ce prétraitement consiste à évaluer l'angle d'inclinaison locale d'un

caractère ou l'angle d'inclinaison moyenne des lettres à l'intérieur d'un mot et à effectuer une rotation en sens inverse (Figure 1.7(b)). Plusieurs méthodes sont disponibles pour la détection et la correction de cette inclinaison. Parmi elles, on peut citer l'approche basée sur l'analyse des histogrammes de projection qui est utilisée pour estimer l'angle d'inclinaison des caractères dans les directions proches de la verticale [24], l'approche basée sur le gradient des directions [25] et le filtre de Gabor [26].



(a)

(b)

Figure 1.7. (a) Redressement de la ligne d'appui, (b) Redressement des écritures penchées

1.5.2.5. Squelettisation

La squelettisation consiste à réduire l'épaisseur du tracé d'un caractère manuscrit en supprimant certains pixels pour ne garder que ceux situés le long de son axe (Figure 1.8). L'objectif est de garder la forme générale afin de pouvoir en extraire certaines caractéristiques de nature topologique comme les points terminaux (extrémités), points multiples (croisements), boucles, arcs, etc…

Cependant, il n'existe pas de définition unique du squelette dans l'espace discret. C'est pourquoi, on se limite à garantir un nombre de propriétés telles que l'unicité d'épaisseur du squelette, la préservation de la géométrie et de la topologie de l'objet.

Figure 1.8. Squelettisation

Généralement, le processus de squelettisation discret est basé sur l'amincissement. Dans cette catégorie de méthodes, le squelette est obtenu en supprimant itérativement les pixels frontières de l'objet. Ces pixels (les pixels simples et qui ne sont pas des pixels de fin) sont enlevés successivement [21] ou en parallèle [27], [28] ou bien encore à l'aide d'opérations morphologiques [29]. Ces méthodes conduisent à un squelette mince, géométriquement

représentatif (si les pixels de fin ont été correctement caractérisés), mais pas nécessairement centré.

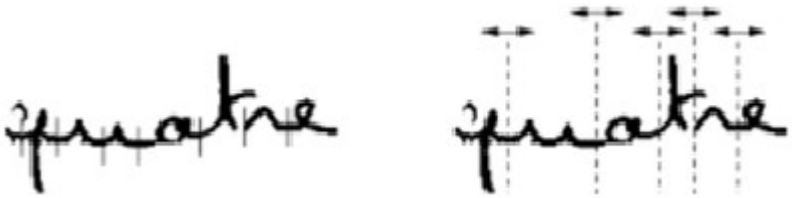
Une autre catégorie de méthodes de squelettisation dans le contexte d'images binaires est basée sur les transformées en distance ou sur l'extraction de carte de distances afin d'identifier l'axe médian d'un objet (caractère). La carte des distances consiste à associer à chaque pixel sa distance au pixel de contour le plus proche. Les maxima locaux de cette carte des distances correspondent aux pixels du squelette de l'objet [30]. Cependant, cette méthode dépend non seulement de la métrique employée mais aussi de l'orientation des balayages successifs [31].

1.5.3. Segmentation

La segmentation est une technique très importante dans le domaine du traitement d'images, en particulier en reconnaissance de l'écriture. A titre d'exemple, dans le traitement automatique de document, cela consiste à décomposer le document en ses différentes composantes à partir desquelles l'information nécessaire pour l'étape de reconnaissance est extraite. Suivant les applications envisagées et les objectifs à atteindre, il existe deux types de segmentation [32]: la segmentation externe et la segmentation interne.

La segmentation externe consiste à séparer les blocs de texte des blocs graphiques (les photographies, tableaux, graphes). En revanche, la segmentation interne ou analytique concerne les applications où les informations traitées sont les pages de texte. Dans ce cas, il s'agit de décomposer, tout d'abord, le bloc texte en lignes puis chaque ligne en mots et enfin à segmenter chaque mot en lettres ou en graphèmes. Pour les systèmes de reconnaissance utilisant le mot comme support, cette dernière segmentation n'est pas réalisée. A l'issue de l'étape de segmentation, l'image initiale est décomposée en une multitude d'imagettes correspondant soit aux mots, soit aux caractères.

Plusieurs approches de segmentation analytique sont proposées dans la littérature. Un état de l'art concernant ces approches est présenté dans [33,34]. Principalement, ces méthodes sont classées en deux catégories qui sont les méthodes explicites (segmentation puis reconnaissance) et implicites (segmentation-reconnaissance) (Figure 1.9).



Segmentation explicite Segmentation implicite

Figure 1.9. Types de segmentation

La première stratégie consiste à segmenter d'abord la chaîne de caractères et d'envoyer les fragments obtenus à un moteur de reconnaissance. Elle présente un point délicat quant au choix du meilleur chemin de segmentation. Elle nécessite alors des métriques d'évaluation très sélectives pour pouvoir choisir la bonne segmentation, car ce choix est définitif et ne peut pas être mis en cause ultérieurement. En revanche, la deuxième stratégie consiste à générer un ensemble de chemins de segmentation possibles et de choisir le meilleur en se basant sur les résultats fournis par le module de reconnaissance.

1.5.3.1. Segmentation explicite ou dissection

Cette approche de segmentation est basée sur l'extraction des caractéristiques globales telles que les minima locaux du contour, les espacements ou encore les points d'intersection d'un squelette qui permettront par la suite d'isoler dans l'image les différentes parties textuelles telles que les lignes, les mots puis les caractères [35].

Pour segmenter les lignes de texte, plusieurs approches ont été proposées dans la littérature. La plus utilisée est celle qui s'appuie sur l'analyse de l'histogramme de projection horizontale [36,37]. Cette méthode consiste à projeter horizontalement les pixels d'une image pour détecter les vallées ou les minima (nombre de points noirs par ligne) sur cet histogramme. A partir de ces vallées, on peut déterminer les espacements entre les lignes. Ainsi, un espacement constitue un point de segmentation si sa valeur est inférieure à un seuil préalablement fixé par l'utilisateur (Figure 1.10).

Parallèlement à la segmentation en lignes, la segmentation en mots puis en caractères est basée sur la technique des histogrammes de projection verticale (Figure 1.11). De nombreuses approches de segmentation basées sur cette technique ont été proposées pour la reconnaissance des caractères. Dans [38], l'approche développée combine la technique des histogrammes de projection verticale et un ensemble de règles floues pour extraire les points de segmentation potentiels. Une approche similaire est proposée par Shinde [39] pour segmenter des textes imprimés. Cette segmentation est basée les profiles de projection.

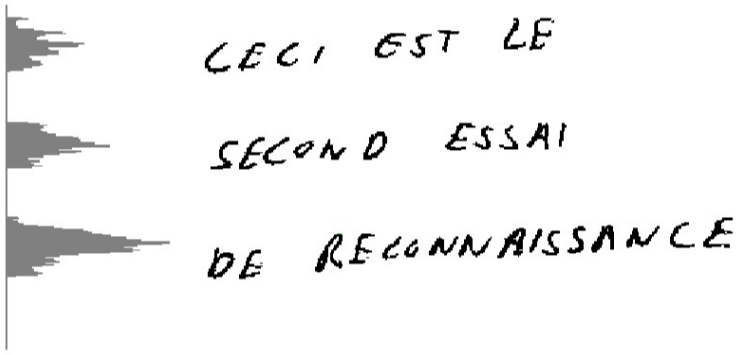


Figure 1.10. Calcul de l'histogramme de projection horizontale [37]

Figure 1.11. Calcul de l'histogramme de projection verticale [37]

En pratique, le nombre d'applications utilisant seulement l'information fournie par l'approche de segmentation basée sur des projections est limité, car il est difficile de segmenter de façon idéale vu l'augmentation du nombre de scripteurs rencontrés et les différents types d'écriture utilisés. En effet, dans le cas de documents complexes non contraints, les lignes de texte peuvent être inclinées et même ondulées [40] (Figure1.12).

Pour faire face à cette limitation, certains chercheurs se sont investis dans ce domaine pour proposer de nouvelles méthodes à partir de celles existantes. Par exemple, pour l'extraction des lignes d'un texte manuscrit, l'approche proposée dans [41] est basée sur les minima du profil de projection partielle et un suivi de contour partiel. Une autre approche est basée sur l'application d'un filtre gaussien et un histogramme de projection verticale pour segmenter les lignes des documents manuscrits latins [42]. Dans [43], une méthode combinant la transformée de Hough et la technique de projection a été proposée pour la segmentation des caractères. Une autre méthode qui permet de détecter les points de segmentation potentiels, est celle basée sur des réservoirs [44]. Elle consiste à détecter les vallées et les collines séparant deux chiffres liés appartenant à la même composante connexe. Cependant, cette méthode est peu efficace dans le cas des lettres, car ces dernières sont de tailles variables.
Avec les avancés récentes en apprentissage profond, dans [45], des chaines de chiffres manuscrits peuvent être reconnues sans avoir recours au module de segmentation. Cette

approche consiste à combiner 4 réseaux de neurones convolutifs (CNNs) pour prédire la taille et la classe de la séquence des chiffres manuscrits.

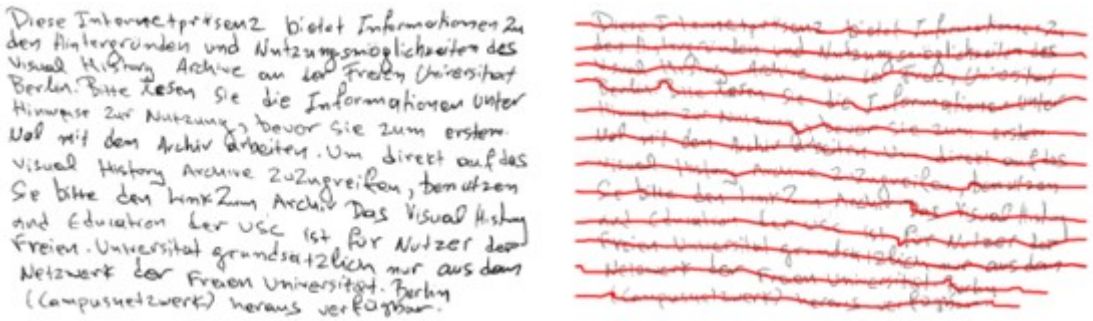


Figure 1.12. Quelques problèmes d'extraction de lignes de texte manuscrit

1.5.3.2. Segmentation implicite

L'approche de segmentation implicite ou de segmentation-reconnaissance est basée sur le module de reconnaissance pour valider les hypothèses de segmentation. Dans ce cas, la segmentation et la reconnaissance sont réalisées conjointement [46,47]. Son principe est basé sur la recherche dans l'image des composantes de caractères ou graphèmes correspondant aux classes prédéfinies.

Cette approche a l'avantage d'éviter les difficultés de segmentation explicite rencontrées pendant la recherche des points de segmentation potentiels. Dans le cas d'un tracé cursif, la localisation de ces points qui permettent d'identifier les lettres du tracé est un problème qui relève du paradoxe de Sayer [48]: «pour reconnaitre des lettres dans un tracé cursif, il faut d'abord passer par la segmentation, mais il est difficile de segmenter le tracé avant de les avoir reconnus ».

Dans ce cas, les méthodes implicites sont utilisées comme alternative pour intégrer les processus de segmentation et de reconnaissance. Cela consiste à découper le mot en sous parties représentées par des graphèmes et à utiliser la reconnaissance de ces graphèmes pour corriger les erreurs de segmentation. Ce découpage est effectué implicitement par l'utilisation d'une fenêtre glissante se déplaçant sur l'image [49]. Pour chaque position, un segment représentant soit un caractère ou un graphème est identifié par le module de reconnaissance en utilisant son vecteur de caractéristiques. Ainsi, l'ensemble de ces vecteurs forme les séquences d'observations qui sont modélisées par les modèles de Markov cachés (HMMs) de caractères. Cette méthode reste la plus répandue et la plus puissante pour la modélisation de séquences de caractères, où chaque caractère est représenté par un HMM [50]. Ensuite, la reconstruction du mot est réalisée par concaténation des modèles qui le composent.

Ce type d'approche produit de bons résultats, mais elle est très coûteuse en temps de calcul, puisque toutes les hypothèses générées doivent être évaluées. De plus, elle présente un problème qui est lié au choix du nombre de segments. Pour améliorer les résultats de reconnaissance, d'autres travaux de recherche proposent l'hybridation des HMMs avec d'autres modèles, tels que les réseaux de neurones [51].

1.5.3.3. Problèmes liés à la segmentation

La procédure de segmentation s'avère être une phase critique car les erreurs commises dans cette phase se répercuteront sur les étapes ultérieures, pouvant ainsi dégrader les performances globales du système de reconnaissance. En effet, l'objectif de la segmentation repose sur sa stratégie de décision qui définit la meilleure option de coupure et permet d'isoler le mieux possible un caractère afin qu'il soit reconnu comme tel par le module de reconnaissance. Il suffit qu'un caractère ne soit pas bien séparé pour qu'il ne soit pas reconnu. Selon [52], la multiplicité des polices et la variation des styles d'écriture empêchent de stabiliser les seuils de séparation, conduisant à la génération de blancs inexistants ou au contraire à l'ignorance de blancs séparateurs de mots. D'après [53], il n'existe aucune méthode permettant une segmentation exacte d'un mot manuscrit en lettres. Néanmoins, selon le type d'objet à reconnaître (mots cursifs, caractères isolés, symboles), il existe de nombreuses techniques permettant d'obtenir une bonne segmentation, à défaut d'être parfaite.

1.5.4. Extraction de caractéristiques

L'extraction des caractéristiques a pour but de décrire une donnée (image d'un caractère) par un vecteur de caractéristiques. La similarité ou dissimilarité entre les caractères est alors mesurée sur la base de leurs vecteurs de caractéristiques.

C'est une étape importante d'un système efficace de reconnaissance de caractères. En effet, l'utilisation d'un classifieur très performant ne peut compenser une représentation mal adaptée ou peu discriminante. La difficulté de cette étape provient du fait que la qualité d'une représentation ne peut se juger que sur un problème particulier (reconnaissance de chiffres, lettres, symboles).

S'il est difficile de sélectionner à priori un extracteur de caractéristiques pour un problème donné, on ne peut pas pour autant choisir d'extraire toutes les caractéristiques possibles. Il y'a plusieurs raisons à cela: d'une part, l'utilisation d'un grand nombre de caractéristiques avec une méthode de classification implique pour la phase d'apprentissage un nombre d'exemples

qui augmente d'une manière exponentielle avec la dimension de la représentation (nombre de caractéristiques). D'autre part, la présence de caractéristiques non discriminantes dégrade les performances du classifieur. Pour remédier à ce problème, des méthodes de sélection de caractéristiques ont été développées, visant à limiter la représentation aux descripteurs les plus pertinents, et améliorer ainsi les performances du système. Trier et al. [54] ont précisé que le choix et la sélection du type de caractéristiques lors de l'extraction est la partie la plus importante dans le système de reconnaissance, car celle-ci dépend du type de problème à résoudre.

Dans la littérature, un grand nombre de travaux sur les techniques d'extraction de caractéristiques pour la reconnaissance de caractères ont été proposées. Généralement, ces techniques peuvent être classées en trois catégories: structurelles et topologiques, statistiques et celles basées sur des transformations globales.

1.5.4.1. Techniques structurelles et topologiques

Les techniques structurelles et topologiques peuvent être regroupées dans la même catégorie puisqu'elles sont liées à l'analyse physique ou morphologique du caractère à reconnaitre. Le principe consiste à décomposer l'image du caractère en extrayant des formes élémentaires, appelées aussi primitives. Ces primitives peuvent être extraites des données brutes ou à partir d'une autre représentation du caractère (squelette ou contour). Elles sont généralement très robustes vis à vis de la rotation, la translation, l'homothétie. Il s'agit principalement [55]:

- de segments de droite (strokes), leurs nombre, la direction et la pente,
- des arcs, des boucles,
- des concavités et les convexités dans les quatre directions principales,
- les points d'intersections et croisements,
- les points terminaux,
- la hauteur et la largeur du caractère,
- le nombre de points diacritiques et leur position par rapport à la ligne de base dans le cas de l'écriture arabe,
- les jambages (descendants) et les hampes (ascendants),
- rapport de la hauteur sur la longueur de la fenêtre englobant l'image,
- mesure de surfaces et périmètres.

1.5.4.2. Techniques statistiques

Les méthodes statistiques cherchent à extraire à partir de la distribution des pixels noirs de la forme de chaque caractère (ou du mot) des caractéristiques qui la décrivent localement. Ces caractéristiques permettent ainsi de représenter chaque caractère sous forme d'un vecteur de caractéristiques. L'ensemble de ces vecteurs définit ainsi un nouvel espace de représentation des données initiales et qui sont ensuite utilisés par un classifieur pour distinguer les différentes classes à l'intérieur de cet espace.

Différents types de caractéristiques ont été suggérées pour la reconnaissance de caractères. Parmi elles, on cite celles basées sur les histogrammes de projection, les profils, le zonage, les moments géométriques invariants et les moments de Zernike [56].

1.5.4.3. Techniques basées sur des transformations globales

En plus de ces deux grandes familles de caractéristiques citées précédemment, d'autres sont extraites à la suite d'une transformation de l'image. Dans cette famille, nous distinguons les caractéristiques basées sur la transformée de Fourier, les filtres de Gabor et la transformée en ondelettes [57].

1.5.5. Classification

La classification consiste à déterminer la classe d'appartenance d'un caractère soumis à son entrée. Elle se déroule en deux phases: l'apprentissage et la décision (ou reconnaissance).

1.5.5.1. Phase d'apprentissage

Dans cette phase, on dispose d'un ensemble d'échantillons de caractères décrits par des vecteurs de caractéristiques. Le rôle du classifieur est de créer par apprentissage à partir de ces vecteurs, des modèles de références pour chaque classe, en regroupant les caractères ayant des caractéristiques similaires. Autrement dit, il s'agit de partitionner l'espace des caractéristiques en classes distinctes.

Le choix de l'algorithme de classification dépend des informations dont on dispose sur les données à traiter. Trois types d'algorithmes de classification sont à distinguer: les algorithmes de classification supervisée, non supervisée et semi supervisée.

1.5.5.1.1. Classification supervisée

La classification supervisée est utilisée dans le cas où les classes des échantillons (caractères) sont connues à priori. Chaque caractère de cet échantillon est identifié. Dans ce cas, les caractères d'une même classe constituent les prototypes de cette classe. L'objectif de la classification supervisée consiste alors à construire à partir de l'ensemble des données prototypes, un modèle mathématique pour chaque classe. Ces modèles peuvent être un simple centre de gravité, une distribution statistique ou une surface séparatrice entre les classes.
En reconnaissance de l'écriture manuscrite et en particulier de caractères manuscrits, les méthodes de classification supervisées sont largement utilisées du fait que les classes sont connues à priori. Parmi ces méthodes, on cite la règle des k plus proches voisins (kppv) [58], la décision Bayesienne [59], les machines à vecteurs de support (SVM) [60] et les réseaux de neurones [61].
Les k plus proches voisins est une méthode qui n'induit aucun modèle mathématique à partir des prototypes. Son principe consiste à affecter une forme inconnue à la classe majoritaire parmi ses k plus proches voisins étiquetés. Cette affectation est basée sur une mesure de distance évaluée dans l'espace des caractéristiques. Son inconvénient réside dans l'augmentation du temps de calcul requis.

L'approche Bayésienne suppose que la distribution des prototypes de chaque classe est décrite par une fonction de densité de probabilité conditionnelle connue. Ces prototypes sont alors utilisés pour estimer les paramètres de ces distributions durant la phase d'apprentissage. La règle de Bayes permet de déterminer à partir des probabilités conditionnelles, la probabilité à postériori qu'un caractère appartienne à une classe, donc d'identifier un caractère.

Les machines à vecteurs de support (SVM) sont des méthodes qui cherchent les surfaces (hyperplans) qui séparent au mieux les prototypes des classes. Leur principe repose sur la détermination de vecteurs de supports; formes remarquables de la base d'apprentissage qui permettront de discriminer les caractères. Les SVMs offrent des performances intéressantes dans la reconnaissance de caractères manuscrits [62].

Grâce à leurs propriétés de parallélisme et d'adaptation, les réseaux de neurones tels que le perceptron multicouches (MLP) sont particulièrement utilisés pour reconnaitre des formes globales telles que les lettres et les chiffres isolés [63]. Récemment, les réseaux de neurones convolutifs (CNN) qui sont des cas particuliers des MLP, ont attirés beaucoup d'attention et

ce, grâce au succès qu'ils ont montré dans une application de reconnaissance de chiffres manuscrits [1]. On assiste actuellement à un regain d'intérêt à ce type d'architecture [64].

1.5.5.1.2. Classification non supervisée

En classification non supervisée, on ne dispose d'aucune information à priori sur l'appartenance des échantillons aux classes. Autrement dit, les caractères de l'échantillon sont inconnus. Dans ce cas, le classifieur s'auto-organise pour créer des classes de données (caractères) de sorte que les objets (caractères) d'une même classe soient très similaires et que des objets (caractères) de classes différentes ne le soient pas. L'algorithme des k-moyennes (k-means) et sa version floue (Fuzzy c-means) sont des exemples de classifieurs non supervisés et souvent utilisés pour la reconnaissance de caractères manuscrits [65,66].

1.5.5.1.3. Classification semi supervisée

La classification semi supervisée est située entre les deux techniques de classification précédentes à savoir la classification supervisée et non supervisée, qui sont complètement opposées en terme de connaissances à priori dont elles disposent. En effet, la classification semi supervisée semble être plus réaliste en intégrant une faible quantité d'informations à priori relative à l'appartenance des données à chaque classe. L'objectif de cette approche est d'attribuer une classe à chaque donnée qui n'est pas étiquetée au départ et ce, en faisant propager l'information des données étiquetées sur l'ensemble des données non étiquetées. Dans le cas de reconnaissance de chiffres manuscrits isolés, différentes techniques de classification semi supervisée ont été proposées [67].

1.5.5.2. Phase de reconnaissance ou de décision

L'étape de reconnaissance est une phase qui permet d'identifier un caractère inconnu en l'assignant à l'une des classes définie pendant l'apprentissage. A partir de la description du caractère par son vecteur de caractéristiques, le module de reconnaissance cherche parmi les modèles de référence en présence, celui qui est le plus proche de la forme de ce caractère pour lui attribuer une étiquette. Cette décision peut se faire en utilisant soit une distance (kppv), en déterminant la plus forte probabilité à postériori (Bayes) ou en déterminant de quel coté de la frontière se situe le caractère (SVM).

La reconnaissance peut conduire à un succès si la réponse est unique (un seul modèle répond à la description de la forme du caractère). Elle peut être ambigüe ou confuse si la réponse est

multiple (plusieurs modèles correspondent à la description). Enfin elle peut également conduire à un rejet de la forme si aucun modèle ne correspond à sa description. Dans les deux premiers cas, la décision peut être évaluée avec un taux de reconnaissance.

1.5.6. Post-Traitement

Le post-traitement est la dernière étape du processus de reconnaissance de mots isolés [68]. Son objectif principal est de vérifier l'exactitude des résultats issus de la phase de reconnaissance, pour s'assurer de la cohérence des mots trouvés puis des phrases. Il permet ainsi de corriger l'étape de reconnaissance par l'apport de connaissances d'ordre lexical, grammatical et sémantique. Généralement, les connaissances lexicales sont les plus utilisées dans un système de reconnaissance de l'écriture pour vérifier la présence d'un mot dans un dictionnaire (lexique ou vocabulaire). Il contient la liste des mots que le système de reconnaissance pourra être amené à reconnaitre.

- ✓ Lexical: valide la reconnaissance en ne retenant que des mots du dictionnaire, et en rejetant les listes de lettres inconsistantes.
- ✓ Syntaxique: se base sur l'utilisation de la grammaire, la construction de phrases. Elle consiste à détecter les séquences de mots qui n'ont pas d'usage dans la langue.
- ✓ Sémantique: fait intervenir la notion de sens; c'est à dire que le système doit comprendre le sens de la phrase reconstruite.

Notons que certaines techniques de prétraitement ainsi que les étapes de segmentation et de post-traitement ne sont pas nécessairement utilisées dans tous les systèmes de reconnaissance de caractères. Certains systèmes supposent que l'entrée est déjà débarrassée du bruit. D'autres utilisent directement des caractères qui sont préalablement segmentés. Cependant, il existe peu de travaux de recherche dans lesquels l'étape de post-traitement est considérée, car la tâche de vérification prend beaucoup de temps surtout dans le cas d'un vocabulaire de très large taille.

1.6. Applications de la reconnaissance de caractères

La multiplication des applications de reconnaissance de caractères est liée au besoin de limiter la quantité et le volume des documents à stocker et ainsi de faciliter leur traitement. En effet, la reconnaissance de caractères évite toutes les phases de codage manuel réalisées par des opérateurs en utilisant le clavier comme interface. Elle aura donc pour effet d'économiser du temps, de simplifier une tâche fastidieuse et de limiter la quantité de papier utilisée.

La reconnaissance optique de caractères est un domaine très vaste qui a engendré de nombreuses applications, telles que celles reportées dans [69]. Parmi ces principales applications, on peut citer:

- L'aide à la lecture pour les non-voyants: les systèmes de reconnaissance associés à des synthétiseurs vocaux permettent la compréhension de documents et livres pour les aveugles [70].
- La lecture de formulaires: ce type d'application concerne le traitement des formulaires pré-imprimés, tels que les formulaires de constats d'assurance, fiches de renseignements, bons de commande, formulaires de recensement, formulaires de déclaration d'impôts, formulaires médicaux [71]. L'utilisation d'un système de reconnaissance capable de lire directement les données dans les zones réservées permet d'effectuer rapidement la saisie de ces documents.
- La gestion automatique des chèques bancaires ou postaux: l'une des applications industrielles les plus importantes de l'écriture manuscrite est le traitement régulier d'un grand nombre de chèques, d'où la nécessité d'automatiser cette tâche. Certains systèmes se contentent de lire uniquement le montant numérique, tandis que d'autres systèmes associent également le montant littéral pour valider la lecture des chèques. De plus, pour les besoins de sécurité, les signatures peuvent être éventuellement vérifiées.
- Le tri automatique de courrier: dû à une énorme quantité de courrier qui circule quotidiennement, le développement des machines de tri automatique de courrier s'avère indispensable. C'est l'une des tâches les plus difficiles à réaliser vu que les données à traiter étant essentiellement du type manuscrit. Pour réduire le temps de traitement et ainsi augmenter l'efficacité du traitement, le système doit associer la lecture du code postal à la lecture de l'adresse de destination.
- La vérification et l'identification de la signature: malgré le progrès technologique, jusqu'à ce jour, la signature reste le moyen le plus utilisé pour authentifier un document, valider un contrat ou une transaction financière. La signature est donc reconnue comme mode de validation associé à l'identité d'une personne [72].

1.7. Conclusion

Dans ce chapitre, nous avons donné un aperçu sur la reconnaissance de l'écriture manuscrite. Ainsi, nous avons passé en revue un certain nombre de techniques utilisées pour la mise en œuvre d'un système de reconnaissance de caractères. Les différents modules le composant ont été décrits: l'acquisition, les prétraitements, la segmentation, l'extraction de caractéristiques et

la classification. Nous avons également rappelé les principales contraintes qui compliquent la reconnaissance de l'écriture manuscrite. Elles sont liées à la qualité de l'acquisition, à la qualité du document et à la variabilité des caractères manuscrits.

Afin d'atténuer l'influence de ces facteurs sur le système de reconnaissance et augmenter ainsi les chances d'aboutir à une bonne reconnaissance, les chercheurs s'accordent à dire que l'étape d'extraction de caractéristiques est probablement le module le plus important dans la chaîne de reconnaissance. En effet, l'utilisation d'un classifieur très performant ne peut compenser une représentation mal adaptée ou peu discriminante. La difficulté de cette étape provient du fait que la qualité d'une représentation ne peut se juger que sur un problème particulier.

Dans le prochain chapitre, nous allons nous intéresser aux techniques d'extraction de caractéristiques utilisées en reconnaissance de caractères manuscrits isolés.

Chapitre 2

Extraction de caractéristiques: Etat de l'art

2.1. Introduction

Le processus d'extraction de caractéristiques consiste à représenter ou coder un caractère présenté sous forme d'une image par un vecteur de caractéristiques ou descripteurs. La représentation la plus simple consiste à construire un vecteur constitué d'autant de composantes qu'il y a de pixels dans l'image. Leur niveau de gris définit alors le codage de ce vecteur. Cependant, ce codage extrêmement simple ne tolère aucune déformation ni aucun bruit. En effet, l'image d'un même caractère bruitée, translatée ou inclinée n'aura pas le même code. Par conséquent, les systèmes basés sur cette représentation sont inefficaces pour effectuer une reconnaissance robuste.

Des techniques de caractérisation des formes de caractères ont été proposées afin de permettre aux systèmes de classification d'être plus efficaces aux transformations et aux dégradations. Elles cherchent à définir un codage, déduit d'un ensemble de mesures, qui distingue le mieux possible les différents types de caractères. Ces mesures doivent être les plus génériques possible pour ne pas dépendre des scripteurs et aussi suffisamment précises pour identifier chaque caractère.

Dans ce chapitre, nous présentons un état de l'art sur les caractéristiques les plus utilisées en reconnaissance hors ligne de caractères manuscrits et imprimés.

2.2. Les techniques de caractérisation

Plusieurs techniques d'extraction de caractéristiques sont proposées pour la reconnaissance des caractères. Elles sont généralement classées de trois manières différentes [73].

Dans la première taxinomie, une distinction est faite entre les différentes caractéristiques selon le format de représentation de l'image: binaire, niveaux de gris et selon la forme du caractère (contour ou squelette) [54]. Une synthèse de ces caractéristiques est présentée dans le tableau 2.1.

Caractéristiques	Image de gris	Image binaire	Contour	Squelette
Appariement	X	X		X
Motifs déformables	X	X		X
Transformation unitaire	X	X		
Transformation log-polaire	X	X		
Moments géométriques	X	X		
Moments de Zernike	X	X		
Ondelettes	X	X		
Histogrammes de projection		X		
Masques	X	X		
Profil du contour			X	
Code de Freeman			X	
Spline			X	
Descripteurs de Fourier			X	X
Zonage	X	X	X	X

Tableau 2.1. Taxinomie des méthodes d'extraction des caractéristiques adoptée par Trier [54]

Pour une image en niveaux de gris, on peut extraire des caractéristiques du type zonage, moments géométriques, moments de Zernike, ondelettes. Pour une image binarisée, les caractéristiques qui peuvent être extraites sont les histogrammes de projection, les moments géométriques, les moments de Zernike, des caractéristiques du type zonage, etc. Sur un contour, des caractéristiques de type projection des profils, Fourier, ondelettes, zonage, code de Freeman peuvent être extraites. Des caractéristiques comme Fourier, motifs déformables sont également extraites à partir d'un caractère représenté par son squelette.

La deuxième classification regroupe les caractéristiques en deux familles: les caractéristiques globales et les caractéristiques locales [74]. Les caractéristiques globales cherchent à représenter la forme générale d'un caractère et sont donc calculées sur des images relativement grandes. Les différentes caractéristiques qui peuvent être extraites sont les moments invariants, les projections, les profils, les transformations globales.

Les caractéristiques locales sont calculées plutôt localement dans une région restreinte de l'image, et ce, par application d'une fenêtre glissante qui parcourt les pixels de l'image avec un pas qui dépend du type de caractéristiques et de la taille de l'image. Dans ce cas, ces caractéristiques représentent des points de singularité permettant d'identifier un caractère

donné. On retrouve parmi ces caractéristiques, les boucles, les intersections, les arcs concaves et convexes ainsi que leurs positions.

Dans la troisième taxinomie, adoptée par la plupart des auteurs, les caractéristiques peuvent être regroupées en trois classes: structurelles et topologiques, statistiques et transformations globales [32,75]. Dans ce qui suit, nous allons développer cette classification de représentation des caractéristiques.

2.2.1. Caractéristiques structurelles et topologiques

Les caractéristiques structurelles représentent les propriétés topologiques et géométriques de la forme. Ces caractéristiques sont extraites à partir de la représentation de la forme du caractère par son squelette ou son contour. Il s'agit essentiellement de concavités, convexités, occlusions, ascendants, descendants, composantes connexes, segments de droites et leurs attributs (position, orientation, ...), mesures de pentes, arcs, boucles, croisements, paramètres de courbures, points extremums et points terminaux, longueur et épaisseur des traits, surfaces et les périmètres [76]. L'extraction des ces caractéristiques nécessite une squelettisation préalable du caractère qui conditionne fortement sur les résultats de la reconnaissance. Néanmoins, elles sont très robustes vis à vis de la rotation, translation et permettent de prendre des décisions rapides dans la reconnaissance de l'écriture [77]. Plus récemment, dans [78], une approche basée sur la décomposition de formes perceptuelles a été proposée pour la reconnaissance des chiffres manuscrits. L'idée repose sur la représentation de la forme du caractère avec quatre primitives visuelles qui sont extraites à partir du contour de l'image du caractère. Pour reconnaitre un caractère, l'approche développée utilise ces primitives dans un ensemble de règles de classification.

2.2.2. Caractéristiques statistiques

Les caractéristiques statistiques décrivent une forme par un ensemble de mesures statistiques [79]. Elles permettent de donner des informations locales sur le contenu de l'écriture. Le choix d'un type de caractéristiques pour une application donnée est très délicat en raison de la grande diversité qu'elles présentent. Dans ce qui suit, nous présentons quelques caractéristiques; certaines d'entre elles sont très populaires et même largement utilisées dans de nombreux travaux de recherche, tandis que d'autres ne sont introduites dans le domaine de reconnaissance des caractères que très récemment.

2.2.2.1. Profils

Les quatre profils (haut, bas, droite, gauche) sont obtenus par l'intermédiaire de sondes appliquées sur le caractère [37]. Pour le profil gauche d'un caractère, on lance des sondes depuis le bord gauche de l'image qui s'arrêtent lorsqu'elles rencontrent le premier pixel noir. Les abscisses des sondes constituent le profil gauche du caractère (Figure 2.1).

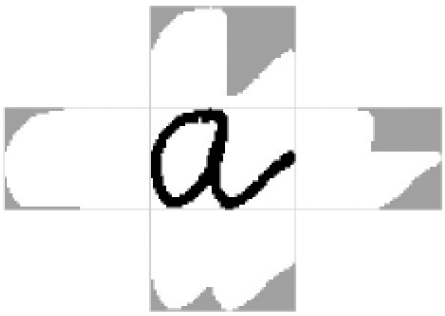


Figure 2.1. Les 4 profils d'un caractère [80]

2.2.2.2. Histogrammes de projection

Les histogrammes de projection horizontale et verticale servent à déterminer les minima locaux qui sont des points de séparation entre les lignes et entre caractères respectivement [81]. Comme ils peuvent être aussi utilisés comme caractéristiques. Cette technique consiste à calculer séparément le nombre de pixels de l'image d'un caractère selon les lignes et les colonnes [82]. Plus précisément, ils sont obtenus par projections horizontale et verticale des pixels de l'image (Figure 2.2). Les caractéristiques utilisées peuvent être représentées directement par les valeurs des histogrammes ou bien extraites de ces histogrammes en cherchant à détecter les pics par exemple. L'inconvénient majeur de cette méthode est sa sensibilité à la rotation, c'est-à-dire que si on fait pivoter le caractère, les histogrammes seront notablement différents.

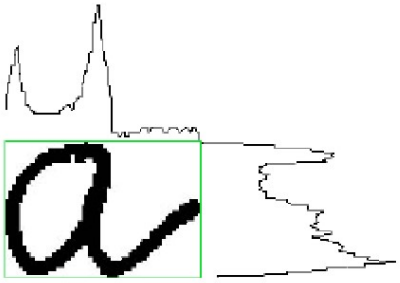
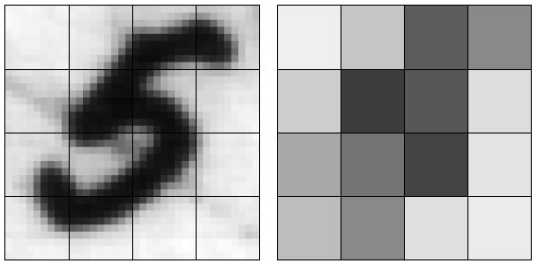


Figure 2.2. Histogrammes des projections horizontale et verticale [80]

2.2.2.3. Technique de zonage

La technique de zonage (zoning en anglais) consiste à découper d'abord une image en zones. Ensuite les caractéristiques telles que la moyenne des niveaux de gris ou la densité des pixels sont extraites dans chaque zone (Figure 2.3). Cette technique très connue en reconnaissance de formes, a été aussi introduite dans des systèmes de reconnaissance de caractères [83].



(a) (b)

Figure 2.3. Zonage de l'image d'un caractère [54]
(a) division de l'image en 4x4 zones,
(b) la moyenne des niveaux de gris dans chaque zone

La difficulté majeure liée à cette approche réside dans le choix du nombre de zones et à la manière de découper l'image en zones. Un état de l'art sur cette approche est dressé dans [84]. D'autres mesures basées sur le calcul des distances sont proposées dans [85]. L'image est découpée en zones de mêmes tailles, et dans chacune d'elles, deux types de distances sont extraits. Elles sont basées sur le calcul des distances moyennes en considérant:

- le centre de gravité de l'image du caractère et chaque pixel présent dans une zone.
- le centre de gravité d'une zone et chaque pixel présent dans celle-ci.

Le vecteur de caractéristiques englobe l'ensemble de toutes ces distances.

2.2.2.4. Transformation des caractéristiques invariantes à l'échelle

Cette technique, connue sous le nom anglais SIFT (Scale Invariant Feature Transform) a été développée par David Lowe [86] pour caractériser les régions de l'image qui sont remarquables visuellement par des caractéristiques. Ces caractéristiques doivent être invariantes par transformations de translation, rotation, changement d'échelle et partiellement invariantes à la luminance [87]. SIFT combine un détecteur de points d'intérêt et une extraction de caractéristiques locales basée sur la distribution des gradients. Elle est appliquée dans plusieurs domaines, et notamment en reconnaissance optique de caractères chinois [88], arabes [89], latins [90] ou encore indiens [91].

Cette méthode se déroule en deux étapes: détection de points d'intérêt et extraction des caractéristiques. Dans la première étape, on effectue une convolution entre l'image de départ I et un filtre gaussien de moyenne nulle et d'écart type σ qui représente le facteur d'échelle.

$$L(x, y, \sigma) = g(x, y, \sigma) * I(x, y) \quad (2.1)$$

avec:

$$g(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (2.2)$$

Cette opération a pour effet de lisser l'image et d'atténuer les contours de faible intensité, permettant ainsi de ne garder que les détails importants. Pour détecter les extrema locaux, on calcule une différence des gaussiennes DoG (Difference of Gaussian) entre deux images correspondant à deux échelles consécutives. Son expression est donnée par:

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \quad (2.3)$$

k est un paramètre fixe de l'algorithme qui dépend de la finesse de la discrétisation de l'espace des échelles.

Le résultat de la différence de deux images consécutives lissées constitue une approximation du Laplacien appelée LoG (Laplacian of Gaussian). Celui ci est utilisé comme un filtre passe-haut pour la détection des contours.

Ensuite, pour détecter les extrema, chaque pixel des images DoG est comparé à ses 26 voisins: 8 voisins situés dans la même échelle et 9 voisins sur les deux échelles inférieure et supérieure (Figure 2.4). Si un pixel est un extremum local, c'est-à-dire sa valeur est supérieure ou inférieure à celles de ses voisins, alors, il est sélectionné en tant que point d'intérêt.

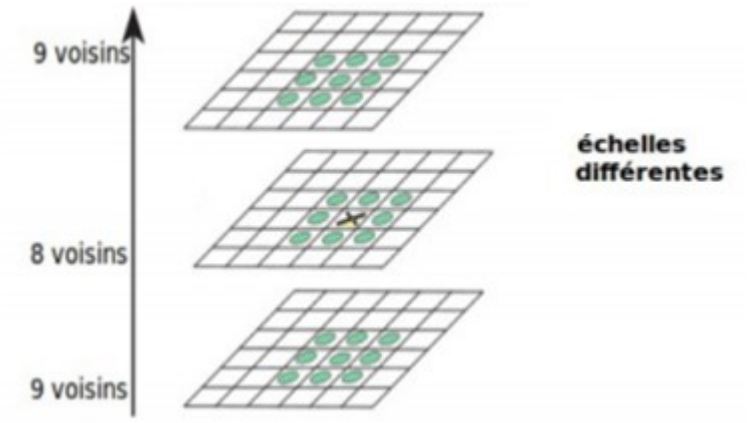


Figure 2.4. Détection des extrema dans les DoG [87]

Après avoir détecté les points d'intérêt, la deuxième étape consiste à calculer le descripteur SIFT. Pour construire ce descripteur, une région de taille 16x16 pixels est sélectionnée autour de chaque point détecté. Ensuite, pour chaque pixel de cette région, une amplitude A et une orientation θ du gradient sont calculées comme suit:

$$A(x,y) = \sqrt{\big(L(x+1,y) - L(x-1,y)\big)^2 + \big(L(x,y+1) - L(x,y-1)\big)^2} \tag{2.4}$$

$$\theta(x,y) = arctg\frac{\big(L(x,y+1) - L(x,y-1)\big)}{\big(L(x+1,y) - L(x-1,y)\big)} \tag{2.5}$$

Puis, ces amplitudes sont pondérées par une Gaussienne pour donner plus de poids aux gradients les plus proches du point d'intérêt. Par la suite, la région sélectionnée est divisée en 4x4 zones de 4x4 pixels chacune (Figure 2.5). Dans chaque zone, un histogramme d'orientation du gradient quantifié sur 8 intervalles est construit, puis pondéré par l'amplitude de ses pixels. Enfin, ces 16 histogrammes sont concaténés, puis normalisés pour former un descripteur SIFT de dimension 128 (4x4x8). Depuis, quelques variantes ont été proposées, telles que PCA-SIFT [92] et SURF [93] pour surmonter certains problèmes du descripteur SIFT.

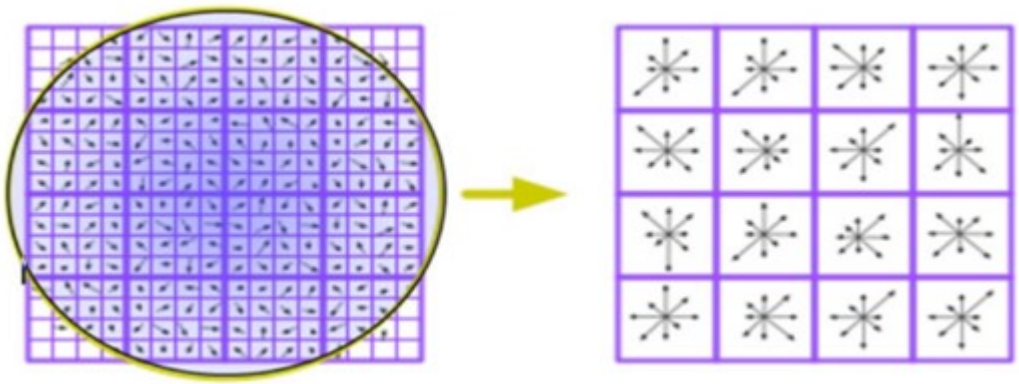


Figure 2.5. Calcul du descripteur SIFT [87]

2.2.2.5. Caractéristiques robustes accélérées

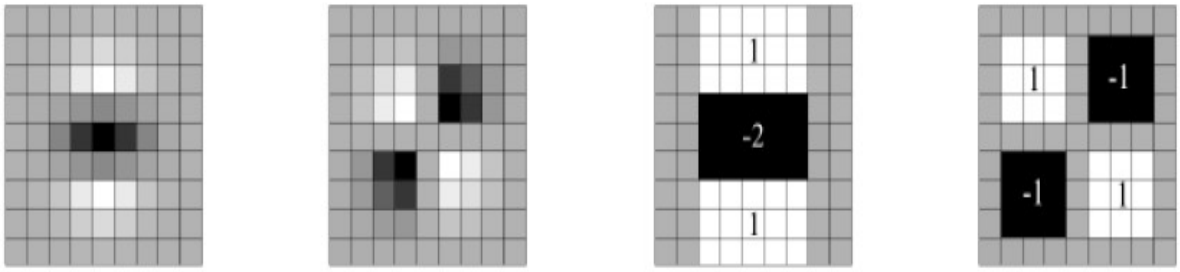
La caractérisation par le détecteur SURF (Speeded Up Robust Features) reste parmi les méthodes les plus connues pour la détection d'objets dans les images [93]. Cette méthode est composée de deux étapes principales: la première consiste à détecter des points d'intérêt, et la seconde permet de décrire ces points à l'aide d'un vecteur de 64 caractéristiques.

La détection de points d'intérêt est obtenue là où le déterminant de la matrice Hessienne atteint un maximum. La matrice Hessienne $H(x, y, \sigma)$ à l'échelle σ est définie par ses dérivées partielles secondes, données par:

$$H(x, y, \sigma) = \begin{bmatrix} L_{xx}(x, y, \sigma) & L_{xy}(x, y, \sigma) \\ L_{xy}(x, y, \sigma) & L_{yy}(x, y, \sigma) \end{bmatrix} \quad (2.6)$$

avec: $L_{xx}(x, y, \sigma)$ est obtenu par le produit de convolution de la dérivée seconde de la Gaussienne $\frac{\partial^2}{\partial x^2} g(x, y, \sigma)$ avec l'image $I(x, y)$. $L_{yy}(x, y, \sigma)$ et $L_{xy}(x, y, \sigma)$ sont obtenus de la même manière.

Dans le but de réduire le temps de calcul, ces Gaussiennes sont ensuite approximées par des masques de convolution suivant les différentes directions. La figure 2.6 montre l'approximation utilisée.



(a) (b) (c) (d)

Figure 2.6. Dérivées partielles secondes de la gaussienne
(a) et (b) discrétisées, (c) et (d) approximées

Cette étape apporte une invariance des points d'intérêts par rapport à la mise à l'échelle, du fait qu'ils sont extraits à différentes échelles par application de filtres Gaussiens de différentes tailles. Finalement, seuls les points d'intérêt dont le déterminant de la matrice Hessienne est positif et qui sont maximum locaux dans un voisinage 3 x 3 x 3 (abscisse x ordonnée x échelle) sont conservés.

Une fois que les points d'intérêt sont détectés, l'étape suivante consiste à calculer le vecteur de caractéristiques. Cela consiste à construire autour de chaque point d'intérêt, une région de forme carrée découpée en 16 zones de taille 4x4 chacune. Dans chaque zone, on calcule les réponses à une ondelette de Haar suivant les directions horizontale Tx et verticale Ty (Figure

2.7). Ensuite, ces réponses sont sommées pour former un vecteur à 4 composantes ($\sum Tx$, $\sum Ty$, $\sum|Tx|$ et $\sum|Ty|$). Ainsi, chaque point d'intérêt peut être décrit par un descripteur ayant une dimension de 4x4x4=64.

Figure 2.7. Ondelettes de Haar suivant les directions horizontale et verticale
(zone noire: -1, zone blanche: +1)

2.2.2.6. Méthode de sacs de caractéristiques visuels

La méthode de sacs de caractéristiques visuels BOF (Bag of visual features) est fondée sur le principe de sacs de mots visuels BOW (Bag of Visual Words) utilisé pour la catégorisation des documents [94]. BOW consiste à représenter un document texte par un vocabulaire (ou dictionnaire) composé d'un ensemble non ordonné de mots.

La méthode BOF, proposée dans le cas de la catégorisation d'images [95] est basée sur le descripteur SIFT (ou SURF) et l'algorithme k-moyennes. Elle est composée de trois étapes: détection de points d'intérêt, description de ces points et création du vocabulaire visuel.

La détection de points d'intérêt consiste à localiser dans une image les régions qui sont visuellement remarquables. Plusieurs détecteurs peuvent être utilisés, mais le plus souvent, la matrice Hessienne est employée [96].
Pour représenter chaque point d'intérêt, un descripteur SURF est extrait autour de chacun de ces points. Souvent, le nombre de descripteurs utilisé pour créer un vocabulaire visuel est très élevé. Afin de surmonter ce problème, on fait appel à l'algorithme des k-moyennes pour regrouper ces descripteurs en un nombre réduit de classes. Ainsi, l'ensemble de ces classes forment le vocabulaire visuel, et leurs centres correspondent aux caractéristiques visuelles (BOF). Enfin, la représentation vectorielle de l'image peut être obtenue par un histogramme, et chacune de ses composantes représente l'importance d'une caractéristique visuelle dans cette image.

2.2.2.7. Histogrammes de gradients orientés

Les histogrammes de gradients orientés (HOG) sont introduits initialement par Dalal pour la détection des piétons [97]. Cette approche consiste à décrire l'apparence locale d'un objet en

discrétisant l'orientation du gradient dans l'image à l'aide d'histogrammes. Ainsi les contours, donc la forme des objets, sont codés ce qui permet de les reconnaître. Cette méthode, relativement populaire au sein de la communauté scientifique, a été appliquée avec succès pour la reconnaissance optique de caractères manuscrits [98,99,100].

Le principe du descripteur HOG, consiste dans un premier temps, à diviser l'image de l'objet à caractériser en cellules qui sont elles mêmes regroupées en blocs qui se recouvrent (Figure 2.8). Dans chaque cellule, les orientations des gradients de l'ensemble des pixels sont calculées et collectées dans un histogramme monodimensionnel. Ensuite, ces histogrammes sont normalisés localement en intensité au niveau de chaque bloc. Plusieurs versions existent pour effectuer cette normalisation. Certaines agissent sur la normalisation des histogrammes en considérant uniquement les gradients des orientations [101], tandis que d'autres comptabilisent aussi le module du gradient au lieu des occurrences seules [102].

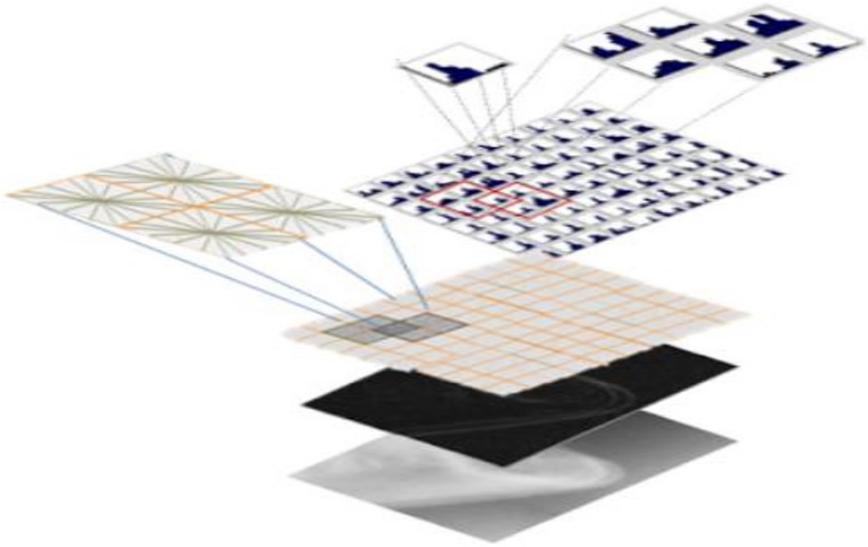


Figure 2.8. Illustration de la technique des histogrammes des gradients orientés

L'histogramme formé à partir des orientations du gradient des pixels à l'intérieur de chaque cellule est quantifié en 9 intervalles qui sont de même tailles couvrant 180 ou 360 degrés d'orientation.

Il existe deux possibilités pour attribuer les pixels à chaque intervalle d'orientation:

- Le nombre de pixels concernés par un intervalle est le nombre de pixels appartenant à un contour et dont l'orientation du gradient appartient à l'intervalle.
- Le nombre de pixels concernés par un intervalle est le nombre de pixels dont l'orientation du gradient appartient à l'intervalle pondéré par le module du gradient.

Finalement, la concaténation de l'ensemble de ces histogrammes forme un vecteur de caractéristiques appelé HOG.

2.2.2.8. Motifs binaires locaux

Les descripteurs basés sur les motifs binaires locaux ont été proposés par Ojala pour la classification des textures [103]. Grace à leur succès, ils ont été exploités dans d'autres applications telles que la reconnaissance de caractères imprimés et manuscrits [104,105].

Le principe de la méthode basée sur les LBP consiste à assigner un code binaire à chaque pixel de l'image en fonction de ses 8 voisins, disposés sur une grille qui peut être de forme carrée. La valeur du niveau de gris g_c du pixel central est utilisée pour seuiller ses pixels voisins g_p afin de générer un motif binaire (Figure 2.9). Les pixels de ce motif binaire sont alors multipliés par des poids, puis additionnés afin d'obtenir un code LBP:

$$LBP(x_c, y_c) = \sum_{p=0}^{P-1} s(g_p - g_c)2^P \tag{2.7}$$

avec:

$$s(g_p - g_c) = \begin{cases} 1 \; si \; g_p - g_c \geq 0 \\ 0 \; si \; g_p - g_c < 0 \end{cases} \tag{2.8}$$

(x_c, y_c) sont les coordonnées du pixel central g_c, P est le nombre de pixels voisins et g_p le pixel voisin d'indice p. Dans le cas d'un voisinage 3*3, on a $P = 8$.

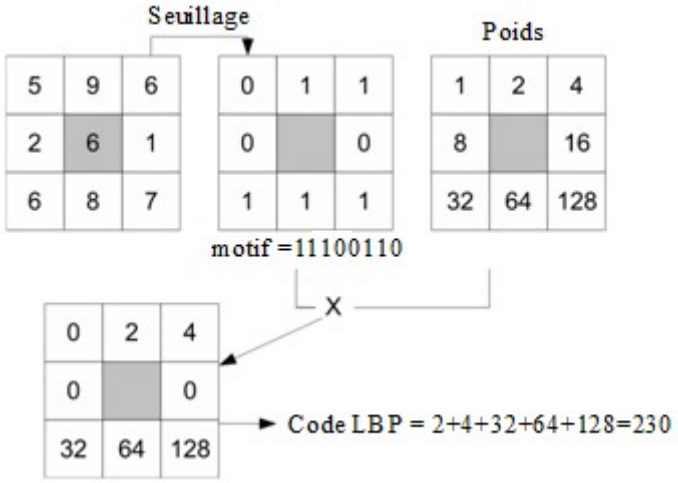


Figure 2.9. Exemple de calcul du code LBP

Le masque de poids utilisé pour calculer le code LBP prend des valeurs entre 2^0 et 2^{P-1}. Une fois que le code LBP est déterminé pour chaque pixel, un histogramme est construit à partir de ces codes pour former ainsi un descripteur LBP.

Des extensions de cette méthode ont été proposées dans [106] en introduisant un voisinage circulaire de rayon R et possédant un nombre P de voisins (Figure 2.10). Dans ce cas, on parle de $LBP_{P,R}$. Les niveaux de gris des points échantillonnés sur le bord de ce cercle sont comparés avec celle du pixel central. Les valeurs des P points échantillonnés, régulièrement espacés sont obtenues par interpolation. Ensuite, pour décrire l'image, un histogramme du nombre d'occurrences des différents motifs binaires possibles est construit. L'invariance à la rotation permet d'avoir au total 256 motifs, ce qui correspond à 2^8 possibilités. Ojala [106] a montré qu'un sous ensemble de ces motifs pour lesquels le nombre de transitions 0/1 est au plus égal à 2 (motifs uniformes) contient plus de 90% d'information. Dans ce cas, pour un rayon R = 1, un ensemble de 9 motifs binaires uniformes (00000000, 00000001, 00000011, 00000111, 00001111, 00011111, 00111111, 01111111, 11111111) peuvent être déterminés. Ensuite, en considérant les différentes rotations possibles de ces 9 motifs, nous obtenons un ensemble de 59 motifs uniformes. Cette méthode, connue sous le nom de LBP^{u2}, est adoptée dans ce travail pour caractériser les caractères.

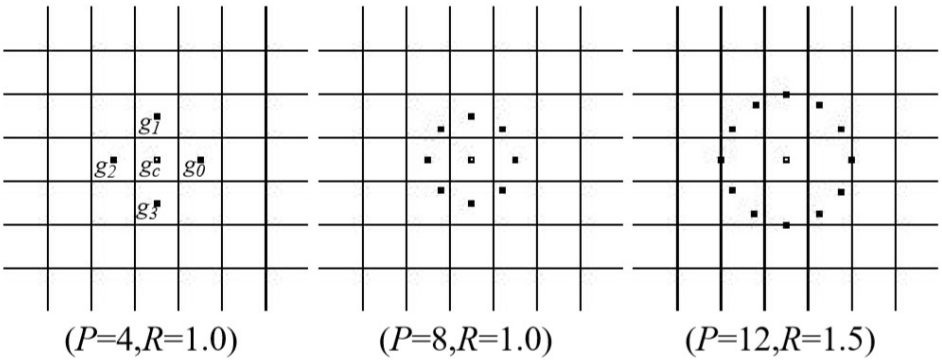


Figure 2.10. Exemple de voisinages avec différentes valeurs de (P, R)

2.2.3. Caractéristiques basées sur des Transformations globales

Contrairement aux approches basées sur l'apparence visuelle de la forme à reconnaitre, les techniques de transformations cherchent à extraire à partir des images, des informations non visibles. Elles utilisent une transformation globale de manière à changer l'espace de représentation et ainsi faciliter l'extraction de caractéristiques pertinentes. Cette approche est

largement utilisée dans le domaine de reconnaissance de formes, en particulier celle de l'écriture manuscrite.

2.2.3.1. Moments invariants et moments de Zernike

Les moments ont été fréquemment utilisés comme descripteurs pour la reconnaissance des caractères manuscrits [74,107]. Ils permettent de décrire l'aspect global d'une forme de caractère à l'aide des propriétés statistiques. Ces descripteurs sont robustes aux transformations géométriques comme la translation, la rotation et le changement d'échelle.

2.2.3.1.1. Moments invariants

Les moments de Hu [108] sont composés de combinaisons de moments statistiques centrés et normalisés d'ordres 2 et 3. Ils sont invariants à la translation, à la rotation et au changement d'échelle.

Pour une image $I(x, y)$, on définit les moments statistiques d'ordre $(p + q)$ comme:

$$M_{pq} = \sum_{x=1}^{m} \sum_{y=1}^{n} x^p y^q I(x, y), \quad \forall (p, q) \in \mathbb{N} \tag{2.9}$$

Soit:

$$\bar{x} = \frac{M_{10}}{M_{00}}, \ \bar{y} = \frac{M_{01}}{M_{00}} \tag{2.10}$$

Les moments statistiques centrés normalisés d'ordre $(p + q)$ sont obtenus comme suit:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{(p+q+2)/2}} \tag{2.11}$$

avec:

$$\mu_{p,q} = \sum_{x=1}^{m-1} \sum_{y=1}^{n-1} (x - \overline{x})^p (y - \overline{y})^q I(x, y), \qquad \forall (p, q) \in \mathbb{N} \tag{2.12}$$

et $p + q \geq 2, \forall (p, q) \in \mathbb{N}$.

Les 7 moments de Hu s'expriment de la façon suivante:

$$\phi(1) = (\eta_{20} + \eta_{02}) \tag{2.13}$$

$$\phi(2) = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \tag{2.14}$$

$$\phi(3) = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \tag{2.15}$$

$$\phi(4) = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \tag{2.16}$$

$$\phi(5) = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - 3\eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (2.17)$$

$$\phi(6) = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \quad (2.18)$$

$$\phi(7) = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (2.19)$$

2.2.3.1.2. Moments de Zernike

Pour surmonter le problème lié aux moments invariants, Teague [109] a proposé les moments orthogonaux de Zernike comme descripteurs. De nombreuses études comparatives ont montré la supériorité de ce descripteur par rapport à d'autres approches, telles que celles basées sur les moments géométriques [110].

Les moments de Zernike Z_{pq} correspondent à la projection de la forme $f(x, y)$ décrivant une image sur un espace de polynômes V_{pq}.

$$Z_{pq} = \frac{p+1}{\pi} \sum_{x=0}^{N} \sum_{y=0}^{M} V_{pq}^{*}(x, y) f(x, y) \quad (2.20)$$

(*) définit le complexe conjugué. La base de polynômes de Zernike ZP est définie sur un cercle unité par:

$$ZP = \{V_{pq}(x, y)/(x^2 + y^2) \leq 1\} \quad (2.21)$$

où le polynôme complexe V_{pq} d'ordre p et de répétition q est défini avec $p \in N^+$ et $q \in Z$, tel que $p - |q|$ soit pair et $|q| \leq p$. En coordonnées polaires $(x = \rho cos\theta, y = \rho sin\theta)$, V_{pq} prend la forme suivante:

$$V_{pq}(x, y) = V_{pq}(\rho, \theta) = R_{pq}(\rho) exp(jq\theta) \quad (2.22)$$

avec:

$$R_{pq}(\rho) = \sum_{s=0}^{\frac{p-q}{2}} (-1)^s \frac{(p-s)!}{s!\left(\frac{p-2s+q}{2}\right)!\left(\frac{p-2s-q}{2}\right)!} \rho^{p-2q} \quad (2.23)$$

et:

$$\theta = tan^{-1}\left(\frac{y}{x}\right), \rho = \sqrt{x^2 + y^2}, \qquad j = \sqrt{-1} \quad (2.24)$$

Les moments sont invariants par rotation, translation et changement d'échelle (après normalisation de la taille de la forme). De plus, grâce à l'exploitation d'une base de fonctions orthogonales, ces moments sont peu corrélés.

Il est à noter que l'ordre des moments possède une grande influence sur la conservation de l'information angulaire. Plus l'ordre est élevé, plus les variations angulaires décrites sont fines [27]. La figure 2.11 en donne une illustration.

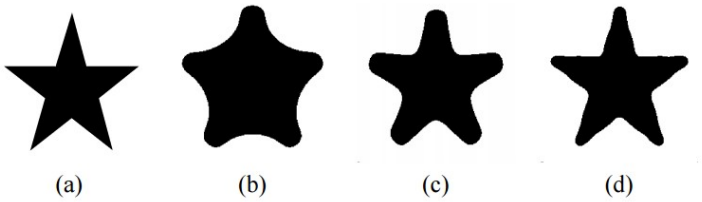


Figure 2.11. Exemple de reconstructions à partir des descripteurs de Zernike, (a) image d'origine, (b) reconstruction d'ordre 10, (c) reconstruction d'ordre 20, (d) reconstruction d'ordre 40 [27]

Cependant, le calcul des moments de Zernike pose beaucoup de problèmes. Ces difficultés sont liées à la normalisation de l'espace de coordonnées de l'image et au choix de l'ordre du polynôme.

2.2.3.2. Descripteurs de Fourier

Les descripteurs de Fourier sont largement utilisés en reconnaissance des caractères manuscrits [111,112]. Ils sont obtenus à partir de la décomposition en série de Fourier d'une signature extraite du contour d'une forme. Parmi les signatures les plus courantes, on trouve les positions complexes [112], les variations angulaires ou courbure [113] et la distance au centre de gravité de la courbe [114]. Zhang et Lu ont montré que cette dernière solution est plus performante. Pour extraire la signature selon la distance au centre de gravité, le contour d'une forme doit être d'abord échantillonné en N points, puis la distance r au centre de gravité de coordonnées (x_c, y_c) est établie pour chaque point k du contour de coordonnées $(x(k), y(k))$.

$$r(k) = \sqrt{(x(k) - x_c)^2 - (y(k) - y_c)^2}\ , k = 0, 1, \dots, N-1 \qquad (2.25)$$

avec: $x_c = \frac{1}{N}\sum_{k=0}^{N-1} x(k),\ \ y_c = \frac{1}{N}\sum_{k=0}^{N-1} y(k)$

La transformée de Fourier discrète (TFD) de $r(k)$ est donnée par:

$$D_n = \frac{1}{N}\sum_{k=0}^{N-1} r(k) exp\left(-\frac{j2\pi nk}{N}\right), n = 0, \dots, N-1 \qquad (2.26)$$

A partir de la TFD, seul le module $|D_n|$ est conservé vu son invariance à la rotation.

Le descripteur de Fourier $F_D$est formé de $\frac{N}{2}$ coefficients $|D_n|$ distincts, normalisés par $|D_0|$ pour s'affranchir des changements d'échelle [113].

$$F_D = \left\{ \frac{|D_1|}{|D_0|}, \frac{|D_2|}{|D_0|}, \ldots, \frac{\left|D_{\frac{N}{2}}\right|}{|D_0|} \right\} \tag{2.27}$$

2.2.3.3. Filtres de Gabor

Un filtre de Gabor est un filtre sélectif dont la réponse impulsionnelle est définie par une fonction Gaussienne modulée par une fonction sinusoidale, telle que:

$$g(x, y, f_0, \theta) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\frac{1}{2}\left[\left(\frac{x'}{\sigma_x}\right)^2 + \left(\frac{y'}{\sigma_y}\right)^2\right]} e^{j2\pi f_0 x'} \tag{2.28}$$

où:

$$x' = x\cos\theta + y\sin\theta \tag{2.29}$$

$$y' = -x\sin\theta + y\cos\theta \tag{2.30}$$

σ_x (respectivement σ_y) est la variance de la Gaussienne selon l'axe X (respectivement Y) qui permet de déterminer la largeur de bande du filtre de Gabor. Le paramètre f_0 est la fréquence de la sinusoïde et θ est l'orientation.

Les filtres de Gabor ont été largement utilisés dans différentes applications de traitement d'images et notamment en reconnaissance des caractères manuscrits. Dans [115], les auteurs ont proposé une approche basée sur un banc de filtres de Gabor pour l'identification de scripteurs. L'usage de ce type de filtres est motivé par leur capacité d'analyser l'écriture manuscrite et ce, sous différentes échelles et orientations. Au total, 24 images filtrées sont obtenues en fixant le nombre de fréquences à 3 et le nombre d'orientations à 8. Ensuite, des caractéristiques de types moyenne et variance sont extraites à partir de chaque image filtrée, pour former ainsi un vecteur de caractéristiques.

Dans [116], une méthode basée sur les filtres de Gabor a été proposée pour la reconnaissance des caractères manuscrits Gumurkhi. L'application de ces filtres à l'image du caractère permet de générer 35 images filtrées (7 orientations et 5 échelles). A partir de chaque image filtrée, trois autres images d'énergie qui correspondent au module, partie réelle et partie

imaginaire ont été obtenues. Ensuite, la moyenne et la variance sont calculées dans chacune d'elles, pour former un vecteur de caractéristiques.

Dans le même contexte, un autre vecteur de caractéristiques a été également obtenu. Ces caractéristiques ont été extraites des parties réelles et imaginaires de chaque image filtrée. Ensuite, une analyse en composante principale a été appliquée aux vecteurs de caractéristiques pour réduire leurs dimensions. Enfin, l'image filtrée retenue est celle qui fournie un meilleur taux de reconnaissance.

2.2.3.4. Transformée en ondelettes

La transformée en ondelettes permet de décomposer une image en ses différentes composantes fréquentielles. Son avantage réside dans l'usage des fenêtres d'analyses de tailles variables pouvant détecter dans une image des caractéristiques pertinentes difficilement décelables voire invisibles sur l'image originale. Elle peut donc représenter cette image selon différentes résolutions. Associée au formalisme de l'analyse multirésolution (AMR) introduit par S. Mallat [117], la transformée en ondelettes discrète a trouvé différentes applications, telles que l'analyse des signaux [118], la compression d'images [119] et la reconnaissance de visages [120]. En reconnaissance des caractères, cette technique a été également exploitée dans de nombreux travaux de recherches [121,122], où différentes types d'ondelettes ont été introduites pour caractériser la forme des caractères.

2.3. Conclusion

Nous avons présenté dans ce chapitre quelques techniques d'extraction de caractéristiques les plus utilisées dans le domaine de reconnaissance des caractères imprimés et manuscrits. On retient de cet état de l'art assez exhaustif, la richesse et la variété des descripteurs utilisés dans ce domaine. Certains descripteurs sont basés sur le calcul des caractéristiques topologiques, d'autres sur des caractéristiques statistiques comme SIFT, SURF, BOF, HOG et LBP, alors que d'autres descripteurs s'appuient sur des transformations globales comme les moments de Zernike, les filtres de Gabor et la transformée en ondelettes. De par sa propriété de localisation en espace et en échelle, la transformée en ondelettes permet d'extraire à la fois l'information locale et globale dans une image. Nous nous intéressons particulièrement à cette transformée afin d'extraire des caractéristiques.

Dans le prochain chapitre, nous allons donner plus de détails sur la transformée en ondelettes ainsi que son application à la reconnaissance des caractères manuscrits.

Chapitre 3

Reconnaissance des chiffres manuscrits à base de la transformée en ondelettes

3.1. Introduction

De nombreuses techniques d'extraction de caractéristiques ont été proposées pour la reconnaissance de caractères manuscrits. Dans ce chapitre, nous nous sommes intéressés, particulièrement à la transformée en ondelettes discrète (TOD) et son application à la reconnaissance des chiffres manuscrits.

Concrètement, la TOD permet de décomposer une image en sous-bandes images d'approximation et de détails à différentes échelles par l'intermédiaire d'une ondelette. En pratique, la TOD est confrontée au choix de l'ondelette et des sous-bandes pour la caractérisation des images.

Nous débuterons ce chapitre par présenter les bases de données des chiffres manuscrits utilisées dans nos tests, ainsi qu'une brève description de l'algorithme SVM adopté pour effectuer la classification des chiffres manuscrits. Par la suite, nous présenterons dans les trois dernières parties de ce chapitre, nos contributions apportées dans le cadre de la reconnaissance des chiffres manuscrits.

La première est consacrée à l'étude de l'influence de l'ondelette et à la détermination des sous-bandes images qui conviennent le mieux à la caractérisation des chiffres manuscrits. Dans la seconde, nous effectuons une étude comparative entre plusieurs descripteurs de l'état de l'art pour la reconnaissance des chiffres manuscrits. Cette étude sera suivie de la description d'une méthode de caractérisation hybride associant la TOD à la technique basée sur les histogrammes de gradients orientés (HOG) pour la discrimination des chiffres manuscrits.

Nous proposons dans la troisième partie, une technique de réduction et de sélection des caractéristiques qui combine l'Analyse en Composantes Principales (ACP) et la méthode de Sélection Séquentielle Ascendante (SFS) pour la sélection des caractéristiques les plus pertinentes pour la discrimination des chiffres manuscrits.

3.2. Bases de données de chiffres manuscrits

3.2.1. Base USPS

La base USPS contient 9298 images réelles de chiffres isolés issus de la segmentation des codes postaux écrits manuellement et scannés par le service des postes américain. Ces images en niveaux de gris ont une taille de 16x16. 7291 images forment la base d'apprentissage et 2007 images forment la base de test (Tableau 3.1). Il est connu que l'ensemble de test est plutôt difficile, l'erreur humaine est estimée à 2.5% correspondant à un taux de reconnaissance de 97.5% [123]. La figure 3.1 montre quelques échantillons de la base USPS et leurs étiquettes correspondantes. On voit bien la difficulté à reconnaitre certains caractères, cela est dû à la présence de formes mal étiquetées et d'autres formes méconnaissables.

Chiffre	USPS		MNIST		CVLSD		SVHN	
	Apprentissage	Test	Apprentissage	Test	Apprentissage	Test	Apprentissage	Test
0	1194	359	5923	980	700	700	4948	1744
1	1005	264	6742	1135	700	700	13861	5099
2	731	198	5958	1032	700	700	10585	4149
3	658	166	6131	1010	700	700	8497	2882
4	652	200	5842	982	700	700	7458	2523
5	556	160	5421	892	700	700	6882	2384
6	664	170	5918	958	700	700	5727	1977
7	645	147	6265	1028	700	700	5595	2019
8	542	166	5851	974	700	700	5045	1660
9	644	177	5949	1009	700	700	4659	1595
Total	7291	2007	60000	10000	7000	7000	73257	26032

Tableau 3.1. Nombre d'échantillons dans chaque classe

Figure 3.1. Quelques échantillons issus de la base USPS

3.2.2. Base MNIST

La base de données MNIST est une base standard, publique, très utilisée dans le domaine de la reconnaissance des chiffres manuscrits. Elle est composée de 70000 images en niveaux de gris, extraites d'une base plus large (NIST). 60000 images sont utilisées pour former la base d'apprentissage et les 10000 autres images constituent la base de test [1]. Le nombre d'images dans chaque classe est indiqué dans le tableau 3.1. La taille de chaque image est de 28x28. Le taux de reconnaissance obtenu par un être humain est évalué à 99.8% [123]. La figure 3.2 montre quelques échantillons de cette base.

Figure 3.2. Quelques échantillons extraits de la base MNIST

3.2.3. Base CVLSD

La base de données CVLSD (Computer Vision Lab Single Digit) a été utilisée dans la compétition de reconnaissance de chiffres manuscrits ICDAR 2013. Elle fait partie d'une base de chaines de caractères manuscrits plus large (CVL HDdb: CVL Handwritten Digit database), collectée sur 303 scripteurs parmi des étudiants de l'université de technologie de Vienne en Autriche [124]. CVLSD est composée de 14000 images couleur, dont 700 images par classe, formées par 67 scripteurs sont réservées pour l'apprentissage et les 700 autres

images par classe, fournies par 60 autres scripteurs sont utilisées pour le test (Tableau 3.1). La taille de chaque image est de 28x28. La figure 3.3 montre quelques images des caractères provenant de cette base.

Figure 3.3. Quelques échantillons de la base CVLSD

3.2.4. Base SVHN

La base de données SVHN (Street View House Numbers) construite par Google Street View [125] est utilisée pour la détection des numéros des immeubles et maisons. Elle est composée de 99289 images couleur de taille 32x32 chacune, dont 73257 images forment la base d'apprentissage et les 26032 autres images constituent la base de test (Tableau 3.1). Il est à noter que cette base est assez complexe; les images sont à faible résolution, les chiffres sont caractérisés par une certaine déformation et ne sont pas tout à fait isolés. Le taux de reconnaissance obtenu par un être humain est estimé à 98%. La figure 3.4 montre quelques échantillons issus de cette base.

Figure 3.4. Echantillons de la base SVHN

3.3. Classifieur SVM

Les machines à vecteurs de support SVM (Support Vector Machines) sont une famille de classifieurs supervisés qui ont été introduits par Vapnik [126]. Ces méthodes ont montré leur efficacité dans de nombreuses applications, notamment en reconnaissance de caractères manuscrits [127,128]. Dans sa version standard, SVM est un modèle linéaire conçu pour la classification binaire. Pour étendre ce modèle au cas non linéaire, l'idée consiste à projeter les données dans un espace de grande dimension où la séparation entre les classes devient possible, partant du principe que les données sont linéairement séparables dans cet espace de grande dimension. Cependant, il s'est avéré que les calculs dans cet espace augmenté deviennent très couteux en complexité et en ressources. Des fonctions particulières dites noyaux non linéaires ont été ainsi introduites afin de remédier à ce problème (pour plus de détails, se référer à l'annexe B).

Si nous disposons d'un ensemble de N données étiquetées $\{(x_i, y_i), i = 1, \ldots, N\}$, où $x_i \in \mathcal{R}^D$, $y_i \in \{-1, +1\}$, et si nous notons par $K(x_i, x_j)$ la fonction noyau qui permet de les projeter dans un espace de grande dimension, alors, le SVM recherche un hyperplan qui permet la séparation optimale de ces deux classes de données.
Mathématiquement, le principe du SVM consiste à résoudre un problème dual d'optimisation suivant [129]:

$$\max_{\alpha_i} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \tag{3.1}$$

avec les contraintes:

$0 \leq \alpha_i \leq C, \quad i = 1, \ldots, N$

et:

$$\sum_{i=1}^{N} \alpha_i y_i = 0$$

où C est un paramètre de pénalisation qui règle le degré de compromis désiré entre la séparabilité des classes et l'étanchéité du modèle aux erreurs d'apprentissage. α_i sont des multiplicateurs de Lagrange et $K(.\ ,\ .)$ un noyau (kernel) qui peut prendre plusieurs formes, tel que le noyau Gaussien défini par:

$$K(x_i, x_j) = exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \quad (3.2)$$

σ est un paramètre qui règle la largeur de la fonction Gaussienne.

La résolution du problème dual (3.1) permet de calculer les valeurs des multiplicateurs de Lagrange α_i.

La fonction de décision du SVM non linéaire permettant de reconnaitre un nouveau caractère $x_z = [x_{z1},\ x_{z2},\ \ldots,\ x_{zD}]$ non déjà appris s'écrit:

$$f(x_z) = signe\left(\sum_{i=1}^{N} \alpha_i y_i K(x_i, x_z) + b\right) \quad (3.3)$$

où le terme biais b peut être obtenu par:

$$b = \frac{1}{N_s}\left(y_i - \sum_{j=1}^{N} \alpha_j y_j K(x_i, x_j)\right) \quad (3.4)$$

N_s est le nombre de vecteurs de support, correspondant aux nombre de coefficients α_j non nuls.

Les méthodes classiques d'utilisation des SVM pour le cas multi-classes consistent à décomposer, dans un premier temps, le problème en une série de classifieurs binaires (un-contre-un, un-contre-tous) [130]. Ensuite, les décisions des classifieurs élémentaires sont combinées par une stratégie de fusion (vote, utilisation de la théorie probabiliste) pour permettre la discrimination multi-classes. L'approche «un-contre-un» propose d'utiliser $\frac{nc(nc-1)}{2}$ discriminateurs binaires pour décrire toutes les dichotomies possibles parmi les nc classes. Quant à l'approche «un-contre-tous», elle utilise nc classifeurs binaires dont chacun est spécialisé pour la reconnaissance d'une classe opposée à la fusion des $(nc-1)$ autres classes. Dans la suite de ce chapitre, nous considérons un classifieur SVM de type «un contre un» permettant de ne retenir que les caractères les plus discriminants pour la classification. Ce choix est basé sur l'étude faite dans [131,132] montrant que cette approche est plus précise que la technique un contre tous.

3.4. Transformée en ondelettes 2D

La transformée en ondelettes est très utilisée comme outil d'analyse et de représentation d'un signal ou d'une image et ce, grâce à l'introduction du concept multi-résolution [117]. Cette

notion de multi-résolution, fortement liée à la représentation de l'information à des échelles différentes, permet de distinguer la transformée en ondelettes des autres transformées comme celle de Fourier et celle de Gabor.

La transformée en ondelettes continue consiste à décomposer un signal sur une base de fonctions $\psi_{a,b}$ obtenues par translations et dilatations d'une ondelette mère ψ [133]:

$$\psi_{a,b}(x) = \frac{1}{\sqrt{a}}\psi\left(\frac{x-b}{a}\right) \quad a \in \mathcal{R}^{+*}, b \in \mathcal{R} \tag{3.5}$$

a et b représentent respectivement les paramètres d'échelle et de translation.

La transformée en ondelettes continue (TOC) d'une fonction $I(x) \in L^2(\mathcal{R})$ s'écrit:

$$TOC\{I(x)\} = C_{a,b} = \int_{-\infty}^{+\infty} I(x)\psi_{a,b}(x)dx \tag{3.6}$$

Pour une implémentation efficace, Meyer [133] a construit des bases d'ondelettes orthonormales en discrétisant les paramètres d'échelle et de translation (voir annexe A). Ces paramètres sont choisis tels que: $a_j = 2^j$ et $b_k = 2^j k$. La transformée en ondelettes discrète (TOD) est obtenue par:

$$TOD\{I(x)\} = C_{j,k} = \int_{-\infty}^{+\infty} I(x)\psi_{j,k}(\mathrm{x})dx, \quad (j,k) \in Z \tag{3.7}$$

avec:

$$\psi_{j,k}(x) = 2^{-j/2}.\psi\left(2^{-j}x - k\right) \tag{3.8}$$

L'analyse multi-résolution par la transformée en ondelettes discrète, permet d'analyser un signal à différents niveaux de résolution, en le décomposant en une suite de signaux d'approximations (basses fréquences) et de détails (hautes fréquences). Cette décomposition peut être implémentée grâce à des bancs de filtres qui sont liés à l'ondelette choisie [117].

Appliquée à un signal bidimensionnel (image), la transformée en ondelettes discrète permet de décomposer l'image $I(x,y)$ en 4 sous-bandes fréquentielles. Ce processus de décomposition consiste à appliquer un filtre passe-bas $l(x)$ et un autre passe-haut $h(x)$ sur les lignes de $I(x,y)$ pour générer respectivement une sous-bande de basses fréquences et une autre de hautes fréquences. Ensuite, sur les colonnes de ces sous-bandes, sont appliqués les mêmes filtres, pour produire une sous-bande d'approximation $I_{LL}(x,y)$ et 3 sous-bandes de détails $I_{LH}(x,y)$, $I_{HL}(x,y)$, $I_{HH}(x,y)$ au premier niveau de résolution. Ces dernières

correspondent respectivement aux détails horizontaux, verticaux et diagonaux. Chaque filtrage est suivi d'une opération de sous-échantillonnage (ou de décimation) par 2. Le même processus de décomposition peut se répéter sur la sous bande d'approximation $I_{LL}(x,y)$ jusqu'à atteindre un niveau de résolution donné. La transformée par paquet d'ondelettes répète le même principe de décomposition sur les sous-bandes de détails et d'approximation. La figure 3.5 illustre le processus de décomposition en un niveau de résolution.

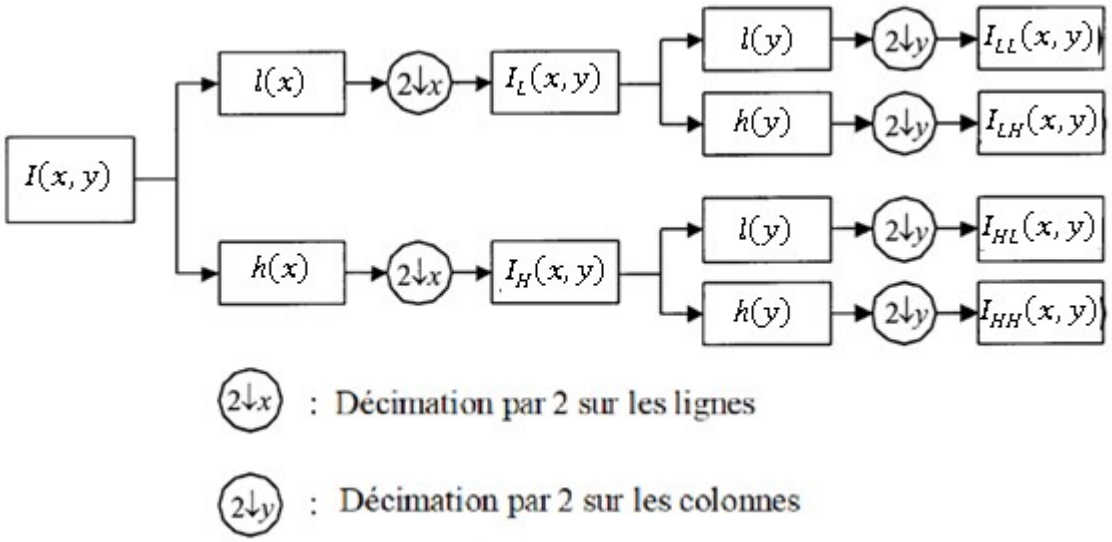


Figure 3.5. Décomposition de l'image en un niveau de résolution

La figure 3.6 montre un exemple d'application de la TOD à l'image du caractère 5, en utilisant l'ondelette Symlet d'ordre 8, avec un seul niveau de décomposition.

3.5. Etat de l'art sur l'application de la transformée en ondelettes à la reconnaissance des caractères manuscrits

Plusieurs approches de caractérisation basées sur la transformée en ondelettes ont été proposées pour la reconnaissance de caractères. Pour former le vecteur de caractéristiques, certains travaux se sont basés sur le calcul d'énergie de chaque sous-bande image, d'autres s'appuient sur le calcul des caractéristiques statistiques à savoir la moyenne et la variance.

Dans [134], une approche basée sur la combinaison de la transformée en ondelettes et la technique de décomposition en mode empirique a été développée pour la détection de caractères sur des plaques signalétiques des véhicules. Ce système utilise une ondelette de Haar pour extraire les détails verticaux. Un signal monodimensionnel est obtenu par

projection de ces détails sur l'axe des Y. Ensuite, l'analyse en mode empirique est appliquée à ce signal pour extraire des caractéristiques.

Dans [135], une méthode basée sur la transformée en ondelettes a été mise en place afin de reconnaitre des caractères numériques non contraints. L'ondelette biorthogonale spline de Cohen-Daubechies-Feauveau (CDF 3/7) est utilisée pour décomposer l'image initiale en un niveau de résolution. Les coefficients issus des quatre sous-bandes images obtenues forment un vecteur de caractéristiques.

Bathacharya [136] a présenté une approche de reconnaissance de caractères numériques imprimés basée sur la transformée en ondelettes moyennant l'ondelette de Daubechies d'ordre 4. L'image initiale est décomposée en trois niveaux de résolution. Les coefficients des trois sous-bandes d'approximations ainsi obtenues sont utilisés comme caractéristiques.

Dans [137], l'ondelette de Daubechies d'ordre 4 a été utilisée pour décomposer l'image de chaque caractère en 3 niveaux de résolution. Une technique de zonage a été appliquée sur toutes les sous-bandes images issues de cette décomposition. Dans chaque zone, des caractéristiques de densités de pixels et de codes de Freeman dans les 8 directions sont extraites des sous-bandes d'approximation et de détails respectivement. Ensuite, ces caractéristiques sont concaténées pour former un descripteur.

Dans l'approche proposée par Sasi [138], la transformée en paquet d'ondelettes a été proposée pour la reconnaissance de caractères. L'image est décomposée en 3 niveaux de résolution au moyen de l'ondelette Symlet d'ordre 8. Des caractéristiques basées sur le calcul des écarts types sont extraites des sous-bandes images résultantes.

Dans [139], la transformée en paquet d'ondelettes est appliquée sur l'image du caractère en utilisant différentes ondelettes comme les Daubechies, les Symlets et les Coiflets. Les caractéristiques extraites sont essentiellement basées sur le calcul d'énergie, d'écart-type, de la moyenne et d'entropie.

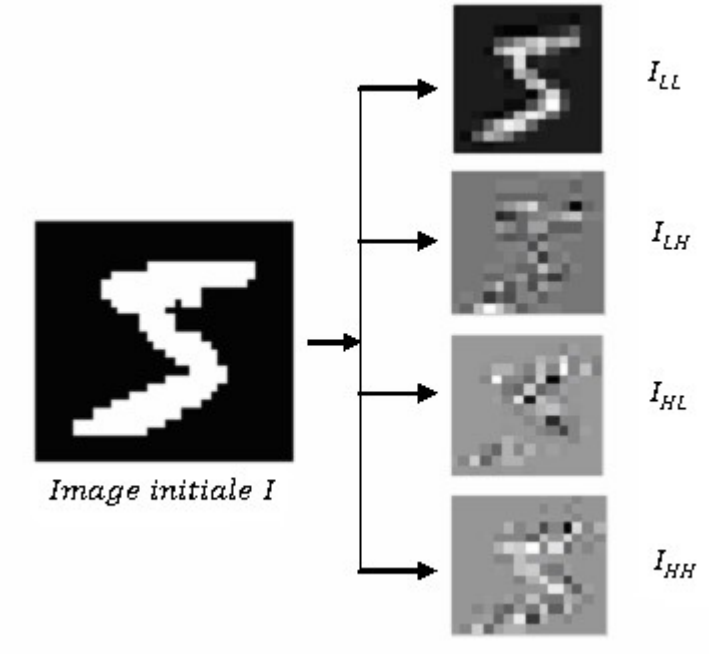


Figure 3.6. Résultats obtenus de la décomposition de l'image du caractère 5

3.6. Choix de l'ondelette et de la sous-bande image

Dans la littérature, le choix du type de l'ondelette ainsi que le type de la sous-bande image utilisée pour extraire des caractéristiques est souvent non justifié. Dans cette partie de la thèse, nous étudierons l'influence de l'ondelette ainsi que la pertinence des sous-bandes dans la reconnaissance des caractères manuscrits. Cette étude sera complétée par une comparaison avec quelques variantes de descripteurs basés sur la transformée en ondelettes et quelques méthodes tirées de la littérature.

La base de données MNIST est entièrement utilisée dans cette étude et le classifieur SVM a été choisi pour évaluer les caractéristiques utilisées. Rappelons que cette base est divisée en deux sous-ensembles disjoints. Le premier est utilisé pour l'apprentissage du classifieur SVM, le second est utilisé pour estimer le taux de reconnaissance sur des échantillons non appris. Ce taux est évalué comme le rapport entre le nombre de caractères correctement reconnus et le nombre total de caractères utilisés.

3.6.1. Choix de l'ondelette

Dans cette section, nous avons fait appel aux ondelettes les plus récurrentes dans le domaine de reconnaissance des caractères manuscrits afin de vérifier leurs pertinences à discriminer les caractères. Ainsi, nous avons évalué les ondelettes suivantes (Figure 3.7):

- la famille des Daubechies (haar, db2, db4, db5),
- la famille des Coiflets (coif1, coif3, coif5)
- la famille des Symlets (sym2, sym3, sym4, sym5, sym6, sym7, sym8, sym9, sym12),
- la famille de Cohen Daubechies Feauveau (CDF 3/7 et CDF 9/7).

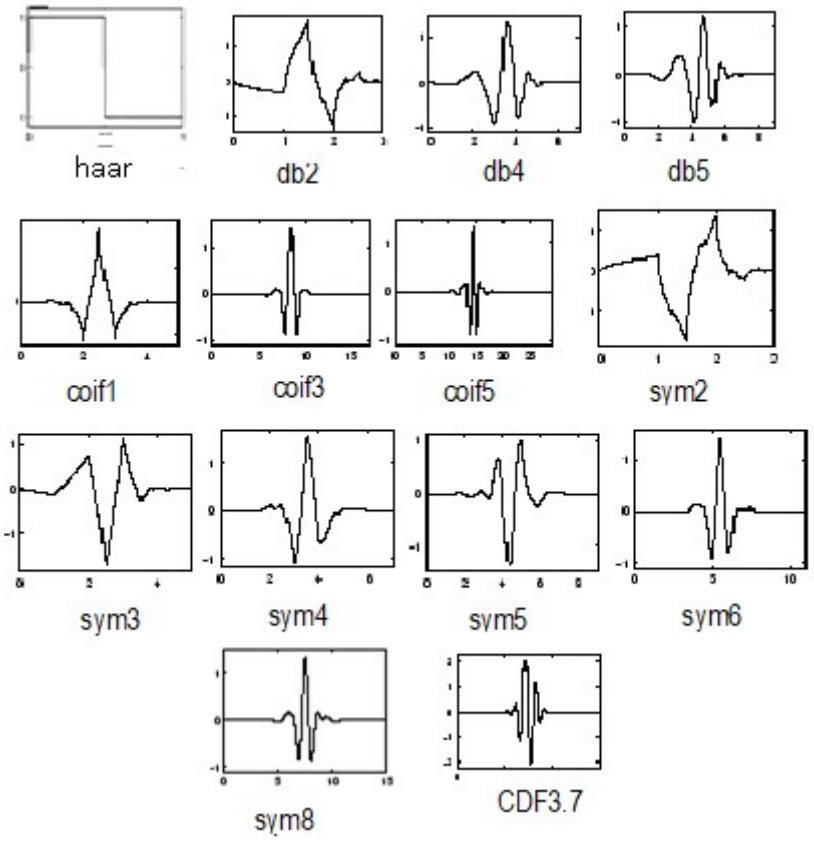


Figure 3.7. La forme de quelques ondelettes

En utilisant la transformée en ondelettes, l'image initiale est décomposée au premier niveau de résolution. Cette décomposition génère 4 sous-bandes images (une approximation $I_{LL}(x,y)$ et 3 images de détails $I_{LH}(x,y)$, $I_{HL}(x,y)$ et $I_{HH}(x,y)$)

Chaque sous-bande image est caractérisée par une matrice de coefficients (14x14). Seuls les coefficients de la sous-bande d'approximation sont retenus comme caractéristiques. Par conséquent, chaque image du chiffre manuscrit est caractérisée par un vecteur de caractéristiques, composé de 196 coefficients. Le tableau 3.2 affiche le taux de reconnaissance obtenu par chaque ondelette. Notons que dans toutes ces expériences, les paramètres du classifieur SVM sont réglés de manière expérimentale, tels que le paramètre de pénalité C ainsi que celui du noyau Gaussien représenté par gamma $\gamma = \frac{1}{2\sigma^2}$, sont fixés respectivement à 6 et à 0.25.

Ondelettes	Taux de Reconnaissance	Ondelettes	Taux de Reconnaissance
haar	98.56%	sym4	98.70%
db2	98.66%	sym5	98.73%
db4	98.68%	sym6	98.71%
db5	98.65%	sym7	98.66%
coif1	98.68%	sym8	**98.76%**
coif3	98.68%	sym9	98.75%
coif5	97.90%	sym12	98.67%
sym2	98.66%	CDF 3/7	98.52%
sym3	98.67%	CDF 9/7	98.64%

Tableau 3.2. Résultats de reconnaissance sur la base MNIST

Les résultats du tableau 3.2 montrent que les caractéristiques dérivées de la transformée en ondelettes discriminent correctement les caractères manuscrits, puisque le taux de reconnaissance reste supérieur à 98% dans la plus part des cas. Cependant, nous constatons que le taux de reconnaissance varie en fonction de l'ondelette. Parmi l'ensemble des ondelettes testées, Symlet d'ordre 8 affiche le meilleur taux de reconnaissance, estimé à 98,76%, dans le cas de la base de données MNIST.

3.6.2. Choix de la sous-bande image

Dans l'expérience précédente, les caractéristiques utilisées pour classer les caractères manuscrits correspondent aux coefficients de la sous-bande d'approximation (I_{LL}) obtenue au premier niveau de décomposition. Dans cette section, notre objectif est d'examiner la pertinence des caractéristiques extraites à partir de l'ensemble des quatre sous-bandes images I_{LL}, I_{LH}, I_{HL}et I_{HH}, générées par la transformée en ondelettes. Ainsi, nous considérons comme caractéristiques les coefficients de chaque sous-bande image seule ou en les combinant par les opérateurs maximum (max), le minimum (min) ou la moyenne (moy) entre deux sous-bandes images seulement, entre trois sous-bandes images uniquement ou entre toutes les-sous bandes images. Compte tenu des résultats de l'expérience précédente, l'ondelette Symlet d'ordre 8 (sym8) est utilisée dans cette expérience en considérant même dans ce cas, un niveau de décomposition.

Les taux de reconnaissance, pour les différentes combinaisons réalisées, sont résumés dans le tableau 3.3.

I_{LL}, I_{LH}, I_{HL},I_{HH}	Max	Min	Moy
1000	**98.76%**	**98.76%**	**98.76%**
0100	83.82%	83.82%	83.82%
0010	81.19%	81.19%	81.19%
0001	70.77%	70.77%	70.77%
1100	95.84%	96.95%	97.65%
1010	95.57%	96.93%	97.40%
1001	93.58%	96.60%	97.10%
0110	83.04%	84.15%	85.75%
0101	78.66%	79.55%	77.35%
0011	77.01%	77.85%	75.03%
1110	94.55%	94.98%	93.55%
1101	93.39%	94.96%	97.82%
1011	93.73%	94.88%	91.56%
0111	82.39%	83.29%	73.56%
1111	93.71%	92.58%	94.04%

Tableau 3.3. Taux de reconnaissance obtenus par les sous-bandes combinées sur la base MNIST

Les résultats montrent clairement que le meilleur taux de reconnaissance est obtenu lorsque seule la sous-bande image d'approximation est utilisée comme caractéristiques. L'utilisation des coefficients de détails comme caractéristiques entraine la diminution du taux de reconnaissance. Dans la dernière ligne de le tableau 3.3, toutes les sous-bandes images sont combinées pour générer les caractéristiques et ce, en appliquant les opérateurs maximum, minimum ou moyen.

Une autre façon de fusionner l'information contenue dans les quatre sous-bandes images est d'attribuer des poids à chacune d'elles pour générer de nouvelles caractéristiques, telle que:

$$I_1 = w_1.\mathrm{I}_{LL} + w_2.\mathrm{I}_{LH} + w_3.\mathrm{I}_{HL} + w_4.\mathrm{I}_{HH} \quad (3.9)$$

Les poids w_1, w_2, w_3 et w_4 prennent des valeurs entre 0 et 1 et doivent satisfaire la condition: $w_1 + w_2 + w_3 + w_4 = 1$. Les valeurs optimales de $w_i (i = 1, 2, 3, 4\ )$ conduisant au meilleur

taux de reconnaissance sont données dans le tableau 3.4. Elles sont déterminées expérimentalement par une recherche exhaustive.

Sous bandes images	I_{LL}	I_{LH}	I_{HL}	I_{HH}
Poids	$w_1 = 0.7$	$w_2 = 0.1$	$w_3 = 0.1$	$w_4 = 0.1$
Taux de reconnaissance	98.66%			

Tableau 3.4. Taux de reconnaissance basé sur la pondération des sous-bandes images

Le fait que la valeur de w_1 est plus grande que celles des autres poids indique que nous devons accorder plus d'importance à la sous-bande d'approximation qu'aux sous-bandes de détails. Un taux de reconnaissance de 98.66% est obtenu, mais cette valeur reste inférieure au taux obtenu (98.76%) par la sous-bande d'approximation prise seule. Finalement, les caractéristiques extraites uniquement de la sous-bande d'approximation en utilisant l'ondelette Symlet d'ordre 8 (sym8) offrent une meilleure discrimination entre les caractères manuscrits de la base MNIST.

3.6.3. Comparaison avec des méthodes de zonage

Pour montrer la pertinence des caractéristiques extraites de la sous-bande d'approximation en utilisant l'ondelette Symlet d'ordre 8, nous les avons comparées avec celles obtenues par deux autres méthodes dont le principe repose sur l'approche multirésolution.

Le principe de la première méthode, appelée «image normalisée réduite», consiste à appliquer la technique de zonage sur l'image du caractère. Cela revient à diviser l'image initiale en (14x14) zones distinctes, chacune de taille (2x2 pixels). Ensuite, la moyenne des niveaux de gris dans chaque zone est utilisée comme composante du vecteur de caractéristiques. Enfin, chaque image du caractère est représentée par un vecteur de caractéristiques de 196 valeurs.

Le taux de reconnaissance obtenu par le classificateur SVM en utilisant ce vecteur de caractéristiques est égal à 98,55% (Tableau 3.5). Notons que ce résultat reste inférieur à celui obtenu par les caractéristiques extraites de la sous-bande d'approximation fournie par la décomposition de l'ondelette sym8 qui est de 98,76%.

La deuxième technique de multirésolution consiste à appliquer la technique de zonage sur l'image gradient obtenue par l'application de l'opérateur Sobel sur l'image du caractère. Par

conséquent, l'image résultante, appelée ici ''image gradient réduite'', est considérée comme une sous image de détail. Le vecteur de caractéristiques est extrait en utilisant le même principe de zonage que la méthode précédente. Le taux de reconnaissance obtenu par ces nouvelles caractéristiques est égal à 97,55% (Tableau 3.5). Cette valeur est plus élevée que les taux de reconnaissance fournis par les caractéristiques extraites des images de détails seuls ou de leurs combinaisons (Tableau 3.3), mais reste inférieure à la valeur 98,76% obtenue en utilisant uniquement les caractéristiques extraites de la sous-bande d'approximation.

Caractéristiques	Apprentissage	Test	Taux de Reconnaissance
Image réduite normalisée	60000	10000	98.55%
Image gradient réduite	60000	10000	97.55%

Tableau 3.5. Taux de reconnaissance obtenus par les méthodes de zonage sur la base MNIST

3.6.4. Comparaison avec d'autres travaux tirés de la littérature

Nous avons effectué une étude comparative avec d'autres travaux réalisés sur la même base de données que la notre (MNIST) et ayant recours à une analyse multirésolution. La comparaison est effectuée avec les méthodes présentées dans la littérature [136,140,141,142]. Les taux de reconnaissance sont présentés dans le tableau 3.6.

Dans [136], le taux de reconnaissance de 97,57% est obtenu en utilisant un ensemble de classifieurs MLP entrainés sur une base d'apprentissage composée de 50000 images de caractères. Les coefficients de la sous-bande d'approximation obtenus par application de l'ondelette de Daubechies d'ordre 4 (db4) sont utilisés comme caractéristiques. Cependant, cette valeur reste inférieure à notre résultat (98,60%).

Dans l'approche développée par Rehman [140], une mesure de similarité structurel SSIM (Structural Similarity Index) a été utilisée pour la reconnaissance des caractères manuscrits. Cette mesure est appliquée sur des images des caractères, représentés dans le domaine de la transformée en ondelettes complexe. Le taux de reconnaissance estimé de 98,09% reste inférieur au notre (98,76%).

Dans [141], une méthode basée la combinaison de caractéristiques multiéchelles a été proposée pour la discrimination de caractères numériques. Des caractéristiques de types gradient et orientation ont été extraites en utilisant l'ondelette continue Chapeau Mexicain. Le taux de reconnaissance de 98,22% obtenu reste également inférieur à 98.76%.

Dans [142], les auteurs ont fait appel à la transformée en ondelette (CDF 9/7) ainsi qu'à l'analyse en composante principale (ACP). En décomposant l'image initiale au premier niveau, les coefficients d'approximation sont utilisés comme caractéristiques. Un taux de reconnaissance de 98,64% est obtenu. Ce résultat a été amélioré pour atteindre 99,32% en utilisant des caractéristiques multirésolution et un ensemble de classifieurs SVMs.
Nous considérons que le taux de reconnaissance obtenu a été visiblement amélioré, vu l'emploi de deux techniques connues pour leurs coût en matière de temps; à savoir l'ACP (analyse en composantes principales) et la méthode de fusion de classifieurs (Bayesienne).
Notre méthode nous a permis d'obtenir un taux de reconnaissance plausible (98.76%) malgré l'emploi d'un seul type de descripteur et d'un seul classifieur avec un temps de calcul plus réduit.

Méthode	Ondelette	Classifieurs	Base d'Apprentissage	Base de Test	Taux de Reconnaissance
Ait Aider et al. [143]	Symlet ordre 8	SVM	60000	10000	98.76%
Ait Aider et al. [143]	Symlet ordre 8	SVM	50000	10000	98.60%
Bhattacharya et Chaudhuri [136]	Daubechies ordre 4	MLP	50000	10000	97.57%
Rehman et al.[140]	Ondelette complexe	SVM	60000	10000	98.09%
Romero et al.[141]	Chapeau mexicain	MLP	60000	10000	98.22%
Seijas et Segura [142]	CDF(9/7)	SVM	60000	10000	98.64%
Seijas et Segura [142]	CDF(9/7)	SVMs	60000	10000	99.32%

Tableau 3.6. Résultats des taux de reconnaissance avec la base de données MNIST

3.7. Comparaison entre différents types de caractéristiques

Dans cette section, nous effectuons une étude comparative entre les descripteurs dérivés de la transformée en ondelettes et ceux déduits des filtres de Gabor, des moments de Zernike, des

opérateurs de Kirsch, de l'histogramme de gradients orientés (HOG), des motifs binaires locaux (LBP) et des sacs de caractéristiques visuels (BOF) pour la reconnaissance des chiffres manuscrits. Cette comparaison est menée sur les 4 bases de données USPS, MNIST, CVLSD et SVHN.

Chaque base est divisée en une base d'apprentissage afin d'entrainer un classifieur et une base de test pour évaluer la pertinence des caractéristiques à discriminer des caractères non appris. L'évaluation est alors effectuée en comptabilisant le taux de reconnaissance obtenu par le classifieur sur l'ensemble de test. Dans cette étude, nous avons utilisé le classifieur SVM avec les paramètres de pénalité C et du noyau Gaussien (Gamma) fixés respectivement à 6 et 0.25.

Notons que les descripteurs de Gabor sont obtenus en divisant l'image initiale en 4 blocs de tailles identiques. Sur chaque bloc, nous appliquons plusieurs filtres de Gabor obtenus en faisant varier le nombre d'échelles de 1 à 6. Le nombre d'orientations est fixé à 18. L'énergie moyenne calculée dans chaque bloc, est utilisée pour former un vecteur de caractéristiques dont la dimension est fixée à 432 (4x6x18) (Tableau 3.7).

Le descripteur de Zernike que nous avons utilisé est construit en prenant en compte à la fois les informations d'amplitude et de phase. Ces informations sont extraites à partir des moments de Zernike complexes obtenus jusqu'à l'ordre 10. Le vecteur de caractéristiques obtenu est de dimension 132 et ce dans le cas des 4 bases (Tableau 3.7).

Les opérateurs de gradient de Kirsh sont utilisés pour extraire des cartes de caractéristiques suivant les 4 directions (verticale, horizontale et les deux diagonales). L'information globale représentée par l'image initiale est également prise en compte. Pour réduire la taille de chacune de ces cartes, nous avons utilisé la technique de zonage. Puis, la valeur moyenne des intensités des pixels est calculée dans chaque zone (2x2). La concaténation de toutes ces valeurs forme un vecteur de caractéristiques. Sa dimension est: ((taille de l'image/2) x 5) (Tableau 3.7).

Pour déterminer le descripteur de HOG, nous avons calculé dans un premier temps l'amplitude et l'orientation de gradient de chaque pixel. Puis nous avons procédé au découpage de l'image du caractère en zones (ou cellules) distinctes. Dans chaque zone, les orientations du gradient sont quantifiées en 9 directions pour former ainsi un histogramme de gradients orientés. Ensuite, nous avons regroupé ces zones en blocs de taille 2x2 chacun, se

chevauchant avec un pas égal à 1. Enfin, après avoir procédé à la normalisation de ces histogrammes par bloc, nous les avons concaténés pour former le descripteur HOG dont la dimension est égale au produit entre le nombre de blocs, le nombre de zones par bloc et le nombre de directions de l'histogramme (Tableau 3.7). Notons que le nombre de blocs dépend de la taille de l'image ainsi que celle de la zone obtenue après découpage.
Dans nos tests, le seul paramètre que nous avons fait varier est la taille de la zone. Pour les 4 bases de données utilisées, la taille retenue est celle qui donne un meilleur taux de reconnaissance. Celle-ci est fixée comme suit:

- 4x4 pour USPS et SVHN.
- 8x8 pour MNIST et CVLSD.

Pour calculer le descripteur LBP, on applique d'abord la technique de zonage sur l'image du caractère afin de la diviser en un ensemble de zones distinctes. Pour chaque pixel dans une zone, un motif binaire est déterminé en fonction de ses 8 voisins localisés sur un rayon R=1. Dans nos tests, nous avons considéré les motifs binaires uniformes et invariants à la rotation. A partir de ces motifs, on obtient 59 codes LBP. Ces codes sont ensuite utilisés pour former un histogramme. La concaténation de l'ensemble de ces histogrammes constitue le descripteur LBP. Sa dimension est égale au produit entre le nombre de zones et le nombre de codes LBP (Tableau 3.7).
Après avoir effectué plusieurs tests sur les 4 bases de chiffres manuscrits, nous avons retenu la taille de la zone qui offre un taux de reconnaissance élevé. Elle est fixée comme suit:

- 3x3 pour USPS.
- 7x7 pour MNIST.
- 5x5 pour CVLSD et SVHN.

La dimension du descripteur BOF est fixée à 500 dans le cas des 4 bases de chiffres manuscrits (Tableau 3.7). Cette valeur correspond au nombre de classes utilisés par l'algorithme des k-moyennes.
Les taux de reconnaissance obtenus par application de chacun de ces vecteurs de caractéristiques au classifieur SVM sont regroupés dans le tableau 3.8.

Méthode	Nombre de caractéristiques Base USPS	Nombre de caractéristiques Base MNIST	Nombre de caractéristiques Base CVLSD	Nombre de caractéristiques Base SVHN
TOD (haar)	64	196	196	256
TOD (db4)	64	196	196	256
TOD (sym4)	64	196	196	256
TOD (sym8)	64	196	196	256
Gabor	432	432	432	432
Zernike	132	132	132	132
Kirsh	320	980	980	1280
HOG	324	144	144	1764
LBP	1475	944	1475	2124
BOF	500	500	500	500

Tableau 3.7. Dimension des vecteurs de caractéristiques utilisées par chaque méthode

Caractéristiques	USPS	MNIST	CVLSD	SVHN
TOD (haar)	94.87%	98.56%	90.79%	74.73%
TOD (db4)	94.37%	98.68%	90.86%	76.80%
TOD (sym4)	95.17%	98.70%	90.83%	75.63%
TOD (sym8)	94.92%	**98.76%**	90.83%	75.94%
Gabor	94.22%	98.05%	92.19%	73.56%
Zernike	86.45%	96.26%	83.11%	58.03%
Kirsch	95.07%	98.40%	92.11%	81.70%
HOG	**96.41%**	**98.77%**	93.87%	**89.27%**
LBP	95.76%	98.00%	**94.54%**	81.75%
BOF	85.58%	96.52%	86.56%	43.17%

Tableau 3.8. Taux de reconnaissance obtenus sur les bases USPS, MNIST, CVLSD et SVHN

Le tableau 3.8 montre que les résultats de reconnaissance obtenus dépendent fortement de la base utilisée. La plus part des descripteurs fournissent des taux de reconnaissance comparables à l'exception des descripteurs de Zernike et BOF dont les taux de reconnaissance restent médiocres. Ceci peut s'expliquer par le fait que les descripteurs de Zernike ne sont pas suffisamment pertinents pour discriminer correctement les caractères entre eux.

Concernant BOF, nous avons constaté que le nombre de points d'intérêt détectés sur certaines images n'est pas suffisamment significatif pour pouvoir décrire les images par une variété de descripteurs SURF.
Sur la base USPS, les descripteurs TOD (sym4), Kirsh, HOG et LBP affichent un taux supérieur à 95% mais HOG avec un taux de 96.41% reste plus performante.
Sur la base MNIST, c'est TOD (sym8) et HOG qui donnent les meilleurs taux de reconnaissance.
Concernant la base CVLSD, les descripteurs HOG et LBP affichent les meilleurs taux de reconnaissance, tandis que sur la base SVHN, HOG reste la plus performante.
Globalement, le descripteur HOG a montré sa supériorité par rapport aux autres descripteurs, pratiquement sur l'ensemble des 4 bases. Les descripteurs dérivés de la TOD discriminent mieux les caractères de la base MNIST que ceux des bases USPS, CVLSD et SVHN. Lorsqu'on compare les ondelettes entre elles, on constate que Symlet d'ordre 8 convient mieux à la discrimination des caractères des bases MNIST et SVHN, alors que Symlet d'ordre 4 semble la plus indiquée pour USPS et Daubechies d'ordre 4 pour CVLSD. Ceci confirme que le type de l'ondelette peut influencer les performances de la reconnaissance des chiffres manuscrits.

3.8. Combinaison de la TOD et des descripteurs HOG

Compte tenu des résultats obtenus précédemment, dans cette section, nous proposons une approche qui combine la transformée en ondelettes et le descripteur HOG pour la reconnaissance des chiffres manuscrits. Cette méthode consiste à appliquer le descripteur HOG à la sous-bande image d'approximation issue de la décomposition de l'image du caractère à la première résolution. Les ondelettes de Haar, Daubechies d'ordre 4, Symlet d'ordre 4 et Symlet d'ordre 8 sont utilisées pour transformer l'image initiale en une autre représentation. Le classifieur SVM est utilisé dans cette expérience pour estimer le taux de reconnaissance. Ses paramètres C et Gamma sont fixés respectivement à 6 et 0.25. Les résultats obtenus sont récapitulés dans le tableau 3.9.

Nous constatons que la combinaison de la TOD avec les descripteurs de HOG a enregistré une nette amélioration du taux de reconnaissance comparativement à l'utilisation de ces deux techniques séparément (Tableau 3.8). L'ondelette Symlet d'ordre 4 est cette fois ci, la plus

indiquée, car elle fournit les meilleurs résultats sur l'ensemble des 4 bases à l'exception d'USPS.

Caractéristiques	USPS	MNIST	CVLSD	SVHN
haar+HOG	96.06%	98.86%	94.36%	89.19%
db4+HOG	**96.31%**	98.69%	92.49%	89.16%
sym4+HOG	95.47%	**99.04%**	**94.61%**	**89.32%**
sym8+HOG	95.02%	99.02%	94.34%	89.19%

Tableau 3.9. Résultats de reconnaissance obtenus par combinaison de la TOD et HOG

3.9. Réduction et sélection de caractéristiques

Nous avons présenté précédemment plusieurs techniques de caractérisation permettant de discriminer les chiffres manuscrits. Dans la plupart de ces méthodes, le nombre de caractéristiques proposé est élevé (Tableau 3.7). Or, il est fréquent que ces caractéristiques ne soient pas toutes aussi informatives; elles peuvent contenir des informations redondantes ou inutiles pour la phase de classification. De plus, certaines caractéristiques peuvent introduire des difficultés au niveau du classifieur, entrainant ainsi la détérioration de ses performances.

Pour réduire le nombre de caractéristiques et ne garder que les plus pertinentes, nous avons proposé dans [144], une approche qui combine une technique d'extraction des caractéristiques, en l'occurrence l'analyse en composantes principales (ACP) et une technique de sélection de caractéristiques à savoir la méthode de sélection séquentielle ascendante (SFS: Sequential Forward Selection). Habituellement, ces deux techniques ne sont pas conjointement utilisées.

L'ACP [145] est une méthode fondamentale d'analyse de données multidimensionnelles que l'on rencontre dans diverses applications de traitement d'images et notamment en reconnaissance de caractères [146,147]. Elle permet de construire un sous-espace de d nouvelles caractéristiques par combinaison linéaire des D caractéristiques initiales. Les nouvelles caractéristiques ne sont pas forcement pertinentes, car l'ACP ne fait pas appel à un classifieur pour évaluer le pouvoir discriminatoire de ces caractéristiques.

Les méthodes de sélection consistent à choisir à partir d'un ensemble initial de caractéristiques celles qui sont les plus pertinentes. D'une manière générale, ces méthodes sont classées en trois approches: ''filter'', ''wrapper'' et ''embedded'' [148].
Dans les méthodes de type filter, un critère d'évaluation comme celui de corrélation ou de Fisher est utilisé pour évaluer le degré de pertinence de chacune des caractéristiques. L'évaluation se fait généralement indépendamment d'un classificateur.

L'approche wrapper se base sur un algorithme de classification pour rechercher parmi tous les sous ensembles de caractéristiques sélectionnées, celui qui fournit un taux d'erreur de classification le plus petit. Par rapport aux méthodes filter, wrapper offre de meilleurs résultats. Cependant, son principal inconvénient est le temps nécessaire pour la sélection de caractéristiques, il est nettement plus long que celui de l'approche filter.

La troisième approche dite ''embedded'', combine l'approche ''filter'' pour la présélection d'un sous ensemble de caractéristiques plus réduit et une méthode de sélection de type ''wrapper''.

La sélection séquentielle ascendante (SFS) qui fait partie des méthodes de type wrapper a été largement utilisée pour la reconnaissance de caractères manuscrits [149,150]. Cette méthode basée sur une recherche heuristique (ou séquentielle), utilise un algorithme itératif dont le principe est de commencer avec un ensemble vide, puis évalue toutes les caractéristiques séparément. Celle qui optimise un critère d'évaluation, c'est-à-dire, qui provoque une erreur de classification la plus minimale possible est sélectionnée pour former un sous ensemble de caractéristiques sélectionnées. Ensuite, à chaque itération, une caractéristique est choisie parmi celles restantes, qui, combinée avec le sous ensemble précédemment sélectionné fournit le meilleur taux de classification. Les caractéristiques ainsi choisies sont supposées être pertinentes.

La méthode hybride proposée, nommée ACP-SFS permet de sélectionner un ensemble de caractéristiques jugées pertinentes parmi celles obtenues par la méthode de réduction (ACP). Cette technique est semblable aux méthodes de type ''embedded'', mais utilise l'ACP au lieu d'une technique de type ''filter'' pour réduire le nombre de caractéristiques initiales. L'idée de cette approche est basée sur le fait que le choix d'un ensemble de caractéristiques ne signifie pas forcément la sélection d'un ensemble composé seulement de variables jugées pertinentes.

Il peut y avoir des caractéristiques non pertinentes, mais qui offrent une meilleure performance, prise avec d'autres variables.

Pour évaluer cette approche, nous avons utilisé une partie de la base USPS à cause du temps de calcul énorme requis par SFS. Pour ce faire, nous avons extrait 1000 images de chiffres manuscrits, dont 60 images par classe pour l'apprentissage et les 40 autres images par classe pour le test. Dans cette expérience, nous avons considéré comme caractéristiques les coefficients de la sous-bande d'approximation obtenue au premier niveau de décomposition, en considérant les ondelettes de Haar, Daubechies d'ordre 4, Coiflet d'ordre 1, Symlet d'ordre 4 et Symlet d'ordre 8. Ces caractéristiques sont ensuite utilisées par le classifieur SVM pour discriminer les différentes classes des chiffres manuscrits.

Le tableau 3.10 regroupe les taux de reconnaissance obtenus sur la base USPS en considérant les différentes caractéristiques à savoir:

- Les coefficients de la sous bande d'approximation,
- Les caractéristiques obtenues par application de la méthode de réduction (ACP),
- Les caractéristiques obtenues par la méthode de sélection (SFS),
- Les caractéristiques obtenues par la méthode ACP-SFS.

Ondelette	Nombre de caractéristiques	Taux de reconnaissance	Nombre de caractéristiques extraites par l'ACP	Taux de reconnaissance	Nombre de caractéristiques sélectionnées par SFS	Taux de reconnaissance	Nombre de caractéristiques réduites par PCA+SFS	Taux de reconnaissance
haar	64	95.00%	29	95.50%	36	96.25%	23	96.75%
db4	64	95.50%	32	96%	43	97%	31	97.25%
coif1	64	94.75%	39	96.75%	42	97%	25	97.25%
sym4	64	96.50%	39	96.50%	36	96.75%	30	97.25%
sym8	64	96.25%	35	96.75%	52	97%	29	97.25%

Tableau 3.10. Taux de reconnaissance obtenus sur la base de données USPS

Les résultats obtenus montrent une nette amélioration au niveau des taux de reconnaissance enregistrés en considérant la combinaison des deux techniques l'ACP et SFS. Il se trouve que la dimension des vecteurs de caractéristiques obtenue est réduite à plus d'une moitié, comparativement aux vecteurs initiaux. De plus, on constate que les taux de reconnaissances obtenus par ACP-SFS sont pratiquement indépendants du type de l'ondelette.

3.10. Conclusion

Ce chapitre est organisé en trois parties, et dans chacune d'elle, nous avons apporté une contribution.

Dans la première partie, nous avons mené une étude détaillée sur l'application de la transformée en ondelettes à la reconnaissance des caractères manuscrits. A cet effet, nous avons réalisé plusieurs expériences en vu de déterminer la meilleure ondelette et la meilleure sous-bande image qui permettent de discriminer mieux les caractères. Les résultats expérimentaux menés sur la base de données standard MNIST révèlent que l'ondelette Symlet d'ordre 8 surpasse les autres types d'ondelettes. Ils montrent également que les caractéristiques extraites de la sous-bande image d'approximation suffisent pour atteindre un taux de reconnaissance satisfaisant (98.76%). Ce résultat a été validé en le comparant avec ceux fournis par d'autres techniques employant la TOD.

Dans la deuxième partie, nous avons mené une étude comparative entre les descripteurs basés sur les ondelettes avec d'autres types de descripteurs de l'état de l'art, tels que Gabor, HOG, LBP et BOF. Cette comparaison effectuée sur 4 bases de chiffres manuscrits, montre que HOG surpasse les autres descripteurs. Compte tenu de ce résultat, nous avons combiné la TOD avec le descripteur HOG pour la reconnaissance des chiffres manuscrits. Les résultats obtenus révèlent que la méthode proposée est efficace puisque les taux de reconnaissance ont été améliorés pour la plupart des bases de données testées.

La dimension des descripteurs est souvent élevée et ce, quelque soit la technique de caractérisation. De plus, ces caractéristiques peuvent être redondantes et ne sont toutes pertinentes. Pour surmonter cet inconvénient, nous avons proposé dans la troisième partie de ce chapitre, une méthode PCA-SFS qui consiste à réduire d'abord le nombre de caractéristiques par l'intermédiaire de l'ACP. Ensuite, la méthode SFS est appliquée à cet ensemble réduit de caractéristiques pour en extraire les plus pertinentes. Les résultats obtenus sur la base USPS réduite avec les descripteurs ondelettes sont prometteurs.

Chapitre 4

Reconnaissance des chiffres manuscrits à base des réseaux de neurones convolutifs

4.1. Introduction

Le choix des caractéristiques pertinentes demeure une tâche difficile en reconnaissance des caractères manuscrits. En effet, de nombreuses techniques de caractérisation peuvent être testées avant de trouver une description adaptée au problème de discrimination des caractères. De plus, ces caractéristiques peuvent être redondantes, et le plus souvent, elles conviennent pour une application bien particulière. Par conséquent, elles sont très peu génériques.

Pour faire face à ces inconvénients, une approche plus récente et qui est actuellement très en vogue est proposée. Cette approche, issue de l'intelligence artificielle, est connue sous le nom d'apprentissage profond ou *Deep Learning*. Elle ne requiert aucun choix de techniques spécifiques d'extraction de caractéristiques puisqu'elles sont générées de manière automatique par apprentissage à partir d'un ensemble de données.
On s'intéresse dans ce chapitre à un modèle d'apprentissage profond très populaire, à savoir les réseaux de neurones convolutifs (Convolutional Neural Network ou CNN) et leur application à la reconnaissance des caractères manuscrits.

Nous commençons par rappeler les fondements des réseaux de neurones classiques. Quelques exemples de réseaux de neurones tels que le perceptron multicouches, la machine de Boltzmann restreinte, et les auto-encodeurs sur lesquels se sont fondés les modèles d'apprentissage profond sont brièvement rappelés. Ensuite, nous exposons brièvement les principaux modèles d'apprentissage profond, et d'une manière plus détaillée, les modèles neuronaux convolutifs. Nous proposons finalement un réseau CNN qui intègre la transformée en ondelettes et son application dans le domaine de reconnaissance des chiffres manuscrits.

4.2. Réseaux de neurones classiques

L'évolution de la théorie des réseaux de neurones artificiels ou formels est liée directement au développement des travaux biologiques sur le cerveau humain. Ce dernier est l'organe de commande le plus complexe et le moins connu de la biologie de l'homme ou de l'animal.

Les cellules nerveuses, appelées neurones, sont les éléments de base du système nerveux central. Ce dernier en possède des centaines de milliards de neurones qui s'interagissent entre eux par l'intermédiaire d'un ensemble de connexions. Les neurones présentent des caractéristiques qui leur sont propres et qui se retrouvent au niveau des quatre fonctions spécialisées qu'ils assument:

- recevoir des signaux en provenance des neurones voisins à travers ses dendrites;
- intégrer ces signaux;
- engendrer un influx nerveux;
- transmettre l'influx nerveux aux neurones voisins par l'axone.

En se basant sur ces notions biologiques, le premier neurone artificiel ou formel a été modélisé par Pitts en 1943[151].

4.2.1. Neurone formel

Un neurone formel est une modélisation simplifiée du neurone biologique (Figure 4.1).

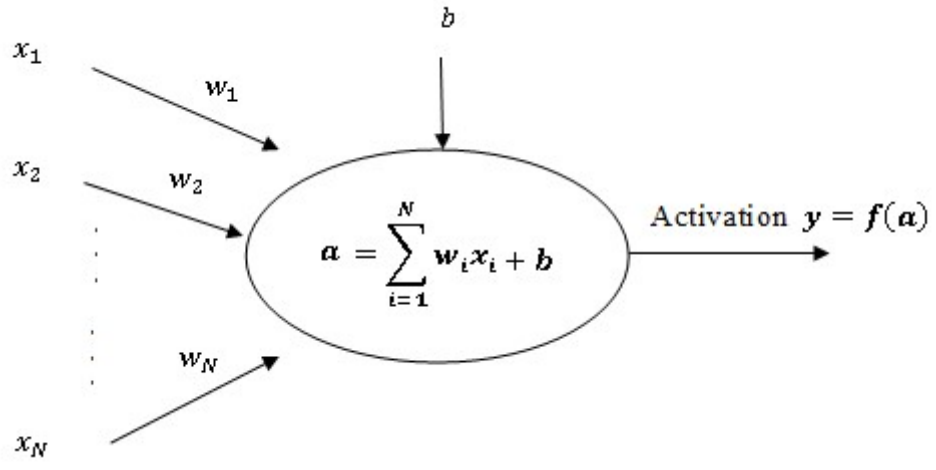


Figure 4.1.Schéma d'un neurone formel à N entrées

Un neurone formel est donc caractérisé par:

- Ses entrées x_i pondérées par des poids w_i, modélisant les poids synaptiques des différentes entrées.
- Un biais b représentant le seuil d'activation du neurone.
- L'activité du neurone déterminée par:

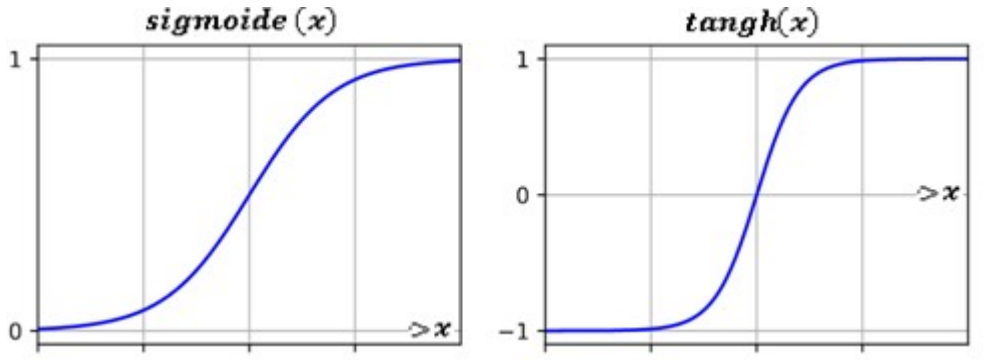
$$a = \sum_{i=1}^{N} w_i x_i + b \tag{4.1}$$

En sortie, le neurone délivre une valeur $y = f(a)$. La fonction f est appelée fonction d'activation.

Plusieurs fonctions d'activation peuvent être envisagées, mais le plus souvent, on fait appel à des fonctions ramenant le résultat à l'intérieur de bornes prédéfinies. Les plus communément utilisées sont les fonctions sigmoïde et tangente hyperbolique (Figure 4.2). Leurs expressions analytiques sont données par:

$$sigmoide(x) = \frac{1}{1 + e^{-x}} \tag{4.2}$$

$$tangh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{4.3}$$



(a) (b)

Figure 4.2. Fonctions d'activation: (a) Sigmoide, (b) Tangente hyperbolique

4.2.2. Réseaux de neurones artificiels

Un réseau de neurones artificiels est un assemblage de neurones formels connectés entre eux selon une architecture donnée. On distingue essentiellement deux types de réseaux de neurones: les réseaux non bouclés et les réseaux de neurones bouclés ou récurrents.

Les réseaux de neurones non bouclés sont souvent organisés en couches (une couche d'entrée, une couche de sortie et une ou plusieurs couches cachées ou carrément aucune couche

cachée). Dans un tel réseau, le flux de l'information circule des entrées vers les sorties sans retour en arrière. Les connexions entre les neurones de deux couches successives peuvent être complètes (tous les neurones d'une couche sont reliés à tous les neurones de la couche suivante) ou partielles (chaque neurone d'une couche partage un lien avec un seul ou quelques neurones d'une autre couche). Les connexions entre les neurones d'une même couche peuvent être également interdites ou non. Parmi les réseaux de neurones non bouclés, le perceptron multicouches est le plus populaire.

Contrairement aux réseaux de neurones non bouclés, les réseaux récurrents admettent tout type de connexion, c'est-à-dire, qu'un neurone peut être connecté à n'importe quel autre y compris lui-même. Ils possèdent des connexions sous forme de boucles ou avec des connexions orientées dans les deux sens.

4.2.3. Apprentissage des réseaux de neurones

Le comportement d'un réseau de neurone est en grande partie réglé par sa connectivité et plus particulièrement par les poids synaptiques des liens entre les neurones.

L'apprentissage est un aspect intrinsèque de l'intelligence. L'apprentissage d'un réseau de neurone est le processus de détermination de ces poids synaptiques. En général, c'est un processus graduel, itératif où les poids du réseau sont modifiés plusieurs fois avant d'atteindre leurs valeurs finales. L'apprentissage se fait de manière supervisée ou non supervisée comme pour la classification.

L'apprentissage non supervisé aussi appelé auto organisation ou sans superviseur, modifie les poids du réseau en fonction de critères internes comme la coactivation des neurones. Les comportements résultants de cet apprentissage sont en général comparables à des techniques de classification non supervisée. Ce type d'apprentissage consiste à organiser les entrées présentées au système en classes ou groupes présentant des caractéristiques communes. La distinction entre ces classes est basée sur des mesures de similarité entre les entrées.

Dans l'apprentissage supervisé, la sortie désirée ou correcte du réseau à une entrée donnée est connue à priori. L'apprentissage du réseau consiste alors à mesurer la différence entre son comportement actuel et le comportement de référence (la sortie désirée) et corriger ses poids de façon à réduire cette erreur.

4.2.4. Exemples de réseaux de neurones

4.2.4.1. Perceptron multicouches

Le perceptron multicouches (MLP) est un réseau de neurones non bouclé, composé d'une couche d'entrée, d'une ou plusieurs couches cachées et d'une couche de sortie (Figure 4.3). La couche de sortie sert à classifier les vecteurs de caractéristiques correspondant aux données d'entrée. Typiquement, il ya en sortie autant de neurones que de classes à identifier. Le neurone le plus actif de la couche de sortie représente la classe d'appartenance du vecteur d'entrée.

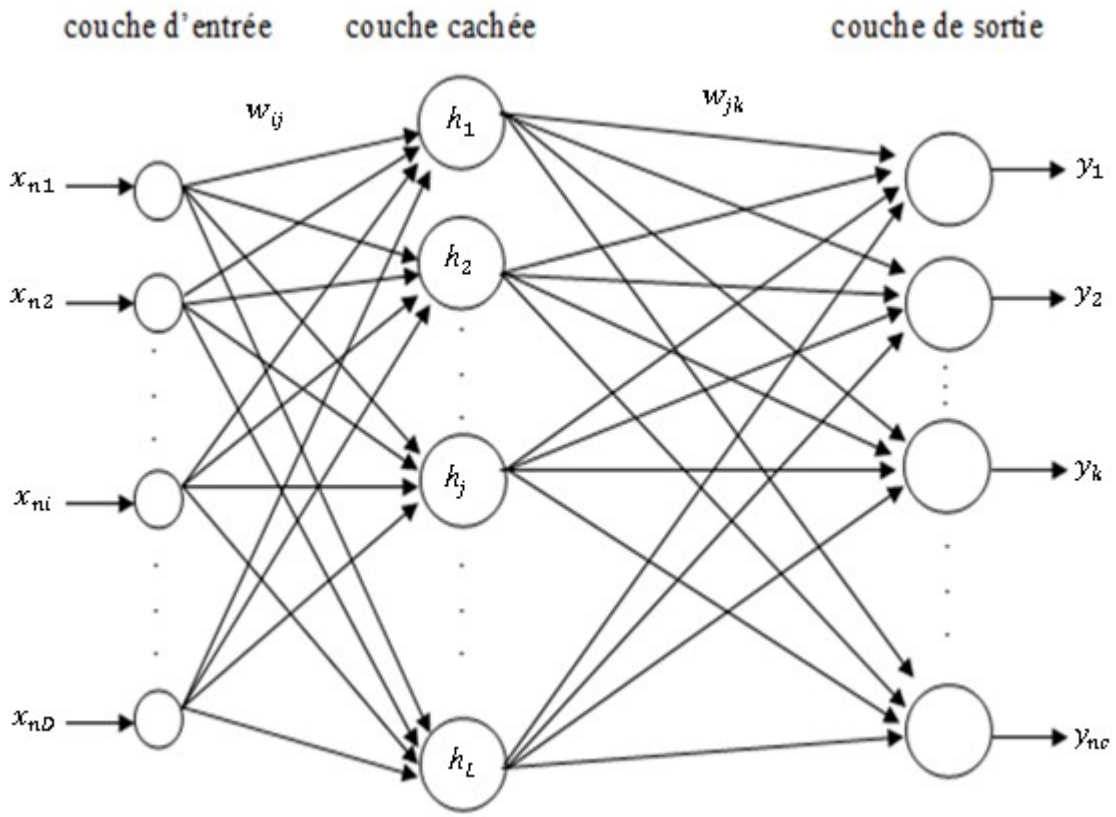


Figure 4.3. Architecture d'un perceptron multicouches

Lors de l'apprentissage, on présente un ensemble d'exemples constitué de couples (entrée, sortie désirée) au réseau qui calcule les sorties correspondantes. Ces calculs s'effectuent depuis la couche d'entrée vers la couche de sortie, de telle sorte que lorsqu'une donnée $x_n = [x_{n1},\ x_{n2}, . .\ .,\ x_{ni}, . .\ .,\ x_{nD}]$ est présentée en entrée, la réponse du $j$ème neurone de la couche cachée h est donnée par:

$$h_j = f\left(\sum_{i=1}^{D} w_{ij} x_{ni} + x_0\right) \tag{4.4}$$

Au niveau de la couche de sortie, la réponse du $k\grave{e}me$ neurone est exprimée par:

$$y_k = f\left(\sum_{j=1}^{L} w_{jk} h_j + h_0\right) \tag{4.5}$$

Les valeurs x_0 et h_0 sont des biais, fixés généralement à 1, f étant la fonction d'activation. L'apprentissage du réseau MLP est effectué en utilisant la règle de rétro-propagation de gradient dont le principe consiste à propager vers les couches internes l'erreur quadratique commise entre la sortie du réseau $f(x_n) = y_n$ et celle désirée d_n.

$$E_n = \frac{1}{2}\sum_{k=1}^{K}(d_{nk} - y_{nk})^2 \tag{4.6}$$

Ensuite, cette erreur est rétro-propagée vers l'entrée du réseau pour modifier l'ensemble des poids synaptiques des différentes couches, de telle sorte qu'à la prochaine itération l'erreur commise entre les sorties du réseau et les sortie désirées soit minimisée [152]. Ce processus est réitéré jusqu'à convergence; c'est-à-dire que l'erreur quadratique moyenne E observée pour l'ensemble des données d'apprentissage devient inférieure à un seuil prédéfini ou atteindre un nombre d'itérations donnée.

$$E = \frac{1}{N}\sum_{n=1}^{N} E_n \tag{4.7}$$

L'adaptation des poids synaptiques par la méthode du gradient est basée sur les formules itératives suivantes:

$$w_{jk}(t+1) = w_{jk}(t) - \eta \Delta w_{jk}(t) \tag{4.8}$$

$$w_{ij}(t+1) = w_{ij}(t) - \eta \Delta w_{ij}(t) \tag{4.9}$$

où η est le pas d'apprentissage.

et:

$$\Delta w_{jk} = \frac{\partial E_n}{\partial w_{jk}} = h_j E_k = h_j y_k (1 - y_k)(d_k - y_k) \tag{4.10}$$

$$\Delta w_{ij} = \frac{\partial E_n}{\partial w_{ij}} = x_i E_j = x_i h_j (1 - h_j) \sum_{k=1}^{nc} w_{jk} E_k \tag{4.11}$$

4.2.4.2. Machine de Boltzmann Restreinte

La machine de Boltzmann restreinte RBM est un réseau de neurones probabiliste, non orienté, non supervisé, qui apprend un modèle généré à partir de données observées. Il est composé de d'une couche visible v qui représente les données, et une couche cachée ou latente h, utilisée pour extraire les caractéristiques de ces données. Les deux couches sont entièrement connectées par l'intermédiaire d'un ensemble de poids w_{ij} et des biais b_i et c_j (Figure 4.4). Cependant, les connexions entre les neurones d'une même couche sont interdites. Cette restriction est d'ailleurs à l'origine du nom RBM, par rapport aux machines de Boltzmann standards qui ne sont pas restreintes et pour lesquelles toutes les connexions sont autorisées.

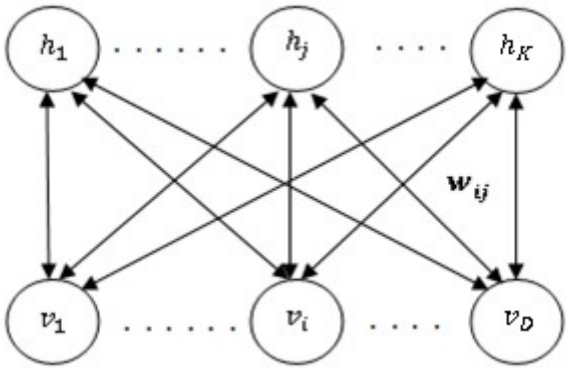


Figure 4.4. Machine de Boltzmann restreinte RBM

Dans un RBM classique, la configuration des connexions entre les unités binaires visibles et les unités binaires cachées est définie par une fonction d'énergie donnée par :

$$E = -\left(\sum_{i,j} w_{ij} v_i h_j + \sum_i b_i v_i + \sum_j c_j h_j \right) \tag{4.12}$$

avec:

w_{ij} le poids de la connexion entre le neurone i et le neurone j.

v_i est l'état du neurone visible i.

h_j est l'état du neurone caché j.

b_i et c_j sont respectivement les biais du neurone i et du neurone caché j.

La probabilité conjointe d'avoir une configuration (v_i, h_j) est alors donnée par:

$$P(v = v_i, h = h_j) = \frac{exp\left(-E(v_i, h_j)\right)}{Z} \tag{4.13}$$

Avec Z une constante de normalisation, définie comme fonction de partition:

$$Z = \sum_{v_i} \sum_{h_j} exp\left(-E(v_i, h_j)\right) \tag{4.14}$$

Pour déterminer les paramètres du RBM (poids et biais des neurones cachés et visibles), un algorithme rapide appelé Divergence Contrastive(CD) est utilisé pour entrainer le réseau [153]. Cet algorithme d'apprentissage est basé sur la minimisation d'une énergie, qui se traduit par la minimisation de la distance entre les données originales et les données générées.
La mise à jour des paramètres du réseau RBM, à savoir les poids et les biais des neurones cachés et visibles est formalisée par:

$$w_{ij} = w_{ij} + \eta\left(\langle v_i h_j \rangle_{données} - \langle v_i h_j \rangle_{reconstruction}\right) \tag{4.15}$$

$$b_i = b_i + \eta(\langle v_i \rangle_{données} - \langle v_i \rangle_{reconstruction}) \tag{4.16}$$

$$c_j = c_j + \eta\left(\langle h_j \rangle_{données} - \langle h_j \rangle_{reconstruction}\right) \tag{4.17}$$

Où η est le taux d'apprentissage.
Le terme $\langle v_i h_j \rangle_{données}$ désigne la fréquence avec laquelle l'unité visible i et l'unité cachée j sont activées mutuellement, quand le réseau est stimulé au niveau de la couche visible avec les données d'apprentissage.
Le terme $\langle v_i h_j \rangle_{reconstruction}$ désigne la fréquence avec laquelle l'unité visible i et l'unité cachée j sont activées mutuellement, quand le réseau est stimulé au niveau de la couche cachée avec des données reconstruites.

4.2.4.3. Auto-encodeur

Un auto-encodeur est un type de réseau de neurones non récurent, semblable au perceptron multicouches ayant une couche d'entrée, une couche de sortie ainsi qu'une ou plusieurs

couches cachées. Il a pour but de reconstruire ses entrées, plutôt de les classer. Sa sortie désirée est l'entrée du réseau, non pas un numéro de classe. Par conséquent, son apprentissage est de type non supervisé. Il est souvent utilisé comme un outil de réduction de la dimensionnalité [154]. Le réseau est constitué de deux parties: la première représente la partie de codage $h = f(x)$, et la deuxième représente la partie de décodage $y = g(h)$. Cette architecture est représentée sur la figure 4.5.

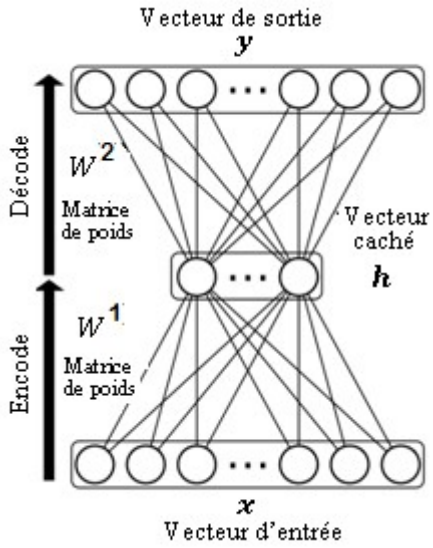


Figure 4.5. Auto-encodeur à une couche cachée

Dans le cas d'un auto-encodeur à une couche cachée, le vecteur d'entrée x de dimension D est transformé par une fonction d'encodage f en une couche cachée h (souvent appelée code):

$$h = f(W^1 x + b^1) \quad (4.18)$$

où W^1 est la matrice des paramètres de la fonction d'encodage, b^1 le vecteur de biais et f la fonction d'activation. Après avoir obtenu le code, une fonction de décodage est appliquée au code pour revenir à la dimension du vecteur d'entrée, ce qui résulte en un vecteur de reconstruction y.

$$y = g(W^2 h + b^2) \quad (4.19)$$

w^2 est la matrice des paramètres de la fonction de décodage. Cette matrice est souvent choisie de telle sorte: $W^2 = (W^1)^T$.

L'apprentissage d'un auto-encodeur a pour but de déterminer à partir des données d'apprentissage, les paramètres W^1, b^1, b^2 qui minimisent l'erreur quadratique de reconstruction suivante:

$$E(x, y) = \|x - y\|^2 \quad (4.20)$$

4.3. Les réseaux de neurones profonds

Les techniques d'apprentissage profond constituent une classe d'algorithmes d'apprentissage automatique introduits dans le but de se rapprocher de l'intelligence artificielle. Ils peuvent apprendre plusieurs niveaux de représentation dans le but de modéliser des relations complexes entre des données. Ils sont actuellement très utilisés dans de nombreux domaines pour leurs performances.

Les techniques d'apprentissage profond sont basées sur de réseaux de neurones, c'est pourquoi ils sont souvent désignés sous le nom de réseaux de neurones profonds. Le terme « profond » revient au nombre de couches cachées du réseau de neurones. Ces derniers comportent que 1 à 2 couches cachées, tandis que les réseaux profonds peuvent en compter jusqu'à 150. L'entraînement des modèles s'effectue à l'aide d'un ensemble très large de données étiquetées (prototypes) dans le but d'apprendre les caractéristiques directement depuis les données et en même temps de construire le modèle de classification.

En effet, une des grandes différences entre les techniques d'apprentissage profond et les algorithmes d'apprentissage automatique traditionnels (Machine Learning) c'est qu'ils s'adaptent bien. Plus la quantité de données fournie est grande, plus les performances des algorithmes d'apprentissage profond sont meilleurs. De plus, l'étape d'extraction de caractéristiques est indépendante des algorithmes traditionnels (Figure 4.6), alors qu'au niveau d'apprentissage profond, elle fait partie intégrante du réseau neuronal (Figure 4.7).

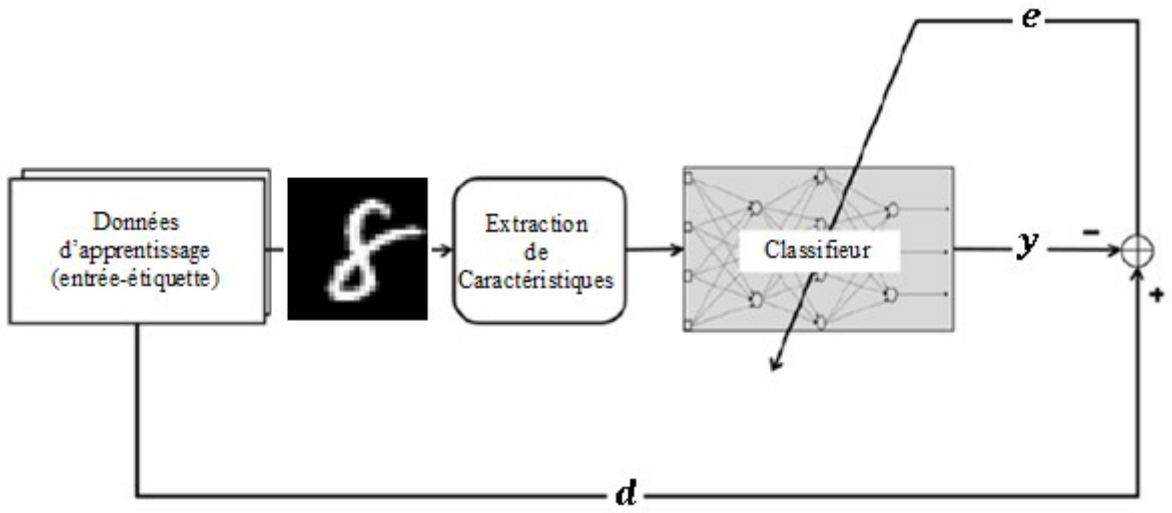


Figure 4.6. Algorithme d'apprentissage automatique traditionnel

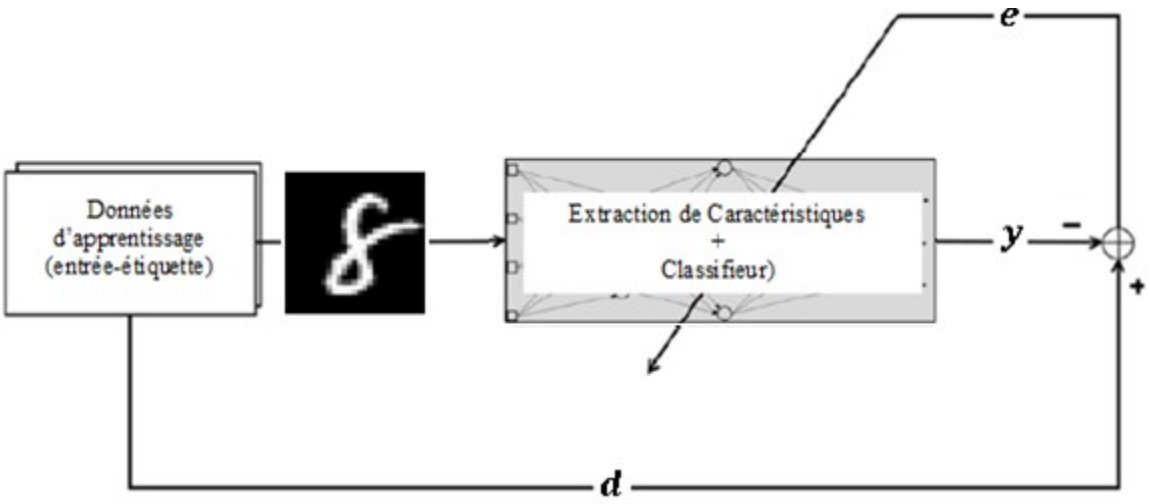


Figure 4.7. Algorithme d'apprentissage profond

Principalement, les techniques d'apprentissage profond peuvent être classées en trois groupes: les machines de Boltzmann profondes, les auto-encodeurs profonds et les réseaux de neurones convolutifs.

4.3.1. Machine de Boltzmann profonde

Une machine de Boltzmann profonde DBM (Deep Boltzmann Machine), encore appelée réseau de croyance profonde ou DBN (Deep Belief Network), est un réseau de neurones formé de l'empilement d'un ensemble de machines de Boltzmann restreintes [155] (Figure 4.8).

Les DBN sont organisés en plusieurs couches, de sorte quel'activation des unités cachées de la première couche deviennent l'entrée de la deuxième couche et ainsi de suite.

Les DBN sont généralement utilisés à des fins de pré-entrainement [156]. Ils ont été proposés pour améliorer les performances des classifieurs supervisés en utilisant les poids synaptiques pré-entrainés pour initialiser les paramètres du perceptron multicouches par exemple. Les DBN ont été utilisés dans des applications de classification d'images et dans la reconnaissance de caractères [157].

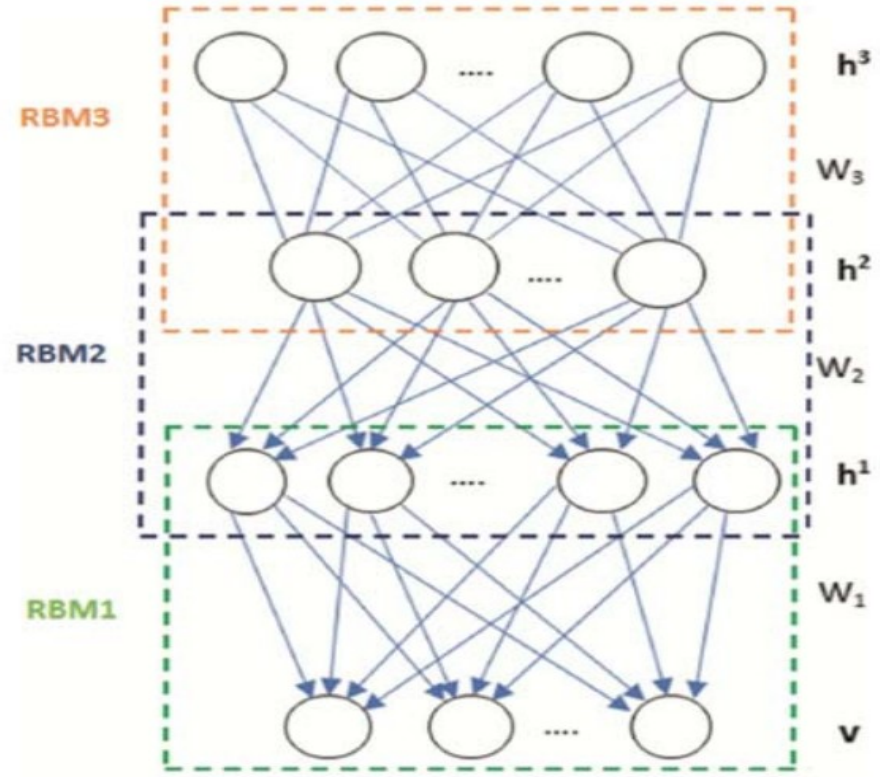


Figure 4.8. Machine de Boltzmann profonde DBM
à deux couches cachées [158]

4.3.2. Auto-encodeur profond

Un auto-encodeur composé de plusieurs couches cachées peut-être vu comme étant un réseau profond [159]. Les auto-encodeurs profonds (ou Deep Auto-Encoders DAE) sont généralement structurés en deux parties: encodeur-décodeur. Le décodeur contient autant de couches cachées que l'encodeur. Le nombre de neurones d'une couche cachée du décodeur est égal au nombre de neurones de la couche cachée (occupant la même position) de l'encodeur. La couche code sépare les couches cachées de l'encodeur de celles du décodeur (Figure 4.9). Les paramètres de DAE (poids synaptiques et biais) sont déterminés par le même algorithme d'apprentissage utilisé dans le cas de l'auto-encodeur à une couche cachée. Une fois que le réseau est entrainé, la couche cachée appelée code est utilisée comme vecteur de caractéristiques représentatif des données initiales. Les DAE ont été utilisés dans de nombreuses applications, telle que la reconnaissance de caractères [160,156].

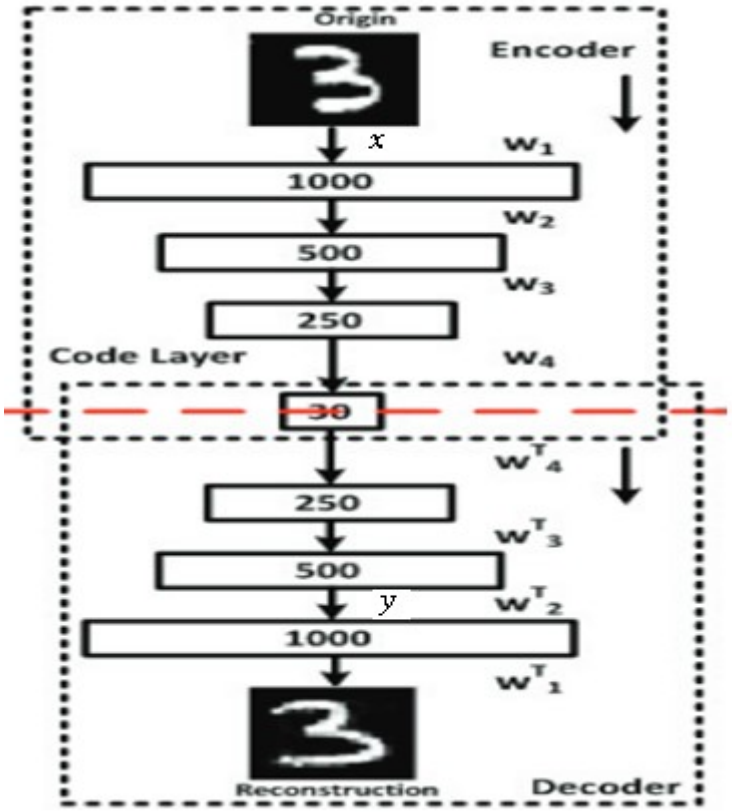


Figure 4.9. Auto-encodeur à plusieurs couches cachées [156]

4.3.3. Réseaux de neurones convolutifs

Un réseau de neurones convolutif CNN (Convolutional Neural Network) est un réseau de neurones avec une ou plusieurs couches de convolution. C'est un cas particulier de perceptron multicouches adapté aux images. Ce type de réseaux a été inspiré par les travaux effectués sur le fonctionnement biologique du cortex visuel chez les mammifères [161]. Il est basé sur trois idées architecturales:

- Des champs récepteurs locaux associés à des convolutions qui permettent de détecter des caractéristiques élémentaires sur l'image, formant ainsi une carte de caractéristiques.
- Un principe de partage des poids qui consiste à apprendre les mêmes paramètres (ou poids) d'une convolution et par conséquent extraire les mêmes caractéristiques pour toutes les positions sur l'image. C'est l'idée clé des CNNs permettant de réduire considérablement la complexité en diminuant le nombre de paramètres à apprendre, et d'avoir ainsi des architectures multicouches qui opèrent sur des entrées de grande dimension tout en étant de taille raisonnable (ce qui n'était pas réalisable avec les

MLPs). De plus, le partage des poids permet d'améliorer la capacité de généralisation du réseau.

- Des opérations de sous-échantillonnage ou de pooling qui permettent de réduire la sensibilité aux translations, ainsi que le coût du traitement.

La structure du CNN a été proposée initialement par Fukushima en 1980 [162] grâce à ses travaux sur le Néocognitron. Cependant, il n'a pas été largement popularisé à cause de la complexité de l'algorithme d'apprentissage. Ce n'est que dans les années 1990, que Lecun [163] a popularisé les réseaux de neurones convolutifs en appliquant son premier réseau développé LeNet-1 à la reconnaissance des caractères.

Ces deux dernières décennies, avec le développement de ressources informatiques spécifiques (GPU) et la disponibilité de bases de données plus larges, d'autres variantes de réseaux convolutifs de plus en plus profonds ont fait leur apparition. Dans la section suivante, nous présenterons en détails réseaux de neurones convolutifs.

4.4. Description des réseaux de neurones convolutifs

L'architecture d'un CNN peut être divisée en deux parties (Figure 4.10). La première a pour but d'extraire des caractéristiques. Elle est composée d'une succession de couches de convolution et de couches de sous-échantillonnage ou de pooling. La deuxième partie est formée d'une part, d'un ensemble de couches totalement connectées destinées à construire des caractéristiques de haut niveau en combinant celles de bas niveau issues de l'étage précédent, et d'autre part, d'une couche de sortie qui classe les vecteurs de caractéristiques des données d'entrée.

L'alternance entre convolution et sous-échantillonnage donne une structure pyramidale au réseau CNN. Ainsi, les couches supérieures représentent des caractéristiques de plus en plus globales de l'entrée, car leur champ perceptif correspond à une plus grande partie de l'entrée.

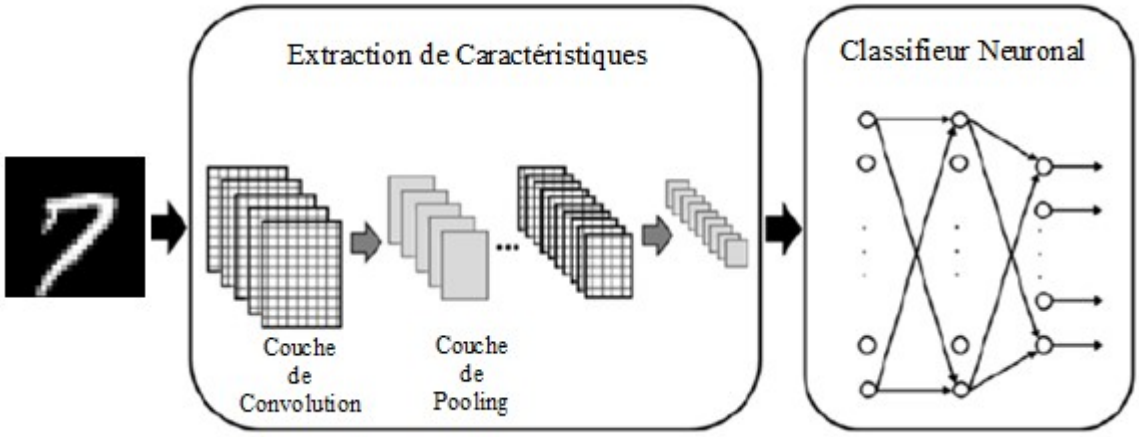


Figure 4.10. Exemple de réseau de neurones convolutif

4.4.1. Couche de convolution

Les couches de convolution constituent la partie la plus importante des réseaux CNN. Le terme ‘’convolution’’ vient du fait que les réseaux CNN utilisent des opérations de produit de convolution 2D. Le produit de convolution entre une image I et un filtre F est défini par:

$$S(i,j) = \sum_m \sum_n I(i-m, j-n)F(m,n) \tag{4.21}$$

Ce produit indique que le filtre interagit seulement avec une petite région de l'image d'entrée, spécifiée par sa taille $(F_m x F_n)$. Pour effectuer cette opération sur toute l'image, deux paramètres doivent être fixés au préalable: le pas de déplacement du filtre ou ‘’stride’’ s et le zero padding p. Le zéro padding est réalisé par ajout de zéros aux bordures de l'image d'entrée en vu de traiter les pixels situés dans ces bordures. Ces deux paramètres permettent de contrôler la taille de la carte de caractéristiques résultante du produit de convolution.

L'opération de convolution produit en sortie une image appelée ‘’carte de caractéristiques’’ ou ‘’feature map’’ composée de neurones. En général, dans une couche de convolution, on n'applique pas qu'un seul filtre de convolution, mais un ensemble F filtres. On obtient alors une pile de F cartes de caractéristiques. Les coefficients de chaque filtre sont assimilés à des poids synaptiques.

D'une manière générale, dans une couche de convolution (l) (couche l du réseau), chaque carte de caractéristiques M_j^l (où j est la position occupée par cette carte) est le résultat d'une somme de convolution des cartes de la couche précédente M_i^{l-1}par son noyau de convolution respectif. Un biais b_j^l est ensuite ajouté et le résultat est donné par la relation suivante [164]:

$$M_j^l = b_j^l + \sum_{i \in R_j} M_i^{l-1} * F_{ij}^l \qquad (4.22)$$

Où R_j est une région de M_i^{l-1} localisée par le $j\grave{e}me$ filtre.

Les poids synaptiques F_{ij}^l de chaque filtre ainsi que les biais constituent les paramètres du réseau CNN. Ils sont ajustés durant la phase d'apprentissage. Notons que les paramètres du noyau de convolution sont les mêmes pour tous les neurones de la même carte, ce qui entraine une réduction du nombre de paramètres du réseau CNN.

Finalement, une couche de convolution est définie par le nombre et la taille des cartes de caractéristiques (ou de convolution), la taille des noyaux (ou filtres) de convolution ($F_{m,}F_n$), le pas de déplacement du noyau sur l'image et le schéma de connexion à la couche précédente [165]. Un exemple de convolution 2D est illustré par la figure 4.11.

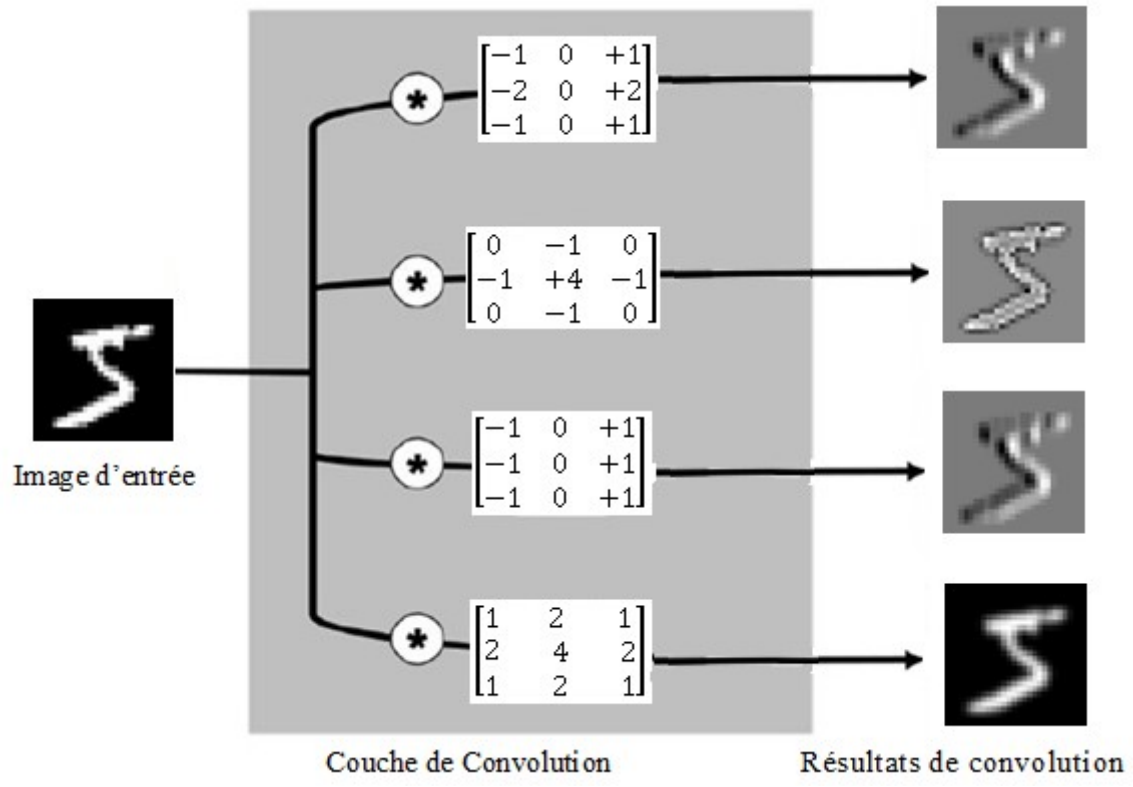


Figure 4.11. Exemple d'une convolution 2D

4.4.2. Couche de normalisation par lot

La normalisation par lot (batch normalization) est une méthode de régularisation introduite dans l'apprentissage d'un réseau pour améliorer les capacités de généralisation du modèle appris. Cette couche est souvent insérée entre la couche de convolution et la couche

d'activation ReLU. Elle consiste à normaliser les activations d'une couche de convolution (l) avant de les envoyées à la couche suivante. Ainsi, chaque unité ou neurone x_j de la carte d'activation M_j^l est normalisé en lui soustrayant la moyenne μ, puis on le divisant par l'écart type σ, telle que [166,167]:

$$x_j^{norm} = \frac{x_j - \mu}{\sigma} \tag{4.23}$$

Où μ et σ sont respectivement des vecteurs des moyennes et des écarts type correspondant à chaque unité (ou neurone). Ces vecteurs sont calculés sur des lots (ou batch) de données d'apprentissage

Cette opération permet non seulement d'accélérer la phase d'apprentissage, elle rend également le réseau plus stable. En outre, elle permet de réduire les écarts grossiers qui peuvent avoir un trop gros impact sur la mise à jour des poids. Cependant, cette normalisation peut réduire le pouvoir expressif du réseau. Pour éviter ce problème, des paramètres de mise en échelle γ et de translation β sont introduits pour chaque élément normalisé x_j^{norm}. Ceci se traduit par l'équation:

$$y_j = \gamma x_j^{norm} + \beta \tag{4.24}$$

où les variables γ et β sont des paramètres à déterminer par apprentissage.

4.4.3. Couche de correction (ReLU)

Cette couche est généralement indissociable de la couche de convolution, elle a pour objet de calculer la fonction d'activation des neurones d'une couche de convolution. Considérons M_j^{l-1} la $j\grave{e}me$ carte de caractéristiques de la couche de convolution $(l-1)$. Le résultat de l'application de la fonction d'activation à cette couche est donné par:

$$M_j^l = f\left(M_j^{l-1}\right) \tag{4.25}$$

Où f est une fonction d'activation.

Les fonctions d'activations telles que sigmoïde ou tangente hyperbolique peuvent être utilisées. Cependant, lorsque le réseau est très profond, il devient difficile à entrainer à cause du problème ''vanishing gradient'' qui survient lors de l'apprentissage. Pour éviter ce problème, ces fonctions ont été remplacées par une fonction nommée ReLU (Rectified Linear Unit) [168]. Cette fonction représentée sur la figure 4.12 est définie par l'équation suivante:

$$ReLU(x) = max(0, x) \tag{4.26}$$

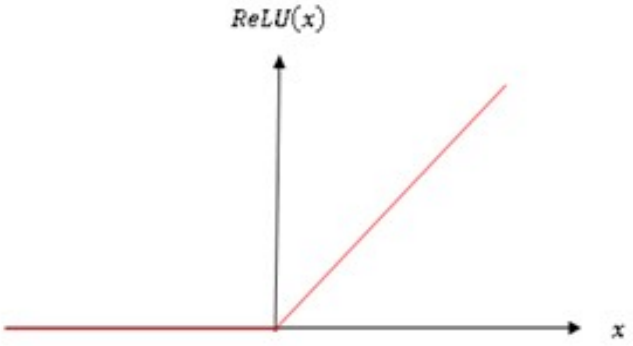


Figure 4.12. Fonction d'activation ReLU

Notons que ReLU est une fonction non linéaire différentiable qui force les neurones à ne retourner que des valeurs positifs. Son évaluation est plus rapide par rapport à celles des fonctions d'activation classiques comme les fonctions sigmoide et tangente hyperbolique qui font appel aux calculs des exponentielles.

4.4.4. Couche de sous échantillonnage ou de pooling

La couche de pooling ou de sous-échantillonnage est souvent insérée entre deux couches de convolution. L'opération de sous échantillonnage est appliquée indépendamment sur chaque carte de caractéristiques issues de la couche de convolution précédente. Elle permet de réduire la résolution spatiale des cartes de caractéristiques, tout en préservant les informations les plus importantes qu'elles contiennent. Par conséquent, le nombre de paramètres du réseau est réduit, ce qui rend son apprentissage plus rapide. En outre, elle rend le réseau moins sensible aux faibles rotations et translations.

Les techniques de sous-échantillonnage les plus utilisées sont le max-pooling et average-pooling (Figure 4.13). Le max-pooling permet d'extraire dans chaque région R_j la valeur maximale et de l'attribuer à la carte de caractéristiques réduite. Cet opérateur est formulé par l'équation suivante:

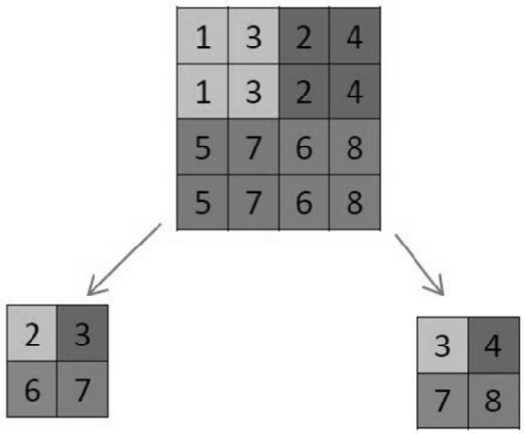
$$P_j^l = \max_{i \in R_j}(M_i^l) \tag{4.27}$$

Quant à l'average-pooling, elle consiste à calculer la valeur moyenne des entrées d'une région R_j et de l'attribuer à la carte de caractéristiques réduite. Cette fonction est exprimée par la relation suivante:

$$P_j^l = \frac{1}{|R_j|} \sum_{i \in R_j} M_i^l \tag{4.28}$$

où $|R_j|$ représente la taille de la région R_j.

Généralement, la région R_j est choisie de petite taille pour ne pas perdre trop d'informations. Les choix les plus communs sont des régions adjacentes de taille (2x2) ou (3x3) qui ne se recouvrent pas. Par conséquent, le pooling produit le même nombre de cartes de caractéristiques qu'en entrée, mais de tailles plus petites.



(a) (b)

Figure 4.13. Pooling avec une cellule de (2x2) et un pas de 2
(a) average pooling, (b) max pooling

Le principal avantage de l'opérateur max-pooling est qu'il est efficace lorsqu'on s'intéresse aux contours des objets, tandis qu'average-pooling est efficace pour la détection de structures très fines (de faible gradient). En pratique, l'opérateur de max-pooling est le plus utilisé vu qu'il fournit de meilleurs résultats comparativement à celui d'average-pooling [169]. Toutefois, dans la plupart des cas, ces opérateurs présentent certaines limites, du fait que chacun d'eux convient à un type de données bien particulier. Pour pallier ces lacunes, il existe d'autres méthodes de pooling, telles celle basée sur la combinaison des deux opérateurs max-pooling et average-pooling [170], la méthode de pooling-stochastique [171], ou encore la méthode proposée dans [172] qui consiste à remplacer la couche de pooling par la transformée en ondelettes discrète (TOD).

4.4.5. Couche totalement connectée

La partie composée de couches de convolution, ReLU et de pooling fournit au final un ensemble de caractéristiques sous forme de cartes 2D. Ces cartes sont concaténées en un vecteur de caractéristiques, appelé code CNN. Ce code constitue l'entrée de la deuxième partie du réseau. Elle est composée d'une ou de plusieurs couches totalement connectées, assimilées aux couches cachées du perceptron multicouches MLP. Chaque neurone d'une couche donnée est connecté à tous les neurones de la couche précédente à travers des poids synaptiques. Leurs activités sont calculées avec une multiplication matricielle entre leurs entrées x et leurs poids synaptiques correspondant W, à laquelle un biais b est rajouté. Le résultat est passé à la fonction d'activation f.

$$y = f(W.x + b) \tag{4.29}$$

4.4.6. Couche de perte (LOSS)

La couche de perte constitue la dernière couche du réseau CNN. Elle vise à optimiser le modèle CNN en minimisant une fonction de coût, appelée aussi fonction de perte. La fonction de perte softmax cross entropie est souvent utilisée dans les réseaux convolutifs [173,174]. Elle est définie par:

$$E(\theta) = -\frac{1}{N}\sum_{n=1}^{N} E_n(x_n, \theta) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{nc} d_{nk} log\big(y_{nk}(x_n, \theta)\big) \tag{4.30}$$

Où:

θ un vecteur contenant les paramètres du réseau, d_{nk} indique la vraie classe de la donnée x_n et $y_{nk}(x_n, \theta)$ correspond à la sortie estimée par le réseau (nc est le nombre de classes). La sortie $y_{nk}(x_n, \theta)$ est obtenue en appliquant la fonction d'activation softmax à la dernière couche totalement connectée. Cette fonction permet de calculer la distribution de probabilités sur les classes de sortie du réseau de neurones convolutif.

$$y_k(x, \theta) = \frac{exp\big(a_k(x, \theta)\big)}{\sum_{k=1}^{nc} exp\,\big(a_k(x, \theta)\big)} \tag{4.31}$$

Avec:

$0 \leq y_k \leq 1$ et $\sum_{k=1}^{nc} y_k = 1$.

a_k l'activité du $kième$ neurone de la couche de sortie et x correspond à la donnée présentée à l'entrée du réseau.

4.4.7. Apprentissage des réseaux de neurones convolutifs

L'apprentissage profond consiste à entrainer un réseau de neurones constitué d'une série de modules, chacun représentant une étape de traitement. Chaque module comporte des paramètres similaires aux poids des réseaux de neurones classiques. L'ensemble de ces paramètres sont ajustés de manière à rapprocher la sortie estimée par le système de la sortie de la vérité terrain. Cela peut se faire par l'intermédiaire d'un algorithme d'optimisation basé sur la descente de gradient stochastique par lot (ou mini-batch).

L'optimisation d'un réseau de neurones consiste à minimiser une fonction $E(\theta)$ en mettant à jour les paramètres du réseau CNN.

Soit θ le vecteur contenant l'ensemble des paramètres du réseau CNN, et $E(\theta, y, d)$ la fonction de perte à optimiser. La mise à jour des paramètres du réseau CNN est donnée par cette équation:

$$\theta(t+1) = \theta(t) - \eta \frac{1}{B_s} \sum_{n=1}^{B_s} \nabla_\theta E_n(\mathrm{t})\big(\theta(t), y_n(t), d_n(t)\big) \tag{4.32}$$

Où: η est une constante positive appelée taux ou pas d'apprentissage, B_s est la taille du lot. $B(t) = \big(y_n(t), d_n(t)\big)_{n\in[1,B_s]}$ est le lot de données tiré à l'itération t, $y_n(t)$ et $d_n(t)$ sont respectivement la sortie du réseau estimée et l'étiquette de la donnée $x_n(t)$.

$E(t) = \frac{1}{B_s}\sum_{n=1}^{B_s} E_n(t)$ est l'approximation stochastique de la fonction de coût globale à l'itération t sur le lot $B(t)$, décomposée en une somme de fonctions différentiables $E_n(t)$ liées à chaque paire $\big(x_n(t), d_n(t)\big)$.

Un élément important dans l'algorithme de descente de gradient est le pas (ou taux) d'apprentissage. Si le pas trop petit, l'algorithme devra effectuer un grand nombre d'itérations pour converger et prendra beaucoup de temps. Inversement, si le pas est trop élevé, l'algorithme risque de diverger et de s'éloigner ainsi de la bonne solution. De plus, ce pas d'apprentissage est global, ce qui signifie que tous les neurones utilisent le même taux, alors que toutes les données ne suivent pas forcément la même distribution et donc ne nécessitent pas d'adapter le réseau de la même manière.

Pour pallier ces problèmes, de nombreuses variantes de l'algorithme de descente de gradient stochastiques ont été proposées. Parmi elles, on cite la méthode Adaptative Adam [175] que nous avons utilisée dans nos tests. Adam est l'un des algorithmes les plus récents et les plus efficaces pour l'optimisation par descente de gradient. Il calcule un taux d'apprentissage

adaptatif pour chaque paramètre. En outre, le gradient dépend des estimations adaptatives des moments de premier et second ordre. La mise à jour des paramètres θ du réseau s'effectue comme suit:

$$\forall\, n, \big(m(t+1)\big)_n = \beta_1\big(m(t)\big)_n + (1-\beta_1)\Big(\nabla_{\theta_t} E\big(\theta(t)\big)\Big)_n \tag{4.33}$$

$$\forall\, n, \big(v(t+1)\big)_n = \beta_2\big(v(t)\big)_n + (1-\beta_2)\Big(\nabla_{\theta_t} E\big(\theta(t)\big)\Big)_n^2 \tag{4.34}$$

$$\forall\, n, \big(\theta(t+1)\big)_n = \big(\theta(t)\big)_n - \eta\frac{\sqrt{1-\beta_2}}{1-\beta_1}\frac{\big(m(t)\big)_n}{\sqrt{\big(v(t)\big)_n + \varepsilon}} \tag{4.35}$$

$\eta, \varepsilon > 0$ et $\beta_1, \beta_2 \in]0,1[$.

Généralement, $\eta \in]0,1]$ et les valeurs des paramètres β_1, β_2 et ε sont fixés comme suit: $\beta_1 = 0.9, \beta_2 = 0.999$ et $\varepsilon = 10^{-8}$.

$m(t)$ et $v(t)$ sont respectivement les moments de premier (la moyenne) et de second (variance non-centrée) ordres du gradient.

Une autre considération pratique dont on doit tenir compte lors de l'apprentissage profond concerne l'initialisation des poids du réseau de neurones. Cette étape a une grande influence sur la vitesse et la convergence de l'apprentissage. En effet, si les poids sont trop faibles, l'adaptation des poids des couches d'entrée devient difficile à cause de l'affaiblissement du gradient à travers les couches. S'ils sont trop importants, les activations deviennent saturées dans le cas de la fonction sigmoïde (ou tangente hyperbolique) et les gradients s'approchent de zéro. Pour pallier cet inconvénient, différentes techniques d'initialisation ont été proposées dans [176,177]. Dans [176], les valeurs initiales des poids sont générées d'une manière aléatoire. En utilisant une distribution uniforme, les poids d'une couche (l) peuvent être initialisés dans l'intervalle $w_{ij} \sim U\left[-\frac{1}{\sqrt{m,}}, \frac{1}{\sqrt{m}}\right]$ (m est la dimension de la couche (l) et U une distribution uniforme), ou dans l'intervalle $w_{ij}^l \sim U\left[-\sqrt{\frac{6}{m_h^l + m_h^{l-1}}}, \sqrt{\frac{6}{m_h^l + m_h^{l-1}}}\right]$ (m_h^l et m_h^{l-1} représentent la dimension des couches cachées (l) et $(l-1)$ respectivement).

4.5. Les réseaux de neurones convolutifs en reconnaissance des caractères manuscrits

La première application des réseaux de neurones convolutifs a été consacrée par LeCun à la reconnaissance des caractères manuscrits [1]. Ce dernier a mis en place LeNet-1; un réseau

convolutif, composé de deux couches de convolution, deux couches d'average-pooling et d'une couche de sortie entièrement connectée. Plus tard, LeNet-1 a été amélioré pour donner naissance à LeNet-4. Il est composé de 5 couches cachées dont deux couches de convolution, deux couches d'average-pooling, une couche entièrement connectée et d'une couche de sortie. Par la suite, une autre architecture LeNet-5 -similaire à LeNet-4- mais plus performante, a été développée pour des bases d'images plus larges (Figure 4.14).

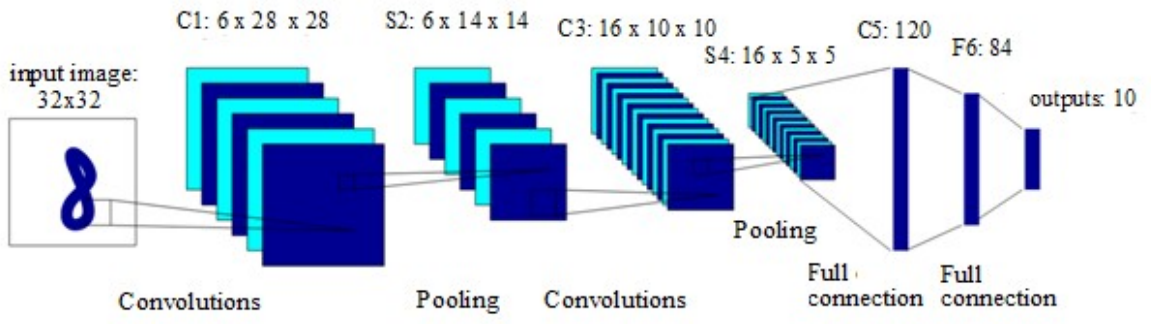


Figure 4.14. Architecture du réseau LeNet-5 [178]

D'autres architectures de CNN intégrant de nouvelles fonctions d'activation ont été développées et appliquées à la reconnaissance des caractères. Dans [158], une architecture basée sur les CNN a été proposée pour la reconnaissance de caractères manuscrits Tifinagh. Ce réseau est composé de 7 couches: 4 couches de convolution et 3 couches entièrement connectées. A la différence des autres couches, la première et la quatrième couche de convolution sont suivies par des couches de max-pooling. Dans la couche de sortie, la fonction softmax est utilisée pour calculer la distribution des probabilités des différentes classes.

Dans [179], une architecture neuronale combinant les réponses de 64 CNNs identiques est proposée pour la reconnaissance de caractères manuscrits perses. Inspiré de la structure LeNet-5, chaque réseau est composé de deux couches de convolution, deux couches de sous échantillonnage de type max-pooling et deux couches entièrement connectées à la sortie du réseau CNN. Dans cette architecture, chaque couche de convolution est suivie d'une couche de max-pooling, permettant ainsi de réduire la dimension spatiale des cartes de caractéristiques obtenues. Les fonctions d'activation ReLU et softmax sont utilisées dans les couches de convolution et de sortie respectivement.

Dans [180], une méthode basée sur un réseau de neurones convolutif CNN a été proposée pour la reconnaissance des caractères manuscrits Bengali. L'image du caractère est soumise au réseau CNN. Il est composé de deux couches de convolution dédiées à l'extraction de caractéristiques, deux couches de sous échantillonnage (avarage-pooling) pour la réduction de la dimension des cartes de caractéristiques et deux couches entièrement connectées dédiées à la classification.

Dans [181], une architecture G-CNN basée sur un réseau de neurones convolutif et les filtres de Gabor a été proposée pour la reconnaissance de chiffres manuscrits. Cette architecture est composée de 3 couches de convolution alternées avec 2 couches de pooling et une couche de sortie de type MLP. Au niveau de la première couche de convolution, 12 filtres de Gabor ont été utilisés, d'une part, pour extraire des caractéristiques selon différentes orientations et échelles, et d'autre part, pour réduire le nombre de paramètres du réseau CNN.

Dans [182], des améliorations ont été apportées à l'architecture du réseau de neurones convolutif ConvNet standard. Cette architecture a été élargie par l'ajout des liaisons supplémentaires entre les couches de pooling et la couche de classification. En outre, au niveau des couches de pooling, les opérateurs max-pooling et average-pooling traditionnels ont été remplacés par le L_p pooling utilisant le noyau Gaussien. Cette nouvelle architecture a été évaluée sur la base de chiffres manuscrits SVHN.

Dans [183], une approche basée sur la combinaison de la transformée en ondelettes avec les réseaux de neurones convolutifs CNNs a été proposée pour la reconnaissance de chiffres manuscrits. L'architecture du CNN utilisé est composée de 3 couches de convolution, 2 couches de max-pooling et 1 couche de sortie entièrement connectée. En considérant les 4 sous-bandes images issues de la décomposition de l'image du caractère par l'intermédiaire de l'ondelette de Haar, deux schémas de combinaison ont été mis en place.
Dans le premier schéma''CNN-WAV2'', les sous-bandes d'approximation et celle issue de la combinaison des 3 sous-bandes de détails ont été appliquées respectivement à 2 réseaux CNNs dont les sorties sont combinées entre elles avec l'opérateur OU logique.
Dans le deuxième schéma ''CNN-WAV4'', les 4 sous bandes images sont appliquées séparément à 4 réseaux CNNs dont les sorties sont combinées par la technique précédente.

4.6. Réseaux de neurones convolutifs proposés

Nous décrivons dans cette section trois réseaux de neurones convolutifs que nous avons développé pour la reconnaissance des chiffres manuscrits. Le premier, nommé ‘’CNN-MLP’’, est un réseau standard, composé des couches de convolution et de pooling qui permettent d’extraire des caractéristiques à partir des images des caractères, et des couches complètement connectées (MLP) destinées à effectuer la classification supervisée des caractères. Le second, que nous désignons par ‘’CNN-SVM’’, possède une architecture identique à celle du CNN-MLP, mais qui effectue la classification des caractères par l’intermédiaire du classifieur SVM. Quant au dernier, il combine la transformée en ondelettes (TOD) avec le CNN-SVM (CNN-MLP) proposé.

4.6.1. CNN- MLP

L’architecture du réseau CNN-MLP que nous avons développée est inspirée du réseau LeNet-5. Il est composé d’une couche d’entrée, 3 couches de convolution, 2 couches de max-pooling et une couche de sortie entièrement connectée (MLP), comme le montre la figure 4.15.

La couche d’entrée reçoit l'image en niveaux de gris de taille 28x28. La première couche de convolution est composée de 25 filtres de taille 5x5 chacun. Le produit de convolution entre chaque filtre avec l’image initiale permet de produire au total 25 cartes de caractéristiques différentes. Dans une carte de caractéristiques, chaque neurone est connecté partiellement à la couche précédente à travers une région définie par le filtre correspondant. Ensuite, la valeur de ce neurone est passée dans la fonction d’activation ReLU pour la prise en compte que des valeurs positives. Ces 25 cartes de caractéristiques de taille 24x24 chacune, sont considérées par la suite comme étant des entrées pour la deuxième couche de max-pooling. Cette couche, ayant pour objectif la réduction de l’espace des caractéristiques, applique sur chaque carte une fenêtre glissante de taille 2x2 avec un pas de déplacement de 2 et un opérateur max-pooling pour ne garder que la valeur maximale. A la sortie de cette couche, nous obtenons 25 nouvelles cartes de taille 12x12 chacune. Les mêmes opérations sont répétées sur la deuxième couche de convolution et la troisième couche de max-pooling en utilisant 50 filtres de convolution. A la sortie de cette dernière, nous obtenons 50 cartes de caractéristiques de taille 4x4 chacune. Celles-ci seront présentées à l’entrée de la dernière couche de convolution composée de 500 filtres de taille 4x4 chacune, produisant ainsi un vecteur de dimension 500. Finalement, la dernière couche du réseau, entièrement connectée, de type MLP, est dédiée à la classification. Elle reçoit en entrée ce vecteur pour fournir en sa sortie 10 valeurs représentant

le nombre de classes à discriminer. Ces valeurs sont par la suite converties en probabilités par l'intermédiaire d'une fonction d'activation softmax. Notons que pour accélérer l'apprentissage, chaque couche de convolution est suivie d'une étape de normalisation.

L'architecture du CNN ainsi décrite est celle adoptée pour la reconnaissance des chiffres manuscrits des deux bases MNIST et CVLSD (Tableau 4.1). De légères modifications ont été apportées à cette architecture pour l'adapter aux bases USPS et SVHN (Tableau 4.1). Ces modifications sont principalement liées à la taille des filtres utilisés dans les couches de convolution.

Dans le cas de la base SVHN, la taille des filtres employés dans la troisième couche de convolution est fixée à 5x5, alors que dans le cas de la base USPS, des filtres de tailles de 3x3 et 5x5 sont respectivement utilisés dans la deuxième et la troisième couche de convolution. En outre, nous avons omis la deuxième couche de pooling dans USPS pour éviter de perdre trop d'information étant donné la taille réduite de ses images.

	USPS 16x16	MNIST 28x28	CVLSD 28x28	SVHN 32x32
1ère couche de convolution	p=0, s=1 25x3x3	p=0, s=1 25x5x5	p=0, s=1 25x5x5	p=0, s=1 25x5x5
Max-pooling	p=0 s=2	p=0 s=2	p=0 s=2	p=0 s=2
2ème couche de convolution	p=0, s=1 50x3x3	p=0, s=1 50x5x5	p=0, s=1 50x5x5	p=0, s=1 50x5x5
Max-pooling	_	p=0 s=2	p=0 s=2	p=0 s=2
3ème couche de convolution	p=0, s=1 500x5x5	p=0, s=1 500x4x4	p=0, s=1 500x4x4	p=0, s=1 500x5x5

Tableau 4.1. Les différents paramètres du réseau CNN proposé

Pour estimer les taux de reconnaissance sur les bases de test, nous avons évalué le rapport entre le nombre de caractères bien classés sur le nombre de caractères testés. Dans nos expériences, nous avons utilisé l'algorithme de descente de gradient stochastique de type ADAM, avec des mini-lots de taille 500. Le nombre d'époques qui correspond au nombre de fois que la base d'apprentissage est présentée au réseau est fixé à 4. Les poids synaptiques des réseaux CNN sont initialisés à des valeurs aléatoires en utilisant une distribution uniforme de moyenne nulle et de variance 0.01. Les biais sont initialisés à zéro et le pas d'apprentissage est fixé à 0.01. Dans le cas du classifieur SVM, nous avons utilisé le noyau linéaire.

Le tableau 4.2 montre les taux de reconnaissance obtenus par le classifieur MLP pour les différentes bases. Ces taux sont largement supérieurs à ceux obtenus par les descripteurs de la TOD, Gabor, Zernike, Kirsh, HOG, LBP et BOF (Tableau 3.7), ainsi que ceux trouvés par la méthode hybride TOD-HOG proposée au chapitre précédent (Tableau 3.8).

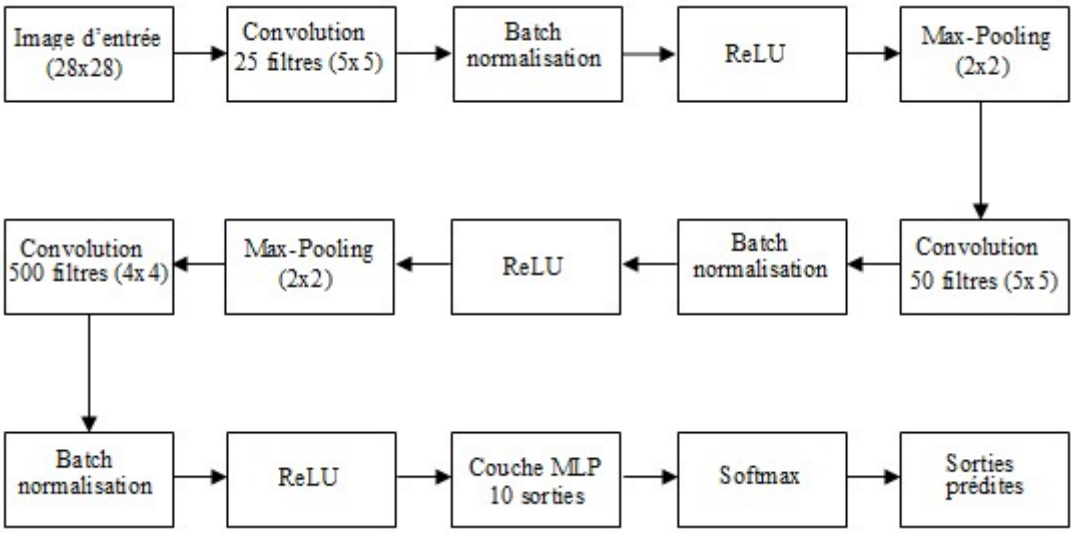


Figure 4.15. Réseau CNN avec une couche de classification MLP

Méthodes	USPS	MNIST	CVLSD	HVSN
CNN-MLP	96.61%	99.50%	95.81%	91.63%
CNN-SVM	**96.66%**	99.52%	95.90%	91.90%
TOD-CNN-MLP	96.56%	99.51%	96.14%	92.22%
TOD-CNN-SVM	96.56%	**99.59%**	**96.37%**	**92.44%**

Tableau 4.2. Taux de reconnaissance obtenus par les réseaux convolutifs proposés

4.6.2. CNN-SVM

Dans CNN-SVM, le classifieur MLP est remplacé par SVM comme le montre la figure 4.16. L'idée du réseau CNN-SVM est d'utilisé le réseau CNN afin de calculer par apprentissage les caractéristiques de chaque caractère et de substituer par la suite le classifieur MLP par SVM plus performant [184]. A cet effet, le réseau CNN est d'abord entrainé comme dans le cas du CNN-MLP. Ensuite, le vecteur de caractéristiques se trouvant à l'avant dernière couche de sortie du CNN est soumis au classifieur SVM. Ce vecteur représente le code CNN de chaque caractère de la base d'apprentissage.

Durant la phase de décision, le code CNN de chaque caractère de la base de test est soumis au SVM qui fournit en sortie le numéro de la classe.

Les taux de reconnaissance obtenus par CNN-SVM sur les quatre bases des chiffres manuscrits sont meilleurs que ceux délivrés par CNN-MLP (voir Tableau 4.2).

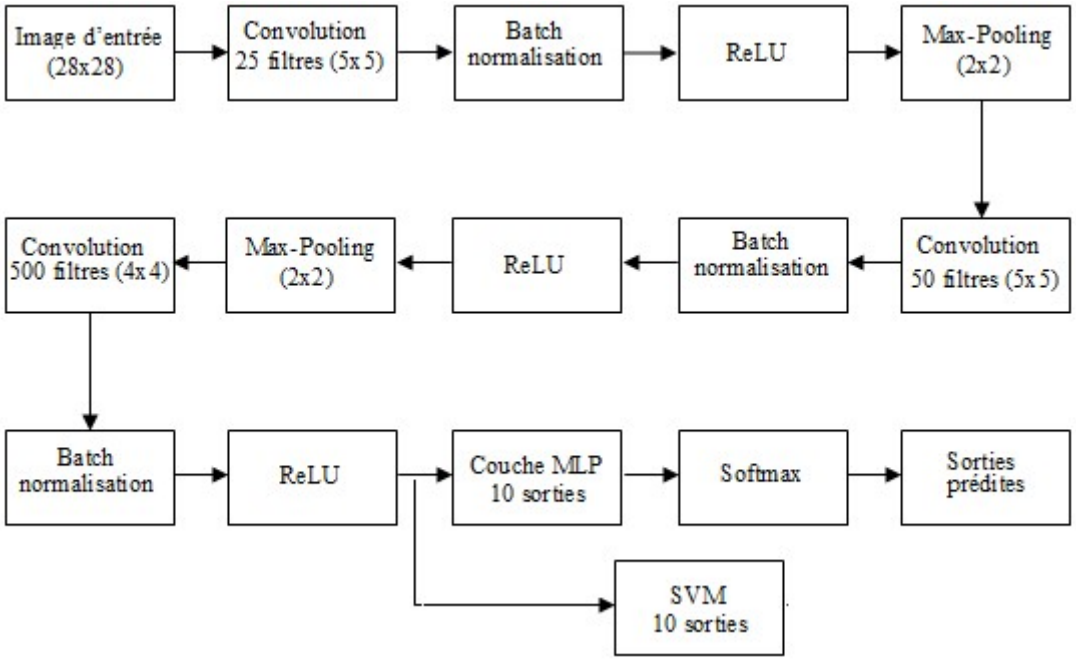


Figure 4.16. Réseau CNN combiné au classifieurSVM

4.6.3. TOD-CNN-SVM

Nous avons également développé un modèle hybride nommé TOD-CNN-SVM qui combine le réseau CNN-SVM (CNN-MLP) proposé précédemment avec la TOD. Dans ce cas, le réseau CNN, utilisé en tant qu'extracteur de caractéristiques, reçoit en entrée non pas une image en niveaux de gris, mais plutôt l'image d'approximation obtenue par application de la transformée en ondelettes sans décimation sur l'image originale. L'architecture du CNN est identique à celle du CNN-MLP et CNN-SVM (Figure 4.17). L'ondelette Symlet d'ordre 8 est utilisée pour décomposer l'image du caractère au premier niveau. Pour la phase de classification, le classifieur MLP ou SVM peut être employés. Les résultats de reconnaissance obtenus sur les quatre bases des chiffres manuscrits sont affichés dans le tableau 4.2.

Ces résultats montrent que l'introduction de la TOD a permis d'améliorer les performances du CNN. Le réseau TOD-CNN-SVM surclasse les réseaux CNN-MLP, CNN-SVM, TOD-CNN-MLP et ce, dans le cas des 3 bases MNIST, CVLSD, SVHN, à l'exception de USPS. Pour cette dernière, CNN-SVM reste le plus performant. Aussi, le classifieur SVM confirme sa supériorité vis-à-vis du classifieur MLP.

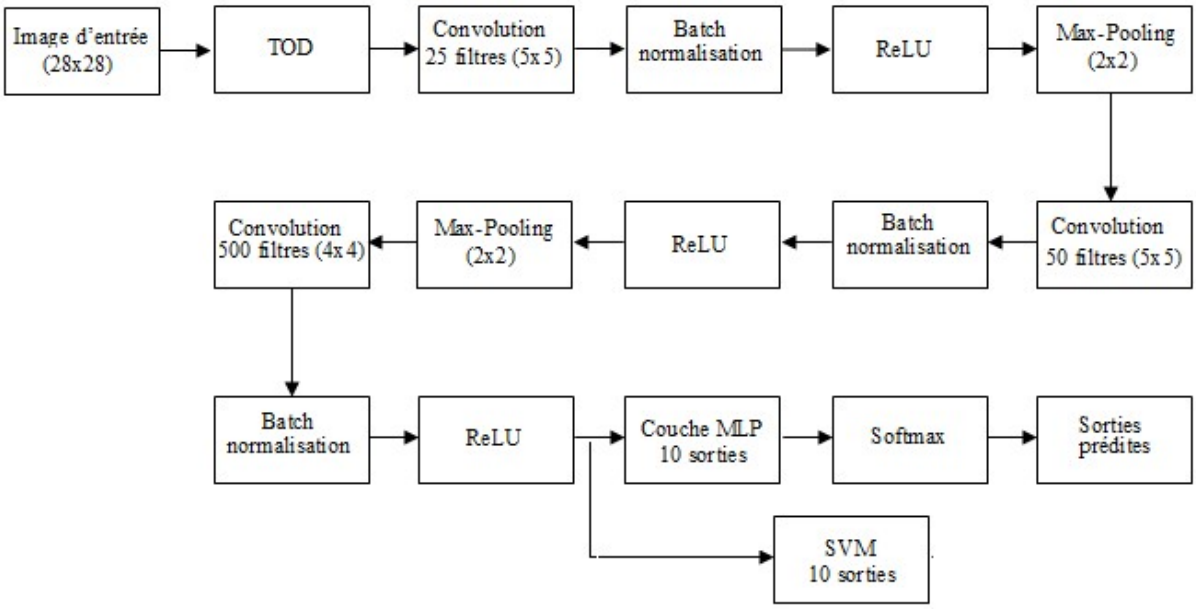


Figure 4.17. Réseau TOD-CNN-SVM proposé

4.6.4. Comparaison avec les méthodes de l'état de l'art

Pour valider les résultats obtenus par l'ensemble des réseaux convolutifs proposés, nous les avons confrontés à ceux trouvés dans l'état de l'art (voir Tableau 4.3). Ces résultats montrent que l'ensemble des réseaux proposés dépasse les réseaux concurrents. Cependant, TOD-CNN-SVM reste le plus performant. En effet, il surpasse les réseaux LeNet-5, ConvNet, et CNN [185] dans le cas de la base SVHN. Ces résultats indiquent également que les taux de reconnaissance obtenus par LeNet-5 et CNN [185] dans le cas de la base CVLSD sont nettement inférieurs au taux 96.37% trouvé par TOD-CNN-SVM. Sur la base MNIST, le taux obtenu par TOD-CNN-SVM, estimé à 99.59% est très encourageant vu qu'il dépasse ceux trouvés par les autres réseaux à l'exception du CNN-WAV4 qui combine 4 CNNs. Pour la base USPS, CNN-SVM offre le meilleur taux de 96.66%.

Méthodes	USPS	MNIST	CVLSD	SVHN
CNN-MLP	96.61%	99.50%	95.81%	91.63%
CNN-SVM	**96.66%**	99.52%	95.90%	91.90%
TOD-CNN-MLP	96.56%	99.51%	96.14%	92.22%
TOD-CNN-SVM	96.56%	**99.59%**	**96.37%**	**92.44%**
LeNet-1[1]	95.8%	98.3%	-	-
LeNet-4[1]	-	98.9%	-	-
LeNet-5[1]	-	99.05%	93.35%	86.28%
ConvNet/MS/L2 [182]	-	-	-	91.55%
CNN [185]	-	-	95.71%	92.22%
GCNN [181]		99,16%		
GCNNB [181]		99.32%		
CNN[183]	-	99.11%	-	-
CNN-WAV2 [183]	-	99.40%	-	-
CNN-WAV4[183]	-	**99.67%**	-	-

Tableau 4.3. Résultats de comparaison des réseaux CNNs proposés avec l'état de l'art

4.7. Conclusion

Dans ce chapitre, nous avons donné dans un premier temps un aperçu sur les réseaux de neurones classiques et les algorithmes d'apprentissage traditionnels les plus courants. Ensuite, nous avons défini les réseaux de neurones profonds ou Deep Learning. Plus particulièrement, nous nous sommes intéressés aux réseaux de neurones convolutifs, largement utilisés ces dernières années par la communauté de chercheurs. Dans cette optique, nous avons exposé en détails le principe de fonctionnement des CNNs, tout en décrivant chacune de ses parties, en commençant par la couche de convolution, de pooling et on termine par celle de la classification.

Ensuite, nous avons proposé 3 réseaux de neurones pour la reconnaissance des chiffres manuscrits. Le premier réseau CNN-MLP, possède une structure semblable à celle d'un CNN standard, mais avec des paramètres différents (nombre de couches de convolution, pooling, nombre de filtres dans chaque couche). Le second CNN-SVM, utilise à la place du MLP le classifieur SVM pour la discrimination des chiffres manuscrits. Dans le 3ème réseau, TOD-CNN-SVM (TOD-CNN-MLP), le même réseau CNN-SVM (CNN-MLP) reçoit en entrée l'image approximation obtenue par application de la TOD sans décimation à l'image originale.

Plusieurs tests ont été conduits sur les 4 bases des chiffres manuscrits pour évaluer les performances de l'ensemble des réseaux convolutifs proposés. Les résultats ont montré d'une part, que l'ensemble de ces CNN fournissent des taux de reconnaissance meilleurs par rapport aux techniques d'apprentissage automatique classiques. Cela signifie que les caractéristiques obtenues par apprentissage profond sont plus discriminantes que les autres types de caractéristiques. D'autre part, l'architecture des CNNs proposée est plus performante que celles des CNNs tirées de la littérature. On peut également mentionner la supériorité du classifieur SVM vis-à-vis du MLP. Enfin, ces résultats montrent également que l'introduction de la TOD a permis d'améliorer les performances du CNN. Ceci constitue une contribution majeure de notre travail.

Conclusion

Dans cette thèse, nous avons présenté nos contributions dans le cadre dela reconnaissance de chiffres manuscrits isolés. Elles s'appuient principalement sur la transformée en ondelettes discrète (TOD), les réseaux de neurones convolutifs (CNN) et les machines à vecteurs de support (SVM).

1. Synthèse des chapitres

Dans le chapitre 1, nous avons présenté les différents modules qui composent un système de reconnaissance des caractères manuscrits hors ligne ainsi que les principales difficultés rencontrées en reconnaissance de l'écriture manuscrite. La variabilité de l'écriture manuscrite ainsi que les différentes applications, fait que la reconnaissance des caractères demeure un domaine de recherche très vivace. Un système de reconnaissance des caractères se base sur les principes de l'intelligence artificielle, et plus exactement celui de l'apprentissage automatique. Il comporte plusieurs phases dont celles d'extraction des caractéristiques et de classification. La caractérisation influe considérablement sur les performances d'un système de reconnaissance. Ce qui nous a conduits dans le chapitre 2, à dresser un état de l'art sur les techniques d'extraction de caractéristiques en reconnaissance des caractères manuscrits.

La revue de littérature du domaine de reconnaissance des caractères manuscrits, permet de constater la grande diversité des techniques d'extraction de caractéristiques. Certaines méthodes sont basées sur des transformations globales telles que la transformée de Gabor, les moments de Zernike et la transformée en ondelettes (TOD), alors que d'autres sont fondées sur l'extraction des caractéristiques statistiques (locales), comme le zonage, la transformation de caractéristiques invariantes à l'échelle (SIFT), les caractéristiques robustes accélérées (SURF), les techniques de sacs de caractéristiques visuels (BOF), les histogrammes de gradients orientés (HOG) et les motifs binaires locaux (LBP). Parmi toute la panoplie de ces

méthodes, nous avons retenu la technique basée sur la transformée en ondelettes car elle permet d'extraire à la fois des caractéristiques globales et locales.

Nous avons ainsi consacré le 3ème chapitre à l'application de la TOD pour la reconnaissance des chiffres manuscrits. Trois de nos contributions ont été présentées.

La première est consacrée à l'étude de l'influence du type de l'ondelette sur les performances des caractéristiques dérivées de la TOD. A cet effet, différents types d'ondelettes ont été testées et évaluées en vue de ressortir celle qui correspond mieux à la description des formes des caractères utilisés. Outre, le choix de l'ondelette, la sélection du type de la sous-bande à utiliser pour la description des formes des caractères manuscrits à été abordée dans cette étude en analysant différentes combinaisons entre les différentes sous-bandes. Les expériences réalisées sur la base de données MNIST, montrent bien que l'information contenue dans la sous-bande image lissée est suffisamment pertinente pour discriminer correctement les caractères, et l'ondelette sym8 est la plus indiquée. Une étude comparative avec les méthodes multirésolution tirées de l'état de l'art, ont permis de valider nos résultats.

Dans la seconde contribution, nous avons mené une étude comparative entre plusieurs types de caractéristiques (TOD, Gabor, moments de Zernike, Kirsch, HOG, LBP, BOF) sur des bases de données contenant des chiffres manuscrits (USPS, MNIST, CVLSD et SVHN). Les résultats obtenus confirment, d'une part, que le type de l'ondelette influe sur les taux de reconnaissance et d'autre part, que les descripteurs HOG sont les plus performants. Compte tenu de ces résultats, nous avons proposé une technique qui combine la TOD et les descripteurs HOG. L'application de cette méthode sur les 4 bases de test, nous a permis d'améliorer les taux de reconnaissance.

A partir de la même étude, nous avons pu constater que le nombre de descripteurs extraits par la plupart des techniques de caractérisation est très élevé. Or, rien ne justifie que toutes ces caractéristiques sont pertinentes et non redondantes. C'est ainsi que nous avons proposé dans notre troisième contribution, une méthode de réduction et de sélection (ACP-SFS) des caractéristiques les plus pertinentes. Cette méthode utilise d'abord l'ACP pour produire un nombre réduit de nouvelles caractéristiques, puis applique la sélection séquentielle ascendante (SFS) pour choisir parmi ces nouvelles caractéristiques celles qui provoquent la plus faible erreur de classification. Cette méthode a été évaluée que sur un échantillon réduit de la base USPS avec les caractéristiques de la TOD. Les résultats obtenus sont assez encourageants, d'où l'intérêt de retravailler sur cette méthode et de l'évaluer sur d'autres bases de chiffres manuscrits plus complètes en utilisant d'autres types de caractéristiques.

Dans le 4ème chapitre, nous nous sommes intéressés à l'apprentissage profond (Deep Learning) et plus précisément aux réseaux de neurones convolutifs (CNN) ainsi que leurs applications à la reconnaissance des chiffres manuscrits isolés. Les modèles d'apprentissage profond et à leur tête les CNN ont l'avantage de générer automatiquement des caractéristiques par apprentissage et ne nécessitent donc pas le choix préalable d'une technique d'extraction de caractéristiques. L'architecture d'un CNN se compose d'une succession de couches de neurones qu'on peut diviser en deux parties. La première contient principalement des couches de convolution et de pooling et a pour but d'extraire un ensemble de caractéristiques. La deuxième partie, effectue une classification supervisée des données à partir de ces caractéristiques. Elle contient un ensemble de couches totalement connectées semblable au réseau perceptron multicouches (MLP). Nous avons proposé 3 CNNs pour la reconnaissance des chiffres manuscritsisolés.

Le premier nommé ''CNN-MLP'', possède une structure semblable à celle d'un CNN standard, mais avec des paramètres différents (nombre de couches de convolution, pooling, nombre de filtres dans chaque couche). Le second ''CNN-SVM'', utilise le classifieur SVM à la place du MLP. Dans le 3ème réseau, nommé ''TOD-CNN-SVM'' (TOD-CNN-MLP), l'image approximation obtenue par application de la TOD sans décimation est injectée à l'entrée du réseau.

Pour évaluer les performances de l'ensemble des réseaux convolutifs proposés, plusieurs tests ont été conduits sur les 4 bases de chiffres manuscrits. Les résultats montrent, d'une part, que l'ensemble de ces CNN fournissent des taux de reconnaissance meilleurs par rapport aux techniques d'apprentissage automatique (machine learning) qui associent l'extraction des caractéristiques par des techniques telles que TOD, HOG, LBP et TOD-HOG et le classifieur SVM. Cela signifie que les caractéristiques obtenues par apprentissage profond sont plus discriminantes que les autres types de caractéristiques. D'autre part, l'architecture des CNNs proposée est plus performante que celles des CNNs tirées de la littérature. On peut également mentionner la supériorité du classifieur SVM vis-à-vis du MLP. Enfin, ces résultats montrent également que l'introduction de la TOD a permis d'améliorer les performances du CNN. Ceci constitue une contribution majeure de notre travail.

2. Perspectives

Les perspectives ouvertes par ces travaux sont très nombreuses et peuvent se résumer en plusieurs points:

- Pour une meilleure représentation des caractères, la fusion descaractéristiques globales et locales vues dans le chapitre 2 et 3 peut apporter des informations complémentaires et amener à une caractérisation pertinente. La plus simple méthode de fusion consiste donc à concaténer ces vecteurs de caractéristiques.

- Comme nous l'avons évoqué au chapitre 3, la technique ACP-SFS proposée pour sélectionner les caractéristiques les plus pertinentes doit être évaluée sur plusieurs bases de données complètes et avec d'autres types de caractéristiques.

- D'autres méthodes de fusion telles que celle basée sur la combinaison de classifieurs SVMs ou CNNs pourrait améliorer la prise de décision. En effet, la combinaison de plusieurs décisions, permet éventuellement d'en cumuler les avantages. En outre, elle est considérée comme une excellente alternative à l'utilisation d'un unique classifieur.

- L'initialisation a une grande influence sur la convergence et sur la vitesse d'apprentissage. Si les poids sont trop faibles, le signal aura tendance à décroitre jusqu'à s'annulé et s'ils sont trop grands, le signal peut devenir trop grand pour être utilisable.
L'une des solutions envisageable est de faire un transfert d'apprentissage. Cela consiste à utiliser les poids d'un autre réseau de neurones déjà entrainé tel que la machine de Boltzmann profonde ou un auto-encodeur profond pour initialiser les poids d'un CNN.

- On s'est limité dans cette thèse qu'à la reconnaissance de chiffres manuscrits isolés qui sont préalablement pré-segmentés. Nous préconisons dans l'immédiat d'appliquer les techniques développées sur des caractères ou symboles manuscrits isolés. Une autre extension possible de ce travail est de développer une application réelle de reconnaissance d'une chaine de caractères manuscrits sur un Smartphone.

Annexe A

Transformée en Ondelettes

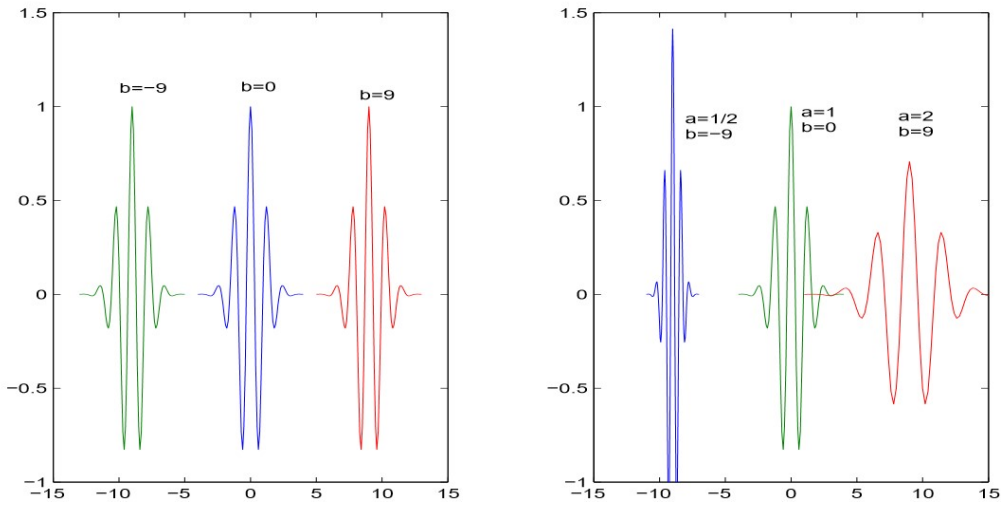
A.1. Définition

La transformée en ondelettes permet de décomposer une fonction d'énergie finie sur une base de fonctions élémentaires: les ondelettes. Cette base ou famille d'ondelettes $\{\psi_{a,b}\}$ est générée par translations et dilatations d'une fonction $\psi(x)$, appelée ondelette mère, qui s'écrit:

$$\psi_{a,b}(x) = \frac{1}{\sqrt{a}}\psi\left(\frac{x-b}{a}\right) \quad a \in \mathcal{R}^{+*}, b \in \mathcal{R} \qquad (A.1)$$

Les paramètres a et b sont respectivement les facteurs de dilatation (ou d'échelle) et de translation. La constante $\frac{1}{\sqrt{a}}$ est un facteur de normalisation.

La figure (A.1) présente un exemple de génération d'une base ou famille d'ondelettes $\{\psi_{a,b}\}$.



(a) (b)

Figure A.1. (a) Translation, (b) translation-dilatation d'une ondelette

Dans l'image (a), on voit deux ondelettes ($b = -9$, $b = 9$), obtenues par translation de l'ondelette mère ($b = 0$).

Sur l'image (b), les ondelettes de gauche et de droite de l'ondelette mère ($a = 1, b = 0$) ont subi respectivement une compression ($a = \frac{1}{2}, b = -9$) et une dilatation ($a = 2, b = 9$).

A.2. Transformation en ondelettes continue

La transformée en ondelettes continue (TOC) utilise des translations et des dilatations de la fonction ondelette mère durant tous l'intervalle du temps de manière continue. La TOC d'une fonction $I(x) \in \mathcal{L}^2(\mathcal{R})$ est une projection de celle-ci sur une famille d'ondelettes choisie. Autrement dit, elle consiste à mesurer sa similarité avec des bases d'ondelettes $\psi_{a,b}$. Son expression est donnée par:

$$WT_f(a,b) = C_{a,b} = \int_{-\infty}^{\infty} I(x)\, \psi_{a,b}\,(x)dx \qquad (A.2)$$

où $C_{a,b}$ représentent les coefficients d'ondelettes.

La reconstruction du signal $I(x)$ peut être effectué par application de la transformée en ondelettes inverse.

$$I(x) = \frac{1}{C_\psi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} C_{a,b}.\psi_{a,b}(x) \frac{da.db}{a^2} \qquad (A.3)$$

avec:

$$C_\psi = \int_{-\infty}^{+\infty} \frac{|\Psi(f)|^2}{|f|} df \ < \infty \qquad (A.4)$$

où $\Psi(f)$ est la transformée de Fourier de $\psi(x)$.

L'équation (A.4) représente la condition d'admissibilité pour avoir une transformée inverse unique. Si cette condition est respectée, on aura:

$$\int_{-\infty}^{+\infty} \psi(x)dx = \Psi(f)|_{f=0} = 0 \qquad (A.5)$$

Cependant, les applications de cette transformée sont très limitées. Cela est dû d'une part, à la redondance des coefficients d'ondelettes et d'autre part, au temps de calcul énorme requis. Pour une implémentation efficace, cette transformée doit être discrétisée.

A.3. Transformée en ondelettes discrète dyadique

La transformée en ondelettes discrètes (TOD) est conçue en discrétisant les paramètres d'échelle a et de translation b tels que: $a = a_0^j$ $(j \in Z)$ et $b = kb_0a_0^j$ $(a_0 > 1,\ b_0 > 0)$ [186]. Ainsi, la nouvelle famille d'ondelettes s'écrit:

$$\psi_{j,k}(x) = a_0^{-j/2}\psi\left(a_0^{-j}x - kb_0\right), \qquad (j,k) \in Z^2 \tag{A.6}$$

La transformée en ondelettes discrète de la fonction $I(x)$ est donnée par:

$$C_{j,k} = a_0^{-j/2} \int_{-\infty}^{+\infty} \psi\left(a_0^{-j}x - kb_0\right) I(x)dt \tag{A.7}$$

Les valeurs de $a_0 = 2$ et de $b_0 = 1$ sont choisies de telle sorte que la famille d'ondelettes discrète $\psi_{j,k}$ constitue une base orthonormée de $L^2(R)$ [133]. L'équation (A.7) devient:

$$\psi_{j,k}(x) = 2^{-j/2}.\psi\left(2^{-j}x - k\right) \tag{A.8}$$

Cette famille d'ondelettes $\left\{\psi_{j,k}\right\}_{j,k\in Z}$ constitue une base orthonormée de $\mathcal{L}^2(\mathcal{R})$.

De même, la reconstruction du signal $I(x)$ à partir des coefficients d'ondelettes peut s'écrire sous la forme:

$$I(x) = \sum_{j=-\infty}^{+\infty} \sum_{k=-\infty}^{+\infty} 2^{-j/2}.C_{j,k}.\psi\left(2^{-j}x - k\right) \tag{A.9}$$

A.4. Analyse de multirésolution

Le concept de l'analyse multirésolution consiste à représenter un signal comme étant une limite de ses approximations successives, où chaque approximation est une version lissée de la précédente. Ces approximations successives sont présentées à différentes résolutions, d'où le nom de multirésolution. Ainsi, lorsque la résolution augmente, les images successives approximent le signal de mieux en mieux, par contre, lorsque la résolution diminue, la quantité d'informations contenue dans les images diminue aussi jusqu'à s'annuler. Lors du passage d'une résolution à une autre, la différence entre deux approximations successives est représentée par les coefficients d'ondelettes.

L'analyse multirésolution, développée par Mallat [117], permet la représentation d'un signal par des approximations dans une suite d'échelles 2^{-j}. Ces échelles sont définies par des espaces imbriqués.

A.4.1. Espace des approximations

Les espaces d'approximations obtenus à des échelles 2^{-j} successives forment un ensemble d'espaces emboités $\{V_j\}_{j\in Z}$ de $\mathcal{L}^2(\mathcal{R})$, vérifiant les propriétés suivantes:

$$\forall\, j \in Z, \qquad V_j \subset V_{j+1} \tag{A.10}$$

$$\bigcap_{j\in Z} = V_j = \{0\} \tag{A.11}$$

$$\bigcup_{j\in Z} V_j = \mathcal{L}^2(\mathcal{R}) \tag{A.12}$$

$$\forall\, I \in L^2(\mathcal{R}^D), \forall\, j \in Z, \;\; I(x) \in V_j \;\leftrightarrow I(2x) \in V_{j+1} \tag{A.13}$$

$$\forall\, I \in L^2(\mathcal{R}^D), \forall (j,k) \in Z^2, \qquad I(x) \in V_0 \;\leftrightarrow I(x-k) \in V_0 \tag{A.14}$$

Il existe une fonction d'échelle ϕ qui par dilatation et translation engendre une base orthonormée de V_j. Cette fonction est notée par $\phi(x) \in L^2(\mathcal{R})$ et les fonctions de bases sont construites suivant la relation:

$$\phi_{j,k}(x) = 2^{-j/2}\phi\left(2^{-j}x - k\right) \qquad (j,k) \in Z \tag{A.15}$$

- La propriété (A.10) traduit le fait que toute information accessible à l'échelle 2^{-j} (projection sur V_j), l'est aussi à l'échelle $2^{-(j+1)}$. L'information contenue dans le signal est dégradée lorsque j décroit.
- (A.11) implique que plus on descend en résolution, plus on perd tous les détails de I.
- (A.12) entraine une convergence des approximations vers le signal I.
- (A.13) garantit que l'on a une approximation à une résolution plus grossière $(j+1)$.
- La propriété (A.14) signifie qu'un signal translaté est invariant par rapport à la résolution.

La projection orthogonale de la fonction $I \in L^2(\mathcal{R})$ sur l'espace V_j permet de calculer les coefficients d'approximation $A_j I$.

$$\forall\, I \in \mathcal{L}^2 : A_j I = proj_{V_j}(I) = \sum_{k=-\infty}^{+\infty} \langle I(x), \phi_{j,k} \rangle\, \phi_{j,k} \qquad (A.16)$$

A.4.2. Espace de détails

On définit pour chaque espace V_j son complément orthogonal W_j, tel que:

$$V_{j+1} = V_j \oplus W_j,\ j \in Z \text{ et } W_j \perp V_j \qquad (A.17)$$

$$\forall j, j \neq k, W_j \perp W_k \qquad (A.18)$$

Les sous-espaces W_j ne forment pas une famille d'espaces emboités, mais les propriétés d'échelle et d'invariance par translation sont conservées.

L'équation (A.17) signifie que l'information contenue dans la projection d'un signal sur V_{j+1} est équivalente à celle contenue dans les projections sur V_j et W_j.

La figure (A.2) schématise cette décomposition, les sous espaces sont représentés symboliquement par des rectangles.

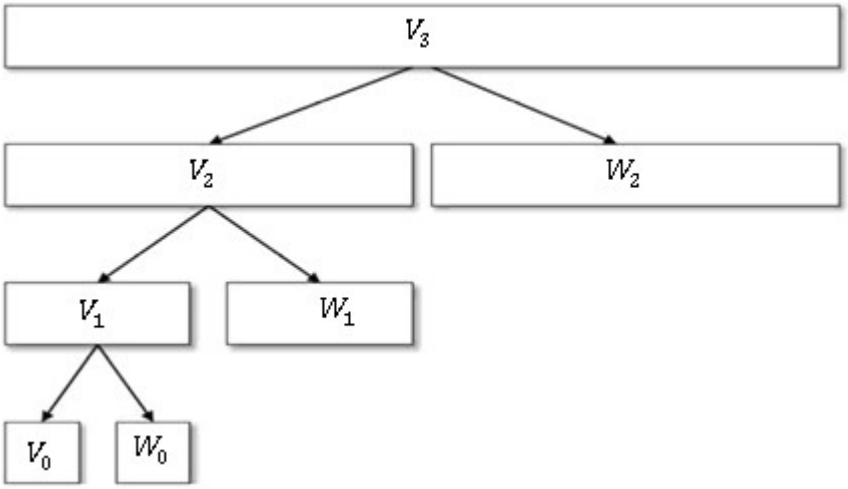


Figure A.2. Principe de l'analyse multirésolution

Lors du passage de l'espace d'approximation V_{j+1} à un espace V_j, des détails du signal $I(x)$ sont perdus. Les espaces qui caractérisent ces détails sont appelés espaces d'ondelettes W_j qui sont engendrés par une fonction $\psi(x)$ appelée ondelette mère.

Ainsi, les coefficients de détails $D_j I$ peuvent être obtenus par la projection orthogonale de la fonction $I(x) \in L^2(R)$ sur l'espace W_j.

$$\forall\, I \in \mathcal{L}^2(\mathcal{R}): D_j I = proj_{W_j}(I) = \sum_{k=-\infty}^{+\infty} \langle I(x), \psi_{j,k} \rangle \psi_{j,k} \qquad (A.19)$$

A.4.2. Décomposition par banc de filtres

A.4.2.1. Cas de signaux monodimensionnels

En pratique, la théorie des ondelettes est mise en place grâce à des bancs de filtres qui permettent de décomposer un signal en plusieurs bandes de fréquence (basses fréquences et hautes fréquences).

Les approximations et les détails sont obtenus respectivement par un filtrage passe-bas et passe-haut, suivis d'un échantillonnage uniforme à la période 2^j.

Les coefficients d'approximations $A_j(n)$ et de détails $D_j(n)$ sont obtenus par application d'un filtre passe-bas h et un filtre passe-haut g suivi d'un sous échantillonnage par 2 (2↓). La figure (A.3) représente cette décomposition.

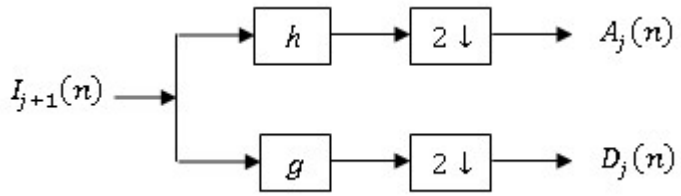


Figure A.3. Décomposition à un seul niveau

Ces coefficients sont donnés par:

$$A_j(n) = I_{j+1}(n) * h(2n) = \sum_{l=-\infty}^{+\infty} I_{j+1}(l).h(l-2n) \qquad (A.20)$$

$$D_j(n) = I_{j+1}(n) * g(2n) = \sum_{l=-\infty}^{+\infty} I_{j+1}(n).g(l-2n) \qquad (A.21)$$

où $*$ est le produit de convolution. La relation entre les deux filtres est donnée pa :

$$h(L-1-n) = (-1)^n g(n) \qquad n = 0, 1, \dots, L-1 \qquad (A.22)$$

où L est la taille du filtre.

L'algorithme pyramidal [117] permet de répéter le même processus de décomposition sur les coefficients d'approximation $A_j(n)$ obtenus à la résolution j.

A.4.2.2. Cas de signaux bidimensionnels

D'après S.Mallat [187], les résultats établis pour des signaux monodimensionnels peuvent être étendus dans le cas d'un espace à deux dimensions.

Soit ϕ une fonction d'échelle et ψ l'ondelette correspondante. On définit trois ondelettes $\psi_1(x,y)$, $\psi_2(x,y)$, $\psi_3(x,y)$, telle que:

fonction d'échelle $\phi_2(x,y) = \phi(x)\phi(y)$

fonctions d'ondelettes $\psi_1(x,y) = \phi(x)\psi(y)$

$\psi_2(x,y) = \psi(x)\phi(y)$

$\psi_3(x,y) = \psi(x)\psi(y)$

La transformée en ondelettes à deux dimensions consiste à appliquer la transformée en ondelettes monodimensionnelle en chaque élément de la ligne, ensuite en chaque élément de la colonne. Cette décomposition permet de générer quatre sous-bandes: une sous-bande d'approximation I_{LL} et trois sous-bandes de détails: I_{LH}, I_{HL} et I_{HH}, correspondant respectivement aux détails horizontaux, verticaux et diagonaux. Ceci, se traduit par application d'un filtre passe-bas $l(x)$ et d'un filtre passe-haut $g(x)$ de façon séparable sur les lignes et sur les colonnes. Une représentation pyramidale peut être obtenue à la suite d'une décomposition successive de la sous bande d'approximation. La figure (A.4) montre la décomposition successive par la transformée en ondelettes discrète d'une image quelconque jusqu'à 2 niveaux de résolution. La figure (A.5) présente la décomposition par la transformée en ondelettes de l'image hehagone en deux niveaux de résolution.

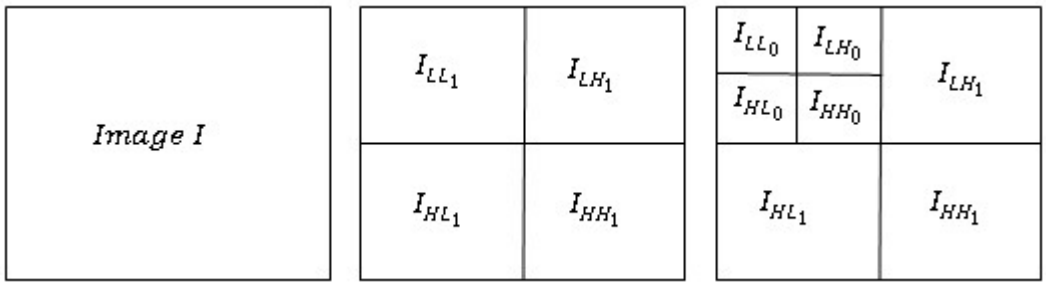


Figure A.4. Décomposition successive par la transformée en ondelettes discrète (jusqu'à 2 niveaux)

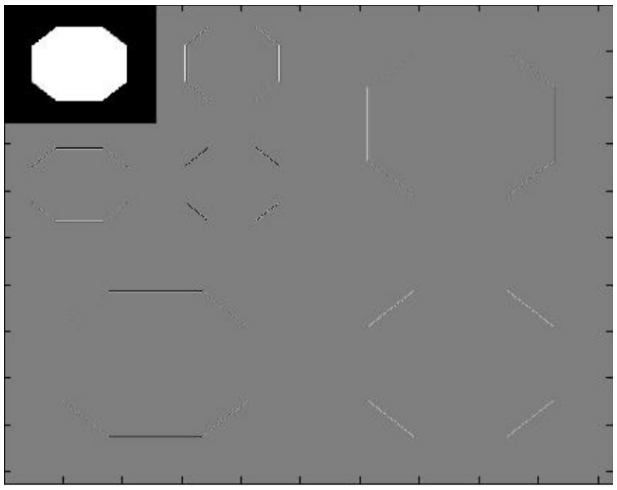


Figure A.5. Décomposition en ondelettes d'un hexagone
en 2 niveaux de résolution

A.5. Famille d'ondelettes

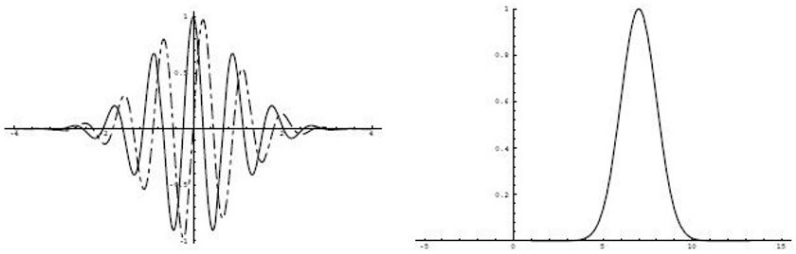
En pratique, le choix de l'ondelette la plus adaptée à une application donnée est crucial. La solution à cette problématique consiste à étudier leurs influences sur l'application envisagée et ce, en fonction des objectifs à atteindre. Cependant, certaines propriétés telles que la régularité, la symétrie, le support, l'orthogonalité, le nombre de moments nuls peuvent aider à faire une présélection.

A.5.1. Ondelettes continues

A.5.1.1. Ondelette de Morlet

L'ondelette de Morlet est une ondelette complexe qui a un grand intérêt dans l'′etude des signaux sismiques, puisque sa forme ressemble beaucoup à celle de l'ondelette sismique (Figure A.6). Cette ondelette est inspirée du signal élémentaire de Gabor. Elle est obtenue par modulation d'une gaussienne :

$$\psi(x) = (\pi x_0)^{-\frac{1}{2}} exp\left[-\frac{1}{2}\left(\frac{x}{x_0}\right)^2 + 2j\pi u_0 x\right] \qquad (A.23)$$



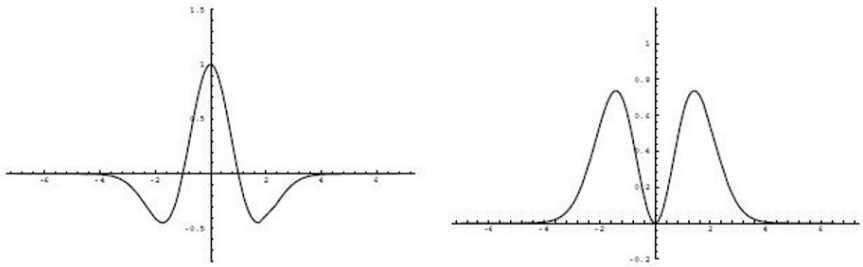
(a) Partie réelle (trait plein) et partie imaginaire (pointillés) (b) Transformée de Fourier

Figure A.6. Ondelette de Morlet et le module de sa transformée de Fourier

A.5.1.2. Chapeau mexicain

Le chapeau mexicain est une ondelette réelle qui doit son nom à sa forme, est construite à partir de la dérivée seconde de la gaussienne (Figure A.7). En effet, bien qu'une gaussienne ne soit pas une ondelette, toutes ses dérivées le sont. Son expression est donnée par:

$$\psi(x)(1-x^2)exp\left(-\frac{1}{2}x^2\right) \qquad (A.24)$$



(a) Ondelette (b) Sa transformée de Fourier

Figure A.7. Ondelette Chapeau Mexicain et module de sa transformée de Fourier

Cette ondelette est symétrique, ce qui permet de ne pas introduire de déphasage dans la transformée en ondelettes, contrairement aux ondelettes non symétriques.

A.5.2. Ondelettes orthonormales

A.5.2.1. Ondelette de Haar

Historiquement, la première base d'ondelettes orthonormale est celle de Haar (Figure A.8), donnée par l'expression:

$$\psi(x) = \begin{cases} 1, & 0 \leq x < \frac{1}{2} \\ -1, & \frac{1}{2} \leq x < 1 \\ 0 & sinon \end{cases} \qquad (A.25)$$

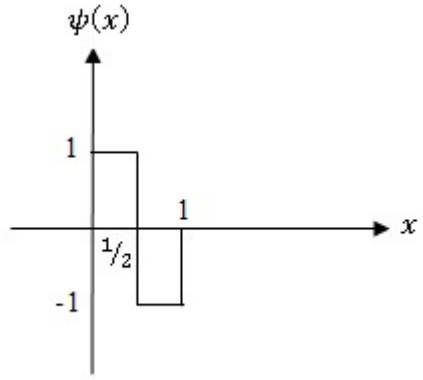


Figure A.8. Ondelette de Haar

Cette ondelette très simple et même facile à implémenter est à support compact. En pratique, cette ondelette est peu utilisée du fait de sa forme peu adaptée à des signaux réguliers.

A.5.2.2. Ondelettes de Daubechies

Les ondelettes de Daubechies [188] ont été construites de telle sorte qu'elles aient le support le plus petit pour un nombre de moments nuls m donné par:

$$\int x^k \psi(x) dx = 0, \qquad \forall\, k \in [0, m-1] \qquad (A.26)$$

La fonction d'échelle est d'ordre m, et son support de taille $d = 2m$. Ces fonctions de base sont irrégulières et très fortement asymétriques (Figure A.9).

Les ondelettes à support compact de Daubechies sont particulièrement intéressantes dans la mesure où on peut choisir la régularité voulue en imposant un certain nombre de moments nuls. La régularité augmente avec le nombre de moments nuls. Les ondelettes à support compact ne sont pas adaptées pour la détection des frontières puisqu'elles ne sont pas symétriques.

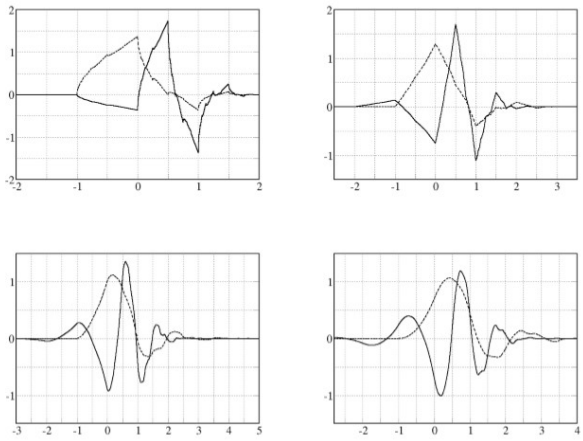


Figure A.9. Ondelettes (trait plein) et fonctions d'échelle (pointillés)
de Daubechies - ordres 2, 3, 4 et 5.

A.5.2.3. Les Symlets

Daubechies a construit des ondelettes à support compact les plus symétriques possibles (Figure A.10). Les symlets ont le même nombre de moments nuls que les ondelettes de Daubechies. Pour un support donné: on a d = 2m, et le nombre d'éléments non nuls du filtre est 2m.

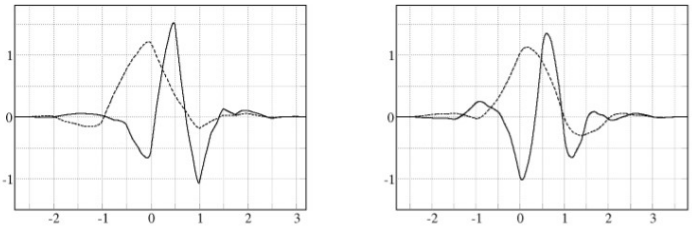


Figure A.10. Ondelettes (trait plein) et fonctions d'échelle (pointillés) de symlets
A gauche, ordre 4, à droite, ordre 5.

A.5.2.4. Les Coifflets

Les coifflets sont des ondelettes à m moments nuls et ayant une taille de support minimale (Figure A.11), dont la fonction d'échelle ϕ vérifie les propriétés:

$$\int \phi(x)dx = 1 \text{ et } \int x^k \phi(x)dx = 0, \quad 0 \leq k < m \qquad (A.27)$$

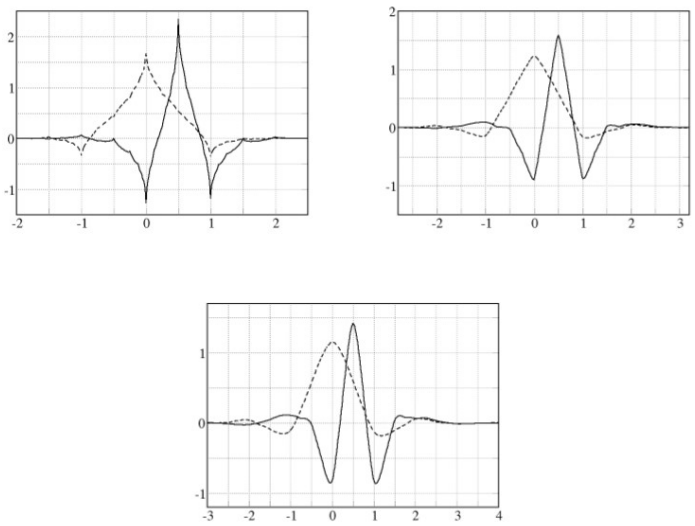


Figure A.11. Ondelettes (trait plein) et fonctions d'échelle (pointillés)
de Coiflets d'ordres 2,4 et 6.

Annexe B

Machines à Vecteurs de Support SVM

B.1. Séparateurs à vaste marge

La technique des machines à vecteurs de support (SVM), connue en analyse discriminante comme une technique de classification très performante, est principalement issue des travaux de Vapnik [126]. Contrairement aux méthodes traditionnelles qui cherchent à minimiser l'erreur sur l'ensemble d'apprentissage, Vapnik propose de la remplacer par la minimisation du risque structurel.

Les fondements théoriques des machines à vecteurs de support reposent sur les notions d'hyperplan optimal et des fonctions de noyaux pour séparer les classes.

B.4.1. Détermination d'une frontière linéaire (SVM linéaire)

B.4.1.1. Hyperplan à mage optimal

Considérons le problème de recherche de frontière linéaire dans le cas de deux classes. Nous disposons ainsi d'un ensemble de paires de données étiquetées $\{(x_i, y_i)\}_{i=1,\dots,N}$. y_i est l'étiquette de la donnée x_i. Elle vaut +1 pour les données positives et -1 pour les données négatives. Dans le cas où les classes sont séparables par un hyperplan, l'équation de celui-ci peut s'écrire:

$$f(x) = w.x + b = 0 \qquad (B.1)$$

Pour décider à quelle classe une donnée x appartient, il suffit de prendre le signe de la fonction de décision $y = sgn(f(x))$ et à classer la donnée x dans la première classe, caractérisée par l'étiquette $y = 1$, si $f(x) > 0$, ou dans la seconde classe caractérisée par l'étiquette $y = -1$, si $f(x) < 0$. Autrement dit, il existe au moins un hyperplan qui vérifie:

$$f(x_i).y_i > 0, (i = 1, \dots, N) \qquad (B.2)$$

La recherche de l'hyperplan optimal (Figure B.1) consiste à déterminer les valeurs des paramètres w et b qui vérifient la contrainte (B.2), et qui maximisent la marge de séparation (voir figure B.2). Cette marge correspond à la distance minimale entre les données d'apprentissage et leurs projections sur cet hyperplan. La marge de séparation est donnée par l'expression: $\frac{2}{\|w\|}$

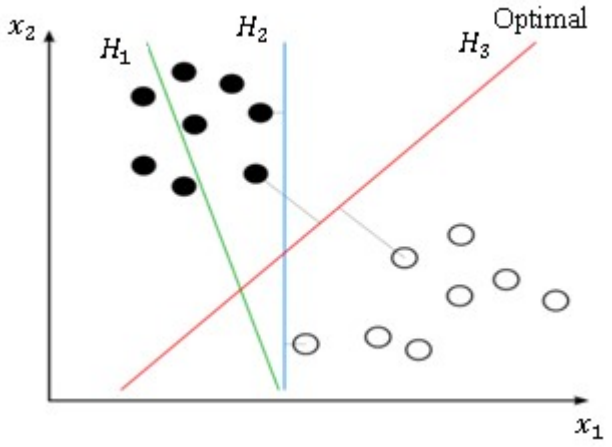


Figure B.1. Recherche d'un hyperplan optimal

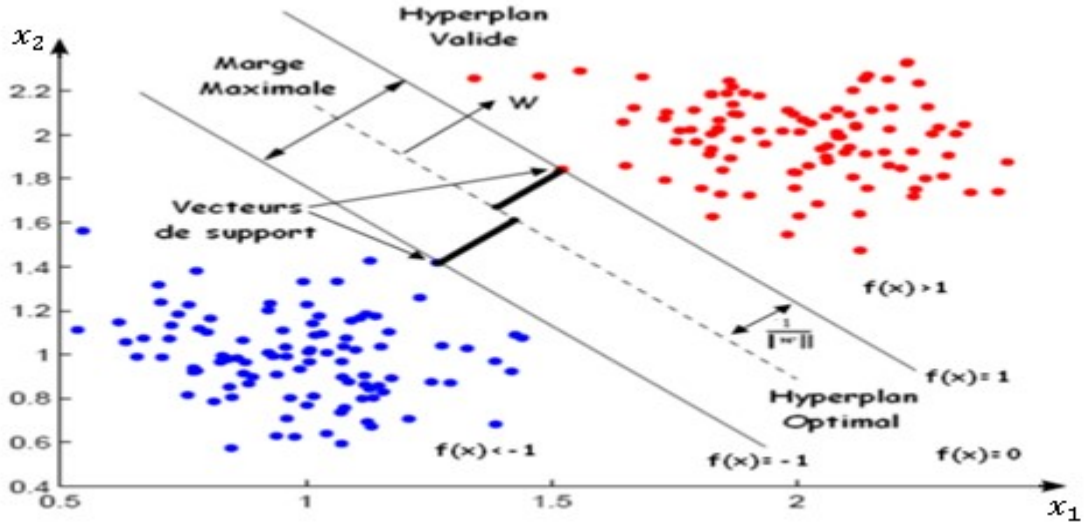


Figure B.2. Relation entre marge, vecteurs de support et hyperplan optimal

Déterminer l'hyperplan optimal revient à résoudre un problème d'optimisation, qui dans sa formulation primale consiste à minimiser:

$$J(w) = \frac{1}{2}\|w\|^2 \qquad (B.3)$$

sous la contrainte:

$$y_i(w.x_i + b) \geq 1; \quad (i = 1, \dots, N) \qquad (B.4)$$

Pour résoudre ce problème d'optimisation sous contraintes, on fait appel aux multiplicateurs de Lagrange, tels que:

$$L(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{N} \alpha_i (y_i(w.x_i + b) - 1) \qquad (B.5)$$

Les variables α_i correspondent aux facteurs de Lagrange des contraintes définies dans l'équation (B.4).

La minimisation de la fonction $L(w, b, \alpha)$ par rapport à w et b est effectuée par:

$$\frac{\partial}{\partial w} L\,(w, b, \alpha) = w - \sum_{i=1}^{N} \alpha_i y_i x_i = 0 \qquad (B.6)$$

et:

$$\frac{\partial}{\partial b} L\,(w, b, \alpha) = -\sum_{i=1}^{N} \alpha_i y_i = 0 \qquad (B.7)$$

La résolution des équations (B.6) et (B.7) donne:

$$w = \sum_{i=1}^{N} \alpha_i y_i x_i \qquad (B.8)$$

et

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \qquad (B.9)$$

A partir de ces équations, on peut réécrire le Lagrangien minimal uniquement en fonction des variables duales α_i:

L'équation (B.5) devient:

$$L(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j x_i x_j \qquad (B.10)$$

Ainsi, minimiser l'équation (B.5) revient à résoudre le problème dual qui consiste à maximiser $L(\alpha)$ avec les contraintes:

$$\alpha_i \geq 0, \quad i = 1, \dots, N \qquad (B.11)$$

et:

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \qquad (B.12)$$

La solution de ce problème d'optimisation sera un vecteur $\alpha^* = (\alpha_1^*, \alpha_2^*, . . ., \alpha_N^*)$.

Notons que seuls les α_i^* correspondants aux données se trouvant sur les hyperplans canoniques sont non nuls. Ces données sont appelés vecteurs de support (SV).

La décision après apprentissage est donnée par la fonction de décision $f(x)$.

$$f(x) = sgn\left(\sum_{i=1}^{N} \alpha_i y_i (x_i x) + b\right) \qquad (B.13)$$

B.4.1.2. Hyperplan à marge molle

Dans la plupart des problèmes réels, les données ne sont pas tout à fait linéairement séparables, à cause des données aberrantes (Figure B.3). Cette contrainte peut conduire à l'obtention d'un hyperplan qui n'est pas optimal.

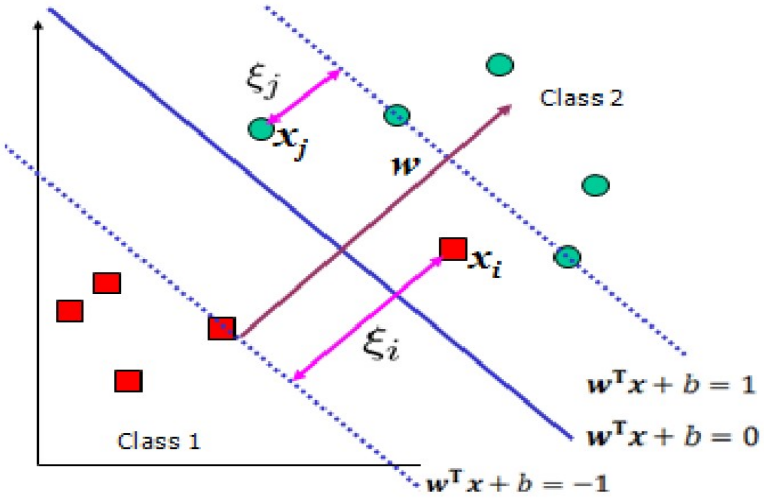


Figure B.3. Variables d'ajustement pour des données non linéairement séparables

Pour prendre en compte la présence de ces erreurs de classification, l'idée est d'ajouter les variables d'ajustement $\xi_i \geq 0$ qui permettent à quelques exemples d'être mal classifiées. Ainsi, l'équation (B.3) devient:

$$min \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{N} \xi_i \qquad (B.14)$$

où C est une constante positive, dite de régularisation.

La formulation duale du problème est similaire à celle du cas linéairement séparable, sauf que les multiplicateurs de Lagrange deviennent bornés par C.

$$max \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j x_i x_j \qquad (B.15)$$

sous les contraintes:

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \qquad (B.16)$$

et

$$0 \leq \alpha_i \leq C \qquad i = 1, \ldots, N \qquad (B.17)$$

Notons enfin que les données qui violent les contraintes de séparation citées ci-dessus ont des multiplicateurs de Lagrange différents de zéro ($\alpha_i \neq 0$) et sont donc sélectionnées également comme vecteurs de support.

B.4.2. Détermination d'une frontière non linéaire (SVM non linéaire)

Dans le cas de problèmes de classification réels, la frontière optimale est souvent non linéaire (Figure B.4). La prise en compte de non linéarités dans le modèle SVM s'effectue par l'introduction de noyaux non linéaires.

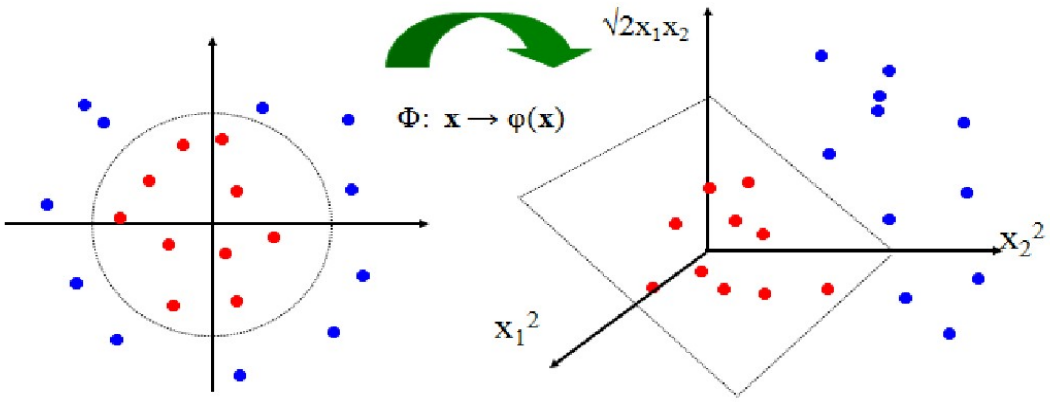


Figure B.4. Frontière de décision non linéaire

Le principe consiste à projeter les données d'apprentissage x_i dans un espace de plus grande dimension, en vue de chercher dans ce nouvel espace un hyperplan de décision optimal. En pratique, ce passage est effectué grâce à une la transformation $\phi(.)$, définie par un noyau de kernel K:

$$\phi(x).\phi(x_i) = K(x, x_i) \qquad (B.18)$$

Le critère d'optimalité à maximiser dans le cas des SVM est la marge, c'est-à-dire la distance entre l'hyperplan et le point $\phi(x_i)$ le plus proche de l'ensemble d'apprentissage. Les α_i^* permettant d'optimiser ce critère s'obtiennent en résolvant le problème suivant:

$$\max_{\alpha_i} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \qquad (B.19)$$

sous les contraintes:

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \qquad (B.20)$$

et:

$$0 \leq \alpha_i \leq C \qquad i = 1, \dots, N$$

Où C est un coefficient pénalisant les données se trouvant dans la marge et permettant d'effectuer un compromis entre le nombre de ces derniers et la largeur de la marge.
La valeur de b peut être calculée par [129]:

$$b = \frac{1}{N_s} \left(y_i - \sum_{j=1}^{N} y_i \alpha_j K(x_i, x_j) \right) \qquad (B.21)$$

Où N_s représente le nombre de vecteurs de support.

B.4.2.1. Quelques exemples de noyaux

Parmi les différentes formes de noyaux les plus usuels, on trouve:

- Le noyau linéaire: $K(x, y) = x.y$.
- Le noyau polynomial: $K(x, y) = (x.y + 1)^p$, où p est le degré du noyau polynomial.
- Le noyau gaussien: $K(x, y) = exp\left(-\frac{\|x-y\|^2}{\sigma^2}\right)$, où σ est un réel positif qui représente la largeur du noyau.

Bibliographie

[1] Y. Lecun et al., Gradient-based learning applied to document recognition. Proc. of IEEE, vol.86, pp.2278-2324, 1998.

[2] M. Hanmandlu et al. Fuzzy based approach to the recognition of multi-fonts numerals. Proc. of 2nd National Conf. on Document Analysis and Recognition (NCDAR), pp.118-126, 2003.

[3] R. Plamondon, and S.N. Srihari. On-line and off-line handwriting recognition: A comprehensive survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.22, no.1, pp.63-84, 2000.

[4] T. Starner et al. On-line cursive handwriting recognition using speech recognition methods. In Proc. of the International Conference on Acoustics, Speech and Signal Processing,vol.5, pp.125-128, 1994.

[5] M.E. Morita. Automatic recognition of handwritten dates on Brazilian bank check. PHD Thesis, 2003.

[6] D.C. Ciresan et al. Flexible, high performance convolutional neural networks for image classification. Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, 2011.

[7] A. Nosary et al. Reconnaissance de textes manuscrits par adaptation au scripteur. In Colloque International Francophone sur l'Ecrit et le Document, CIFED'2002, pp.365-374, 2002.

[8] C. C. Tappert et al. The state of the art in on-line handwriting recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.12, no.8, 1990.

[9] G. Kim and V. Govindaraju. Banck check recognition using cross validation between legal and courtesy amount. Automatic Bank Check Processing. World Scientific, pp.195-212, 1997.

[10] Site web: (www.axl.cefan.ulaval.ca/langues/1 div_recens.htm).

[11] R. Hussain et al. A comprehensive survey of handwritten document benchmarks: structure, usage and evaluation. EURASIP Journal on Image and Video Processing, vol.46, 2015.

[12] L. Likforman-Sulem. Apport du traitement des images à la numérisation des documents anciens. Document Numerique, vol.7, no.3-4, pp.13-26, 2003.

[13] Y. Zhang and W.U. Lenan. A fast document image denoising method based on packed binary format and source word accumulation. Journal of Convergence Information Technology, vol.6, no.2, pp.131-137, 2011.

[14] L. O. Gorman and R. Kasturi. Document image analysis. IEEE Computer Society Press, 1995.

[15] G. Leedham et al. Separating text and background in degraded document images: A comparison of global thresholding techniques for multi-stage thresholding. Proceedings of the Eighth International Workshop on Frontiers in Handwriting Recognition (IWFHR'02), 2002.

[16] N. Otsu. A threshold selection method from grey-level histograms. IEEE Transactions on Systems, Man and Cybernetics, vol.9, no.1, pp.62-66, 1979.

[17] R. Firdousi and S. Parveen. Local thresholding techniques in image binarization. International Journal of Engineering and Computer Science, vol.3, Issue 3, pp.4062-4065, 2014.
[18] I. K. Kim et al. Document image binarization based on topographic analysis using a water flow model. Pattern Recognition, vol.35, pp.265-277, 2002.
[19] T.K. Gill. Document image binarization techniques: A review. International Journal of Computer Applications, vol.98, no.12, pp.0975-8887, 2014.
[20] O. D. Trier and T. Taxt. Evaluation of binarization methods for document images. On Pattern Analysis and Machine Intelligence, vol.11, no.12, pp.312-314, 1995.
[21] M. Cheriet et al. Character recognition systems. A guide for students and practioners. Published by John Wiley & Sons, Inc., 2007
[22] S. B. Rezaei et al. Skew detection of scanned document images. Proc. of the International Multi Conference of Engineers and Computer Scientists (IMECS), vol.1, 2013.
[23] S. Rezaei et al. Adaptive Document Image Skew Estimation. Proc. of the International Multi Conference of Engineers and Computer Scientists (IMECS), vol.1, 2017.
[24] F. Zeeuw. Slant correction using histograms. Bachelor's Thesis, 2006.
[25] C. Sun and D. Si. Skew and slant correction for document images using gradient direction. 4th International Conf. on Document Analysis and Recognition, pp.142-146, 1997.
[26] J.D. Gupta and B. Chanda. Novel methods for slope and slant correction of off-line handwritten text word. Third International Conference on Emerging Applications of Information Technology (EAIT), 2012.
[27] D. Arrivalt. Apport des graphes dans la Reconnaissance non-contrainte de caractères manuscrits anciens. Thèse de Doctorat, 2002.
[28] M. Couprie. Note on fifteen 2d parallel thinning algorithms. Technical Report, Institut Gaspard-Monge, Unité Mixte de Recherche CNRS-UMLV-ESIEE, 2006.
[29] D. Yu and H. Yan. Reconstruction of broken handwritten digits based on structural morphological features. Pattern Recognition, vol.34, pp.235-254, 2001.
[30] W.P. Chois et al. Extraction of the euclidean skeleton based on a connectivity criterion. Pattern Recognition, vol.36, pp.721-729, 2003.
[31] S.Bag and G. Harit. A medial axis based thinning strategy and structural feature extraction of character images. Proc. of IEEE, 17th International Conference on Image Processing, 2010.
[32] N. Arica. An off line character recognition system for free style handwriting. Thèse pour l'obtention du diplôme Master en science, 1998.
[33] A. Kaur et al. Study of various character segmentation techniques for handwritten off line cursive words: A review. International Journal of Advances in Science Engineering and Technology, ISSN: 2321-9009, vol.3, Issue-3, 2015.
[34] C. Patel et al. A Review of character segmentation methods. International Journal of Current Engineering and Technology, ISSN 2277- 4106, vol.3, no.5, 2013.
[35] B. Singh et al. Parallel implementation of Devnagari text line and word segmentation approach on GPU. International Journal of Computer Applications, vol.24, no.9, pp.7-14, 2011.
[36] N. Dave. Segmentation methods for handwritten character recognition. International Journal of Signal Processing, Image Processing and Pattern Recognition, vol.8, no.4, pp.155-164, 2015.
[37] B. Gosselin . Application de réseaux de neurones artificiels à la reconnaissance automatique de caractères manuscrits, Thèse de Doctorat, Faculté Polytechnique de Mons, 1996.

[38] G.A. Farulla et al. A fuzzy approach to segment touching characters. Expert Systems with Applications, 2017.
[39] A.A. Shinde and D.G. Chougule, Text pre-processing and text segmentation for OCR. IJCSET, vol.2, Issue 1, pp.810-812, 2012.
[40] N. Ouwayed. Segmentation en lignes de documents anciens : Application aux documents arabes. Thèse de Doctorat, Nancy 2, 2010.
[41] A. Bennasri et al. Extraction des lignes d'un texte manuscrit arabe. Vision Interface'99, pp.42- 48, 1999.
[42] A. Nicolaou, B. Gatos. Handwritten text line segmentation by shredding text into its lines. International Conference on Document Analysis and Recognition, pp. 626-630, 2009.
[43] V. Shapiro et al. Handwritten document image segmentation and analysis. Pattern Recognition Letters, vol.14, no.1, pp.71-78, 1993.
[44] U. Pal and S. Datta. Segmentation of Bangla unconstrained handwritten text. Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR03), 2003.
[45] A.G. Hochuli et al. Handwritten digit segmentation: Is it still necessary? Pattern Recognition, vol.78, pp.1-11, 2018.
[46] N. Arica and F.T. Y. Vural. An overview of character recognition focused on off-line handwriting. IEEE Transactions on Systems, Man, and Cybernetics, Applications and Reviews, vol.31, no.2, 2001.
[47] A. Rehman and al. Implicit vs explicit based script segmentation and recognition: A performance comparison on benchmark database. Int. J. Open Problems Compt. Math., vol.2, no.3, ISSN 1998-6262, 2009.
[48] K.M. Sayer. Machine recognition of handwritten words: A project report. Pattern Recognition Pergamon Press, vol.5, pp.213-228, 1973.
[49] S. Karthik and S. Murphy. Segmentation and recognition of handwritten Kannada text using relevance feedback and histogram of oriented gradients: A novel approach. International Journal of Advanced Computer Science and Applications (IJACSA), vol.7, no.1, 2016.
[50] P. Cavalin et al. An implicit segmentation based method for recognition of handwritten strings of characters. Proc. of ACM Symposium on Applied Computing, pp.836-840, 2006.
[51] C.K. Dhande and P.M. Mahajan. A survey on evaluating neural network and hidden Markov model classifiers for handwriting word recognition. International Journal of Advances in Engineering & Technology, January 2014.
[52] S. V. Rice et al. Optical character recognition: An illustrated guide to the frontier. Kluwer Academic Publisher, 1999.
[53] A. Vinciarelli. A survey on off-line cursive word recognition. Pattern Recognition, vol.35, no.7, pp. 1433-1446, 2002.
[54] O.D. Trier et al. Feature extraction methods for character recognition: A survey. Pattern Recognition, vol.29, no.4, pp.641-662, 1996.
[55] M. Bahashwan et al. Efficient segmentation of Arabic handwritten characters using structural features. IAJIT, 2015.
[56] C.D. Aravinda et al. Kannada handwritten character recognition using multi feature extraction techniques. International Journal of Science and Research (IJSR), 2014.
[57] R. Gaur et al. A survey on feature extraction techniques for handwritten character recognition. International Journal on Recent and Innovation Trends in Computing and Communication, vol.5, issue 5, 2017.

[58] M.I. Jubair and P. Banik. A simplified method for handwritten character recognition from document image. International Journal of Computer Applications, vol.51, no.14, 2012.
[59] V. Kumar and S. Joseph. Handwritten character recognition using bayesian decision theory. Int. Journal of Computer Science & Emerging Technologies (IJCSET), vol.1, no.2, 2010.
[60] M.R. Phangtriastu et al. Comparison between neural network and support vector machine in optical character recognition. 2nd Int. Conference on Computer Science and Computational Intelligence (ICCSCI), 2017.
[61] T. Jindal and U. Bhattacharya. Recognition of off-line handwritten numerals using an ensemble of MLPs combined by Adaboost. Proc. of the 4th International Workshop on Multilingual OCR (MOCR '13), 2013.
[62] Liu et al. Classification and learning for character recognition: Comparison of methods and remaining problems. Int. Workshop on Neural Networks and Learning in Document Analysis and Recognition, 2005.
[63] A. Hirwani. Handwritten character recognition system using neural network. International Journal of Advance Research in Computer Science and Management Studies, vol.2, issue 2, 2014.
[64] A. Yuan et al. Off-line handwritten English character recognition based on convolutional neural network. 10th IAPR International Workshop on Document Analysis Systems, 2012.
[65] M. Mathur and A. Saroliya. Handwritten character recognition using K-means clustering algorithm. International Journal of Science and Research (IJSR), vol.3, issue 7, July 2014.
[66] X. J. Tong et al. Handwritten numeral recognition based on fuzzy c-means algorithm. Ninth Int. Symposium on Distributed Computing and Applications to Business, Engineering and Science, 2010.
[67] H. Cecotti. Active graph based semi-supervised learning using image matching: Application to handwritten digit recognition. Pattern Recognition Letters, vol.73, pp.76-82, 2016.
[68] B. Al-Badr, S.A. Mahmoud. Survey and bibliography of Arabic optical text recognition. Signal Processing, vol.41, pp.49-77, 1995.
[69] H. Modi and M.C. Parikh. A review on optical character recognition techniques. Int. Journal of Computer Applications, vol.160, no.6, February 2017.
[70] V. Gaudissart et al. Sypole : Mobile reading assistant for blind people. 9th Conference Speechand Computer St. (SPECOM'2004), 2004.
[71] R. Milewski and V. Govindaraju. Automatic indexing of handwritten medical forms for search engines. Tenth Int. Workshop on Frontiers in Handwriting Recognition, 2006.
[72] L.G. Hafemann et al. Learning features for off-line handwritten signature verification using deep convolutional neural networks. Pattern Recognition, 2017.
[73] M. Lemaitre. Approche markovienne bidimensionnelle d'analyse et de reconnaissance de documents manuscrits. Thèse de doctorat, université de Paris 5, 2007.
[74] L. Heutte et al. A structural/statistical feature based vector for handwritten character recognition. Pattern recognition letters, vol.19, pp.629-641, 1998.
[75] A. Deppa and R.R. Rao. Feature extraction techniques for recognition of Malayalam handwritten characters: Review. Int. Journal of Advanced Trends in Computer Science and Engineering, vol.3, no.1, pp. 481-485, 2014.
[76] R. El-Hajj, C. Mokbel and L. Likforman. Reconnaissance de l'écriture Arabe cursive: Combinaison de classifieurs MMCs à fenêtres orientées. Actes de CIFED, pp.271-276, 2006.

[77] S. Mozaffari et al. A hybrid structural/ statistical classifier for handwritten Farsi/Arabic numeral recognition. IAPR, 2005.
[78] K.S. Dash et al. Unconstrained handwritten digit recognition using perceptual shape primitives. Pattern Anal Applic. 2016.
[79] R. El-Hajj, C. Mokbel and L. Likforman. HMM-based Arabic cursive handwritten recognition System. Int. Conference on Research Trends in Science and Technology, March 2005.
[80] A. L. Koerich. Unconstrained handwritten character recognition using different classification strategies. ANNPR, 2003.
[81] H. R. Mamatha and K. Srikantamurthy. Morphological operations and projection profiles based segmentation of handwritten Kannada document. Int. Journal of Applied Information Systems (IJAIS), ISSN: 2249-0868, vol.4, no.5, October 2012.
[82] J. L. Henry. Une étude sur le choix des caractéristiques pour la représentation de caractères imprimés. Vision Interface '99, Trois-Rivières, Canada, pp.19-21, 1999.
[83] K. Dharmapala et al. Sinhala Handwriting recognition mechanism using zone based feature extraction. ITRU Research Symposium, 2015.
[84] D. Impedovo and G. Pirlo. Zoning method for handwritten character recognition: A survey. Pattern Recognition, 2013.
[85] S.V. Rajashekararadhya and P. V. Ranjan. Efficient zone based feature extraction algorithm for handwritten numeral recognition of four popular south Indian scripts. Journal of Theoretical and Applied Information Technology, pp.1171-1181, 2008.
[86] D.G. Lowe. Object recognition from local scale-invariant features. Proc. of the Int. Conference on Computer Vision, 1999.
[87] D.G. Lowe. Distinctive image features from scale-invariant keypoints. Int. Journal of Computer Vision, 2004.
[88] Z. Zhang et al. Character-SIFT: A novel feature for off line handwritten Chinese character recognition. 10th Int. Conference on Document Analysis and Recognition, 2009.
[89] M. Zahedi and S. Eslami. Farsi/Arabic optical font recognition using SIFT features. Procedia Computer Science 3, pp.1055-1059, 2011.
[90] M. Diem and R. Sablatnig. Recognition of degraded handwritten characters using local features. 10th Int. Conference on Document Analysis and Recognition, 2009.
[91] O. Surinta et al. Recognition of handwritten characters using local gradient feature descriptors. Engineering Applications of Artificial Intelligence, vol.45, pp. 405-414, 2015.
[92] Y. Ke and R. Sukthankar. PCA-SIFT: A more distinctive representation for local image descriptors. In CVPR (2), pp.506-513, 2004
[93] H. Bay, T. Tuytelaars and L. Van Gool. SURF: Speeded up robust features. Proc. of the ninth European Conference on Computer Vision, May 2006.
[94] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. In Tenth European Conference on Machine Learning ECML-98, pp.137-142, 1999.
[95] G. Csurka et al. Visual categorization with bags of keypoints. In ECCV workshop on Statistical Learning in Computer Vision, pp.59-74, 2004.
[96] K. Mikolajczyk and C. Schmid. Indexing based on scale invariant interest points. In ICCV, vol.1, pp.525-531, 2001.
[97] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In CVPR , 2005.

[98] S. Iamsaat and P. Horata. Handwritten character recognition using histograms of oriented gradient features in deep learning of artificial neural network. Int. Conference on Information Technology Convergence and Security (ICITCS), 2013.
[99] M.W.A. Kesiman et al. Study on feature extraction methods for character recognition of Balinese script on Palm Leaf manuscript images. 23rd Int. Conference on Pattern Recognition, 2016.
[100] R. E. zadeh, M. Jampour. Efficient handwritten digit recognition based on histogram of oriented gradients and SVM. International Journal of Computer Applications (0975-8887), vol.104, no.9, October 2014.
[101] F. Suard et al. Pedestrian detection using infrared images and histograms of oriented gradients. Intelligent Vehicles Symposium, 2006.
[102] J. Bégard et al. Real-Time humans detection in urban scenes. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008.
[103] T. Ojala et al. A comparative study of texture measures with classification based on feature distributions. Pattern Recognition, vol.29, no.l, pp.51-59, 1996.
[104] A. Hirwani and S. Gonade. Handwritten character recognition system using neural network. Int. Journal of Advance Research in Computer Science and Management Studies, vol.2, issue 2, February 2014.
[105] C.S. Yang and Y.H. Yanga. Improved local binary pattern for real scene optical character recognition. Pattern Recognition Letters, 2017.
[106] T. Ojala, and M. Pietikainen. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.24, pp.971-987, 2002.
[107] K. Jayech et al. Arabic handwritten word recognition based on dynamic Bayesian network. The Int. Arab Journal of Information Technology, vol.13, no.3, May 2016.
[108] M. Hu. Visual pattern recognition by moment invariants. IRE Transactions on Information Theory, IT-08, pp.179-187, 1962.
[109] M.R. Teague. Image analysis via the general theory of moments. J. Optical Soc. Am., vol.70, no.8, pp.920–930, August 1980.
[110] Ryszard S. Chora. Feature extraction of gray-scale handwritten characters using Gabor filters and Zernike moments. Computer Recognition Systems 2, ASC 45, pp.340-347, 2007.
[111] P. Phokharatkul and C. Kimpan. Handwritten THAI character recognition using Fourier descriptors and genetic neural networks. Computational Intelligence, vol.18, no.3, 2002.
[112] G.G. Rajput and S.M. Mali. Marathi handwritten numeral recognition using Fourier descriptors and normalized chain code. IJCA Special Issue on Recent Trends in Image Processing and Pattern Recognition, 2010.
[113] H. Kauppinen et al. An experimental comparison of autoregressive and Fourier-based descriptors in 2-D shape classification. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.17, pp.201-207, 1995.
[114] D. Zhang, G. Lu. A comparative study of curvature scale space and Fourier descriptors for shape-based image retrieval. J. Vis. Commun. Image R. vol.14, pp.41-60, 2003.
[115] S. Shahabi and M. Rahmati. Comparison of Gabor-based features for writer identification of Farsi/Arabic handwriting. 10th Int. Workshop on Frontiers in Handwriting Recognition (IWFHR'06), pp.545-550, 2006.
[116] S. Singh et al. Use of Gabor filters for recognition of handwritten Gurmukhi character. Int. Journal of Advanced Research in Computer Science and Software Engineering, vol.2, issue5, May2012.

[117] S. Mallat. A Theory for multiresolution signal decomposition: The wavelet representation. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.2, no.7, 1989.
[118] H. Lacoma. Analyse multi-échelles par ondelettes complexes : Application aux signaux électrophysiologiques intracraniens chez les patients épileptiques. Thèse de Doctorat, 2014.
[119] N. Wang et al. Wavelet compression technique for high-resolution global model data on an icosahedral grid. Journal of Atmospheric and Oceanic Technology, vol.32, 2015.
[120] J. Bobulski. Wavelet transform in face recognition. Biometrics, Computer Security Systems and Artificial Intelligence Applications, Springer Science and Business Media, New York, USA, pp.23-29, 2006.
[121] A. Mowlaei. Feature extraction with wavelet transform for recognition of isolated handwritten Farsi/Arabic characters and numerals, DSP, 2002.
[122] D.J. Romero et al. Directional continuous wavelet transform applied to handwritten numerals recognition using neural networks. JCST, 2007.
[123] D. M. Keysers, Modeling of image variability for recognition. PHD Thesis, 2006.
[124] M. Diem et al. Competition on handwritten digit recognition. 12th International Conference on Document Analysis and Recognition (ICDAR2013), 2013.
[125] Y. Netzer et al. Reading digits in natural images with unsupervised feature learning. In NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011.
[126] V. Vapnik. Statistical learning theory. John Wiley & Sons, 1995.
[127] L. S. Oliveira and R. Sabourin. Support vector machines for handwritten numerical string recognition. Proceedings of the 9th Int'l Workshop on Frontiers in Handwriting Recognition (IWFHR-9 2004), 2004.
[128] D.C.Tran et al. Accented handwritten character recognition using SVM: Application to French. 12th International Conference on Frontiers in Handwriting Recognition, 2010.
[129] N. E. Ayat. Sélection de modèle automatique des machines à vecteurs de support: Application à la reconnaissance d'images de chiffres manuscrits. Thèse de doctorat en Génie PH.D, Montréal, 2004.
[130] C. Cortes and V.N. Vapnik. Support vector networks. Machine Learning, vol.20, no.3, pp.273-297, 1995.
[131] C. C. Chang and C. J. Lin. LIBSVM: A Library for support vector machines. Last updated 2013. Software available at http://www.csie.nte.edu.tw/cjlin/libsvm, 2013.
[132] C. W. Hsu and C. J. Lin. A comparison of methods for multi-class support vector machines. IEEE Transactions on Neural Networks, vol.13, no.2, pp.415-425, 2002.
[133] Y. Meyer. Les ondelettes: Algorithmes et applications, Armand Collin, 1992.
[134] S. Yu et al. A novel license plate location method based on wavelet transform and EMD analysis. Pattern Recognition, vol.48, pp.114-125, 2015.
[135] S. E. N. Correia and J. D. Carvalho. Recognition of unconstrained handwritten numerals using biorthogonal spline wavelets. Proc. of the XII Brazilian Symposium on Computer Graphics and Image Processing, 2000.
[136] U. Bhattacharya and B.B.Chaudhuri. A majority voting schema for multiresolution recognition of handprinted numerals. Proc. of the seventh international conference on document analysis and recognition, 2003.
[137] U. Bhattacharya and B.B. Chaudhuri. Handwritten numeral databases of Indian scripts and multistage recognition of mixed numerals. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.31, no.3, 2009.
[138] S. Sasi et al. Wavelet packet transform and neuro-fuzzy approach to handwritten character recognition, 1997.

[139] M.S. Akhtar and H.A. Qureshi. Handwritten digit recognition through wavelet decomposition and wavelet packet decomposition. Eighth International Conference on Digital Information Management (ICDIM 2013), 2013.
[140] A. Rehman et al. Image classification based on complex wavelet structural similarity. Int. Conference on Image Processing, Belgium, September, 2011.
[141] D. Romero et al. Wavelet-based feature extraction for handwritten numerals. Springer-Verlag. ICIAP 2009, LNCS 5716, pp.374-383, 2009.
[142] L. M. Seijas and E. C. Segura. A wavelet based descriptor for handwritten numeral classification. Int. Conference on Frontiers in Handwriting Recognition, 2012.
[143] M. Ait Aider et al. Recognition of handwritten characters based on wavelet transform and SVM classifier. The International Arab Journal of Information Technology, vol.15, no.6, November 2018.
[144] M. Ait Aider et al. Wavelet feature selection based on support vector machine. Recent Advances in Systems Theory, Signal Processing and Computation, 2013.
[145] A. Tharwat. Principal component analysis: A tutorial. Int. J. Applied Pattern Recognition, vol.3, no.3, 2016.
[146] R. Li and S. Zhang. Handwritten digit recognition based on principal component analysis and support vector machines. CSEE, Part I, CCIS 214, pp.595-599, 2011.
[147] M. Miciak. Radon transformation and principal component analysis method applied in postal address recognition task. Int. Journal of Computer Science and Applications, vol.7, no.3, pp.33- 44, 2010.
[148] J. A. G. Sargo and al. Binary fish school search applied to feature selection: Application to ICU readmissions. IEEE Int. Conference on Fuzzy Systems, 2014.
[149] L. Cordella et al. A Feature selection algorithm for handwritten character recognition. Pattern Recognition, 19th International Conference on Pattern Recognition (ICPR), 2008.
[150] R. Panthong and A. Srivihok. Wrapper feature subset selection for dimension reduction based on ensemble learning algorithm. Procedia Computer Science, vol.72, pp.162-169, 2015.
[151] D. Panzoli. Proposition de l'architecture Cortexionist pour l'intelligence comportementale de créatures artificielles. Thèse de doctorat, université de Toulouse, 2008.
[152] D.E. Rumelhart et al. Learning internal representations by error propagation. Parallel distributed processing, vol.1, no.8, pp.319-362, 1986.
[153] M. Welling, and G.E. Hinton. New learning algorithm for mean field Boltzmann machines. ICANN, Madrid, In Dorronsoro J.R. (ed) Lecture notes in Computer Science, vol. 2415, pp.351-357, 2002.
[154] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. Science, Vol.313, pp.504-507, 2006.
[155] G. E. Hinton. A fast learning algorithm for deep belief nets. Neural Computation 2006.
[156] J. Wang et al. A folded neural network autoencoder for dimensionality reduction. Proceedings of the International Neural Network Society Winter Conference, 2012.
[157] G. Hu et al. An improved dropout method and its application into DBN-based handwriting recognition. Proceedings of the 36th Chinese Control Conference, 2017.
[158] L. Sadouk et al. Handwritten Tifinagh character recognition using deep learning architectures. Proceedings of the 1st International Conference on Internet of Things and Machine Learning (IML '17), 2017.
[159] O. K. Oyedotun et al. Deep learning in character recognition considering pattern invariance constraints. I.J. Intelligent Systems and Applications, pp.1-10, 2015.

[160] Y. Saeed et al. Handwritten digit recognition using stacked autoencoders. Proceedings of Student-Faculty Research Day, CSIS, 2017.
[161] D. Hubel and T. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. Journal of Physiology, vol.160, pp.106-154, 1962.
[162] K. Fukushima. Néocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biol. Cybernetics, vol.36, pp.193-202, 1980.
[163] Y. LeCun et al. Handwritten digit recognition with a backpropagation network. In: Touretsky, D.S. (ed.) Advances in Neural Information Processing Systems, vol.2, 1990.
[164] J. Bouvrie. Notes on convolutional neural networks. 2006.
[165] J. Nagi et al. Max-pooling convolutional neural networks for vision-based hand gesture recognition. IEEE International Conference on Signal and Image Processing Applications (ICSIPA), 2011.
[166] J. Bjorck et al. Understanding batch normalization. 32nd Conference on Neural Information Processing Systems (NeurIPS 2018), 2018.
[167] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International Conference on Machine Learning, pp.448-456, 2015.
[168] V. Nair and G.G. Hinton. Rectified linear units improve restricted Boltzmann machines. Proceedings of the 27th International Conference on International Conference on Machine Learning (ICML'10), pp.807-814, 2010.
[169] D.Scherer et al. Evaluation of pooling operations in convolutional architectures for object recognition. 20th International Conference on Artificial Neural Networks (ICANN), September 2010.
[170] D. Yu et al. Mixed pooling for convolutional neural networks. Springer International Publishing Switzerland, 2014.
[171] M. D. Zeiler and R. Fergus. Stochastic pooling for regularization of deep convolutional neural networks. Proceedings of the International Conference on Learning Representation (ICLR), 2013.
[172] T. Williams. Wavelet pooling for convolutional neural network. ICLR 2018.
[173] W. Liu et al. Large-margin Softmax loss for convolutional neural networks. International Conference on Machine Learning, New York, NY, USA, JMLR: W&CP, vol. 48, 2016.
[174] J. Gu et al. Recent advances in convolutional neural networks. Pattern Recognition, vol.77, pp.354-377, 2018.
[175] D.P. Kingma and J.L. Ba. ADAM: A method for stochastic optimization. ICLR 2015.
[176] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In International Conference on Artificial Intelligence and Statistics, pp.249-256, 2010.
[177] Y. Lecun et al. Efficient backprop. In Neural networks, tricks of the trade, Lecture Notes in Computer Science LNCS 1524, Springer Verlag, 1998.
[178] F. Lauer F., Suen C.Y., and Bloch G. A trainable feature extractor for handwritten digit recognition. Pattern Recognition, vol. 40, no. 6, pp. 1816-1824, 2007.
[179] S. Roohi and B. Alizadehashrafi. Persian handwritten character recognition using convolutional neural network. 10th Iranian Conference on Machine Vision and Image Processing, 2017.
[180] M. M. Rahman et al. Bangla handwritten character recognition using convolutional neural network. I.J. Image, Graphics and Signal Processing, vol.8, pp.42-49, 2015.

[181] Calderon et al. Handwritten digit recognition using convolutional neural networks and Gabor filters. In Proc. Int. Congr. Comput. Intell, 2003.
[182] P. Sermanet et al. Convolutional neural networks applied to house numbers digit classification. In ICPR, pp.3288-3291, 2012.
[183] T. Williams and R. Li. Advanced image classification using wavelets and convolutional neural networks. 15th IEEE International Conference on Machine Learning and Applications, 2016.
[184] X. X. Niu and C.Y. Suen. A novel hybrid CNN-SVM classifier for recognizing handwritten digits. Pattern Recognition, vol.45, pp.1318-1325, 2012.
[185] C. Shi et al. Fisher vector for scene character recognition: A comprehensive evaluation. Pattern Recognition, vol.72, pp.1-14, 2017.
[186] I. Daubechies. Orthonormal hases of compactly supported wavelets. Commun. Pure Appl. Math., vol.91, pp.909-996, 1988.
[187] S. Mallat and W. L. Hwang. Singularity detection and processing with wavelets. IEEE Transactions on Information Theory, vol.38, no.2, pp.617-643, 1992.
[188] I. Daubechies. Ten Lectures on Wavelets. SIAM, 1992.

Résumé

Cette thèse est consacrée à la reconnaissance des chiffres manuscrits isolés en s'appuyant principalement sur la Transformée en Ondelettes Discrète (TOD), le classifieur Machine à Vecteurs de Support (SVM) et les Réseaux de Neurones Convolutifs (CNN). Quatre contributions sont apportées:
La première porte sur le choix du type de l'Ondelette et des sous-bandes images générées par la TOD qui conviennent le mieux à la discrimination des chiffres manuscrits.
La seconde s'est basée sur l'étude comparative de plusieurs techniques de caractérisation des chiffres manuscrits. Cette étude nous a permis alors de mettre en place une méthode d'extraction de caractéristiques associant la TOD à une technique basée sur les Histogrammes de Gradients Orientés (HOG). Toutefois, les caractéristiques dégagées restent impertinentes, nombreuses voire redondantes. Afin de surmonter cette difficulté; nous avons développé une technique de réduction et de sélection de caractéristiques qui combine l'Analyse en Composantes Principales (ACP) et la méthode de Sélection Séquentielle Ascendante (SFS).
La quatrième et majeure contribution consiste en la suggestion d'un Réseau de Neurones Convolutif (CNN) permettant d'extraire des caractéristiques multiéchelles par apprentissage. Ce dernier combine le CNN standard; le classifieur SVM et la Transformée en Ondelettes.

Mots clés : Reconnaissance des Chiffres Manuscrits, Ondelettes, SVM, Réseaux de Neurones Convolutifs, HOG, ACP.

Abstract

This thesis is devoted to the recognition of isolated handwritten digits mainly based on the Discrete Wavelet Transform (DWT), Support Vector Machine classifier (SVM) and Convolutional Neural Networks (CNN). Four contributions are made:
The first one deals with the choice of Wavelet type and subband images generated by the DWT that are most suitable for discrimination of handwritten digits.
The second one is based on a comparative study of several techniques for characterizing handwritten digits. This study allowed us to set up a feature extraction method associating the DWT with a technique based on the Histograms of Oriented Gradients. However, the features released remain impertinent. Many of them even are redundant. To overcome this difficulty; we have developed a feature reduction and selection technique that combines Principal Component Analysis (PCA) and the Sequential Forward Selection (SFS).
The fourth and major contribution consists in suggesting the use of a Convolutional Neural Network allowing to extract multiscale features by learning. This combines the standard CNN, the SVM classifier and the Wavelet Transform.

Key words: Handwritten Digit Recognition, Wavelets, SVM, Convolutional Neural Networks, HOG, PCA.