REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET

DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE DE CONSTANTINE

FACULTE DES SCIENCES DE L'INGENIEUR

DEPARTEMENT D'ELECTRONIQUE

THESE DE MAGISTER

Présenté par

Mr : Azizi Rebiai

Option

Traitement du signal

Thème :

UNE APPROCHE HYBRIDE POUR LA RECONNISSANCE D'ECRITURE ARABE MANUSCRITE

Examiné par le jury

Président	Mr. A.Charef	Professeur	Université de constantine
Rapporteur	Mr. A.Bennia	Professeur	Université de constantine
Examinateur	Mr. M.Khamadja	Professeur	Université de constantine
Examinateur	Mr. Menssouri	Professeur	Université de constantine

ANNEE 2006/2007

TABLE DES MATIERES

CHAPITRE II PRETRAITEMENT

CHAPITRE III EXTRACTION DES CARACTERISTIQUES

CHAPITRE IV CLASSIFICATION NEURONALE ET STRUCTURELLE

INTRODUCTION GENERALE

La reconnaissance de l'écriture manuscrite est le vieux rêve de tous ceux qui ont eu besoin d'entrer des données dans un ordinateur. Il remonte à plus d'une trentaine d'années. Aujourd'hui, il existe plusieurs domaines dans lesquels la reconnaissance de l'écriture manuscrite est attendue avec impatience, par exemple dans le tri automatique du courrier, le traitement automatique de dossiers administratifs, des formulaires d'enquêtes, ou encore l'enregistrement des chèques bancaires.

Ces applications montrent clairement les spécificités du domaine de la reconnaissance de l'écriture manuscrite par rapport à celui de la reconnaissance optique des caractères (OCR : Optical Character Recognition) qui concerne les caractères imprimés ou dactylographiés. Il est nécessaire de distinguer également la reconnaissance en ligne (on-line) de l'écriture manuscrite, qui relève plutôt de l'interfaçage entre l'homme et l'ordinateur (un stylo spécial est connecté à la machine et ne fonctionne que sur une tablette sensible), de la reconnaissance hors ligne (off-line). Seule la reconnaissance hors ligne sera considérée dans ce travail.

Dans la reconnaissance de l'écriture en général, nous distinguons deux grandes catégories d'applications : applications à vocabulaire limité, où généralemet le nombre de mots à reconnaître constitue un lexique de taille infèrieure à 100 mots ; et les applications à

vocabulaire étendu, où plusieurs milliers, voire des dizaines de milliers de mots forment un dictionnaire d'appui.

Quelques systèmes de reconnaissance de l'écriture manuscrite ont été réalisés et sont opérationnels à ce jour. Cependant, ils sont spécifiques à un domaine précis et sont encore limités. Par exemple, en ce qui concerne la poste, la reconnaissance de l'écriture manuscrite, contrairement à celle des caractères imprimés, se limite au code postal en chiffres ainsi qu'à la ville en caractères majuscules.

La reconnaissance universelle de l'écriture arabe manuscrite par l'ordinateur est du domaine de la fiction pour quelques années encore. Tous les chercheurs sont confrontés à un problème difficile et incontournable, celui de la segmentation. La segmentation fait partie du processus de prétraitement et d'extraction de l'information, qui est un préalable à toute reconnaissance. A cet effet, quelques travaux de recherche ont été consacrés aux caractères isolés, alors que d'autres ont été orientés vers les textes, où la segmentation des mots en caractères et les variations des formes suivant la psition dans le mot sont analysées. D'autres recherches ont été consacrées spécifiquement à la reconnaissance des mots manuscrits isolés à vocabulaire limité.

Notre travail s'insère dans le cadre général de la reconnaissance automatique hors-ligne (off-line) de l'écriture arabe manuscrite à vocabulaire limité, avec application sur la reconnaissance des montants littéraux des chèques postaux. Nous nous somme fixé comme objectif de développer et valider une approche hybride de reconnaissance et pour cela nous proposons une approche qui tire profit de la combinaison de deux méthodes : connexioniste et structurelle.

D'une part, nous utilisons la méthode connexioniste basée sur un nombre réduit de caractéristiques significatives de l'écriture arabe afin d'obtenir l'ensemble des mots hypothèses. D'autre part, la méthode structurelle sélectionne la meilleure hypothèse parmi celles générées par le classificateur neuronal.

Dans un premier chapitre, une analyse des méthodes et outils de la reconnaissance des textes manuscrits sera présentée. L'étude du système de reconnaissance est faite en introduisant un modèle général à cinq étapes reflétant d'une manière générale le processus de reconnaissance dans les différents systèmes existants dans la littérature.

Le deuxième chapitre sera consacré à l'étape préliminaire de tout système de reconnaissance, celle de prétraitement. Les différents traitements que doivent subir l'image brute avant sa manipulation ont été présentés, parmi lesquelles figurent entre outre, la binarisation, la correction de l'inclinaison, la suppression du bruit et la normalisation.

Le troisième chapitre est entièrement consacré à une phase cruciale et prépondérente : l'extraction des caractéristiques qui détermine lui seule presque deux tiers de l'efficacité d'un système de reconnissance. Les points essentiels dans la conception de ce module sont mentionnés. Ensuite, des méthodes d'extraction de certaines primitives ont été examinées.

Dans le chapitre suivant sera présentée une étude des deux méthodes adoptées dans ce travail : Le classificateur neuronal et l'approhe structurelle.

Dans le dernier chapitre, nous ferons la synthèse de notre contribution axée sur l'utilisation d'une approche hybride pour la reconnaissance d'écriture arabe manuscrite à vocabulaire limité. La méthodologie adoptée pour la reconnaissance ainsi que ces différentes phases sont détaillées. Les résultats expérimentaux obtenus seront aussi discutés.

Le travail se termine par une conclusion générale en présentant quelques percpectives pour la poursuite de ce travail.

CHAPITRE I
RECONNAISSANCE D'ECRITURE MANUSCRITE

1. INTRODUCTION

La lecture automatique de l'écriture manuscrite présente un intérêt indéniable dans l'accomplissement des tâches fastidieuses comme celles que l'on rencontre dans certains domaines : le tri postal, la lecture de chèques bancaires, la lecture des bordereaux, des bons de commande, des feuilles de déclaration. A priori, on pourrait concevoir la reconnaissance du manuscrit comme une émanation de la reconnaissance de l'écriture imprimée ou dactylographiée, pour laquelle les principaux obstacles semblent avoir été surmontés. Mais le passage de la reconnaissance de l'imprimé à celle du manuscrit n'est pas aussi évident que certains avaient pu le supposer initialement. Sa mise en œuvre est beaucoup plus complexe. Les difficultés rencontrées proviennent principalement de la nature même de l'écriture manuscrite, de son caractère cursif et de son extrême variabilité.

Cependant, malgré ces difficultés, la particularisation des problèmes à résoudre et la spécification des tâches à accomplir ont permis de parvenir aujourd'hui à un nombre de plus en plus important de réalisations tangibles.

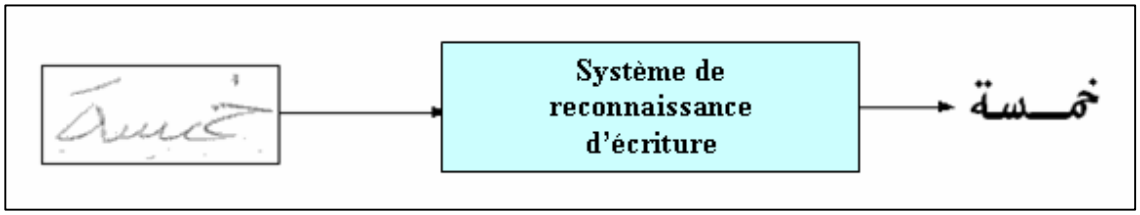


Fig. I.1. Schéma synoptique d'un système de reconnaissance d'écriture (arabe).

2. DIFFERENTS ASPECTS DE RECONNAISSANCE DE L'ECRITURE

Aucun système n'est actuellement capable de reconnaître l'écriture manuscrite de façon universelle comme peut le faire tout être humain lettré. Les systèmes existants se répartissent en deux grandes classes de méthodes de reconnaissance (figure I.2):

- La reconnaissance hors ligne ou « Statique ».
- La reconnaissance en ligne ou « Dynamique ».

- Dans le cas hors ligne, il s'agit de reconnaître des textes manuscrits à partir de documents écrits au préalable. L'image du texte écrit est numérisé à l'aide d'un scanner, les informations recueilles se présentent sous la forme d'une image discrète constituée d'un ensemble de pixels. L'écriture prend l'aspect d'un signal spatial bidimensionnel numérisé.
- Dans le cas en ligne, il s'agit de reconnaître l'écriture au fur et à mesure de son tracé. Le texte est saisi avec un stylo et une tablette à numériser, les informations recueilles sont constituées par une suite ordonnée de points (définis par leurs coordonnées) échantillonnés à cadence fixe. L'écriture prend l'aspect d'un couple de signaux temporels numérisés.

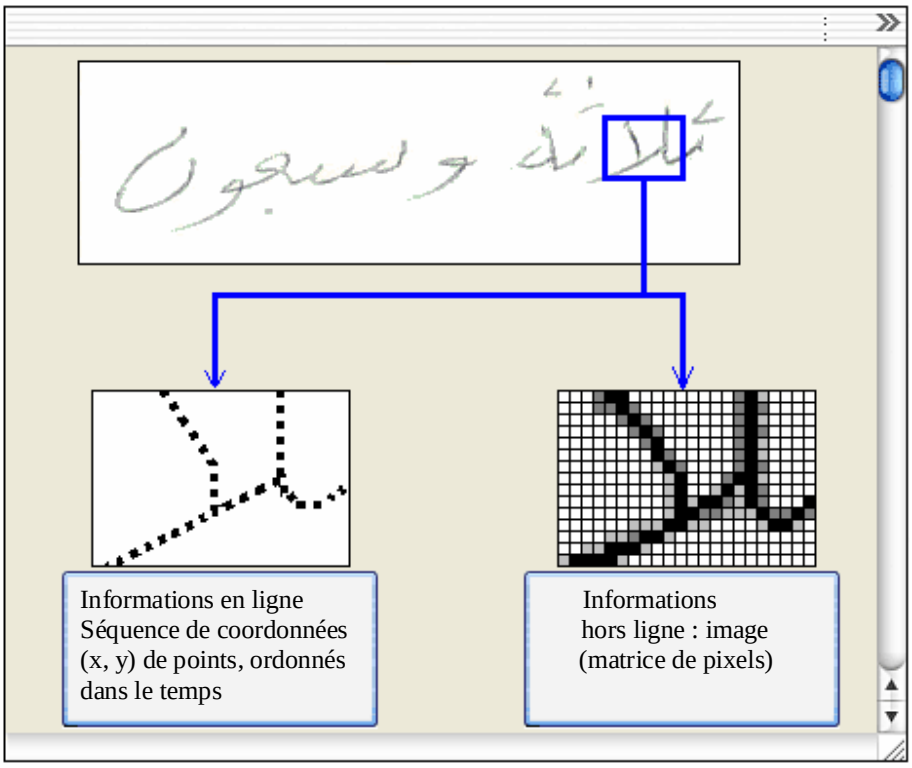


Fig. I.2. Ecriture en ligne et hors ligne.

La reconnaissance hors ligne ne peut pas a priori s'appuyer sur l'information temporelle du tracé qui est perdue, mais elle peut tenir compte de l'épaisseur du tracé (les pleins et les déliés). La reconnaissance en ligne peut disposer de l'information temporelle (vitesse, accélération, levés de stylo, retours en arrière, barres de t, points diacritiques), mais d'aucune information sur l'épaisseur du tracé si on ne dispose pas d'un signal de pression de la pointe du stylet sur le support.

3. PARTICULARISATION DES PROBLEMES

Dans les deux cas, la reconnaissance de l'écriture manuscrite n'a pu progresser que grâce à une particularisation des problèmes à résoudre. Par cette particularisation, le but recherché est de diminuer l'influence de la variabilité sur la reconnaissance. Ainsi, les spécialistes 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 : le type d'écriture considéré, le style d'écriture, le nombre de scripteurs potentiels, la taille du vocabulaire utilisé.

3.1 Type d'écriture

Des contraintes plus ou moins fortes concernant le type d'écriture peuvent être imposées au scripteur. On distingue le précasé pour lequel le scripteur doit s'efforcer d'écrire dans des cases prédéfinies (ex. : sur des bordereaux et les formulaires), le zoné où le scripteur écrit dans des zones bien délimitées (ex. : dans des zones grisées ou colorées), le guidé caractérisé par l'existence d'une ligne support (comme par exemple sur les chèques), et le cas général où l'emplacement de l'écriture est libre.

3.2 Style d'écriture

À ces dispositions s'ajoutent des contraintes concernant le style d'écriture. *Tappert* a établi la classification suivante (figure 1.3) par ordre de difficulté croissante de reconnaissance : caractères précasés de type script, caractères scripts détachés, écriture scripte pure (lettres séparées), écriture cursive (lettres entièrement liées à l'intérieur des mots), écriture mixte mélangeant le script et le cursif.

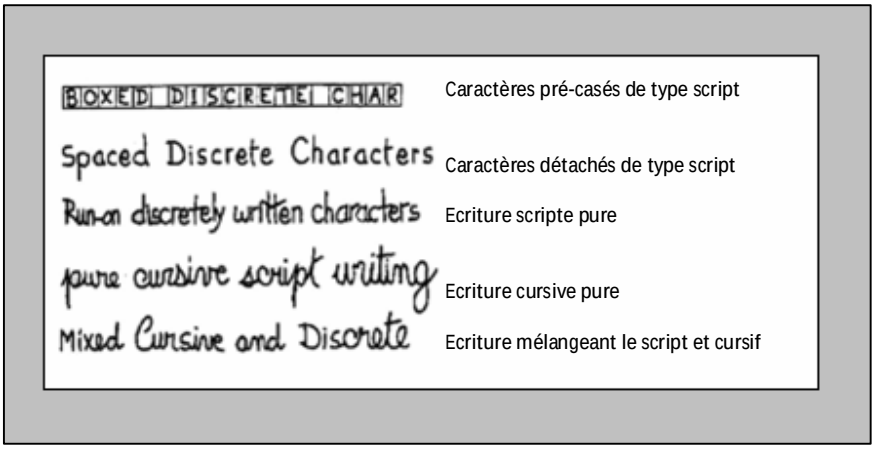


Fig. 1.3 Les différents types d'écriture manuscrite, d'après *Tappert.*

3.3 Nombre de scripteurs potentiels

La diminution du nombre de scripteurs permet évidemment une diminution de la variabilité inter-scripteur. On peut distinguer trois types de systèmes de reconnaissance d'écriture, classés par ordre de complexité de fonctionnement croissante :

- **mono-scripteur** : un seul scripteur peut utiliser le système de reconnaissance après apprentissage de son écriture ;
- **multi-scripteur** : le système peut reconnaître les écritures d'un groupe restreint de personnes, soit après adaptation à l'écriture de chacun, soit sans adaptation ;
- **omni-scripteur** : le système est censé reconnaître toutes les écritures. Dans ce cas, la variabilité intra-scripteur s'ajoute à la variabilité inter-scripteur.

3.4 Taille du vocabulaire à reconnaître

A priori, on peut distinguer trois grandes catégories d'applications :

- **applications à vocabulaire très limité :** où le nombre de mots (ou de symboles) à reconnaître constitue un lexique de taille réduite (inférieure à 100 mots), comme par exemple dans le cas de l'ensemble des mots utilisés pour écrire en toutes lettres les montants des chèques. Il est alors possible de confronter chaque mot à reconnaître à l'ensemble des mots du lexique ;

- **applications à vocabulaire étendu :** mais pouvant être réduit de façon dynamique, comme l'ensemble des noms de rues associés à un bureau de poste distributeur ;
- **applications à vocabulaire très étendu :** plusieurs milliers ou dizaines de milliers de mots formant un dictionnaire et pour lesquels la confrontation systématique n'est plus possible comme par exemple le dictionnaire des noms de commune.

4. APPLICATIONS

Grâce à la particularisation des problèmes, la reconnaissance de l'écriture a permis le développement d'applications particulières et efficaces concernant la reconnaissance de l'écriture hors ligne et en ligne.

4.1 Hors ligne

La reconnaissance de l'écriture hors ligne connaît un essor important dans les domaines associés au développement d'intérêts économiques.

- **Lecture des adresses postales**

La lecture des codes postaux manuscrits associés à la lecture des noms des villes a permis d'étendre le développement des machines de tri automatique du courrier (lettres et objets plats).

- **Domaine bancaire**

La reconnaissance des montants littéraux manuscrits est associée à la reconnaissance des montants numériques pour valider la lecture des chèques. La reconnaissance des chèques peut être associée à celle des coupons correspondants. Des machines capables de lire plusieurs milliers de chèques à l'heure sont déjà en service actuellement.

La reconnaissance des montants numériques a permis la création des guichets automatiques de remise de chèques bancaires. Dans ces automates, le client, identifié par sa carte bancaire, indique sur le clavier le montant du chèque. Si le montant coïncide avec le montant numérique reconnu, l'encaissement du chèque est immédiatement validé.

- **Formulaires et bordereaux**

Il s'agit principalement des applications de la reconnaissance des caractères précasés mauscrits à la lecture des formulaires de sondage, les bordereaux de commande, les constats d'assurance.

4.2. En ligne

La reconnaissance de l'écriture en ligne connaît un essor important dans les domaines associés à une communication homme-machine naturelle, conviviale et ergonomique ainsi qu'à l'informatique nomade.

- **Ordinateurs sans clavier**

La reconnaissance de l'écriture en ligne a pour ambition de remplacer le clavier et la souris de l'ordinateur par un stylo. Cette modification a pour but de rendre les ordinateurs plus conviviaux en permettant de les utiliser dans des situations très diversifiées (prises de notes, rédaction de commandes, de constats d'accidents, enseignement de l'écriture, etc.). Il s'agit, d'une part, de rendre l'informatique nomade.

La reconnaissance en ligne de l'écriture manuscrite présente également un intérêt fondamental lorsque le nombre de caractères à saisir et à reconnaître est très grand, c'est notamment le cas pour tous les textes en caractères orientaux : chinois, japonais, coréen… etc.

5. ORGANISATION GENERALE D'UN SYSTEME DE RECONNAISSANCE DE L'ECRITURE

La reconnaissance de l'écriture manuscrite en ligne et hors ligne peut être considérée, à première vue, comme étant typiquement un problème de reconnaissance des formes. Elle s'articule autour des principes habituels : l'acquisition de l'information par un organe sensoriel (capteur), l'extraction d'une représentation codée de la forme, le classement de l'objet à partir de sa représentation codée parmi un ensemble de catégories ou classes possibles. [1]

Pour faciliter la compréhension, nous avons considéré que l'organisation d'un système classique de reconnaissance de l'écriture se déroulait en cinq opérations successives (figure 1.4) :

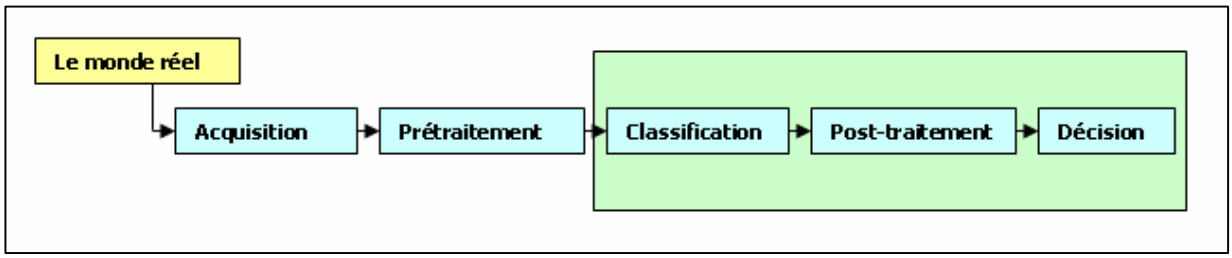


Fig. I.4 Organisation d'un système de reconnaissance d'écriture manuscrite.

5.1. Le monde physique (réel)

C'est un espace analogique de dimension infini appelé espace des formes. Les objets, dans cet espace, sont décrits de différentes façons avec plusieurs propriétés. La loi de passage au monde discret nécessite une sélection et par conséquent une certaine simplification.

5.2. Acquisition de l'écriture

L'écriture est numérisée et elle est transformée en une image. C'est l'entrée de ce système. Cette étape est assez simple mais très importante car elle influence sérieusement les étapes suivantes.

Il y a deux paramètres importants :

- **Résolution** : la résolution normale est 300 dpi. Pourtant, quand la taille de l'écriture est petite, il faut augmenter la résolution.
- **Niveau d'éclairage** : si on ajuste le scanneur pour que l'image soit plus claire, le bruit est réduit mais des traits minces disparaissent aussi.

1. Acquisition de l'écriture hors ligne

Dans le contexte de l'écriture hors ligne, les systèmes d'acquisition les plus courants sont essentiellement des scanners ou des caméras linéaires (barrettes CCD).

La résolution de l'image numérisée influence les étapes ultérieures d'un système de lecture automatique. Il est communément admis que la résolution optimale d'une image est fonction de l'épaisseur du trait d'écriture.

2. Acquisition de l'écriture en ligne

Dans le cas de l'acquisition « en ligne », les dispositifs d'acquisition les plus répandus sont des tablettes à numériser ou des « papiers électroniques ». L'échantillonnage du tracé délivre une série de coordonnées décrivant la trajectoire du stylet au cours du temps. Les tablettes les plus perfectionnées permettent également d'avoir accès aux informations de pression et d'inclinaison du stylo.

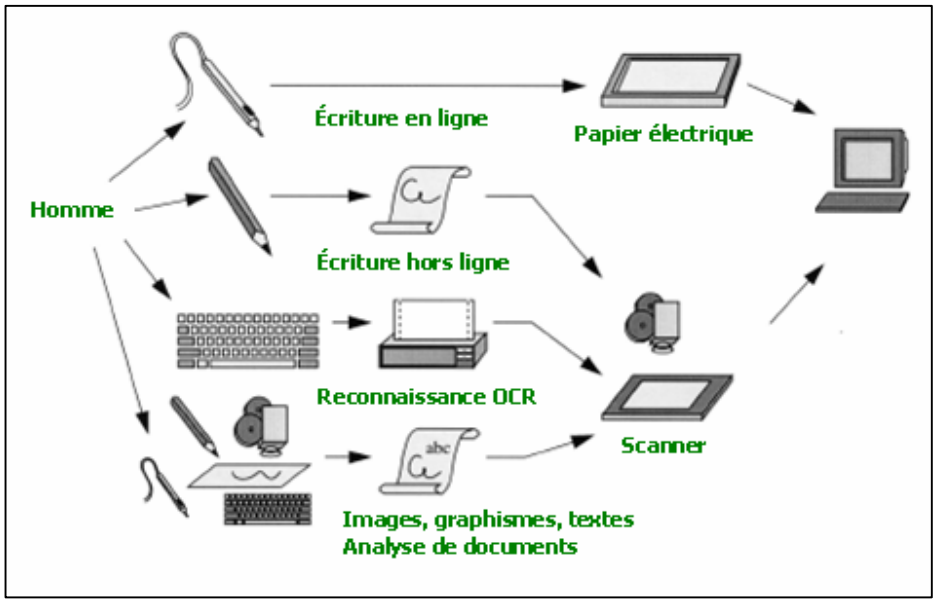


Fig. I.5 Communication écriture Homme-machine.

5.3. Prétraitement

L'étape de prétraitement est l'une des premières tâches d'un système de reconnaissance après l'étape d'acquisition. Cette étape n'est pas spécifique à la reconnaissance du manuscrit mais fait partie de tout système de reconnaissance de forme visant à améliorer la qualité de l'information pour les étapes de traitements qui vont suivre. Dans le cadre du manuscrit, elle regroupe l'ensemble des processus visant au bon conditionnement du message écrit et qui sont indispensables à son identification. On peut catégoriser les pré-traitements en fonction du mode d'acquisition de l'écriture (en-ligne, hors-ligne). On peut subdiviser, dans les deux modes, les étapes de pré-traitements en deux groupes d'une part pré-traitements qui concernent le ré-échantillonnage et la quantification du signal provenant du capteur (scanner dans le cas hors-ligne, tablette à digitaliser dans le cas en-ligne), et d'autre part les pré-traitements qui concernent la localisation ainsi que la normalisation de l'information manuscrite afin d'éliminer dans une certaine mesure la variabilité due à la disposition spatiale de l'information (taille, inclinaison…). Si le premier groupe de pré-traitements reste propre à chaque mode d'acquisition, le second en revanche est commun aux deux.

5.4. Reconnaissance [1]

La reconnaissance regroupe les deux tâches d'apprentissage et de décision qui jouent des rôles assez proches dans les systèmes de reconnaissance des formes. En effet, à partir de la même description de la forme en paramètres, elles tentent, toutes les deux, d'attribuer cette forme à un modèle de référence. Le résultat de l'apprentissage est, soit la réorganisation ou le renforcement des modèles existants en tenant compte de l'apport de la nouvelle forme, soit la création d'un nouveau modèle représentant la forme entrée. Le résultat de la décision est un « avis » sur l'appartenance ou non de la forme aux modèles de l'apprentissage. Nous allons revenir dans la suite sur la fonction de ces tâches en développant quelques approches classiques qui les réalisent.

1. Apprentissage

Dans le cas d'apprentissage il s'agit en fait de fournir au système un ensemble de formes qui sont déjà connues (on connaît la classe de chacune d'elles). C'est cet ensemble d'apprentissage qui va permettre de « régler » le système de reconnaissance de façon à ce qu'il soit capable de reconnaître ultérieurement des formes de classes inconnues.

Il existe deux types d'apprentissage, supervisé et non supervisé.

- **Apprentissage supervisé**

L'apprentissage est dit supervisé si les différentes familles des formes sont connues a priori et si la tâche d'apprentissage est guidée par un superviseur ou professeur, c'est-à-dire le concepteur, indique, pour chaque forme échantillon rentrée, le nom de la famille qui la contient. La tâche d'apprentissage tente de conserver ses liens de parenté en répartissant les familles dans des classes séparées entre elles.

- **Apprentissage non supervisé**

On l'appelle aussi, suivant l'approche utilisée, *classification automatique*, inférence ou encore *apprentissage sans professeur*. Il s'agit, à partir d'échantillon de référence et de règles de regroupement ou de modélisation, de construire automatiquement les classes ou les modèles sans intervention de l'opérateur. Ce mode d'apprentissage nécessite un nombre élevé d'échantillons et des règles de construction précise et non contradictoires pour bien assurer la formation des classes. Il évite l'assistance d'un opérateur mais n'assure pas toujours une classification correspondant à la réalité (celle de l'utilisateur).

2. La décision

La décision est l'ultime étape de reconnaissance. A partir de la description en paramètres, elle recherche, parmi les modèles d'apprentissage en présence, ceux qui sont les plus « proche ». La notion de proximité à un différent en fonction de la nature de la représentation et de type de la méthode. La décision peut conduire à un succès si la réponse est unique (un seul modèle répond à la description de la forme). Elle peut conduire à une confusion si la réponse est multiple (plusieurs modèles correspondent à la description, ce qui dénote une certaine ambiguïté volontaire ou non au sein de l'apprentissage). Enfin, la décision peut conduire à un rejet de la forme si aucun des modèles ne correspond à sa description. Dans les deux premiers cas, la décision peut être accompagnée d'une mesure de vraisemblance, appelée aussi *score* ou *taux* de reconnaissance.

6. Méthodes de reconnaissances

Deux approches s'opposent en reconnaissance des mots : globale et analytique :

6.1. Méthodes globales

Les méthodes globales de reconnaissance des mots sont a priori séduisantes puisqu'elles ne rencontrent pas les difficultés liées aux ambiguïtés provenant de la segmentation. Toutefois, elles ne peuvent se concevoir que dans le cas d'un vocabulaire limité, ou dans le cas de vocabulaires de taille plus grande mais limitée dynamiquement.
Elles reposent sur une caractérisation globale du mot.

6.2. Méthodes analytiques

Contrairement aux méthodes de reconnaissance globales, les méthodes analytiques cherchent à identifier les graphèmes issus de la segmentation (fragments de lettres, des lettres ou des regroupements de lettres) pour reconstituer les mots. Elles présentent l'intérêt de pouvoir se généraliser à la reconnaissance d'un vocabulaire étendu ou limité dynamiquement, à partir d'une phase d'apprentissage des classes de graphèmes.

7. Différentes approches

Après avoir extrait les caractéristiques de la forme à reconnaître, le système OCR procède automatiquement à la classification. La classification est le stade principal de prise de décision. Celle-ci concerne l'identification de la forme, comme étant le membre d'une classe. Elle se fait

en comparant les caractéristiques de la forme à celles d'un modèle de référence. Souvent, le résultat de cette étape est un ensemble de solutions au lieu de générer une solution unique.
La classification cherche, à partir des caractéristiques de la forme, parmi les modèles d'apprentissages en présence, ceux qui sont les plus proches. La nature de la proximité a un sens différent en fonction de la nature de représentation et de type de méthode. La décision peut conduire à un succès si la réponse est unique, Elle peut conduire à une confusion si la réponse est multiple. Enfin, la décision peut conduire à un rejet de la forme si aucun modèle ne correspond à sa description.

Les principales méthodes utilisées sont les méthodes stochastiques, neuro-mimétiques, markoviennes, des méthodes issues de l'intelligence artificielle, de la théorie des sous-ensembles flous ou des méthodes mixtes combinant ces différentes approches.

7.1. Méthodes stochastiques

Dans ces méthodes de reconnaissance des formes ou il va s'agir de regrouper les formes inconnues dans des classes, on va considérer que chacune des classes est régie par un phénomène stochastique (par exemple on dira qu'on s'intéresse à une classe de comportement gaussien : chaque forme de la classe a une certaine probabilité de se produire et la loi de probabilité correspondante est gaussienne).

Décision bayésienne

La théorie de la décision Bayésienne est la théorie centrale des méthodes stochastiques où les problèmes de décision sont traités en termes de probabilités. Le point central de cette théorie est la règle de Bayes qui permet en fait de choisir l'hypothèse ayant la probabilité la plus élevée.

La règle de Bayes permet de calculer ce qu'on appelle les probabilités a posteriori des classes, c'est à dire $P(w_i/x)$ à partir des probabilités a priori des classes $P(w_i)$ et des fonctions de densité des probabilités $p(x/w_i)$.

$$P(w_i / x) = \frac{P(x / w_i) \times P(w_i)}{P(x)} \quad \forall i = 1,2 \text{ avec } P(x) = \sum_{j=1}^{2} P(x / w_j) \times P(w_j)$$

L'interprétation de $P(w_i/x)$ est la suivante: ayant obtenu une forme (décrite par une valeur x de) $P(w_i/x)$ donne la probabilité d'avoir alors la classe w_i.

La règle de décision Bayésienne est alors pratiquement immédiate :

Par exemple, pour *i=1,2,* ayant remarqué une forme inconnue *x* (la forme est assimilée au paramètre qui la caractérise) on dira que la forme inconnue *x* est de la classe w_1 si :

$$P(w_1/x) > P(w_2/x)$$

Sinon elle est de la classe w_2 Soit sous une forme simplifiée:

REGLE DE DECISION BAYESIENNE

Ayant observé *x* on décide :

w_1 *si* $P(w_1/x) > P(w_2/x)$

w_2 sinon

Ou encore

w_1 si $P(x/w_1) \times P(w_1) > P(x/w_2) \times P(w_2)$

Dans cette règle on voit de façon particulièrement claire qu'il est nécessaire et suffisant de connaître pour toutes les classes cherchées la probabilité a priori et la loi de densité de probabilité de cette classe.

7.2. Méthodes géométriques ou statiques

Elles consistent à extraire de la forme donnée un ensemble de mesures qui consistent les composants d'un vecteur d'un espace de représentation de dimension m. Ces mesures sont en nombre assez élevé pour la reconnaissance hors ligne ; elles comprennent principalement des informations topologiques et métriques : surfaces, régions, profils, concavités, boucles, intersections par des droites, rapports de longueurs et rapports de surfaces…

Par la reconnaissance en ligne, les mesures sont en nombre plus réduit ; elles consistent notamment en nombre de lettres dépassant le corps du mot (zone médiane dans laquelle s'inscrivent les minuscules sans extension) vers le haut ou le bas, en coefficients de transformées *(ex : descripteurs de Fourier),* en mesures sur des portions de traits (courbure, rapports, de longueurs, etc.). Le processus de classification s'effectue généralement par une partition de l'espace des représentations à l'aide d'une méthode de classement, notamment par séparation linéaire (ex : méthode des hyperplans).

7.3. Méthodes neuro-mimétiques

Elles sont basées sur l'utilisation de réseaux de neurones de tous types, perceptrons multicouches, carte de Kohonen, *TDNN* (Time Delay Neural Network), *RBF* (Radial Basis Function), neocognitron, etc. Les premières méthodes partaient directement des signaux ou des images normalisés, les méthodes plus récentes extraient d'abord un ensemble de caractéristiques qui servent ensuite à la reconnaissance d'écriture.

Les résultats obtenus par ces sont bons à condition de disposer de bases de données d'apprentissage de très grande taille. En revanche, les temps d'apprentissage peuvent être très longs. Comme l'information est répartie, les causes d'erreurs de ces systèmes peuvent difficilement être analysées.

7.4. Méthodes markoviennes

Les MMC ou modèles de Markov cachés (*HMM* : Hidden Markov Models) sont devenus une technique largement utilisée dans le domaine de la reconnaissance de l'écriture en ligne et hors ligne. Ils ont été créés pour modéliser la production par une source cachée de séquences de signaux variables au cours du temps. Ces derniers sont engendrés par un double processus aléatoire. Le premier processus peut se trouver, à un instant donné, dans un état appartenant à un ensemble de n états distincts. Le passage d'un état à un autre s'effectue selon une matrice de probabilités de transition. La séquence des états successifs n'est pas directement observable, d'où le nom de caché. Chaque transition entre deux états successifs(ou chaque état) émet une observation Oi selon une loi de probabilité associée à chaque transition (où à chaque état).

7.5. Classificateur euclidien [12]

Il s'agit de l'un des plus simples classificateurs qui puissent être conçus. La classe dont le vecteur de caractéristiques moyen est le plus proche, au sens de la distance euclidienne, du vecteur de caractéristiques de l(objet à classifier est assignée à ce dernier. Les fonctions discriminantes utilisées sont donc de la forme suivante :

$$\Phi_i(X) = -\frac{1}{2}(X - M_i)^T (X - M_i)$$

7.6. Le classificateur quadratique [12]

Comme le nom l'indique, les frontières de décision fournies par ce modèle de classificateur sont quadratiques. Les fonctions discriminantes s'expriment par:

$$\Phi_i(X) = -\frac{1}{2}(X - M_i)^T (X - M_i)$$

Où $M_i = E(X / C_i)$ est le vecteur de caractéristiques moyen des éléments qui appartiennent à la classe $C_i, E\{.\}$ désignant l'opérateur d'opérateur d'espérance mathématique et $\{.\}^T$ celui de transposition.

Le terme quadratique XX^T est indépendant de la classe de l'objet, et les fonctions discriminantes peuvent également s'écrire:

$$\Phi_i(X) = M_i^T X - \frac{1}{2} M_i^T M_i$$

Les frontières qui séparent les classes dans l'espace R^d sont ici linéaires.

7.7. La méthode du plus proche voisin

L'algorithme *KNN* (K Nearest Neighbors) associe une forme inconnue à la classe. De son plus proche voisin en le comparant aux formes stockées dans une classe de références nommés prototypes. Il reste les K formes les plus proches de la forme à reconnaître suivant un critère de similarité. Une stratégie de décision permet d'affecter des valeurs de confiance à chacune des classes en compétition et d'attribuer, la plus vraisemblance (au sens de la métrique choisie) à la forme inconnue, le critère de similarité entre deux formes est basé sur la distance euclidienne.

7.8. Méthodes mixtes

Elles consistent à mettre en œuvre des systèmes à structure mixte comme, par exemple, un système combinant n règles floues et modèles de Markov ou encore carte kohonen et règles floues, etc.

Une autre approche consiste à combiner les résultats pondérés de plusieurs modules de classification qui sont de nature différente [25] et à prendre pour décision, la réponse fournie par le principe du "vote majoritaire" ou des techniques plus complexes comme les techniques de décision bayésienne ou la théorie de l'évidence de Dempster-schafer. Pour augmenter les

performances, cette combinaison doit utiliser les points forts de chaque classifieur et écarter leur faiblesse

Ø Post-traitements

Lorsque la reconnaissance aboutit à la génération d'une suite de mots possibles éventuellement classés par ordre de vraisemblance, les post-traitements permettent d'améliorer les taux de reconnaissance par la prise en compte de connaissances pragmatiques et de connaissances linguistiques.

- **Connaissances pragmatiques**

Dans certaines applications comme celles des systèmes de lecture d'adresses, le nombre de chiffres du code postal est connu. D'autre part, la confrontation entre la reconnaissance de ce code postal et du bureau distributeur permet de lever certaines ambiguïtés.

- **Connaissances linguistiques**

L'intégration, à un système de lecture, de connaissances linguistiques de différents niveaux (lexical, syntaxique ou sémantique) permet au système d'effectuer des vérifications et des corrections.

L'utilisation d'un lexique ou d'un dictionnaire permet de valider a posteriori la reconnaissance effectuée. Mais pour éviter un temps de calcul prohibitif. L'utilisation d'un modèle du langage permet de moduler le taux de confiance des hypothèses de mots reconnus.

La syntaxe confirme ou non, suivant les règles grammaticales prédéfinies, la séquence de mots proposés. Celle-ci est largement utilisée dans le cadre de la lecture des montants littéraux et numériques des chèques. Le contrôle sémantique est lié à l'aspect polysémique d'un mot ou d'une phrase. Il permet de réduire la liste des mots candidats, mais sa généralisation à de grands vocabulaires pose des difficultés.

8. Mesure des performances

Après aborder les principes liés à la reconnaissance, il est bon de rappeler les principaux critères permettant de juger le degré d'efficacité de ces systèmes et de mesurer leurs performances. Les évaluations sont généralement caractérisées comme dans le cas de l'imprimé, par les différents taux :

- **le taux de reconnaissance** : pourcentage de mots bien reconnus;
- **le taux de confusion** : pourcentage de mots pour lesquels le système fait une erreur ;
- **le taux de rejet** : pourcentage de mots pour lesquels le système refuse de se prononcer ;
- **le taux de confiance** : pourcentage de mots bien reconnus par rapport à la somme des mots bien reconnus et des mots mal reconnus.

Néanmoins, il ne faut pas considérer ces évaluations de manière absolue mais les replacer dans le contexte de l'application. Pour un type d'application donné, la comparaison des performances des différents systèmes de reconnaissance ne prend une signification que si ces systèmes sont testés sur des bases de données communes. Certains systèmes peuvent présenter des bonnes performances pour un type d'application et être très décevants pour un autre type. Tout dépend de la difficulté de l'application. Cette difficulté est déterminée par deux facteurs : l'ambiguïté des formes à reconnaître qui est difficilement mesurable, la dimension de l'espace des solutions ; la reconnaissance d'un vocabulaire réduit comprenant vingt mots n'est pas comparable à celle d'un vocabulaire de mille mots, encore moins à celle de 40 000 mots. D'autre part, la reconnaissance dépend de l'occurrence de chacun de ces mots, c'est pourquoi certains auteurs ont introduit pour l'écriture la notion de perplexité, déjà employée dans la reconnaissance de la parole. La perplexité est un indicateur basé sur l'entropie des solutions possibles.

9. Etat de l'art

Les progrès réalisés dans le domaine de la reconnaissance de l'écriture manuscrite tant d'un point de vue théorique que méthodologique se caractérisent aujourd'hui par un certain nombre d'applications opérationnelles. Les applications les plus avancées et industrialisées dans le domaine de la reconnaissance de l'écriture hors-ligne concernent la lecture automatique du courrier et la lecture de chèques.

Il est important de souligner que la reconnaissance automatique de l'écriture manuscrite est un problème rarement abordé en dehors des applications spécifiques citées précédemment. En effet, peu de travaux sur la reconnaissance du manuscrit ont abordé le problème de la lecture de textes non contraints.

Contrairement à d'autres systèmes d'écriture peu de travaux ont été menés concernant la reconnaissance de l'écriture arabe.

Le premier travail publié sur la reconnaissance optique de textes arabes remonte jusqu'à 1975. Depuis les années 80, les travaux de recherches en reconnaissance d'écriture arabe sont accentués, et sont devenus relativement importants durant ces dernières années. Parmi ces travaux on trouve :

Al-Yousefi 92 propose une approche de reconnaissance analytique hors-ligne en choisissant comme primitives les moments invariants, le classifieur utilisé est un classifieur bayésien, le taux de reconnaissance est de l'ordre de 99.5 %.

Bouhlila 88 propose un système de reconnaissance hors ligne de l'écriture imprimée en utilisant l'approche analytique par exploitation des caractéristiques structurelles, l'arbre de décision est le classifieur utilisé, un taux de reconnaissance de l'ordre de 91 % est atteint.

Bouslama 98 a utilisé la logique floue pour la reconnaissance en-ligne des caractères, les caractéristiques utilisées sont des caractéristiques structurelles et linguistiques, un résultat de 100% a été enregistré.

Fehri 94 propose une approche analytique pour la reconnaissance hors-ligne de l'écriture manuscrite en utilisant des caractéristiques structurelles et statistiques, la programmation dynamique a été utilisée pour la classification, le taux de reconnaissance est de l'ordre de 98 %.

Leroux et al proposent une stratégie de contrôle pour la reconnaissance des montants littéraux des chèques postaux, c'est un système omni-scripteur fondé sur la stratégie de coopération entre une méthode analytique et méthode globale.

Lemarie et al effectuent une coopération neuro-markovienne pour la reconnaissance des montants littéraux des chèques, le bilan de cette technique été positif, un taux de reconnaissance de l'ordre de 80,28 % est atteint sur une base de test de 2879.

Miled propose deux approches Markovienne pour la reconnaissance hors-ligne omni-scripteur de l'écriture arabe dans un vocabulaire relativement étendu (232 classes de mots différents correspondants à des noms de villes tunisiennes), les modèles ont été entraînés sur une base de 4720 mots et les tests ont été effectués sur une autre base de 5900 mots. Des résultats modestes de 60 % ont été enregistrés.

Cheriet et al proposent un système de reconnaissance de chaînes de chiffres indiens dédie à la lecture automatique des montants numériques de chèques arabes, fondé sur la stratégie de coopération neuro-flou, le taux de reconnaissance est de l'ordre de 95.80 %.

Nazif propose un algorithme de segmentation et de reconnaissance d'écriture manuscrite hors-ligne. La phase de reconnaissance est réalisée par l'approche markovienne, un taux de reconnaissance de 89.3 % pour une base de test de 30000 mots et d'environ 88.8 % pour une base de test de 40000 mot.

Alceu et al proposent une approche markovienne pour la reconnaissance hors-ligne de chiffres arabes. Les tests ont été effectués sur une base de 12802 chiffres et donne un taux de reconnaissance de l'ordre de 89.6 %.

10. Conclusion

Dans ce chapitre nous avons présenté une étude générale sur la reconnaissance d'écriture manuscrite.

Nous avons en premier lieu discuté les principaux aspects liés à la reconnaissance automatique de l'écriture d'une manière générale. Ensuite, Un modèle général à cinq étapes a été introduit. Ce dernier reflète d'une manière générale le processus de reconnaissance. Ensuite, nous avons abordé chaque étape du modèle (Acquisition, prétraitement, extraction des caractéristiques, classification, post-traitement). En précisant les principales méthodes de reconnaissance. Il est à noter qu'il existe différentes issues pour aborder ce domaine et qu'il n'existe pas de méthodes clé pour le traitement automatique d'une écriture donnée.

CHAPITRE III
EXTRACTION DES CARACTERISTIQUES

1. INTRODUCTION

L'extraction de caractéristiques est de fait l'une des deux fonctions essentielles d'un système de reconnaissance d'écriture. Elle comprend la mesure des caractéristiques de la forme d'entrée (nettoyée) qui sont pertinentes à la classification. Lorsque l'extraction des caractéristiques est terminée, la forme est représentée par l'ensemble des caractéristiques extraites.

Il existe un nombre important de caractéristiques possibles que l'on peut extraire d'une forme finie à deux dimensions. Il faut toutefois ne s'attarder qu'aux caractéristiques qui ont une pertinence possible pour la classification, ce qui suppose qu'au cours de la période de conception, le spécialiste s'attarde aux caractéristiques qui, selon une certaine technique de classification, apporteront les résultats les plus certains et les plus efficaces.

Nous traiterons de l'extraction des primitives dans le cas général, c'est-à-dire quel qu'en soit le niveau (lettre, mot) en présentant une synthèse des méthodes employées que nous avons élaborée à partir d'une étude bibliographique sur plusieurs systèmes de reconnaissance.

2. LES OBJECTIFS DE L'EXTRACTION DES PRIMITIVES

Au cours de l'extraction des primitives, plusieurs objectifs, qui précèdent la reconnaissance, peuvent être envisagés. Les principaux objectifs que nous définirons dans la perspective de la reconnaissance de l'écriture manuscrite sont : l'analyse, le paramétrage, la modélisation, le codage et la classification. [11]

- L'**Analyse**, dont la définition littérale est "la décomposition d'un tout en ses parties", consiste généralement en l'extraction d'un ensemble d'attributs caractéristiques du texte. Concrètement, l'analyse d'un texte manuscrit consiste à recueillir des informations statistiques telles que la disposition des lignes d'écriture, leur orientation, leur régularité, l'espacement des mots et des lettres, la régularité et l'inclinaison des lettres, l'épaisseur du trait, ainsi que la ligature des lettres à l'intérieur des mots.

- **Le Paramétrage** consiste à établir une liste d'attributs représentés par une variable binaire (attribut présent ou absent) ou multivaluée (proportionnelle à l'importance de l'attribut), qui ont été détectés et évalués dans l'image. A la différence de l'analyse, le paramétrage ne concerne qu'un mot ou qu'une lettre en vue de sa reconnaissance.

- **La Modélisation** est la construction d'une représentation approximative de la forme entière. A la différence du paramétrage, l'objectif est la réduction de l'information utile, au minimum nécessaire pour représenter complètement la forme, en particulier son aspect et sa structure. Si la modélisation est une description proche d'aspect de la forme originale, on parle alors d'une schématisation, avec une approximation plus ou moins importante. Sinon, il s'agit d'un codage de l'information.

- **La classification** peut être considérée comme une identification partielle de l'information. L'objectif de la classification est, lorsque l'on ne dispose pas de toute l'information nécessaire à l'identification complète de la forme (cas de la reconnaissance), de déterminer quand même une catégorie à laquelle elle appartient. Le principe est que moins d'information est nécessaire pour distinguer les caractères que pour les reconnaître.

3. LA PROBLEMATIQUE DE L'EXTRACTION DE L'INFORMATION [11]

L'objectif de l'étape de l'extraction de l'information est la sélection de l'information pertinente qui se trouve noyée dans la masse de l'information brute acquise. La problématique de cette étape a pour origine le risque de perte d'information signifiante pour la reconnaissance. La minimisation de ce risque est conditionnée par deux dilemmes liés chacun

à un paradoxe de causalité. Il s'agit du dilemme de la réduction et du dilemme de la segmentation.

Nous allons développer dans cette partie la nature de ces dilemmes en présentant les différentes origines de perte d'information lors du processus de la reconnaissance de l'écriture manuscrite.

La première étape du processus est l'acquisition de l'information. Il est évident que l'on ne peut pas extraire une information qui n'est pas présente dans le tracé. C'est pourquoi, avant de traiter de la perte d'information pendant la phase d'acquisition, nous considérerons la perte d'information pendant la phase de production de l'écriture.

Ø La perte d'information pendant la phase de production de l'écriture

Cette première origine de perte d'information, qui a lieu pendant l'écriture elle-même, est due au facteur humain. Cette question, qui s'écarte un peu de notre sujet, a cependant des conséquences déterminantes sur la reconnaissance.

Le caractère subjectif de l'écriture tient au fait que le lecteur étant aussi le scripteur, il peut ne pas s'apercevoir que son écriture est illisible !

L'auteur a en outre la liberté de s'appliquer à ne bien tracer que les lettres permettant de distinguer les mots. Cette attitude a une conséquence importante sur la stratégie de lecture à adopter pour la reconnaissance. En effet, elle implique d'avoir la connaissance du vocabulaire parmi lequel l'auteur a estimé pouvoir distinguer les mots.

La limite à considérer pour ce type de perte d'information est que si un humain ne peut pas lire un texte, un ordinateur ne le pourra probablement pas non plus. Cela fixe une borne inférieure à la qualité d'écriture admissible pour un système de reconnaissance.

Ø La perte d'information pendant la phase d'acquisition

L'objectif de cette étape est d'acquérir le maximum d'informations, afin d'obtenir des images numériques les plus précises. La perte d'information pendant l'acquisition est liée le plus souvent à la faiblesse des capteurs (scanner ou éventuellement caméra), ou aux conditions d'acquisition.

A ce niveau du processus, il reste à déterminer si l'information signifiante est accessible ou non parmi toute l'information acquise.

Ø La perte d'information pendant la phase de réduction

La phase de réduction consiste à extraire l'information signifiante qui est noyée dans le bruit. Dans le cas de la classification, l'objectif est d'éliminer davantage de bruit que d'information afin de faciliter la discrimination. Dans le cas de l'approximation, l'objectif est de modéliser en minimisant la perte d'information.

Une autre perte d'information se produit également au cours de la phase de segmentation.

Ø La perte d'information pendant la phase de segmentation

le problème de la segmentation nous ramène au paradoxe de causalité. Pour pouvoir segmenter le mot en lettres afin de les reconnaître isolément, il est nécessaire au préalable de les localiser, or comment localiser ces lettres sans les avoir reconnues auparavant?

4. LES APPROCHES DE L'EXTRACTION DES PRIMITIVES

En fonction de l'objectif fixé et de la méthode d'extraction choisie, l'approche de l'extraction des primitives peut être systématique ou heuristique.

- La modélisation et le codage conduisent à une approche systématique dans la mesure où l'objectif fixé est la détermination d'une représentation complète de la forme, même de façon approximative. Dans la modélisation, les primitives sont obtenues a posteriori, par le résultat de l'approximation, tandis que, en ce qui concerne le codage, les catégories de primitives sont définies a priori. Un test, qui est par exemple réalisé à l'aide d'une sonde, permet de valider la présence de chacune des primitives sur l'ensemble de la forme.
- Le paramétrage conduit plutôt à une approche heuristique. Dans ce cas, on ne cherche pas nécessairement une représentation complète mais seulement des indices significatifs. De même que dans le cas du codage, ces indices sont des primitives définies a priori.

5. LES CATEGORIES DE PRIMITIVES

Nous avons distingué quatre catégories principales de primitives : les primitives topologiques, statistiques, structurelles et globales.

5.1. Les primitives topologiques ou métriques

Le terme métrique désigne la mesure d'une distance. La topologie est "l'étude des propriétés de l'espace (et des ensembles) du seul point de vue qualitatif". Concrètement, la topologie

consiste, à l'aide de sondes appliquées directement sur l'image "brute", à effectuer par exemple sur l'échantillon les mesures et les tests suivants :

- compter dans une forme le nombre de trous,
- évaluer les concavités,
- mesurer des pentes et autres paramètres de courbures et évaluer des orientations principales,
- mesurer la longueur et l'épaisseur des traits,
- détecter les croisements et les jonctions des traits,
- mesurer les surfaces, les périmètres,
- déterminer le rectangle délimitant l'échantillon, ou le polygone convexe,
- évaluer le rapport d'élongation (ou allongement) longueur/largeur, ...
- rendre compte de la disposition relative de ces primitives.

5.2. Les primitives structurelles

A la différence des primitives topologiques, les primitives structurelles sont généralement extraites non pas de l'image brute, mais à partir d'une représentation de la forme par le squelette ou par le contour. Ainsi, on ne parle plus de trous, mais de boucles. Cependant, pour le reste, les primitives structurelles correspondent à peu près aux primitives topologiques, il s'agit principalement :

- des segments de droite,
- des arcs, boucles et concavités, des pentes,
- des angularités, points extremum et points terminaux, jonctions et croisements.

5.3. Les primitives statistiques

Elles véhiculent une information qui est distribuée sur toute l'image. L'histogramme, qui représente le nombre de pixels sur chaque ligne ou colonne de l'image, en est un exemple classique et simple à calculer. L'histogramme des transitions, comme l'indique son nom, ne retient que le nombre des transitions 0-1 et 1-0.

On utilise aussi une autre primitive statistique basée sur un moyennage des pixels situés à l'intérieur d'un masque rectangulaire. La construction d'une matrice de masque recouvrant la totalité de la forme permet une représentation statistique à partir d'un nombre très réduit de valeurs correspondant à chaque masque.

5.4. Les primitives globales

Elles sont naturellement basées sur une transformation globale de l'image. La caractéristique d'une primitive globale est de dépendre de la totalité des pixels d'une image. Nous en étudierons deux, les transformées de Hough et de Fourier, dans ce chapitre.

6. PRESENTATION DES METHODE D'EXTRACTION DES PRIMITIVES

La détection des caractéristiques est l'une des étapes les plus délicates dans la construction d'un système de reconaissance d'écriture manuscrite. Nous présentons dans cette partie les principales caractéristiques et leurs détections.

6.1. Les mesures topologiques

Parmi ces mesures élémentaires, nous trouvons :

- La surface (égale, par exemple, au nombre de pixels situés à l'intérieur de la frontière de l'objet) ;
- Le périmètre (égal au nombre de pixels situés le long de la frontière de l'objet) ;
- Le nombre de cavités (égal au nombre de trous ou zones composées de pixels n'ayant pas les mêmes niveaux de gris que les pixels de l'objet).

6.2. Les mesures d'orientation

Pour les régions présentant des allongements caractéristiques, il est possible de calculer les directions suivant lesquelles ces allongements ont été réalisés. Ces directions sont calculées à partir des moments d'inertie centrés du deuxième ordre :

$$m_{xx} = \sum_{(i,j)\in R} (j - j_G)^2,$$
$$m_{yy} = \sum_{(i,j)\in R} (i - i_G)^2,$$
$$m_{xy} = \sum_{(i,j)\in R} (i - i_G)(j - j_G),$$

Où i_G et j_G représentent les coordonnées du centre de gravité de la région *R* et *(i, j)* les coordonnées des pixels. La direction α de l'axe d'inertie principal de *R* se déduit de la formule suivante :

$$tg(2a) = \frac{2m_{xy}}{m_{yy} - m_{xx}}$$

Cette orientation n'a de sens que si la région ne présente pas de symétrie de révolution, c'est-à-dire si $m_{yy} = m_{xx}$.

6.3. Les mesures de forme [1]

Une revue des différentes méthodes utilisées pour caractériser la forme d'une région est donnée dans [Pav78, Sha80, Fu77]. En voici deux exemples de nature globale :

- La compacité : $C = \frac{S}{P^2}$,

  Où *S* est la surface et *P* le périmètre. En théorie, cette valeur est maximale pour les cercles et plus petite pour les autres formes. Mais, à cause de la discrétisation, cette valeur est également maximale pour les carrés ou les polygones à peu prés réguliers.

- L'allongement : $C = \frac{long}{l\,\arg}$,

  Où *long* correspond à la plus grande dimension de la région, et *larg* à la plus petite.

6.4. Détection des boucles

Lorsque les boucles sont présentes dans les lettres du mot, leur détection améliore la reconnaissance du mot ; si elles sont absentes, par exemple dans le cas des hampes sans boucles, ou non perceptibles, les lettres restent ambiguës mais cela ne pénalise pas la reconnaissance.

La détection des boucles consiste ainsi à affiner la description des classes génériques en classes plus précises afin de profiter d'une caractéristique facilement détectée. Elle est simplement réalisée grâce à une procédure de remplissage du fond à partir du cadre extérieur du mot, de sorte que toute surface qui n'a pu être remplie est une boucle.

Ø Remplissage

Le cadre extérieur englobant le mot est entièrement rempli à l'aide de l'algorithme de propagation de pixels indiqué plus bas :

On dispose d'une pile dans laquelle on peut mémoriser les coordonnées d'un nombre suffisant de pixels. Un premier pixel vide (de valeur nulle) est choisi, par exemple au coin en haut à gauche, pour initialiser le remplissage : il est placé en première position dans la pile.

Pour chaque pixel vide empilé : on lui attribue une valeur non nulle (remplissage) et chaque pixel à proximité, au sens de la 4-connexité, est empilé seulement s'il est vide et compris dans les limites du cadre

Afin de réduire la taille nécessaire de la pile, on utilise une pile circulaire de sorte que l'emplacement mémoire des premiers pixels sera utilisé de nouveau pour empiler d'autres pixels. La procédure s'arrête lorsque l'indice courant devient égal à l'indice du dernier pixel empilé. Mais si c'est l'indice du dernier pixel empilé qui rattrape en premier lieu l'indice courant, cela signifie que la pile est saturée et qu'une taille plus grande doit être prévue au départ. Ainsi, Les surfaces de l'image non remplies à l'issue de la procédure sont des boucles.

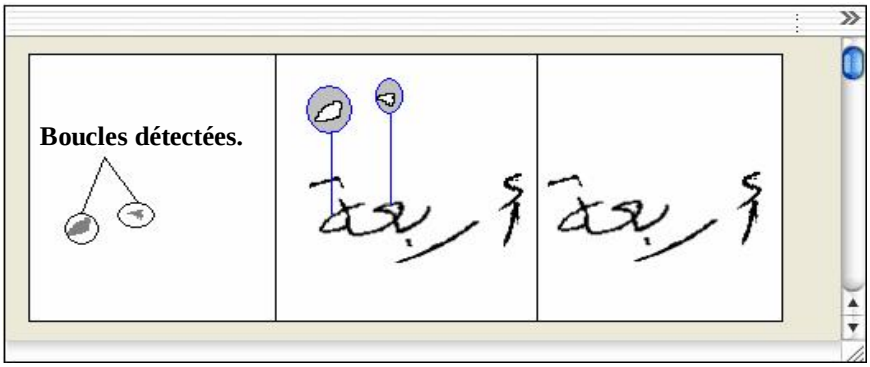


Fig. III.1. Détection des boucles d'un mot.

6.5. Localisation des hampes et jambages

La première étape est généralement la détermination de la zone médiane du mot, c'est-à-dire, la zone qui permet de distinguer les lettres à hampes ou à jambages, de celles qui en sont dépourvues. La méthode la plus simple utilisée est basée sur l'analyse de l'histogramme horizontal de l'image. [11]

Ø Analyse de l'histogramme horizontal

L'histogramme permet de mettre aisément en évidence la zone médiane du mot car la contribution des lettres sans hampe ni jambage y est déterminante. C'est une fonction *h* :

$$h : \quad [1, n] \longrightarrow N^{+}$$
$$i \longrightarrow h(i)$$

Où l'indice *i* représente l'indice des lignes et *n* le nombre de lignes de l'image.

On recherche dans un premier temps une ligne de l'image appartenant à la zone médiane, quelle que soit la combinaison des lettres constituant le mot. Pour cela, on suppose que la hauteur attendue de la zone médiane soit comprise aux alentours d'une valeur h_M fixée à l'avance, car il n'est pas envisagé ici de reconnaître des mots de taille quelconque. On calcule, pour $h_M/4 < i <= n - h_M/4$, la somme $S(i)$ suivante :

$$S(i) = \sum_{j=i-h_M/4}^{i+h_M/4} h(j)$$

On opère ainsi un important lissage. L'indice *i* correspondant à la ligne où la somme *S(i)* est maximum, est noté *M* : dans la plupart des cas, cette ligne d'indice *i* se trouve à l'intérieur de la zone médiane du mot, même si elle est parfois plus près d'un bord de la zone que de l'autre.

Dans un deuxième temps, on recherche dans la partie supérieure à la ligne d'indice *M* ainsi que dans la partie inférieure, les indices des minimums de l'histogramme respectivement m_h et m_b. Dans le cas idéal, ces deux minimums délimitent la zone médiane.

La figure (III.2) illustre la détection de ces paramètres sur le mot "ألفان".

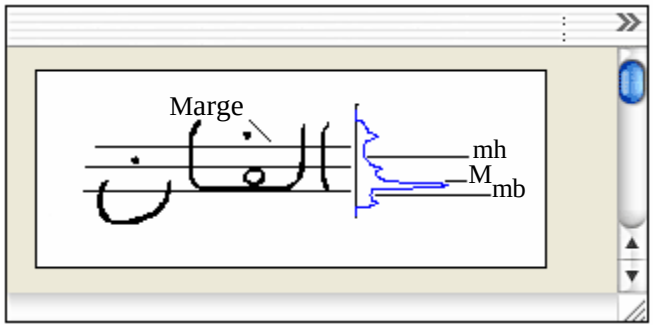


Fig. III.2. Détection des paramètres de l'histogramme.

Lorsque la zone médiane du mot est obtenue, les lettres à hampe ou à jambage sont distinguées des lettres médianes quand elles dépassent les limites de la zone médiane d'une grandeur supérieure à une marge fixée.

Le bon fonctionnement de cette procédure est naturellement tributaire de la direction de l'axe du mot : lorsque celle-ci est oblique, la zone principale de l'histogramme horizontal s'étale, la zone médiane devient ainsi plus difficile à localiser. Et même lorsque la zone médiane est assez bien localisée, les hampes et jambages ne peuvent être détectés que si leur taille est suffisante, c'est-à-dire s'ils présentent un dépassement supérieur à la somme du produit de la longueur du mot par le sinus de l'angle entre l'horizontale et l'axe du mot, et de la marge fixe. En même temps, les lettres sans hampe ni jambage au début et à la fin du mot risquent de dépasser la zone médiane figure (III.3).

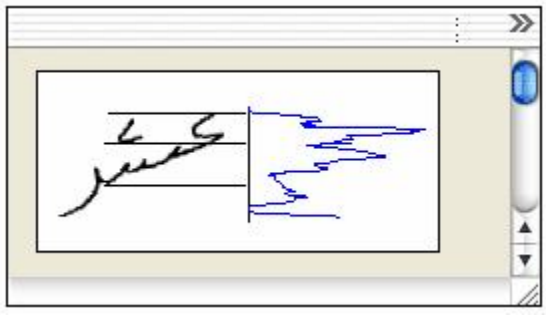


Fig. III.3. Exemple illustrant la confusion dans la détection.

Lorsque l'axe du mot est très oblique, deux solutions sont envisageables. La première, et la plus simple, consiste à effectuer une rotation inverse de celle détectée. La seconde est également intéressante mais plus laborieuse à mettre en oeuvre. Elle consiste à appliquer l'ensemble de la procédure de détection en tenant compte de l'inclinaison mesurée, ce qui suppose, entre autres, le calcul d'histogrammes directionnels.

Cette procédure de détection de la zone médiane du mot donne des résultats satisfaisants lorsque la hauteur moyenne des lettres est connue a priori. D'autre part, la marge de distinction des hampes et des jambages est également un paramètre qui doit être connu a priori : plus l'écriture est régulière et les hampes et jambages courts, plus la marge doit être faible, et plus l'écriture est irrégulière et les hampes et jambages longs, plus la marge doit être importante.

6.6. L'approche VH2D

L'approche VH2D (vertical-Horizontal-2Diagonal), proposée par Cheng et Xia, consiste à projeter chaque caractère sur l'abscisse, l'ordonnée, ainsi que sur les diagonales 45°et 135°. Les projections s'effectuent en calculant la somme des valeurs des pixels i_{xy} selon une direction donnée. [7]

Définitions

- ***Projection verticale***

La projection verticale d'une image $I = (i_{xy})$ (de dimension $N \times N$) représentant un caractère C est dénotée par :

$$P_y^v = \sum_{x=1}^{N} i_{xy} \qquad \text{où} \quad P^v = [P_1^v, P_2^v, ..., P_N^v,]$$

- ***Projection horizontale***

La projection horizontale d'une image $I = (i_{xy})$ (de dimension $N \times N$) représentant un caractère C est dénotée par :

$$P_y^h = \sum_{y=1}^{N} i_{xy} \qquad \text{où} \qquad P^h = [P_1^h, P_2^h, ..., P_N^h,]$$

- ***Projection sur la diagonale 45°***

La projection sur la diagonale 45° d'une image $I = (i_{xy})$ (de dimension $N \times N$) représentant un caractère C est dénotée par :

$$P^{d1} = [P_1^{d1}, P_2^{d1}, ..., P_{2N-1}^{d1},] \qquad \text{où :}$$

$$P_m^{d1} = \begin{cases} \sum_{l=N-m+1}^{N} \sum_{k=1}^{m} i_{lk} & 1 \le \mathrm{m} \le N \quad \text{et } l = k + N - m \\ \sum_{l=1}^{2N-m} \sum_{k=m-N+1}^{N} i_{lk} & N+1 \le m \le 2N-1 \quad \text{et } l = k + N - m \end{cases}$$

- **Projection sur la diagonale *135°***

La projection sur la diagonale 135° d'une image $I = (i_{xy})$ (de dimension $N \times N$) représentant un caractère C est dénotée par :

$$P^{d2} = [P_1^{d2}, P_2^{d2}, ..., P_{2N-1}^{d2},] \qquad \text{où :}$$

$$P_m^{d2} = \begin{cases} \sum_{l=1}^{m} \sum_{k=1}^{m} i_{lk} & 1 \le \mathrm{m} \le N \quad \text{et } k = m - l + 1 \\ \sum_{l=m-N+1}^{2N-m} \sum_{k=m-N+1}^{N} i_{lk} & N+1 \le m \le 2N-1 \quad \text{et } k = m - l + 1 \end{cases}$$

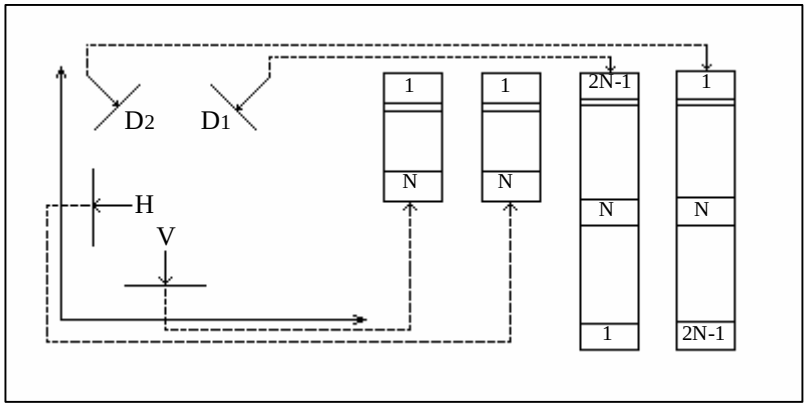


Fig. III. 4. Vecteurs obtenus en effectuant les projections VH2D.

6.7. Extraction et classification des tracés secondaires

Dans les mots arabes, les parties « diacritiques » (Hamza, point,….) font partie intégrante des caractères. On ne peut pas les omettre, leur nombre et leur position au-dessus ou au-dessous du caractère, change le sens de ce dernier. Cette propriété nous a amené à détecter les parties diacritiques séparément des composantes connexes, en vu d'une description globale efficace.

Nous avons effectué dans un premier temps, une discrimination qui sépare le mot en deux types de tracés : le premier représente les points diacritiques, le deuxième groupe les composantes connexes qui peuvent être formées d'un ou plusieurs caractères. Les tracés de types diacritiques rassemblent tout ce qui peut être considéré comme trait secondaire, c'est-à-dire Hamza, point, deux points liés ou trois points liés.

A cet effet, nous avons adopté le paramètre de compacité défini par $s = \frac{A.4p}{p^2}$ tel que : A est l'aire de l'entité connexe, et p son périmètre. Par conséquent, la discrimination entre les tracés secondaires et principaux d'une part, et entre les tracés secondaires eux-mêmes, peut se faire par application de l'organigramme suivant sur chaque tracé dans l'image de mot :

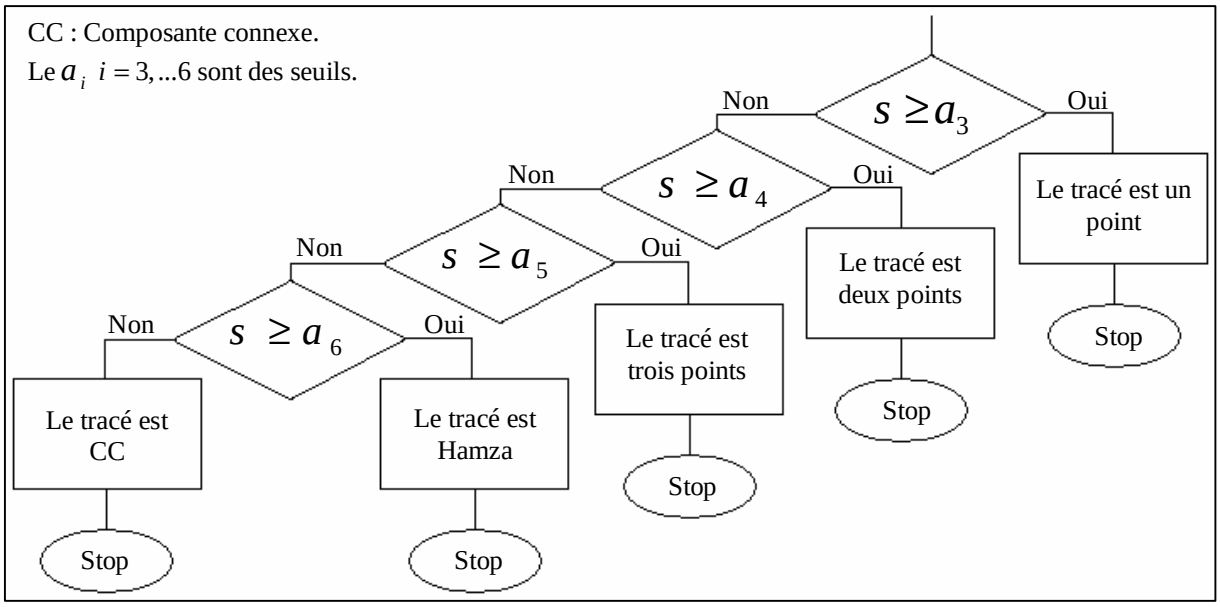


Fig. III.5. Organigramme de classification des tracés.

Les tracés secondaires (point, deux points liés) sont examinés pour déterminer leurs positions vis-à-vis de la ligne de base. Donc, un tracé secondaire est considéré au-dessus de la

ligne de base, si le centre de gravité de son contour est situé au-dessus de cette ligne. Sinon, il est considéré au-dessous. [14]

6.8. Caractéristiques issues de la technique de zonage « Zoning »

Une information globale caractérisant la répartition des points constituant le caractère. Dans la fenêtre minimale est également exploitée. La fenêtre englobant le caractère ou le mot étudié est divisée en régions de façons différentes (figure (III.6)). Pour chaque découpage, on détermine la réparation des points de l'image entre les régions constituant ce découplage. [12]

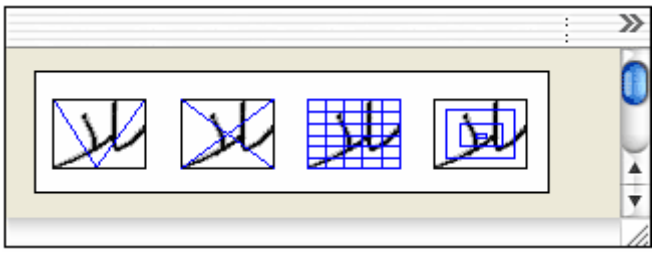


Fig. III.6. Division de l'image en zone de quatre.

6.9. Extraction des profils

L'utilisation des profils pour caractériser les caractères manuscrits a été suggérée par divers auteurs. Le caractère est encadré dans une fenêtre minimale. Le profil compte le nombre de pixels (distance) entre la boîte de bondissement de l'image de caractère et le bord du caractère. Le profil d'un caractère peut être pris à n'importe quelle position, gauche, droite, supérieur, inférieur et orienté (figure (III.7)). Les profils décrivent bien les formes externes des caractères et laissent distinguer entre un grand nombre de lettres.

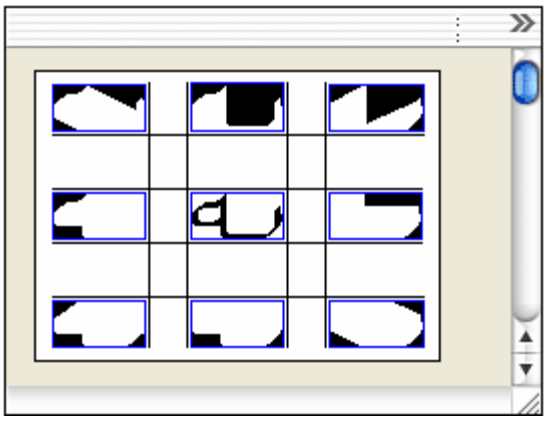


Fig. III.7. Exemple des différentes projections des profils pour la composante « ثـــة ».

6.10. La transformation globale du contour

L'objectif de la transformation globale du contour est d'intégrer à la reconnaissance, des propriétés d'invariance à certaines déformations ou transformations que peut subir la forme, telles que :

- la rotation,
- le changement de taille ou d'échelle,
- une transformation non linéaire (torsion, distorsion, ...).

Lorsque l'on compare une forme test avec une forme apprise, il est nécessaire de tenir compte d'autres variables telles que, par exemple, le point de départ du contour dans la chaîne, ou la résolution du contour (nombre de vecteurs pour le décrire, quelle que soit sa taille).

Les principales transformations globales du contour sont les suivantes :

- codage du contour,
- calcul des moments,
- descripteurs de Fourier.

Ces transformations sont adaptées à des problèmes généraux de reconnaissance des formes, tels ceux rencontrés dans le domaine de l'analyse de scène en robotique, où les formes quelconques peuvent être vues dans toutes les directions, en perspective, avec des parties cachées et souvent du bruit, et aussi en trois dimensions. Dans le domaine de la reconnaissance de l'écriture, les caractères ne sont pas des formes quelconques, elles sont toujours vues à plat, et on ne cherche jamais à les reconnaître à l'envers, dans un miroir, déformées ou partiellement occultées. Il reste cependant des déformations qui sont effectivement rencontrées.

6.10.1. Codage des contours

Le codage des contours est une technique exprimant les frontières discrètes de l'image du caractère par une technique de code ou une suite de symboles. Un premier type de symboles donne la position absolue d'un point apparentant au contour appelé point de départ. Un deuxième type décrit la position relative de tous les autres points des contours **(**figure (III.8)) pris séquentiellement dans un sens défini arbitrairement.

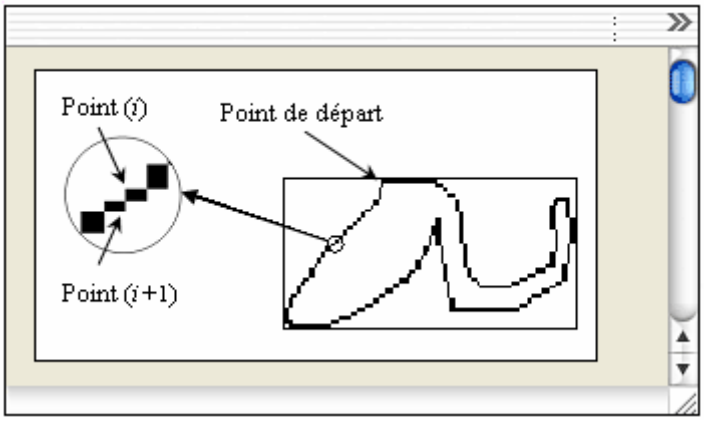
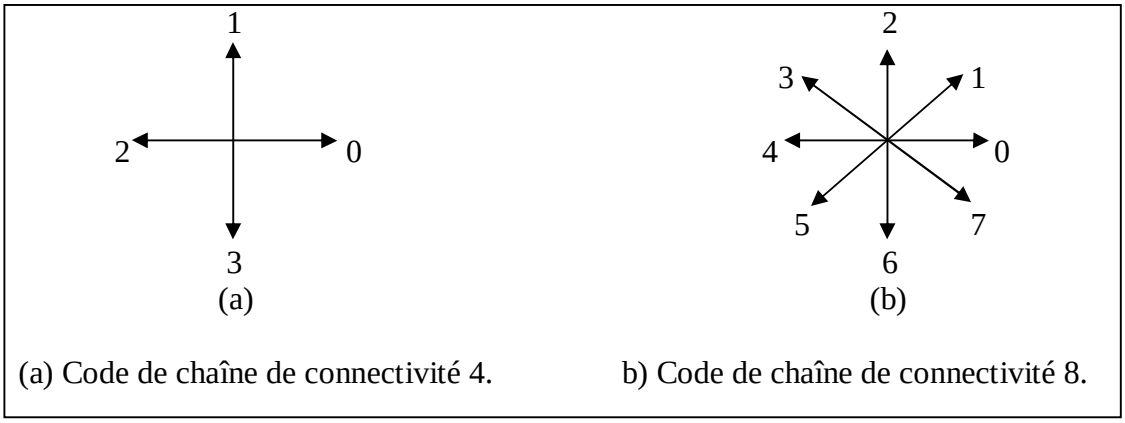


Fig. III.8. Représentation d'un contour de la composante «ثـــة ».

Il existe une technique qui permet de définir la position relative d'un ensemble des points à savoir le code de chaîne ou codage de Freeman.

Dans ce code la position d'un nouveau point voisin $i+1$ par rapport au point i, connexe de connexité 4 où 8, s'exprime au moyen d'un mot binaire à 2 ou à 3 bits. L'analyse des codes successifs peut conduire à distinguer les zones de faible courbure.

Le nombre spécifique indiqué par le mot binaire correspond à l'une des directions qui relie les deux points comme indiqué à la figure (III.9).



(a) Code de chaîne de connectivité 4. b) Code de chaîne de connectivité 8.

Fig. III.9. Définition du code de chaîne.

L'inconvénient essentiel de cette représentation est qu'elle est étroitement liée à un repère et très sensible à l'orientation de l'objet codé. En effet, deux contours peuvent être très semblables et avoir des représentations différentes.

6.10.2. Les moments invariants [5]

La méthode des moments géométriques, comme toutes les méthodes statistiques, permet d'extraire des paramètres propres à la forme à reconnaître. Ces derniers permettent de la distinguer de toutes les autres formes qui lui sont peu semblables. Les moments invariants proposés par Hu ont le mérite de répondre à trois critères qui sont :

- L'invariance par rapport à la translation.
- L'invariance par rapport au changement d'échelle.
- L'invariance par rapport à la rotation.

Si $f(x,y)$ est une fonction continue et non nulle seulement dans une région du plan (x,y), il existe une suite de moments géométriques d'ordre (p,q) unique définie par :

$$mpq = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} x^p y^q f(x,y)dxdy$$

avec p et $q \in IN$

Ces moments ne satisfont aucune des trois propriétés citées plus haut. Par conséquence, une deuxième série de moments invariants par rapport à la translation a été proposée par Papoulis. Elle est définie par :

$$\mu_{pq} = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}(x-\bar{x})^p (y-\bar{y})^q f(x,y)dxdy$$

Tels que $(\bar{x},\bar{y})$ représente les coordonnés du centre de gravité et sont données par :

$$\bar{x} = \frac{m_{10}}{m_{00}}, \quad \bar{y} = \frac{m_{01}}{m_{00}}, \quad \text{et } \mu_{pq} \text{ est le moment centralisé d'ordre } (p,q).$$

Pour une image discrète les relations (3.1) et (3.2) deviennent.

$$m_{pq} = \sum_x \sum_y (x)^p (y)^q f(x,y).$$

$$m_{pq} = \sum_x \sum_y (x-\bar{x})^p (y-\bar{y})^q f(x,y).$$

Dans cette étude on se limitera aux moments d'ordre (3). En effet ils donnent une meilleure séparation des classes par aux moments d'ordre (2) puisque on a plus de paramètres pour décrire la forme à reconnaître.

Les moments centrés d'ordre 3 sont donnés par les relations suivantes :

$$\mu_{10} = m_{10} \quad \frac{m_{10}}{m_{00}}(m_{00}) = 0.$$

$$\mu_{11} = m_{11} \quad \frac{m_{10}}{m_{00}}(m_{01}).$$

$$\mu_{02} = m_{02} \quad \frac{m_{01}^2}{m_{00}}.$$

$$\mu_{20} = m_{20} \quad \frac{m_{10}^2}{m_{00}}.$$

$$\mu_{30} = m_{30} \quad 3\frac{m_{10}}{m_{00}}(m_{20}) + 2\frac{m_{10}^2}{m_{00}^2}(m_{10}).$$

$$\mu_{12} = m_{12} \quad 2\frac{m_{01}}{m_{00}}(m_{11}) \quad \frac{m_{10}}{m_{00}}(m_{02}) + 2\frac{m_{01}^2}{m_{00}^2}(m_{10}).$$

$$\mu_{21} = m_{21} \quad 2\frac{m_{10}}{m_{00}}(m_{11}) \quad \frac{m_{01}}{m_{00}}(m_{20}) + 2\frac{m_{10}^2}{m_{00}^2}(m_{01}).$$

$$\mu_{03} = m_{03} \quad 3\frac{m_{01}}{m_{00}}(m_{02}) + 2\frac{m_{01}^3}{m_{00}^2}.$$

Les moments centrés et normalisés sont définis par :

$$n_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}}.$$

$$Avec: \gamma \quad \frac{p+q}{2} + 1.$$

Ces derniers ont la propriété d'être invariants par rapport à la translation et le changement d'échelle. Hu a proposé dans ses travaux, une autre série de moments qui regroupent l'invariance par rapport à la translation, la rotation et le changement d'échelle.

Ils sont donnés par :

$$\varphi_1 = n_{20} + n_{02}.$$

$$\varphi_2 = (n_{20} - n_{02})^2 + 4n_{11}^2.$$

$$\varphi_3 = (n_{30} - 3n_{12})^2 + (3n_{21} - n_{03})^2.$$

$$\varphi_4 = (n_{30} - n_{12})^2 + (n_{21} + n_{03})^2.$$

$$\varphi_5 = (n_{30} - 3n_{12})^2(n_{30} + n_{12})\left[(n_{30} + 3n_{12})^2 - (3n_{21} + n_{03})^2\right] + (3n_{21} - n_{03})(3n_{21}^2 + n_{12})$$
$$[3(n_{30} + n_{12})^2 - (n_{21} + n_{03})^2].$$

$$\varphi_6 = (n_{20} - n_{02})[(n_{30} + n_{12})^2 - (n_{21} + n_{03})^2] + 4n_{11}(n_{30} + n_{12})(n_{21} + n_{03}).$$

$$\varphi_7 = (3n_{21} - n_{30})(n_{30} + n_{12})[(n_{30} + 3n_{12})^2 - 3(n_{21} + n_{03})^2] + (3n_{12} \quad n_{30})$$
$$(n_{21} + n_{02})[3(n_{30} + n_{12})^2 - (n_{21} + n_{03})^2].$$

On obtient finalement les moments invariants par translation, rotation et changement d'échelle appelé j_i.

La figure suivante montre une image de composant connexe «خمسة» déformés par rotation, translation et par changement d'échelle ainsi les moments j_i,

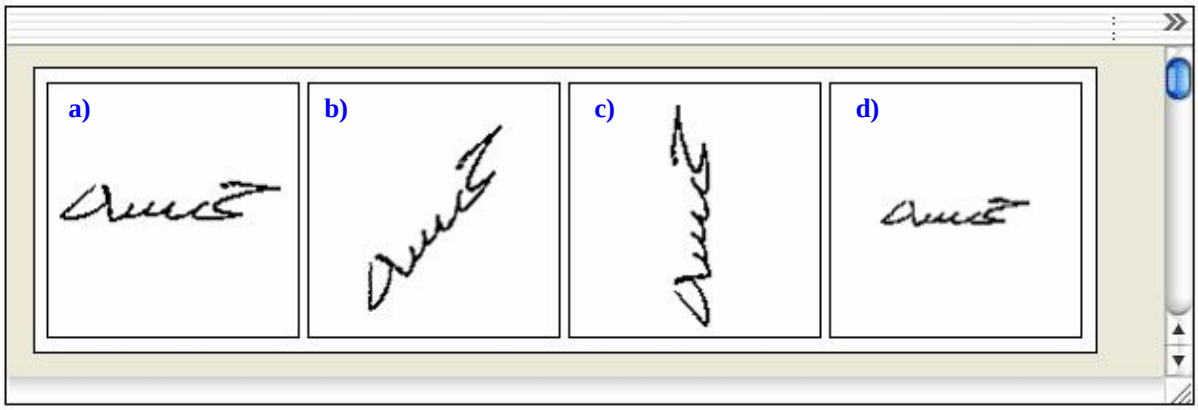


Fig. III.10. Opération géométrique.

a) Image originale.
b) Image tournée de 45°.
c) Image tournée de 90°.
d) Image dont la taille est réduite par 50%. (h/l) 44/138----------30/100

Moments j_i, i=1,…, 7 calculés sur les quatre images décrites au tableau (III.1).

	Figure (a)	**Figure (b)**	**Figure (c)**	**Figure (d)**
j_1	0.0069	0.0069	0.0069	0.0069
j_2	0.0528	0.0473	0.0055	0.0165
j_3	-0.0014	-0.0014	-0.0014	-0.0014
j_4	0.0001	0.0001	0.0001	0.0001
j_5	0.0043	0.0024	-0.0181	-0.0008
j_6	0.0190	0.0146	0.0062	0.0036
j_7	0	0	0	0

Tableau. III.1. Résultat de calcul de moments invariants.

En observant les moments décrits dans le Tableau (III.1), on constate une assez bonne propriété d'invariance.

6.10.3. La transformée de Fourier (TF)

La transformée de Fourier est fréquemment utilisée en traitement d'image lorsque l'on désire effectuer un filtrage un peu délicat qui ne pourrait être réalisé dans le domaine spatial à l'aide d'une simple convolution par un masque. [11]

Ø Aspects théoriques de la TF

Considérons un signal monodimensionnel de type temporel. La *TF* de ce signal est définie de la façon suivante :

$$X(f) = TF\left[x(t)\right] = \int_{-\infty}^{+¥} x(t)\, exp(-i2\pi ft)\, dt$$

Les signaux traités dans cette étude sont codés sur leurs parties réelles seulement, leurs parties imaginaires étant fixées à zéro.

$$\begin{aligned} X(f) &= TF\left[x(t)\right] \quad avec\ x(t)\ réel \\ &= \int_{-\infty}^{+\infty} x(t)\, cos(2\pi ft)\, dt \; - \; i\int_{-\infty}^{+\infty} x(t)\, sin(2\pi ft)\, dt \\ &= Re\{X(f)\} + i\, Im\{X(f)\} \\ &= |X(f)|\ exp(i\, V_x(f)) \end{aligned}$$

Le module, appelé spectre d'amplitude, est une fonction paire. Le spectre de phase (qui est une fonction impaire, sensible aux translations, ne sera pas pris en considération et nous nous limiterons donc à l'étude de |X (f)| pour les seules fréquences positives.

Ø TF discrète (TFD) et TF rapide

Soit Te la cadence à laquelle on prélève N échantillons pendant la durée T d'un signal x ; en introduisant les simplifications de notation suivantes :

$x(k) = x(kTe/T)$;

$X(n) = X(n/T*fe)$;

$W_N = exp(2i\pi/N)$ avec N pair

la Transformation de Fourier Discrète *(TFD)* est définie par :

$$X(n) = \sum_{k=k_0}^{k_0+N\ \ 1} x(k)\ W_N^{-nk}$$

Et la transformation inverse par:

$$x(k) = \frac{1}{N} \sum_{n=-N/2}^{N/2-1} X(n) W_N^{nk}$$

Où *x* est le vecteur du signal discret monodimensionnel et *X* sa *TFD.*

Ø Propriétés de la TF

La *TF* est une transformation linéaire orthogonale qui conserve l'énergie et l'entropie. Toute l'information de l'image est contenue dans sa transformée, c'est une transformation réversible.

A tout produit dans le domaine spatial correspond un produit de convolution dans le domaine fréquentiel, et réciproquement.

Le spectre d'amplitude est invariant par translation (celle-ci ne modifie que le spectre de phase). En dehors de la zone des hautes fréquences du spectre, la *TF* est relativement insensible au bruit car chaque point de l'image transformée contient une contribution de chaque point de l'image originale.

Ø Définition complexe de descripteur de Fourier

Après une opération de suivi de contour d'un objet, on obtient la suite ordonnée de ces *N* points représentée sous la forme complexe (figure (III.11)).

Un point s'y déplaçant génère un signal monodimensionnel complexe *z(n)* où *n* représente l'abscisse curviligne du contour.

$Z(n) = x(n) + j\, y(n)$

x : représente la ligne,

y : représente la colonne.

Pour *n* variant de *0* à *N-1.*

On remarque que cette fonction est périodique de période *T*, ce qui formelise par la relation suivante :

$Z(n+kT)=z(n)$: avec *k* entier

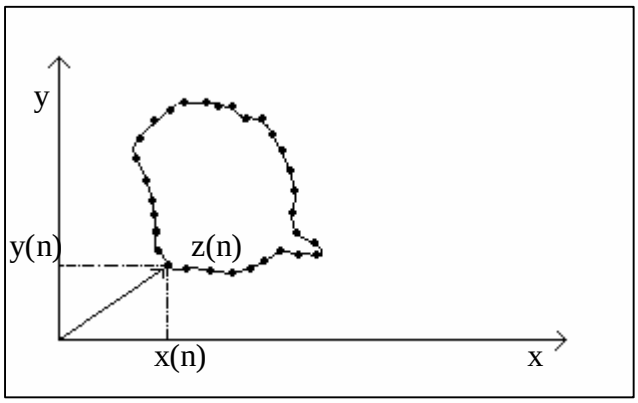


Fig. III.11. Représentation complexe d'un contour.

Compte tenu de la nature périodique *(*de période *N)* de cette suite, on peut la représenter en utilisant la transformation de Fourier discrète *(DFT)*.

$$z(n) = (1/N) \sum_{k=0}^{N-1} Z(k)\, e^{2j\pi kn/N}, \text{Pour } 0 \le n \le N-1$$

$$Z(k) = \sum_{k=0}^{N-1} z(n)\, e^{2jpkn/N}, Pour 0 \le k \le N-1.$$

Les coefficients *Z(k) (k=0, 1,…, N-1)* désignent *les descripteurs de Fourier* du contour. Après application de la FFT, le contour est décrit par les coefficients (*descripteurs*) de *Fourier* que l'on visualise (en module) sous forme de « raies ». La reconstruction du contour peut être réalisée par l'application de la transformée de *Fourier* inverse aux coefficients.

On peut également effectuer une opération de « *filtrage* », par exemple en supprimant certains coefficients. Après transformée inverse, on obtient un contour fermé qui approxime plus ou moins bien le contour initial.

7. Conclusion

Nous avons exposé dans ce chapitre les problèmes liés à l'extraction des primitives. Ces problèmes concernent en premier lieu le choix des primitives qui doivent être très représentatives de l'ensemble des formes à prendre en compte. Ce choix est important car il conditionne toute la méthodologie mise en oeuvre pour la reconnaissance. Il est irréversible et ne peut donc être remis en cause au cours de déroulement du système de reconnaissance.

CHAPITRE II

PRETRAITEMENT

1. INTRODUCTION

Le prétraitement, comme on a déjà vu auparavant, est une technique qui sert à préparer les données reçues du capteur à la phase suivante d'analyse consacrée à l'extraction des paramètres. Cette phase n'est possible et surtout fiable que si les données du capteur sont dénuées de bruit, corrigées de leurs erreurs éventuelles, homgénéisées, normalisées et réduites à l'essentiel. Nous allons développer dans la suite les principales opérations de pré-traitement en les illustrant par des exemples dans le cas des images.

2. BINARISATION

Pour les images acquises en niveaux de gris, la binarisation devient nécessaire avant d'attaquer la phase du traitement. La binarisation permet de mieux distinguer les caractères du fond, elle consiste à attribuer à chaque pixel de l'image une valeur de 0 ou 1 : « 0 » qui représente le noir (le texte), et le « 1 » représente le blanc (le fond de la page). Pour cela, elle applique en premier lieu l'opération de seuillage. [12]

Seuillage

Il consiste à déterminer la valeur du seuil à partir duquel tous les pixels ayant un niveau de gris inférieur à cette valeur sont représentés par un zéros « 0 » le noire, et tout les pixels de niveau de gris supérieur auront la valeur un « 1 » (le blanc).

La valeur du seuil est déterminée à partir de l'histogramme du niveau de gris se trouvant dans la vallée entre les deux pics de l'histogramme figure (II.1).

Le seuil doit être calculé d'une manière adéquate, car les composantes du texte liées des traits fins peuvent se déconnecter, ce qui modifie la forme originale du texte.

L'histogramme de l'image représente la répartition des niveaux de gris. Son calcul se fait en comptant le nombre de pixels dans l'image correspondant à chaque valeur du niveau de gris.

Algorithme de binarisation

Entrée : Image avec niveaux de gris I en format BMP ;
Sortie : Image noire et blanche (1 et 0) I' ;
Début
  Copier le contenu de I dans I' ;
  H = Histogramme (I') ;
$\overline{X}_{Histo} = seuil$
  Pour chaque pixel P de I' **faire**
    Si P< $\overline{X}_{Histo}$ alors P = 1 ; % rendre le pixel noir
    Sinon p = 0 ; % rendre le pixel blanc
Fin

$\overline{X}_{Histo}$ représente le seuil que nous avons choisi, pour une segmentation de l'image par binarisation, (les pixels ayant un niveau de gris inférieur à cette valeur appartiennent à l'objet (noir) et ceux ayant une valeur supérieure appartiennent au fond (blanc)).

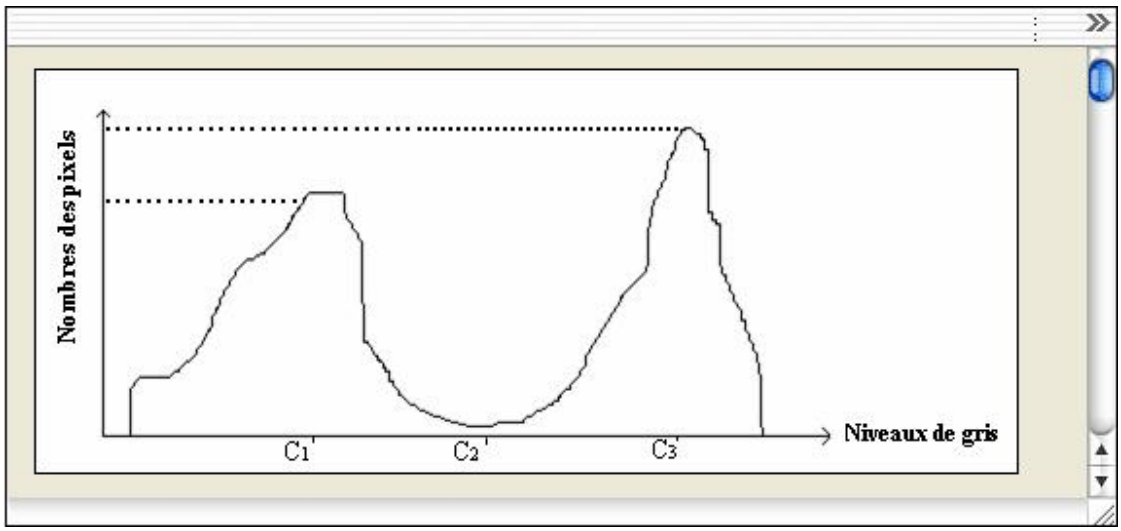


Fig. II.1. Histogramme de niveaux de gris.

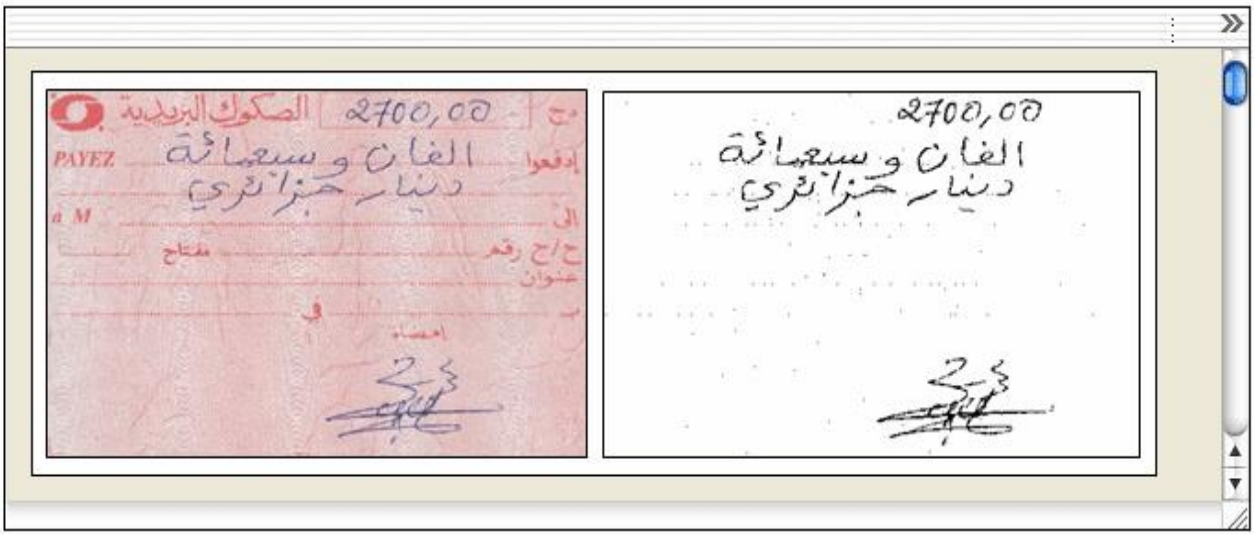


Fig. II.2. Exemple de la binarisation (seuil=170).

3. SUPPRESSION DU BRUIT (TECHNIQUES DE LISSAGE) [1]

L'image de caractères peut être entachée de bruit dû aux artefacts de l'acquisition et souvent à la qualité du document, conduisant soit à des absences de points (trous) soit à des empâtements ou des excroissances et donc à une surcharge de points. Les techniques

de lissage permettent de résoudre ces problèmes par des opérations locales appelées nettoyage et bouchage.

Nous utilisons les notions suivantes pour la description des masques de transformation de voisinage : « 1 » pour un point de la forme et « 0 » pour un point du fond et X pour un point quelconque.

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

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

0	0	0
X	1	0
0	0	0

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

X	0	0
1	1	0
X	0	0

- par élimination des points formant des coins, en appliquant le masque suivant dans les quatre directions principales :

0	0	0
X	1	0
0	X	0

Pour le bouchage, il s'agit d'égaliser les contours et de boucher les trous internes à la forme du caractère en lui ajoutant des points noirs. On distingue :

- le bouchage de trous isolés. Si le voisinage d'un point du fond correspond au masque suivant, alors ce point est mis à 1 :

1	1	1
1	0	1
1	1	1

- la correction des irrégularités des contours dues à un effritement de la forme. On applique sur l'image le masque suivant dans les huit directions :

X	1	1
0	0	1
X	1	1

- la vibration du caractère en vue de simuler les distorsions apparaissant sur le contour. Elle est engendrée par un des masques suivants suivi d'une opération de seuillage :

1	2	1
2	4	2
1	2	1

3	5	3
5	10	5
3	5	3

0	1	1
0	3	1
0	1	1

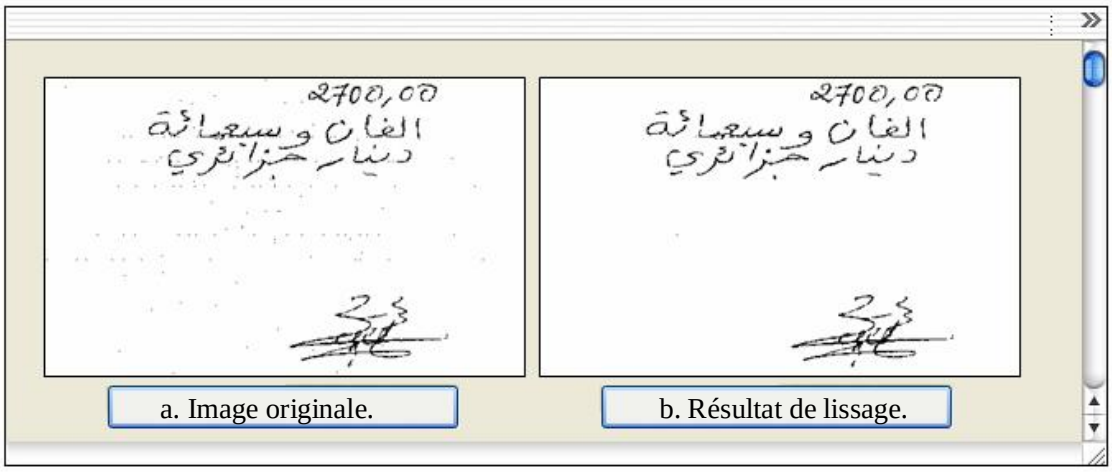


Fig. II.3. Exemple de lissage par les filtres présentés.

4. LA CORRECTION DE L'INCLINAISON [11]

Cette opération consiste à corriger la pente d'un mot ou à redresser l'inclinaison des lettres dans un mot afin de faciliter la segmentation. L'opération est effectuée à l'aide

d'une transformation ligne par ligne où chaque pixel noir de coordonnées *(x, y)* est remplacé par les coordonnées *(x', y')* données par :

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & -\tan q \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Détection de l'inclinaison

Deux sortes d'inclinaisons sont distinguées dans les images de mots. La première est l'inclinaison globale du mot par rapport au cadre de l'image, que l'on représente par l'axe moyen du mot. La seconde est l'inclinaison des lettres par rapport à cet axe.

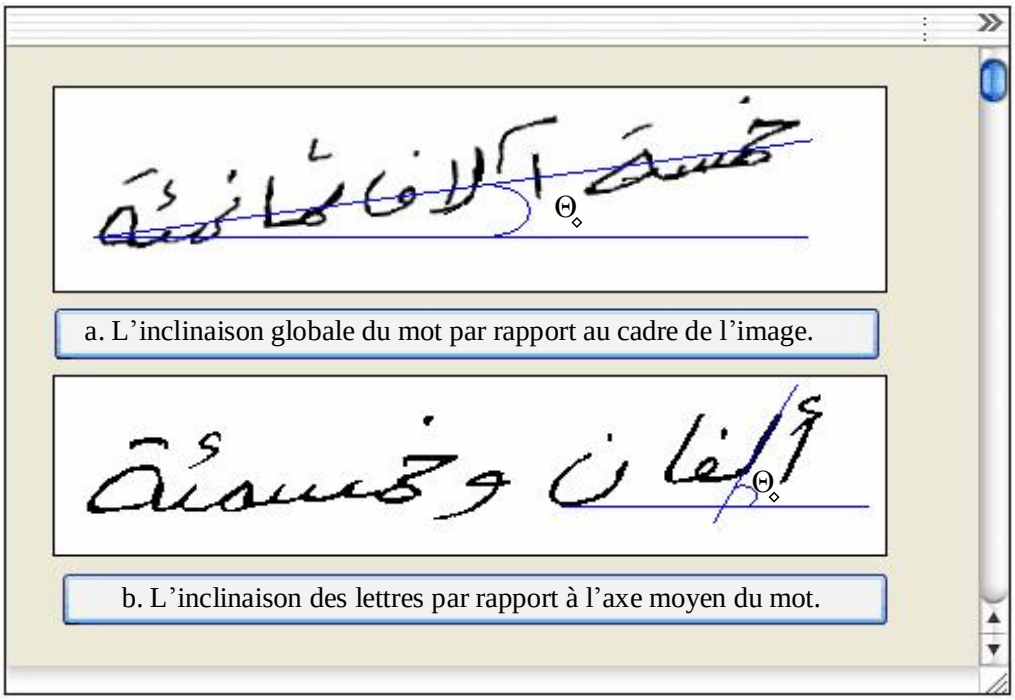


Fig. II.4. Détection de l'inclinaison.

5. NORMALISATION DE LA TAILLE [1]

La taille des caractères peut varier d'une fonte à l'autre et même au sein d'une même fonte après agrandissement ou réduction, ce qui peut causer une instabilité des paramètres. Une technique naturelle de prétraitement consiste à ramener les caractères à

la même taille. Nous allons donner deux exemples d'algorithmes de normalisation de la taille.

L'algorithme de Srihari

Cet algorithme opère en deux étapes. La première normalise le caractère en hauteur et la seconde, en largeur. L'ordre de normalisation ainsi choisi évite que les caractères fins ne se déforment par rapport à des caractères épais.

Il s'agit de transformer l'image du caractère de dimension *hi, li* en une image de dimension *h, l.* une étape intermédiaire consiste à produire une image de dimension *h', l'* avec :

$$P=h/hi \qquad \text{et} \qquad l'=P*li$$

La normalisation en hauteur est exécutée en transformant chaque pixel *(x, y)* noir de l'image du caractère en *(p*x, p*y).* La normalisation en largeur de l'image ainsi obtenue se fait par l'examen de deux cas. Si *l' < l*, alors l'image normalisée en hauteur est centrée dans une surface de dimension h, l. Si, par contre *l'> l*, alors l'algorithme balaye cette image et assigne à noir tout pixel de coordonnées *((x/l')*l, y)* si le pixel *(x, y) l'*est aussi.

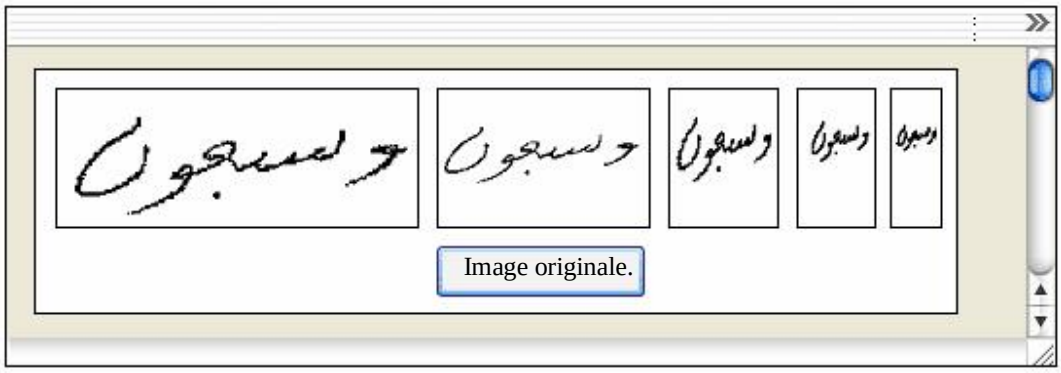


Fig. II.5. Résultats de la normalisation.

6. EXTRACTION DES COMPOSANTES CONNEXES [11]

L'extraction des composantes connexes, procédure également appelée capture des connexités ou étiquetage des pixels, est largement utilisée en Reconnaissance des Formes (RdF) pour segmenter les images binaires. La technique consiste à regrouper les pixels voisins dans un ensemble appelé composante connexe. Chaque ensemble est disjoint des

autres et peut ensuite être aisément isolé. La 4-connexité est distinguée de la 8-connexité suivant que le critère de voisinage comprend les 4 ou les 8 voisins d'un pixel.

Il existe deux principaux algorithmes pour accomplir cette tâche :

- le premier est basé sur une procédure de suivi de contour : en parcourant le contour d'un objet et en revenant au point de départ, une composante connexe est délimitée, à l'exclusion cependant des contours intérieurs correspondant aux éventuels trous.
- le second algorithme procède par une propagation d'un étiquetage des pixels lorsque l'on effectue un balayage des lignes et des colonnes de l'image.

Nous avons élaboré un algorithme de ce type fonctionnant en une seule passe, suivant le critère de 4-connexité.

La propagation de l'étiquette des pixels suivant les colonnes (verticalement) est prioritaire sur la propagation suivant les lignes (horizontalement). On procède de la manière suivante :

En parcourant une ligne horizontale de gauche à droite, on associe un numéro (une étiquette) à chaque pixel de telle sorte que tous les pixels voisins portent le même numéro (le numéro zéro est réservé pour un pixel "vide"). Lorsque sur cette ligne, le voisinage est interrompu, puis reprend plus loin, le numéro est incrémenté de 1. Les étiquettes sont représentées par les lettres A, B et C sur la figure (II.6.a).

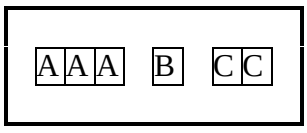


Fig. II.6.a Propagation de l'étiquetage des pixels de gauche à droite sur une ligne horizontale.

Lorsqu'une nouvelle ligne est commencée, on propage naturellement l'étiquetage de haut en bas en recopiant le numéro du pixel qui se trouve au-dessus du premier pixel de la nouvelle ligne. S'il n'y a pas de pixel au-dessus, un nouveau numéro est utilisé (fig II.6.b).

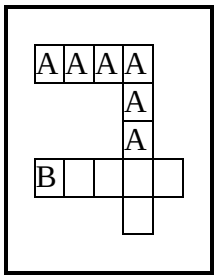


Fig. II.6.b Propagation de l'étiquetage des pixels de haut en bas sur une colonne verticale.

Lorsqu'un conflit se présente entre la propagtion horizontale et la propagation verticale des étiquettes, deux cas se présentent alors (fig. II.6.c) :

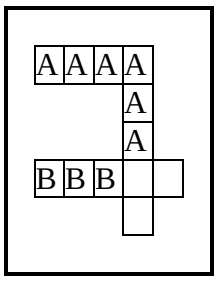


Fig. II.6.c Conflit entre la propagation horizontale et verticale.

1er cas : si le numéro des pixels horizontaux correspond à une nouvelle étiquette, alors il est facile de résoudre le conflit en remplaçant la nouvelle étiquette de tous les pixels à gauche du point de conflit par le numéro prioritaire du pixel de la ligne précédente (figII.6.d). La nouvelle étiquette est alors annulée;

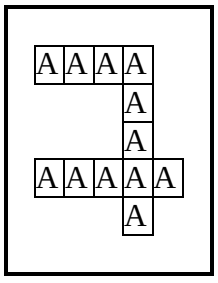


Fig. II.6.d Résolution du conflit correspondant au 1e cas.

2e cas : Si le numéro des pixels horizontaux a déjà été propagé à partir de la ligne précédente, il serait trop long de remplacer tous les pixels correspondants. Aussi, on note

dans un tableau que les deux étiquettes en conflit désignent une unique composante connexe (fig. II.6.e).

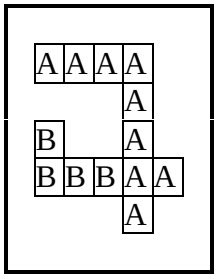


Fig. II.6.e Résolution du conflit correspondant au 2e cas.

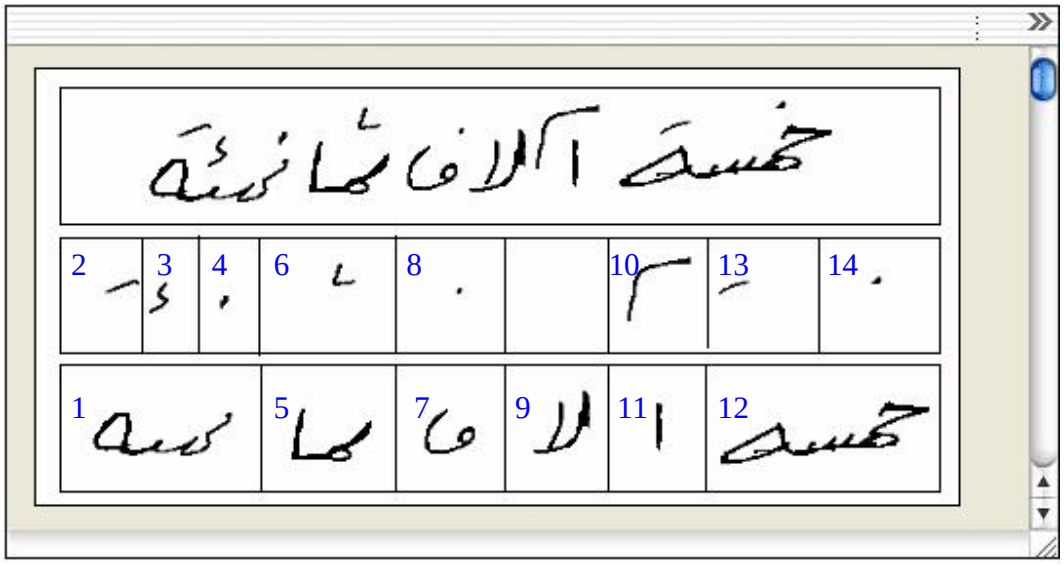


Fig. II.7 Détection des composants connexes.

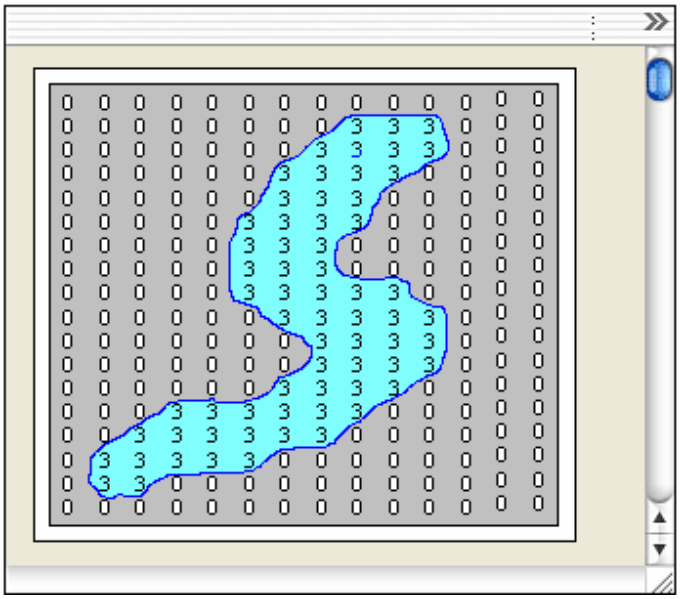


Fig. II.8 Labelage de composante 3.

7. CONTOUR

L'exploitation du contour est fréquente, tant en reconnaissance de l'écriture qu'en RdF en général, étant donné la facilité de l'extraction.

Le contour est également utilisé comme étape préalable à un changement de représentation de l'information, en tant qu'empreinte des formes contenant une quantité réduite de données.

Extraction du contour

Dans les images à niveaux de gris, il est intéressant d'extraire le contour à l'aide d'un calcul de gradient. Ce contour est alors d'autant plus marqué que le niveau des pixels résultant du gradient est élevé. En revanche, dans les images binaires, il est plus avantageux d'utiliser un algorithme de suivi de contour car il fournit directement une liste ordonnée de points.

7.1. Le gradient

Le contour se manifeste dans l'image par des variations locales importantes des valeurs de niveaux de gris mis en évidence par des élévations de la dérivée première de la fonction image. [12]

Le gradient d'une fonction $f(x,y)$ est défini par :

$$\nabla f(x,y) = \frac{\partial f}{\partial x}\vec{i} + \frac{\partial f}{\partial y}\vec{j} \qquad \text{(II.1)}$$

$\vec{i}$ et $\vec{j}$ sont les vecteurs unitaires sur x et y.

Dans le domaine discret les dérivées suivant x et y peuvent être exprimées par les approximations suivantes :

$$Ax = f(x+1,y) \quad f(x,y) \quad \textit{suivant} \quad x \qquad \text{(II.2)}$$

$$Ay = f(x,y+1) \quad f(x,y) \quad \textit{suivant} \quad y \qquad \text{(II.3)}$$

L'amplitude du gradient est donc :

$$A = \sqrt{A_x^2 + A_y^2} \qquad \text{(II.4)}$$

Elle peut être exprimée autrement par l'une des deux normes suivantes:

$$A_1 = MAX\left(|A_x|, |A_y|\right) \qquad \textit{(II.5)}$$

$$A_2 = |A_x| + |A_y| \qquad \textit{(II.6)}$$

Les approximations citées plus haut sont très sensibles au bruit, c'est pour cela que d'autres approximations ont été proposèes. Dans notre cas on a utilisé les approximations suivantes :

$$A_x = f(x+1,y+1) + f(x+1.y) + f(x+1,y \quad 1) \quad f(x \quad 1,y+1)$$
$$f(x \quad 1,y) \quad f(x \quad 1,y \quad 1) \qquad \text{(II.7)}$$

$$A_y = f(x \quad 1,y \quad 1) + f(x.y \quad 1) + f(x+1,y \quad 1) \quad f(x \quad 1,y+1)$$
$$f(x,y+1) \quad f(x+1,y+1) \qquad \text{(II.8)}$$

L'opérateur associé à ces deux approximations est celui de *Prewitt*. Défini par les deux masques suivants :

-1	0	1
-1	0	1
-1	0	1

-1	-1	-1
0	0	0
1	1	1

L'image est donc convoluée avec ces deux opérateurs, et le pixel est remplacé par l'amplitude calculée avec l'une des équations (II.5) ou (II.6).

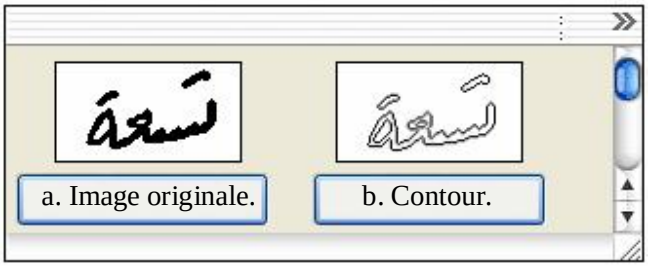


Fig. II.9. Exemple d'extraction de contour.

7.2. Suivi de contour [11]

C'est une étape nécessaire, car en parcourant le contour du noyau on élimine des points tout en conservant une épaisseur 1.

Pour suivre le contour, nous avons utilisé la méthode du "Robot qui avance dans un labyrinthe en conservant toujours un mur à sa droite" : il ne voit qu'une partie très réduite de son environnement, pourtant il peut découvrir systématiquement la totalité de son parcours accessible.

Pour réaliser cela, il est seulement nécessaire de connaître la configuration de 2 pixels notés pix1 et pix2, voisins de la position courante d'un pixel-curseur appartenant au contour.

Pour amorcer l'algorithme, il faut rechercher un premier pixel de la forme en dessous d'un pixel du fond, et qui n'appartienne pas à un contour déjà examiné : ensuite, on progresse en fonction des combinaisons des 2 pixels ; quatre cas sont à examiner (fig. II.10).

1er cas : pix1 et pix2 vides simultanément : (A1)

- on arrive donc à un bord (B1), on tourne dans le sens + pour arriver dans la nouvelle position : (C1) ;

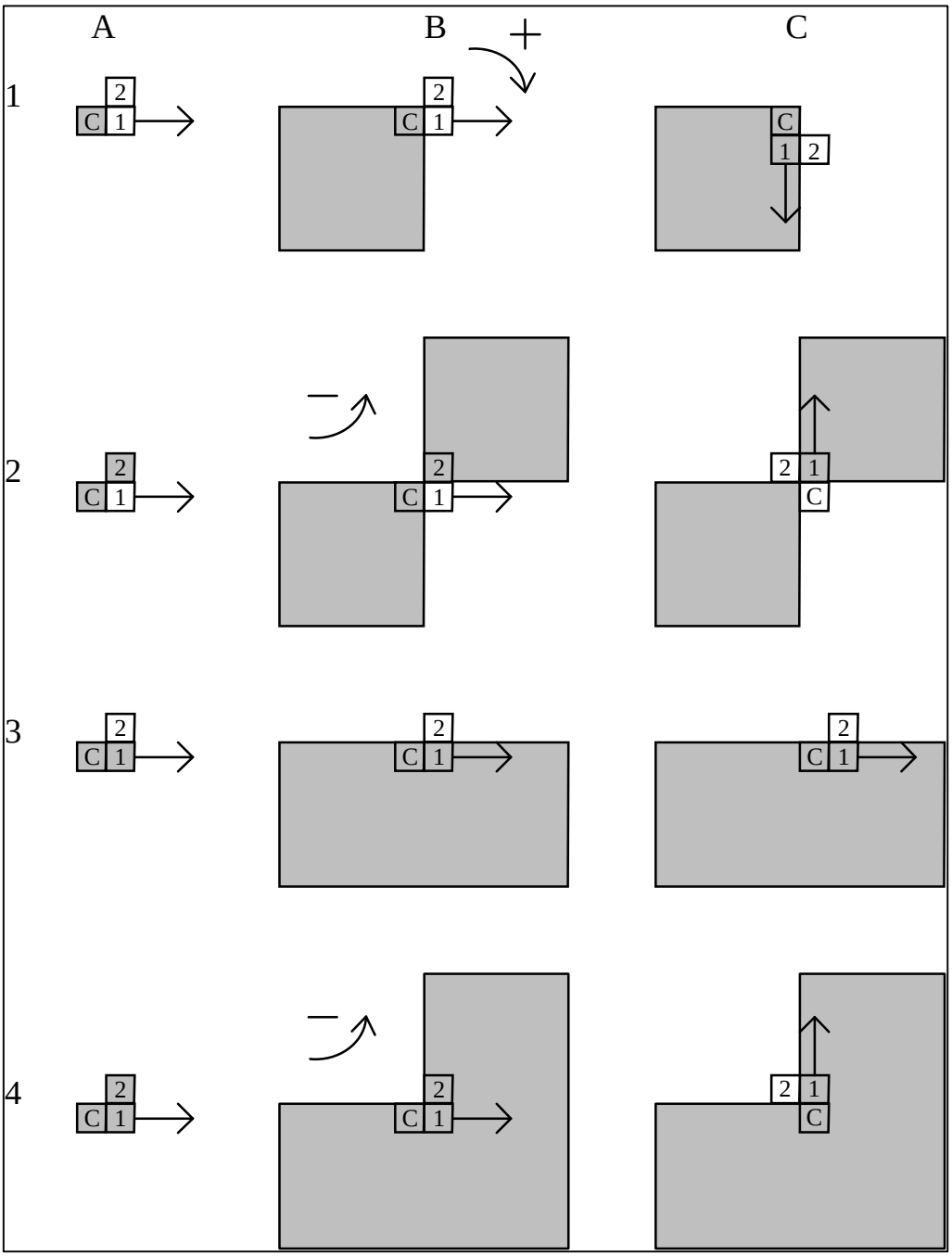


Fig. II.10. Suivi de contour.

2ème cas : pix1 vide et pix2 plein : (A2)

- on marque le pixel courant comme appartenant au contour (B2) ;
- on avance d'une position dans le sens de la flèche ;
- puis on tourne dans le sens rétrograde pour arriver dans la position (C2) ;

3ème cas : pix1 plein et pix2 vide : (A3)

- on marque le pixel courant comme appartenant au contour (B3) ;
- on avance d'une position dans le sens de la flèche (C3) ;

4ème cas : pix1 et pix2 pleins simultanément : (A4)

- on marque le pixel courant comme appartenant au contour (B4) ;
- on avance d'une position dans le sens de la flèche ;
- puis on tourne dans le sens rétrograde pour arriver dans la position (C4).

Au fur et à mesure que le contour est parcouru, on stocke la valeur 2 dans les pixels "contour" pour indiquer qu'ils ont été examinés, les autres restent notés 1 ou 0.

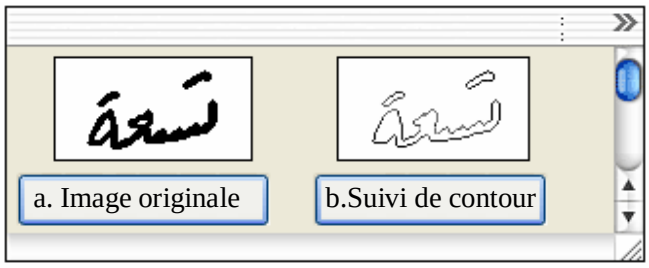


Fig. II.11. Exemple d'extraction de contour extérieur.

Nous préciserons maintenant quelles sont les différentes représentations possibles auxquelles on peut aboutir à partir du contour.

Ø La vectorisation

Le changement de représentation le plus simple est la vectorisation : il permet d'obtenir l'invariance par translation, c'est-à-dire, l'invariance à la position de la forme. Pour cela la quantité d'information contenue dans la liste des pixels du contour est avantageusement réduite à l'aide du code de *Freeman* en une liste de vecteurs unitaires. Le principe est de mémoriser seulement la position de chaque pixel par rapport au pixel précédent (4 ou 8

directions en fonction de la continuité du contour) au lieu des deux coordonnées de tous les pixels dans le plan.

- **La transformation globale du contour**

A partir de cette liste de vecteurs, l'objectif de la transformation globale du contour est d'intégrer à la reconnaissance, des propriétés d'invariance à certaines déformations ou transformations que peut subir la forme, telles que :

- la rotation,
- le changement de taille ou d'échelle,
- une transformation non linéaire (torsion, distorsion, ...).

Les principales transformations globales du contour sont les suivantes :

- descripteurs de Fourier.
- calcul des moments.
- codage du contour.

- **Autres représentations obtenues à partir du contour**

L'extraction du contour peut servir d'étape préliminaire aux deux autres changements de représentation que sont l'extraction des composantes connexes, et la construction du squelette.

8. SQUELETTISATION

La squelettisation est une opération souvent utilisée en représentation de l'écriture manuscrite, car l'écriture est assimilable à un long ruban replié sur lui-même et d'épaisseur constante. Le modèle à l'origine de la production de l'écriture est un modèle sans épaisseur, de sorte que la représentation par le squelette est naturelle et d'aspect proche de l'écriture. En ce sens, elle est proche de la représentation "en-ligne", sauf en ce qui concerne l'ordre temporel du tracé ainsi que les "barbules" qui ne sont pas significatives du tracé.

Ø Extraction du squelette [1]

L'opération de squelettisation consiste donc, dans le cas particulier de l'écriture, à éliminer l'épaisseur du trait ou plutôt à l'amincir jusqu'à l'épaisseur minimale d'un pixel.
Les critères retenus pour les méthodes de squelettisation sont les suivants :

- l'épaisseur de squelette doit être de 1 pixel ;
- le squelette doit conserver les propriétés topologiques de la forme comme le nombre de trous et la connexité ;
- le squelette doit respecter les propriétés métriques de la forme comme la longueur totale et la distance entre partie de la forme.

Ø L'amincissement

Les termes érosion ou éclaircissage ont un sens assez semblable à celui d'amincissement employé en morphologie mathématique et qui est contenu dans le terme anglais "thinning". Cette technique consiste à appliquer un élément structurant d'une transformation de tout ou rien dans laquelle le pixel central noir est remplacé par un pixel blanc en fonction de la configuration des pixels voisins. Il existe trois stratégies sur la manière d'appliquer ce masque à l'ensemble des pixels de l'image. Dans la première, un balayage horizontal et vertical de toute l'image est effectué en plusieurs passes jusqu'à ce qu'il ne reste plus de pixels à éroder, ou encore en une seule passe avec une érosion directe des pixels à chaque translation du masque. La troisième stratégie consiste en un suivi de contour appliqué successivement d'une manière analogue à une érosion "naturelle".

A l'intérieur de chacune de ces stratégies, on rencontre de nombreuses variantes qui conduisent à des améliorations portant sur les imperfections classiques du squelette.

Ø Les algorithmes à critères topologiques

Ils sont aussi appelés algorithmes de pelage. Ce sont des algorithmes itératifs supprimant à chaque étape, le long de la frontière de la forme, les points appelés inessentiels (c'est-à-dire n'appartenant pas au squelette). Ils utilisent souvent des fonctions booléennes opérant sur des voisinages de points, déterminant à chaque passage la validité des points frontières.

L'algorithme général

```
début
  répéter
      inessentiel ← faux
      pour tout point dans image faire
          si POINTINESSENTIEL (point) alors
                inessentiel ← vrai
                image (point) ← 0
          fsi
      fpour
  jusqu'à inessentiel=faux
 fin
```

Pour déterminer si un point est inessentiel, *Serra* utilise les masques suivants :

X	0	0
1	1	0
X	1	X

0	0	0
X	1	X
1	1	1

L'ensemble des voisinages 3×3 d'un point inessentiel est obtenu à partir de ces configurations par rotation de 90°.

L'algorithme général est de type séquentiel. Il consiste à balayer l'image ligne par ligne et à supprimer les points inessentiels au fur et à mesure qu'ils sont rencontrés. Il ne peut donc pas tenir compte de l'environnement dans lequel il opère et a tendance à enlever trop de points d'un seul coup.

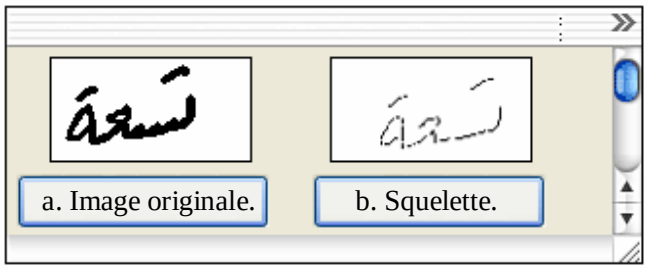


Fig. II.12. Exemple du mot avant et après squelettisation.

9. SEGMENTATION

Après avoir établi un bilan de principales méthodes dont nous disposions pour le prétraitement, il nous faut en venir à l'obstacle auquel se sont affrontés tous les chercheurs : la segmentation. Segmentation des lignes en mots, segmentation des mots en lettres.

La segmentation est une étape critique et décisive dans plusieurs systèmes de reconnaissance. En d'autres termes, l'efficacité d'un système de reconnaissance en dépend fortement.

Elle est définie, comme étant l'opération qui cherche à décomposer une image de texte en pseudo-images de symboles individuels. Le résultat de cette opération, est une forme isolée à partir d'une image et qui pourrait être un caractère ou non.

Cette opération est l'étape la plus critique dans plusieurs systèmes de reconnaissance d'écriture, elle est dans la plupart des cas sujette aux erreurs et très pénalisante en temps de calcul.

Suivant les applications envisagées et les objectifs à atteindre, on peut distinguer plusieurs niveaux de segmentation. A titre d'exemple, dans le traitement automatique de document, consiste à isoler les différentes parties logiques (graphiques, textes, photos, etc.) constituant une page de document. En revanche, dans les applications où les informations traitées sont des pages de texte, la séparation des lignes de paragraphes, l'extraction des mots ou pseudo-mots d'une ligne, et enfin, la décomposition des mots ou pseudo-mots en caractères.

9.1. La segmentation des mots [7]

Beaucoup d'efforts ont été consacrés à la reconnaissance des lettres successives constituant les mots. Séparer les lettres constitue une opération délicate et coûteuse, tant les écritures sont variées, et les lettres souvent liées les unes aux autres. Une fois cette opération faite, il reste à reconnaître chaque lettre, ce qui est également difficile. Cette voie trop analytique ne semble donc pas très encourageante. Mais les humains ne lisent pas lettre à lettre en général, sauf en cas d'écriture vraiment très mal formée ou quand un mot est inconnu : dans la très grande majorité des cas, ce sont les mots qui sont reconnus

en tant que tels, sans qu'il soit besoin de les épeler. Cette façon de procéder est évidemment plus efficace.

Une des étapes du traitement automatique consiste donc à segmenter les mots, c'est à dire à séparer les images de ceux-ci à partir de l'image globale d'une phrase manuscrite. C'est au départ une opération seulement morphologique

Les exemples d'écriture que nous avons choisis sont extraits de chèques bancaires, dont un assortiment est montré sur la figure (II.13).

Notre tâche consistera à trouver des méthodes aussi simples et robustes que possible pour

- isoler chaque ligne dans le cas où il y en a plus d'une ;
- isoler chaque mot par un rectangle enveloppant minimal.

9.2. Localisation des lignes d'écriture

Pour localiser les lignes de texte il est possible de s'appuyer sur un modèle physique de disposition de l'écriture. La complexité de cette étape est très variable. En effet, on peut supposer en général que les lignes d'écriture sont plus ou moins parallèles et horizontales et mettre en œuvre alors des méthodes de détection relativement simples et satisfaisantes. Toutefois les problèmes ne sont absolument pas résolus dans le cas de documents manuscrits présentant des dispositions variables, des ratures, des inclinaisons ou des chevauchements de lignes. Cependant, quelles que soient les approches envisagées, des problèmes subsistent notamment lorsque les lignes d'écriture se superposent partiellement en présence d'extensions hautes ou basses qui s'étendent jusque sur la ligne d'écriture inférieur ou supérieure. Le choix des points de coupure reste alors un problème difficile à résoudre sans faire appel au module de reconnaissance.

La méthode triviale de séparation de lignes fait appel à la projection horizontale qui n'est rien d'autre qu'une simple somme du nombre de points allumés par ligne.

On peut dire le début ou la fin d'une ligne de texte est détectée, si la valeur de projection horizontale figure (II.13) est inférieure à un seuil.
Le seuil est obtenu au minimum de l'histogramme.

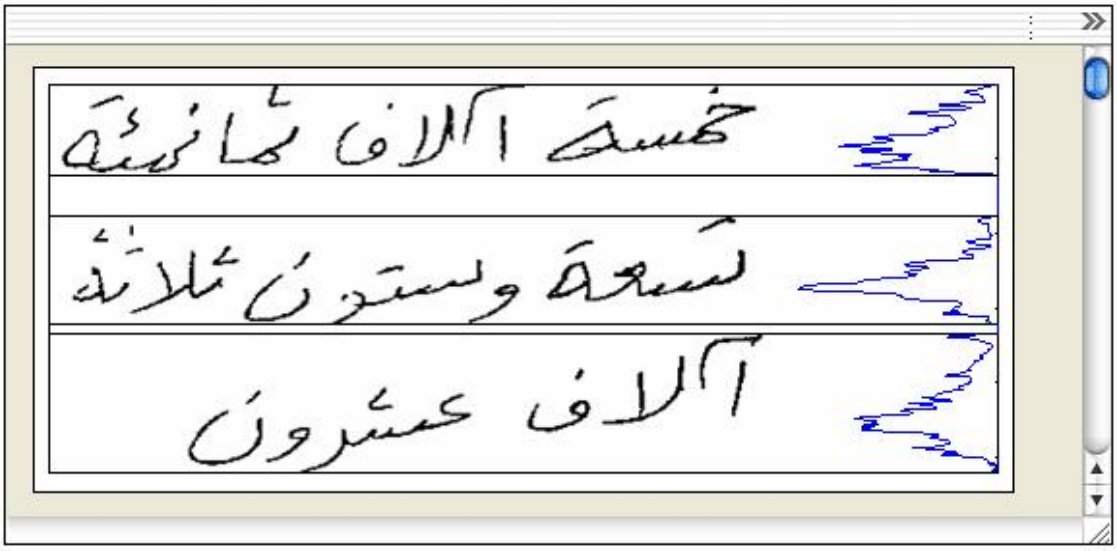


Fig. II.13. Exemple de localisation des lignes d'écriture.

9.3. La segmentation des mots

La localisation des mots dans les textes manuscrits est un problème qui relève du paradoxe de *Sayre* (1973) « Il faut segmenter le tracé pour reconnaître une lettre, mais il faut reconnaître une lettre pour segmenter le tracé ». En effet, si théoriquement les règles de disposition de l'écriture imposent de marquer des espaces entre les mots plus importants (lue les espaces entre caractères ce qui est le cas pour l'imprimé), en pratique ces règles ne sont pas toujours vérifiées sur les écritures manuscrites non contraintes. Et on doit admettre que pour résoudre le problème, il est nécessaire de demander l'aide d'un système de reconnaissance.

De ce fait la localisation des mots dans une ligne de texte est un problème qui s'apparente à celui de la reconnaissance des caractères dans les mots.

On peut classer les approches proposées dans la littérature en deux grandes familles:

- La première concerne les approches qui recourent à une métrique spécifique adaptée au problème afin d'ordonner de la meilleur façon possible les espaces détectés dans les lignes pour qu'une simple technique de seuillage puisse permettre de séparer les espaces inter-mots des espaces inter-caractères.

- La seconde met en œuvre une étape de pré-connaissance pour attribuer les espaces détectés dans la phrase à l'une des deux classes, espace inter-mots, espace inter-lettres.

La méthode triviale de séparation de lignes fait appel à la projection verticale qui n'est rien d'autre qu'une simple somme du nombre de points allumés par colonne comme l'indique la figure (II.14).

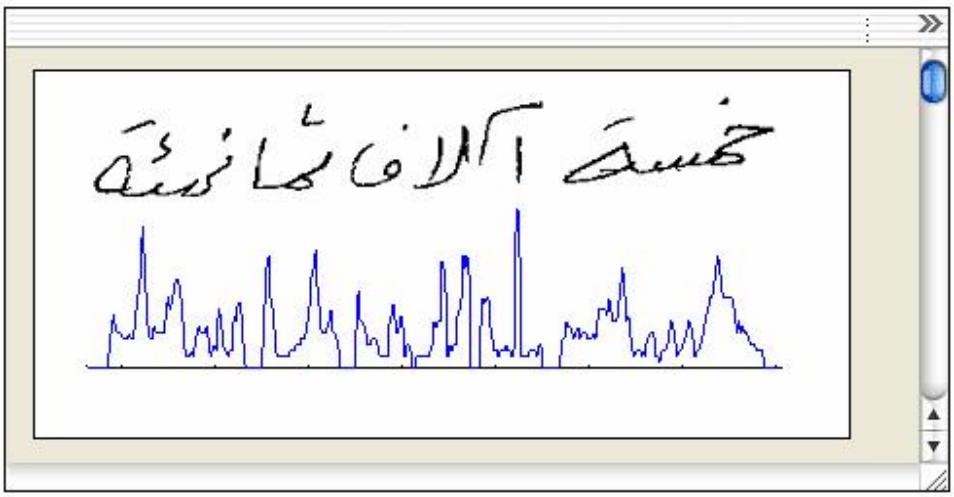


Fig. II.14. Histogramme de la segmentation mots.

9.4. Localisation des lignes de référence

D'un point de vue local, on peut résumer les règles de disposition des caractères entre eux par la connaissance de trois zones d'écriture distinctes (voir Figure (II.15)) la zone de corps des minuscules (2), la zone des extensions hautes (1) et la zone des extensions basses (3). De plus, ces zones sont normalement horizontales et identiques sur toute une ligne d'écriture.

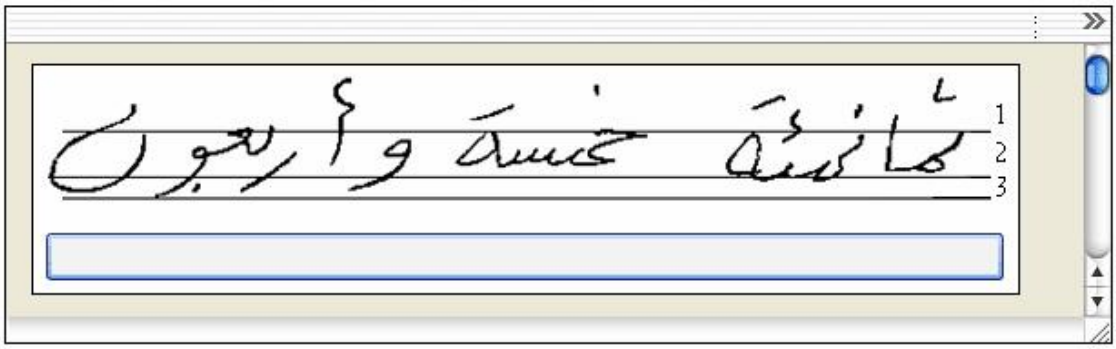


Fig. II.15. Les différentes zones d'écritures.

Dans certaines situations ces règles de disposition ne sont pas toujours respectées en pratique. La zone du corps des minuscules peut présenter des variations importantes de hauteur à l'intérieur d'un même mot.

L'ensemble de ces déformations introduit une variabilité importantes dans la disposition du message écrit qu'il est souhaitable de prendre en compte et de corriger afin d'une part, de ne pas dégrader le comportement des processus de pré-traitement qui suivront (essentiellement la segmentation du mot), et d'autre part de rendre le plus possible invariante la description du mot et des formes qui le composent, cela toujours dans un but de reconnaissance.

Pour cela, la localisation des lignes de référence peut s'accompagner d'une étape de normalisation qui implique le redressement de l'écriture et éventuellement le redressement des caractères qui peuvent également présenter une certaine inclinaison par rapport aux lignes de référence. D'autres méthodes cherchent à localiser les lignes de référence locales qui peuvent être inclinées par morceaux en exploitant l'information des minima ou des maxima du tracé.

Notons également que dans le cas où les écritures ne présenteraient pas autant de variabilité au niveau de la disposition spatiale des entités connexes, il est possible de baser la détection des lignes de référence sur l'histogramme des projections horizontales des lignes de texte.

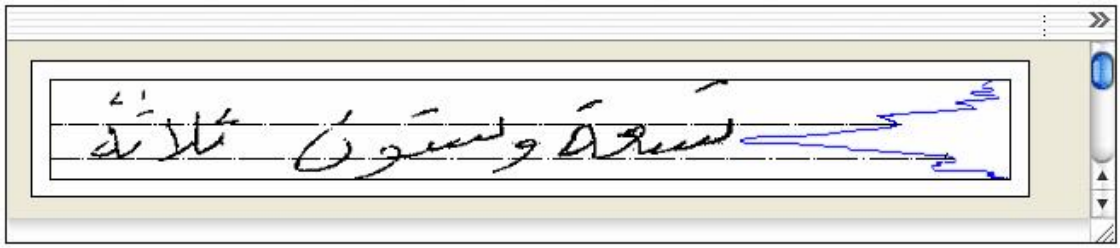


Fig. II.16. Exemple de localisation des lignes de référence.

10. CONCLUSION

Nous avons vu dans ce chapitre les premières transformations nécessaires au traitement des formes. Elles montrent les différentes aberrations du système d'acquisition contre lesquelles il faut se prémunir avant d'entreprendre toute action d'analyse. Comme l'écriture arabe est par nature cursive, l'un des grands obstacles rencontrés dans un effort pour améliorer la reconnaissance d'écriture arabe manuscrite est le manque de solutions effectives et efficaces au problème de segmentation. Les recherches sur la segmentation en particulier, et sur la reconnaissance en général, doivent se concentrer sur le traitement des textes dégradés, inclinés et les textes ayant des caractères qui se chevauchent.

CHAPITRE IV
CLASSIFICATION NEURONALE ET STRUCTURELLE

1. INTRODUCTION

Récemment, la combinaison de classifieurs a été proposée comme une voie de recherche permettant de fiabiliser la reconnaissance en utilisant la complémentarité qui peut exister entre les classifieurs. Sur ce point, la littérature abonde de travaux présentant des méthodes de combinaison qui se différencient aussi bien par le type d'informations apportées par chaque classifieur que par leurs capacités d'apprentissage et d'adaptation.

Alors que les premières expériences en combinaison de classifieurs ne datent que des années 80, cette technique est devenue une voie de plus en plus utilisée pour améliorer la qualité des systèmes de reconnaissance dans plusieurs applications : reconnaissance d'images midicales, reconnaissance de chiffres, de caractères et de mots manuscrits, de visages, vérification de signatures, reconnaissance de la parole, etc. Ces systèmes diffèrent par leur type de sorties des classifieurs combinés, par la nature des classifieurs utilisés ainsi que par les stratégies de combinaison choisies.

L'objectif de ce chapitre est de présenter les deux classifieurs adoptés dans ce travail à savoir les réseaux de neurones et les méthodes structurelles.

2. LES RESEAUX DE NEURONES

2.1. Introduction

Le souci d'améliorer les performances des systèmes de reconnaissance est une raison déterminante pour l'introduction de nouvelles stratégies de classification dans les techniques de reconnaissance d'écriture. La reconnaissance du fait que le cerveau fonctionne de manière entièrement différente de celle d'un ordinateur conventionnel a joué un rôle très important dans le développement des réseaux de neurones artificiels.

Nous abordons les réseaux de neurones dans la seule perspective de faire de la reconnaissance des formes et de la classification. Notre objectif n'est donc pas celui des neuroscience consistant à chercher à utiliser la puissance des ordinateurs pour simuler l'intelligence humaine ou animale.

2.2. Historique

Les premiers travaux sur les neurones artificiels ont débuté au début des années 1940 et ont été menés par McCulloch et Pitts. Ils décrivent les propriétés du système nerveux à partir de neurones idéalisés : ce sont des neurones logiques (0 ou 1).

Dix années plus tard, on a constitué le premier modèle réel d'un réseau de neurones. En 1960, le premier perceptron est créé par Rosenblatt. Puis, durant les années 1970 il y eut une remise en cause de l'intérêt des réseaux car les ordinateurs de neurones apprenaient lentement, coûtaient très cher et leurs performances n'étaient pas si impressionnantes. La disponibilité croissante des minis et microordinateurs, vers la fin des années 1970, a permis aux réseaux de neurones de rendre un nouveau départ. On attribue à Hopfield (un physicien de Caltech) un rôle majeur dans cette résurrection.

2.3. Base biologique

Le neurone biologique (figure IV.1) est une cellule vivante spécialisée dans le traitement des signaux électriques.

Les neurones sont reliés entre eux par des liaisons appelées axones. Ces axones vont eux-mêmes jouer un rôle important dans le comportement logique de l'ensemble. Ils conduisent les signaux électriques de la sortie d'un neurone vers l'entrée (synapse) d'un autre neurone.

Les neurones font une sommation des signaux reçus en entrée et en fonction du résultat obtenu vont fournir un courant en sortie.

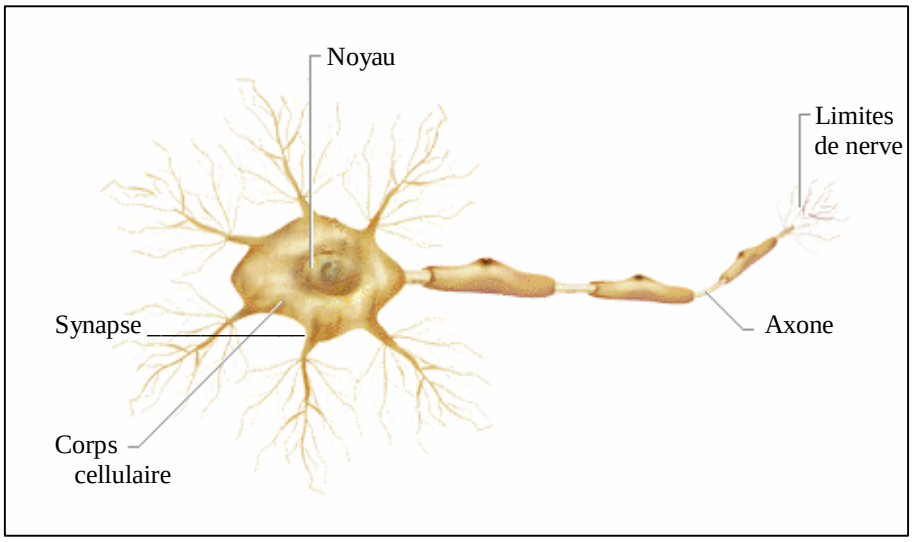


Fig. IV.1. Neurone biologique.

2.4. Neurone artificie

La première étude systématique du neurone artificiel est due au neuropsychiatre McCulloch et au logicien Pitts [Lippmann, 87] qui, s'inspirant de leurs travaux sur les neurones biologiques. Proposèrent en 1943 le modèle illustré à la figure IV.2

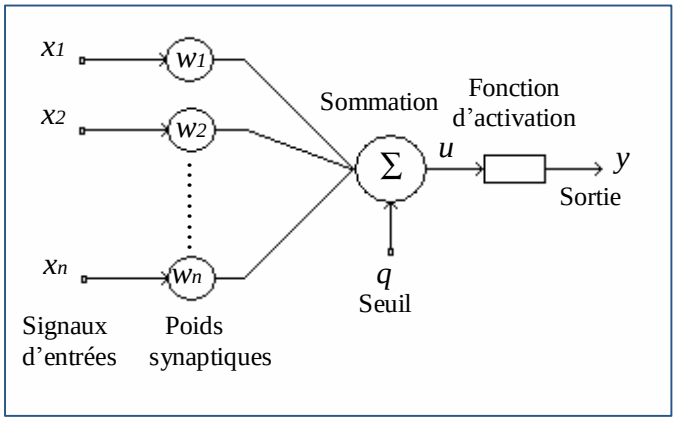


Fig. IV.2. Neurone artificiel.

Ce neurone formel est un processeur élémentaire qui réalise une somme pondérée des signaux qui lui parviennent. La valeur de cette sommation est ensuite comparée à un seuil, et la sortie du neurone est :

$$u = \sum_{j=1}^{n} w_j x_j - q \qquad \text{.....IV.1}$$

$$y = f(u) \qquad \text{.....IV.2}$$

Tel que :

$j = 1 \ldots n$: les entrées du neurone formel.

y : sa sortie.

w_j : les paramètres de pondération.

f : la fonction d'activation.

Dans le modèle original de McCulloch et Pitts, la non linéarité était assurée par la fonction seuil de Heaviside (figure IV.3), définie par :

$$H(u) \overset{\Delta}{=} \begin{cases} 1 & \text{si } u > 0 \\ 0 & \text{sinon} \end{cases}$$

En place de la fonction de Heaviside, il est également possible de choisir la fonction de signe **(**figure IV.4), définie comme suit :

$$Sgn(u) \overset{\Delta}{=} \begin{cases} 1 & \text{si } u > 0 \\ -1 & \text{sinon} \end{cases}$$

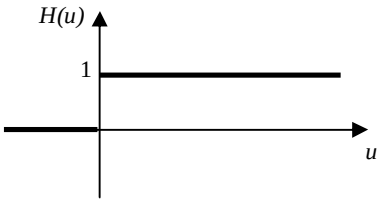


Fig. IV.3. La fonction d'Heaviside.

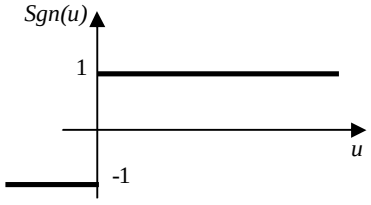


Fig. IV.4. La fonction signe.

2.5. Structure d'interconnexion

Les connexions entre les neurones qui composent le réseau décrivent la « topologie » du modèle. Le plus souvent, cette topologie fait apparaître une certaine régularité de l'arrangement des neurones ; cependant, celui-ci peut être quelconque.

2.5.1. Réseau multicouche

Les neurones sont arrangés par couche (figure IV.5). On place ensuite bout à bout plusieurs couches et l'on connecte les neurones de deux couches adjacentes. Les entrées des neurones de la deuxième couche sont en fait les sorties des neurones de la couche amont.

Les neurones de la première couche sont reliés au monde extérieur et reçoivent le vecteur d'entrée. Ils calculent alors leurs sorties qui sont transmises aux neurones de la seconde couche qui calculent eux aussi leurs sorties et ainsi de suite de couche en couche jusqu'à celle de sortie. Il peut y avoir une ou plusieurs sorties à un réseau de neurones.

Dans un réseau multicouche classique, il n'y a pas de connexion entre neurones d'une même couche et les connexions ne se font qu'avec les neurones de la couche aval. Tous les neurones de la couche amont sont connectés à tous les neurones de la couche aval.

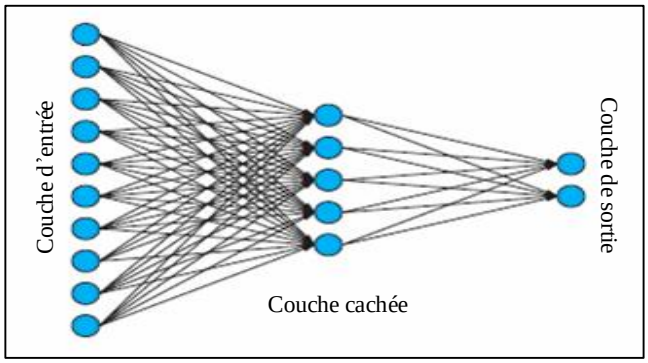


Fig. IV.5. Réseau multicouche classique.

Les couches extérieures du réseau sont appelées respectivement couches d'entrée et de sortie ; les couches intermédiaires sont appelées couches cachées.

2.5.2. Réseau à connexions locales

C'est aussi un réseau multicouche, mais tous les neurones d'une couche amont ne sont pas connectés à tous les neurones de la couche aval (figure IV.6). Nous avons donc dans ce type de réseau de neurones un nombre de connexions moins important que dans le cas du réseau de neurones multicouche classique.

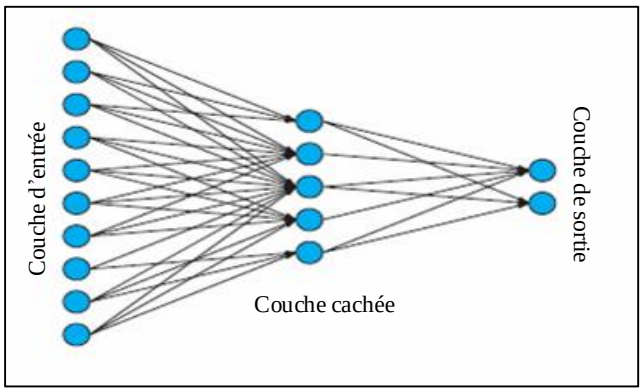


Fig. IV.6. Réseau multicouche à connexions locales.

2.5.3. Réseau à connexions récurrentes

Un réseau de ce type signifie qu'une ou plusieurs sorties de neurones d'une couche aval sont connectées aux entrées des neurones de la couche amont ou de la même couche. Ces connexions récurrentes ramènent l'information en arrière par rapport au sens de propagation défini dans un réseau multicouche (figure IV.7).

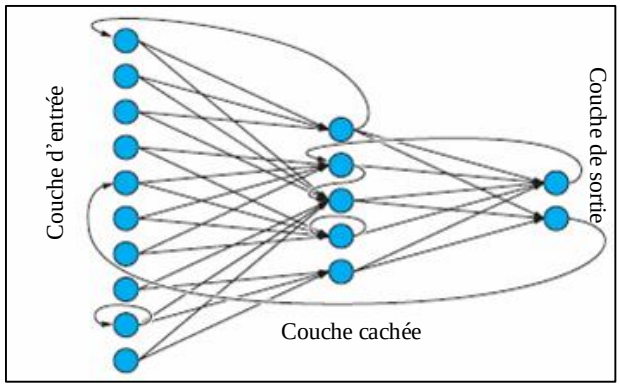


Fig. IV.7. Réseau à connexions récurrentes.

Les réseaux à connexions récurrentes sont des réseaux plus puissants car ils sont séquentiels plutôt que combinatoires comme l'étaient ceux décrits précédemment. La rétroaction de la sortie vers l'entrée permet à un réseau de ce type de présenter un comportement temporel.

2.5.4. Réseau à connexions complexes

Chaque neurone est connecté à tous les neurones du réseau y compris lui-même, c'est la structure d'interconnexion la plus générale (figure IV.8).

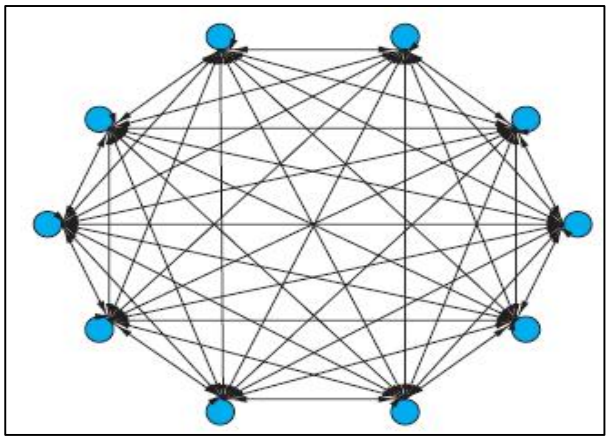


Fig. IV.8. Réseau à connexions complexes.

2.6. Les réseaux les plus célèbres

Il y a de très nombreuses sortes de réseaux de neurones actuellement. Personne ne sait exactement combien. De nouveau (ou du moins des variations de réseaux plus anciens) sont inventés chaque semaine. On en présente ici de très classiques. [12]

2.6.1. Le perceptron (F. Rosenblatt, 1958)

C'est histotiquement le premier réseau de neurone artificiel. C'est un réseau de neurones très simple avec deux couches de neurones acceptant seulement des valeurs d'entrées et de sorties binaires. Le procédé d'apprentissage est supervisé et le réseau est capable de résoudre des opérations logiques simples : somme, AND ou OR. Il est aussi utilisé pour la classification. Sa principale limite est qu'il ne peut résoudre des problèmes linéairement séparables.

2.6.2. Le modèle ADALINE

L'ADALINE (Adaptatif Linéaire Elément) conçu par B. Widrow dans les années 1960, est un perceptron sans couche cachée donc à un seul neurone qui reçoit le stimulus arrivant de la couche d'entrée et donne la réponse correspondante.

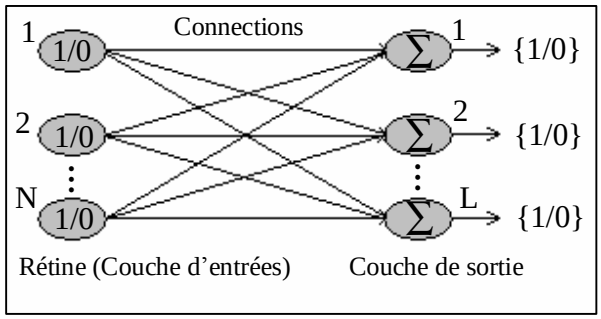


Fig. IV.9. Structure d'un perceptron.

2.6.3. Le perceptron multicouche (PMC)

C'est une amélioration du perceptron comprenant une ou plusieurs couches intermédiaires dites couche cachée. Ils utilisent, pour modifier leur poids, un algorithme de rétropropagation du gradient, qui est un généralisation de la règle de WIDROW-HOFF, les PMC agissent comme un séparateur non linéaire et peuvent être utilisés pour la classification, le traitement d'image, l'aide de la décision ou la commande d'un processus.

2.7. L'apprentissage

Le principal problème pour les réseaux de neurones est d'arriver à trouver un ensemble de valeurs les synapses (poids), qui sont les porteurs de l'information, tel que les configurations d'entée se traduisent par les réponses voulues, et cela en partant d'une valeur particulière des poids des connections, le réseau améliore ces réponses en ajustant ses coefficients selon un algorithme ou une règle d'apprentissage.

Il existe trois classes d'apprentissage : l'apprentissage supervisé, semi supervisé et non supervisé.

- **Apprentissage supervisé**

Les apprentissages supervisés demandent que l'on donne au réseau neuronal des exemples résolus, c'est-à-dire des couples de vecteurs entrée/sortie. Cet apprentissage exploite le plus

souvent plusieurs idées simples. L'idée principale est la minimisation itérative d'un critère de l'erreur en sortie du réseau. On initialise les matrices de connexion au hasard, puis l'on fait évoluer ces matrices de manières à ce qu'elles autorisent l'association souhaitée, c'est-à-dire jusqu'à ce qu'un critère de l'erreur (entre les sorties réellement obtenues et les sorties souhaitées) soit quasi nul.

- **Apprentissage semi-supervisé**

Dans ce type d'apprentissage l'utilisateur possède seulement des indications imprécises sur le comportement final du réseau, mais en revanche, il possible d'obtenir des indications qualitatives (correctes/ incorrecte).

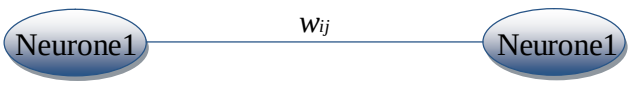
- **Apprentissage non- supervisé (auto- organisation)**

Dans ce type d'apprentissage (sans professeur) les poids synaptiques du réseau sont modifiés selon des critères internes comme la co-activation des neurones. Le comportement de se type d'apprentissage est comparable à des techniques d'analyse de données. Un exemple des réseaux à apprentissage non- supervisé : les cartes topologiques de Kohonen. Enfin, certains réseaux associent les deux types d'apprentissage supervisé et non- supervisée, c'est le cas par exemple du réseau de Boltzmann.

2.8. Règles d'apprentissages

2.8.1. La règle de Hebb

Dans le domaine de la recherche sur le fonctionnement des neurones biologiques et sur les mécanismes d'apprentissage de l'intelligence humaine, Hebb a proposé un type de réseau de neurones totalement interconnecté (c'est-à-dire ou les neurones sont reliés par des connexions de types synapse fonctionnant à la fois en « entrée » et en « sortie ». Ces connexions sont affectées de poids qui évoluent au cours du temps et en fonction de l'activation de chacun des deux neurones extrémités de cette connexion. Il a ainsi défini une règle d'apprentissage dite de Hebb :



Cette règle considère alors que toute connexion entre deux neurones se renforce si ces deux neurones sont actifs au même moment. Si on note A1 et A2 l'activation des neurones 1 et 2 et si on suppose qu'un neurone actif à son activation qui vaut 1et qu'un neurone inactif a son activation qui vaut 0 alors l'expression de la règle de Hebb est la suivante :

$$w_{ij}(t+dt) = w_{ij}(t) + m \times A_i \times A_j \quad \text{avec } m \ \mathbf{f}\ 0 \qquad \text{.....IV.3}$$

Au départ on a $w_{ij}(0) = 0$ pour tout i, j.

Ce type de réseaux a surtout une vocation pour la modélisation des neurones biologiques et reste peu utilisé en reconnaissance des formes. Un des inconvénients majeurs de ce modèle vient du fait que les w_{ij} ne peuvent qu'augmenter au cours du temps.

2.8.2. La règle de Widrow-Hoff

La règle de Widrow-Hoff ou règle delta proposée en 1960, consiste à modifier chaque pas, les poids et les biais afin de minimiser la somme des carrées des erreurs en sortie en utilisant la règle suivante :

$$w_{ij}(k+1) = w(k) + h(t_k - y_k)x_k^T \quad \text{.....IV.4}$$

A chaque pas d'apprentissage *k*, l'erreur en sortie est calculée comme la différence entre la cible recherchée *t* et la sortie *y* du réseaux

$$E_k = e_k^T e_k = (t_k - y_k)^T (t - y) = \frac{1}{2}(t_k^T t_k + y_k^T y_k - 2y_k^T t_k) \text{.....IV.5}$$

Le gradient se calcule comme suite :

$$\nabla E_{kw} = \frac{1}{2}\nabla\left[y_k^T y_k - 2y_k^T t_k\right]_w \quad \text{.....IV.6}$$

$$\nabla E_{kw} = \frac{\partial E_k}{\partial w} = \frac{\partial E_k}{\partial y_k}\frac{\partial y_k}{\partial w} \quad \text{.....IV.7}$$

D'après l'expression de E_k et avec $y_k = wx_k + b$ les dérivées partielles sont :

$$\frac{\partial E_k}{\partial w} = (y_k - t_k)$$

$$\frac{\partial (wx_k + b)}{\partial w} = x_k$$

La mise à jour des poids se fait par l'équation :

$$w_{ij}(k+1) = w(k) + h\nabla E_{kw} \quad \text{.....IV.8}$$

Avec h : le gain d'apprentissage, 0‹ η ‹ 1

De même on obtient l'expression de la modification du biais :

$$b(k+1) = b(k) + h(t_k - y_k) = b(k) + h\nabla E_{kb} \quad \text{..... IV.9}$$

2.8.3. Apprentissage du perceptron multicouches

L'apprentissage du perceptron multicouches est supervisé et consiste à adapter les poids des neurones de manière à ce que le réseau soit capable de réaliser une transformation données, représentée par un ensemble d'exemples constitue d'une suite de N vecteurs d'entrées.
Donc l'apprentissage fonctionne sur le même principe que la règle de Widrow-Hoff ; on dispose d'un ensemble d'exemples qui sont des couples (entrées-sorties désirées). A chaque étape un exemple est présenté en entrée du réseau. Une sortie réelle est calculée. Ce calcul est effectué de proche en proche de la couche d'entrée à la couche de sortie. Cette phase est appelée propagation avant ou relaxation du réseau. Ensuite l'erreur est ensuite rétro propagée dans le réseau, donnant lieu à une modification de chaque poids.

L'apprentissage du perceptron multi-couches consiste à minimiser l'erreur quadratique commise sur l'ensemble des exemples, ce problème de minimisation de l'erreur a été résolu par des méthodes de *rétro propagation du gradient d'erreur*.

La difficulté majeure rencontrée lors de l'utilisation de ce type de classifieur consiste à déterminer le nombre de couches cachées, le nombre de neurones dans chacune des couches et les poids des connexions entre les différentes couches (taux d'apprentissage…). De ce fait, pour une application donnée, la construction du classifieur de type MLP utilise des règles empiriques et nécessite un certain nombre d'essais afin d'obtenir des performances de généralisation intéressantes.

Pour déterminer les poids de toutes les connexions du réseau, l'utilisation des algorithmes d'apprentissage est obligatoire. L'objectif des algorithmes d'apprentissage est de minimiser l'erreur de décision effectuée par le RNA en ajustant les poids à chaque présentation d'un vecteur d'entraînement. Pour ce qui est de l'ajustement des poids à une étape donnée de la phase d'apprentissage, l'erreur à minimiser est habituellement celle produite lors de l'application d'un vecteur de l'ensemble d'apprentissage à l'entrée du réseau. Pour ce qui est de l'évaluation de la

qualité d'apprentissage du réseau, l'erreur cumulée par tous les vecteurs de l'ensemble d'entraînement est évaluée. Cette erreur cumulée est calculée pour tous les cycles de la phase d'entraînement et est définie à partir de l'erreur quadratique. Cette mesure de l'erreur illustre la précision obtenue après *P* cycles d'apprentissage. En résumé, nous utiliserons pour l'apprentissage du réseau l'algorithme de rétro propagation avec minimisation du gradient d'erreur qui est défini par les étapes suivantes :

1. initialiser les poids à des petites valeurs et les seuils du réseau.

2. insérer à l'entrée du réseau une observation (exemple) de la base de données en forme de vecteur de caractéristiques, puis calculer sa valeur d'activation et sa fonction d'activation en utilisant les formules (IV. 10) et (IV. 11).

3. évaluer le signal d'erreur des sorties du réseau en utilisant la formule (IV. 12).

4. ajuster les poids en utilisant la formule (IV. 10).

5. évaluer le signal d'erreur pour chaque couche cachée en utilisant la formule (IV. 13).

6. ajuster les poids de la couche cachée en utilisant la formule (IV. 11).

7. répéter les étapes 2 à 6 pour l'ensemble des observations de la base d'apprentissage tant que le critère d'arrêt n'a pas été atteint.

Il existe plusieurs critères d'arrêts. Ces critères peuvent être combinés entre eux.

Le premier critère est basé sur l'amplitude du gradient de la fonction d'activation, puisque par définition le gradient sera à zéro au minimum. L'apprentissage du réseau du type MLP utilise la technique de recherche du gradient pour déterminer les poids du réseau.
Le second critère d'arrêt est de fixer un seuil que l'erreur quadratique ne doit pas dépasser. Toutefois ceci exige une connaissance préalable de la valeur minimale de l'erreur qui n'est pas toujours disponible. Dans le domaine de la reconnaissance de formes, il suffit de s'arrêter lorsque tous les objets sont correctement classés.

Les deux premiers critères sont sensibles aux choix des paramètres (par exemple : le nombre de nœuds dans la couche cachée, le seuil d'erreur,…) et si le choix n'est pas bon alors les résultats obtenus seront mouvais ou le temps de calcul de la performance du système de

reconnaissance sera plus lent (par exemple définir un grand nombre de nœuds dans la couche caché). Cependant, la méthode du crossvalidation n'a pas ce genre de problème.

En général, les formules utilisées par cet algorithme sont :

- pour l'ajustement de poids entre le nœud *j* (sortie ou cachée) et le nœud *i* (cachée ou entrée)

$$\nabla w_{ij} = h d_j o_i \qquad \text{.....IV.10}$$

Où *h* est la valeur de la constante d'apprentissage.

En général, $0.1 < h < 0.9$

Et o_i est la valeur d'activation du neurone *i* tel que $o_i = f(net_i)$

$$net_i = \sum_j w_{ij} o_j \qquad \text{.....IV.11}$$

Avec $f(net_i)$ est la fonction d'activation. La fonction utilisée dans le cas de notre projet est la fonction sigmoïde définie :

$$f(net_i) = \frac{1}{1+\exp^{-net_i}} \qquad \text{.....IV.12}$$

- pour le calcul du signal d'erreur du neurone *j* de la couche de sortie avec *dj* est la valeur désirée du neurone *j* :

$$d_j = (d_j - o_j) o_j (1 - o_j) \qquad \text{.....IV.13}$$

Avec *dj* la valeur désirée du neurone *j*.

- pour le calcul du signal d'erreur du neurone *j* de la couche cachée avec *k* nœuds :

$$d_j = o_j (1 - o_j) \sum_k w_{jk} d_k \qquad \text{.....IV.14}$$

Ces formules ont été dérivées de la formule de calcule de l'erreur quadratique de l'ensemble de la base d'apprentissage définie comme suit :

$$E_p = \frac{1}{2} \sum_p (\sum_k (d_{pk} - o_{pk})^2) \qquad \text{.....IV.15}$$

Où *p* est l'indice d'un exemple de la base et *k* est l'indice du nœud de sortie.

L'objectif est de minimiser cette erreur.

Ø Evolution du perceptron multicouche

Les réseaux de neurones multicouches entraînés par l'algorithme de rétropropagation du gradient sont les modèles connexionnistes les plus étudiés et utilisés à ce jour. Les champs d'application de ces réseaux sont très vastes : classification, identification de processus et prédiction de séries temporelles, commande de processus et de robot, traitement d'images, et de paroles. Ce sont des modèles robustes et dont les entrées et les sorties peuvent indépendamment être choisies binaires ou réelles, ce qui permet de traiter de très vastes classes de problèmes. Cependant, lors de la réalisation d'une application sur un réseau de neurones multicouches entraîné par l'algorithme de rétro propagation, il faut prendre en compte les points suivants :

- la mise en œuvre de cet algorithme exige souvent des temps de calcul très longs qui rendent son application incommode pour de nombreux problèmes de taille raisonnable.
- La rétro- propagation peut trouver piège dans les minima locaux, et ne peut pas donner la bonne réponse. Pour remédier à ce problème, on peut entraîner le réseau à partir de plusieurs choix initiaux de poids pour ne garder que le meilleur ou bien ajouter un bruit aléatoire puis relancer l'apprentissage.
- Le réseau apprend à partir des exemples de la base d'apprentissage. Si cette dernière concerne un fonctionnement dans un domaine réduit, le réseau ne saura répandre en dehors de ce domaine. Si le superviseur donne des informations erronées, le réseau apprendra un modèle erroné et si le processus que l'on cherche à modéliser est non stationnaire ou change brutalement, le réseau réapprendra un bon modèle.

Aujourd'hui, les perceptrons multicouches sont les réseaux utilisés par les développeurs d'applications. Ces résultats théoriques sur le mécanisme de comportement de ces réseaux sont encore très pauvres, mais des résultats satisfaisants ont été mis en valeur dans des domaines d'applications très divers. C'est la raison pour laquelle de nombreux chercheurs étudient ce modèle et tentent de mieux le comprendre et de l'améliorer.

2.9. Réseau Hopfield (J.J. Hopfield 1982)

Le modèle de Hopfield fut présenté en 1982. Ce modèle très simple est basé sur le principe des mémoires associatives. C'est d'ailleurs la raison pour laquelle ce type de réseau est dit associatif. Outre un intérêt pratique, ce réseau admet une analyse théorique précise et complète. Il a par ailleurs contribué à relancer les recherches sur les réseaux de neurones. [7]

2.9.1. Les mémoires associatives

Dans mémoire associative, le terme « mémoire » fait référence à la fonction de stockage de l'information et le terme « associative » au mode d'adressage. Dans le cas des mémoires auto-associatif, dont leur fonction ressemble beaucoup à celle d'un filtre dont le but est de restituer une information en tenant compte de sa perturbation ou de son bruit. L'information doit alors se rapprocher d'une information apprise ou connue. Ces mémoires sont donc principalement utilisées pour la reconstruction de données : l'opérateur fourni une information partielle que le système complète.

2.9.2. Description de réseau de Hopfield

Un réseau de Hopfield est un auto-associateur, et donc son architecteur est semblable à celle d'une mémoire auto-associative**,** il se différencie de cette dernière par : la réponse est binaire, et la mise à jour de la réponse des cellules est asynchrone.

Les valeurs d'entrée sont binaires (-1, 1) mais peuvent être aisément remplacées par les valeurs binaires usuelles (0,1) en utilisant une simple transformation $A_{(-1,1)} = 2.A_{(0,1)} - 1$

L'activation a_j du neurone j se calcule comme suit :

$$a_j \sum_i^N x_i w_{ij} \qquad \text{.....IV.16}$$

Avec :

x_i : valeur de sortie du neurone *i.*

w_{ij} : poids de la connexion entre les neurones *i* et *j.*

On calcule la valeur de sortie du neurone *j* en bridant son activation à l'aide de la fonction signe :

$$x_j = \text{sgn}(a_j) = \begin{cases} -1 & \text{pour } a_j < 0 \\ \text{-1} & \text{pour } a_j > 0 \end{cases}$$

On peut aussi utiliser un seuil :

$$x_j = \text{sgn}(a_j) = \begin{cases} -1 & \text{pour } a_j < q \\ -1 & \text{pour } a_j > q \end{cases}$$

2.9.3. L'apprentissage

La règle d'apprentissage proposée par Hopfield est basée sur la règle de Hebb. La règle de Hebb consiste à forcer les poids des liaisons entre les neurones actifs au même moment. Par contre, les poids seront diminués si les neurones sont dans des états contraires. Dans le cas de Hopfield, cette règle est légèrement étendue si l'on considère que deux neurones dans l'état -1 sont actifs. La règle de modification des poids devient donc :

$$w_{ij} = \sum v_i^p v_j^p \quad \text{avec } w_{ii} = 0, w_{ij} = w_{ji} \quad \text{.....IV.17}$$

w_{ij} : le poids entre les neurones *i* et *j*.

p : le nombre d'exemples présenté au réseau.

v_i^p : le $i^{éme}$ élément du $p^{éme}$ exemple (vecteur).

On remarque que la phase d'apprentissage est immédiate en calculant directement les poids à l'aide de cette fonction. C'est d'ailleurs l'un des seuls réseaux qui permet un calcul aussi simple et analytique des poids.

Ø La généralisation

Les poids ont été modifiés au cours de la phase d'apprentissage et ne seront plus modifiés. On présente alors le vecteur d'entrée au réseau. Ce dernier calcule la sortie correspondante et la ré-injecte à l'entrée. On itère le processus jusqu'à ce que l'entrée à l'étape t et la sortie correspondante (entrée de l'étape t+1) soient identiques (convergence du réseau). C'est à cause de ce processus que le réseau de Hopfield est classifie dans les réseaux récurrents.
Hopfield a démontré d'une part que son réseau évolue forcément vers un point fixe (état final du réseau appelé attracteur).

2.9.4. Réseau de Hopfield synchrone

L'analyse de la fonction d'énergie est grandement facilitée par le fait que le réseau de Hopfield est asynchrone. Pour de nombreuses applications, toutefois il est préférable d'avoir des modèles

synchrones parce qu'ils se prêtent plus facilement à la formalisation matricielle et s'implémente plus facilement sur des ordinateurs parallèles. C'est-à-dire qu'à ce niveau l'activation des neurones ne dépend que de l'état précédent et plus de l'état en cours, contrairement à la modification asynchrone qui dépend à la fois de l'état courant et précédent.

La matrice des poids se calcule par : $W = XX^T$

$W(N \times N)$: matrice des poids.

$X(N \times K)$: matrice des exemples.

N : nombre de neurones.

K : nmobre d'exemples.

2.9.5. Hopfield continue

On peut aussi essayer de généraliser les réseaux de hopfiled au cas continu. C'est-à-dire, les neurones ne répondent plus par oui ou par non, mais que leur activation est restreinte à l'intervalle [-1, +1]. En pratique, cela revient à substituer la fonction sgn(a_i) par la fonction tangente hyperbolique :

$$x_i = \tanh(a_i) = \frac{e^{ai} - e^{-ai}}{e^{ai} + e^{-ai}} \qquad \text{.....IV.18}$$

2.9.6. Etude comparative entre les deux réseaux

Pour mieux résumer les caractéristiques des deux réseaux, nous les décrivons dans le tableau suivant :

Caractéristiques	**Rétropropagation**	**Hopfield**
Stockage	Dépend du nombre de couches	Dépend du type et de la taille du modèle « l'entrée » (La matrice des poids à pour dimension N×N)
L'apprentissage	Calculs lourds et coûteux	Calculs simples et immédiats
Le pas d'apprentissage	Trop petit : convergence lente vers la solution -Trop grand : risque d'oscillation	Il est utilisé dans la phase de désapprentissage, et doit être très petit.
Le nombre d'itérations	Dépend de l'erreur (sortie désirée-sortie obtenue) et d'un nombre d'itérations fixe (très grand)	Dépend de la différence entre l'entrée et la sortie

Tableau IV.1. Comparatif entre les deux réseaux.

3. METHODES STRUCTURELLES [1]

3.1. Introduction

L'approche vue précédemment ne permette pas de prendre en compte l'information structurelle et contextuelle d'une forme. Ces informations sont pourtant importantes dans un grand nombre de problèmes de reconnaissance de formes. C'est, par exemple, le cas en analyse d'images où souvent la décision n'est pas uniquement la détermination d'un seul nom ou le calcul d'un seul nombre mais plutôt la description d'une certaine situation ; par exemple, les objets se trouvant dans l'image et leur disposition mutuelle. Cette description peut être réalisée par l'analyse de l'objet en termes de ses composants, de leurs propriétés et de leur connexion. C'est que l'on appelle l'approche structurelle.

La différence principale entre ces méthodes et les méthodes statistiques est qu'elles introduisent la notion d'ordre dans la description d'une forme.les méthodes les plus répandues utilisent le calcul de la distance d'édition entre deux chaînes et la programmation dynamique.

3.2. Aperçu historique

Dans le domaine de la reconnaissance des formes, les méthodes dites structurelles ou syntaxiques se sont développées à partir de 1965 environ. A l'origine, la reconnaissance structurelle a été introduite d'une part par les linguistes pour la compréhension des longues naturelles, et d'autre part par les informaticiens préoccupés par les langages de programmation et leur compilation. Aussi, le formalisme et la terminologie utilisés aujourd'hui en reconnaissance des formes s'inspirent-ils encore largement de cette époque.

L'analyse des structures, à l'origine linguistiques, s'est alors rapidement étendue à la reconnaissance de structures plus générales notamment dans les domaines du traitement de l'image et de la parole. Un des plus anciens exemples, et aussi un des plus connus, est celui de la classification automatique de chromosomes. D'autres domaines d'application ont bénéficié de ces nouvelles méthodes, tels que la reconnaissance de caractères, de textures, de formules mathématiques, mais aussi l'analyse de signaux sismiques, d'électrocardiogrammes, ou encore l'étude des photos de chambres à bulles.

3.3. Notion de structure

La notion de structure recouvre des concepts variés. Une structure doit décrire la composition d'un objet complexe en termes d'objets plus simples ainsi que les liens entre eux. Cette description est faite selon un modèle particulier, adapté aux objets considérés.

Le modèle de structure le plus simple est la chaîne dont le principe consiste à représenter une forme comme une suite d'objets élémentaires. La présence ou l'absence d'un de ces objets, ainsi que leur ordre d'apparition caractérisent la forme globale. Le codage de Freeman d'une courbe dans un maillage en est un exemple.la figure 1111 représente une courbe discrétisée et sa représentation sous la forme d'une chaîne de chiffres (à gauche) conforme au codage défini (à droite).

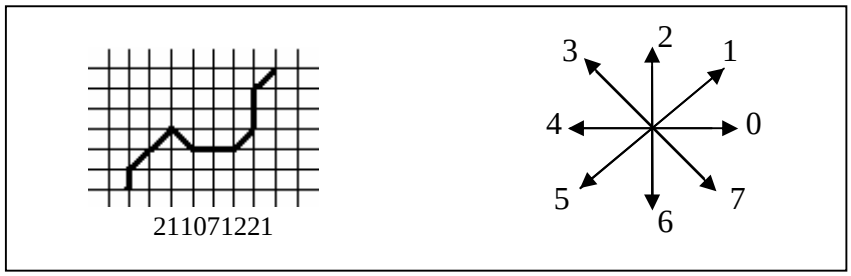


Fig. IV.10 Codage de freeman.

Les chaînes jouent un rôle prépondérant dans l'étude théorique de la reconnaissance structurelle. En effet, toute structure peut être représentée à l'aide d'une chaîne moyennant une convention de codage.

3.4. Structure de chaîne

Les méthodes structurelles consistent à représenter des formes quelconques, complexes, comme des assemblages structurés de motifs élémentaires, dans le cas le plus simple, celui d'une structure de chaîne, qui est envisagé dans ce chapitre, on représente les formes à reconnaître comme étant des suites ordonnées, appelées phrases, de motifs élémentaires, appelés lettres (suivant la terminologie de la théorie des longages. On peut dire aussi symboles), qui appartiennent à un alphabet (on dit aussi un vocabulaire) fini :

$X = \{\alpha_1, \alpha_2, \ldots \alpha_\infty\}$.

L'opération élémentaire est la concaténation, notée *. Soient a = a_1.....a_n et $b = b_1$…….b_m. Alors :

a* b = a_1.....a_n b_1…….b_m.

La concaténation * est associative mais non commutative.

La phrase vide est la phrase de longueur nulle notée λ.

3.5. Notions de distances

La distance séparant deux chaînes de caractères peut être vue comme le nombre minimum de transformations élémentaires à appliquer à une chaîne pour obtenir la deuxième. La classification et la décision par la distance constitue une des approches les plus simples et les plus intuitives de la reconnaissance des formes. La motivation première d'une telle approche est qu'il est naturel de considérer qu'un élément appartient à une classe s'il est plus proche de cette classe que de toutes les autres.

Il est donc nécessaire de définir la distance entre un point et une classe à partir de la distance entre points. Voilà donc un nouveau problème auquel va être confrontée la RF de manière constante surtout que la distance dépend de la forme à traiter et des paramètres extraits.

Ø Définition générale d'une distance

Avant toute chose, il apparaît fondamental de définir ce qu'est une distance dans le cas général et d'étuduer ensuite quelles sont les différentes distances entre représentations.
Considérons un ensemble quelconque *E* de points. Nous dirons que *E* est un espace métrique réel s'il existe une fonction appelée distance, notée :

$D : E \text{ x } E \longrightarrow R,$

Vérifiant les quatre propriétés suivantes :

1. Séparabilité : $\forall(a,b) \in E^2, \quad a \neq b \Rightarrow a(a,b) > 0$
2. Réflexivité : $\forall\, a \in E, \quad d(a,b) = 0$
3. Symétrie : $\forall(a,b) \in E^2, \quad d(a,b) = d(b,a),$
4. Inégalité triangulaire : $\forall(a,b,c) \in E^2, \quad d(a,c) \leq d(a,b) + d(b,c),$

Ø Distance entre vecteurs

Soit *E* un ensemble de vecteurs. Soient *X ={xi}* et *Y = {yi}* avec *i = 1, ...,N,* deux éléments de *E*. On peut définir un certain nombre de distances classiques :

- La distance de Hamming : $d_1(X,Y) = \sum_{i=1}^{N} |x_i - y_i|;$
- La distance euclidienne : $d_2(X,Y) = \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2};$

- La distance du maximum : $d_\infty(X,Y) = \max_{i=1,n} |x_i - y_i|$;
- La distance d_n : $d_n(X,Y) = (\sum_{i=1}^{N} |x_i - y_i|^n)^{\frac{1}{n}}$;

Ceci correspond à la définition générale de la distance. En effet, avec *n = 1*, on retrouve la distance de Hamming, avec *n = 2*, on retrouve la distance euclidienne, et avec *n = ∞*, on retrouve la distance du maximum.

Ø Distance d'un point à une classe

Cette notion est importante dans un système de reconnaissance. En effet, pour attribuer un élément *x* à une classe C_k, il faut définir un critère de proximité comme suit :

$$x \in C_k \Leftrightarrow C_k = Arg\min c_i \; d(x, C_i),$$

Où $Arg\min(f(C_i))$, est la fonction qui donne la classe C_i minimisant la fonction f. Il est donc nécessaire de définir la distance d'un point à une classe. Comme nous le disions précédemment, la définition de cette distance n'est pas unique et dépend des formes traitées. Nous allons donner dans la suite une définition possible de cette distance.

On considère un espace métrique *E*. La distance d_1 d'un point *p* de *E* à une classe *C* de *E* est définie par :

$$d_1(p, C_i) = \inf\{d(p,m);\ \ m \in C\}.$$

La distance d_2 entre deux classes C_1 et C_2 est définie par :

$$d_2(C_1, C_2) = \inf\{d(p,m);\ p \in C_1 \text{ et } m \in C_2\}.$$

Plus la distance consédirée est petite, plus on admet que la ressemblance, au point de vue de la reconnaissance des formes, est grande.

Distance entre chaînes

La distance entre chaînes peut être calculée par minimisation de distances locales entre les composantes des chaînes et optimisation donc de la ressemblance [Wag74].

Si l'on prend l'exemple de deux chaînes de caractères {abc} et {abde}, on s'aperçoit que la comparaison peut se faire en termes de transformations pouvant faire passer d'une chaîne à l'autre. Par exemple, pour passer de la première chaîne à la deuxième, il faut conserver les deux premiers caractères « ab », changer le troisième, remplacer la lettre « c » par la lettre « d » et ajouter un « e ». on remarque que ces opérations peuvent être effectuées à l'aide des trois transformations suivantes : SUB, DES, CRE, définies par :

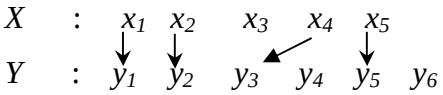
SUBstitution $x_i \longrightarrow y_i$ SUB (x_i, y_i),

DEStruction $x_i \longrightarrow \lambda$ DES (x_i, λ),

CREation $\lambda \longrightarrow y_i$ CRE (λ, y_i).

Où λ est le caractère vide. A chacune de ces transformations est associé un coût $c(x_i, y_i)$, pour SUB, $c(x_i, \lambda)$ pour DES et $c(\lambda, y_i)$ pour CRE. La distance entre deux chaînes est égale au coût total des transformations les moins coûteuses de passage d'une chaîne à l'autre. Elle est appelée distance d'édition $D(X,Y)$. Le schéma de transformation, appelé trace, reproduit par des flèches les transformations concernées.

Soient $X=\{x_1, x_2, x_3, x_4, x_5\}$ et $Y=\{y_1, y_2, y_3, y_4, y_5, y_6\}$. Supposons que la trace de la transformation de X en Y soit la suivante :



Elle peut aussi s'écrire comme suit : SUB(x_1, y_1) ; SUB(x_2, y_2) ; DES(x_3, λ) ; SUB(x_4, y_3) ; SUB(x_5, y_5) ; CRE (λ, y_6). Chaque opération « trace » est affectée d'un coût et la distance est assciée au coût minimal.

De manière générale, soient $X(n)=\{x_1, x_2, ..., x_n\}$ et $Y(m) = \{y_1, y_2, ..., y_m\}$, deux chaînes de longueurs respectives *n* et *m*. Il existe trois manières de progresser dans les deux chaînes pour aboutir à (x_r, y_k) :

- Venir de (x_{r-1}, y_{k-1}) et faire SUB (x_r, y_k) ;
- Venir de (x_r, y_{k-1}) et faire CRE (λ, y_k) ;
- Venir de (x_{r-1}, y_k) et faire DES (x_r, λ).

3.6. Programmation dynamique

La programmation dynamique est utilisée en RF pour chercher la ressemblance entre deux chaînes par le calcul d'une distance.

La programmation dynamique est une méthode de recherche d'optimum fondée sur le principe d'optimalité locale de Bellman : « Dans une séquence optimale de décisions, quelle que soit la première décision prise, les décisions subséquentes forment une sous-séquence optimale, compte tenu des résultats de la première décision »[Bel57]. Ce principe indique que tout chemin optimal est constitué de portions de chemins elles-mèmes optimales. En effet, si on considère un chemin optimal reliant *A* à *B* et un point quelconque *M* de ce chemin, les chemins de *A* à *M* et de *M* à *B* sont tous deux optimaux.

3.7. Distance d'édition

La distance d'édition (également appelé "distance de Levenstein") *D* entre *X* et *Y* est le coût minimum d'une suite de transformations élémentaires permettant de passer de *X* à *Y*.
Classiquement, on considère trois transformations élémentaires :

- L'insertion d'une lettre a_i ,de coût $\gamma(\lambda, a_i)$,
- La suppression d'une lettre b_i ,de coût $\gamma(b_i, \lambda)$,
- La substitution d'une lettre a_i par une lettre b_i , de coût $\gamma(a_i, b_i)$.

L'algorithme de *WAGNER* et *FISHER* calcule la distance d'édition avec une complexité *O(m.n)* par programmation dynamique :

On note *n* et *m* les longueurs respectives de *X* et *Y*.

$\delta(0,0) := 0$;

pour *i* variant de 1 à *n* effectuer :

$\delta(i,0) := \delta(i-1,0) + \delta(a_i, \lambda)$;

pour *j* variant de 1 à *m* effectuer :

$\delta(0, j) := \delta(0, j-1) + \delta(\lambda, b_i)$;

pour *i* variant de 1 à *n* effectuer :

pour *j* variant de 1 à *m* effectuer :

$m_1 := \delta(i-1, j-1) + \gamma(a_i, b_j)$;
$m_1 := \delta(i-1, j) + \gamma(a_i, \lambda)$;
$m_2 := \delta(i, j-1) + \gamma(\lambda, b_j)$;
$\delta(i, j) := Min(m_1, m_2, m_3)$;

fin

la distance d'édition *D(X,Y)* est alors le résultat final de l'algorithme $\delta(m,n)$. La matrice δij, $1 \le i,j \le n$, correspond aux distances cumulées.

- **L'ajustement dynamique**

L'ajustement dynamique, dù à *SAKOE* et *CHIBA*, est une famille de méthodes sous-optimales comparables à la méthode précédente. On cherche un chemin d'ajustement entre l'indice de départ *(0,0)* et l'indice d'arrivée *(m,n)*. La différence essentielle porte sur les contraintes : une lettre peut correspondre à plusieurs lettres de l'autre phrase et aucune phrase

et aucune lettre ne peut être transformée en la lettre vide. On peut également restraindre les valeurs de *(i,j)* à un couloir diagonal.

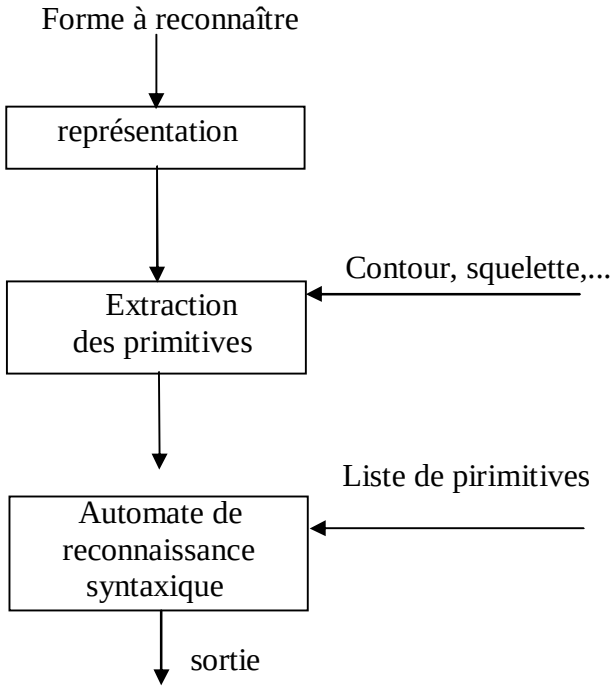


Fig. IV.11 principe général des méthodes structurelles (organigramme).

La forme la plus simple de la méthode donne l'algorithme correspond à prendre :

$$m_1 := \delta'(i-1, j-1) + 2c(i, j);$$
$$m_2 := \delta'(i-1, j) + c(i, j);$$
$$m_3 := \delta'(i, j-1) + c(i, j);$$
$$\delta'(i, j) := \mathrm{Min}(m_1, m_2, m_3);$$

Sur un couloir diagonal s'écrivant *{(i,j) / j-r ≤ i ≤ j + r}*. Les coûts des insertions et des suppressions élémentaires sont pris égaux à c, et le coût des substitutions par des lettres différentes est égal à 2*c*. La distance entre *X* et *Y* vaut alors :

$$D(x, y) = \frac{\delta'(n, m)}{m + n}$$

L'ajustement dynamique est très utilisé en reconnaissance de la parole.

4. CONCLUSION

A notre connaissance, tous les résultats présentés dans la littérature montrent clairement que la combinaison de classifieurs est une voie de recherche prometteuse. Elle permet d'améliorer les performances du système de reconnaissance et de s'adapter à une grande variété de situations par l'exploitation de connaissance à priori sur les comportements des classifieurs.

L'objectif visé dans ce chapitre est de donner un aperçu général sur les deux approhes adoptées dans ce travail. Dans un premier temps, nous avons présenté les réseaux de neurones artificiels et les différentes règles d'apprentissage. En pratique, l'utilisation de la méthode neuronale pose certaines difficultés. La principale difficulté est l'optimisation de la phase d'apprentissage et Le choix de l'architecture adéquate. Pour L'approhe structurelle, elle consiste à mettre en relation la structure des formes analysées et la syntaxe d'un longage formel. La description des formes est réalisée par l'intermédiaire de phrases et le problème de classification est ramené à un problème d'analyse de grammaire.

CHAPITRE V

UNE APPROCHE HYBRIDE POUR LA RECONNAISSANCE D'ECRITURE ARABE MANUSCRITE

1. INTRODUCTION

Contrairement au latin, la reconnaissance de l'écriture arabe manuscrite ou imprimée reste encore aujourd'hui au niveau de la recherche et de l'expérimentation, le problème n'est pas encore résolu bien que l'on sache atteindre des taux assez élevés dans certaines applications pour lesquelles soit le vocabulaire est limité, soit la fonte est unique. Les travaux sont généralement axés sur la méthodologie du développement plutôt que sur la réalisation d'un produit fini vendable. Une version commercialisable reste encore au stade du rêve, les efforts doivent se multiplier pour le réaliser.

La reconnaissance de l'écriture arabe date des années 80. Depuis, les recherches se sont multipliées dans ce domaine. Certains chercheurs se sont intéressés à la reconnaissance en temps réel en utilisant des tablettes graphiques, ce qui simplifie en partie le problème en restituant le sens du tracé, d'autres se sont penchés sur l'imprimé et/ou le manuscrit en "Off-line" en utilisant un scanner ou une caméra pour la saisie des documents. Depuis, plusieurs outils de pré-traitement ont été développés pour la squeletisation de l'image, de son lissage,

pour la détermination du contour et l'extraction de primitives...et différentes techniques de reconnaissance ont été élaborées, parmi lesquelles figurent des méthodes connexioniste, statistiques, structurelles ou géométriques, avec ou sans segmentation en caractères. Généralement toutes ces méthodes et bien d'autres, tendent à extraire, chacune à sa façon, une catégorie de caractéristiques et d'évaluer par la suite la vraisemblance entre les primitives extraites et celles de formes prototypes déjà apprises par le système de reconnaissance.

Dans ce cadre, nous proposons une approche hybride, de reconnaissance de chaînes de caractères arabes manuscrites correspondant à des mots ou des parties de mots à vocabulaire limité, Reposant sur la combinaison de deux méthodes, connexioniste et structurellee.

D'une part, on utilise l'approche connexioniste basée sur un nombre restraint de caractéristiques significatives de l'écriture d'un mot (jambages, hampes, nombre de composantes connexes...etc.), pour obtenir un ensemble réduit de mots hypothèses (mots possédant les mêmes caractéristiques que celles du mot à reconnaître). D'autre part, pour sélectinner le vrai mot parmi celles données par le niveau de classification en classes (réalisée par le classifieur neuronal), on procède à un codage des résultats condidats en exploitant des informations structurelles et syntaxiques. L'approche structurelle, quant à elle, permet de choisir la meilleure hypothèse selon sa point de vue qui, certes, n'était pas engendré par le pur hasard, mais, en contre, il est basé sur une stratégie adoptée dans la littérature du métier.

Dans ce qui suit, nous décrirons notre approche adoptée pour la reconnaissance des mots arabes manuscrits à vocabulaire limité avec application sur les montants littéreaux des chèques postaux.

Du fait que l'écriture arabe présente certaines caractéristiques qui sont à l'origine de la complexité du traitement, particulièrement parce qu'elle est cursive aussi bien dans sa forme imprimée que manuscrite, nous jugeons nécessaire avant d'entreprendre le travail de jeter un coup d'œil sur les principales caractéristiques de l'écriture arabe.

2. CARACTERISTIQUES DE L'ECRITURE ARABE

Les caractères arabes (voir tableau V.1) différent d'autres types de caractères (Latins, Chinois….) par leur propre structure et leur mode de liaison pour former un mot, ce qui rend l'application directe des techniques de reconnaissance développées pour les caractères chinois ou latins, par exemple, une tâche délicate pour leur reconnaissance.

L'écriture arabe se caractérise par :

- Les caractères qui s'écrivent de droite à gauche.
- L'arabe a 28 caractères de base et il n'y a pas de majuscules.
- L'écriture arabe est par nature cursive, qu'elle soit imprimée ou manuscrite.
- Selon sa position dans le mot (en tête, à l'intérieur, à la fin ou isolé), un même caractère arabe peut avoir jusqu'à quatre formes différentes, ce qui multiplie le nombre de modèles (tableau V.1).
- Certains caractères ont le même corps, mais la présence et la position d'un point ou d'un groupe de points, sont les traits déterminants pour distinguer ces caractères. La figure (V.1) montre quelques caractères qui ont le même corps qui se différencient seulement par la présence et le nombre de points au-dessus ou au-dessous de leur corps.

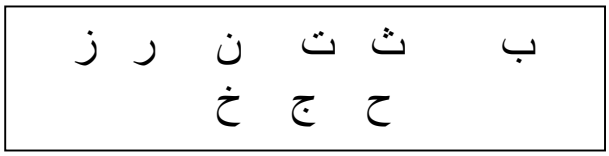


Fig.V.1. Caractères dépendant des points diacritiques.

- Un caractère arabe peut contenir un trait vertical (TAA(ط)), un trait oblique (KAF(ك)) ou un zigzag (ALIF(أ)).
- Les points et le zigzag (Hamza) sont appelés des secondaires.
- Un mot arabe se compose généralement d'une ou plusieurs composantes connexes, chacune contient un ou plusieurs caractères (Fig. V.2).

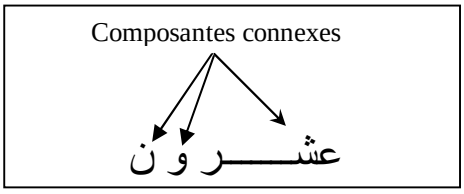


Fig.V.2. Exemple de mot arabe avec trois composantes connexes.

Caractère	Au début	Au milieu	Fin	Isolé		Caractère	Au début	Au milieu	Fin	Isolé
alif	أإ	ـأـا△	ـأـا	أإ		dhad	ضـ	ـضـ	ـض	ض
ba	بـ	ـبـ	ـب	ب		tad	طـ	ـطـ	ـط	ط
tâ	تـ	ـتـ	ـت	ت		thad	ظـ	ـظـ	ـظ	ظ
thâ	ثـ	ـثـ	ـث	ث		ayn	عـ	ـعـ	ـع	ع
jim	جـ	ـجـ	ـج	ج		ghayn	غـ	ـغـ	ـغ	غ
ha	حـ	ـحـ	ـح	ح		fa	فـ	ـفـ	ـف	ف
kha	خـ	ـخـ	ـخ	خ		kaf	قـ	ـقـ	ـف	ق
del	د	ـد△	ـد	د		kef	كـ	ـكـ	ـك	ك
dhel	ذ	ـذ△	ـذ	ذ		lam	لـ	ـلـ	ـل	ل
ra	ر	ـر△	ـر	ر		mim	مـ	ـمـ	ـم	م
zei	ز	ـز△	ـز	ز		noun	نـ	ـنـ	ـن	ن
sin	سـ	ـسـ	ـس	س		ha	هـ	ـهـ	ـه	ه
shin	شـ	ـشـ	ـش	ش		wew	و	ـو△	ـو	و
sad	صـ	ـصـ	ـص	ص		ye	يـ	ـيـ	ـي	ي

Le triangle △ montre les caractères qui ne peuvent pas être attachés à leur successeur dans le mot

Tableau V.1 Caractères de l'alphabet arabe.

- la discrimination entre les groupes de caractères d'un mot, est due aux six caractères de l'alphabet arabe qui ne peuvent pas être attachés à leur successeur dans le mot (tableau V.1).
- Une caractéristique distinguée de l'écriture arabe, est la présence de ce qu'on appelle la ligne de base. La ligne de base possède un nombre maximum de pixels (noirs) dans le texte.
- Certains caractères dans un mot, peuvent se chevaucher verticalement sans contact (Figure V.3).
- Les caractères arabes ne possédant pas une taille fixe (hauteur et largeur), leur taille varie d'un caractère à un autre et d'une forme à une autre à l'intérieur d'un même caractère (tableau V.1).
- A l'intérieur d'un mot, les caractères peuvent avoir des diacritiques (voyelles courtes). Ces diacritiques sont écrites comme des traits au-dessus de la ligne de base :
- Fat-hah « ◌َ », Dhammah « ◌ُ », Shadah « ◌ّ », Maddah « ◌ٓ », Soukoun « ◌ْ », ou au-dessous de la ligne de base comme Kasrah « ◌ِ ». En plus de Tenween qui peut être formé en faisant doubler le Fat-hah « ◌ً », Dhammah « ◌ٌ » ou Kasrah « ◌ٍ ».

Notons qu'une diacritique différente sur un caractère peut changer le sens d'un mot (ex. Drapeau (عَلَمٌ), Science(عِلْمٌ)).

- Généralement, les textes arabes ne sont pas voyellisés, notamment dans la presse et les livres. Les lectures de l'arabe sont habituées à lire ces textes en déduisant le sens à partir du contexte.
- Plusieurs caractères peuvent se combiner verticalement pour former une ligature (voir figure V.3)
- Comme l'écriture arabe est une écriture calligraphique, six styles graphiques différents sont souvent utilisés : Tholoth, Naska, Requeh, Dewani, Farci, et Koufique.

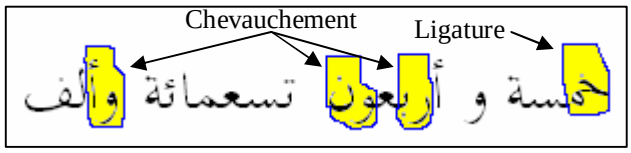


Fig.V.3. Un échantillon de l'écriture arabe illustrant certaines de ses caractéristiques.

En résumé, les principales propriétés graphiques de l'écriture arabe sont les suivantes :

1- L'écriture se fait de droite à gauche, et elle est cursive par nature.

2- La liaison entre les caractères se fait par l'intermédiaire de branches de faibles rayons de courbure. Les points de jonction se situent le plus souvent près de la ligne d'écriture,

3- La plupart des caractères sont composés de boucles et de courbes, souvent tracées dans le sens horaire ; quelques-uns comportent des traits verticaux,

4- Le chevauchement des caractères et la présence de ligatures dans un mot, rendent l'utilisation des méthodes de segmentations, caractère par caractère, du mot à reconnaître une tâche très délicate,

5- D'une même forme peut exister dans des caractères différents. Par exemple, la forme de boucle apparaît dans les caractères SSAD, DHAD, TTA, DDHAH, QAF, FA, mais peut apparaître seule comme pour le caractère MEEM.

3. METHODOLOGIE

L'objectif que nous nous sommes assignés s'articule autour de développement d'un système rapide et fiable pour la reconnaissance des montants littéreaux arabes des chèques postaux omni-scripteurs. La figure (V.4) montre la méthodologie, que nous avons adoptée pour la reconnaissance.

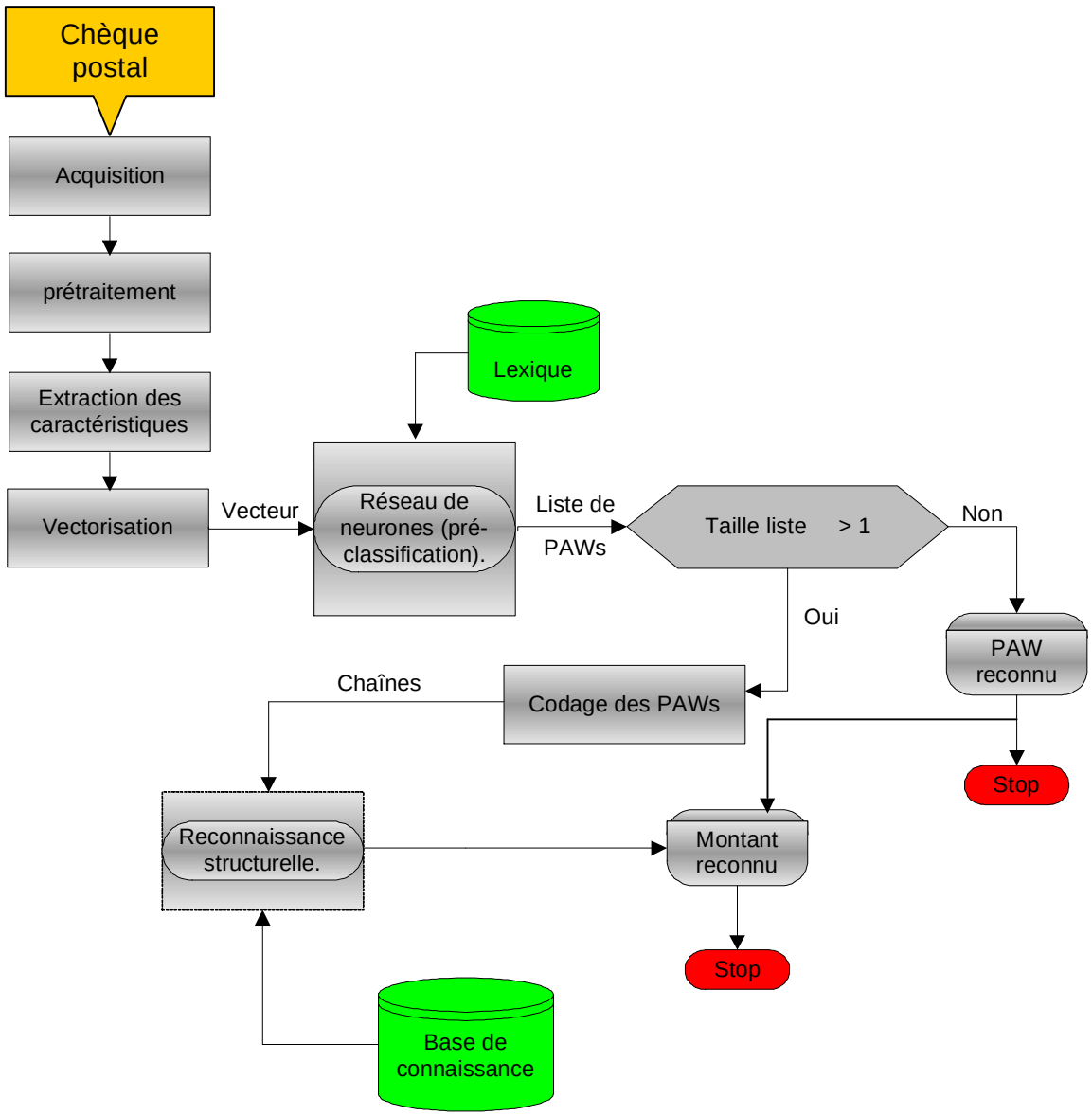


Fig.V.4. Méthodologie adoptée pour la reconnaissance.

L'image du montant à reconnaître passe, après l'aquisition, par les étapes de prétraitement et d'extraction de primitives. Le réseau de neurone donne à sa sortie l'ensemble des PAWs condidats, puis, le l'approche structurelle déterminera le vrai montant.

4. STRATEGIE DE RECONNAISSANCE

Plusieurs solutions ont été proposées pour la reconnaissance de l'écriture arabe, comme pour toute écriture cursive, des méthodes globales et analytiques ont été testées.

Cependant, à cause des caractéristiques de l'écriture arabe vues précédemment (Tableau V.1), la solution n'est pas triviale. En effet, l'étude morphologique de l'arabe montre qu'il est difficile d'opérer la reconnaissance au niveau du caractère ce qui nous a conduit à s'orienter vers la chaîne de caractères, qu'il est convenu d'appeler PAW-Piece of Arabic Word ; le PAW peut correspondre au mot ou à la partie du mot.

Les PAWs présentent une structure facile à isoler (séparation en composantes connexes, extraction de contours...), ils peuvent occuper différentes positions dans le mot sans pour autant changer de formes. De plus, ils caractérisent toute la morphologie de l'écriture arabe ; on y trouve des caractères isolés et des caractères ligaturés.

Cependant, les PAWs contiennent des ligatures horizontales et verticales bien spécifiques :

- Les ligatures horizontales sont variables, elles se traduisent par un allongement de la ligature de base ce qui complique le processus de segmentation.
- Les ligatures verticales, bien qu'elles soient rares, posent d'importants problèmes lorsqu'elles existent. Le chevauchement vertical des caractères modifie souvent la morphologie de certains d'entre eux et la ligne de base n'est plus horizontale.

Le choix du PAW comme forme de base à traiter est pénalisé par le nombre important de PAWs dans le cadre d'un vocabulaire libre : il serait inconcevable d'envisager un modèle par PAW.

Néanmoins, cette solution s'avère très intéressante dans le cas d'une application spécifique pour laquelle le vocabulaire associé est limité. Ce vocabulaire est réduit davantage à cause de la redondance des PAWs dans l'écriture arabe ; en effet, le même PAW peut appartenir à différents mots. A titre d'exemple pour l'application que nous avons choisie (reconnaissane des montant littéreaux des chèques postaux), le nombre de PAWs de base est réduit à la moitié uniquement.

Mot composés de quatre composantes connexes	Mot composés de trois composantes connexes	Mot composés de deux composantes connexes	Mot composés d'une composante connexes
أربعون	اثنان	ستون	خمسة
أربعمائة	أربعة	ثلاثة	ستة
جزائري	عشرون	ثمانية	سبعة
	ثلاثون	تسعمائة	عشر
	ثمانون	تسعون	تسعة
	مائتان	ستمائة	و
	ثلاثمائة	سبعون	
	ثمانمائة	سبعمائة	
	ألفان	اثنا	
	آلاف	مائة	
	دينار	مائتا	
		خمسون	
		خمسمائة	
		عشرة	
		احد	
		الف	
		الفا	

Tableau V.2. Décomposition du vocabulaire.

خمسة	ستة	سبعة	تسعة	عشر	خمسما	ستما	سبعما	تسعما
خمسو	ستو	سبعو	تسعو	نية	بعة	بعما	بعو	ثلا
ثو	ثنا	لفا	لف	لا	ئتا	أ	ن	ة
ئر	جز	و	ئة	نو	ينا	د	حد	ما
نما	ف	ي	ثما	ثة	ر			

Tableau V.3. Liste des PAWs.

5. PRE-TRAITEMENT

Le pré-traitement inclut toutes les fonctions effectuées avant l'extraction des caractéristiques pour produire une version « nettoyée » de l'image d'origine afin qu'elle puisse être utilisée directement et efficacement par le composant d'extraction de caractéristiques.
Les prétraitements appliqués sont typiquement des opérations de traitement d'images spécifiques telles que la binarisation, le lissage, la correction de l'inclénaison, en plus de la localisation de la ligne de référence (ligne de base) et l'extraction de la zone médiane.

6. EXTRACTION DE CARACTERISTIQUES

Cette étape cherche à construire un vecteur caractéristique décrivant globalement l'image de PAW. Cette description est basée sur l'extraction des primitives significatives de l'écriture des mots arabes manuscrits.

Malheureusement, il n'y a pas de théorie pour guider le choix de ces primitives. Ceci reste pour l'instant dans le domaine de l'art plus que dans celui de la science. On peut toutefois formuler les points suivants à propos des primitives :

- Discriminabilité : une bonne primitive doit être avoir des valeurs significativement différentes pour des formes appertenant à des classes différentes,
- Fiabilité : une bonne primitive doit avoir des valeurs très similaires (à faible variabilité) pour les formes d'une même classe,
- Indépendance : dans un ensemble de primitives choisies, une primitive quelconque ne doit pas dépendre d'une autre primitive.
- Nombre : le coût, la complexité et les exigences en temps de calcul d'un système de reconnaissance croissent avec le nombre de primitives utilisées. Il faut donc choisir les meilleures primitives pour diminuer ce nombre tout en satisfaisant la probabilité d'erreur acceptable, fixée à l'avance.

L'ensemble des paramètres choisis pour décrire globalement une image de PAW forme un vecteur à partir des caractéristiques choisies à savoir :

- Le nombre de composantes connexes dans l'image ;
- Le nombre de jambages (descendants) détectés dans l'image ;

- Ø Le nombre de Hampes (ascendants) détectés dans l'image ;
- Ø Le nombre de points au-dessus de la ligne de base ;
- Ø Le nombre de points au-dessous de la ligne de base ;
- Ø Le nombre de Hamzas (zigzags) dans le PAW ;
- Ø Le nombre de boucles dans le PAW ;
- Ø Le rapport entre la surface de la boucle et la surface du PAW ;
- Ø La position de la boucle (au début ou à la fin).

Ces paramètres sont les caractéristiques de l'écriture arabe les plus stables.

Les boucles								Les points	
Une boucle	Deux boucle	Sans Boucles	Boucle à la fin	Boucle au début	Rapport SB/SPAW		1. P Au-dessus	2. P Au-dessus	3. P Au-dessus
ة ما نو ف لفا لف نما و ثو ثما ثة ئة نية ستو ستما ستة	بعة بعو بعما تسعو سبعو خمسو تسعما سبعما خمسما تسعة سبعة خمسة	أ ر ن د ي حد جر ئر ئتا ينا ثنا لا عشر	ئة ثة سبعة تسعة ستة نية بعة خمسة ة	ف ما و	ما و ف نو ثو ثما ثة نية ئة نما لف لفا بعة بعو ة	سبعة تسعة ستة ستو ستما سبعة تسعو تسعما خمسة خمسو خمسما	ن ف لف لفا جز نو ينا نما خمسو خمسما	بعة ستو ستما سبعة تسعو تسعما	ثما ثو ثلا نية خمسة عشر

Tableau V.4. Liste des caractéristiques.

Les points						Les hampes et jambages		
1.P Au-dessous	2.P Au-dessous	4.P Au-dessus	5.P Au-dessus	Avec Zigzag	Sans points	Avec Hampe	Avec Jambage	Sans H et J
جز بعو بعما بعة سبعو سبعما سبعة	نية ينا	ستة تسعة ثنا	ثة	أ ئتا ئة	أ د ر حد و ما لا	أ ما نما ثما ينا ئتا لا لف لفا ثتا ثلا بعما تسعما سبعما ستما خمسما	ر و ي ن نو جز ئر ثو بعو تسعو سبعو ستو خمسو عشر	د ة ثة ئة نية حد بعة خمسة ستة سبعة تسعة

Tableau V.4. Liste des caractéristiques.

7. VECTORISATION

Cette procédure génère pour chaque PAW un vecteur caractéristique. Les différents paramètres caractérisant le PAW ont été déterminés pour vectoriser les données acquises sur chaque PAW analysé. Le vecteur obtenu permettra l'identification du sous-ensemble de tout les PAWs dans le lexique ayant les mêmes caractéristiques se confondant avec les éléments de ce vecteur.

Le format du vecteur caractéristique est illustré dans le tableau (V.5).

Caractéristique	Code
Une boucle	1_Bou
Deux boucles	2_Bou
Boucle située à droite	Poi_D
Boucle située à gauche	Poi_G
Un point au dessus	1_P_dus
Deux points au dessus	2_P_dus
Trois points au dessus	3_P_dus
Quatre points au dessus	4_P_dus
Cinq points au dessus	5_P_dus
Un point au dessous	1_P_dous
Deux points au dessous	2_P_dous
Avec zigzag	A_zig
Avec Jambage	A_J
Avec Hampe	A_H
Avec deux Hampes	A_2H

Tableau V.5. Constitution du vecteur de cractéristiques.

8. CLASSIFICATION

Apprés l'établissement des vecteurs caractéristiques de chaque PAW, les classes ont été sélectionnées. Un réseau de Hopfield est utilisé pour faire la classification, et afin de lui donner le comportements désiré, nous avons construit un ensemble d'apprentissage de PAWs.

Dès lors, des images de PAWs sont extraites à partir de l'image du montant à reconnaître. Les vecteurs caractéristiques ont été construits, chaque vecteur se sert comme une entrée du réseau, après convergence le réseau Hopfield va donner à sa sortie la classe correspondante. Si cette classe contient un seul PAW, donc le processus de reconnaissance sera d'emblée interrompu et le PAW est considéré comme sous-mot reconnu, sinon la liste des PAWs engendrée par le réseau passera au second manche.

Ø Reconnaissance de vrai PAW à l'aide de l'approche structurelle

La reconnaissance de vrai PAW consiste à établir la liste, classée par ordre de coûts croissants, de tous les PAWs contenant dans la liste. L'objectif est d'obtenir le PAW correct en première position dans cette liste avec évidemment un coût inférieur aux coûts associés aux autres PAWs.

Codage des PAWs

En premier lieu, toutes les PAWs des classes qui contiennent plus d'un PAW sont codé par une séquence de nombres déterminés à partir de l'histogramme de transition 0-1, horizontal et vertical. Et ce, afin de former le dictionnaire qui sert comme référence de comparaison.

A l'issue de l'étape de pré-classification, les PAWs sont à leur tour codés par la même procédure.

Une distance d'édition (voir chapitre IV) est utilisée entre deux chaînes, la première chaîne est la séquence de nombres issues du codage de chaque PAW à reconnaître, et la seconde est une chaîne de nombres formant un PAW du dictionnaire correspondant au PAW de la liste se trouvant à la sortie du réseau.

Cette procédure est réitérée pour chaque PAW de la liste à examiner. Enfin, les mots sont classés par ordre de coût croissant. Le vrai PAW se trouve ainsi en première position de cette liste.

9. RESULTATS ET DISCUSSION

Le test que nous avons réalisé est basé sur la reconnaissance des PAWs du vocabulaire lié à la rédaction des chèques (42 PAWs). Ces mots ont été écrits par dix scripteurs différents.

Un tableau est présenté ci-dessous dans lequel est rapporté le bilan de la reconnaissance. On retient un pourcentage de reconnaissance (réussite) de 100% avec la base d'apprentissage. Pour ce qui concerne le reste (la base de test) le taux de reconnaissance a été diminué à cause de diverses erreurs relevées dans le test.

Il s'agit par exemple d'une opération de binarisation qui a échoué, zone médiane incorrecte (cette erreur est toujours fatale puisque sa détection est principalement liée à la localisation des hampes et jambages), boucle non fermée (Cette erreur est aussi fatale car il n'y a pas de règle pour reformer des lettres avec une boucle), etc.

	Taux de Reconnaissance en (%)	Taux d'échec en (%)
Base d'apprentissage	100 %	0 %
Base de test	86 %	12 %

Tableau V.6. Taux de reconnaissance sur les ensembles d'apprentissage et de test.

CONCLUSION GENERALE

Le problème de la reconnaissance de l'ériture arabe manuscrite non-contrainte est complexe. Dans un premier lieu, il faut trouver une modélisation qui tire profit de ses propriétés bidimensionnel intrinsèques. De plus, cette modélisation doit prendre en compte des erreurs qui peuvent affecter les données qui arrivent souvent entachées de bruit dû à l'acquisition ou aux prétraitements (ex., binarisation) qui se trouvent en amont du système de reconnaissance.

La modélisation de l'écriture manuscrite peut se faire à travers deux approches totalement différentes : une approche globale ou bien une approche analytique.

Ce travail repose sur la reconnaissance globale de sous-mots (PAW). Le choix de cette approche se justifie par une raison simple, lorsque on examine de plus prés les carctéristiques de l'écriture arabe, on arrive et sans aucune difficulté, pour réduire le degré de complexité de conception et pour ne pas emprunter des voies pleines d'obstacles, au choix de la méthode globale comme stratégie de reconnaissance. De plus, les mots arabes peuvent être constitués de plus d'une partie (sous-mot ou piece of arabic word), donc il est nécessaire de se pencher sur la reconnaissance de sous-mots si nous n'avons pas dans les mains un petit système de reconnaissanse de présence de mots.

Cependant, cette approche présente des inconvénients, à savoir :

- La discrimination de mots orthographiquement proches est très difficile. Pour cette raison, ce type d'approche est limité à des vocabulaires réduits allant jusqu'à quelques centaines de mots,
- Elle est coûteuse en place car la complexité croît linéairement avec la taille du lexique,
- Un changement de vocabulaire nécessite l'apprentissage de tous les mots nouveaux.

Dans ce travail nous avons présenté en détail les différentes étapes d'un système de reconnaissance hors-ligne de l'écriture, dédié à une application spécifique à vocabulaire limité : la lecture automatique des montants littéraux de chèques postaux.

Comme nous l'avons vu, une approche hybride qui tire profit de la complémentarité de deux approches (connexioniste et structurelle) semble une solution intéressante au problème de la reconnaissance de l'écriture arabe manuscrite à vocabulaire limité. L'approche neuronale a été appliquée pour réduire le plus possible la liste des PAWs condidats, en exploitant les caractéristiques pertinentes de l'écriture arabe. Pour cela, nous avons utilisé des primitives trés robustes (jambages, hampes, boucles, etc.). si la liste contient plus d'un PAW, le PAW réel dans la liste générée par le classificateur neuronal est ensuite trouvé à l'aide de l'approche structurelle en repésentant les PAWs sous forme de chaînes à partir desquelles elle choisie le plus proche.

Cependant, ce travail ne représente qu'un simple maillon dans une chaîne de travaux qui doivent être effectués pour arriver à mettre en oeuvre un système opérationnel pour la lecture des textes arabes manuscrits non-contraints.

Les perspectives de poursuite de ce travail sont nombreuses. Les plus importants sont :

- D'aprés le premier jeu de tests effectués, la majorité des erreurs de reconnaissance peuvent être attribuée à la phase d'extraction de caractéristiques où les méthodes employées donnent, dans certains cas, des fausses alertes. Donc, si on pourra dans les prochaines années édifier un module d'extraction robuste, on a le droit de dire que notre système peut être commercialisé, du moins pour une application à vocabulaire limité.

- Bien que la combinaison de classifieurs ait été implicitement invoquée dans l'approche développée pour la reconnaissance des mots arabes manuscrits, nous estimons que tout système robuste de reconnaissance automatique de l'écriture doit faire usage de cette technique. En effet, il est légitime de s'attendre à ce que plusieurs classifieurs travaillent sur des représentations distinctes de la forme. Il a été observer pour plusieurs applications que lorsqu'il s'agit de trouver le meilleur système de reconnaissance parmi un ensemble proposé, les erreurs de classification commises par ces systèmes n'étant pas nécessairement les mêmes, laissant ainsi penser qu'il est possible d'exploiter les informations complémentaires mises en jeu par ces systèmes pour créer un système plus performant.

En conclusion, on peut envisager dans un proche avenir que l'ordinateur atteindra des performances égales à l'homme dans la reconnaissance de l'écriture manuscrite arabe omniscripteur hors ligne.

LISTE DES FIGURES

LISTE DES TABLEAUX

BIBLIOGRAPHIE

[1] A. Belaïd et Y. Belaïd, *Reconnaissance des formes méthodes et applications*, InterEdition, 1992.

[2] J.F Jodouin, *Les réseaux neuro-mimétiques méthodes et applications*, Hermès, 1994.

[3] Yi-tzuu Chien, *Interactive pattern recognition*, Dekker, 1980.

[4] T. Pavlidis, *Structural pattern recognition*, Springer Verlag, 1977.

[5] R.C. Gonzalez et P. Wintz, *Digital image processing*, Adison-wesley,1987.

[6] M. Kunt, *Traitement de l'information : reconnaissance de formes et analyse de Scènes*, Volume 3, 2000.

[7] H.A. Beghdadi et M. Senouci, *Réseaux de neurones théorie et pratique*, Office Des Publications Universitaire, 2005.

[8] P. Fabre, *Exercices de reconnaissance des formes par ordinateur*, Masson, 1989.

[9] B. Gosselin, *Classification et reconnaissance statistique de formes*,TCTS,2000.

[10] L. Fausett, *Neural networks : architectures, algorithms and applications*, Prentice Hall, 1993.

[11] P. Dargenton, "*contribution à la segmentation et à la reconnaissance de l'écriture manuscrite* ", Thèse de doctorat de l'institut de science appliquée de lion, 1994.

[12] A. Menasria, " *Reconnaissance d'écriture manuscrite par technique à base de réseaux de neurones* ", Thèse de Magister de l'université de constantine, 2005.

[13] N. Zermi, " *Reconnaissance de mots manuscrits arabes par un modèle neuro-markovien* ", Thèse de Magister de l'université Badji Mokhtar-Annaba, 2000.

[14] A. Benouareth, " *Reconnaissance de l'écriture arabe manuscrite par une approche hybride* ", Thèse de Magister de l'université Badji Mokhtar-Annaba, 2000.

[15] O. Hachour, " *Reconnaissance Hybride des Caractères Arabes Imprimés* ", *JEP-TALN 2004, Traitement autmatique de l'arabe*, Fès, 20 Avril 2004.

[16] H. Zouari, L. Heutte, " *Un Panorama des Méthodes de Combinaison de Classifieurs en Reconnaissance de Formes* ", *in Proc. RFIA'2002*, Angers, France, vol. 2, pp 499-508, 2002.

[17] J. Flusser, T. Suk, " *Affine Moment Invarients : A New Tool for Character Recognition* ", *Pattern Recognition Letters*, vol. 15, pp. 433-436, 1994.

[18] M.T. Musavi, K.H. Chan, " *On the Generalization Ability of Neural Network Classifiers* ", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, NO. 6, JUNE 1994.

[19] M.D. Garris, D.L. Dimmick, " *Form disign for High Accuracy Optical Character Recognition* ", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, VOL, 18, NO. 6, JUNE 1996.

[20] G. Seni, R.k. Srihari, N. Nasrabadi, " *Large Vocabulary Recognition of On-Line Handwriten Cursive Words* ", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, VOL. 18, NO. 7, JULY 1996.

[21] J. Hu, M. Brown, W. Turin, " *HMM Based On-Line Handwriting Recognition* ", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, VOL. 18, NO. 10, OCTOBER 1996.

[22] P.D. Gader, M.A. Khabou, " *Automatic Feature Generation for Handwriten Digit Recognition* ", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, VOL 18, NO. 12, DECEMBER 1996.

[23] S. Cho, " *Neural-Network Classifiers for Recognizing Totally Unconstrained Handwriting Numerals* ", *IEEE Transactions on Neural Networks*, VOL. 8, NO. 1, JANUARY 1997.

[24] B-K. Sin, J. Kim, " *Ligature Modeling for Online Cursive Script Recognition* ", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, VOL. 19, NO. 6, JUNE 1997.

[25] J. Cai, Z-Q. Liu, " *Integration of Structural and Statistical Information for Unconstrained Handwritten Numeral Recognition* ", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, VOL. 21, NO. 3, MARCH 1999.

[26] A. Jain, R. Duin, J. Mao, " *Statistical Pattern Reognition : A Review* ", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, VOL. 22, NO. 1, JANUARY 2000.

[27] Y. Bengio, R. De Mori, G. Flammia, " *Global Optimization of a Neural Network-Hidden Markov Model Hybrid* ", *IEEE Transactions on Neural Networks*, VOL. 3, NO. 2, MARCH 1992.

[28] H. Boulard, C. Wellekens, " *Links Between Markov Models and Multilayer Perceptrons* ", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, VOL. 12, NO. 12, DECEMBER 1990.

[29] M. Chen, A. Kundu, " *Off-Line Handwritten Word Recognition Using a Hidden Markov Model Type Stochasti Network* ", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, VOL. 16, NO. 5, MAY 1994.

[30] W. Andrew, A. Robinson, " *An Off-Line Cursive Handwriting Recognition System* ", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, VOL. 20, NO. 3, MARCH 1998.

[31] I. Bazzi, R. Schwartz, J. Makhoul, " *An Omnifont Open-Vocabulary OCR System for English and Arabic* ", *IEEE Transactions on Pattern Analysis and Machine ntelligence*, VOL. 21, NO. 6, JUNE 1999.

[32] M. Magdi, P.Gader, " *Handwritten Word Recognition Using Segmentation-Free HMM and Segmentation-Based Dynamic Programming Techniques* ", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, VOL. 18, NO. 5, MAY 1996.

[33] N. Arica, F. Yarman-Vural, " *Optical Character Recognition for Cursive Handwriting* ", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, VOL. 24, NO. 6, JUNE 2002.

[34] L. Oukhellou, P. Aknin, " *Modified Fourier Descriptors* ", *III International Workshop on Advences in Signal Processing for Non Destructive Evaluation of Materials*, Quebec, 1997.

[35] J.F. Hébert, N. Ghazzali, " *Learning to Segment Cursive Words using Isolated Characters* ", *Vision Interface '99*, Trois-Rivières, Canada, 19-21 May.

[36] A. Bennasri, A. Zahour, B. Taconet, " *Extraction des Lignes d'un Texte Manuscrit Arabe* ", *Vision Interface '99*, Trois-Rivières, Canada, 19-21 May.

[37] T. Paquet, M. Avila, C. Olivier, " *Word Modeling for Handwritten Word Recognition* ", *Vision Interface '99*, Trois-Rivières, Canada, 19-21 May.

[38] T. Frêche, N. Vincent, " *Some New Global Parameters to Qualify the Handwriting* ", *Vision Interface '99*, Trois-Rivières, Canada, 19-21 May.

[39] K. Belkacem-Boussaid, A. Beghdadi, "*A Contour Detection Methode Based on some Knowledge of the Visual System Mechanisms* ", *Vision Interface '99*, Trois-Rivières, Canada, 19-21 May.

Résumé

Ce travail de thèse s'intéresse plus particulièrement à la réalisation d'un système de reconnaissance d'écriture arabe manuscrite dont l'intérêt est la lecture automatique des montants littéraux de chèques postaux.

L'objectif visé est de proposer une méthodologie pour la mise en œuvre de ce système. Pour ce faire, dans un premier temps une phase de prétraitement qui permet de préparer les données a été introduite, ces données sont ensuite exploitées par le module d'extraction des caractéristiques les plus pertinentes pour la reconnaissance. Les résultats de cette étape sont transformés en séquences d'informations sous forme de vecteurs. Ces derniers se servent comme entrées au module de reconnaissance qui est basé sur la combinaison de deux approches : connexionniste et structurelle. Une classification des données a été effectuée par ce dernier.

Abstract

This work of thesis is interested more particularly in the realization of a system of recognition of handwritten Arab writing whose interest is the automatic reading of the literal amounts of postal cheques.

The had aim is to propose a methodology for the implementation of this system. With this intention, initially a phase of pretreatment which makes it possible to prepare the data was introduced, these data are then exploited by the module of extraction of the most relevant characteristics for the recognition. The results of this stage are transformed into sequences of information in the form of vectors. The latter are used as entered for the module of recognition which is based on the combination of two approaches: connexionnist and structural. A classification of the data was carried out by this last.

REMERCIEMENTS

Mes remerciements les plus vifs sont tout d'abord adressés à Monsieur **A.H. Bennia** pour avoir accepté de diriger mes recherches, pour ses conseils si précieux et ses critiques constructives.

Je tiens également à exprimer ma reconnaissance aux membres du jury :

Monsieur **A. Charef**, Professeur à l'université de constantine pour l'honneur qu'il me fait en acceptant de présider le jury.

Monsieur **M. Khamadja**, Professeur à l'université de constantine pour l'honneur qu'il me fait en acceptant d'examiner ce travail.

Monsieur **Menssouri**, Professeur à l'université de constantine pour l'honneur qu'il me fait en acceptant de faire partie du jury.

Que toutes les personnes qui ont contribué de près ou de loin, directement ou indirectement à ce travail, trouvent ici le témoignage de ma profonde reconnaissance.

ملخص

في هذه الأطروحة تم التركيز أساسا على إنجاز نظام قصد التعرف على الكتابة الخطية للغة العربية بطريقة أوتوماتيكية وذلك بأخذ حالة خاصة والتي تتمثل في معالجة الكتابة الحرفية للصكوك البريدية .

من أجل ذلك ، في بادىء الأمر قمنا بعرض المرحلة الأولى والتي تتمثل في معالجة الصورة التي تم إدخالها إلى الجهاز عن طريق فاحص الصور السكانير أو آلات التصوير الرقمية .

المرحلة التالية تتمثل في استخلاص الخصائص المميزة لكل كلمة وذلك من أجل استخدامها في ما بعد بواسطة المصنف وهو عبارة في حالتنا هذه عن دمج بين طريقتين مستخدمتين في هذا الميدان وهما شبكة العصبونات الصناعية وتلك المؤسسة على تركيب وتتابع أجزاء الكتابة .

يتم التصنيف باستعمال الخصائص المشتركة من الحروف أو الكلمة والتي هي مدخل المصنف وأسماء الحروف أو الكلمات هي مخرج المصنف .