

THESE DE DOCTORAT

Pour obtenir le grade de :
DOCTEUR DE L'UNIVERSITE DE TOULON ET DU VAR

Discipline : Optique, Image et Signal

Présentée par :
Audrey SEROPIAN

**Analyse de Document
et
Identification de Scripteurs**

soutenue le 18 décembre 2003 devant le jury composé de :

M.	Michel	Grimaldi	Université de Toulon	Codirecteur
M.	Jacques	Lemaitre	Université de Toulon	Examineur
M.	Thierry	Paquet	Université de Rouen	Rapporteur
M.	Patrick	Paysan	Neurone Informatique	Invité
M.	Bruno	Rossetto	Université de Toulon	Examineur
M.	Georges	Stamon	Université de Paris5	Rapporteur
Mme	Nicole	Vincent	Université de Paris5	Directrice de thèse

Résumé

Les problèmes d'indexation de documents et d'authentification des scripteurs sont des problèmes de plus en plus fréquents. Le thème de notre étude se limite à des formulaires dont certains champs sont remplis à l'aide d'une écriture manuscrite. Après avoir élaboré un modèle de formulaire utilisable dans le contexte d'un nombre fini de scripteurs, notre but est ici de créer un module d'identification de scripteur par analyse fractale du style de son écriture. Cette identification a un double but. Elle nous permet d'une part d'identifier la personne qui a écrit, et d'autre part d'adapter au scripteur le module de reconnaissance de son écriture manuscrite non contrainte. L'identification de chaque scripteur repose sur l'extraction d'un ensemble de caractéristiques qui doivent être propres à l'auteur du document. Les propriétés d'autosimilarité dans l'écriture sont exploitées. Pour cela, des formes invariantes caractérisant l'écriture d'un scripteur sont extraites au cours d'un traitement proche du processus de la compression fractale. Ces formes sont ensuite organisées dans une base de référence permettant d'analyser une écriture inconnue dans un processus de Pattern Matching. Les résultats de l'analyse sont évalués par le rapport signal/bruit. Ainsi, il permet en fonction d'un ensemble de bases de référence, d'identifier le scripteur d'un texte dont on cherche l'identité de l'auteur. Enfin, Nous montrons comment la méthode développée peut être appliquée à de nombreux autres problèmes.

Mots clés : identification de scripteur, analyse de documents, extraction de similarités, Pattern Matching, compression fractale.

Abstract

The problems of the indexing of documents and the authentication of writers are problems more and more frequent. The axe of our study is limited to forms some areas of which are filled by cursive handwriting. After we elaborated a model of form used in a context of finite number of writers, our aim is here to create a process of identification of writer through fractal analysis of handwritten style writer. This identification has got a double goal. We can on the one hand identify the person who wrote and on the other hand adapt a process of unconstraint handwritten recognition to the writer. The identification of each writer depends on the extraction of a set of features that have to be intrinsic of the author of the document. The properties of autosimilarity in the handwriting are used. Some invariant patterns are extracted by the process of fractal compression to characterize the handwriting of the writer. These patterns are organized in a reference base to allow the analyze of an unknown handwriting through a Pattern matching process. The results of this analyze are evaluated by the signal to noise ratio. We can identify the text of a writer that we look for the identity of the author according to the set of reference bases. At last, we show how our method could be use to many others problems.

Keywords : writer identification, analysis of documents, extraction of similarities, Pattern Matching, fractal compression.

TABLE DES MATIERES

Introduction Générale	8
1. Contexte Industriel	13
1.1. Le problème	13
1.2. Elaboration des fiches	16
1.3. Analyse de document	17
1.3.1. Différenciation des fiches	17
1.3.2. Recalage	19
1.3.3. Détection des croix	22
1.3.4. Extraction de l'écriture manuscrite	26
1.4. Conclusion	29
2. Identification de scripteur - Etat de l'art	31
2.1. Introduction	31
2.2. Approches classiques	32
2.2.1. Approches contextuelles	33
2.2.2. Approches non contextuelles	34
2.3. Méthodes d'identification par styles	39
2.3.1. Modélisation d'allographes	39
2.3.2. Méthode globale des styles	40
2.4. Détermination des invariants	44
2.5. Conclusion	46
3. Identification par compression fractale	49
3.1. Compression fractale	50
3.1.1. Rappels théoriques	50
3.1.2. IFS (Iterative Function System)	51
3.1.3. Compression des images naturelles	55
3.1.3.1. Méthode de A. Jacquin	58
3.1.3.2. Méthode de Y. Fisher	60
3.1.4. Décompression	62
3.1.5. Application à l'écriture	62
3.2. Mise en place des données et des paramètres	64
3.2.1. Acquisition des textes manuscrits	64
3.2.2. Les prétraitements	65
3.2.3. Paramètres théoriques	71
3.3. Apprentissage d'un scripteur	77
3.4. Identification du scripteur	80
3.4.1. Recherche de similarités	80
3.4.2. Reconstruction de l'image	84
3.5. Résultats	84
3.6. Conclusion	89

4. Autres Applications	92
4.1. Introduction	92
4.2. Reconnaissance d'écriture	93
4.2.1. Rappels bibliographiques	93
4.2.1.1. Approche analytique	94
4.2.1.2. Approche globale	95
4.2.1.3. Autres Approches	96
4.2.2. Notre méthode de reconnaissance	98
4.2.2.1. Accumulation d'hypothèses	98
4.2.2.2. Résultats	100
4.2.3. Conclusion	103
4.3. Différenciation de deux alphabets	104
4.4. Vérification de signatures	108
4.5. Caractéristiques d'images quelconques	112
4.6. Conclusion	115
Conclusion Générale	118
Annexe 1	121
Annexe 2	122
Annexe 3	123
Bibliographie	124
Publications de l'auteur	135

TABLE DES ILLUSTRATIONS

Figures

Figure 1.1 : Fiches d'analyse élaborées pour l'entreprise	16
Figure 1.2 : Exemples de logos se trouvant sur les bords des fiches	17
Figure 1.3 : Corrélation entre le logo de l'image scannée et les deux logos de référence	18
Figure 1.4 : Résultat des corrélations entre les logos et l'image de référence en (a) et l'image scannée en (b)	20
Figure 1.5 : Image reconstruite	21
Figure 1.6 : Etape de détection des cases cochées	22
Figure 1.7 : Ensemble de croix constituant un ensemble d'apprentissage	23
Figure 1.8 : Résultat de la corrélation entre l'image de test et la croix de référence	24
Figure 1.9 : Densité de probabilité des cases cochées (en rouge) et des cases vides (en bleu)	24
Figure 1.10 : Partie manuscrite des fiches (a) vide et (b) remplie	27
Figure 1.11 : Résultat de la récupération de l'écriture manuscrite (a) et après filtrage (b)	27
Figure 1.12 : Projection (a) verticale et (b) horizontale du manuscrit	28
Figure 1.13 : Récupération des écritures dans les différents champs	29
Figure 2.1 : Extraction de caractéristiques	40
Figure 2.2 : (a) deux montants littéraux, (b) leur diagramme d'orientation, (c) parties rectilignes du tracé, et (d) les 4 lobes d'orientation	41
Figure 2.3 : Distribution de l'écriture selon les dimensions fractales extrait de [VIN95a]	43
Figure 2.4 : Classement de l'écriture par compression fractale	44
Figure 2.5 : Exemple d'invariants extraits d'un texte de [NOS02]	46
Figure 3.1: Génération d'image fractale	55
Figure 3.2 : Méthode de la compression fractale avec (a) la représentation théorique, et (b) la représentation appliquée à une image	57
Figure 3.3 : Exemple d'image d'écriture scannée à 200 ppp	63
Figure 3.4 : Image précédente décompressée après compression fractale, avec un PSNR de 39,51dB	64
Figure 3.5 : Exemple d'écriture non rectiligne	68
Figure 3.6 : Estimation de l'angle de la ligne de base	68
Figure 3.7 : Image redressée du texte présenté sur la Figure 3.5	69
Figure 3.8 : Exemple d'image déjà dilatée pour la recherche d'éléments connectés	70
Figure 3.9 : Exemple de dilatation	70
Figure 3.10 : Isolement d'un mot avec les étapes successives suivantes (a) binarisation du manuscrit, (b) dilatation du texte, (c) extraction des composantes, (d) récupération d'un mot	71
Figure 3.11 : Exemple de notre méthode de découpage	73
Figure 3.12 : Mot découpé de façon régulière (a), selon la méthode proposée (b)	73
Figure 3.13 : Qualité de la décompression en fonction de la taille des Ranges	74
Figure 3.14 : Nombre d'images dans la base par rapport à la taille des Ranges	75
Figure 3.15 : Découpage de la lettre e par Ranges de taille 8x8 pour deux scripteurs différents après normalisation	75
Figure 3.16 : Nombre d'images dans la base en fonction du nombre de mots dans le texte	78
Figure 3.17 : Exemple de base associée à un scripteur	80
Figure 3.18 : Identification par compression et décompression fractales	81
Figure 3.19 : Résultats de la reconstruction de (a) l'image de test à partir de (b), (c), (d), (e) différentes bases extraites pendant la phase d'apprentissage	82
Figure 3.20 : Recherche de similarités par rapport à la base de référence.	83
Figure 3.21: Textes d'apprentissage.	85
Figure 3.22 : Bases extraites des trois scripteurs (a), (b) et (c) respectivement	86
Figure 3.23 : Ecritures utilisées pour l'identification.	87
Figure 3.24 : Images résultant du processus d'identification du scripteur texte 3.23(a).	88
Figure 4.1 : Système complet de reconnaissance d'écriture	93
Figure 4.2 : Système complet de reconnaissance d'écriture extrait de [COT97]	98
Figure 4.3 : Étiquetage des positions de la lettre c	99
Figure 4.4 : Récapitulatif de toute la méthode	101
Figure 4.5 : Étiquetage des lettres après découpage, avec signalement des lettres déjà faites	102
Figure 4.6 : Résultat de la reconnaissance	103
Figure 4.7 : Textes d'apprentissage de deux alphabets différents	104

Figure 4.8 : Bases extraites des deux textes d'apprentissage	105
Figure 4.9 : Deux textes de tests	105
Figure 4.10 : Image de test en français reconstruite avec (a) la base française et (b) la base arabe	106
Figure 4.11 : Image de test en arabe reconstruite avec (a) la base française et (b) la base arabe	107
Figure 4.12 : Texte d'apprentissage et de test écrit par une personne bilingue	107
Figure 4.13 : Extraction de caractéristiques sur l'écriture par la compression fractale	108
Figure 4.14 : Exemple de signatures	109
Figure 4.15 : Exemplaires de signatures de S2	109
Figure 4.16 : Evolution des images décompressées pour 2 itérations avec pour image de départ (a) la signature correspondant à la base utilisée, (b) une signature quelconque, et (c) un autre exemplaire de la signature (a)	110
Figure 4.17 : Exemples d'images	112
Figure 4.18 : Exemples d'images de (a) Lena (b) avec rajout d'un bruit uniforme ou (c) décalée de 10 pixels vers la droite	115

Tableaux

Tableau 1.1 : Comparaison des deux critères de détection des cases cochées	25
Tableau 1.1 : Nombre de pixels pour presque toutes les cases de la Figure 1.7	26
Tableau 2.1 : Résultats de l'identification de scripteurs par la moyenne des mesures des histogrammes vertical et horizontal extraits de [ARA77]	35
Tableau 3.1 : Résultats des images décompressées	86
Tableau 3.2 : Résultats du processus d'identification de scripteur avec les valeurs des PSNR.	89
Tableau 3.3 : Résultats du processus d'identification de scripteur avec les valeurs des coefficients de corrélation.	89
Tableau 4.1 : Tableau regroupant les informations statistiques des lettres pour un scripteur donné	100
Tableau 4.2 : Résultats des calculs du coefficient de corrélation sur la différenciation d'alphabets	106
Tableau 4.3 : Résultats dans le cas d'une personne bilingue	108
Tableau 4.4 : Résultats de la vérification de signatures	111
Tableau 4.5 : Résultats de l'identification d'images quelconques	113
Tableau 4.6 : Résultats de la ressemblance d'images quelconques par leur base	114
Tableau 4.7 : Tests sur la robustesse avec calcul de V	115

Introduction Générale

Introduction Générale

Dans notre vie quotidienne le nombre des documents que nous utilisons va sans cesse croissant. Même si aujourd'hui ces documents prennent le plus souvent une forme électronique, il est encore bien des cas où un document, par exemple un formulaire, est en partie manuscrit. Malgré les progrès quotidiens de la technologie dans le domaine de la miniaturisation, il demeure que le stylo reste toujours le moyen d'écriture le plus mobile et qui ne nécessite aucune source d'énergie extérieure au scripteur pour être opérationnel. D'autre part le bon usage des documents suppose que ceux-ci se trouvent à disposition au moment où ils doivent servir. Le bon aiguillage des documents pour la réalisation de la tâche les concernant et leur rangement devient de plus en plus délicat et dans bien des cas une automatisation de la classification de ces documents serait bénéfique. Dans cet objectif, une reconnaissance de la nature des documents et de leur contenu est nécessaire [CLA01] [KEB98].

Le problème de l'authentification d'un document est né en même temps que naissait la notion même de document. L'identification du scripteur représente une phase cruciale de cette authentification et remonte pratiquement à la naissance de l'écriture. De nos jours et dans notre environnement, la nécessité de valider un texte écrit est un problème récurrent, il n'est pas limité au monde juridique dans lequel il est fondamental de pouvoir authentifier l'auteur d'un document, d'un testament, ou d'un acte de vente par exemple. Il est beaucoup d'autres circonstances où l'authentification est indispensable. A ce titre, le sceau, la signature ou le mot de passe a été introduit pour particulariser les documents. Ces moyens représentent une première phase permettant une authentification ultérieure, mais qui n'est pas toujours effectuée au moment de la réalisation du document. Dans tous les cas, c'est à partir d'une

caractérisation du scripteur que l'authentification peut être réalisée. On fait alors appel à des experts en graphologie ou en écriture qui examinent un ensemble de points caractéristiques de l'écriture et d'écritures de références. Ces comparaisons les conduisent à décider ou non quelle est l'origine de l'écriture.

Depuis quelques années, il est un autre domaine où l'identification du scripteur est devenue un élément important, c'est celui de la reconnaissance automatique de l'écriture [LEC94][CRE96][NOS02]. En effet, la variabilité importante des écritures, d'un scripteur à l'autre, rend le problème de la reconnaissance particulièrement difficile. Dans le cas de la reconnaissance d'écriture manuscrite dite « off line » qui nous intéresse ici, des solutions existent dans des cas bien particuliers. On peut en noter trois principalement. Tout d'abord le cas d'une approche mono-scripteur où la variabilité, si elle est encore notablement plus grande que dans le cas omni-scripteur, est plus réduite. De même, dans le cas où l'on se limite à un vocabulaire réduit, de bons résultats ont été obtenus comme pour la lecture des montants littéraux sur les chèques par exemple. Enfin si l'on impose au scripteur des contraintes de forme des lettres à respecter on peut également obtenir des taux de reconnaissance tout à fait utilisables dans les applications. En revanche, si l'on considère le cas omni-scripteur, le problème n'a toujours pas reçu de solution satisfaisante. Une des voies de recherche consiste à limiter la phase de reconnaissance pure à des sous-ensembles définis après réduction de la variabilité des formes qui interviennent dans l'écriture. L'utilisation d'un processus identifiant le scripteur concerné, ou son style, permettrait d'effectuer de manière globale une approche multi-scripteurs tout en s'adaptant automatiquement à chaque individu ou aux caractéristiques du style reconnu pour l'écriture. Ainsi, la résolution de divers problèmes identiques mais plus simples permettrait d'apporter une solution générale.

Plusieurs études ont été menées pour apporter une solution à ce problème. De manière implicite, les styles sont très naturellement repérables par le lecteur humain qui sans lire un document, identifie facilement l'identité de son auteur si ce dernier fait partie de ceux dont il connaît l'écriture. « Connaître une écriture », c'est en quelque sorte savoir identifier son scripteur. Le style, quant à lui, recouvre une notion plus globale qui concerne un ensemble de scripteurs pour lesquels on peut constater une ressemblance plus ou moins importante des écritures. Le niveau d'observation est assez global pour que l'on ne rentre pas précisément dans les détails de l'écriture. Les styles peuvent être définis par des caractéristiques de formes locales fréquentes dans l'écriture concernée [CRE95]. J.-C. Simon désigne par invariants ces caractéristiques de l'écriture.

Ces éléments invariants, que l'on peut observer au sein de l'écriture, sont de natures variées [KHA96]. Ils peuvent être géométriques (boucles, lignes droites verticales, etc.) et/ou topologiques (points de croisement, points extrêmes, etc.). Durant la dernière décennie, plusieurs études ont été menées, parmi lesquelles il convient de citer celles de J.-P. Crettez [CRE94] et de L. Heutte [HEU00]. Ces auteurs ont extrait les propriétés spécifiques de chaque écriture en utilisant des attributs comme la pente ou le nombre de composants connexes dans le contour du texte. N. Vincent dans [VIN94], [VIN98] et [VIN00] a développé une approche plus globale montrant que les images d'écriture ont un comportement fractal. La dimension fractale apparaît comme étant un paramètre robuste, constant pour chaque scripteur et qui varie continûment avec certaines caractéristiques du style.

L'extraction de formes invariantes d'une écriture [NOS99] constitue une avancée majeure dans le processus d'identification d'un scripteur. L'auteur établit que les invariants sont les caractéristiques propres à chaque scripteur. Cette technique s'apparente à l'expertise graphologique.

Dans notre étude, nous nous intéressons à un système global de traitement d'une chaîne d'utilisation de documents manuscrits dans un contexte spécifique. Si nous tentons d'utiliser autant que possible les particularités de l'application pour atteindre une efficacité maximum, nous gardons néanmoins toujours à l'esprit la volonté d'assurer la généralisation de nos développements pour des problèmes voisins.

La chaîne de traitement comporte, assez classiquement, différents stades. Tout d'abord le formulaire papier vierge est rempli, sur son lieu d'exercice, par un scripteur. Le document est transféré dans un autre service où il doit être traité, donc classé dans un premier temps. Cette étape est réalisée par un transfert électronique du document qui a été scanné en une image de résolution suffisante, en fonction des besoins ultérieurs. Ensuite seul le document numérique est utilisé. Une authentification de ce document peut déclencher une action qui dépend du contenu sémantique de la zone manuscrite. Cette zone devra être reconnue de manière automatique.

Dans une première partie nous verrons comment une élaboration spécifique des formulaires peut simplifier leur classification dans la phase d'utilisation et comment nous avons réalisé cette classification reposant sur l'aspect physique du document. Il vient ensuite l'étape d'identification du scripteur, qui représente la plus grande partie de ce mémoire. Elle permettra d'obtenir des informations sur chaque individu à partir de quelques caractéristiques, dans notre cas des formes invariantes, extraites du texte manuscrit. La méthode mise en œuvre, dérivée de la technique de la compression fractale, est utilisée pour prendre en compte

les autosimilarités d'une image. Une étape de « Pattern Matching » permet ensuite de mettre en relation ces similarités. La dernière partie sera consacrée à la phase de reconnaissance de l'écriture des individus identifiés. Elle reposera en grande partie sur les caractéristiques extraites pendant la phase d'apprentissage et d'authentification. Diverses applications qui peuvent être faites à partir de ces caractéristiques fractales seront présentées.

Chapitre 1

Contexte industriel

1. Contexte Industriel

1.1. Le problème

Le problème qui nous est posé se place dans un contexte hospitalier. Au sein de l'hôpital, les médecins sont amenés, quotidiennement, à établir des ordonnances pour des examens complémentaires concernant les patients hospitalisés. Ces examens, analyses de sang, d'urine, ou autres, sont réalisés ultérieurement dans un service interne ou délocalisé de l'hôpital. De ce fait, la prescription doit changer de service, et être relue pour que les analyses demandées soient effectuées. Actuellement les ordonnances sont rédigées à la main par les médecins, scripteurs dont l'écriture a la réputation non usurpée d'être « illisible » par le non initié. Les noms sont inscrits soit en minuscules soit en majuscules, à la hâte et dans des circonstances parfois difficiles, par des scripteurs différents, avec des tailles, des inclinaisons, des formes et des instruments différents. Il est difficile, dans ce contexte, d'obliger le scripteur à écrire de façon lisible. Par contre, le nombre de scripteurs est relativement réduit, et les médecins autorisés sont parfaitement connus. L'objet de ce travail consiste à améliorer l'ensemble des traitements liés à la vie de cette ordonnance, en y introduisant, bien évidemment, le moins de contraintes possibles.

Les fiches remplies par les médecins, actuellement de deux types, correspondent respectivement à un bilan général et à un bilan spécialisé. Il est indispensable de prévoir qu'elles peuvent évoluer vers des types plus variés dans un avenir plus ou moins proche. Les informations utiles contenues dans ces fiches sont de deux natures différentes. Une première partie manuscrite comporte le nom de l'hôpital, le service demandeur (par exemple :

Urgences, Réanimations, ...), les noms du médecin traitant, et du patient concerné. Elle est remplie par le médecin lors de l'établissement de la fiche. La seconde partie comporte l'ensemble des analyses demandées pour le patient, elles sont répertoriées dans une liste prédéfinie.

Ainsi deux types de traitement assez différents doivent être envisagés. D'une part le traitement de la partie d'identification constitué principalement d'un problème de reconnaissance d'écriture et d'autre part, celui de la partie analyse, codifiée. Rappelons que cette dernière est constituée d'une liste d'analyses à effectuer qui devront avoir été validées en cochant les cases. Celles-ci devront dans un premier temps être localisées. Dans ce cas, le traitement consiste, lors de la lecture du document, à évaluer les analyses à faire en détectant si les cases sont cochées ou non.

Nous allons présenter les méthodes que nous avons étudiées et les résultats obtenus concernant d'une part la détection des cases cochées et d'autre part la récupération du texte manuscrit. L'étape suivante consistera à entamer le processus de reconnaissance d'écriture manuscrite en codant les mots sous formes de graphèmes (pseudo-lettres).

En tenant compte de l'état actuel d'utilisation du système, deux fiches d'analyses différentes ont été mises au point en collaboration avec la société NEURONE Informatique. L'objectif est double. Il est d'une part d'assurer une différenciation aisée des différents types de fiches, et d'autre part d'extraire les informations utiles. Ces informations ont été naturellement regroupées en fonction de leur sémantique. La partie haute de chacune des fiches contient le texte manuscrit et la partie basse de celles-ci contient la liste des analyses à réaliser en fonction de l'identification de la présence de croix dans les cases. Ce document détaille les choix faits de la construction même des fiches et les étapes de l'analyse de ces fiches, et présente les résultats obtenus. De plus, le prétraitement de l'image numérisée conditionne les résultats de la reconnaissance de l'écriture. Cette étape a donc elle aussi son importance.

Ainsi le déroulement du travail, de ce que nous appellerons le prétraitement se décompose en trois étapes principales :

1. Différenciation entre les deux types de fiches : «bilan général» et «bilan spécialisé».
2. Construction de la liste de toutes les analyses à faire par identification des cases cochées.
3. Mise en forme de la partie haute de la fiche scannée (nom du praticien, nom du patient, nom de l'hôpital,...) pour faciliter le processus d'identification et de reconnaissance.

Pour permettre une reconnaissance d'écriture efficace, un traitement de remise en forme de l'image acquise est nécessaire. Cette mise en forme consiste à segmenter les portions de l'image limitées aux écritures manuscrites situées dans la partie supérieure et à localiser les croix dans les cases sur les fiches remplies.

Dans les sections qui suivent nous allons tout d'abord exposer comment nous avons choisi les éléments qui nous permettront de classifier les fiches et indiquer une première connaissance sémantique sur le contenu de la fiche. Ensuite nous préciserons les traitements de bas niveau qui conditionnent la robustesse de la méthode générale. La lecture des croix dans les cases sera également exposée dans le présent chapitre.

L'analyse de documents est un domaine de recherche qui ne cesse de s'accroître avec le développement de l'informatique. En effet, la Gestion Electronique de Documents (GED) générale est un problème complexe [NAG00] dû au grand nombre de documents différents que ce soit dans un même service [CES03] ou encore plus d'un pays à un autre. La plupart des travaux de recherches dans le domaine porte sur les problèmes de structuration de documents [HER01]. Les auteurs n'ont pas de connaissances *a priori* sur la structure physique des documents [DUO02] [SHI03]. Ainsi avant de procéder à des traitements, un repérage des zones de textes et d'images doit être fait [FAI02]. Lorsqu'il existe une connaissance *a priori* de la structure, les traitements sont alors adaptés à chaque contexte. Les techniques utilisées [JAI00] sont souvent celles de la reconnaissance de formes (les coefficients de Kharunen-Loève, les opérateurs morphologiques, le calcul des moments, les composantes connexes [CHE01]...) pour l'extraction de caractéristiques et celles des statistiques (loi de Bayes, réseau de neurones, ...) pour la prise de décision.

Notre problème est plus simple que celui des systèmes de Gestion Electronique de Documents (GED) et même des systèmes particuliers pour lesquels des études poussées ont été développées comme dans le cas des Caisses d'Allocations Familiales [CLA01] [KEB98], celui des partitions musicales [COU96], celui des références bibliographiques [PAR98] ou celui des caractères multi-échelles [ADA01]. En effet, nous avons d'une part une connaissance *a priori* de la forme de nos documents et d'autre part les informations à extraire sont seulement de deux types dont une constante (les cases à détecter).

1.2. Elaboration des fiches

Les documents doivent être composés d'une seule feuille de papier. Chaque fiche est de format A4. Le contenu des fiches, c'est-à-dire les renseignements imprimés, est décidé en collaboration avec les membres de l'entreprise et les médecins. Nous avons choisi de ne pas modifier cette partie des fiches en conservant la disposition qui est utilisée actuellement dans l'établissement. La différenciation pourrait alors être réalisée par une analyse structurale des deux types de fiches. On voit sur la Figure 1.1 que dans la fiche « bilan général » l'écart entre les cases cochables est plus grand que dans le document « bilan spécifique ». Il nous est apparu que cette approche ne permettrait pas, de manière aisée, d'augmenter le nombre de fiches différentes susceptibles d'être traitées par la suite. Aussi nous avons choisi d'ajouter à la fiche des éléments distinctifs qui peuvent être interprétés comme des enjolivures. Cette option nous a semblé appropriée à notre application pour laquelle nous avons la totale maîtrise de la conception des fiches. Les fiches n'ont pas vocation à se multiplier, les impressions n'auront pas des origines multiples et les auteurs des fiches ne pourront les créer de manière indépendante.

(a) bilan général

(b) bilan spécialisé

Figure 1.1 : Fiches d'analyse élaborées pour l'entreprise

Pour faciliter la différenciation des deux fiches, nous ajoutons deux « logos » différents sur chacune d'entre-elles. Ces logos sont des disques constitués chacun de quatre quadrants de couleurs différentes. Deux zones sont blanches et les deux autres sont noires. Deux zones adjacentes ne peuvent avoir la même couleur.



Figure 1.2 : Exemples de logos se trouvant sur les bords des fiches

Il existe donc deux formes possibles pour ce logo. Cela est suffisant pour différencier les deux types de fiches que nous avons à gérer. Avec un objectif de généralisation possible nous avons considéré 3 emplacements pour un logo. Les 3 motifs permettront de différencier 8 types de fiches différents et cela indépendamment du contenu de la fiche. Ce sont ces éléments qui vont être utilisés pour analyser les documents. La présence de plusieurs logos sur la page nous permet en même temps que la différenciation des fiches de mieux repérer la position de la page dans l'image scannée. De plus, la non-symétrie de la disposition de ces trois logos permettrait de caractériser si nécessaire, le haut de la page.

1.3. Analyse de document

Dans cette partie, nous allons détailler l'étape d'analyse de document. Elle se décompose en 4 étapes successives : la première est la différenciation entre les deux fiches élaborées actuellement, la deuxième est le recalage de l'image numérisée par rapport à la fiche de référence, la troisième est la détection des cases cochées et la dernière étape est la récupération des zones d'écriture manuscrite.

1.3.1. Différenciation des fiches

Il s'agit de différencier les deux types de fiches. Le logo se trouvant dans l'angle supérieur gauche de chacune est utilisé. Le logo présent doit être comparé aux deux logos possibles. Les méthodes de reconnaissance de formes applicables à notre cas [SEY98] sont très nombreuses et tous les choix pourront être critiqués. Nous avons choisi de réaliser un calcul de corrélation entre la zone où se trouve le logo, extraite de l'image numérisée, et les logos de référence. La

zone choisie est une zone carrée située dans le coin supérieur gauche de l'image. Cette méthode nous permet d'obtenir simultanément d'une part la reconnaissance du logo mais aussi sa position dans l'image.

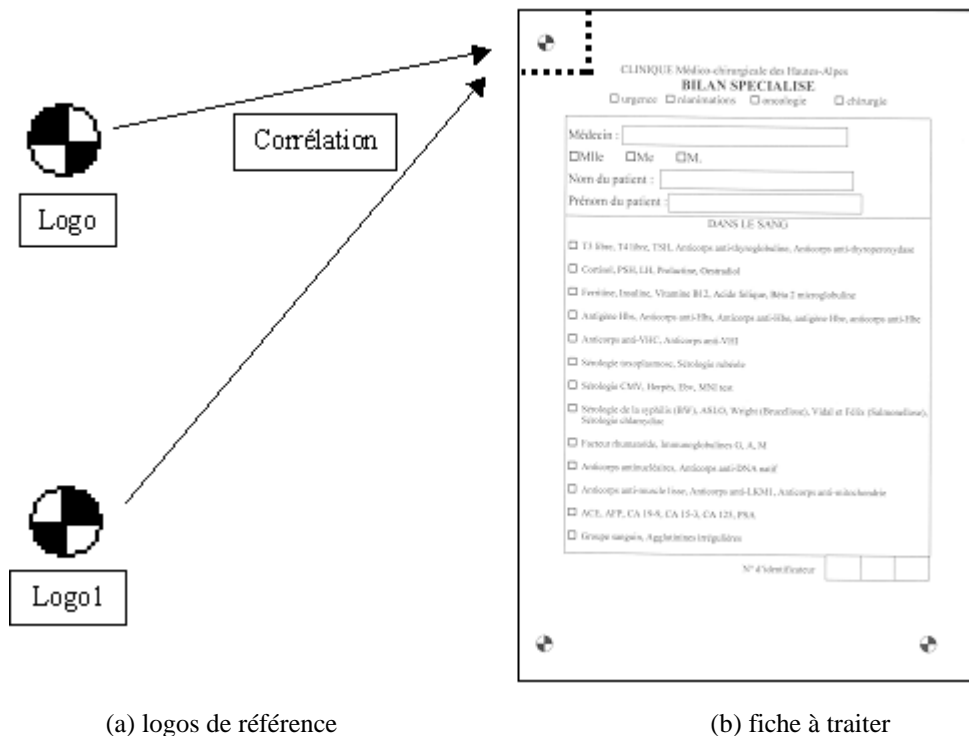
Au point de coordonnées (x,y), la définition du produit d'intercorrélation entre deux images notées F et G est donnée par :

$$C(x, y) = \sum_i \sum_j F(i, j)G(i + x, j + y)$$

Dans notre cas, F représente la zone du document scanné où se trouve le logo et G l'image du logo. Comme nous utilisons 2 logos, nous calculons deux images résultats de l'intercorrélation avec chacun des logos de référence.

Le produit de corrélation fournit un résultat compris entre 0 et 1 (0 correspond à une différence totale des images et 1 correspond à une coïncidence totale).

La présence du logo se caractérise donc par un maximum dans l'image des corrélations. Le maximum le plus important extrait des deux produits d'intercorrélation est recherché. C'est la comparaison entre les deux maximums qui nous importe. La décision de reconnaître un logo plutôt qu'un autre, est évidemment prise à condition que la valeur maximum obtenue en considérant les images représentatives de cette corrélation soit supérieure à 0,5.



(a) logos de référence (b) fiche à traiter
 Figure 1.3 : Corrélation entre le logo de l'image scannée et les deux logos de référence

Ce traitement nous amène à une des conclusions suivantes : « *Bilan général* », « *Bilan spécialisé* », « *Mauvaise feuille* ». Il est important de considérer de manière automatique une classe de rejet qui permettra de traiter de manière manuelle un document qui ne serait pas une fiche et qui ne se trouverait pas à sa place.

Cette information acquise, on connaît de par la phase de conception des fiches, la structure de celle-ci et il est possible maintenant de localiser les cases à lire. Néanmoins, nous avons choisi d'introduire une phase de prétraitement comprenant le redressement et le recalage de la feuille pour augmenter la précision des positions des cases.

1.3.2. Recalage

Lors du processus d'acquisition de l'image des fiches par le scanner, des déformations dues au mouvement de la feuille apparaissent. Il est alors nécessaire de recalibrer [YIN01] [YUA03] et de recadrer l'image scannée par rapport à une image de référence de manière à optimiser les détections des analyses et la localisation du manuscrit. La méthode choisie repose sur trois points de repère présents simultanément sur les deux images et qui doivent être mis en correspondance par l'intermédiaire d'une transformation géométrique. Le modèle de déformations choisi est un modèle affine.

Précisons les notations et le modèle choisi :

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} 1 & i & j & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & i & j \end{pmatrix} * \begin{pmatrix} a1 \\ a2 \\ a3 \\ a4 \\ a5 \\ a6 \end{pmatrix} = \begin{pmatrix} a1 + a2i + a3j \\ a4 + a5i + a6j \end{pmatrix}$$

où $\begin{pmatrix} u \\ v \end{pmatrix}$ indique les composantes du vecteur de déplacement à trouver et $\begin{pmatrix} i \\ j \end{pmatrix}$ représente les coordonnées des pixels de l'image.

Les coefficients a_k (avec $k=1, \dots, 6$) représentent le modèle de déplacement, ils définissent les variables de notre problème. Ils constituent une matrice A de déplacement. Ce modèle permet d'estimer aussi bien une translation caractérisée par les coefficients $a1$ et $a4$, qu'une rotation ou une homothétie caractérisées par les coefficients $a2$, $a3$, $a5$ et $a6$ ou qu'une combinaison des trois.

Cette méthode se caractérise par la résolution d'un système d'équations permettant de déterminer les variables a_k . Elle conduit à résoudre un système simple du premier degré de six équations à six inconnues (dite 'Estimation locale'). Le nombre des points nécessaires à la résolution du système est choisi en fonction du modèle. Chaque point fournit 2 des équations du système. Trois points sont donc nécessaires à la détermination de la transformation.

Dans cette étape nous considérons les centres des trois logos situés sur l'image scannée dans des zones qui ont été sélectionnées, et comme il a été fait pour le logo supérieur gauche dans l'étape d'indexation du document, les images de corrélation avec le logo correspondant au type de fiche détecté précédemment sont calculées. D'autre part, nous disposons d'une image de fiche scannée de manière très précise et sans déformation, c'est l'image de référence. Il est procédé de même pour cette image de référence de manière à assurer une comparaison entre la situation idéale et la situation réelle, c'est-à-dire qu'une image de corrélation avec le logo est déterminée dans chacune des 3 zones.

La Figure 1.4 montre les images résultant de la corrélation entre le logo de référence et la zone extraite de l'image de référence d'une part (Figure 1.4(a)) et celle de l'image scannée d'autre part (Figure 1.4(b)). Les valeurs obtenues sont représentées par des couleurs en niveau de gris. Les valeurs faibles sont en noir et les valeurs plus fortes sont en blanc.

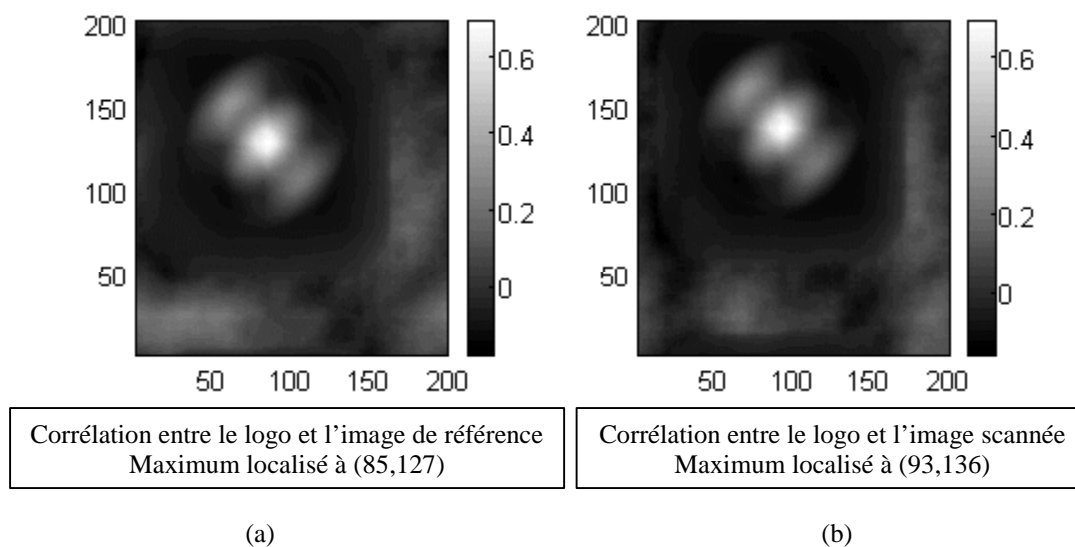


Figure 1.4 : Résultat des corrélations entre les logos et l'image de référence en (a) et l'image scannée en (b)

Sur chacune des 3 images de corrélation semblables à celle représentée Figure 1.4(b) les coordonnées (x, y) des maximums sont extraites. De même sur les 3 images de corrélation associées à l'image de référence on a calculé dans la phase de mise en place du système les coordonnées des maximums. Ainsi le système suivant est construit :

$$\begin{bmatrix} X_i \\ Y_i \end{bmatrix} = A \times \begin{bmatrix} x_i \\ y_i \end{bmatrix} \quad \text{Avec } i = (1,2,3)$$

où (X,Y) sont les coordonnées des maximums de la corrélation entre la zone du logo et l'image de référence.

Le seul élément inconnu de ce système est la matrice A de mouvement. C'est grâce aux coordonnées de ces points caractéristiques des fiches que les coefficients de cette matrice sont calculés. La transformation associée pourra ensuite être appliquée à toute l'image scannée pour assurer le redressement de la page.

La Figure 1.5 présente le résultat de l'application globale de la transformation, c'est-à-dire l'image reconstruite. Nous pouvons noter que quelques défauts apparaissent sur cette image reconstruite. Ils sont dus au fait que les coefficients de la matrice de mouvement A estimés, ne sont pas des valeurs entières alors que la transformation que nous avons appliquée sur l'image scannée est à valeurs entières. La différence entre les résultats obtenus par les transformations entières et réelles existe dans pratiquement tous les cas. Nous n'avons néanmoins pas jugé nécessaire d'avoir recours à la précision de la transformation réelle pour la suite de nos traitements, car cela aurait rendu le temps de traitement trop long.

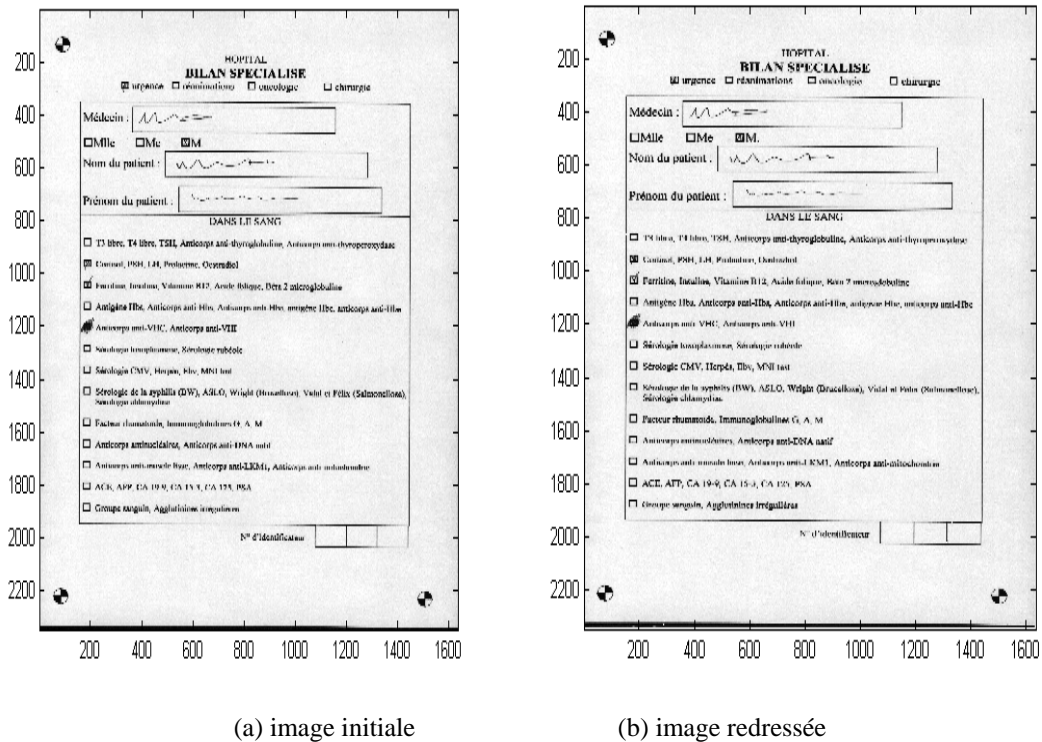


Figure 1.5 : Image reconstruite

On peut noter que la différence n'est pas visible à l'œil nu sur la page mais seulement notable au niveau des pixels.

Ces quelques transformations de l'image constituent en quelque sorte une phase de prétraitement destinée à rendre plus aisée et plus robuste, l'étape de détection des croix dans les cases définissant les analyses à sélectionner dans la liste.

1.3.3. Détection des croix

L'image ainsi recalée, le problème de la détection des cases cochées se pose en terme de prise de décision. La position des croix est connue dans l'image de référence associée à la fiche dont on connaît maintenant la nature. Il suffit de répondre à la question binaire : « Y a-t-il présence de croix dans la case ? ».

La superficie occupée par la croix est généralement inférieure à celle de la case (elle occupe approximativement le quart de celle-ci). Nous avons choisi cette taille de référence pour être certaine de prendre au maximum en considération les différents types de croix possibles. En effet, comme le montre la Figure 1.6, les façons de marquer une case sont multiples, une case peut être cochée par deux barres croisées comme dans le modèle de croix utilisé, mais aussi dans certains cas par seulement deux barres simples voire même par un noircissement de la case.

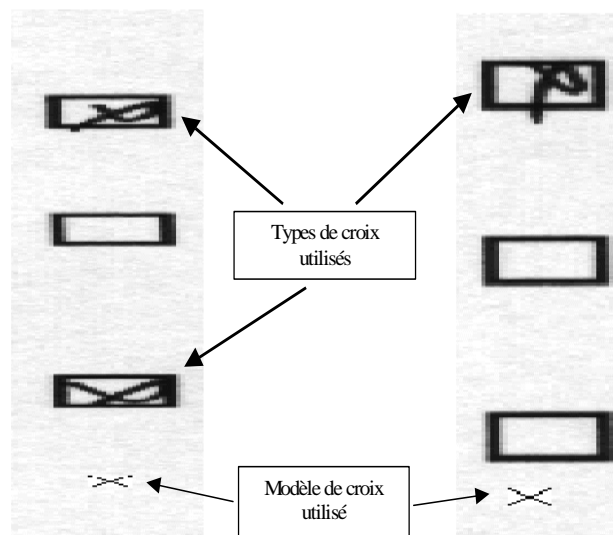


Figure 1.6 : Etape de détection des cases cochées

Le principe d'intercorrélation nous ayant donné de bons résultats dans la phase d'identification des fiches, il nous a semblé possible de l'utiliser de nouveau et de faire reposer la détection là encore sur le calcul de l'intercorrélation entre une zone contenant une case extraite de la fiche à analyser et une image de croix de référence que nous avons élaborée. Cette croix est en fait un squelette de croix de taille assez réduite, environ la moitié de la taille de la case. Ainsi nous favorisons la présence de corrélation.

C'est grâce au seuillage de l'image résultant de la corrélation que les cases cochées sont détectées. Les coordonnées des points extrêmes où il y a des détections sont extraites et permettent de créer le fichier comportant toutes les analyses à effectuer.

Le problème est de définir un seuil permettant de prendre la bonne décision. Pour choisir un seuil correct, une phase d'apprentissage est envisagée, que nous exposons ici. Une image test (cf. Figure 1.7) composée de cases cochées de manières variées et de cases vides est constituée. L'image de corrélation entre cette image et une image de croix type est montrée sur la Figure 1.8. Nous avons présenté cette image segmentée en deux parties, d'une part, en (a) celle comportant des cases contenant des croix et d'autre part en (b) celle constituée des cases vides. On observe nettement la différence des réponses.

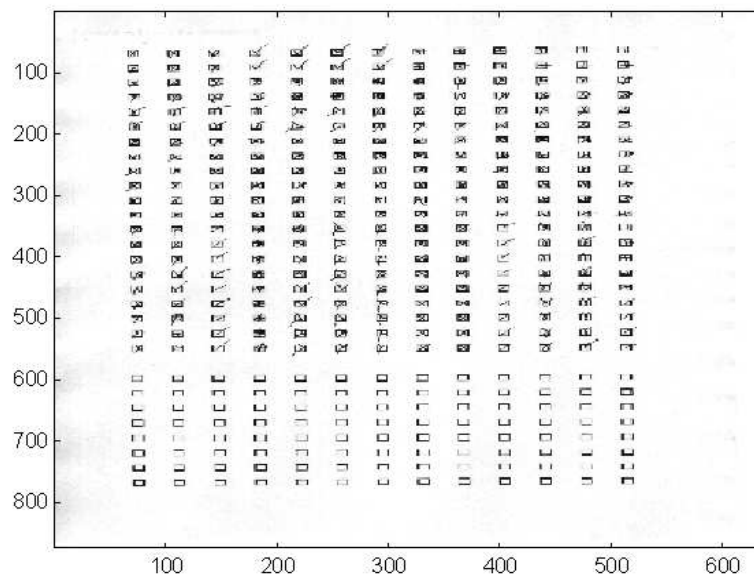


Figure 1.7 : Ensemble de croix constituant un ensemble d'apprentissage

Le résultat de l'intercorrélation entre la croix de référence présentée sur l'image Figure 1.6 et l'image précédente est illustrée Figure 1.8.

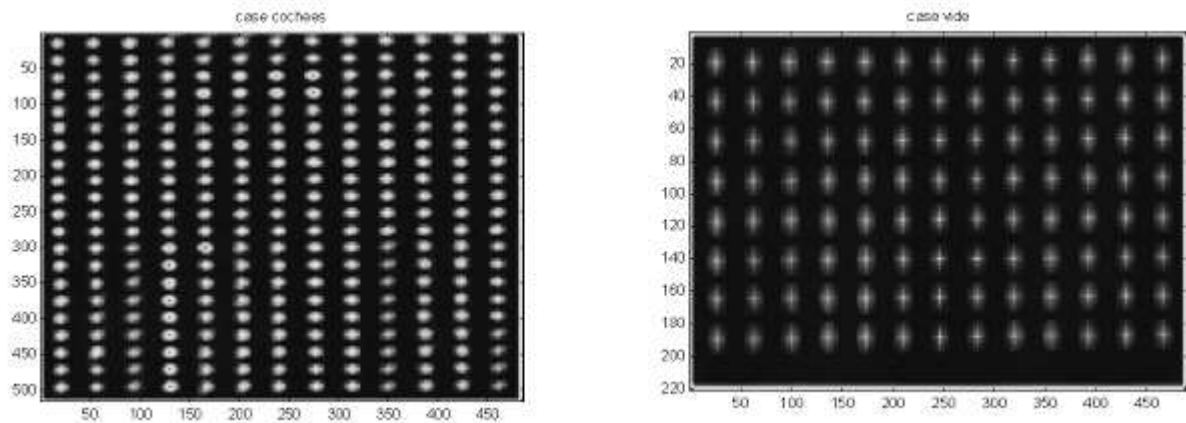


Figure 1.8 : Résultat de la corrélation entre l'image de test et la croix de référence

Le maximum local extrait est associé à chaque case, et ce, quel que soit son état considéré, i.e. coché ou non. Deux tableaux sont créés, l'un comporte les maximums correspondant à des cases cochées et l'autre ceux correspondant à des cases vides. Si l'on considère le résultat comme les occurrences d'une variable aléatoire comprise entre 0 et 1, alors la Figure 1.9 représente l'estimation de la densité de probabilité de la variable aléatoire considérée.

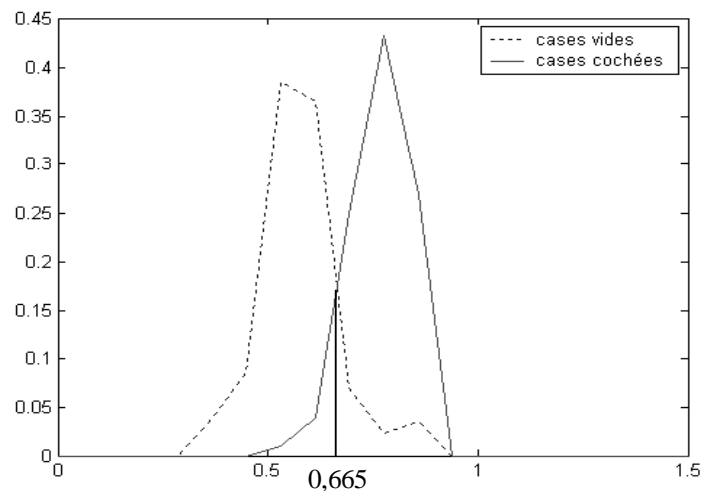


Figure 1.9 : Densité de probabilité des cases cochées (en rouge) et des cases vides (en bleu)

On observe que les deux états n'apparaissent pas de manière non ambiguë. Les deux courbes se coupent et leur intersection se situe à 0,665. Cette valeur correspond à la valeur optimale du seuil que l'on choisirait dans le cadre d'une décision bayésienne si l'hypothèse d'équiprobabilité des deux classes est acceptable. Dans le cas présent, l'aire correspondant à la portion de plan qui se trouve simultanément sous les deux courbes est trop importante pour nous satisfaire pleinement. Elle correspond aux cas d'erreur ou de confusion entre les deux

états. C'est pour cela que nous avons élaboré une autre méthode qui permet de rendre la détection plus robuste.

Nous avons décidé d'utiliser une méthode plus classique de détection des cases cochées. Elle consiste à compter le nombre des pixels non blancs dans les cases cochées par rapport au nombre de pixels non blancs dans des cases vides. Pour ce faire, nous utilisons l'image d'apprentissage précédente de la Figure 1.7. Elle est binarisée pour obtenir le nombre maximum de pixels noirs d'une case vide et les valeurs minimum et maximum du nombre de pixels noirs dans une case cochée. Nous avons ainsi trois seuils qui correspondent respectivement aux limites entre les différents états que peut prendre une case : une case vide, une case cochée et une case barrée. Sur le Tableau 1.2, nous avons répertorié le nombre de pixels comptés pour des cases cochées et des cases vides à partir de la Figure 1.7. Nous pouvons constater qu'il y a un léger recoupement du nombre de pixels entre les cases vides et cochées. En effet, le nombre maximum de pixels d'une case vide est de 65 pixels et le nombre minimum de cases cochées est de 52 pixels.

Nous avons appliqué les deux méthodes sur l'image de la Figure 1.7 et nous résumons les résultats sur le Tableau 1.1. Nous avons calculé le pourcentage de cases cochées détectées et le pourcentage de cases détectées cochées sur la deuxième partie de l'image, c'est-à-dire les cases vides. Sur la première colonne, on constate que la deuxième méthode, celle avec les seuils, est bien meilleure que la première, celle fonctionnant avec la corrélation. De plus, le nombre de fausses alarmes, c'est-à-dire détecter une case cochée alors qu'elle ne l'est pas, est plus grand pour la première méthode. Ainsi la deuxième méthode est meilleure au niveau détection et aussi plus robuste plus qu'elle a moins de mauvaises détections.

	% de cases cochées détectées	% de cases détectées cochées alors qu'elles sont vides
Méthode n°1	76	7
Méthode n°2	97	2

Tableau 1.1 : Comparaison des deux critères de détection des cases cochées

Les résultats obtenus permettent d'obtenir une augmentation du taux de reconnaissance de 20% (cf. Tableau 1.1) par rapport à la première méthode à condition évidemment d'utiliser toujours la même résolution pour scanner la fiche (i.e. 200 points par pouce). Si les fiches ne

sont pas scannées avec la même résolution que celles utilisées pour constituer les références, le nombre de pixels par case change et les seuils ne conviennent plus. Cette remarque s'applique aussi aux autres méthodes possibles de détection des cases cochées.

Cases vides											
63	56	36	42	56	65	46	38	46	45	56	57
52	36	48	57	47	48	51	52	55	46	46	46
50	56	42	55	50	47	55	54	46	52	56	46
57	58	65	57	56	46	57	57	47	53	46	54
55	46	46	54	57	46	42	46	60	55	54	41
47	46	56	56	36	55	53	50	49	40	46	52
46	46	56	57	62	58	53	42	57	43	45	49
48	57	49	52	54	57	46	41	46	45	53	55
Minimum=36 pixels, maximum=65 pixels											
Cases cochées											
78	85	97	94	93	84	87	78	91	80	105	
84	66	93	86	79	92	87	73	81	85	105	
72	75	92	77	89	98	98	114	85	90	105	
79	100	102	84	105	88	102	115	78	83	109	
98	92	102	101	99	76	83	88	83	85	94	
110	81	81	87	99	84	85	86	86	85	104	
81	91	81	84	87	90	96	96	91	96	133	
65	95	91	76	86	82	96	107	99	109	137	
80	76	93	85	79	70	88	85	81	85	104	
90	72	90	90	83	85	69	88	85	79	86	
80	78	83	79	77	93	85	86	87	85	89	
77	87	67	68	81	91	77	64	88	90	104	
83	90	71	102	111	87	93	90	100	72	119	
79	83	79	98	84	75	90	81	78	66	102	
89	82	76	110	85	80	93	90	84	69	99	
102	74	69	108	91	92	87	82	92	52	98	
89	82	68	104	87	89	83	78	88	59	106	
72	90	58	109	92	91	94	74	95	64	106	
Minimum=52 pixels, maximum=137 pixels											

Tableau 1.2 : Nombre de pixels pour presque toutes les cases de la Figure 1.7

Les informations liées à la structure du document et aux traits ayant été extraites, nous allons maintenant nous intéresser à l'extraction des zones manuscrites qui devront être traitées pour l'identification des scripteurs et la reconnaissance du texte écrit.

1.3.4. Extraction de l'écriture manuscrite

Après numérisation de la fiche en niveaux de gris, la zone supérieure dont on connaît la localisation et qui contient les informations manuscrites est recueillie. La même opération a

été appliquée à l'image de référence au cours de la phase d'apprentissage. Un exemple de ces zones correspondantes est montré sur la Figure 1.10.

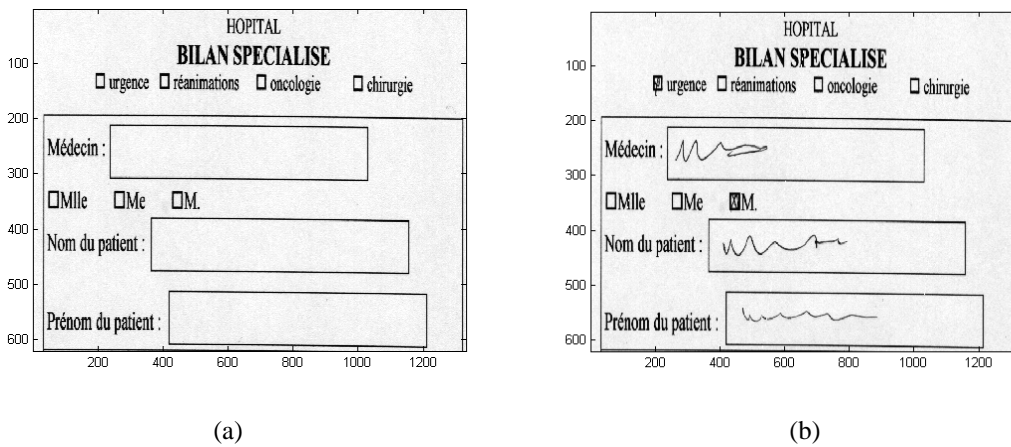


Figure 1.10 : Partie manuscrite des fiches (a) vide et (b) remplie

Pour extraire les zones d'écritures dans le cas de deux images parfaitement identiques quant à la taille et la localisation des champs, il suffit de soustraire ces deux images l'une de l'autre comme le montre la Figure 1.11(a). Mais il existe encore un décalage entre l'image de référence et l'image scannée recalée qui est dû aux arrondis de calcul lorsque l'on applique la transformation de mouvement sur chaque pixel. C'est pour cela qu'un seuillage est appliqué sur cette image. Cette technique consiste à éliminer les pixels de valeurs négatives.

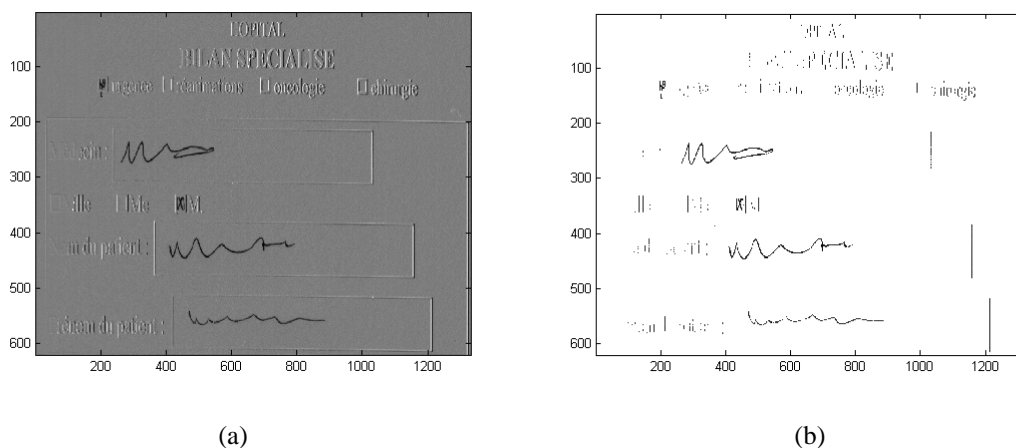
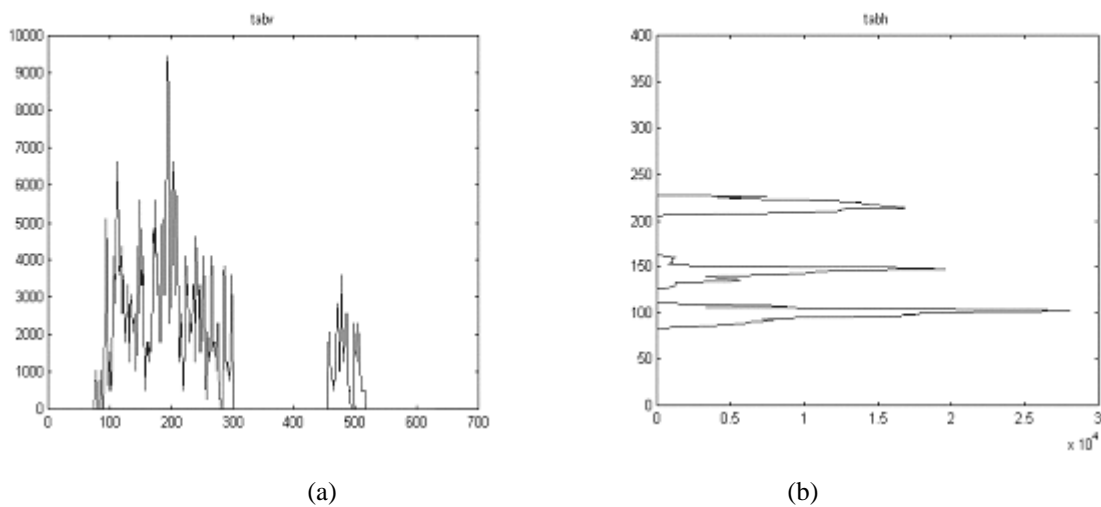


Figure 1.11 : Résultat de la récupération de l'écriture manuscrite (a) et après filtrage (b)

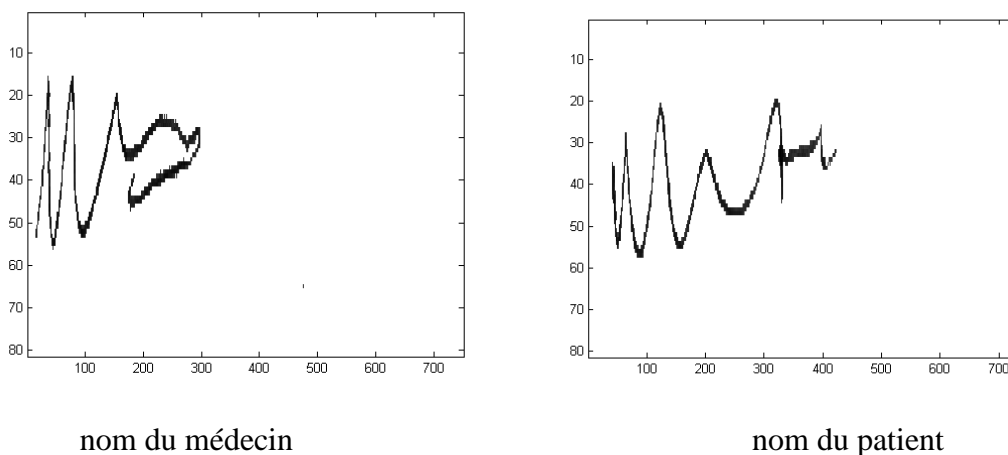
Une érosion par un élément structurant centré carré de 3 pixels de coté suivie d'une dilatation conditionnelle à l'image initiale, Figure 1.11(b), nous permet d'éliminer une bonne partie du bruit.

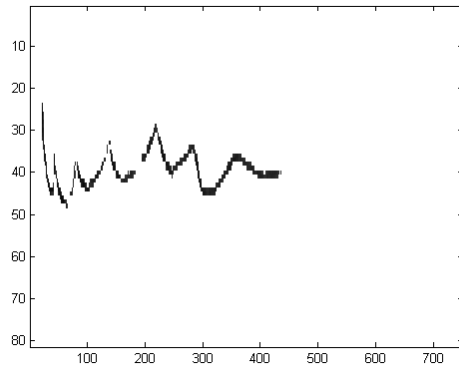
Connaissant *a priori* la position des différents champs, il a semblé intéressant d'utiliser cette information.

Les différents champs du formulaire sont à extraire. Pour différencier alors l'identité (nom et prénom) du patient et celle (nom) du médecin, on détermine leurs positions par projection horizontale. La projection verticale est utilisée pour les longueurs des identités. Le principe de la « projection » est d'effectuer la somme des valeurs d'intensités des pixels selon les lignes pour une projection horizontale et selon les colonnes pour une projection verticale. Ensuite il ne reste plus qu'à récupérer les indices de début et de fin de chaque pic pour découper l'image en bandes horizontales puis en bandes verticales.



L'avantage de cette méthode par rapport à l'extraction de zones prédéfinies de taille fixe est qu'elle permet de ne pas limiter la taille des noms écrits par le médecin à l'intérieur du cadre.





prénom du patient

Figure 1.13 : Récupération des écritures dans les différents champs

1.4. Conclusion

L'objectif de cette première étape était de concevoir des fiches et d'être capable de les différencier à partir de leur image scannée puis de détecter les cases cochées et d'extraire les zones manuscrites contenues dans l'en-tête.

La différenciation utilisant l'intercorrélation répond entièrement à notre objectif puisqu'elle distingue les fiches et qu'elle détecte aussi par un affichage si une feuille d'une autre nature a été scannée. Nous avons même la possibilité d'ajouter 6 autres types de fiches sans remettre en cause la méthode générale.

Pour détecter les cases, le même procédé d'intercorrélation mis en œuvre n'a pas donné les résultats escomptés. Une méthode utilisant le nombre de pixels moyen pour une case vide et une case cochée fournit de meilleurs résultats.

Après avoir recalé l'image scannée de manière à la soustraire à l'image de référence, nous avons réussi à éliminer le texte non désiré et à extraire l'écriture manuscrite.

Il s'agit maintenant, en s'appuyant sur le contenu des zones manuscrites extraites, et avant de s'attaquer à la phase de reconnaissance de l'écriture, d'étudier comment chaque scripteur peut être identifié parmi les praticiens autorisés. Cette phase est primordiale dans un contexte où la sécurité est une des préoccupations majeures.

Nous ferons précéder l'exposé de la solution que nous proposons par un chapitre présentant un bref état de l'art sur le sujet.

Chapitre 2

Identification de scripteur - Etat de l'art

2. Identification de scripteur - Etat de l'art

2.1. Introduction

L'écriture manuscrite est d'abord pour l'Homme un moyen de communication avec d'autres Hommes. Elle a été patiemment élaborée et évolue au cours des temps. En fonction des circonstances de l'Histoire, temporelles ou géographiques, plusieurs types d'écritures ont été mis en place mais ils ont tous le même objectif. Ils permettent la transmission d'un message entre individus. C'est un des éléments qui distinguent l'Homme du monde animal. Bien qu'apparaissant comme un héritage commun à toute une communauté, l'écriture manuscrite apparaît en même temps comme une représentation de la personne du scripteur au même titre que la parole et cela se produit en fait à son insu. L'apparence de l'écriture manuscrite dépend du contexte éducationnel, culturel mais elle a aussi un lien fort avec les caractéristiques innées de chaque personne. Cela explique la grande variabilité observée entre les écritures de scripteurs différents, même s'ils sont issus de communautés proches. Il en résulte que l'écriture est propre au scripteur [SRI01] et qu'elle peut être analysée tout comme les empreintes digitales pour identifier ou authentifier les individus. C'est ainsi que l'étude de l'écriture est à l'origine de nombreux travaux développés dans le domaine de la graphologie et dans le domaine juridique. On pourrait affirmer que l'identification de scripteur fait partie du domaine de la reconnaissance de formes et de l'analyse de l'écrit. Le problème de l'identification de scripteur inclut celui de la vérification de son identité qui n'est pas celui qui nous intéresse ici. Rappelons que la vérification de scripteur consiste à déterminer si l'auteur identifié est celui qui a écrit. L'identification d'un scripteur consiste à faire correspondre une écriture donnée à une écriture de référence dont on connaît l'auteur. Cette association est

réalisée en fonction d'un indice de similarité. Si nous considérons une base constituée de m échantillons d'écritures provenant de n scripteurs différents, l'identification se définit par l'appartenance ou non d'un échantillon d'écriture x inconnu, à l'une des classes définies à partir des m échantillons d'écriture issus des n scripteurs de cette base et si une correspondance a été trouvée, il est possible d'indiquer le nom de ce scripteur.

Dans les travaux de recherche, l'accent est surtout mis sur l'identification de signatures celle-ci a été développée pendant ces vingt dernières années. Toutefois les méthodes étudiées pourraient s'appliquer à l'aide de modifications mineures au domaine de l'identification de scripteur à partir de l'écriture manuscrite.

Les auteurs R. Plamondon et G. Lorette dans un état de l'art [PLA89] découpent le domaine d'identification de scripteur sous deux approches : l'approche contextuelle et l'approche non contextuelle. Mais depuis dix ans, on peut noter que la recherche s'est orientée vers une méthode d'identification par l'analyse du style du scripteur. Deux niveaux principaux en ressortent : l'un considère une modélisation d'allographes et l'autre considère une caractérisation globale des styles.

Ainsi dans ce chapitre, nous allons donner un aperçu des différentes méthodes d'identification de scripteur. Nous avons rassemblé ces méthodes en trois sections principales. La première fait un bilan des techniques classiques utilisées et qui sont pour la plupart pionnières dans le domaine, la deuxième section décrit les méthodes apparues récemment sur l'identification par styles et la troisième expose une méthode qui repose sur l'extraction des invariants associés à un scripteur et qui peuvent être utilisés aussi bien comme aide à la reconnaissance du texte que pour l'identification du scripteur.

2.2. Approches classiques

Ces systèmes développés sont basés sur des techniques de traitement d'images et de reconnaissance de formes. Les solutions apportées permettent de résoudre les problèmes généraux rencontrés usuellement dans les différentes étapes du traitement d'images. Elles concernent les prétraitements, la sélection de caractéristiques, la comparaison des écritures et des scripteurs ou l'évaluation de la performance globale. Pour cela, sont distinguées les phases d'apprentissage et d'identification. En ce qui concerne l'apprentissage, on suppose que le texte considéré comme exemple de l'écriture est assez long pour contenir toutes les

informations utiles sur le scripteur et on s'accorde généralement pour considérer qu'une longueur de texte est suffisante lorsque le texte comporte entre 3 et 5 lignes.

Nous distinguerons deux types de méthodes classiques, les approches contextuelles et les approches non contextuelles.

2.2.1. Approches contextuelles

Les approches contextuelles ne considèrent pas seulement l'image du texte comme une forme, mais se servent aussi du contenu sémantique du texte. Ces approches s'appuient donc sur une localisation interactive et une segmentation manuelle du texte en mots et lettres dont la connaissance fait partie intégrante des données. L'enregistrement des scripteurs peut être réalisé dans une phase initiale où un protocole approprié a été défini. On peut par exemple demander aux scripteurs d'écrire un texte fixé au préalable en écrivant chaque lettre dans une boîte. L'association entre l'image et le contenu sémantique peut alors être réalisée de manière automatique mais il est à noter que l'écriture du scripteur a été contrainte et a donc perdu en naturel.

Mihelic & al. [MIH77] utilisent en 1977 la transformée de Walsh-Hadamard appliquée aux lettres, aux mots ou au texte global. Ils ne gardent que les coefficients qui sont les plus porteurs d'information comme caractéristiques du scripteur. Le classifieur retenu repose sur le principe du maximum de vraisemblance.

Naske [NAS82] considère la forme particulière de chaque écriture comme une distorsion spécifique au scripteur à partir d'un prototype, *a priori* celui qui est appris durant la scolarité de tout écolier. Le prototype de chaque lettre est choisi commun à tous ceux qui utilisent le même alphabet. Ainsi des caractéristiques sont calculées en prenant en compte les différences existant entre le caractère écrit et le prototype. Deux méthodes sont appliquées, l'une consiste à approximer le caractère écrit à partir d'un prototype par une matrice de déplacement et l'autre procède à partir de l'image du caractère qui est déformée de manière à retrouver au mieux un des prototypes. Une description non-paramétrique d'une fonction de transfert est recherchée. Pour chacune de ces deux méthodes, 50 caractéristiques sont extraites. L'ensemble des caractéristiques est ensuite réduit grâce au critère de Fisher [KLE80] après avoir décorrélé les caractéristiques par une transformée de blanchiment. Ainsi sur les 50 caractéristiques extraites, les 20 plus significatives sont retenues. La méthode d'identification

repose sur le critère de Bayes avec un classifieur de calcul des distances minimums entre d'une part le prototype transformé et le caractère à reconnaître pour la première méthode et d'autre part entre le caractère transformé et les prototypes. Les tests sont effectués sur le mot allemand "DREIHUNDERT" ("trois cent") écrit 10 fois en caractères bâtons par 100 scripteurs. L'auteur obtient plus de 98% d'identification sur une base de 9 caractères. Ces méthodes sont assez contraignantes puisqu'elles ne peuvent pas être utilisées si on considère un seul caractère en majuscule, c'est pour cela que les scripteurs doivent écrire le mot sous forme de plusieurs combinaisons (DRE, DRE.HUND ou DRE.HUNDER).

2.2.2. Approches non contextuelles

Elles sont beaucoup plus nombreuses que les approches contextuelles et permettent de laisser aux scripteurs une plus grande liberté dans leur comportement. Nous présentons ici des systèmes d'identification de scripteur avec une approche non contextuelle selon l'ordre chronologique.

B. Arazi montre dans [ARA77], la similarité existant entre les histogrammes vertical et horizontal de deux échantillons d'écriture d'une même personne. Après avoir scanné en noir et blanc un texte de 300 lignes de 13 scripteurs différents, de façon horizontale (i.e. parallèle aux lignes écrites) et verticale, l'auteur enregistre les histogrammes correspondants. De ces deux histogrammes enregistrés pour chaque scripteur, il en ressort plusieurs mesures. Sur l'histogramme horizontal, la localisation, la largeur et la valeur du pic d'une part, le calcul de similarité entre deux textes identiques écrits par deux scripteurs différents d'autre part. La similarité est la valeur absolue de la différence calculée entre le nombre de *runs* dans les deux histogrammes et la somme de ces valeurs pour toutes les longueurs de *runs* possibles. L'histogramme vertical possède deux pics dont le premier informe sur la largeur des lettres et le deuxième correspond à l'espacement entre deux lignes. Les mesures précédentes appliquées à l'histogramme vertical ne donnant pas assez de résultats concluants, une seule mesure est retenue. Cette mesure est celle de la similarité, précédemment décrite, elle est obtenue comme la somme de toutes les valeurs élevées au carré. Cela permet d'accentuer les écarts entre scripteurs. Une mesure supplémentaire est effectuée en moyennant les valeurs des 2 histogrammes pour pouvoir établir le tableau que nous reproduisons ci-dessous.

Ces valeurs sont considérées comme indiquant la plus grande similarité entre deux scripteurs lorsqu'elles sont les plus faibles. Ainsi, l'auteur obtient 100% d'identification. Ce tableau prend en compte 5 des 13 scripteurs.

SIMILARITY OF HANDWRITING OF DIFFERENT PERSONS (AVERAGE OF HORIZONTAL AND VERTICAL SCANNING)					
	a	c	d	e	f
a	100	358	443	431	465
c	685	100	529	454	500
d	639	367	100	183	230
e	810	383	224	100	181
f	1039	465	297	193	100

Tableau 2.1 : Résultats de l'identification de scripteurs par la moyenne des mesures des histogrammes vertical et horizontal extraits de [ARA77]

Klement décrit dans [KLE81] un système complet d'identification de scripteur et dans [KLE83] comment ce système est intégré au réseau informatique. C'est un regroupement de plusieurs modules réalisés avec d'autres chercheurs dans le cadre d'un projet de recherche appelé « FISH » (Système de l'Information Experte de l'Écriture) financé par le Ministère de la Recherche Allemand. Ce système est composé d'une partie utilisant des techniques d'analyse d'image classiques telles que le critère des moindres carrés pondérés [NAS80] avec extraction de caractéristiques non-contextuelles, et une autre partie utilisant des caractéristiques relatives à un seul caractère [NAS82]. La première étude a été faite par Kuckuck *et al.*

Kuckuck et al. décrivent dans [KUC79] un processus d'identification de scripteur basé sur la méthode d'évaluation « hold-one-out ». Cette méthode consiste à retirer un échantillon de la base d'apprentissage, à entraîner le système, et à tester le système avec l'échantillon retiré. La méthode est utilisée lorsque l'on ne dispose pas d'un ensemble d'apprentissage de taille suffisante. Une étape d'extraction de caractéristiques est d'abord appliquée sur les écritures binaires. Elle comporte trois traitements distincts appliqués en parallèle. Le premier est une description analytique de l'écriture. Après séparation des lignes et squelettisation de l'écriture, une transformée paramétrique est appliquée sur les segments de courbes, elle est de la forme :

$$x(t) = \sum_{k=0}^4 a_k t^k, \quad y(t) = \sum_{k=0}^4 b_k t^k$$

où a_k et b_k sont des coefficients déterminés par la méthode des moindres carrés sur les coordonnées (x, y) des pixels.

La seconde méthode pour extraire des caractéristiques repose sur le calcul de la distribution fréquentielle sur 6 directions de toute l'image.

Et la troisième méthode considère à nouveau la distribution fréquentielle mais elle est ici calculée sur 8 directions. Après application de ces 3 méthodes, 128 caractéristiques sont extraites. Les auteurs appliquent alors une étape de compression en approximant les distributions par les moindres carrés pour obtenir au final 24 caractéristiques par scripteur. Deux classifieurs sont mis en place, l'un a recours à la méthode des plus proches voisins, quant à l'autre, c'est le critère de Bayes en faisant l'hypothèse d'une distribution gaussienne des caractéristiques. Les auteurs obtiennent 96% d'identification sur une base de 20 scripteurs.

Une nouvelle méthode d'extraction de caractéristiques a été apportée à ce processus par W. Kuckuck [KUC80]. Il discute de quelques méthodes basées sur la transformée de Fourier ou sur la fonction d'autocorrélation, qui sont apparentées à l'analyse spectrale. Des caractéristiques sont extraites par calcul de coefficients dérivés du spectre de puissance de la transformée de Fourier. Il applique par ailleurs la fonction d'autocorrélation sur ce spectre pour extraire d'autres caractéristiques. L'auteur obtient aux alentours de 90% d'identification avec ces deux nouvelles méthodes et plus de 95% avec la fonction d'autocorrélation qui demande moins de temps de calcul.

Klement *et al.* [KLE80] intègrent ces différentes méthodes dans le système FISH et élargissent leur application comme le calcul de la distribution fréquentielle des chaînes de pixels noirs pour obtenir une indication sur la forme des boucles. De plus, ils utilisent le critère de Fisher pour classifier ces caractéristiques selon leur discrimination. Ce critère se définit par :

$$J(c_v) = \sum_{i=1}^{K-1} \sum_{j=i+1}^K \frac{(m_{iv} - m_{jv})^2}{\sigma_{iv}^2 + \sigma_{jv}^2}$$

où c_v sont les caractéristiques à sélectionner avec m_{iv} leur moyenne et σ_{iv}^2 leur variance pour chaque scripteur i .

Sur une base constituée de 20 scripteurs avec 40 échantillons par scripteur, ils obtiennent ainsi un taux de 99,6% d'identification avec 15 caractéristiques. Mais ils n'ont pas utilisé tout leur ensemble d'apprentissage et des tests avec plus de caractéristiques restent à faire.

Said *et al.* [SAI98] [SAI00] utilisent une technique d'analyse de texture pour identifier le scripteur sur des écritures redressées non-uniformément. Cette méthode se décompose en trois

étapes principales. La première étape est la normalisation, la seconde est l'extraction de caractéristiques et la dernière est l'identification. La normalisation se divise en deux sous-étapes. Une première consiste à détecter la pente de l'écriture par une projection horizontale et à la corriger par un ajustement des lignes des composantes connexes. La seconde consiste à normaliser le texte en éliminant les blancs entre les mots et les lignes pour pouvoir appliquer l'analyse de texture. L'étape d'extraction de caractéristiques utilise deux méthodes distinctes comparables, un filtre de Gabor multi-canal, et un calcul de matrice de cooccurrence. Le filtre de Gabor multi-canal permet d'extraire des caractéristiques de texture. Il se définit par :

$$\begin{cases} h_e(x, y; f, \theta) = g(x, y) \cos(2\pi f(x \cos \theta + y \sin \theta)) \\ h_o(x, y; f, \theta) = g(x, y) \sin(2\pi f(x \cos \theta + y \sin \theta)) \end{cases}$$

Avec g une fonction gaussienne 2D, f et θ la fréquence radiale et l'orientation qui définissent la localisation du canal dans le plan fréquentiel. L'auteur prédéfinit 4 valeurs pour les fréquences (4, 8, 16 et 32 cycles/degré) et pour chacune d'elles, 4 orientations ($\theta = 0^\circ, 45^\circ, 90^\circ$ et 135°). Ainsi 16 caractéristiques sont obtenues. Pour chaque caractéristique, la moyenne et l'écart type sont calculés, ce qui fait un total de 32 caractéristiques. La deuxième méthode utilise le calcul des matrices de cooccurrence. La matrice de cooccurrence MC est une matrice carrée de dimension N (nombre de niveaux de gris) où chaque élément MC(i,j) représente le nombre de paires de pixels séparés par une distance d pour un angle α , qui ont respectivement un niveau de gris i et j . Les auteurs considèrent 5 distances ($d = 1, 2, 3, 4, 5$) et 4 directions ($\alpha = 0^\circ, 45^\circ, 90^\circ$ et 135°) sur des images binaires ($N=2$). Il y a alors 20 matrices de dimension 2 et pour chacune, ils calculent 3 valeurs selon les symétries diagonales, ce qui fait un total de 60 caractéristiques extraites avec cette méthode. On procède ensuite à une étape d'identification. Pour cela, les auteurs comparent 2 classifieurs particuliers. Le premier utilise la distance euclidienne pondérée définie par :

$$d(k) = \sum_{n=1}^N \frac{(f_n - f_n^k)^2}{(v_n^k)^2}$$

où f_n indique la valeur de la $n^{\text{ième}}$ caractéristique du document à identifier, f_n^k et v_n^k sont respectivement les moyennes et écarts types dans l'ensemble des échantillons de la $n^{\text{ième}}$ caractéristique du scripteur k . Le deuxième classifieur est celui des plus proches voisins. Les tests ont été effectués sur des textes écrits par dix scripteurs différents, scannés à 150 ppp. Les résultats expérimentaux montrent que 96% d'identifications sont obtenus pour une combinaison du filtre de Gabor multi-canal avec la distance euclidienne pondérée.

Zois et Anastossopoulos [ZOI00] identifient des scripteurs en utilisant une transformation morphologique pour extraire un vecteur de caractéristiques sur un mot. Une projection horizontale est effectuée sur le mot squelettisé. Cette projection est alors segmentée à l'aide d'une opération morphologique, la granulométrie. Deux types de fenêtres sont appliqués sur les segments de la projection pour contrôler le flot d'information. Les espaces entre les lettres sont aussi pris en compte dans la formation du vecteur de caractéristiques. L'étape d'identification dépend de la dimension du vecteur qui dépend elle-même de la longueur du mot. La base de données a été constituée de 50 scripteurs qui ont écrit deux mots de même longueur en Anglais et en Grec. Les classifieurs utilisés sont celui de Bayes et un réseau de neurones. Les auteurs obtiennent 95% d'identification.

Marti et al. [MAR01] extraient des caractéristiques par ligne souvent visibles par le lecteur et les classent selon deux classifieurs tels que les réseaux de neurones et l'algorithme des plus proches voisins pour identifier les scripteurs. Leur processus de découpe en trois étapes : prétraitements, extraction de caractéristiques et classification. L'étape de prétraitement consiste à segmenter une page de texte par ligne à l'aide d'une projection puis à binariser ces lignes. Ensuite, pour chaque ligne, un ensemble de caractéristiques est extrait. Cet ensemble se compose de 12 caractéristiques : 6 correspondent aux zones du texte telles que la partie supérieure et inférieure du corps du texte, 2 correspondent à la taille de l'écriture (la distance entre deux mots), les 2 suivantes prennent en compte la pente de l'écriture et les 2 dernières calculent les pentes extraites d'un graphe tracé par le calcul de la dimension fractale. Les tests sont effectués sur 20 scripteurs avec pour chacun, 482 lignes de textes pour l'apprentissage et 182 lignes pour les tests. Ils obtiennent 87,8% d'identification lorsqu'ils utilisent 7 des 12 caractéristiques sur l'algorithme des plus proches voisins et 90,7% en utilisant toutes les caractéristiques sur un réseau de neurones de niveau 20.

Ces méthodes classiques recherchent des caractéristiques qui permettraient de discriminer de manière objective les écritures, ou plutôt les scripteurs. Partant de la constatation que nous savons reconnaître l'écriture des gens qui nous sont proches sans l'analyser mais en la regardant globalement sans chercher à lire le contenu du texte considéré, différentes approches ont cherché à identifier les scripteurs par leur style d'écriture. Le style peut aussi être considéré comme une caractéristique commune à une famille de scripteurs dont les écritures se ressemblent.

2.3. Méthodes d'identification par styles

L'objectif de ces méthodes est d'extraire des caractéristiques propres au scripteur indépendamment du contenu sémantique du texte analysé. Ces caractéristiques permettraient aussi d'adapter la reconnaissance d'écriture au scripteur par le fait que les écritures de même style présentent une variabilité moindre que l'ensemble de toutes les écritures possibles. Il existe deux axes : la modélisation d'allographes qui permet d'absorber la variabilité de l'écriture et une méthode globale par styles qui permet de regrouper les scripteurs par familles.

2.3.1. Modélisation d'allographes

Les modélisations d'allographes consistent à représenter les lettres sous forme d'un ensemble de pseudo-lettre. Elles permettent de s'affranchir de la trop grande variabilité de l'écriture. Plusieurs méthodes ont été utilisées.

M. Gilloux décrit dans [GIL94] une méthode améliorant le taux de reconnaissance d'écriture en incluant le style d'écriture. Pour ce faire, il propose une méthode d'adaptation au scripteur basée sur plusieurs modèles d'écritures estimés par les Modèles de Markov Cachés (MMC). Chaque modèle est entraîné pour prendre en compte un style en particulier qui est connu au préalable. Les modèles de Markov supposent que l'image peut être représentée comme une séquence d'observations. De plus, ils requièrent que ces observations soient indépendantes sous certaines conditions quand l'état caché est connu. L'auteur utilise 3 façons de classer les styles pour s'assurer que les observations sont indépendantes mutuellement quand les états cachés des MMC sont connus. La première étape est la normalisation de l'écriture ce qui permet de garantir une meilleure généralisation du modèle markovien. Les autres aspects du style de l'écriture sont pris en compte dans le MMC par l'utilisation de plusieurs sous modèles.

La détection du style se fait pendant l'étape d'extraction de caractéristiques. Cette étape consiste à segmenter les composantes connexes des mots en contours supérieur et inférieur puis à rechercher les boucles et les extensions supérieures et inférieures de chaque segment de lettres. Un symbole différent est attribué à chaque configuration possible des boucles, des

extensions et de leurs tailles respectives. En tenant compte des espaces, ces caractéristiques peuvent se résumer en un ensemble de 27 symboles. C'est l'agencement de ces symboles associé à une corrélation avec le mot qui permet de classer les différents styles de scripteurs. Une description de l'extraction de caractéristiques se trouve sur la Figure 2.1 .

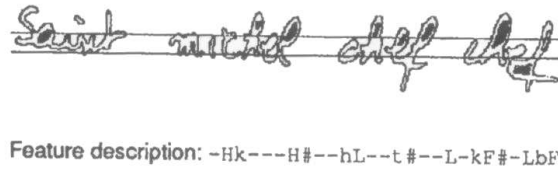


Figure 2.1 : Extraction de caractéristiques

2.3.2. Méthode globale des styles

Cette méthode consiste à considérer plusieurs familles de styles et à essayer d'inclure chaque scripteur dans l'une d'elles.

J.P. Crettez [CRE95] propose une caractérisation des écritures dans le but d'améliorer le module de reconnaissance des chèques postaux par processus de Markov cachés développés au Service de Recherche Technique de la Poste. Le principe consiste à s'adapter à la variabilité de l'écriture et non à l'occulter. Cette caractérisation permet de définir à quelle famille de styles appartient une écriture. L'écriture est une succession d'unités graphémiques appelées allographes. Ces allographes sont constitués par un enchaînement de traits de plume appelés les allotraits. Ils sont les éléments de base d'une écriture. L'auteur définit deux degrés de caractérisation. Le premier est une analyse non-supervisée qui consiste à détecter les allotraits et le deuxième est une analyse supervisée qui consiste à détecter des allographes. Onze observations non-sémantiques sont retenues pour classer les écritures. Trois d'entre elles sont relatives à la structure du mot. Elles représentent l'épaisseur du tracé, le corps du texte et la densité spatiale de l'écriture. Huit autres sont relatives aux orientations du tracé. Elles sont détectées sur les différentes parties rectilignes et sont obtenues à partir d'un diagramme décomposé en 4 lobes d'amplitudes et d'orientations différentes qui sont à leur tour projetés selon l'axe vertical. Cette projection permet de sélectionner 4 groupes d'allotraits. Ainsi le premier groupe représente l'ossature des hampes et des jambages, tandis que son orientation correspond à l'inclinaison de l'écriture. Le deuxième groupe renferme les liaisons naturelles inter- et intra-lettres et ainsi de suite jusqu'à obtenir les 8 observations.

L'analyse de ces 8 observations se fait par une analyse en composantes principales. En prenant les 5 premiers vecteurs propres, 80% de la variabilité est représentée et permet de regrouper les écritures en « clusters ». Avec une base de données de 980 montants littéraux de chèques postaux, il y a environ 5 familles de styles.

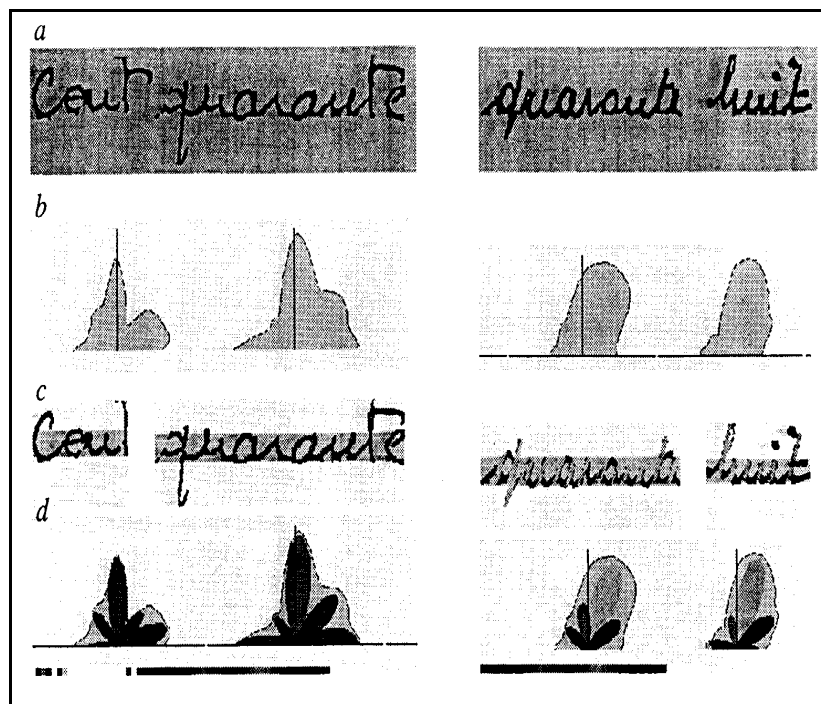


Figure 2.2 : (a) deux montants littéraux, (b) leur diagramme d'orientation, (c) parties rectilignes du tracé, et (d) les 4 lobes d'orientation

Evidemment, cette technique n'a pas pour objectif avoué de discriminer les écritures de manière suffisamment précise pour caractériser les scripteurs mais elle s'appuie néanmoins sur des propriétés statistiques extraites de l'écriture comme le font les graphologues et les experts en écriture [SEY98].

Analyse fractale

Une première étude développée par N. Vincent dans [VIN95a] montre qu'il est possible de mesurer les degrés d'irrégularité des écritures grâce à la dimension fractale. C'est la dimension de Minkowski qui a été choisie pour en estimer la valeur pour des raisons de simplicité des calculs informatiques. Si $A(X)$ représente l'aire d'un ensemble X , alors la dimension peut être définie comme suit :

$$D(X) = \lim_{r \rightarrow 0} \left[2 - \frac{\ln A(X_r)}{\ln r} \right]$$

où X_r représente le recouvrement de X par des boules de rayon r . Un graphe est alors construit pour permettre des comparaisons visuelles entre différents ensembles X . Il est défini par $\left(x = \ln(n), y = \ln \frac{A(X_n)}{A(X)} - \ln(n) \right)$ où X_n est l'ensemble X dilaté $n-1$ fois. C'est la même notation qui est employée dans le X_r précédent mais ici seules des valeurs entières sont considérées. Ces dilatations successives permettent d'approximer la réunion des boules réalisant le recouvrement, leur rayon est déterminé par la taille de l'élément structurant de la dilatation. L'évolution de l'aire des dilatés en fonction de la taille de l'élément structurant utilisé dans la dilatation permet de tracer le graphe d'évolution caractérisant chacune des écritures testées. Les valeurs de la dimension fractale sont estimées à partir de ce graphe d'évolution. L'analyse des courbes obtenues sur les écritures étudiées permet de distinguer deux zones principales. La première est celle de la vision ordinaire à faible échelle correspondant aux fréquences spatiales élevées. La seconde est celle de la vision plus éloignée correspondant aux fréquences spatiales basses. Ces deux zones permettent de définir deux paramètres : $dim1$, dimension associée à la vision de près et $dim2$, dimension associée à la vision de loin. Ces deux paramètres permettent une classification des écritures suivant la quantité de détails contenus dans le trait, suivant leur lisibilité et tiennent compte de la résolution utilisée lors de l'acquisition de l'image, Figure 2.3. Dans cette figure, chaque point représente une écriture pour laquelle on a indiqué les valeurs de $dim1$ sur l'axe horizontal, tandis que l'axe vertical indique les variations de $dim2$. Les carrés correspondent aux textes imprimés qui utilisent des polices dactylographiées ou imprimées, tandis que les croix correspondent aux écritures manuscrites écrites par différents scripteurs. Deux groupes, séparés par le trait discontinu, peuvent être distingués : en haut à droite il y a un groupe de points correspondant à des écritures régulières et bien lisibles tandis que le groupement inférieur en bas à gauche regroupe les écritures irrégulières de lecture difficile. Ainsi il est montré en particulier que par cette méthode il est possible de distinguer l'imprimé du manuscrit en se basant sur l'analyse fractale.

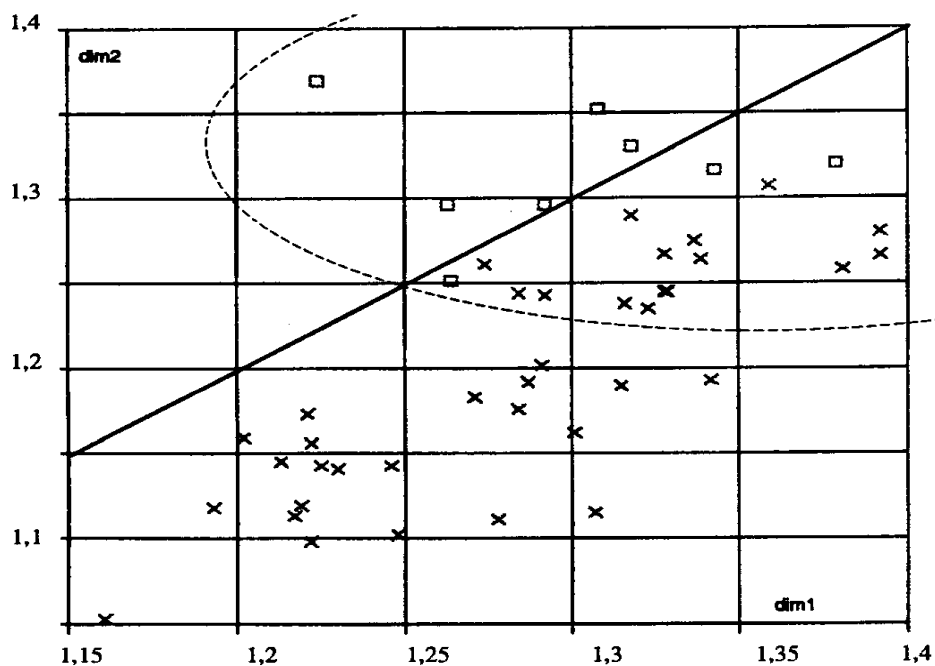


Figure 2.3 : Distribution de l'écriture selon les dimensions fractales extrait de [VIN95a]

Il est à noter que si N. Vincent utilise la dimension fractale pour classer les écritures par style, Y. Tang l'utilise pour faire de la vérification de signatures [TAN02], et Y. Tao [TAO01] pour distinguer entre les écritures utilisant différents alphabets donc a fortiori avec des styles différents.

Par la suite, des améliorations ont été apportées afin de pouvoir analyser les écritures manuscrites [VIN95b]. En effet, la dimension fractale appliquée à l'écriture est parfois insuffisante comme mesure discriminante. Une autre mesure y a été ajoutée : la dimension du profil. Elle permet de prendre en compte les écritures lisibles mais non liées qui donnent une valeur faible de la dimension fractale. Le principe de la mesure de profil consiste à extraire les parties hautes et basses de l'image dilatée de l'écriture.

La suite logique de ces études est l'extraction de caractéristiques fractales de l'écriture. C'est ce qui fut fait dans [VIN98]. En utilisant la compression fractale, N. Vincent a montré qu'il était possible de classer les styles d'écritures. Après une étape de compression/décompression, la qualité des images décompressées est évaluée par la mesure du rapport signal/bruit entre cette image reconstruite et l'image initiale. Un test a été fait sur 4 scripteurs différents. La Figure 2.4 montre le classement obtenu avec sur l'axe des abscisses les rapports signal/bruit obtenus et sur l'axe des ordonnées le taux de compression. Ainsi cette étude laisse penser

qu'avec une technique de compression fractale, il est possible de caractériser un scripteur et donc des propriétés inhérentes à celui-ci.

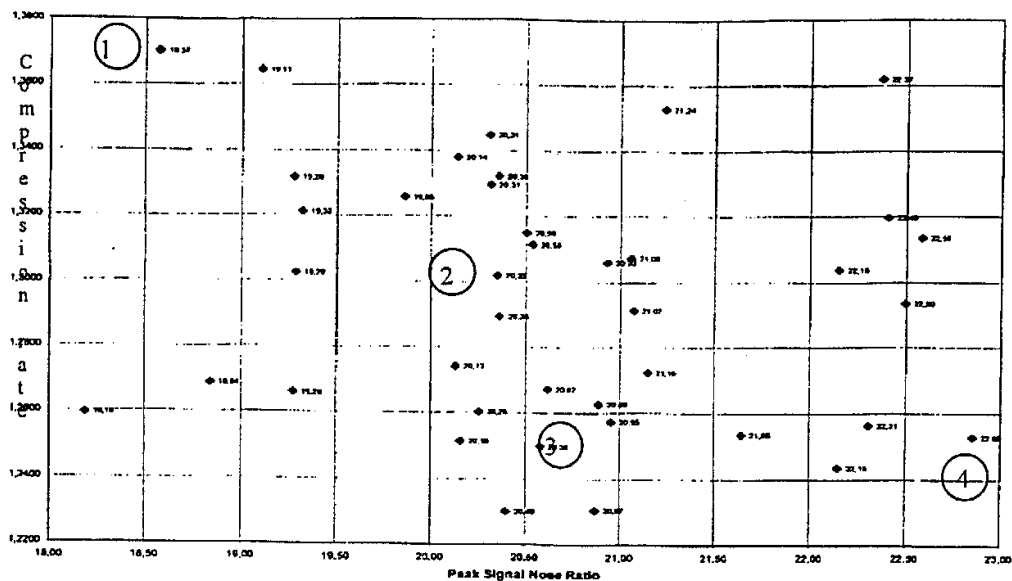


Figure 2.4 : Classement de l'écriture par compression fractale

La dernière technique que nous voulons signaler ici est une méthode de détermination de portions de traits extraites dans l'écriture et qui possèdent des propriétés d'invariance au sein de l'écriture d'un même scripteur.

2.4. Détermination des invariants

Ali Nosary décrit dans sa thèse [NOS02] un système de reconnaissance de textes exploitant le principe d'adaptation du système au type de l'écriture. Ce principe d'adaptation est basé sur l'apprentissage, au cours de la reconnaissance, des particularités graphiques représentatives de l'écriture du scripteur (invariants du scripteur). La reconnaissance de mots procède selon une approche analytique basée sur le principe de segmentation-reconnaissance. L'adaptation en ligne du système de reconnaissance repose sur l'itération de phases de reconnaissance de mots qui permettent d'étiqueter les représentations du scripteur (allographes) sur l'ensemble du texte et de phases de réévaluation des modèles de caractères. La mise en application de sa stratégie de reconnaissance par adaptation nécessite de recourir à un schéma de reconnaissance interactif capable de faire interagir les traitements aux différents niveaux contextuels. Le modèle d'interaction retenu est basé sur des systèmes multi-agent de l'intelligence artificielle. Les tests réalisés sur un échantillon d'une quinzaine de scripteurs

tous inconnus du système montrent l'intérêt du schéma d'adaptation proposé puisqu'il obtient une amélioration de la reconnaissance, à la fois des lettres et des mots, au cours des itérations du processus d'adaptation.

L'extraction d'invariants [NOS99] est une procédure de classification automatique des graphèmes segmentés issus du module de segmentation du texte. Ce module de segmentation des graphèmes génère initialement les graphèmes élémentaires qui constituent le premier niveau de segmentation. La concaténation de deux ou trois graphèmes successifs constitue le deuxième et le troisième niveau de segmentation respectivement. Le but de cette méthode est de regrouper les graphèmes morphologiquement proches au sens d'une distance à préciser. Ce regroupement peut se faire en s'appuyant soit directement sur la morphologie des graphèmes où les informations utilisées seront alors les pixels de l'image du graphème, soit sur un vecteur de caractéristiques contenant un ensemble de primitives extraites de l'image du graphème.

Cette méthode se divise en trois étapes : classification, représentation et comparaison des représentations. La classification des graphèmes en groupes similaires est faite par un regroupement séquentiel. Son algorithme se présente sous 3 étapes :

1. Initialisation
 - Choix d'un seuil
 - Choix du 1^{er} élément
2. A partir du deuxième élément
 - Mettre l'élément courant dans le groupement le plus proche par rapport au seuil de proximité
 - S'il ne se rapproche pas d'un groupement, créer un nouveau groupement
3. Test d'arrêt
 - Continuer tant qu'il reste des éléments

Cette méthode de classification qui permet de regrouper les éléments semblables par groupes est associée à une méthode de comparaison. A. Nosary a choisi la mesure de corrélation comme méthode de comparaison. Elle se définit comme suit :

Soient X et Y , deux vecteurs binaires de dimension n , le calcul de la corrélation est défini par :

$$S(X, Y) = \frac{n_{11}n_{00} + n_{10}n_{01}}{[(n_{11} + n_{10})(n_{01} + n_{00})(n_{11} + n_{01})(n_{10} + n_{00})]^{1/2}}$$

Avec n_{11} , le nombre de fois où les coordonnées des deux vecteurs valent toutes les deux 1 ;

n_{00} , nombre de fois où les coordonnées des deux vecteurs valent toutes les deux 0 ;

n_{01} , nombre de fois où une coordonnée de X vaut 0 et celle de Y vaut 1 ;

n_{10} , nombre de fois où une coordonnée de X vaut 1 et celle de Y vaut 0.



Figure 2.5 : Exemple d'invariants extraits d'un texte de [NOS02]

2.5. Conclusion

Nous venons de voir dans ce chapitre les méthodes qui nous ont semblé les plus caractéristiques depuis ces 30 dernières années dans le domaine de l'identification de scripteurs. Ce sujet fait partie du domaine de la reconnaissance de formes et de l'analyse de l'écrit. La recherche a beaucoup évolué durant cette période.

Au début on pouvait classer les études selon deux approches : l'approche contextuelle et l'approche non contextuelle. Toutes les méthodes décrites obtiennent un taux d'identification aux alentours de 98%. Mais ces taux sont souvent obtenus sur peu de scripteurs, sous contraintes d'écriture ou avec un choix du nombre des caractéristiques les mieux adaptées pour le processus.

Ensuite, Les modélisations d'allographes ont permis de s'affranchir de la trop grande variabilité de l'écriture. Les méthodes globales par style obtiennent de très bons taux d'identification de façon plus robuste et surtout sans contrainte pour le scripteur. Pour l'instant, elles consistent à créer plusieurs familles de styles d'écriture et à essayer de rattacher une écriture inconnue à une des familles. Elles sont surtout utilisées pour l'aide à la reconnaissance à cause de leur application industrielle mais pas pour faire de l'identification un but fini.

Ainsi après avoir passé en revue des méthodes sur l'identification de scripteurs, une méthode très intéressante en est ressortie ; celle de l'équipe de T. Paquet [BEN02] sur l'extraction d'invariants. Cette technique consiste à déterminer des portions extraites dans l'écriture qui possèdent des propriétés d'invariance au sein de l'écriture d'un même scripteur. L'extraction des invariants associés à un scripteur peut être utilisée aussi bien comme aide à la reconnaissance du texte que pour l'identification du scripteur.

Les études menées par N. Vincent ont permis de séparer les écritures par familles de styles grâce au calcul de la dimension fractale. Ainsi il en ressort que les écritures ont un comportement fractal et qu'il est possible d'utiliser la « *fractalité* » de l'écriture pour en extraire des caractéristiques propres à chaque scripteur. Cette étude laisse penser qu'avec une technique de compression fractale, il serait également possible de caractériser un scripteur et donc des propriétés inhérentes à celui-ci.

Associer les invariants au comportement fractal de l'écriture nous est apparu comme étant un axe de recherche prometteur. Il permettrait ainsi, non plus de relier un scripteur à un groupe mais d'extraire des caractéristiques inhérentes à chaque scripteur.

Chapitre 3

Identification par compression fractale

3. Identification par compression fractale

Introduction

Dans le chapitre précédent nous avons exploré les méthodes d'identification existantes qui nous ont semblé les plus caractéristiques. En outre il est ressorti que depuis quelques années cet axe de recherche s'est tourné vers l'extraction de caractéristiques propres au scripteur, cela pour pouvoir adapter ensuite un module de reconnaissance d'écriture spécifique selon le scripteur. La méthode que nous proposons dans ce chapitre s'inscrit dans cet axe. Son originalité principale réside d'une part dans le type des caractéristiques choisies, des formes invariantes et d'autre part dans la technique d'extraction mise en œuvre. Pour l'extraction des caractéristiques choisies, la technique utilisée repose sur le processus de la compression fractale. Elle se définit par une recherche d'éléments similaires, à une transformée près, dans une image.

Nous exposons tout d'abord en détail la théorie et la méthode de la compression fractale. Puis, nous présentons notre méthode proprement dite. De manière classique, le processus peut être décomposé en deux étapes, la première est l'apprentissage de chaque scripteur autorisé qui s'effectue au moment de l'enrôlement de ces derniers, la seconde est l'identification ou l'authentification proprement dite qui s'effectue en ligne. Nous montrons comment, dans l'étape d'apprentissage d'un scripteur, un des éléments de la méthode de compression fractale est utilisé pour extraire des caractéristiques. Dans les troisième et quatrième sections l'étape d'identification d'un scripteur proprement dite est décrite en détail. Enfin, nous exposons les résultats obtenus par notre méthode.

3.1. Compression fractale

3.1.1. *Rappels théoriques*

La géométrie des fractales a été initiée par B. Mandelbrot en 1977 [MAN77]. Par la suite M.F. Barnsley, dans son livre "Fractals everywhere" [BAR88], a réussi à simplifier et à concrétiser les notions théoriques de cette géométrie en leur trouvant des applications (entre autres la compression d'image). Il a aussi présenté les fondements mathématiques relatifs aux systèmes de fonctions itératives (IFS), basées sur l'utilisation de transformations affines contractantes. Mais constatant leur insuffisance (impossibilité de compression d'image naturelle telle que celle d'un visage), il développa un schéma modifié appelé PIFS (Partitionned Iterated Functions Systems) en mettant en évidence un algorithme qui permet, de manière automatique, de convertir une image en un PIFS, entité dont le volume de stockage est beaucoup moins important que celui de l'image initiale. Plus tard, A. Jacquin [JAC90] a pu implémenter cet algorithme en software, c'est sur cette étude que tous les programmes contemporains de compression d'images par fractales sont basés. La difficulté est de déterminer les coefficients des transformations qui constituent le PIFS. Plusieurs variantes et améliorations ont été faites. Alors que les coefficients des transformations (appelées aussi transformations massiques) utilisés par A. Jacquin sont choisis dans des ensembles prédéfinis de valeurs, Y. Fisher [FIS95] utilise un quantificateur scalaire uniforme.

Dans ce chapitre, nous avons choisi de faire reposer notre approche de la compression fractale sur la technique développée par Y. Fisher [FIS95]. Son principe consiste à considérer une image donnée I comme étant l'unique point fixe d'une transformée géométrique ω . Le plus souvent la transformation ω est plus complexe car elle est estimée et l'image est définie comme l'attracteur d'un système donné de fonctions itératives (IFS). L'existence et l'unicité du point fixe pouvant être démontrées par l'application du « théorème du point fixe », l'image point fixe est obtenue comme étant la limite d'une séquence d'images $I = \lim_{n \rightarrow \infty} \omega(I_n)$, celle-ci est définie itérativement par

$$\begin{cases} I_0 & \text{quelconque} \\ I_{n+1} = \omega(I_n) & \forall n \in \mathbb{N} \end{cases}$$

Le problème de la compression fractale consiste à trouver, à partir d'une image connue, un système de transformations qui admettrait précisément cette image comme unique point fixe. Avant de présenter la méthode de compression, la définition d'un IFS doit être donnée de façon plus précise.

3.1.2. IFS (Iterative Function System)

Un IFS est un ensemble de transformations géométriques élémentaires "linéaires" ou "affines" contractantes qui permet de générer une image qualifiée de fractale car elle apparaît comme un attracteur.

Transformation Lipschitzienne

Soit $\omega : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ une transformation définie sur l'espace métrique (\mathbb{R}^2, d) . d désigne la distance entre deux points de \mathbb{R}^2 . La transformation ω est dite Lipschitzienne, avec pour facteur de Lipschitz le réel s strictement positif si :

$$\forall x, y \in \mathbb{R}^2 \quad d(\omega(x), \omega(y)) \leq s.d(x, y)$$

Transformation contractante

Soit $\omega : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ une transformation définie sur l'espace métrique (\mathbb{R}^2, d) . La transformation ω est dite contractante, avec pour facteur de contraction le réel s , s'il existe un réel s vérifiant l'inégalité $0 \leq s < 1$ et tel que :

$$\forall x, y \in \mathbb{R}^2 \quad d(\omega(x), \omega(y)) \leq s.d(x, y)$$

Elle est dite strictement contractante si s est strictement inférieur à 1.

Point fixe

Un point fixe est défini relativement à une application ω définie dans un espace et à une valeur dans ce même espace. C'est un point invariant par la transformation : $\omega(x) = x$.

Il existe de nombreux théorèmes qui assurent, selon des hypothèses variées, qu'une application possède un point fixe et que, mieux encore, il est unique. D'un point de vue pratique nous sommes plus intéressés par la construction du point fixe quand il existe.

Le théorème fondamental du point fixe qui nous concerne énonce que toute application strictement contractante d'un espace métrique (E, d) complet et à valeur dans ce même espace

- admet un point fixe p ,
- que ce point fixe est unique, et
- que pour tout x de E la suite des itérés de x , à savoir $x_1 = \omega(x), x_2 = \omega(x_1), x_3 = \omega(x_2), \dots, x_{n+1} = \omega(x_n)$, converge vers ce point fixe p .

On peut remarquer que ce théorème possède un aspect constructif puisque pour tout point x élément de E , la suite $\{\omega^{on}(x) : n = 0, 1, 2, \dots; n \in \mathbb{N}\}$ converge vers x_f :

$$\forall x \in E \quad \lim_{n \rightarrow \infty} \omega^{on}(x) = x_f$$

Dans notre application l'espace E est 3^2 .

Distance de Hausdorff

Considérons l'espace métrique $(3^2, d)$, où d est une métrique usuelle. $H(3^2)$ désigne l'espace dont les éléments sont les sous-ensembles compacts de 3^2 , différents de l'ensemble vide. La distance du point x élément de 3^2 à un ensemble B de $H(3^2)$, notée $d(x, B)$, est définie par :

$$d(x, B) = \min\{d(x, y) : y \in B\}$$

La fonction d définie *a priori* seulement sur 3^2 peut être étendue en une fonction sur $H(3^2) \times H(3^2)$. La distance d de l'ensemble A élément de $H(3^2)$ à l'ensemble B élément de $H(3^2)$, notée $d(A, B)$, est, par exemple, définie par :

$$d(A, B) = \max\{d(x, B) : x \in A\}$$

Cette fonction ne possède plus les propriétés d'une distance mais permet d'en construire une.

La distance de Hausdorff entre deux ensembles A et B éléments de $H(3^2)$, notée $h_d(A, B)$ est définie par :

$$h_d(A, B) = \max\{d(A, B), d(B, A)\}$$

On peut montrer que cette nouvelle métrique assure à $(H(\mathbb{Z}^2), h_d)$ une structure d'espace complet.

Transformée contractante sur l'espace $H(\mathbb{Z}^2)$

Soit une transformation contractante $\omega: \mathbb{Z}^2 \rightarrow \mathbb{Z}^2$ définie sur l'espace métrique (\mathbb{Z}^2, d) avec le réel s pour facteur de contraction. La généralisation de la transformation $\omega: H(\mathbb{Z}^2) \rightarrow H(\mathbb{Z}^2)$ définie par :

$$\forall B \in H(\mathbb{Z}^2) \quad \omega(B) = \{\omega(x) : x \in B\}$$

est contractante sur $(H(\mathbb{Z}^2), h_d)$, avec le même rapport s pour rapport de contraction. h_d désigne la distance de Hausdorff définie précédemment.

Systèmes de fonctions itérées (IFS)

Un IFS défini dans l'espace métrique complet (\mathbb{Z}^2, d) est composé de N transformations contractantes $\omega_i: \mathbb{Z}^2 \rightarrow \mathbb{Z}^2 (i = 1, \dots, N)$, à chacune desquelles est associé un facteur de Lipschitz s_i . $\max\{s_i : i = 1, \dots, N\}$ constitue un facteur de contraction s des transformations ω est égal à.

Attracteur d'un IFS

Soit un IFS défini sur un espace métrique de base, c'est à dire un ensemble de N transformations $\{\mathbb{Z}^2; \omega_i; i = 1, \dots, N\}$ dont le rapport de contractance est strictement inférieur à 1. L'opérateur $W: H(\mathbb{Z}^2) \rightarrow H(\mathbb{Z}^2)$ défini par :

$$W(B) = \bigcup_{i=1}^N \omega_i(B), \quad \forall B \in H(\mathbb{Z}^2) \tag{1}$$

est contractant et a pour facteur de contraction celui de l'IFS. L'opérateur W possède un *point fixe* unique A_f donné par :

$$A_t = W(A_t) = \lim_{n \rightarrow \infty} W^{on}(X), \quad \forall X \in H(3^2)$$

L'objet A_t est aussi appelé *attracteur* de l'IFS. Il est invariant sous la transformation W et est égal à l'union de N copies de lui-même, transformées chacune par une des différentes transformations $\omega_1, \dots, \omega_N$. L'objet invariant est dit auto-similaire, ou « auto-affine » lorsque les transformées élémentaires ω_i sont affines.

Théorème du collage

Ce théorème fournit une borne supérieure de la distance de Hausdorff h_d entre les itérés sous l'action de la transformation associée à l'IFS, d'un point A inclus dans $H(3^2)$ et l'attracteur A_t de l'IFS.

Théorème :

Soient l'espace métrique $(3^2, d)$, un point A appartenant à $H(3^2)$, et un IFS $\{3^2; \omega_1, \omega_2, \dots, \omega_N\}$ avec le réel $0 < s < 1$ pour facteur de contraction. On démontre la relation :

$$h_d(A, A_t) \leq \frac{1}{1-s} h_d\left(A, \bigcup_{i=1}^N \omega_i(A)\right)$$

Transformée finalement contractante

Soit une transformation Lipschitzienne ω . S'il existe un entier n tel que la transformée ω^{on} soit contractante, alors ω est dite finalement contractante. L'entier n est appelé exposant de contraction.

L'opérateur W défini par (1) peut être finalement contractant même si un nombre limité de transformées ω_i ne sont pas contractantes.

Théorème du collage généralisé

Considérons l'opérateur W finalement contractant, ayant pour exposant de contraction l'entier n ; il existe alors un point fixe unique $x_f \in 3^2$ tel que :

$$x_f = W(x_f) = \lim_{k \rightarrow \infty} W^{ok}(x) \quad \forall x \in 3^2$$

Dans ce cas,

$$h_d(A, A_t) \leq \frac{1}{1-s} \frac{1-\sigma^n}{1-\sigma} h_d(A, W(A))$$

où σ est le facteur de Lipschitz de W et s est le facteur de contraction de W^{on} .

Exemple d'IFS

L'image obtenue comme point fixe d'IFS possède quelques propriétés spécifiques. En particulier elle est autosimilaire : elle est formée par la réunion de copies de l'image modifiée par des transformées (cf. Figure 3.1). Dans cet exemple, le système de transformations est composé d'une réduction, suivie d'un repositionnement aux sommets d'un triangle effectué par une translation. L'image finale obtenue, la fractale, ne dépend pas de l'image initiale et cet IFS choisi dans la Figure 3.1 permet d'obtenir le célèbre triangle de Sierpinski.

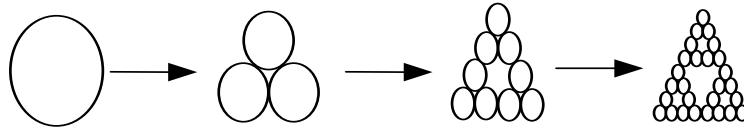


Figure 3.1: Génération d'image fractale

La méthode précédente permet de construire des images fractales mais elle ne nous fournit pas encore le moyen de définir les transformées correspondant à un point fixe donné, à une image donnée. Le but de la compression fractale est de fournir une méthode de construction de ces transformées. De plus, les images obtenues ici sont des images aux propriétés mathématiques très particulières, propriétés que nous ne retrouvons quasiment jamais dans les images que nous voulons traiter.

3.1.3. Compression des images naturelles

Afin d'adapter cette méthode à des images naturelles, on ne s'attache plus à des propriétés globales de l'image, les transformées ω_i sont limitées à des applications soulignant les similarités entre certaines parties de l'image. On peut définir des sous parties de l'image constituant les ensembles de définition d'applications contractantes dans le domaine global.

Alors, le modèle choisi est un système de fonctions itératives partitionnées (PIFS). Un PIFS définit une transformée ω construite à partir de l'union des fonctions contractantes affines du PIFS :

$$W(I) = \omega_1(I_1) \cup \omega_2(I_2) \dots \cup \omega_n(I_n)$$

où l'ensemble de toutes les transformées des sous-images I_i permet d'obtenir entièrement le domaine spatial de I comme ensemble d'arrivée (cf. Figure 3.2). Ainsi, $\{\omega_i(I_i), i = 1 \text{ à } n\}$ constitue une partition du domaine de l'image.

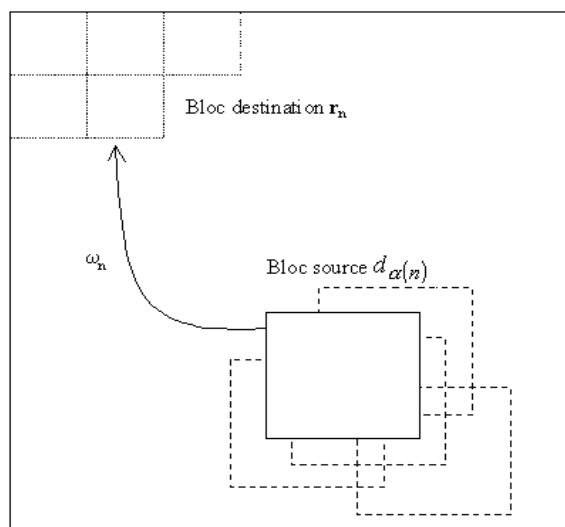
Toute image fractale est alors définie comme l'attracteur d'un IFS ou d'un PIFS. Cet attracteur peut être obtenu à partir d'une image quelconque. Pour des images réelles, on a recours aux PIFS. La plupart des définitions et des propriétés énoncées précédemment restent valables.

Etant donnée une image initiale I , le but de la phase de compression est de déterminer des transformées en nombre suffisant qui constituent un PIFS admettant l'image initiale I comme point fixe. Pour construire le PIFS, l'image est partitionnée en sous-images R appelées Ranges ou *blocs destination*. Ces Ranges doivent être interprétés comme le résultat d'une transformée géométrique affine et contractante ω_i de Domaines D . Ces Domaines sont eux-mêmes des sous-images de l'image initiale qui doivent être choisis le plus judicieusement possible. L'ensemble des couples (D_i, ω_i) constitue une information équivalente à la donnée de l'image initiale qui pourra être reconstruite à partir de toute image de même taille que l'image initiale par application itérée de l'ensemble des transformations ω sur les domaines D . Il n'est pas nécessaire de stocker les ranges R_i puisque l'on peut les régénérer par la relation : $\omega_i(D_i) = R_i$.

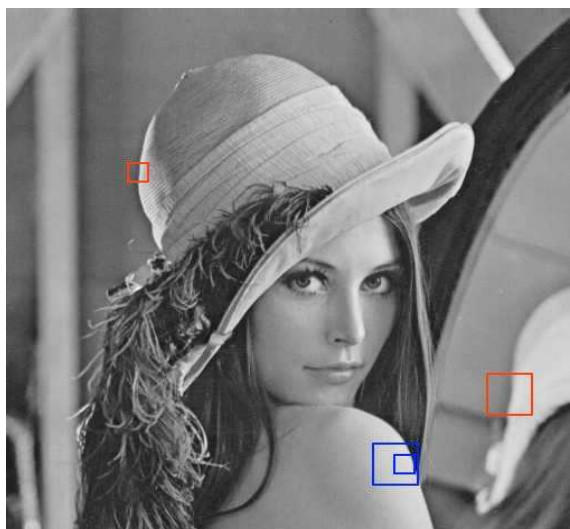
Dans la pratique on s'efforce de limiter le nombre des transformations possibles nécessaires pour passer d'un domaine D à un range R . Toute partie R de l'image est ainsi associée à un Domaine D en considérant que les transformées possibles sont à choisir parmi un ensemble fini de transformations ω_i et minimise la distance entre R et $\omega_i(D)$. C'est la métrique usuelle des moindres carrés (RMS) qui est le plus souvent utilisée :

$$d_{rms}(R, \omega_i(D)) = \sqrt{\int_{R^2} (R(x, y) - \omega_i(D(x, y)))^2 dx dy}$$

Evidemment l'égalité entre R et $\omega_i(D)$ n'est pas toujours assurée, ce qui entraîne que la restitution de l'image initiale ne pourra être faite sans perte. Les pertes ont deux origines, principales, avec d'une part le choix approximatif des transformées mais également le calcul de la limite des itérées approximé par un des éléments de la suite d'indice assez élevé.



(a) décomposition théorique de l'image en range et position des domaines



(b) exemples de ranges et domaines qui se correspondent

Figure 3.2 : Méthode de la compression fractale avec (a) la représentation théorique, et (b) la représentation appliquée à une image

Comme dans la plupart des applications, nous limitons l'exploration des transformées à celle des transformations affines dont la définition analytique est la suivante :

$$\omega \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{bmatrix} a & b & 0 \\ c & d & 0 \\ 0 & 0 & s \end{bmatrix} \times \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} e \\ f \\ o \end{bmatrix} \quad (2)$$

Les paramètres (x,y) indiquent les coordonnées des pixels de l'image et z leur niveau de gris. Les coefficients a, b, c, d, e et f déterminent la partie géométrique de la transformation, e et f précisent la translation, o le contraste et s la luminosité. Il est montré que pour obtenir un ensemble de transformées contractantes, le paramètre s [JAC90] doit être strictement inférieur à 1 cela pour assurer la contractance. La contractance de la partie spatiale de la transformation est souvent forcée de manière indépendante à $\frac{1}{2}$, ce qui fixe la valeur de s . Le processus de la compression fractale revient donc à remplacer l'image initiale par un système de transformations. Chacune de ces transformations est définie par un ensemble de 8 paramètres scalaires.

3.1.3.1. Méthode de A. Jacquin

L'approche de A. Jacquin [JAC90] est basée sur une partition en Ranges régulière et à géométrie carrée. L'image est partitionnée en blocs carrés de largeur fixe, égale par exemple à $B = 8$ pixels. Le principe est alors de rechercher, pour chacun des Ranges r_n , le Domaine carré $d_{\alpha(n)}$ de largeur $D = 2B$ qui minimise l'erreur $d(r_n, \hat{r}_n)$ où \hat{r}_n est l'approximation de r_n calculée à partir d'un bloc de largeur D : $\min_{D,i} d(r_n, \omega_i(D)) = d(r_n, d_{\alpha(n)})$ et $\hat{r}_n = \omega_{\alpha(n)}(d_n)$.

La mesure d'erreur d est donnée par la somme quadratique :

$$d(r_n, \hat{r}_n) = \sqrt{\sum_{j=1}^{B^2} (r_{nj} - \hat{r}_{nj})^2}$$

où r_{nj} et \hat{r}_{nj} sont respectivement les valeurs des pixels d'indice j à l'intérieur d'une part du bloc original r_n et d'autre part de celui du bloc collé \hat{r}_n , c'est en fait l'image du domaine $d_{\alpha(n)}$ obtenue par la meilleure transformation.

L'opération de collage d'un bloc source $d_{\alpha(n)}$ sur un bloc destination r_n , réalisée par la transformation $\omega_{\alpha(n)}$, se décompose en deux parties :

- Une transformation spatiale déforme le support du bloc $d_{\alpha(n)}$

- Une transformation « massique » agit sur la luminance des pixels du bloc $d_{\alpha(n)}$ déformé

La *transformation spatiale* ramène le bloc source $d_{\alpha(n)}$ de taille $D \times D$ ($D = 2 \times B$) à l'échelle et à la position du bloc destination r_n de taille $B \times B$. Le bloc ainsi transformé, noté $b_2^{(n)}$, est obtenu par décimation des pixels du bloc source, ici on utilise un moyennage des niveaux de gris de 4 pixels voisins : le niveau de gris d'un pixel de coordonnées (x_k, y_l) dans $b_2^{(n)}$ est défini par l'équation suivante :

$$b_2^{(n)}(x_k, y_l) = \frac{1}{4} [d_{\alpha(n)}(x_k, y_l) + d_{\alpha(n)}(x_k, y_{l+1}) + d_{\alpha(n)}(x_{k+1}, y_l) + d_{\alpha(n)}(x_{k+1}, y_{l+1})]$$

dans laquelle (x_k, y_l) sont les coordonnées d'un pixel de niveau de gris noté $d_{\alpha(n)}(x_k, y_l)$ à l'intérieur du bloc $d_{\alpha(n)}$.

La *transformation massique* agit sur le bloc $b_2^{(n)}$ pour approximer le bloc destination r_n . La complexité de cette transformation dépend de la nature du bloc r_n considéré. Trois classes de blocs peuvent être envisagées, celle qui regroupe les blocs homogènes, celle qui concerne les blocs texturés et celle groupant les blocs avec contours (simples et divisés) de l'image. Selon la classe à laquelle appartient le bloc destination r_n , une transformation massique plus ou moins complexe lui est associée. Celle-ci dépend du bloc décimé $b_2^{(n)}$ et/ou d'un bloc constant $b_1^{(n)}$ formé de pixels tous égaux à 1. Au bloc $b_2^{(n)}$ sera associé un coefficient d'échelle noté $\beta_2^{(n)}$ et au bloc $b_1^{(n)}$ un coefficient de décalage $\beta_1^{(n)}$. Le choix du type de transformation dépend d'une fonction linéaire définie par :

$$\hat{r}_n = \iota_n (\beta_2^{(n)} b_2^{(n)} + \beta_1^{(n)} b_1^{(n)})$$

où $\beta_2^{(n)} \in \{0,5; 0,6; 0,7; 0,8; 0,9; 1\}$, $\beta_1^{(n)} \in [-255, 255]$, et ι_n représente 8 isométries discrètes (rotations de 0, +90, -90, et +180 degrés, réflexions suivant les axes vertical et horizontal, et réflexions suivant les deux axes diagonaux).

Le coefficient d'échelle β_2 est calculé de manière à rendre égaux les écarts types des deux blocs $b_2^{(n)}$ et r_n . β_2 est ensuite approximé par le coefficient appartenant à un ensemble de

valeurs prédéfinies réelles positives, et inférieures à 1. Le coefficient de décalage β_1 est calculé de manière à ce que les moyennes des pixels des deux blocs $b_2^{(n)}$ et r_n soient égales.

La recherche exhaustive du bloc source $d_{\alpha(n)}$ est effectuée en déplaçant sur l'image un bloc carré d'un pas de $\delta_h = \delta_v = 4$ pixels dans les directions horizontale et verticale. Lorsque les deux blocs sont comparés, les 8 isométries discrètes sont considérées. Pour une image de taille 256×256 , une telle recherche est ainsi effectuée au travers d'une librairie composée de Q blocs source tels que :

$$Q = 8 \left(\frac{256-16}{4} + 1 \right)^2 = 29768$$

Dans le cas d'une image de taille 512×512 , le nombre Q de blocs source possibles s'élève à 125000. Les temps de calcul nécessaires au traitement d'un tel découpage deviennent assez conséquents.

Une procédure de calcul simplifiée a été proposée par Y. Fisher.

3.1.3.2. Méthode de Y. Fisher

Y. Fisher [FIS95] décrit l'opération de collage d'un bloc source sur un bloc destination en utilisant une formule unique :

$$\omega_n \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} a_n & b_n & 0 \\ c_n & d_n & 0 \\ 0 & 0 & \beta_2^{(n)} \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} e_n \\ f_n \\ \beta_1^{(n)} \end{pmatrix}$$

où (x, y) sont les coordonnées d'un pixel intérieur au bloc source $d_{\alpha(n)}$, et z est le niveau de gris du pixel. a_n, b_n, c_n, d_n, e_n et f_n sont les coefficients de la transformation spatiale affine ramenant les pixels du bloc source $d_{\alpha(n)}$ à l'intérieur du bloc destination r_n . $\beta_1^{(n)}$ et $\beta_2^{(n)}$ sont les coefficients de la transformation des niveaux de gris des pixels. Jacobs *et al.* [JAC92] ont montré que dans ce cas, la quantification des coefficients de translation et d'échelle, prise respectivement sur 7 et 5 bits, est optimale en terme de qualité visuelle des images reconstruites.

Calcul des coefficients optimaux de la transformation massique

Pour un bloc destination r , l'approximation \hat{r} est donnée en chaque point par :

$$\hat{r} = \beta_2 b_2 + \beta_1 b_1$$

Le calcul des coefficients de la transformation massique est un problème d'optimisation dans un sous-espace linéaire X de l'espace vectoriel 3^{B^2} . Pour un bloc destination r et un bloc source décimé b_2 , les 6 coefficients spatiaux étant déterminés, le but est alors de trouver les coefficients β_1 et β_2 optimaux minimisant la distance d au sens des moindres carrées entre r et son collage \hat{r} . Dans une deuxième étape on choisira au mieux le bloc source.

Théorème de projection : L'approximation optimale, au sens de la norme euclidienne usuelle, d'un vecteur r élément de 3^{B^2} , dans le sous-espace linéaire X est le vecteur \hat{r} élément de X rendant le vecteur résiduel $r - \hat{r}$ orthogonal à X , c'est-à-dire à tous les vecteurs engendrant le sous-espace X .

Soit le produit scalaire de deux vecteurs x et y sur l'espace 3^{B^2} défini par l'expression suivante :

$$\langle x, y \rangle = \sum_{j=1}^{B^2} (x_j \times y_j) \quad \forall x, y \in 3^{B^2}$$

Déterminer les 2 coefficients β_1 et β_2 optimaux permettant de trouver le vecteur \hat{r} approximant au mieux le vecteur r dans la base b_1, b_2 revient à annuler les deux produits scalaires :

$$\begin{cases} \langle r - \beta_1 b_1 - \beta_2 b_2, b_1 \rangle = 0 \\ \langle r - \beta_1 b_1 - \beta_2 b_2, b_2 \rangle = 0 \end{cases}$$

Il s'agit de résoudre en β_1 et β_2 deux équations. Les coefficients β_1 et β_2 optimaux ainsi calculés ne sont pas contraints, et la contraction de l'opérateur de collage, du PIFS n'est pas assurée. Jacobs et *al.* montrent que le fait de fixer un seuil à 1,5 sur le module du coefficient β_2 assure la convergence finale de la transformation fractale.

Finalement à chaque bloc destination sont associés un domaine et une transformation qui sont stockés et qui suffisent à reconstituer l'image initiale comme point fixe de cette transformation.

3.1.4. Décompression

Nous ne dirons ici que quelques mots sur l'étape de décompression. Elle n'est pas l'objectif de notre étude mais elle nous montre que les renseignements stockés comportent la plus grande partie de l'information contenue dans l'image. Dans l'étape de la décompression, la transformée globale construite à partir de toutes les transformées élémentaires définies sur les domaines sources est itérativement appliquée. C'est-à-dire qu'à chaque itération tous les sous-ensembles d'une image sont traités et le traitement se poursuit jusqu'à ce que le point fixe soit obtenu. On admettra que la convergence est atteinte quand la différence entre deux images successives de la séquence est inférieure à un seuil fixé (généralement proche de zéro).

Pour quantifier la qualité de la compression fractale d'une image, ce n'est pas le taux de compression qui est utilisé, celui-ci ne donne aucun renseignement sur la qualité de l'image qui peut être reconstruite, mais c'est le rapport signal/bruit (PSNR) qui est généralement utilisé. Dans le cas d'une image constituée de n pixels codés sur 256 niveaux de gris, en notant z_i et z'_i les niveaux de gris respectifs du pixel i de l'image initiale et de l'image décompressée, l'expression de ce rapport est alors :

$$PSNR = 10 \times \log \left(\frac{255^2}{rms} \right) \text{ avec } rms = \frac{1}{n} \sum_{i=1}^n (z_i - z'_i)^2 \quad (2)$$

Pour que le processus de décompression soit de qualité, il faut que le PSNR exprimé en décibels soit élevé. En général, une image décompressée est considérée comme étant de bonne qualité lorsque le PSNR calculé entre l'image initiale et l'image décompressée avoisine 30dB. Elle est considérée de qualité excellente quand il est supérieur à 35dB [FIS95].

Avant d'expliquer comment nous avons appliqué ces principes de la compression fractale à notre problème d'identification de scripteur, nous allons montrer l'effet d'une compression fractale sur une image particulière qui est celle d'un document manuscrit.

3.1.5. Application à l'écriture

Les documents contenant les échantillons des écritures sont numérisés avec une résolution de 200 ppp (points par pouce). Nous avons choisi de considérer des images en niveaux de gris, plutôt que binaires, de manière à conserver le maximum d'information. Conjointement à la méthode de compression, nous utilisons les propriétés de similarité mises en avant dans la construction du PIFS, cf. paragraphe 3.1.3. Les images à traiter sont codées sur 256 niveaux de gris (de 0 à 255) ce qui permet de représenter très précisément les variabilités de style,

d'épaisseur ou de pression. En effet, certains détails apparaissant dans l'image en niveaux de gris ne sont pas visibles dans les images binaires.

Pour la compression fractale, un découpage de l'image en Ranges est nécessaire. Il peut être effectué de différentes façons. Les procédés les plus fréquents utilisent une décomposition en carrés ou rectangles de même taille, une décomposition basée sur un découpage en Quadtree défini par l'homogénéité des domaines, ou une partition reposant sur une triangulation de Delaunay s'adaptant au mieux aux caractéristiques propres à l'image. Cette dernière méthode conduit généralement à des résultats de meilleure qualité mais elle est évidemment plus lente. Dans le cas qui nous intéresse, nous pouvons noter que l'écriture occidentale admet les directions verticales et horizontales comme directions privilégiées. Pour adapter la technique de compression fractale à l'écriture, les directions verticales sont aussi privilégiées et les Ranges comme les Domaines ont été choisis sous la forme de carrés à côtés parallèles aux bords de l'image et la taille est à déterminer en fonction de la précision désirée.

Nous illustrons la capacité de la méthode à compresser des images d'écriture grâce aux figures suivantes qui montrent le résultat d'une image décompressée (cf. Figure 3.4). La méthode utilisée est celle décrite dans les paragraphes précédents à partir d'un texte écrit (cf. Figure 3.3). Nous avons limité l'ensemble des transformations possibles des domaines en ranges à l'utilisation d'une seule transformée : l'homothétie de rapport $\frac{1}{2}$. C'est dire que b_n et c_n sont fixés égaux à 0 et a_n et d_n à $\frac{1}{2}$. Sur l'exemple présenté, on peut observer que le PSNR obtenu prend une valeur de 39,51dB ce qui indique que la qualité de l'image décompressée est tout à fait satisfaisante. La lecture de l'image reconstruite ne pose d'ailleurs pas de problème au lecteur humain.

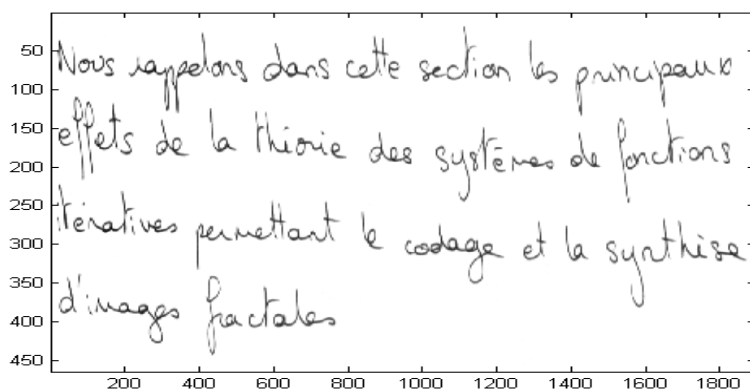


Figure 3.3 : Exemple d'image d'écriture scannée à 200 ppp

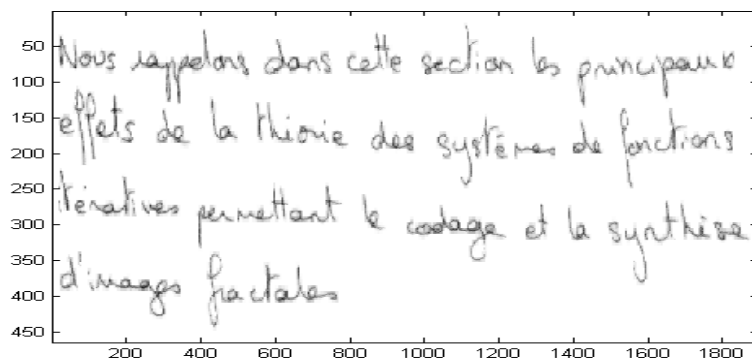


Figure 3.4 : Image précédente décompressée après compression fractale, avec un PSNR de 39,51dB

3.2. Mise en place des données et des paramètres

La qualité des résultats dépend évidemment de différents paramètres qui doivent être fixés judicieusement au cours du traitement. Parmi eux certains dépendent de la manière dont sont collectés les textes manuscrits tels que les réglages du scanner. D'autres dépendent de la mise en forme des textes au sein de l'image : prétraitement de recalage, débruitage, etc. Enfin certains sont liés à la technique proprement dite de compression fractale utilisée dans le processus.

3.2.1. Acquisition des textes manuscrits

Les textes manuscrits ont été réalisés par une vingtaine de scripteurs volontaires travaillant sur le campus de l'Université de Toulon et du Var. Les scripteurs ne sont pas seulement des étudiants mais aussi des membres du personnel de l'Université dans une tranche d'âge de 23 à 50 ans. Tous les textes ont été écrits avec un stylo à bille noir identique afin d'éliminer les variances de ton d'encre qui seraient liées à l'outil d'écriture plutôt qu'au scripteur. Néanmoins ce facteur ne devrait pas influencer la qualité de la méthode car les transformations recherchées se font relativement à l'image elle-même et jamais de manière absolue. Il a été demandé à chaque scripteur d'écrire deux textes de longueurs différentes. Le premier, le plus long, tient sur 3 ou 4 lignes selon le scripteur et il est utilisé pour l'étape d'apprentissage. Le texte est le même pour tous mais cela n'est pas théoriquement nécessaire pour notre application. Il est composé de la phrase suivante :

« Nous rappelons dans cette section les principaux effets de la théorie des systèmes de fonctions itératives permettant le codage et la synthèse d'images fractales ».

Le second texte plus court tient sur 1 ou 2 lignes, et il est différent du premier. Il sera utilisé pour tester notre méthode. Ce texte est :

« L'écriture suit-elle les règles de la géométrie fractale ? ».

Dans l'Annexe 1, un exemple de pages utilisées dans notre étude est montré.

Les textes manuscrits sont numérisés. Une numérisation sur 256 niveaux de gris avec une résolution à 200 ppp (points par pouce) a été considérée comme étant la meilleure solution. La valeur du premier paramètre qui concerne le nombre de niveaux de gris considérés a été choisie de manière à avoir le maximum de détails sur le tracé et donc sur le style d'écriture des scripteurs. Le deuxième paramètre nous permet d'obtenir une résolution spatiale suffisante. Nous réalisons ainsi un compromis entre la taille du fichier associé et la quantité d'information conservée. Nous avons en cela suivi les indications qui sont justifiées dans la thèse de V. Bouléreau [BOU97]. Les images numérisées sont sauvegardées dans un format « bitmap » sans compression. Leur taille varie entre 1400 et 1500 pixels en longueur et entre 300 et 500 pixels en largeur.

3.2.2. Les prétraitements

Lors de la description détaillée de la théorie de la compression fractale, nous avons pu constater que plusieurs paramètres doivent être initialisés. Du choix des valeurs dépend la qualité de la compression et de la décompression fractales. Ces paramètres sont essentiellement liés à la technique de découpage des images en Ranges et en Domaines. Notre objectif n'étant pas d'obtenir la meilleure qualité des images décompressées mais d'appliquer cette technique à l'identification de scripteur, nous n'allons pas traiter les textes tels qu'ils apparaissent dans les images.

La plupart des scripteurs n'écrivent pas de façon rectiligne lorsqu'il n'y a pas de repère sur la feuille. Lorsque nous avons construit la base de données des textes nous avons demandé à chaque individu participant à l'expérience de rédiger les échantillons d'écriture sur des feuilles blanches, et inévitablement presque tous ont écrit leurs textes en suivant des lignes inclinées par rapport au bord de la page.

De manière à fixer une taille de range qui soit significative par rapport aux détails des écritures, nous devons procéder à une normalisation du corps de texte de toutes les écritures.

De plus, pour rester au plus proche des mots écrits dans des mouvements distincts, la détermination de la partition de l'image ne sera pas réalisée sur l'image tout entière mais sur les rectangles circonscrits associés à chaque mot. Les zones de fond ne sont donc pas traitées comme les zones de texte.

C'est pourquoi nous incluons avant tout traitement une étape de prétraitement qui comporte deux phases. Une phase de redressement des lignes de base de chaque scripteur précède une étape d'extraction des mots sur l'image représentant les lignes redressées en niveaux de gris.

Redressement de la ligne de base de l'écriture

Pour redresser le texte, nous proposons de mettre en œuvre une solution itérative jusqu'à minimisation de la hauteur de la ligne considérée par rotation de l'image. Cette hauteur étant donnée par l'histogramme dans le sens vertical de la ligne considérée.

Nous commençons par estimer l'angle entre la première ligne écrite et la direction horizontale représentée par les lignes de pixels. Pour estimer cet angle nous calculons la hauteur H de la ligne à partir de l'histogramme dans le sens horizontal de celle-ci, ainsi que la largeur L de la ligne à partir de l'histogramme dans le sens vertical de celle-ci. Cette technique ne permet pas d'estimer précisément la mesure de l'angle qui conduit à rendre la ligne horizontale. Mais va servir à l'initialisation de la recherche de cet angle.

La rotation de l'image est effectuée pixel par pixel. Un pixel p d'indices (x, y) aura les indices (x', y') après une rotation d'angle θ_i :

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\{\theta_i\} & \sin\{\theta_i\} \\ -\sin\{\theta_i\} & \cos\{\theta_i\} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

Pour s'assurer de la convergence de l'algorithme considérons après N itérations l'angle limite donné par la somme en valeur absolue des θ_i :

$$\sum_{j=0}^N |\theta_j| = \frac{\theta_{init}}{4} + \frac{\theta_{init}}{4} \frac{3}{4} + \dots + \frac{\theta_{init}}{4} \left(\frac{3}{4}\right)^N = \frac{\theta_{init}}{4} \left(1 + \frac{3}{4} + \dots + \left(\frac{3}{4}\right)^N\right)$$

Donc l'angle limite est donné par :

$$\theta_{lim} = \lim_{N \rightarrow +\infty} \sum_{j=0}^N |\theta_j| = \frac{\theta_{init}}{4} \lim_{N \rightarrow +\infty} \left(1 + \frac{3}{4} + \dots + \left(\frac{3}{4}\right)^N\right) = \frac{\theta_{init}}{4} \frac{1}{1 - \frac{3}{4}} = \theta_{init}$$

Comme $|\theta_{init}| > |\theta|$ il est sûr que $\hat{\theta} \approx \theta$.

Le principe est indiqué dans l'algorithme que nous présentons dans le tableau suivant :

Initialisation :

Image initiale I_{init} ; $I_0 = I_{init}$

Calcul de l'histogramme horizontal, et vertical => H_0, L

$$\theta_{init} = \arctan\left\{\frac{H_0}{L}\right\}$$

$$\theta_0 = \frac{\theta_{init}}{4}$$

i=1

La technique utilisée pour calculer θ_{init} nous donne la relation $|\theta_{init}| > |\theta|$ puisque $\theta_{init} = \theta + \theta_c$ lorsque θ_c est l'angle entre la ligne tracée du point le plus bas à gauche au point le plus haut à droite du corps du texte et la ligne inclinée d'angle θ (angle recherché) par rapport à la ligne horizontale. Et ce, que la ligne soit penchée vers le haut ou vers le bas.

Début de la boucle

Application de la rotation d'angle θ_{i-1} à I_{i-1}

Calcul de l'histogramme dans le sens horizontal => H_i

Si $H_i < H_{i-1}$

$$\text{Alors } \theta_i = \frac{3}{4}\theta_{i-1}$$

Si $H_i \geq H_{i-1}$

$$\text{Alors } \theta_i = -\frac{3}{4}\theta_{i-1}$$

Continuer tant que $|\theta_i - \theta_{i-1}| > 10^{-4}$

i=i+1

Fin de la boucle

Estimation de l'angle : $\hat{\theta} = \sum_{j=0}^{i-1} \theta_j$

Application sur l'image initiale I_{init} de la rotation d'angle $\hat{\theta}$ => Image redressée.

Pour illustrer cette technique, considérons le texte de la Figure 3.6 où sont affichés l'image à redresser et les histogrammes dans les sens horizontal et vertical. Les angles θ_{init} , θ_c et θ sont également indiqués.



Figure 3.5 : Exemple d'écriture non rectiligne

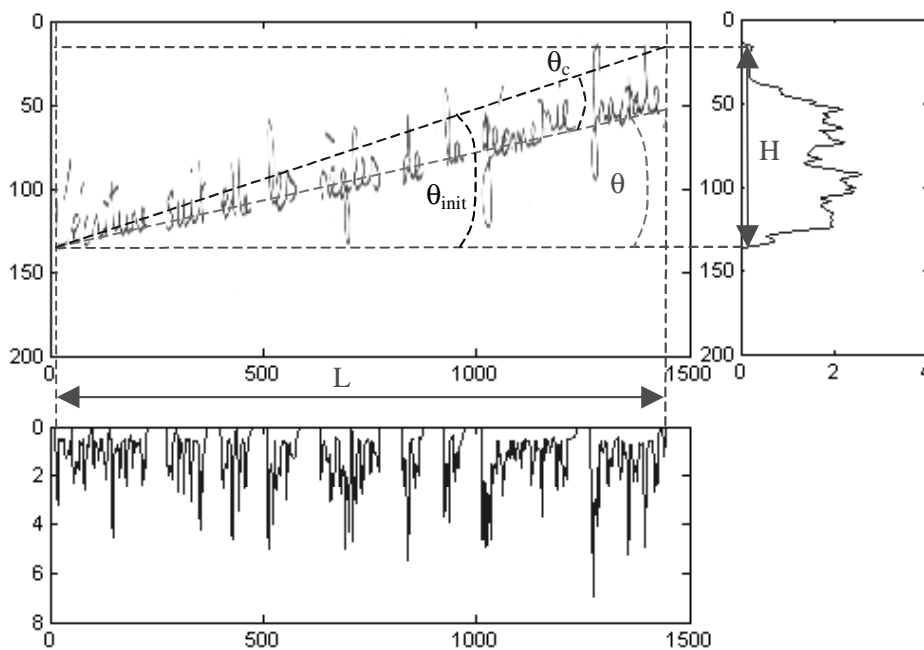


Figure 3.6 : Estimation de l'angle de la ligne de base

L'image redressée est présentée sur la Figure 3.7, après l'estimation de l'angle de rotation par la technique développée précédemment.

L'étape de redressement que nous venons de décrire permet de s'affranchir de différences entre écritures qui seraient dues à un mauvais positionnement du support de l'écriture. Nous allons maintenant tenter de normaliser la taille des textes.

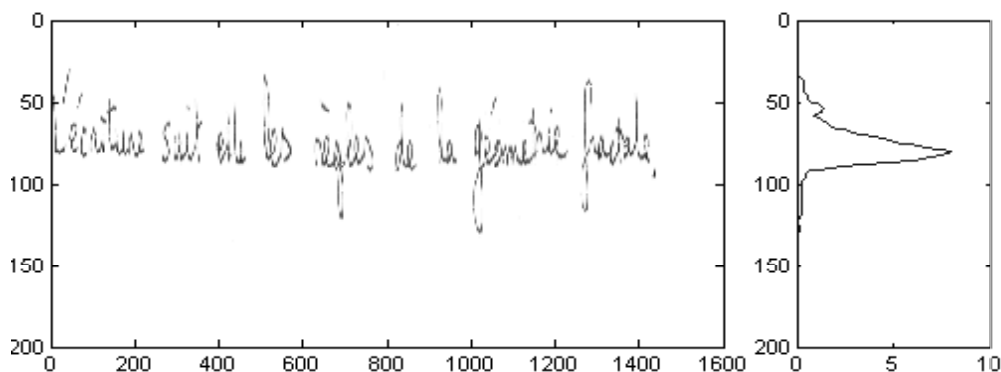


Figure 3.7 : Image redressée du texte présenté sur la Figure 3.5

Normalisation de l'écriture

Il est évident que la comparaison entre écritures n'est possible que si les tailles de celles-ci sont identiques ou très proches. Il est donc nécessaire d'appliquer une étape de normalisation de taille. Cette étape permet un choix uniforme pour les dimensions des imagettes qui constituent les ranges et matérialisent l'échelle d'observation considérée. Les écritures sont normalisées par rapport à la hauteur du corps des lettres. Une homothétie dont le rapport est calculé en considérant uniquement la direction verticale est appliquée sur l'image de manière isotrope.

Nous ne modifions pas la longueur de l'écriture car écrire plus ou moins délié fait partie du style du scripteur. D'autre part le rapport entre la longueur des extensions et le corps des lettres peut être également considéré comme un élément du style.

Extraction des mots

Nous voulons dans cette étape isoler chaque mot du texte manuscrit et appliquer un découpage spécifique décrit dans le paragraphe suivant. Si l'on observe un texte manuscrit, on se rend compte que très souvent les mots ne correspondent pas à une unique composante connexe du texte. Ce point est plus ou moins accentué selon le style du scripteur. Néanmoins on peut considérer que les composantes constituant les mots sont bien plus proches entre elles que ne le sont les mots eux-mêmes. Cette étape d'extraction des mots repose sur une méthode morphologique qui met en œuvre une dilatation suivie d'un processus de recherche des pixels connectés. L'exactitude sémantique de la phase n'est pas essentielle pour notre application,

notre objectif essentiel étant de ne pas traiter ultérieurement sur une même zone d'images des ensembles de traits trop isolés les uns des autres.

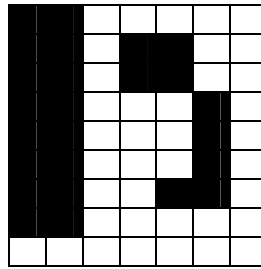


Figure 3.8 : Exemple d'image déjà dilatée pour la recherche d'éléments connectés

Pour la recherche des pixels connectés, deux types de précisions peuvent être envisagés selon qu'on se réfère à la 4-connexité ou à la 8-connexité. Dans l'exemple de la Figure 3.8 la méthode trouve 3 zones si la précision choisie est associée à la 4-connexité et 2 zones si la précision est associée à la 8-connexité. Une étude sur les échantillons nous servant de base d'apprentissage nous a conduit à choisir la 8-connexité.

Rappelons la définition de l'opérateur morphologique de dilatation :

Soit un élément structurant B et une forme X dans une image. La dilatation de X par B est l'ensemble des points p tels que l'ensemble B , translaté de façon que son centre se trouve en p , a une intersection non vide avec X :

$$\delta_b(X) = X \oplus B = \{p : B'_p \cap X \neq \emptyset\}$$

Un exemple concret de dilatation est montré sur la Figure 3.9.

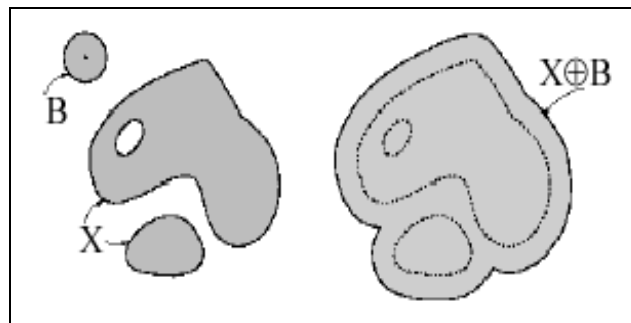


Figure 3.9 : Exemple de dilatation

La dilatation que nous considérerons utilise comme élément structurant, la matrice : $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$,

le centre de l'élément structurant se trouvant au centre de la matrice.

Sur la Figure 3.10 nous pouvons suivre les différentes étapes du traitement. Sur l'image binarisée, une dilatation est appliquée 10 fois sur l'écriture (cf. Figure 3.10(b)). Les composantes connexes sont alors extraites (Figure 3.10(c)).

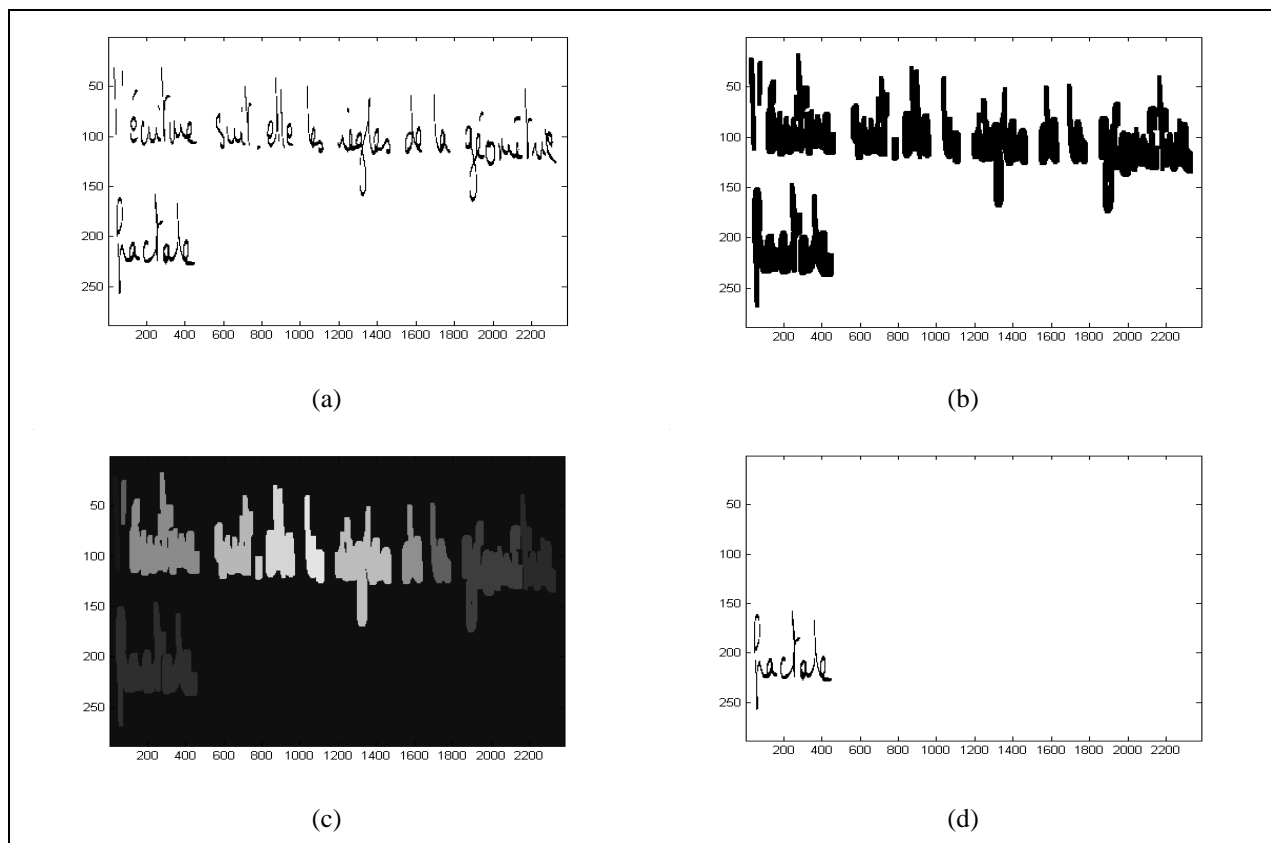


Figure 3.10 : Isolement d'un mot avec les étapes successives suivantes (a) binarisation du manuscrit, (b) dilatation du texte, (c) extraction des composantes, (d) récupération d'un mot

Cette extraction des mots est illustrée sur la Figure 3.10, elle nous permettra, tout en conservant une partition simple de l'image en ranges rectangulaires, de mieux positionner les éléments de la partition.

3.2.3. Paramètres théoriques

Si l'on considère maintenant les paramètres qui dépendent directement de l'approche fractale certains sont choisis en fonction de la littérature relative à la compression fractale, d'autres à partir de critères établis dans le domaine de la reconnaissance d'écriture. Les premiers concernent le type de découpage de l'image, ensuite nous aborderons l'étude de la taille des éléments du découpage. D'autres éléments peuvent être choisis dans la méthode. Il s'agit en particulier du type et du nombre de transformations élémentaires qui sont considérées entre

les ranges et les domaines. La méthode dépend aussi de la mesure utilisée pour qualifier le caractère similaire de deux sous-images.

Découpage des images

Le découpage des images, c'est à dire de l'écriture, est une partie importante du processus. Il conditionne la robustesse de la méthode et la fiabilité des résultats. Même si la compression fractale n'est pas utilisée en tant que technique de compression d'images, elle est utilisée en tant qu'extracteur d'éléments inhérents au scripteur. C'est pour cela que nous avons opté pour un découpage lié à la notion d'écriture et plus précisément de trait, c'est-à-dire en privilégiant la direction verticale. Nous utilisons un découpage en carrés en ce qui concerne les ranges qui contiennent une portion de trait. Evidemment ces carrés sont sans recouvrement pour réaliser la partition. Nous ne porterons que peu d'attention aux zones ne contenant pas de portion de trait d'écriture (zones blanches).

La technique la plus simple consisterait à découper la totalité de l'image régulièrement de gauche à droite et de bas en haut par exemple. Toutefois, comme nous ne traitons pas les imageries vides, ou blanches, nous avons préféré mettre en œuvre un découpage mot par mot ou plus exactement composantes connexes par composantes connexes. L'origine verticale étant ainsi propre à chaque mot, on ajuste le quadrillage horizontalement pour recouvrir le mot avec un minimum d'imageries. Ainsi l'image reste découpée de façon régulière selon les lignes et sur chaque ligne un carré peut être décalé vers la gauche ou vers la droite pour optimiser le partitionnement de l'écriture. La méthode est illustrée sur la Figure 3.11 à partir d'une des composantes extraites, *a priori* un mot.

La Figure 3.12(b) illustre un des apports du meilleur positionnement des ranges sur un mot. En (a) on observe le positionnement d'un découpage régulier de l'image du mot *fractale* pour lequel on a dessiné les carrés qui contiennent une portion du trait. Ce découpage comporte 169 Ranges « utiles », alors que la Figure 3.12(b) correspond à notre découpage adaptatif et permet de réduire à 139 le nombre de Ranges « utiles » tout en améliorant le positionnement des ranges par rapport au contenu de l'image.

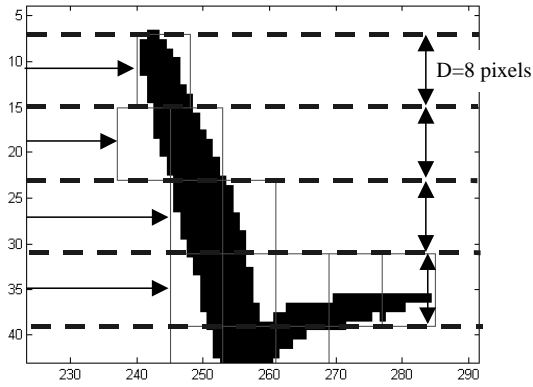
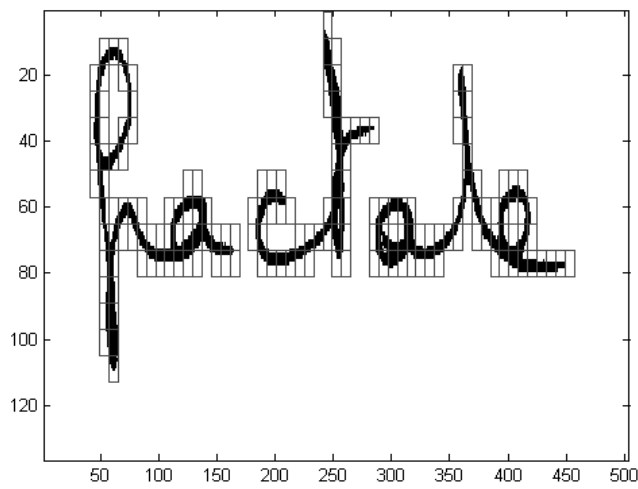
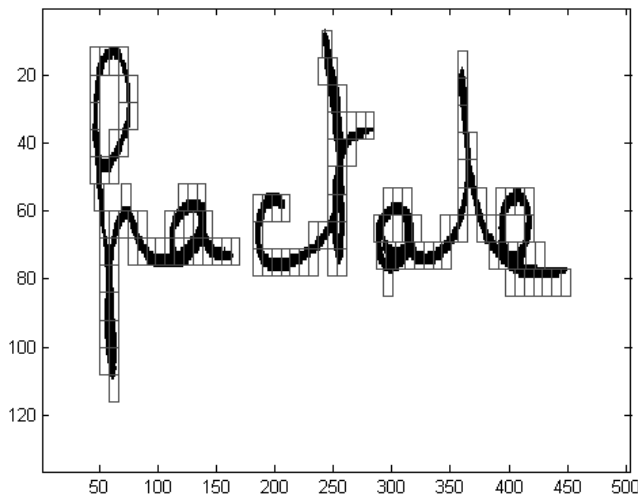


Figure 3.11 : Exemple de notre méthode de découpage



(a) 169 ranges utiles



(b) : 139 ranges utiles

Figure 3.12 : Mot découpé de façon régulière (a), selon la méthode proposée (b)

Taille des Ranges et des Domaines

La technique de positionnement des ranges peut être appliquée quelle que soit la taille des carrés considérés. Dans notre méthode, ces carrés sont de taille fixe mais ils doivent être adaptés aux détails usuels d'une écriture. La taille des Ranges et celle des Domaines sont choisies en fonction de deux critères. Le premier est que la taille doit être assez grande pour contenir assez d'informations sur le style du scripteur et le second est que cette même taille doit être assez petite pour assurer une bonne performance du critère d'identification. Ainsi plusieurs questions se posent : Quelle taille du Range devons-nous choisir ? Quel rapport de contractance entre les Ranges et les Domaines choisir ?

Pour effectuer ce choix nous avons considéré deux critères : le taux d'identification (exprimé en pourcentage) et le nombre de domaines différents utilisés dans la recherche des domaines et des transformations. Deux expérimentations ont été menées. En ce qui concerne le taux d'identification, nous avons fait varier la taille des ranges, utilisé un rapport de $\frac{1}{2}$ entre les tailles des ranges et des domaines, et nous avons appliqué la méthode d'identification à base de compression/décompression fractales (décrite plus précisément dans le paragraphe 3.4) sur notre base de données constituée d'une vingtaine de scripteurs. Les résultats obtenus sont présentés sur la Figure 3.13. Ils montrent que la méthode donne un meilleur taux d'identification pour une taille des ranges de 6 et 8 pixels. La méthode utilisant une taille des ranges à 6 pixels dure environ 7 heures alors qu'avec une taille à 8 pixels, le système dure 4 heures (les tests ont été effectués sur un PC Pentium III, 1.2 GHz et 256M de Ram).

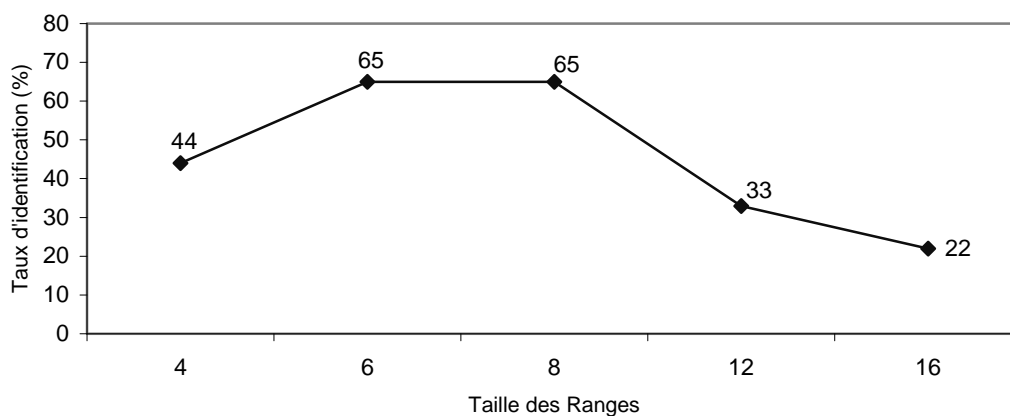


Figure 3.13 : Taux d'identification en fonction de la taille des Ranges

Par contre, on observe sur la Figure 3.14 que lorsque la taille des ranges croît, le nombre de domaines retenus diminue. D'un point de vue combinatoire il faudrait favoriser un nombre

restreint de domaines retenus. Nous devons réaliser un compromis entre les deux critères considérés.

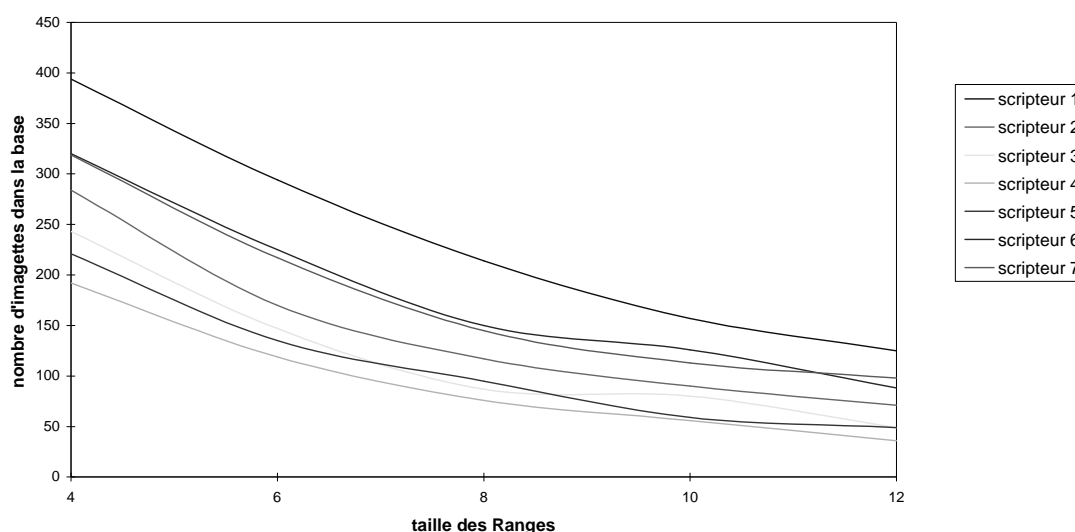


Figure 3.14 : Nombre d'images dans la base par rapport à la taille des Ranges

Nous avons finalement choisi des Ranges carrés de largeur de 8 pixels et un rapport de contractance de $\frac{1}{2}$ qui se traduit par un rapport de taille linéaire entre les Ranges et les Domaines de même valeur.

Normalisation

La taille des ranges et la normalisation des textes sont très liées à un facteur de proportionnalité près. Ainsi, nous avons jugé qu'une lettre devait être représentée par au plus 9 Ranges pour que les éléments puissent être considérés comme caractéristiques du style d'un scripteur.

Nous avons vu que les ranges sont carrés de largeur 8 pixels, aussi pour obtenir au plus 9 Ranges par lettre, le corps du texte doit avoir une hauteur maximum de 24 pixels. Sur la figure ci-dessous, nous voyons qu'avec une taille de 24 pixels, la lettre *e* d'une hauteur de 21 pixels est bien encadrée par les 9 Ranges.

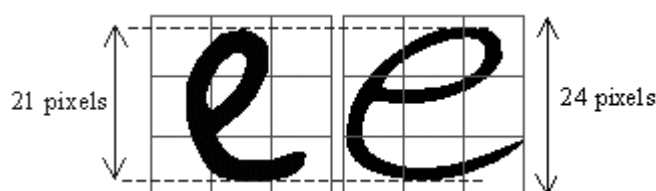


Figure 3.15 : Découpage de la lettre *e* par Ranges de taille 8x8 pour deux scripteurs différents après normalisation

Type et nombre de transformées

La compression fractale s'appuie sur la présence des autosimilarités d'une image. Ces autosimilarités se définissent par l'existence, dans l'image elle-même, d'éléments identiques à une transformée près. Dans le cadre de notre méthode, ces éléments, par leur répétition, sont considérés comme étant propres au scripteur. Or, pour l'usage que nous voulons en faire, ils doivent être recherchés tels quels, c'est-à-dire sans aucune transformation géométrique ou analytique. C'est pour cela que, dans cette application, nous utilisons une seule transformée ; l'homothétie de rapport $\frac{1}{2}$ entre un Domaine et un Range.

Critère de qualité

Le critère de qualité habituellement utilisé dans les processus de compression et décompression fractales est le rapport signal/bruit (PSNR). Il est exprimé en décibels et il varie de 0 à l'infini. L'inconvénient de ce critère est sa plage de variation très grande. C'est pour cela que nous utiliserons conjointement à ce critère un autre critère de qualité, le coefficient de corrélation qui est plus représentatif de l'impression visuelle.

Soient l'image initiale X centrée et l'image à comparer Y centrée elle aussi, toutes deux de même taille $M * N$, le coefficient de corrélation se définit par :

$$CC = \frac{E[XY]}{\sqrt{E[X^2]E[Y^2]}} = \frac{\sum_{i=1}^M \sum_{j=1}^N X_{i,j} Y_{i,j}}{\sqrt{\sum_{i=1}^M \sum_{j=1}^N (X_{i,j})^2 \sum_{i=1}^M \sum_{j=1}^N (Y_{i,j})^2}}$$

où $E[]$ est l'espérance mathématique. Le coefficient de corrélation varie de 0 à 1, il prend la valeur 0 lorsque les deux images ne se ressemblent pas du tout et la valeur 1 lorsqu'elles sont identiques.

Le principe de la compression fractale est le point de départ de l'élaboration de notre méthode d'identification de scripteur. Nous allons montrer comment à partir d'un texte d'apprentissage une base constituée des éléments invariants est ainsi construite pour chaque scripteur. Les paramètres qui viennent d'être définis sont communs aux deux étapes de notre méthode, c'est à dire l'étape d'apprentissage et l'étape d'identification.

3.3. Apprentissage d'un scripteur

Dans ce paragraphe, nous allons montrer comment il est possible d'extraire des éléments invariants à partir de l'écriture [VIN98] en nous basant sur les similarités internes de celle-ci. Ces éléments sont extraits au cours du processus de compression fractale durant lequel les similarités internes ont été révélées dans l'image entre des portions de trait.

Pendant la phase de compression tous les Ranges sont interprétés à partir des Domaines de l'image comme le résultat d'une transformation. La meilleure transformée correspond à celle qui minimise le RMS (Root Mean Square). Pour chaque Range, les paramètres mémorisés sont alors la position du Domaine associé ainsi que la transformée minimisant le critère. Les Domaines possibles sont toutes les sous-images de taille adéquate contenues dans l'image. Nous privilégions les domaines qui apparaissent les premiers durant l'écriture du texte de manière à ne pas augmenter artificiellement le nombre de domaines. Dans la mesure où l'image de l'écriture ne sera pas restituée, nous ne retenons pas la transformée entre le range considéré et le domaine mais seulement le domaine.

Pour chaque scripteur, c'est à dire pour une image de texte manuscrit, une base de référence B_s est construite. Elle est constituée de l'ensemble des Domaines pour lesquels nous faisons l'hypothèse qu'ils représentent les caractéristiques du scripteur. Ils ont été sélectionnés pendant la phase de compression comme référence d'apprentissage. Ce texte doit donc être assez long pour être significatif du style du scripteur et la base ne devrait pas dépendre du contenu du texte.

On pourrait penser que le nombre de Domaines retenus, que nous appellerons désormais imagettes de la base, augmente avec la longueur du texte. Néanmoins, quand le texte est suffisamment long, les motifs qui interviennent se retrouvent et c'est cette constance dans les formes qui constitue le style du scripteur. L'accroissement du nombre des imagettes n'est pas linéaire avec la longueur du texte et, pour chaque scripteur, le nombre des imagettes caractéristiques tend à se stabiliser. On en présente quelques exemples sur la Figure 3.16 pour différents scripteurs, on observe bien le début de l'asymptote. On peut alors raisonnablement supposer que l'ensemble des Domaines contient toute l'information concernant les formes dans l'écriture d'un scripteur. Une preuve de la valeur de cette hypothèse est que ces domaines seraient suffisants pour générer l'image de l'écriture de ce texte d'apprentissage, si le PIFS associé était utilisé dans une phase de décompression de l'image.

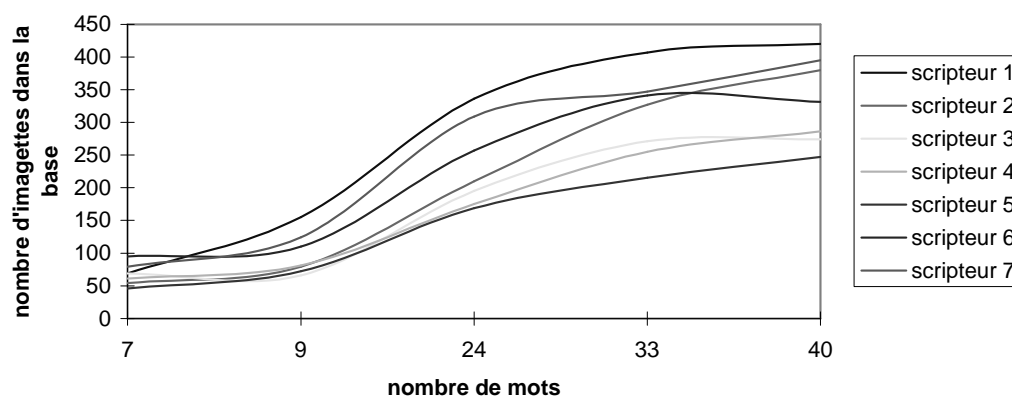


Figure 3.16 : Nombre d'imagettes dans la base en fonction du nombre de mots dans le texte

Nous pouvons admettre que la base de référence ne dépend pas du contenu du texte d'apprentissage quand il est assez long. Les textes peuvent donc être différents pour chaque scripteur, c'est-à-dire qu'un enrôlement de chaque scripteur n'est pas toujours nécessaire si on peut disposer d'un texte déjà écrit par lui. Néanmoins, il est essentiel que chaque base base comprenne tous les aspects invariants relatifs au scripteur.

Nous pouvons résumer la méthode développée par l'algorithme décrit comme suit :

Pour chaque Range

Pour chaque Domaine potentiel

Choix de la meilleure transformée donnant une relation aussi proche que possible de l'égalité.

$R = T(D)$

Choix du meilleur Domaine au sens de l'erreur minimum quadratique entre le Range et l'image du Domaine.

Addition du Domaine dans la base des imagettes de référence après vérification qu'il n'y soit déjà.

Cette base que nous venons de construire, représente les similarités internes contenues dans l'écriture. Pour caractériser au mieux le scripteur nous avons choisi de ne pas retenir toutes les imagettes sélectionnées durant la phase de compression, mais notre choix s'est porté sur les imagettes figurant les similarités les plus représentatives du scripteur. Nous entendons par-là les autosimilarités les plus exactes entre les Ranges et les images des Domaines, la précision étant mesurée par une erreur quadratique. Il ne semble pas raisonnable de conserver les Domaines (ou les Ranges) qui n'ont pas une bonne correspondance. En effet, dans

l'algorithme précédent nous choisissons le meilleur domaine, ou celui inférieur à un seuil prédéfini de manière à accélérer le temps de calcul, mais cela n'assure pas que la similitude entre domaine et range soit vraiment bonne. Suivant le scripteur, l'erreur maximum qui a été rencontrée peut varier. On peut noter qu'une plus ou moins grande dissimilarité entre les formes donne des indications sur le style de l'écriture et nous permettra de nous adapter à la plus ou moins grande variabilité au sein des formes représentatives du scripteur. Ainsi, pour chaque scripteur, nous ne conservons effectivement dans la base que 80% des domaines obtenus lors du processus de compression pour constituer la base de référence de celui-ci. De plus nous sommes en mesure, pour chaque scripteur, de définir un seuil correspondant au seuil de similarité maximal associé aux imagerie finalement retenues. Dans la suite nous noterons ε_s ce seuil associé au scripteur considéré comme référence.

Concrètement ce processus signifie que si un scripteur possède 500 imagerie dans la base, nous ne prendrons en considération pour la suite que 400 imagerie dans la base de référence de ce scripteur. Cette sélection ne se fait pas au hasard, nous classons d'abord les imagerie de la base selon un ordre d'importance que nous allons définir. A la fin du processus de la compression fractale, chaque Domaine peut correspondre à plusieurs Ranges. Or si un Domaine est souvent utilisé comme référence d'une similarité dans l'image, nous le considérons comme plus important que d'autres. A chaque fois que le Domaine considéré est en correspondance avec un Range, nous augmentons son importance, c'est à dire que nous incrémentons un score associé au Domaine. C'est ainsi que la base est classée par ordre d'importance des domaines.

Le seuil associé à chaque scripteur ε_s correspond à la valeur maximale de l'erreur quadratique calculée entre les images des Domaines retenus et les ranges.

La Figure 3.17 montre un exemple de base extraite d'un texte d'apprentissage d'un scripteur. D'un point de vue pratique, la base de référence est constituée des Domaines après réduction d'un facteur $\frac{1}{2}$. Ainsi il est plus facile de comparer les Ranges et les imagerie des bases de référence car ils ont désormais la même taille.

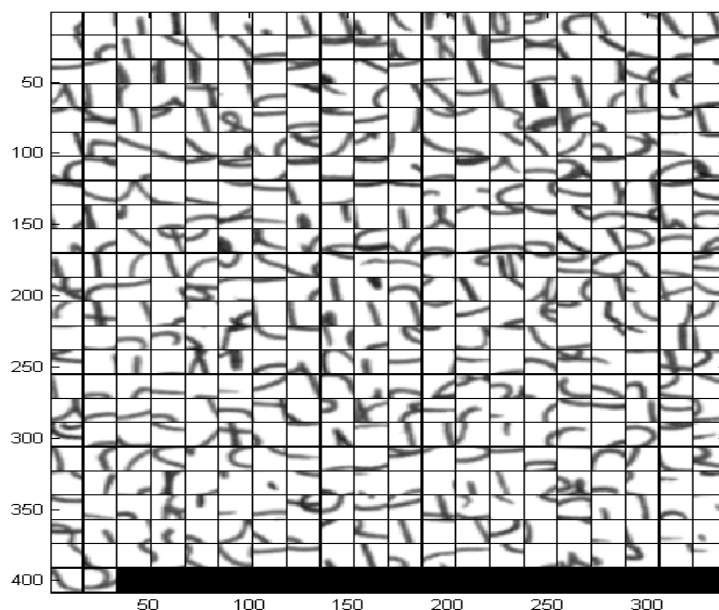


Figure 3.17 : Exemple de base associée à un scripteur

A la fin de la phase d'apprentissage, les scripteurs connus doivent avoir été enrôlés et pour chacun est établie une base de référence ainsi qu'un seuil traduisant la variabilité des formes contenues dans l'écriture. Ce sont tous ces éléments qui seront utilisés dans la phase d'identification du scripteur d'un nouveau texte.

3.4. Identification du scripteur

Nous voulons maintenant pouvoir identifier un scripteur à partir d'un nouveau texte manuscrit. Evidemment, pour être reconnu, le scripteur doit au préalable avoir été caractérisé par sa base de référence établie à partir d'un texte d'apprentissage. Nous disposons donc *a priori* de N bases de référence. Nous pouvons envisager une classe de l'ensemble des scripteurs qui n'ont pas été référencés.

3.4.1. Recherche de similarités

Le nouveau texte pour lequel l'auteur doit être identifié n'est pas nécessairement aussi long que celui utilisé en apprentissage.

Une solution pourrait être de comparer la qualité des images du nouveau texte après compression et décompression fractales en utilisant les bases de référence [SER02]. Comme le montre la Figure 3.18, le principe consiste à appliquer le processus de compression fractale

au nouveau texte en considérant les imagerie de la base comme étant les Domaines de l'image à identifier. Chaque Domaine est remplacé par l'imagerie de la base qui lui ressemble le plus à une erreur quadratique moyenne près. On applique alors le processus de compression/décompression fractale avec les mêmes paramètres que lors de l'apprentissage. Le calcul du PSNR et celui du coefficient de corrélation calculés entre l'image initiale et l'image décompressée nous indiquent l'identité du scripteur. Elle correspond à celle de l'auteur du texte dont ces deux paramètres sont les plus grands.

On peut calculer les taux d'identification correspondant à cette méthode.

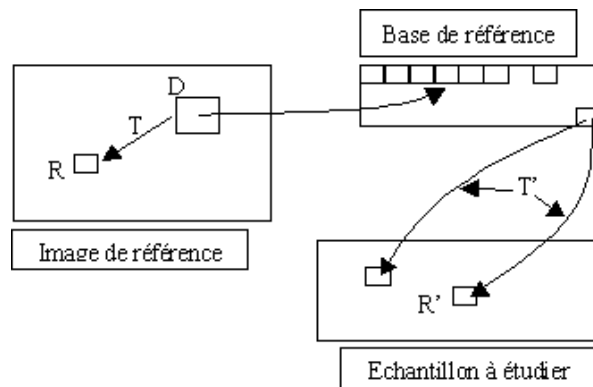
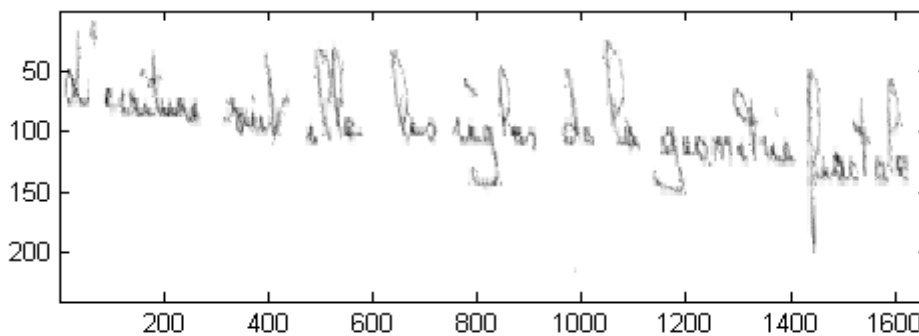


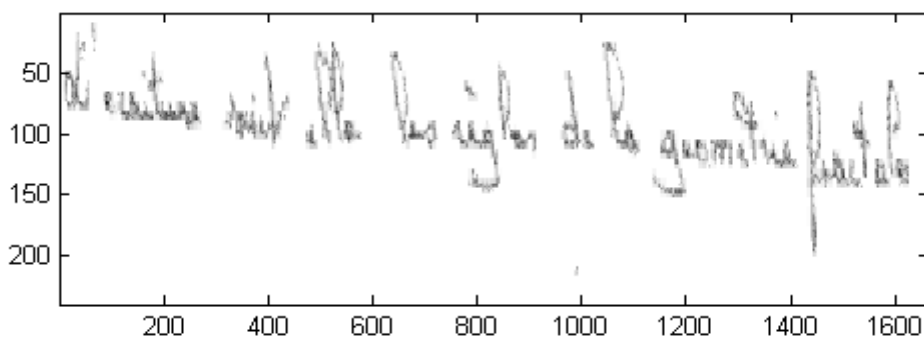
Figure 3.18 : Identification par compression et décompression fractales

d'écriture suit elle les règles de la géométrie fractale

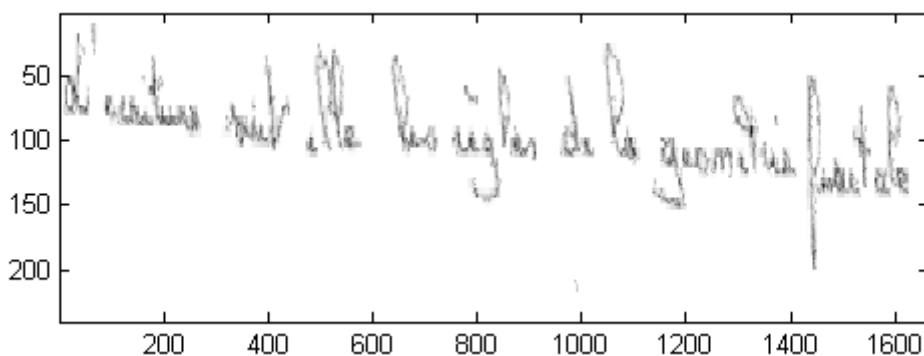
(a) Texte



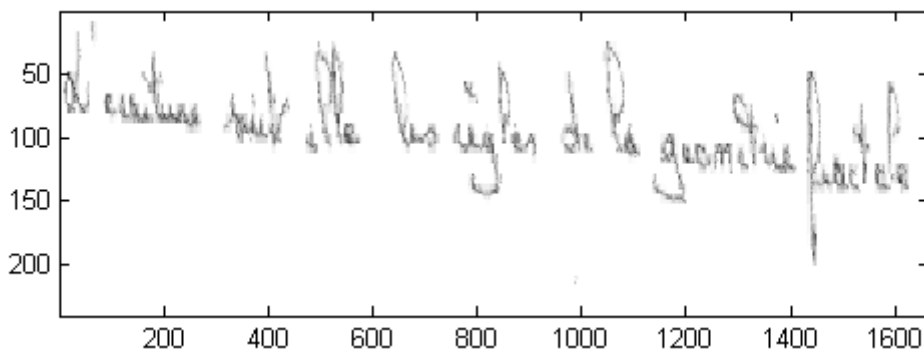
(b) utilisation de la base extraite du texte (a), $CC=0,82$



(c) utilisation de la base d'un scripteur, $CC=0,77$



(d) utilisation de la base d'un scripteur, $CC=0,78$



(e) utilisation de la base d'un scripteur, $CC=0,78$

Figure 3.19 : Résultats de la reconstruction de (a) l'image de test à partir de (b), (c), (d), (e) différentes bases extraites pendant la phase d'apprentissage

Cette approche a permis d'obtenir un pourcentage d'identification de 65%. Les résultats du calcul du coefficient de corrélation entre l'image de test d'un scripteur et les images reconstruites avec différentes bases par cette méthode sont montrés sur la Figure 3.19. Elle fut le point de départ de notre réflexion sur la bonne utilisation des éléments des bases des scripteurs pour l'identification de l'auteur d'un texte.

Nous proposons ici une approche plus performante. De même que dans le processus de compression fractale, l'image est prétraitée (redressement, normalisation, découpage en

Ranges) de manière adaptative. Nous procédons ensuite à un processus de Pattern Matching entre le texte à identifier et les éléments des bases de référence. Le texte sera attribué au scripteur pour lequel la correspondance est la meilleure. La qualité de la correspondance entre l'image initiale et une image reconstruite est calculée par Pattern Matching. Nous allons détailler le processus.

Les Ranges de l'image ont la même taille que les imagerie présentes maintenant dans les bases de référence. Alors, ce ne sont plus les similarités internes de l'image d'écriture qui sont recherchées, mais les similarités qui peuvent exister entre l'image et les éléments d'une base de référence B . Toutes les bases, chacune associée à un scripteur enrôlé, seront testées successivement. Les imagerie identiques aux Ranges sont recherchées dans la base B du scripteur que nous sommes en train de tester. Ainsi, à chaque Range R de l'image inconnue est associée une imagerie de la base B_i en minimisant le critère de l'erreur quadratique $[R_i - T(D_i)]^2$. Cela revient, ici encore, dans la recherche des similarités à considérer uniquement comme transformation possible l'homothétie de rapport $\frac{1}{2}$.

De plus, pour chaque scripteur nous ne considérons dans ce processus que les imagerie de l'ensemble d'apprentissage qui apparaissent le plus fréquemment car ce sont celles qui caractérisent le mieux le scripteur. Ainsi, seulement 80% des imagerie les plus fréquentes, ont été retenues. Par le critère de proximité, nous trouvons toujours une imagerie la plus proche d'un range de l'image, mais nous avons choisi de limiter les appariements quand la similarité entre le range et le domaine de la base n'est pas suffisante. Le seuil utilisé a été fixé à ϵ_s lors de la phase d'apprentissage en fonction du scripteur testé. Cela permet de limiter la comparaison aux éléments les plus représentatifs du scripteur et donc de la base concernée. Ainsi tous les ranges ne sont pas associés à un élément correspondant dans la base de référence d'un scripteur particulier.

Sur la Figure 3.20, le principe de l'étape d'identification de scripteur est schématisé.

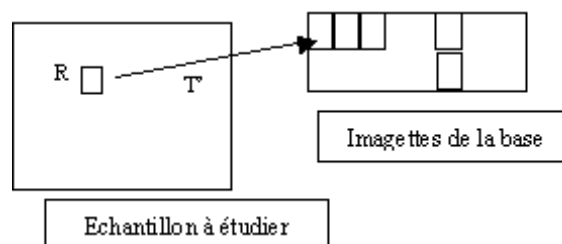


Figure 3.20 : Recherche de similarités par rapport à la base de référence.

3.4.2. Reconstruction de l'image

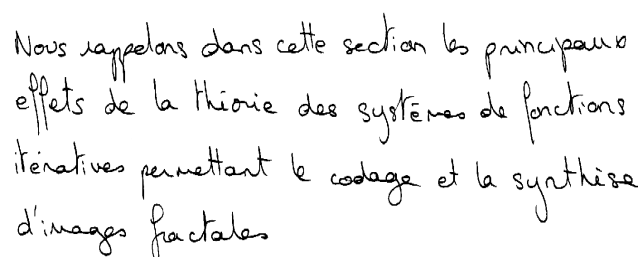
Indépendamment de la recherche de similarités entre l'échantillon d'écriture (Range) et la base de référence, nous construisons une nouvelle image, blanche à l'origine, sur laquelle nous allons recopier à l'emplacement approprié chacune des imagerie issues de la base de référence qui ont été trouvées similaires au Range initial dans le processus précédent. Si aucun domaine ressemblant de la base n'est trouvé, l'emplacement de l'imagerie dans la nouvelle image restera blanc.

Il ne reste plus maintenant qu'à comparer l'image d'écriture à classer avec la nouvelle image construite. La qualité de la ressemblance entre l'image initiale et l'image reconstruite par le processus est quantifiée comme précédemment en utilisant le PSNR et le coefficient de corrélation. Dans le cas où le style à étudier et le style de référence sont proches, ou identiques, l'image reconstruite représente de façon fidèle l'échantillon d'écriture. Dans le cas contraire, la qualité de la reconstruction est fortement dégradée.

3.5. Résultats

Nous avons actuellement travaillé avec une vingtaine de scripteurs qui ont fourni des échantillons de textes manuscrits. Des textes de grande taille pour construire la base de référence et des textes de plus petite taille pour réaliser l'identification du scripteur.

La Figure 3.21 montre un échantillon de trois textes écrits par trois scripteurs différents. Leurs bases respectives sont représentées sur la Figure 3.22.



Nous rappelons dans cette section les principaux effets de la théorie des systèmes de fonctions itératives permettant le codage et la synthèse d'images fractales

(a)

Nous rappelons dans cette section les principaux effets de la théorie des systèmes de fonctions itératives permettant le codage et la synthèse d'images fractales.

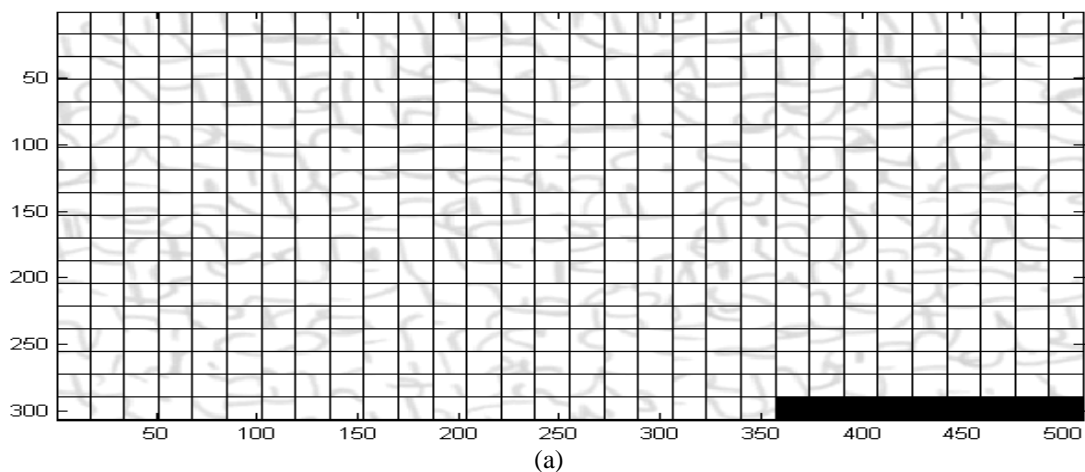
(b)

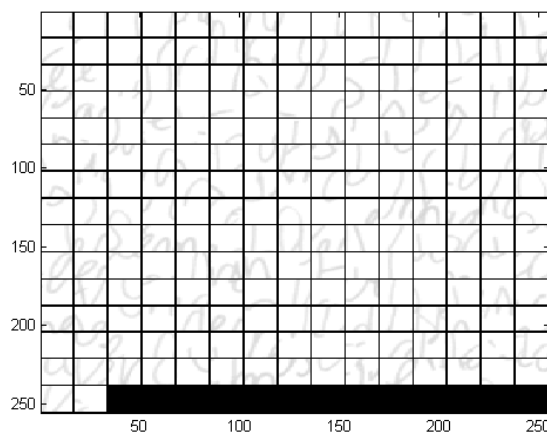
Nous rappelons dans cette section les principaux effets de la théorie des systèmes de fonctions itératives permettant le codage et la synthèse d'images fractales.

(c)

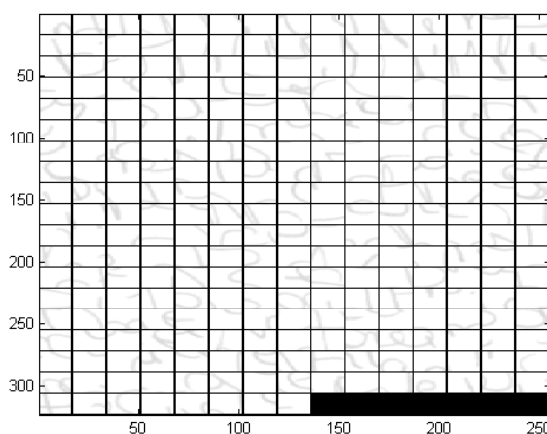
Figure 3.21: Textes d'apprentissage.

Les résultats que nous rapportons ont été obtenus en considérant des Ranges de taille 8x8 et des Domaines de taille 16x16 pixels pour la construction de la base. Dans cette étape, seulement 80% des imageries de la base sont prises en compte pour les éléments les plus représentatifs du scripteur. Ce pourcentage a été choisi car il représente un bon compromis. D'une part il assure l'élimination de certains éléments qui sont en fait trop communs à tous les scripteurs et d'autre part il permet de ne pas trop supprimer les éléments inhérents au scripteur concerné.





(b)



(c)

Figure 3.22 : Bases extraites des trois scripteurs (a), (b) et (c) respectivement

Les PSNR qui sont calculés entre les images initiales et les images décompressées sont de l'ordre de 30dB pour les images décompressées pendant l'étape de la compression fractale (cf. Tableau 3.1) et de 15dB pour celles obtenues lors de l'étape d'identification de scripteur. Pour le coefficient de corrélation, nous avons des valeurs de l'ordre de 0,7 (voire plus) pour les images décompressées et entre 0,2 et 0,5 pour les images reconstruites.

	PSNR (dB)	Coefficient de corrélation	Seuil (ϵ_s)
Scripteur (a)	39,51	0,90	520,5
Scripteur (b)	35,95	0,75	1314,6
Scripteur(c)	36,80	0,78	976,6

Tableau 3.1 : Résultats des images décompressées

Le Tableau 3.1 indique les résultats de la compression/décompression fractales classique pour les 3 scripteurs ci-dessus.

Le premier test que nous avons fait permet seulement de vérifier que les textes utilisés dans la phase d'apprentissage peuvent être identifiés comme étant écrits par leur auteur. Pour cela, nous avons appliqué l'étape d'identification sur les textes d'apprentissage. Le taux d'identification est de 100%.

De nouveaux textes ont été ensuite introduits dans le processus. Sur la Figure 3.23, sont représentés quelques échantillons de textes utilisés dans la seconde partie du test.

l'écriture suit-elle les règles de la géométrie fractale

(a)

L'écriture suit-elle les règles de la géométrie fractale.

(b)

d'écriture suit-elle les règles de la géométrie fractale.

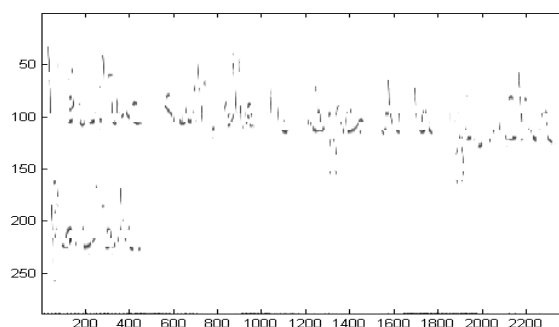
(c)

Figure 3.23 : Ecritures utilisées pour l'identification.

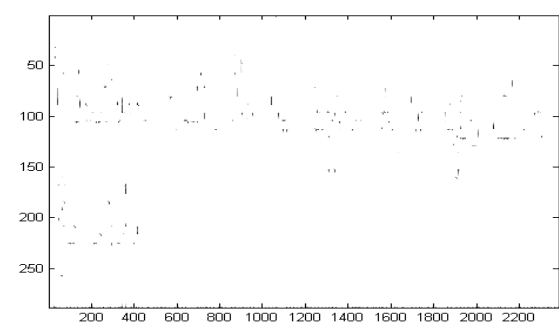
De la même manière que dans la première partie de l'étude, les textes sont normalisés. En utilisant les différentes bases de référence, les images reconstruites à l'aide des similarités avec les bases de référence sont calculées. Notre processus d'identification est appliqué avec un seuil fixé à ε_s . Le PSNR associé avec chaque image reconstruite est calculé et la meilleure base de référence fournit le nom du scripteur supposé avoir écrit le texte. Nous avons obtenu un taux d'identification supérieur à 85%.

Nous illustrons, ici, le processus d'identification de scripteur. Pour cela considérons les trois scripteurs concernés par les trois textes (a), (b) et (c) de la Figure 3.23. Les textes d'apprentissage sont présentés Figure 3.21 dont les trois bases de référence ont pu être extraites, ainsi que les seuils associés à chacun des scripteurs. Nous avons respectivement comme valeur de ε_s : 520,5 ; 1314,6 ; 976,6. Ces valeurs peuvent varier du simple au double

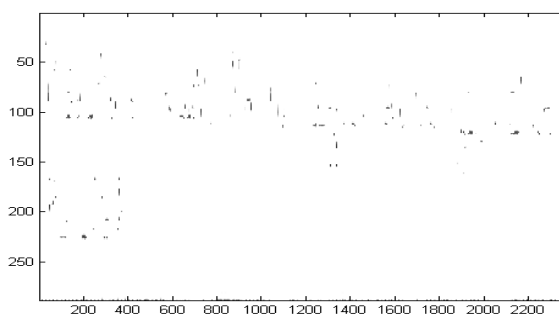
suivant le style de l'écriture. Nous présentons, sur la Figure 3.24, les images recomposées associées au texte de la Figure 3.23(a) en considérant respectivement les bases de référence des scripteurs a , b et c .



PSNR=12,89dB et CC=0,52(a)



PSNR=12,31dB et CC=0,25(b)



PSNR=12,34dB et CC=0,27(c)

Figure 3.24 : Images résultant du processus d'identification du scripteur texte 3.23(a).

Pour un échantillon de scripteur donné, nous avons reconstruit les images à l'aide de 3 différentes bases. Nous indiquons sur le Tableau 3.2 et sur le Tableau 3.3, les valeurs des PSNR et des coefficients de corrélation correspondants à la reconstruction de l'image selon ces différentes bases. Ainsi les tableaux se lisent par colonne.

Echantillon	Scripteur (a)	Scripteur (b)	Scripteur (c)
Bases			
Scripteur (a)	12,89dB	15,38dB	15,54dB
Scripteur (b)	12,31dB	15,91dB	16,05dB
Scripteur (c)	12,34dB	15,69dB	16,36dB

Tableau 3.2 : Résultats du processus d'identification de scripteur avec les valeurs des PSNR.

Echantillon	Scripteur (a)	Scripteur (b)	Scripteur (c)
Bases			
Scripteur (a)	0,52	0,13	0,24
Scripteur (b)	0,25	0,49	0,51
Scripteur (c)	0,27	0,39	0,59

Tableau 3.3 : Résultats du processus d'identification de scripteur avec les valeurs des coefficients de corrélation.

Le test effectué sur la vingtaine de scripteurs donne un taux d'identification de 85%. Dans les exemples montrés ci-dessus nous avons pris 3 types différents de réussite d'identification. Le scripteur (a) est celui qui possède le plus d'imagettes dans la base (531 imagettes), ce qui correspond à une grande variabilité dans l'écriture. Lors de l'étape d'identification, il est largement identifié. En effet, nous pouvons voir sur les tableaux que ses critères de qualité sont largement supérieurs à ceux des deux autres scripteurs. Quant au scripteur (b), il n'a qu'une légère avance par rapport au scripteur (c) et le scripteur (c) est tout juste identifié. D'ailleurs ces deux derniers scripteurs possèdent des PSNR et des coefficients de corrélation assez proches alors que leur style d'écriture est très différent.

3.6. Conclusion

Nous avons présenté dans ce chapitre une méthode originale d'identification de scripteur. Nous avons considéré la représentation de l'écriture dans un espace de dimension finie et pour chaque scripteur des projections dans des espaces différents. L'image est traitée dans sa globalité en prenant en compte des caractéristiques internes de chaque écriture, sans aucune extraction ni comparaison de détails. Pour ce faire, nous avons choisi la compression fractale pour extraire les éléments invariants d'un scripteur pour l'étape d'apprentissage. Ces éléments

sont alors utilisés dans un processus de Pattern Matching pour l'étape d'identification. Les résultats montrent que le comportement fractal de l'écriture permet le développement d'approches innovantes.

En effet, utiliser à nouveau la compression fractale en remplaçant les Domaines par les éléments de la base nous a semblé au premier abord la meilleure solution. Mais nous n'avons pu dépasser un taux d'identification de 65% à cause de l'étape de remplacement. Le remplacement se fait par un calcul d'erreur quadratique moyenne ce qui occasionne des erreurs de remplacement et d'amalgame entre le texte à identifier et les éléments de la base.

Nous obtenons un taux d'identification de 85% grâce à une nouvelle méthode. Il reste, à présent, à mettre en œuvre le module de reconnaissance d'écriture. Nous nous inspirerons d'une méthode structurelle avec adaptation au scripteur mise en œuvre par A. Nosary [NOS02] et dont une brève description est faite dans le chapitre suivant. Notre but est d'utiliser les bases de référence extraites pendant la phase d'apprentissage et ainsi de simplifier cette étape pour la rendre plus performante. Notre méthode s'appuiera plus sur une accumulation d'hypothèses. De plus, nous nous affranchissons du contenu et de la segmentation des textes en entités inférieures à celle des mots.

L'avantage de cette méthode est qu'elle peut s'appliquer à d'autres domaines que celui des textes occidentaux mais aussi à d'autres domaines tels que l'extraction de caractéristiques d'une image quelconque pour l'indexation de documents.

Chapitre 4

Autres applications

4. Autres Applications

4.1. Introduction

Nous venons de voir dans le chapitre précédent comment à partir de la théorie de la compression fractale, nous arrivons à extraire des caractéristiques intrinsèques des scripteurs. Ces caractéristiques représentent dans ce cas le style de chaque scripteur. Ensuite, nous avons montré qu'il était possible d'identifier un scripteur par ces caractéristiques. La suite logique de cette méthode est d'adapter un module de reconnaissance d'écriture utilisant l'information du style issue de l'étape précédente. Néanmoins son intérêt ne s'arrête pas là. Il est possible d'utiliser cette technique pour résoudre de nombreux autres problèmes, dans des applications telles que l'identification de signatures, la différenciation entre des écritures utilisant des alphabets différents et si nous allons plus loin dans des applications telles que l'identification d'images quelconques.

Ainsi dans ce chapitre nous décrivons les différentes applications susceptibles d'utiliser notre méthode. Dans un premier temps, nous allons décrire notre processus de reconnaissance d'écriture, puis nous présenterons comment l'adapter pour l'identification de signatures. Nous étendrons enfin notre raisonnement à la caractérisation d'images quelconques.

Pour chaque application nous exposons les résultats obtenus à ce jour.

4.2. Reconnaissance d'écriture

4.2.1. Rappels bibliographiques

Nous allons faire un bref récapitulatif des méthodes de reconnaissance d'écriture. Tout d'abord il faut savoir qu'il existe deux façons d'acquérir l'écriture, soit en temps réel au moyen d'une tablette numérique, soit de façon globale sur une page au moyen d'un scanner. Dans le premier cas, on parle de reconnaissance *en-ligne* et dans le second de reconnaissance *hors-ligne*.

En ce qui concerne la reconnaissance en ligne [GAR96], celle-ci a connu de gros succès avec l'avènement des « *Organiseur* » et ordinateurs de poche utilisant des écrans tactiles. Pour la reconnaissance de l'écriture manuscrite hors-ligne, de gros progrès ont également été faits ces dernières années. Il en ressort des systèmes à présent bien maîtrisés et utilisés dans beaucoup de secteurs tels que la reconnaissance de chèques bancaires [GUI95][AUG01] ou la reconnaissance d'adresses postales.

Quelle que soit la méthode de reconnaissance d'écriture manuscrite utilisée, une structure commune est appliquée [STE99]. Elle se divise en 4 étapes : les prétraitements, l'extraction de caractéristiques, la reconnaissance et les post-traitements (cf. Figure 4.1).

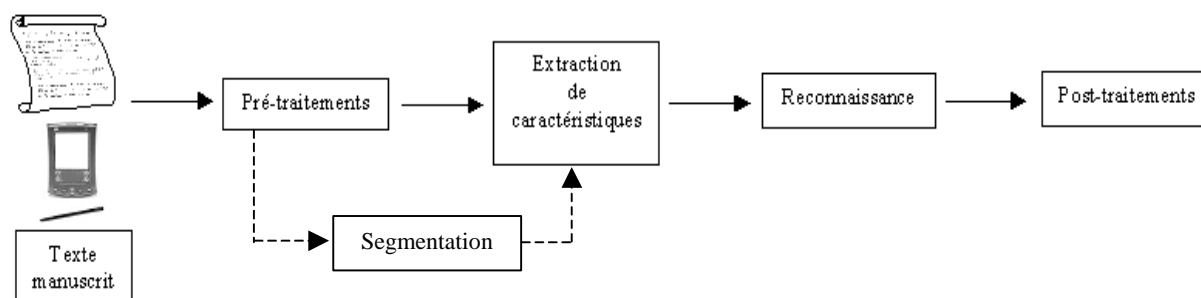


Figure 4.1 : Système complet de reconnaissance d'écriture

Nous allons maintenant détailler chacune de ces étapes en nous attachant plus particulièrement au cas de la reconnaissance hors-ligne d'écriture manuscrite cursive.

L'étape de prétraitements comprend en général des phases de :

- redressement de l'écriture par rotation
- correction de la pente des barres verticales par transformées
- réduction du bruit
- normalisation de taille de l'écriture

L'étape d'extraction de caractéristiques [TAO01][TRI96] permet de représenter les images sous forme d'entités élémentaires :

- les segments de traits
- les graphèmes
- les pseudo-lettres

L'étape de segmentation est une aide à l'extraction de caractéristiques. Elle peut être de deux types : implicite ou explicite. La difficulté de la segmentation explicite est très bien résumée dans le paradoxe énoncé par Sayre en 1973 [SAY73] : « Il est nécessaire de segmenter correctement pour reconnaître, mais il est aussi nécessaire de reconnaître pour segmenter ». On trouvera dans l'article de R. Casey et E. Lecolinet [CAS96] une description détaillée des méthodes de segmentation existantes.

Il existe deux approches possibles pour l'étape de reconnaissance [LEC94][VIN02], l'approche globale (ou méthode descendante) et l'approche analytique (ou méthode ascendante). La première considère le mot dans sa totalité alors que la seconde utilise des représentations élémentaires.

4.2.1.1.Approche analytique

Les méthodes analytiques se caractérisent par l'utilisation de représentations élémentaires d'une part et de segmentation d'autre part. Elles utilisent 3 niveaux de représentation : niveau primitives, niveau mots et niveau sous-mots. Lorsque la segmentation est explicite, 3 étapes sont nécessaires. La première consiste à segmenter l'écriture en petites unités, la deuxième est la reconnaissance individuelle de ces unités et enfin la troisième étape est un post-traitement utilisant une connaissance lexicale. Lorsque la segmentation est implicite, une fenêtre mobile de taille variable est utilisée. Cela permet de générer un treillis de toutes les combinaisons de primitives possibles. Ces méthodes peuvent être basées sur la programmation dynamique par une mesure d'édition par exemple ou sur des méthodes probabilistes. On peut aussi avoir recours à une phase d'apprentissage qui réalise simultanément la segmentation des mots en lettres et leur reconnaissance [AUG01].

- Méthodes probabilistes

La méthode probabiliste la plus courante utilise les Modèles de Markov Cachés (MMC) [RAB88]. Le principe consiste à trouver la structure cachée qui est une représentation

canonique des formes de référence. On appelle structure cachée, une structure *a priori* inconnue se trouvant entre deux structures connues. Une évaluation du meilleur chemin est ensuite faite soit par l'algorithme de Viterbi [RYA93], soit par la programmation dynamique.

L'adaptation des MMC au problème de la reconnaissance d'écriture est décrite par M. Gilloux dans [GIL93]. Deux cas sont pris en considération : une reconnaissance avec un vocabulaire constant de petite taille et une reconnaissance avec un vocabulaire dynamique. Dans le premier cas, pour chaque mot un MMC est utilisé. Le modèle correspondant a été défini avec un nombre d'états dépendant de l'exemple de longueur le plus probable pour chaque catégorie de mot. L'auteur se sert de la connaissance contextuelle qu'il a sur les mots pour être capable d'évaluer la fréquence de chaque montants et de la combiner avec le taux de confiance. Dans le cas d'un vocabulaire dynamique, les MMC sont utilisée au niveau lettre.

- Programmation dynamique et mesure d'édition

La programmation dynamique est une technique générale utilisée pour résoudre les problèmes d'optimisation. Elle comprend la mesure d'édition et l'algorithme de Viterbi. La mesure d'édition calcule une distance entre deux chaînes de symboles. Elle a été introduite par Wagner et Fischer [WAG74] en reconnaissance de la parole. Etant donné un ensemble de transformées élémentaires entre les symboles de deux chaînes, cet algorithme fournit la meilleure séquence des transformées élémentaires nécessaires pour transformer la première chaîne en la seconde. Cette technique est surtout utilisée dans la prise de décision finale dans la reconnaissance d'écriture [MOR91][PAQ91][LER91][MAD93][CAM92].

4.2.1.2.Approche globale

Cette approche traite le mot dans sa totalité et nécessite alors un lexique spécifique de mots. Ainsi pour étendre le lexique une phase d'apprentissage est obligatoire.

Les premières études menées dans les années soixante découpaient le mot selon différentes zones (barres directionnelles, zone horizontale, ..).

C. Parisse [PAR96] utilise les parties hautes et basses du mot. Un profil est constitué d'une série de vecteurs décrivant les contours. Ensuite, il utilise la mesure d'édition pour comparer les différents profils.

T. Paquet et al. compare trois méthodes globales [PAQ99]. Ces trois méthodes sont basées sur les Modèles de Markov Cachés pour la reconnaissance. Les écritures utilisées pour tester ces trois méthodes sont segmentées de la même manière, tout comme l'extraction de caractéristiques. La méthode employée est celle de la description de barres qui consiste à analyser les points connexes de l'écriture squelettisée. Ensuite une séquence de graphèmes est construite pour chaque mot. La première méthode de reconnaissance utilise un modèle composé d'une matrice des probabilités d'état transitoire, d'une matrice des probabilités d'observation, d'un vecteur des probabilités d'état initial et d'un vecteur de probabilités d'état final. L'apprentissage est effectué à l'aide de l'algorithme Baum-Welch et la reconnaissance à l'aide d'une procédure récursive qui calcule la probabilité de la séquence d'observation des graphèmes pour chaque modèle donné. La deuxième méthode utilise la concaténation explicite des modèles de lettres. L'apprentissage prend en compte la fréquence des lettres dans un mot et la reconnaissance est la même que celle décrite pour la première méthode. La troisième méthode donne pour chaque graphème son interprétation correspondante dans un mot. Les taux de reconnaissance obtenus, environ 90% sous conditions, montrent que chaque méthode est plus adaptée à certains types de mots. La première est meilleure pour les mots longs, la deuxième donne de meilleurs résultats pour les mots fréquents dans la base d'apprentissage et la troisième est plus performante dans le cas de l'utilisation de mots ayant des lettres fréquemment employées.

4.2.1.3. Autres Approches

E. Augustin [AUG01] a développé un processus de reconnaissance de mots par un hybride Réseaux de Neurones (RN) et Modèles de Markov Cachés (MMC). L'intérêt des hybrides est d'allier la capacité des MMC à modéliser les séquences d'événements et le pouvoir des RN à discriminer les formes des graphèmes. Son travail a porté plus particulièrement sur la modélisation (sélection des états) et la topologie des MMC et les méthodes d'apprentissage globales du système hybride. Dans son système de reconnaissance de mots, l'écriture est représentée par une séquence gauche-droite à une dimension de graphèmes. Les graphèmes sont extraits par une segmentation de l'écriture. Le principe des systèmes hybrides RN et MMC est présenté avec son apprentissage itératif selon l'algorithme Expectation Maximization (EM). Ce système permet de remplacer la quantification vectorielle des MMC discrets, classification qui perd beaucoup d'information et qui est issue d'une classification

non supervisée, par un RN dont l'annotation d'apprentissage est fournie par les MMC. Cette annotation automatique leur permet d'une part, d'éviter une annotation manuelle des lettres de mots cursifs pour l'apprentissage du RN, et d'autre part, d'adapter l'évaluation des observations par le RN aux MMC. De plus, une annotation qui fournit la distribution des probabilités a posteriori des classes apporte plus d'information et améliore les performances du système complet.

Une évaluation a été faite sur les systèmes de reconnaissance d'écriture manuscrite utilisant les réseaux de neurones et/ou les Modèles de Markov Cachés [OLI02]. Les réseaux de neurones utilisés sont de type MLP avec deux niveaux d'extractions de caractéristiques : un niveau perceptuel (NNP) et un niveau directionnel (NND). Les MMC sont de topologie Bakis (discrète gauche-droite) avec un apprentissage utilisant l'algorithme de Baum-Welch et l'algorithme Forward est utilisé pour la validation. De cette évaluation, il ressort qu'un système utilisant un MMC combiné avec un NNP et un NND obtient 90% de taux de reconnaissance des mots.

M. Côté et al. présentent dans [COT96] et [COT97] une nouvelle méthode de lecture automatique de l'écriture cursive utilisant des concepts perceptifs proposés par les psychologues McClelland & Rumelhart. L'étude proposée se limite à la reconnaissance hors-ligne de mots cursifs isolés. A partir d'observations et d'hypothèses utilisées dans l'élaboration de modèles de lectures, les auteurs décrivent l'organisation de leur système connexionniste et montrent le rôle du mécanisme d'activation dans la perception et la lecture d'écriture cursive. Ils ont introduit quelques caractéristiques spécifiques à l'écriture cursive telles que les ascendants, les descendants et les boucles. Leur système est basé sur 3 niveaux de détecteurs réalisées à partir de réseau de neurones : un niveau caractéristique, un niveau lettre, et un niveau mot. Premièrement, ils utilisent les caractéristiques. Deuxièmement, ils traitent les ambiguïtés relatives à la position des lettres en introduisant une position floue. La position et les lettres manquantes sont déduites à l'aide d'un lexique. Leur modèle est capable de fonctionner avec des mots de différentes longueurs. La base de données utilisée a été construite à partir de montants de chèques bancaires. Elle est composée de 184 images pour l'étape d'apprentissage et de 2929 images pour les tests. Un taux de reconnaissance des mots cursifs isolés de 80% est obtenu.

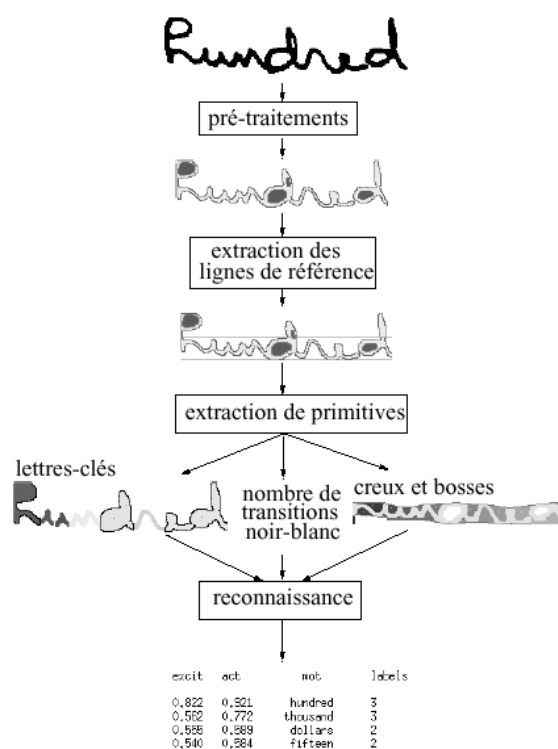


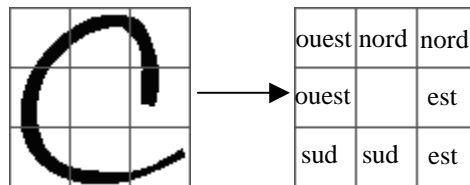
Figure 4.2 : Système complet de reconnaissance d'écriture extrait de [COT97]

4.2.2. Notre méthode de reconnaissance

Notre méthode n'a pas besoin d'étape de segmentation, ni de correction de la pente selon la direction verticale des barres. Elle est basée sur une accumulation d'hypothèses effectuées sur l'image reconstruite après l'étape d'identification du scripteur.

4.2.2.1. Accumulation d'hypothèses

Après avoir identifié le scripteur par la méthode décrite dans le chapitre 3, nous nous sommes efforcés d'utiliser au maximum le résultat des traitements précédents pour reconnaître l'écriture. Tout d'abord, il faut revenir sur l'étape d'apprentissage et y adjoindre un étiquetage des éléments, imquettes, constituant chaque lettre de l'alphabet. Nous avons opté pour un étiquetage assez simple dans lequel chaque imquette de la base est associée à une lettre de l'alphabet sa position cardinale au sein de cette lettre : *nord*, *sud*, *est* et *ouest*. Dans le cas de lettres avec hampe et jambage les positions cardinales possibles sont : *centre*, *hampe* et *jambage*. Chaque imquette de la base se voit associée à une étiquette de deux caractères du type « *lettre direction* ». Exemple : « *en* » (i.e. lettre *e* et orientation *nord*) ou « *fj* » (i.e. lettre *f* et orientation *jambage*). Un exemple des positions de la lettre *c* est montré sur la Figure 4.3.

Figure 4.3 : Etiquetage des positions de la lettre *c*

Ensuite, nous réévaluons ces étiquettes pour prendre en compte les similarités existant entre plusieurs imagettes. Par exemple la partie haute de la lettre *a* peut ressembler fortement à la partie haute de la lettre *e* ; et la boucle de la lettre *f* peut ressembler fortement à la boucle de la lettre *l* selon le style du scripteur. Nous profitons de ces ambiguïtés pour améliorer la robustesse de la reconnaissance en procédant à des étiquetages multiples : une imagette peut représenter plusieurs lettres. Pour ce faire, nous calculons l'erreur quadratique moyenne entre les imagettes prises deux à deux sur l'ensemble de la base. Tous les couples dont l'erreur est inférieure au seuil ε_s du scripteur sont considérés comme similaires et on associe à chacune des deux imagettes l'union des étiquettes correspondantes. Une fois le processus terminé, la base est constituée d'imagettes pouvant avoir plusieurs dizaines d'étiquettes. Cela est dû bien évidemment à la nature même de leur représentation : Un arc commun à beaucoup de lettres (*a, c, e, g, m, n, ...*) engendre beaucoup d'étiquettes ; tandis que la barre horizontale de la lettre *t* ne représente que très peu d'étiquettes.

Nous procédons aussi à une étape statistique. Pour cela, nous calculons les nombres maximum et minimum d'imagettes constituant chaque lettre du texte d'apprentissage, puis nous faisons de même pour les positions. Le Tableau 4.1 illustre les informations statistiques relatives à un scripteur.

Une fois l'étiquetage terminé, nous procédons avec la même méthode que celle utilisée pour l'identification mais en utilisant cette fois-ci toutes les imagettes de la base du scripteur identifié. L'image reconstruite, comme indiqué dans le chapitre 3, par recopie à l'emplacement approprié des imagettes de la base de référence, est alors de très bonne qualité. Etant donné que plusieurs étiquettes sont associées à chaque imagette, il suffit donc de rechercher les lieux où se rassemblent des amas de labels d'une même lettre dans une région proche. La cohérence de ces amas peut être vérifiée par rapport aux valeurs contenues dans le tableau statistique du scripteur dont nous avons montré un exemple dans le Tableau 4.1. La détermination des régions candidates pour une lettre est obtenue par un balayage de l'image avec une fenêtre rectangulaire dont la taille est égale à celle indiquée dans la deuxième colonne du Tableau 4.1. Si les conditions du tableau sont réunies, alors nous considérons que

la lettre est présente et nous supprimons les étiquettes associées à la lettre pour toutes les imageries de la base de référence.

Lettre	Taille (long., larg.)	Nord (min., max.)	Sud (min., max.)	Est (min., max.)	Ouest (min., max.)	Etiquettes totales
a	23, 55	1, 3	1, 5	2, 6	1, 4	9, 14
c	23, 39	1, 2	2, 3	2, 3	1, 2	6, 10
r	31, 47	1, 3	0, 0	2, 5	0, 4	5, 9
e	31, 55	1, 4	1, 4	1, 7	1, 3	5, 15
⋮	⋮	⋮	⋮	⋮	⋮	⋮

(a) lettres contenues dans le corps du texte

Lettre	Taille (long., larg.)	Centre (min., max.)	Hampe (min., max.)	Jambage (min., max.)	Etiquettes totales
f	111, 39	1, 5	11, 18	7, 16	23, 38
t	71, 55	7, 16	4, 10	0, 0	13, 26
l	55, 39	1, 8	4, 6	0, 0	7, 13
⋮	⋮	⋮	⋮	⋮	⋮

(b) lettres avec hampe et jambage

Tableau 4.1 : Tableau regroupant les informations statistiques des lettres pour un scripteur donné

La Figure 4.4 présente un récapitulatif de toute la méthode mise au point, de l'étape d'apprentissage à l'étape de reconnaissance d'écriture.

4.2.2.2. Résultats

Dans cette partie, nous exposons les résultats obtenus sur les premiers tests effectués en utilisant les procédés présentés. Pour l'instant, notre méthode de reconnaissance n'a été appliquée que sur un seul scripteur, identifié avec la méthode d'identification décrite dans le chapitre précédent.

- Apprentissage

Pour un premier test, les imageries de la base ont été étiquetées une par une manuellement (cf. Figure 4.5). Il va de soi que ce travail fastidieux devra être automatisé en utilisant un texte d'apprentissage identique pour tous les scripteurs.

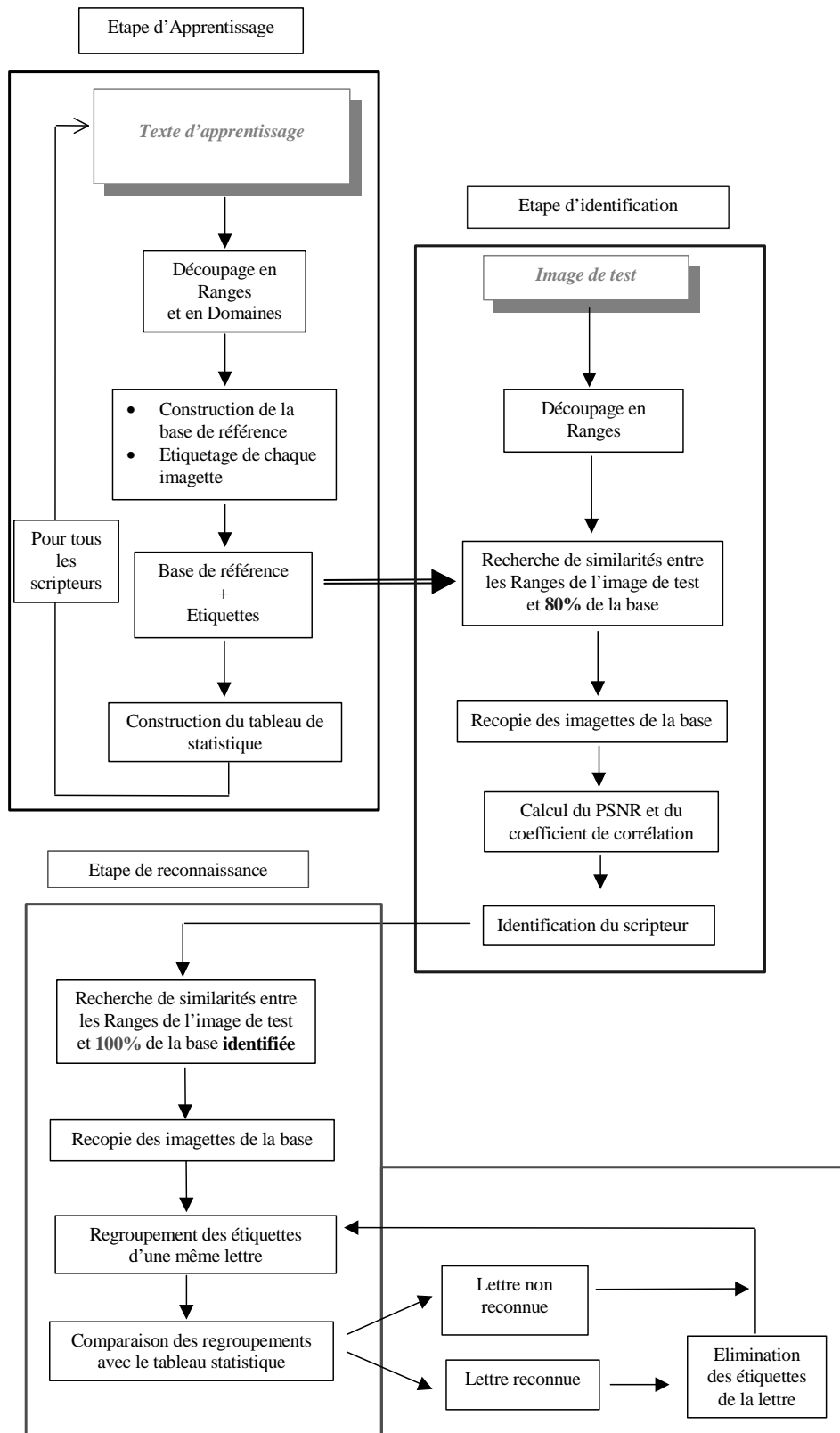


Figure 4.4 : Récapitulatif de toute la méthode

Sur la Figure 4.5, nous voyons que dans la pratique ce sont les Ranges du texte d'apprentissage qui sont étiquetés. Cela est possible grâce à la technique de la compression

fractale. En effet, nous recherchons pendant l'étape d'apprentissage les similarités existant entre un Range et un Domaine. Etiqueter tous les Ranges revient donc à étiqueter aussi les Domaines qui composent la base de référence du scripteur.

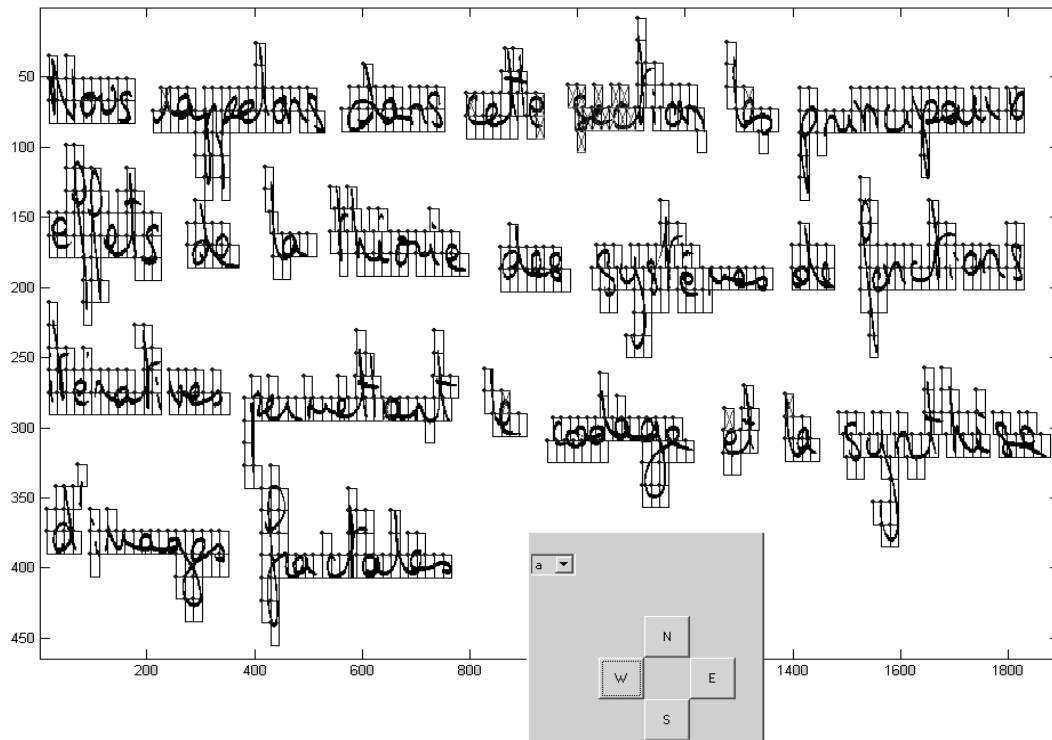
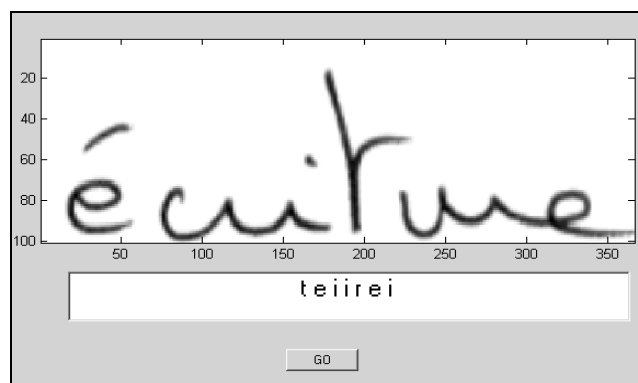


Figure 4.5 : Etiquetage des lettres après découpage, avec signalement des lettres déjà faites

- Reconnaissance

Nous reprenons l'image de test (cf. Figure 3.23(a)) qui a permis l'identification et nous appliquons à nouveau la méthode avec de nouveaux seuils comme cela a été indiqué sur la Figure 4.4. Les résultats obtenus sont montrés sur la Figure 4.6 . Pour chaque mot extrait, les lettres reconnues sont affichées dans l'ordre de lecture (de gauche à droite).



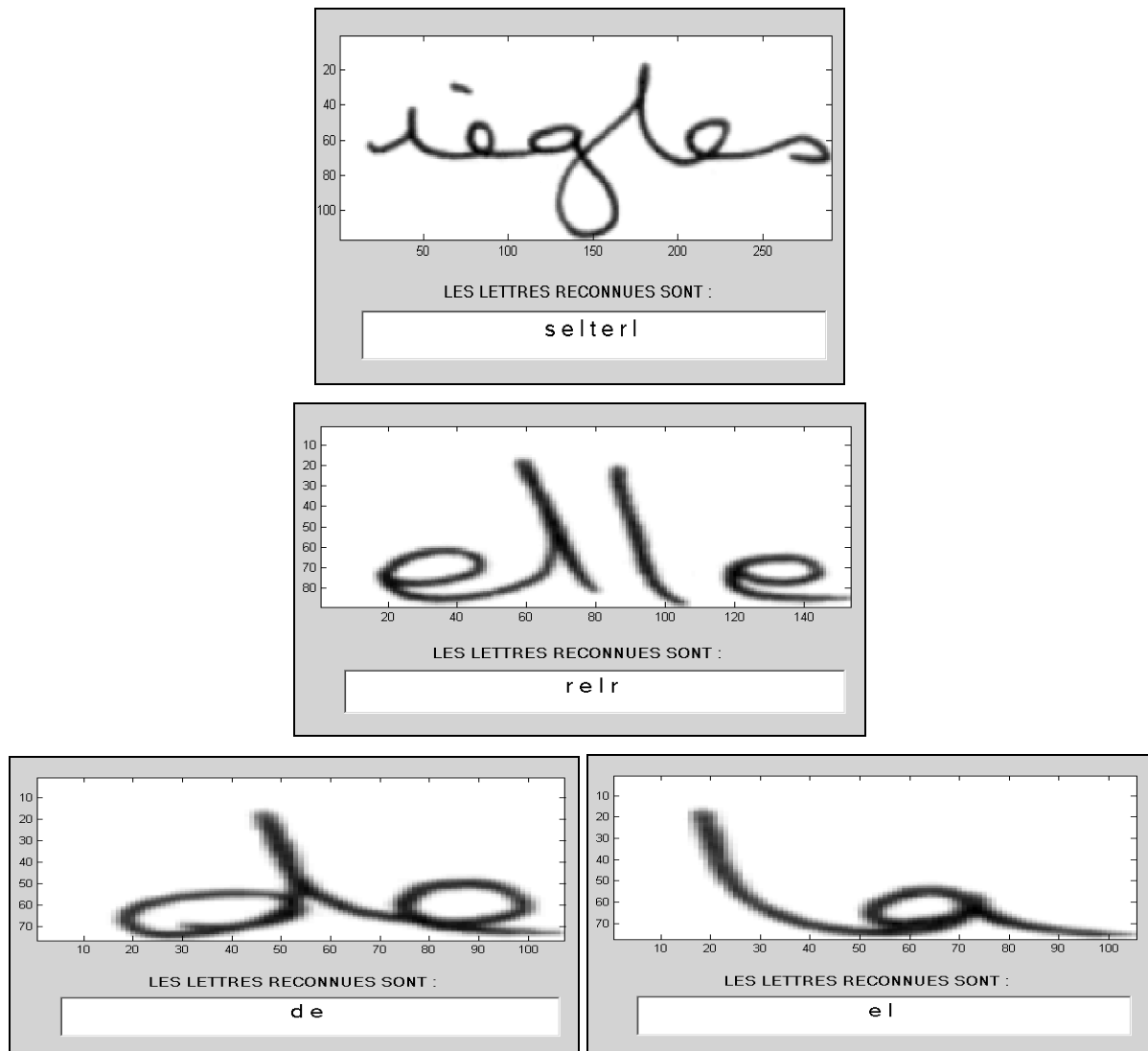


Figure 4.6 : Résultat de la reconnaissance

Ces premiers résultats montrent que notre méthode obtient des meilleurs résultats sur des mots courts. Pour les mots plus longs, les lettres reconnues correctement sont souvent mal localisées. Ceci est dû à la manière dont sont traitées les lignes et les colonnes de l'image représentant les amas d'étiquettes.

4.2.3. Conclusion

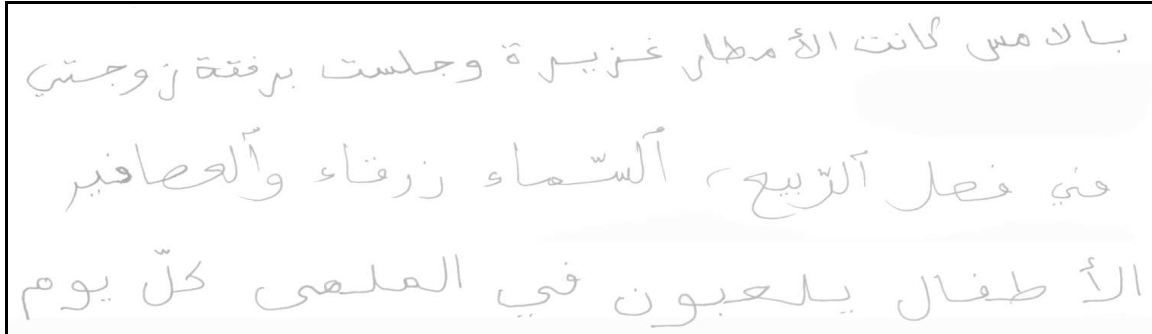
Les premiers tests sont prometteurs, même si une amélioration est encore nécessaire. Notre méthode peut constituer une piste intéressante pour la reconnaissance de l'écriture manuscrite. Bien qu'ils ne débouchent pas sur des résultats directement exploitables, les essais présentés ci-dessus mettent en évidence la faisabilité de cette méthodologie. Bâtie sur une accumulation

d'hypothèses, elle présente l'avantage de ne nécessiter aucune programmation évoluée telle que les réseaux de neurones ou les Modèles de Markov Cachés. D'autre part, elle réutilise une partie des traitements effectués lors de la phase d'identification pour reconnaître toutes les lettres.

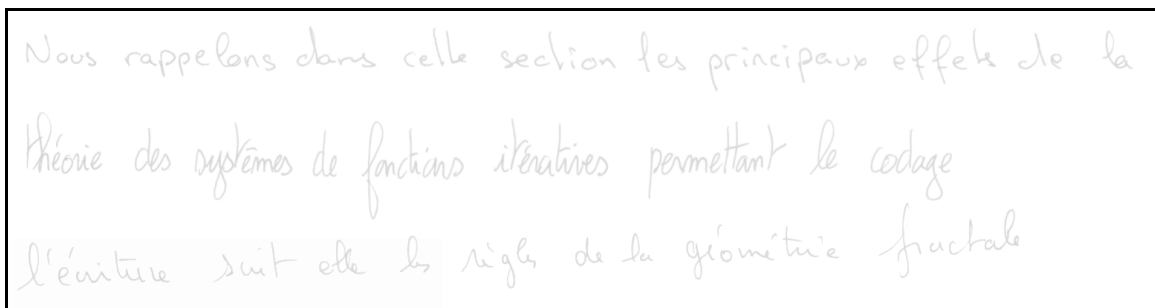
4.3. Différenciation de deux alphabets

Dans le chapitre précédent, nous avons montré qu'il était possible d'associer les caractéristiques extraites pendant la phase d'apprentissage à une lettre d'un texte manuscrit. A présent, nous voulons savoir s'il est possible de différencier deux alphabets complètement différents par notre méthode d'identification de scripteur. En effet, nous partons du principe que des caractéristiques extraites sont assez différentes d'un alphabet à l'autre.

La méthode consiste à prendre deux textes d'apprentissage écrits dans deux langues différentes, comme par exemple un texte écrit en français et un autre écrit en arabe. Ces deux textes contiennent plusieurs lignes écrites par des scripteurs différents. Les deux textes d'apprentissage sont montrés sur la Figure 4.7.



(a) texte d'apprentissage en langue arabe écrit par 3 scripteurs



(b) texte d'apprentissage en langue française écrit par 3 scripteurs

Figure 4.7 : Textes d'apprentissage de deux alphabets différents

Après l'étape d'apprentissage, nous obtenons deux bases (cf. Figure 4.8) à partir de ces deux textes.

Comme précédemment, nous appliquons notre méthode d'identification de scripteurs sur deux textes de tests (cf. Figure 4.9), différents de ceux d'apprentissage. Nous avons opté pour une utilisation complète de toutes les imasettes. Nous calculons le coefficient de corrélation entre l'image initiale et l'image reconstruite selon notre méthode. Nous obtenons alors le Tableau 4.2.

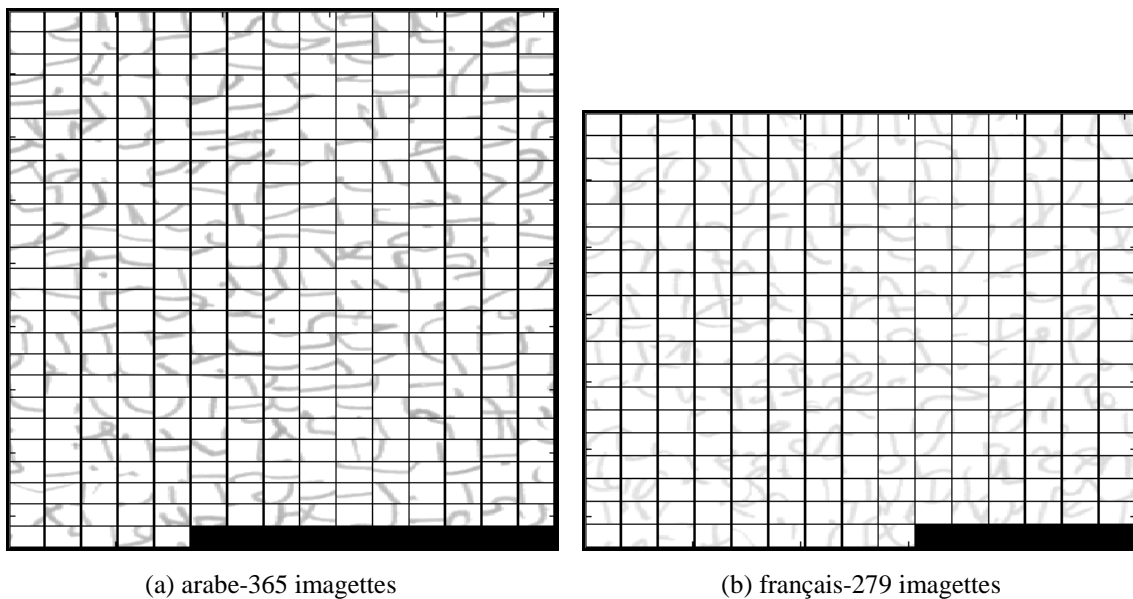


Figure 4.8 : Bases extraites des deux textes d'apprentissage

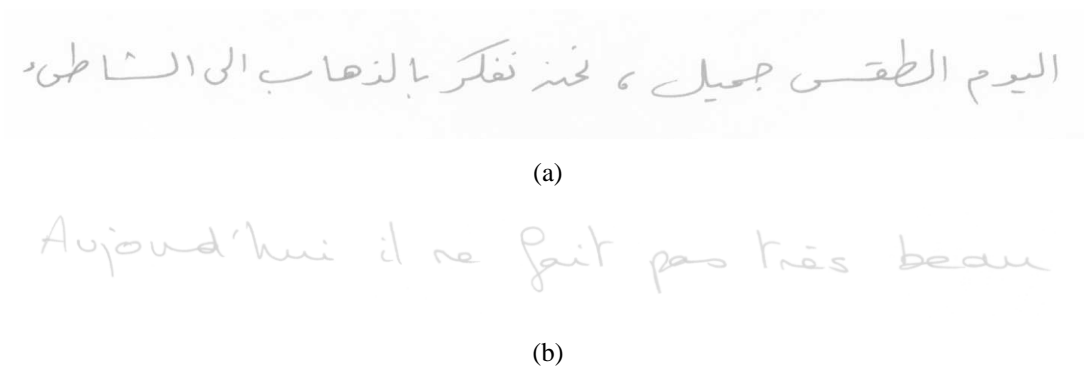


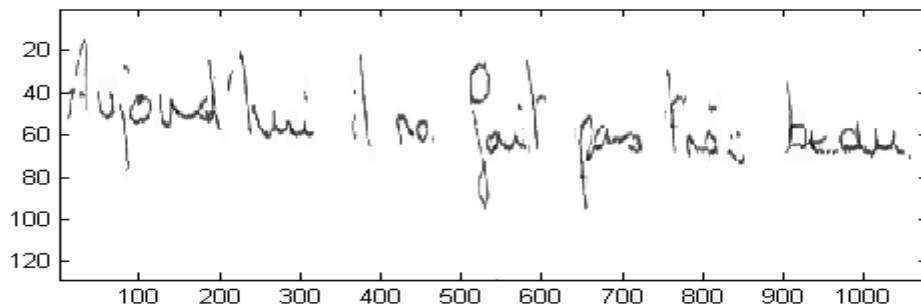
Figure 4.9 : Deux textes de tests

Les résultats obtenus par notre méthode d'identification appliquée à deux alphabets différents sont répertoriés dans le Tableau 4.2. Dans ce tableau, nous avons calculés le coefficient de corrélation entre l'image de test reconstruite à partir d'une base extraite du texte d'apprentissage.

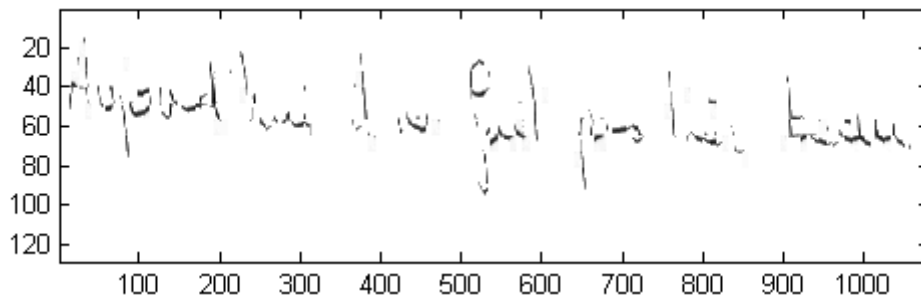
Images de tests Base	Texte de test écrit en arabe	Texte de test écrit en français
Langue arabe	0,94	0,67
Langue française	0,89	0,87

Tableau 4.2 : Résultats des calculs du coefficient de corrélation sur la différenciation d'alphabets

Au vu des résultats obtenus, nous arrivons donc à différencier deux alphabets grâce à notre méthode. L'écart des résultats n'est certes pas énorme mais suffisamment significatif pour les distinguer de manière automatique. Nous montrons sur la Figure 4.10 et la Figure 4.11, les 4 images reconstruites pour une appréciation visuelle.

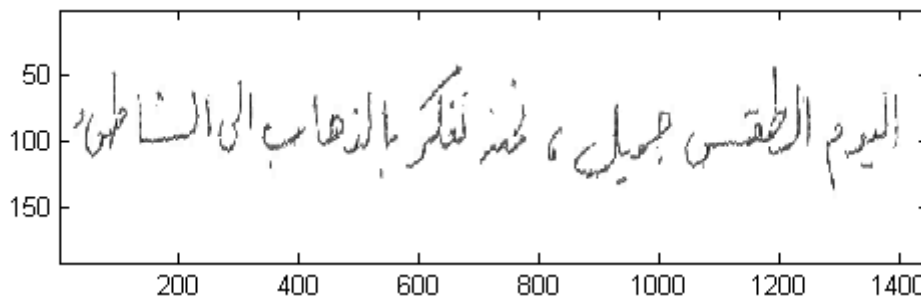


(a)



(b)

Figure 4.10 : Image de test en français reconstruite avec (a) la base française et (b) la base arabe



(a)

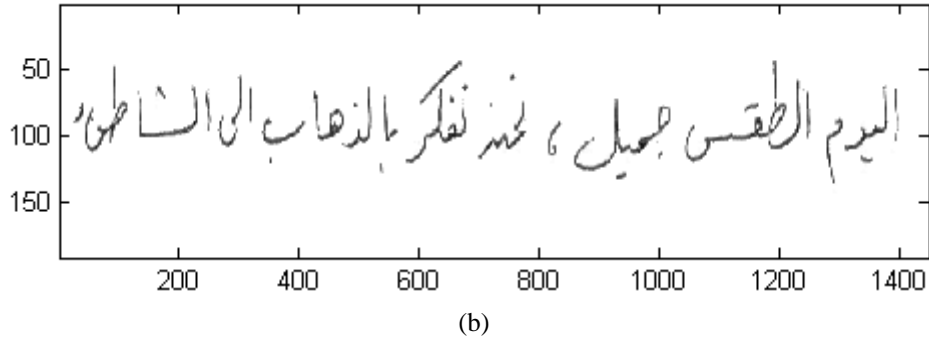
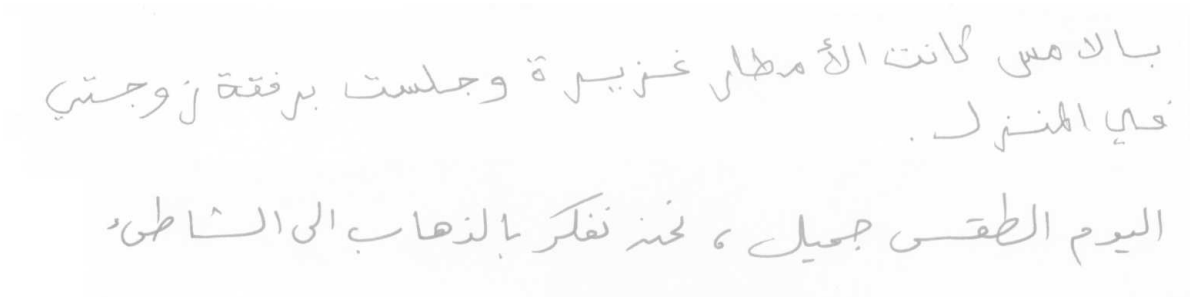


Figure 4.11 : Image de test en arabe reconstruite avec (a) la base française et (b) la base arabe

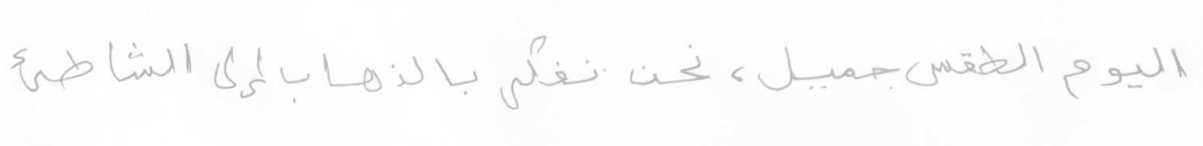
Un test a été fait dans le cas d'un personne bilingue. Nous avons inclus un échantillon d'écriture en langue arabe et en langue française dans les textes d'apprentissage d'une même personne.



(a) texte d'apprentissage en langue arabe écrit par 2 scribeurs

Nous rappelons dans cette section les principaux effets de la théorie des systèmes de fonctions itératives permettant le codage d'images fractales.

(b) texte d'apprentissage en langue française écrit par 3 scribeurs



(c) image de test écrit en arabe

Aujourd'hui il me fait pas très beau

(d) image de test écrit en français

Figure 4.12 : Texte d'apprentissage et de test écrit par une personne bilingue

La première ligne du texte de la Figure 4.12(a), la dernière ligne de la Figure 4.12(b), le texte de la Figure 4.12(c) et (d) ont été écrits par la même personne. Cela nous permet de savoir s'il y a des implications croisées. Les résultats obtenus en appliquant la même méthode que précédemment sont montrés sur le Tableau 4.3. Nous pouvons constater que le scripteur n'influe que très peu par rapport au tableau précédent. Ce qui nous permet de conclure que la différenciation entre deux alphabets fonctionne correctement et ce, quelque soit les scripteurs pris en compte.

Images de tests Base	Texte de test écrit en arabe	Texte de test écrit en français
Langue arabe	0,93	0,74
Langue française	0,89	0,87

Tableau 4.3 : Résultats dans le cas d'une personne bilingue

4.4. Vérification de signatures

Dans cette application, le problème réside sur le peu d'éléments contenus dans une signature. La recherche dans ce domaine est déjà bien avancée [PLA89] avec ses applications possibles dans le domaine juridique et à présent dans les télécommunications. Dans ce paragraphe nous avons appliqué notre méthode d'identification de scripteurs non plus sur des textes manuscrits mais sur des signatures. Les résultats obtenus sont très encourageants.

La méthode utilisée a été modifiée pour qu'elle puisse être appliquée aux signatures. L'extraction de caractéristiques par la compression fractale reste inchangée. C'est l'utilisation de ces caractéristiques qui est modifiée. Nous rappelons sur la Figure 4.13 notre méthode d'extraction de caractéristiques sur une image de textes manuscrits par le processus de compression fractale.

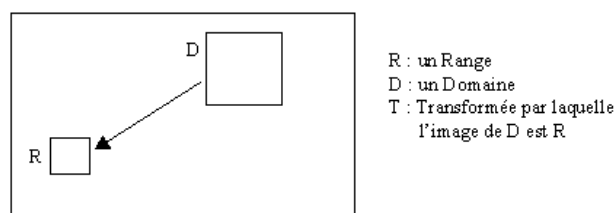


Figure 4.13 : Extraction de caractéristiques sur l'écriture par la compression fractale

La première approche a été de réutiliser le processus de compression et décompression fractales sur la base. Ainsi, en même temps que nous enregistrons comme éléments d'une base les Domaines, nous enregistrons aussi la transformée pour laquelle un Domaine est similaire à un Range. Ensuite, la décompression consiste à appliquer l'ensemble des transformées de chaque base sur l'image de la signature à reconnaître. Ce processus est itératif, c'est-à-dire que nous recommençons l'application des transformées sur l'image obtenue jusqu'à atteindre le point fixe. Ce point fixe est considéré comme obtenu lorsque numériquement le PSNR calculé entre l'image initiale et l'image décompressée ne varie plus.

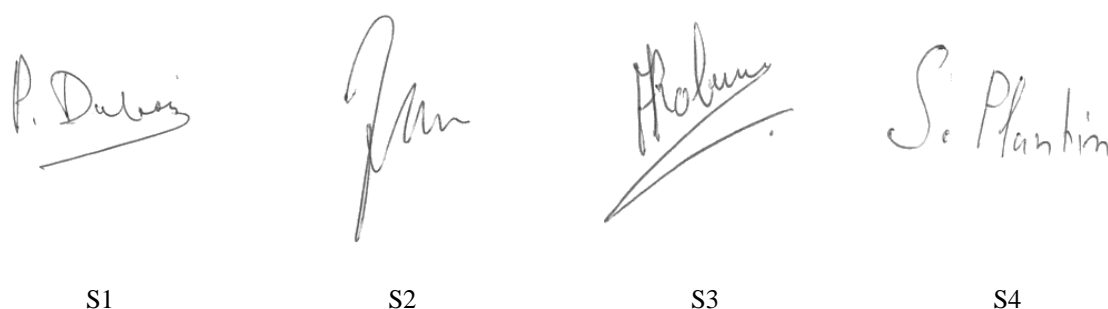


Figure 4.14 : Exemple de signatures

Notre base de données est constituée de 12 signatures. 4 scripteurs (cf. Figure 4.14) ont fourni chacun 3 fois une signature (cf. Figure 4.15). Les signatures sont normalisées et ont une taille de 256x256 sur 256 niveaux de gris. Cette normalisation est nécessaire pour appliquer les différentes transformations

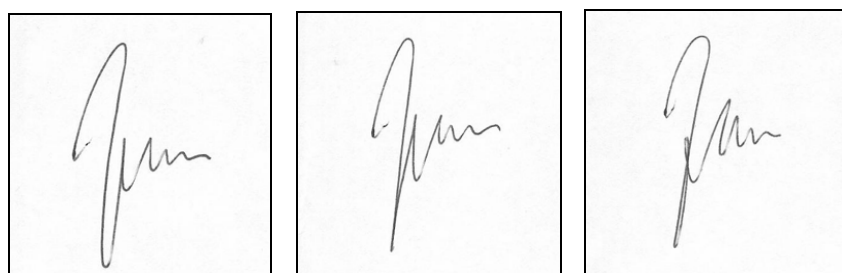


Figure 4.15 : Exemples de signatures de S2

Pour chaque scripteur, on dispose ainsi de 3 transformations caractéristiques du scripteur. On peut penser que le point fixe sera atteint d'autant plus rapidement que l'on part d'une image proche de celui-ci. Pour une signature inconnue, on la considère donc comme terme initial de la suite des itérés et on étudie la distance qui sépare ce premier terme et les premières approximations de l'attracteur d'un scripteur connu.

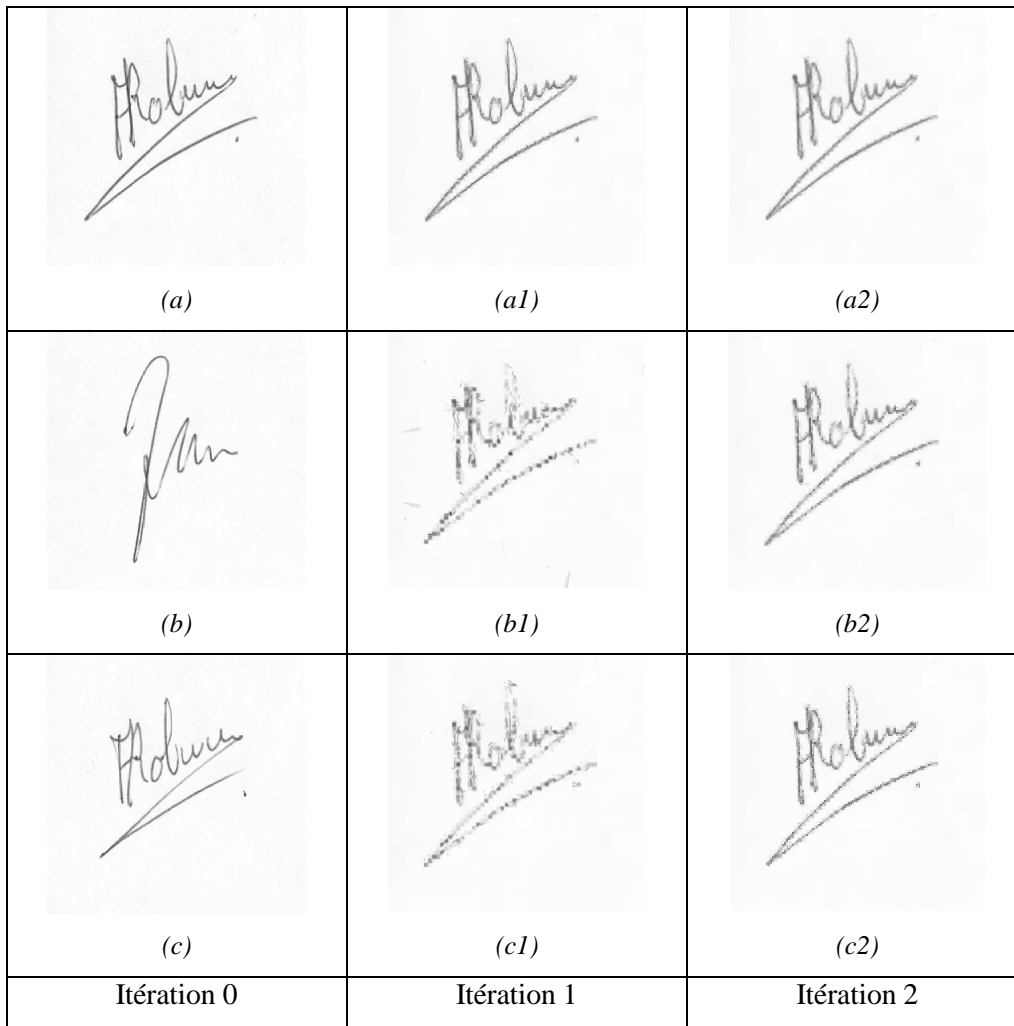


Figure 4.16 : Evolution des images décompressées pour 2 itérations avec pour image de départ (a) la signature correspondant à la base utilisée, (b) une signature quelconque, et (c) un autre exemplaire de la signature (a)

Sur la Figure 4.16, nous pouvons constater que les images résultant de la décompression fractale effectuée sur 2 itérations sont fortement influencées par le choix de l'image de départ. En effet le point fixe est plus vite atteint lorsque l'image de départ correspond à l'application des transformations obtenues par cette même image (cf. Figure 4.16(a)) pendant le processus de compression fractale que si une image quelconque est prise (cf. Figure 4.16(b)). Il en est de même si on prend comme image de départ un autre exemplaire de signature (cf. Figure 4.16(c)) du même scripteur.

La mesure de similarité entre une signature et la signature d'un scripteur qui a été traitée consiste à calculer l'erreur quadratique moyenne (RMS) entre l'image initiale et l'approximation de l'image décompressée à chaque itération, et ce, pour les trois premières itérations. Nous additionnons ensuite les 3 valeurs obtenues pour les RMS. Sur le Tableau 4.4, nous indiquons les résultats obtenus pour les 4 scripteurs. Pour chaque scripteur on a

considéré la signature n°1 et on a testé chacune des 12 bases obtenues dans l'étape d'apprentissage.

Signature de S1 N°1												
Signature testée	S1 N°1	S1 N°2	S1 N°3	S2 N°1	S2 N°2	S2 N°3	S3 N°1	S3 N°2	S3 N°3	S4 N°1	S4 N°2	S4 N°3
distance:	0	4,50	10,54	13,56	10,83	12,56	17,24	15,86	19,58	10,78	7,06	7,06
ordre:	0	1	4	8	6	7	10	9	11	5	2	3
Signature de S2 N°1												
Signature testée	S1 N°1	S1 N°2	S1 N°3	S2 N°1	S2 N°2	S2 N°3	S3 N°1	S3 N°2	S3 N°3	S4 N°1	S4 N°2	S4 N°3
distance:	10,54	8,478	8,855	0	5,085	6,42	11,33	14,78	15,67	9,126	8,703	5,296
ordre:	8	4	6	0	1	3	9	10	11	7	5	2
Signature de S3 N°1												
Signature testée	S1 N°1	S1 N°2	S1 N°3	S2 N°1	S2 N°2	S2 N°3	S3 N°1	S3 N°2	S3 N°3	S4 N°1	S4 N°2	S4 N°3
distance:	10,35	6,003	6,003	9,226	11,75	7,571	0	7,91	7,158	10,21	5,25	5,551
ordre:	10	3	4	8	11	6	0	7	5	9	1	2
Signature de S4 N°1												
Signature testée	S1 N°1	S1 N°2	S1 N°3	S2 N°1	S2 N°2	S2 N°3	S3 N°1	S3 N°2	S3 N°3	S4 N°1	S4 N°2	S4 N°3
distance:	6,24	9,51	9,01	11,52	15,32	14,61	8,55	9,73	7,41	0	4,82	5,50
ordre:	3	7	6	9	11	10	5	8	4	0	1	2

Tableau 4.4 : Résultats de la vérification de signatures

Nous avons classé les résultats selon l'ordre croissant des distances. Ainsi lorsque nous considérons la première signature de S1 et que nous utilisons la base extraite de S1 n°1, la signature dont la distance (i.e. la somme des RMS obtenues) est la plus petite est celle S1 n°2. Nous ne prenons évidemment pas en compte le résultat de S1 n°1, avec une distance logiquement égale à 0.

Sur les 4 signataires, trois ont au moins un de leur deux autres exemplaires de signatures qui permet de les authentifier. Seul le signataire S4 est totalement reconnu avec ses deux signatures classées au début. Par contre, nous pouvons remarquer qu'il est absorbant puisqu'il est classé deuxième pour tous les autres. Quant au signataire S3, la reconnaissance échoue.

L'absorption de S4 vient du fait que c'est le seul dont la signature soit lisible, c'est-à-dire qu'il a écrit son nom. Ainsi par cette méthode beaucoup plus de caractéristiques se retrouvent dans sa base et donc dans les caractéristiques communes avec les autres signataires.

Pour pallier cette absorption il faut éliminer les caractéristiques communes comme cela a été fait pour l'identification de personnes. Ce procédé améliorera de façon notable les résultats.

4.5. Caractéristiques d'images quelconques

Les problèmes de l'indexation, de la classification et de la recherche d'images est d'actualité. Nous montrons comment notre méthode peut y contribuer. Pour cela, nous l'avons appliqué sur des images quelconques, telles que des visages ou des paysages, pour savoir s'il était possible de les distinguer. Sur la Figure 4.17 se trouvent les 5 images que nous avons choisies. Elles sont toutes en 256 niveaux de gris et elles ont une taille de 256x256 pixels.



Figure 4.17 : Exemples d'images

La méthode utilisée est la même que celle de l'identification de scripteur, excepté que la partition en Ranges n'a pas été optimisée. En effet, il est inutile d'appliquer un découpage évolué puisque toutes les parties de l'image contiennent de l'information, c'est-à-dire qu'elles n'ont pas de fond blanc.

De plus, nous avons considéré que toutes les imasettes formant la base de référence de ces images participent à la représentation. Pour ce faire, nous avons choisi les seuils ε_s de façon à prendre en compte 100% des imasettes formant la base de référence lors de l'étape d'apprentissage.

Sur le Tableau 4.5 sont indiqués les coefficients de corrélation calculés entre l'image reconstruite par notre méthode et l'image initiale.

Tests \ Bases	Mandrill	Bateaux	Cameraman	Lena	Lena n°2	Poivrons	Tigre	Zelda
Mandrill	1,00	0,36	0,26	0,28	0,34	0,34	0,41	0,27
Bateaux	0,31	1,00	0,28	0,34	0,37	0,33	0,33	0,29
Cameraman	0,14	0,21	1,00	0,28	0,16	0,30	0,33	0,19
Lena	0,25	0,35	0,32	1,00	0,42	0,27	0,33	0,28
Lena n°2	0,25	0,32	0,29	0,45	1,00	0,25	0,37	0,30
Poivrons	0,21	0,29	0,20	0,36	0,22	1,00	0,34	0,20
Tigre	0,16	0,17	0,08	0,22	0,19	0,29	1,00	0,18
Zelda	0,23	0,22	0,09	0,20	0,26	0,37	0,35	1,00

Tableau 4.5 : Résultats de l'identification d'images quelconques

L'intérêt des résultats rassemblés dans le Tableau 4.5 réside dans les coefficients de corrélation calculés pour des images voisines. Malgré la ressemblance entre l'image de *Lena*, celle de *Lena n°2* et celle de *Zelda*, il n'y pas d'ambiguïté sur l'identification.

A présent, nous allons essayer de regrouper ces images par caractéristiques descriptives, i.e. différencier un visage d'un paysage. Pour cela nous comparons les imasettes de chaque base entre elles par le calcul de l'erreur quadratique moyenne. Il en résulte le Tableau 4.6 qui montre le pourcentage d'imasettes communes entre une base donnée (notée dans le tableau *BaseTests* et située sur les colonnes) et toutes les autres bases. Pour pouvoir obtenir des nombres normalisés par rapport aux différentes tailles des bases, nous avons calculé le nombre d'imasettes communes V comme suit :

$$V = \frac{P^2}{M \times N}$$

Avec M le nombre d'imasettes dans la base de *BaseTests*, N le nombre d'imasettes dans la base à comparer et P le nombre d'imasettes communes.

BaseTests \ Bases	Mandrill	Bateaux	Cameraman	Lena	Lena n°2	Poivrons	Tigre	Zelda
Mandrill	0,99	3,60	6,55	2,84	2,69	2,59	0,07	2,06
Tigre	0,07	1,15	2,48	0,87	1,01	0,76	0,59	0,64
Bateaux	3,60	13,03	20,26	10,74	10,50	9,84	1,15	8,81
Lena	2,84	10,74	17,49	10,42	8,80	8,82	0,87	7,81
Lena n°2	2,69	10,50	17,61	8,80	10,28	7,66	1,01	7,82
Poivrons	2,59	9,84	15,85	8,82	7,66	8,22	0,76	7,12
Zelda	2,06	8,81	14,44	7,81	7,82	7,12	0,64	7,69
Cameraman	6,55	20,26	31,70	17,49	17,61	15,85	2,48	14,44

Tableau 4.6 : Résultats de la ressemblance d'images quelconques par leur base

Quelle que soit la base de test utilisée pour la comparaison il ressort du Tableau 4.6 que trois groupes se forment. En effet, lorsque nous regardons les colonnes, les valeurs associées à l'image *Mandrill* et *Tigre* sont faibles comparées aux autres. On constate que les comparaisons à *Mandrill* et *Tigre* donnent un résultat plus faible sur toutes les autres colonnes. Ces valeurs forment notre premier groupe (valeurs surlignées en gris clair). Le deuxième groupe (valeurs surlignées en gris foncé) est formé par les images *Lena*, *Lena n°2*, *Bateaux*, *Poivrons* et *Zelda*. Et enfin le troisième groupe (en blanc) est représenté par l'image *Cameraman*.

Ainsi chaque groupe met en évidence une caractéristique commune de ces images. Le premier groupe a la particularité de contenir des images d'animaux prises de près, donc les détails sont plus gros et les traits sont rectilignes. Le deuxième groupe contient des images de visages mais aussi de poivrons et de bateaux. La caractéristique commune est le caractère rond de ces images. Et enfin le troisième groupe constitué d'une seule image est celle qui n'a pas pu être classée car elle contient à la fois des traits rectilignes et des traits arrondis.

Donc, nous venons de voir qu'il est possible par notre méthode de regrouper des images par caractéristiques. D'autres tests doivent encore être faits avec un plus grand nombre de variétés dans les images pour connaître les limites de l'application de cette méthode.

Pour tester la robustesse de cette méthode vis-à-vis du bruit et du décalage spatial, nous calculons le nombre V d'imagettes communes avec les 3 images de la Figure 4.18. Nous obtenons alors le Tableau 4.7.

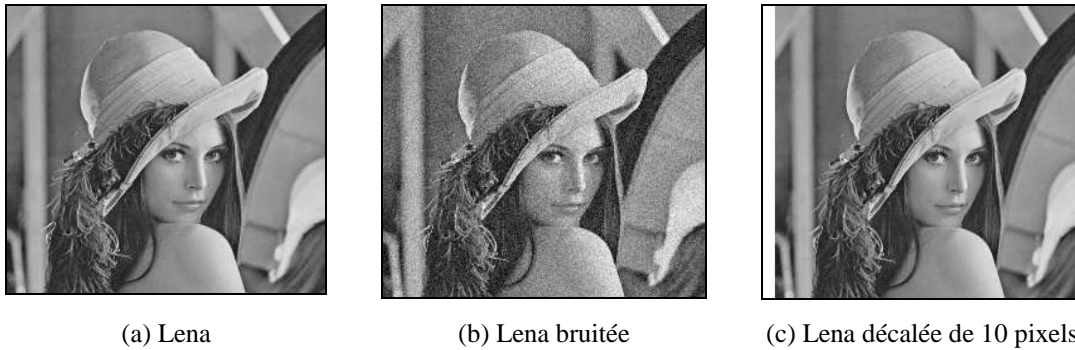


Figure 4.18 : Exemples d'images de (a) Lena (b) avec rajout d'un bruit uniforme ou (c) décalée de 10 pixels vers la droite

Image de test \ Base	Lena	Lena bruitée	Lena décalée
Lena	12,21	7,67	10,26
Lena bruitée	7,67	4,55	6.64
Lena décalée	10,26	6.64	9,21

Tableau 4.7 : Tests sur la robustesse avec calcul de V

Ainsi nous pouvons voir sur le Tableau 4.7 que l'ajout de bruit ou le décalage spatial influence les résultats. Néanmoins ces 3 images restent dans le même groupe de caractéristiques de la colonne *Lena* du Tableau 4.6.

4.6. Conclusion

Nous voulions savoir s'il était possible d'utiliser cette méthode pour d'autres applications. De plus, le contexte industriel veut qu'une méthode de reconnaissance d'écriture soit mise au point.

Dans un premier temps, nous avons mis au point une méthode qui utilise les informations extraites pendant la phase d'apprentissage du processus d'identification. Puisque chaque imagerie de la base est représentative du scripteur, nous avons supposé que ces images étaient aussi associées à un ensemble de graphèmes composant une lettre. C'est pour cela que notre méthode est une méthode analytique : à partir du regroupement de graphèmes nous formons une lettre qui, associée aux autres lettres forme le mot à reconnaître. Les premiers résultats ont montré que certaines lettres sont très différentes des autres telles que la lettre *l*. Toutefois beaucoup d'améliorations restent à faire car seulement 50% des lettres sont reconnues.

Ensuite, nous avons désiré rester dans le domaine de l'identification et de la vérification en appliquant notre méthode sur des signatures. Ce test n'est qu'un début pour la suite puisqu'il a été fait sur 4 personnes seulement. Mais il permet d'obtenir des résultats très encourageants avec une reconnaissance de 3 personnes sur 4. La méthode utilisée est différente de celle utilisée dans l'identification puisque nous avons appliqué dans ce cas le processus de décompression fractale. Cette méthode n'avait pas donné de résultats satisfaisants pour l'identification de scripteur en raison d'un trop grand nombre d'images dans la base. Ici une signature ne comporte qu'un nombre limité d'images (tout au plus 200) ce qui facilite les différenciations entre personnes.

Enfin, nous sommes sortis du domaine de l'écriture pour tester notre méthode d'identification sur des images quelconques. Un critère de regroupement a été instauré pour pouvoir comparer les images entre elles. Les résultats montrent qu'il est possible de former des groupes d'images selon leur caractéristique prédominante. La méthode peut ainsi servir à indexer des images. Les domaines d'application sont variés et peuvent concerner aux bases de données ou à Internet.

Une des premières améliorations qui peut être faite est celle de calculer le centre de gravité des images dans le cas des signatures et des images quelconques, pour annihiler les problèmes de décalage que nous avons mis en évidence.

Conclusion Générale

Conclusion Générale

L'objectif de cette thèse est fortement lié au contexte industriel. L'entreprise NEURONE Informatique désirait obtenir une chaîne complète d'analyse et de traitement de documents spécifiques liés au domaine médical. Ainsi la toute première étape fut une étape de conception de ces documents. Ils se composent de fiches d'analyses comportant une partie codifiée et une partie manuscrite. C'est à partir de ces fiches que le déroulement de la thèse s'est construit. Une première partie a consisté à mettre en forme les fiches pour pouvoir extraire les informations utiles telles que la détection de cases cochées permettant des analyses d'urine et/ou de sang et la récupération des zones manuscrites remplies par le médecin soignant. La seconde partie a consisté à traiter l'écriture manuscrite récupérée pour pouvoir d'une part identifier le scripteur, ce qui empêche toute contrefaçon et d'autre part reconnaître le texte écrit.

La mise en forme des documents scannés a été faite par des outils classiques dans le domaine du traitement de l'image tels que l'intercorrélation ou le seuillage d'images binarisées. Cela nous a permis d'effectuer les 3 tâches importantes de cette partie du traitement : le recalage de l'image scannée par rapport à une image de référence, la détection des cases cochées et donc le listing des analyses à faire liées à ces cases, et surtout la récupération du manuscrit de façon nette. Ces trois tâches sont effectuées correctement et nous avons aussi anticipé la demande de l'entreprise avec la possibilité d'ajouter d'autres fiches sans changer le déroulement général de cette étape. Evidemment d'autres techniques auraient pu être employées, mais le sujet de cette thèse n'étant pas centré sur cette étape, nous avons pris le parti de mettre au point un ensemble de processus permettant d'obtenir des résultats adaptés au contexte.

Après avoir extrait les zones manuscrites nous avons étudié comment chaque scripteur pouvait être identifié parmi les praticiens autorisés. Cette phase est primordiale dans un contexte où la sécurité est une des préoccupations majeures.

Nous avons mis au point une méthode originale d'identification de scripteur qui s'appuie sur l'extraction d'éléments invariants pour chaque scripteur par la compression fractale pendant l'étape d'apprentissage. Tous ces éléments forment une base de référence pour chaque scripteur. La grande difficulté a été de trouver le meilleur processus d'utilisation de ces bases. Par la suite, un processus de Pattern Matching entre le texte à identifier et toutes les bases enregistrées a permis d'obtenir un taux d'identification de plus de 85%. L'avantage de cette méthode est qu'elle est beaucoup plus rapide que la précédente. Ce taux d'identification est très satisfaisant et il permet de valider notre méthode. De plus, cette méthode peut s'appliquer à des textes autres que des textes occidentaux. Il suffit de changer la taille des Ranges et des Domaines pour l'appliquer au mieux à chaque type de textes tels que ceux en écriture asiatique ou arabe.

Ensuite, pour rester dans le contexte industriel, un module de reconnaissance d'écriture a été mis en œuvre. Il s'appuie sur une accumulation d'hypothèses par l'utilisation des bases de référence. Notre méthode s'inscrit dans les méthodes analytiques avec adaptation au scripteur dans le domaine de la reconnaissance d'écriture. Chaque élément de la base est représentatif du scripteur, alors en étiquetant ces éléments par rapport aux lettres du texte d'apprentissage, nous avons considéré que ces éléments étaient représentatifs du contenu. Cette démarche est originale et les résultats obtenus sont encourageants. Une amélioration sur le type d'étiquettes représentant une lettre doit être faite.

Enfin, nous avons tenté de généraliser notre méthode à d'autres domaines. Nous l'avons testée sur l'identification de signature, la différenciation entre les écritures arabe et française, et l'extraction de caractéristiques d'images quelconques. Ces tests ont été faits dans le but d'étudier leur faisabilité aussi ils peuvent être tous améliorés et optimisés. Les résultats obtenus montrent que notre méthode est généralisable à plusieurs domaines très variés.

Pour finir, nous allons énumérer quelques pistes intéressantes ouvertes par notre recherche. Tout d'abord le taux d'identification peut être amélioré en appliquant un découpage en imquettes rectangulaires ce qui serait plus adapté à l'écriture occidentale. Ensuite, l'étiquetage

du texte d'apprentissage pour la reconnaissance d'écriture doit être automatiser par une localisation précise des mots voire en utilisant une base de données de plusieurs copies de toutes les lettres de l'alphabet. Enfin, notre système de reconnaissance d'écriture pourrait être approfondi en choisissant plus de 4 positions pour chaque lettre.

Annexe 1

Exemples de pages manuscrites

Fred

Nous rappelons dans cette section les principales

Nous rappelons dans cette section les principaux effets de la théorie des systèmes de fonctions itératives permettant le codage et la synthèse d'images fractales.

L'écriture suit elle les règles de la géométrie fractale.

Aujourd'hui il me fait pas très beau.

Sabrina

Nous rappelons dans cette section les principaux effets de la théorie des systèmes de fonctions itératives permettant le codage et la synthèse d'images fractales.

L'écriture suit elle les règles de la géométrie fractale.

Aujourd'hui il me fait pas très beau.

Annexe 2

Résultats de l'identification de scripteurs

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18
S1	0.78	0.77	0.83	0.79	0.73	0.82	0.75	0.77	0.80	0.82	0.75	0.79	0.72	0.81	0.70	0.74	0.71	0.70
S2	0.74	0.77	0.83	0.77	0.74	0.83	0.73	0.79	0.79	0.78	0.75	0.79	0.75	0.79	0.65	0.75	0.74	0.69
S3	0.76	0.77	0.84	0.78	0.73	0.83	0.74	0.76	0.79	0.81	0.75	0.79	0.73	0.80	0.64	0.75	0.71	0.71
S4	0.77	0.78	0.83	0.80	0.73	0.83	0.74	0.77	0.79	0.79	0.75	0.77	0.73	0.81	0.68	0.77	0.75	0.73
S5	0.74	0.76	0.82	0.78	0.75	0.80	0.74	0.78	0.77	0.81	0.77	0.78	0.75	0.80	0.68	0.77	0.75	0.74
S6	0.78	0.77	0.84	0.78	0.74	0.84	0.74	0.76	0.79	0.81	0.77	0.76	0.73	0.81	0.65	0.75	0.73	0.67
S7	0.76	0.79	0.82	0.80	0.75	0.83	0.76	0.78	0.78	0.80	0.75	0.79	0.73	0.81	0.71	0.77	0.73	0.72
S8	0.77	0.79	0.82	0.78	0.73	0.81	0.74	0.78	0.78	0.80	0.77	0.76	0.73	0.80	0.72	0.76	0.73	0.72
S9	0.76	0.77	0.83	0.77	0.73	0.84	0.75	0.75	0.79	0.79	0.75	0.77	0.73	0.80	0.65	0.73	0.71	0.73
S10	0.77	0.81	0.83	0.80	0.73	0.82	0.75	0.77	0.78	0.81	0.75	0.77	0.75	0.81	0.70	0.75	0.75	0.71
S11	0.77	0.76	0.83	0.76	0.72	0.82	0.73	0.75	0.78	0.80	0.76	0.76	0.74	0.79	0.68	0.72	0.70	0.69
S12	0.78	0.78	0.81	0.79	0.74	0.83	0.74	0.74	0.79	0.80	0.77	0.76	0.72	0.81	0.62	0.74	0.72	0.70
S13	0.76	0.77	0.81	0.79	0.74	0.80	0.75	0.78	0.78	0.81	0.74	0.78	0.76	0.81	0.72	0.76	0.76	0.74
S14	0.78	0.79	0.84	0.78	0.75	0.83	0.76	0.78	0.79	0.81	0.77	0.80	0.74	0.81	0.67	0.75	0.77	0.74
S15	0.76	0.78	0.82	0.79	0.74	0.81	0.76	0.76	0.79	0.79	0.76	0.78	0.73	0.81	0.76	0.80	0.79	0.76
S16	0.76	0.78	0.82	0.79	0.75	0.81	0.73	0.76	0.79	0.80	0.75	0.78	0.73	0.78	0.73	0.79	0.79	0.76
S17	0.71	0.75	0.80	0.78	0.71	0.77	0.72	0.75	0.78	0.75	0.73	0.77	0.72	0.75	0.70	0.77	0.79	0.75
S18	0.75	0.79	0.80	0.77	0.73	0.81	0.76	0.76	0.77	0.78	0.76	0.79	0.74	0.80	0.76	0.79	0.78	0.78

Le tableau présente les résultats des coefficients de corrélation calculés entre une image de test (les colonnes) et les images reconstruites à partir de différentes bases (les lignes). Les bases ont été extraites selon la méthode décrites dans le chapitre 3.3 et la méthode d'identification est celle décrite dans le chapitre 3.4.

Annexe 3

Résultats de la vérification de signatures

scripteur	S11	S12	S13	S21	S22	S23	S31	S32	S33	S41	S42	S43	S51	S52	S53	S71	S72	S73
Base de S11																		
distance	0	4,50	10,54	13,56	10,83	12,56	17,24	15,86	19,58	10,78	7,06	7,06	8,56	7,74	10,15	9,43	11,55	7,61
ordre:	0	1	9	13	11	12	16	15	17	10	2	3	7	5	8			
Base de S21																		
distance	10,54	8,48	8,85	0,00	5,08	6,42	11,33	14,78	15,67	9,13	8,70	5,30	13,39	14,73	11,30	14,73	10,26	11,33
ordre:				0	1	5						2						
Base de S31																		
distance	10,35	6,003	6,003	9,226	11,75	7,571	0	7,91	7,158	10,21	5,25	5,551	12,83	13,10	8,88	11,71	10,19	10,26
ordre:		3	4			7	0	8	6		1	2						
Base de S41																		
distance	6,24	9,51	9,01	11,52	15,32	14,61	8,55	9,73	7,41	0,00	4,82	5,50	9,45	12,52	12,02	8,69	10,29	8,26
ordre:										0	1	2						
Base de S51																		
distance	22,30	20,27	22,87	19,68	18,27	20,59	26,89	24,91	28,36	29,44	23,24	24,77	0,00	19,04	17,83	20,49	16,64	19,57
ordre:					3								0	4	2		1	
Base de S71																		
distance	18,11	25,68	22,97	30,89	21,80	22,75	26,59	28,20	29,55	23,07	20,66	18,70	22,05	17,22	17,34	0,00	16,10	13,25
ordre:																0	2	1

Images scannées à 200ppp puis redimensionnées à 256X256.

Bibliographie

- [ADA01] S. ADAM, *"Interprétation de documents techniques : des outils à leur intégration dans un système à base de connaissances"*, Thèse soutenue à Rouen le 11/12/2001, Jury : VINCENT N., CHASSERY JM., TOMBRE K., OGIER J.M., LABICHE J., GARDES J., pp 206 (2001).
- [ARA77] B. Arazi, *"Handwriting identification by means of run-length measurements"*, IEEE Transactions on Systems, Man and Cybernetics, vol.SMC-7, no.12, pp. 878-881, Dec. 1977.
- [AUG01] E. Augustin, *"Reconnaissance de mots manuscrits par systèmes hybrides Réseaux de Neurones et Modèles de Markov Cachés"*, thèse de l'Université René Descartes, Mai 2001.
- [BAR88] M. F. Barnsley, *"Fractals Everywhere"*, Academic Press, New York, 1988.
- [BEN02] A. Bensefia, A. Nosary, T. Paquet, L. Heutte, *"Writer identification by writer's invariants"*, Eight International Workshop on Frontiers in Handwriting Recognition (IWFHR'02), Niagara-on-the-Lake, Canada, August 6-8; pp. 274-279, 2002.
- [BOU97] V. Boulétreau, *"Vers un classement de l'écrit par des méthodes fractales"*, Thèse de doctorat 1997.
- [BOU98] V. Bouletreau, N. Vincent, R. Sabourin, H. Emptoz, *"Handwriting and signature : one or two personality identifiers?"*, Proceedings Fourteenth International Conference on Pattern Recognition (ICPR), Los Alamitos, CA, vol.2, pp. 1758-1760, 1998.
- [CAM92] J. Camillerapp, G. Lorette, G. Menier, H. Oulhadj, J. Pettier, *"Off-line and on-line methods for cursive handwriting recognition"*, In S. Impedovo and J.-C.

- Simon, editors, *From Pixels to Features III: Frontiers in Handwriting Recognition*, Elsevier Science Publishers B.V., pp. 273-287, 1992.
- [CAS96] R. Casey, E. Lecolinet, "*A Survey of Methods and Strategies in Character segmentation*", *IEEE-Pattern Analysis and Machine Intelligence*, Vol. 18, pp. 690-706, 1996.
- [CES03] F. Cesarini, E. Francesconi, M. Gori, G. Soda, "*Analysis and understanding of multi-class invoices*", *International Journal on Document Analysis and Recognition (IJ DAR)*, vol. 6, No. 2, pp. 102-114, 2003.
- [CHE01] J.L. Chen, H.J. Lee, "*Field data extraction for form document processing using a gravitation-based algorithm*", *Pattern Recognition*, vol. 34, No. 9, pp. 1741-1750, September 2001.
- [CLA01] E. Clavier, "*Etude des stratégies de tri : application à un système de tri de formulaires*", thèse de l'Université de Caen, Soutenue le 21 décembre 2001.
- [COT96] M. Côté, E. Lecolinet, M. Cheriet, C.Y. Suen, "*Lecture automatique d'écriture cursive utilisant des concepts perceptuels*", *Actes du congrès de l'Association canadienne-française pour l'avancement de la science (ACFAS)*, Montréal, Canada, pp. 131-140, mai 1996.
- [COT97] Myriam Côté, "*Utilisation d'un modèle d'accès lexical et de concepts perceptifs pour la reconnaissance d'images de mots cursifs*", Thèse de doctorat, Département signal, Ecole nationale supérieure des télécommunications de Paris, Paris, France, 178 pages, 6 juin 1997.
- [COU96] B. Couïasnon, "*Segmentation et reconnaissance de documents guidées par la connaissance a priori : application aux partitions musicales*", Thèse de l'université de Rennes 1, Janvier 1996.

- [CRE94] JP Crettez, "*Premier degré de caractérisation des écritures manuscrites : essai de regroupement des écritures en familles*", Conférence Nationale sur l'Écrit et le Document (CNED'94), Rouen, France, pp. 71-80, 1994.
- [CRE95] J.-P. Crettez, "*A set of handwriting families : style recognition*", International Conference on Document Analysis and Recognition (ICDAR'95), Montréal, Canada, pp. 489-494, 1995.
- [CRE96] J. P. Crettez, M. Gilloux, M. Leroux, "*Que dit la « main » du scripteur aux « yeux » du lecteur ?*", 4^{ème} Colloque National sur l'Écrit et le Document (CNED'96), pp. 291-296, Nantes, France, Juillet 1996.
- [DUO02] J. Duong, M. Côté, H. Emptoz, "*Features for Printed Document Image Analysis*", 16th International Conference on Pattern Recognition (ICPR'2002), Québec (Canada), 11-15, vol. III, pp. 245-248, August 2002.
- [FAI02] Y. Fataïcha, J.Y. NIE, M. Cheriet, C.Y. Suen, "*Content Analysis in Document Images: A Scale Space Approach*", In the proceedings of the 16 th International Conference on Pattern Recognition (ICPR'2002) Volume 3, pages 335-338, Québec, Canada, August 2002.
- [FIS95] Y. Fisher, "*Fractal Image Compression : Theory and Application*", Springer-Verlag, New York, 1995.
- [GAR96] S. Garcia-Salicetti, B. Dorizzi, P. Gallinari, Z. Wimmer, S. Gentric, "*Segmentation Dynamique et Réseaux de Neurones Prédicatifs pour la Reconnaissance de Mots en Ligne*", Conférence Nationale sur l'Écrit et le Document (CNED '96), Nantes, France, July 1996.
- [GIL93] M. Gilloux, M. Leroux, J.-M. Bertille, "*Strategies for Handwritten Words Recognition Using Hidden Markov Models*", Proceedings of 2nd International

Conference on Document Analysis and recognition (ICDAR), Tsukuba, Japan, pp. 299-304, 20-22 octobre 1993.

- [GIL94] M. Gilloux, "*Writer adaptation for handwritten word recognition using hidden Markov models*", 12th International Conference on Pattern Recognition (IAPR), Los Alamitos, USA, pp. 135-139 vol.2, 1994.
- [GUI95] D. Guillevic, C. Y. Suen, "*Cursive Script Recognition applied to the Processing of Bank Cheques*", International Conference on Document Analysis and Recognition (ICDAR'95), Montreal, Canada, pp.11-14, August 1995.
- [GRA00] F. Grandidier, R. Sabourin, C.Y. Suen, M. Gilloux, "*Une nouvelle stratégie pour l'amélioration des jeux de primitives d'un système de reconnaissance de l'écriture*", Actes du 2ème Colloque International Francophone sur l'Écrit et le Document (CIFED'2000), Lyon, juillet 2000.
- [HEU00] L. Heutte, T. Paquet, A. Nosary, C. Hernoux, "*Handwritten text recognition using a multiple-agent architecture to adapt the recognition task*", 7th International Workshop on Frontiers in Handwriting Recognition (IWFHR VII), Amsterdam, pp. 413-422, 2000.
- [HER01] P. HEROUX, "*Contribution au problème de la rétro-conversion des documents structurés*", Thèse de doctorat de l'Université de Rouen, 4 janvier 2001; Jury : Rolf Ingold, Jean-Michel Jolion, Jaime Lopez-Krahe, Yves Lecourtier, Maurice Milgram, Eric Trupin; 180 pages (2001).
- [JAC92] E. W. Jacobs, Y. Fisher, R. D. Boss, "*Image Compression : A study of the iterated transform method*", Signal Processing, vol. 29, n°3, pp. 251-263, 1992.

- [JAC90] A. E. Jacquin, "*Fractal image coding based on a theory of iterated contractive image transformations*", Visual Communications and Image Processing (VCIP'90), SPIE Proceedings, Vol. 1360, pp. 227-239, October 1990.
- [JAI00] A. K. Jain, R. P.W. Duin, J. Mao, "*Statistical Pattern Recognition : A Review*", IEEE Transactions on Pattern Analysis and Machine intelligence (PAMI), n°1, vol. 22, pp. 4-37, January 2000.
- [KEB98] S. Kebairi, B. Taconet, A.Zahour, S. Ramdane, "*A Statistical Method For an Automatic Detection of Form types*", Proceedings of the (DAS'98), Nagano, Japan, November 4-6, pp.109-118, 1998.
- [KHA96] K. Khan, I.K. Sethi, "*Handwritten signature retrieval and identification*", Pattern Recognition Letters, vol.17, pp. 83-90, 1996.
- [KLE80] V. Klement, R.-D. Naske, K. Steinke, "*The application of image processing and pattern recognition techniques to the forensic analysis of handwriting*", Proceedings of the Third International Conference Security Through Science and Engineering, Lexington, Kentucky, USA, pp. 5-11, 1980.
- [KLE81] V. Klement, "*Forensic writer recognition*", Digital Image Processing. Proceedings of the NATO Advanced Study Institute, Dordrecht, Netherlands Reidel, pp. 519-524, 1981.
- [KLE83] V. Klement, "*An Application System for the Computer-Assisted Identification of Handwritings*", Proceedings of the International Carnahan Conference on Security Technology, Lexington, Kentucky, USA, pp. 75-79, 1983.
- [KUC79] W. Kuckuck, B. Rieger, K. Steinke, "*Automatic writer recognition*", Proceedings of the 1979 Carnahan Conference on Crime Countermeasures, Lexington, Kentucky, USA, pp. 57-64, 1979.

- [KUC80] W. Kuckuck, "*Writer recognition by spectral analysis*", Proceedings of the Third International Conference Security Through Science and Engineering, Lexington, Kentucky, USA, pp. 1-3, 1980.
- [LEC94] E. Lecolinet, O. Baret, "*Cursive Word Recognition : Methods and Strategies*", in Fundamentals in Handwriting Recognition, S. Impedovo Ed., NATO ASI Series F: Computer and Systems Sciences, Vol. 124, Springer Verlag, pp. 235-263, 1994.
- [LER91] M. Leroux, J. C. Salome, J. Badard, "*Recognition of cursive script words in a small lexicon*", In Proceedings International Conference on Document Analysis and Recognition (ICDAR), pp. 774-782, 1991.
- [MAD93] S. Madhvanath, V. Govindaraju, "*Holistic lexicon reduction*", Proceedings International Workshop on Frontiers in Handwriting Recognition (IWFHR), pp. 71-81, 1993.
- [MAN77] B. Mandelbrot, "*The Fractals Geometry of Nature*", W. H. Freeman and Co., San Francisco, 1977.
- [MAR01] U.-V. Marti, R. Messerli, H. Bunke, "*Writer identification using text line based features*", Proceedings. 6th International Conference on Document Analysis and Recognition (ICDAR), Seattle, USA, pp. 101-105, 2001.
- [MIH77] F. Mihelic, N. Pavesic, L. Gyergyek, "*Recognition of writers of handwritten texts*", International Conference On Crime Countermeasures, pp. 237-240, 1977.
- [MOR91] J.-V. Moreau, B. Plessis, O. Bougeois, J.-L. Plagnaud, "*A postal check reading system*", Proceedings International Conference on Document Analysis and Recognition (ICDAR), pp. 758-766, 1991.

- [NAG00] G. Nagy, *"Twenty Years of Document Image Analysis in PAMI"*, invited, IEEE Trans. PAMI-22, 1, pp. 38-62, 20th Anniversary Issue, January 2000.
- [NAS80] R.-D. Naske, *"Application of a weighted least squares algorithm to writer and speaker recognition"*, Proceedings of the 5th International Conference on Pattern Recognition (ICPR), New York, USA, pp. 27-30, 1980.
- [NAS82] R.-D. Naske, *"Writer recognition by prototype related deformation of handprinted characters"*, Proceedings of the 6th International Conference on Pattern Recognition (ICPR), New York, USA, vol.2, pp. 819-22, 1982.
- [NOS99] A. Nosary, L. Heutte, T. Paquet, Y. Lecourtier, *"Defining writer's invariants to adapt the recognition task"*, International Conference on Document Analysis and Recognition (ICDAR'99), Bangalore (India), pp. 765-768, 1999.
- [NOS02] A. Nosary, *"Reconnaissance Automatique de Textes Manuscrits par Adaptation au Scripteur"*, Thèse de Doctorat de l'Université de Rouen; p. 220, 16 Janvier 2002.
- [OLI02] J.J. de Oliveira Jr, J.M. de Carvalho, C.O. de A. Freitas, R. Sabourin, *"Evaluating NN and HMM Classifiers for Handwritten Word Recognition"*, Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI'02), Fortaleza, Brazil, pp. 210-217, 07 - 10 October 2002.
- [PAQ91] T. Paquet, Y. Lecourtier, *"Handwriting recognition: Application on bank cheques"*, In Proceedings International Conference on Document Analysis and Recognition (ICDAR), pp. 749-757, 1991.
- [PAQ99] T. Paquet, M. Avila, O. Colot, *"Word Modeling for Handwritten Word Recognition"*, Vision Interface'99, Trois Rivières, pp. 49-55, 1999.
- [PAR98] F. Parmentier, *"Spécification d'une architecture émergente fondée sur le raisonnement par analogie. Application aux références bibliographiques"*,

- Thèse de l'Université Henri Poincaré-Nancy 1, 9 juin 1998. [PAR96] C. Parisse, "*Global word shape processing in off-line recognition of handwriting*", IEEE Transactions on Pattern Analysis and Machine Intelligence, 18(4), pp. 460-464, 1996.
- [PAR90] M. Parizeau, R. Plamondon, "*A Comparative Analysis of Regional Correlation, Dynamic Time Warping, and Skeletal Tree Matching for Signature Verification*", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12, pp. 710-717, July 1990.
- [PAR93] M. Parizeau, R. Plamondon, "*Allograph Adjacency Constraints for Cursive Script Recognition*", Proc. of the Third International Workshop on Frontier in Handwriting Recognition (IWFHR), pp. 252-261, May 25-27 1993.
- [PAR95] M. Parizeau, R. Plamondon, "*A Fuzzy-Syntactic Approach to Allograph Modeling for Cursive Script Recognition*", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 17, pp. 702-712, July 1995.
- [PLA89] R. Plamondon, G. Lorette, "*Automatic Signature Verification and Writer Identification –The State of the Art*", Pattern Recognition, Vol. 22, N° 2, pp. 107-131, 1989.
- [RAB88] L. R. Rabiner, "*Mathematical Foundation of Hidden Markov Models*", Journal Expert Systems, Vol. 15, No.1, pp. 48-56, February 1988.
- [RYA93] M. S. Ryan, G. R. Nudd, "*The Viterbi Algorithm*", Research Report, Department of Computer Science, University of Warwick, Number CS-RR-238, February 1993.
- [SAI98] H.E.S. Said, G.S. Peake, T.N. Tan, K.D. Baker, "*Writer identification from non-uniformly skewed handwriting images*", Proceedings of the Ninth British

Machine Vision Conference (BMVC 98), Southampton, England, vol.2, pp. 478-87 1998.

- [SAI00] H.E.S. Said, T. N. Tan, K. D. Baker, "*Personal identification based on handwriting*", Pattern Recognition, Volume 33, Issue 1, pp. 149-160, January 2000.
- [SAY73] K. Sayre, "*Machine Recognition of Handwritten Words: a project report*", Pattern Recognition, vol.5, n°3, pp. 213-228, 1973.
- [SER02] A. Seropian, N. Vincent, "*Writers Authentication and fractal Compression*", 8th International Workshop on Frontiers in Handwriting Recognition (IWFHR), Niagara-on-the-Lake (Canada), pp. 434-439, 2002.
- [SER03a] A. Seropian, M. Grimaldi, N. Vincent, "*L'autosimilarité interne, un moyen de sélectionner une base de formes représentatives*", Journées Francophones des Jeunes Chercheurs en Vision par Ordinateur, ORASIS'03, Gerardmer, France, Mai 2003.
- [SER03b] A. Seropian, M. Grimaldi, N. Vincent, "*Writer Identification based on the fractal construction of a reference base*", 7th International Conference on Document Analysis and Recognition, ICDAR'03, Edimburgh, Grande Bretagne, pp. 1163-1167, Août 2003.
- [SEY98] M. J. Sedeyn, "*Introduction à l'examen objectif des écritures manuscrites*", Éditions Fovea, p. 110, 1998.
- [SHI03] M. Shilman, Z. Wei, S. Raghupathy, P. Simard, D. Jones, "*Discerning Structure from Freeform Handwritten Notes*", 7th International Conference on Document Analysis and Recognition, ICDAR'03, Edimburgh, Grande Bretagne, pp. 60-65, Août 2003.

- [SIM85] J.-C. Simon, *"La Reconnaissance des Formes par Algorithmes"*, Editions Masson, p. 252, 1985.
- [SRI01] S. N. Srihari, S.-H. Cha, H. Arora, S. Lee, *"Individuality of Handwriting: A Validation Study"*, 6th International Conference on Document Analysis and Recognition (ICDAR'01), Seattle, USA, September 10-13, pp. 106-109, 2001.
- [STE99] T. Steinherz, E. Rivlin, N. Intrator, *"Off-line cursive script word recognition -- A survey"*, International Journal on Document Analysis and Recognition, vol.2, no.2-3, pp. 90-110, 1999.
- [STE81] K. Steinke, *"Recognition of writers by handwriting images"*, Pattern Recognition, vol.14, no.1-6, pp. 357-64, 1981.
- [TAN02] Y. Y. Tang, Yu Tao, E. C. M. Lam, *"New method for feature extraction based on fractal behavior"*, Pattern Recognition, Volume 35, Issue 5, pp. 1071-1081, May 2002.
- [TAO01] Yu Tao, Yuan Y. Tang, *"Discrimination of Oriental and Euramerican Scripts Using Fractal Feature"*, 6th International Conference on Document Analysis and Recognition 2001 (ICDAR'01), Seattle, USA, September 10-13, pp. 1115-1119, 2001.
- [TRI96] O. D. Trier, A. K Jain, T. Taxt, *"Feature extraction methods for Character Recognition - A Survey"*, Pattern Recognition, vol 29, pp. 641-662, 1996.
- [VIN94] N. Vincent, H. Emptoz, *"L'intérêt d'une approche fractale dans la reconnaissance de l'écriture manuscrite"*, Conférence Nationale sur l'Écrit et le Document (CNED'94), pp. 43-52, Rouen, 1994.
- [VIN95a] N. Vincent, H. Emptoz, *"A Classification of writings based on fractals"*, Fractal Reviews in the Natural and Applied Sciences, M.M. Novak Editor, London, Chapman and Hall, pp.320-331, 1995.

- [VIN95b] N. Vincent, V. Boulétreau, H. Emptoz, *"A fractal analysis of handwritten texts"*, progress in handwriting recognition, World Scientific, pp. 581-586, 1995.
- [VIN98] N. Vincent, S. Barbezieux, *"Compression of handwriting images : a way to define a writing style"*, Vision Interface, pp. 347-354, 1998.
- [VIN00] N. Vincent, V. Boulétreau, R. Sabourin, H. Emptoz, *"How to use fractal dimensions to qualify writings and writers, Fractals"*, World Scientific, Vol 8, n°1, pp.85-97, 2000.
- [VIN02] A. Vinciarelli, *"A survey on Off-Line Cursive Script Recognition"*, Pattern Recognition, Vol 35, no. 7, pp. 1433-1446, June 2002.
- [WAG74] R. A. Wagner, M. J. Fischer, *"The String-to-string correction problem"*, Journal of the Association for Computing Machinery (JACM), Vol. 21, No. 1, pp. 168-173, Janvier 1974.
- [WIR03] M. Wirotius, A. Seropian, N. Vincent, *"Writer Identification from Gray Level Distribution"*, 7th International Conference on Document Analysis and Recognition (ICDAR'03), Edimburgh, Grande Bretagne, pp. 1168-1172, Août 2003.
- [YIN01] P.-Y. Yin, *"Skew detection and block classification of printed documents"*, Image and Vision Computing, n°19, pp. 567-579, 2001.
- [YUA03] B. Yuan, C. L. Tan, *Skewscope: The Textual Document Skew Detector*, 7th International Conference on Document Analysis and Recognition (ICDAR'03), Edimburgh, Grande Bretagne, pp. 49-53, 2003. [ZOI00] E. N. Zois, V. Anastassopoulos, *"Morphological waveform coding for writer identification"*, Pattern Recognition, Volume 33, Issue 3, pp. 385-398, March 2000

Publications de l'auteur

Congrès internationaux avec comité de lecture et actes

A. Seropian, N. Vincent, "*Writers Authentication and Fractal Compression*", International Workshop on Frontiers in Handwriting Recognition IWFHR'02, Niagara-on-the-lake, Canada, Août 2002; pp. 434-439. Obtention du prix du meilleur poster.

A. Seropian, M. Grimaldi, N. Vincent, "*Writer Identification based on the fractal construction of a reference base*", 7th International Conference on Document Analysis and Recognition ICDAR'03, Edimburgh, Grande Bretagne, pp. 1163-1167, Août 2003.

M. Wirotius, A. Seropian, N. Vincent, "*Writer Identification from Gray Level Distribution*", 7th International Conference on Document Analysis and Recognition ICDAR'03, Edimburgh, Grande Bretagne, pp. 1168-1172, Août 2003.

Congrès nationaux avec comité de lecture et actes

A. Seropian, M. Grimaldi, N. Vincent, "*L'autosimilarité interne, un moyen de sélectionner une base de formes représentatives*", Journées Francophones des Jeunes Chercheurs en Vision par Ordinateur ORASIS'03, Gerardmer, France, pp. 179-187, Mai 2003.

Congrès nationaux sans comité de lecture et sans actes

A. Seropian, N. Vincent, "*Reconnaissance d'écriture hors-ligne et Compression fractale*", Journées thématiques, Document Ecrit organisé par GDR I3-ISDN-GRCE, Lyon, 26 juin 2001.

A. Seropian, M. Grimaldi, N. Vincent, "*Authentication de Scripteurs par une Base de Formes Extraites par Autosimilarité*", Journées thématiques, Document Ecrit organisé par GDR I3-ISDN-GRCE, Tours, 5 juin 2003.